Ref. Ares(2025)2732753 - 04/04/2025

DiMAT

D3.3 VIEWPOINTS

04/04/2025

Grant Agreement No.: 101091496
Call: HORIZON-CL4-2022-RESILIENCE-01
Topic: HORIZON-CL4-2022-RESILIENCE-01-25
Type of action: HORIZON Innovation Actions

D3.3 VIEWPOINTS

Grant agreement number	101091496	Acronym	DiMAT
Full title	Digital Modelling and Simulation for Design, Processing and Manufacturing of Advanced Materials		
Start date	01/01/2023	Duration	36 months
Project url	HTTPS://CORDIS.EUROPA.EU/PROJECT/ID/101091496		
Work package	WP3 - DESIGN: DiMAT Framework Design		
Deliverable	D3.3 – DiMAT Viewpoints		
Task	T3.2 – Business Viewpoint T3.3 – Usage Viewpoint T3.4 – Functional Viewpoint T3.5 – Implementation Viewpoint		
Due date	30/09/2023		
Submission date	04/04/2025		
Nature	Report	Dissemination level	Public
Deliverable lead	2-UPV		
Version	1.5		

Co-funded by the European Union

Authors	Miguel A. Mateo-Casalí (UPV), Andres Boza (UPV), Ana Esteso Álvarez (UPV)
Contributions	Konstantinos Tsitseklis (NTUA), Ignacio Rico (AITEX)
Reviewers	Gabriela Candea (ROPARDO), Ciprian Candea (ROPARDO), Leonardo Cosma (CETMA)
Abstract	This document presents the first version of the different views of the DiMAT Reference Architecture (RA). Four views have been presented (Business, Usability, Functional and Implementation), discussed throughout the deliverable and further reviewed in deliverable D3.4.
Keywords	IIRA, VIEWPOINTS, REFERENCE ARCHITECTURE, DESIGN

Document Revision History

Version	Date	Description of change	List of contributor(s)
0.1	14-Jul-2023	ToC	UPV
0.2	08-Sep-2023	1st Draft	UPV, AITEX, NTUA
0.3	11-Sep-2023	Comments by partners	All partners
0.4	18 -Sep-2023	Internal review	CETMA, ROPARDO
0.5	22-Sep-2023	2nd Draft addressing the comments from internal reviewers	UPV
1.0	29-Sep-2023	Final quality check and issue of final document	CERTH
1.1	15-Nov-2024	Update of deliverable according to PO/reviewers' comments and quality check	CERTH, UPV
1.2	28-Jan-2025	Additional update of the	CERTH

		deliverable according to PO/reviewers' comments	
1.3	4-Feb-2025	Final quality check and issue of final document	CERTH, UPV
1.4	25-Feb-2025	Additional update of the deliverable according to PO/reviewers' comments	CERTH
1.5	4-Apr-2025	Additional update of the deliverable according to PO/reviewers' comments on section 2	CERTH, UPV

DISCLAIMER

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Health and Digital Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

COPYRIGHT NOTICE

© DiMAT Consortium, 2023

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

EXECUTIVE SUMMARY

This document presents the first version of the different views of the DiMAT Reference Architecture (RA). Four views have been presented (Business, Usability, Functional and Implementation), discussed throughout the deliverable and further reviewed in deliverable D3.4.

From the business perspective, the aim was to provide a business perspective within the DiMAT design activities to avoid the risk of a "technology-centric" approach. To this end, concerning the IIRA approach, the task focused on framing the stakeholders' identification and point of view, their vision, values, and key objectives. Stakeholders were defined considering the perspectives and experiences of the companies participating in the project (pilots), which provided a global view of a company's processes and the relevant actors to which they relate.

From a usability point of view, the focus was on defining the capabilities and structure of all the actors that will interact with the solutions, identifying several key points from a usability point of view, such as tasks, activities, or functions, always considering the human factor that will interact with these tools.

The third part of this deliverable will analyze the functional part of the DiMAT solutions. For this purpose, and focusing on the development part of the solutions, their structure, interrelations, and interfaces have been considered.

Finally, in the implementation part, we will technically describe the different components of the DiMAT solutions and how they are interconnected, and we will select the technologies necessary for their correct implementation.

This deliverable is an updated version that addresses all comments provided by the reviewer. To ensure clarity and transparency, we have created a detailed table summarizing the changes made to this document in response to the reviewer's feedback. The Table 118 can be found in the appendix, providing an overview of the revisions and additions for easy reference.

TABLE OF CONTENTS

1 METHODOLOGICAL APPROACH 22
1.1 Introduction 22
1.2 Viewpoints 22
1.3 DIMAT RA: Viewpoints 24
2 BUSINESS VIEWPOINT 26
2.1 Business viewpoint approach for DiMAT 26
2.1.1 Business Viewpoint Concepts 26
2.1.2 Business Viewpoint Methodology 27
2.1.3 Questionnaire (Form) Design 28
2.2 DIMAT Business viewpoint 30
2.2.1 Stakeholder identification 30
2.2.2 Stakeholders' interest in toolkits 31
2.2.3 Business Decision Makers Domain 32
2.2.4 Technical Leaders Domain 41
2.3 DiMAT Business viewpoint conclusions 49
3 USER VIEWPOINT 52
3.1 Usage VIEWPOINT approach for DiMAT 52
3.1.1 Usage Viewpoint Concepts 52
3.1.2 Usage Viewpoint Methodology 53
3.2 DiMAT Cloud Materials Database (Di^{CMDB}) 58
3.2.1 Overview 58
3.2.2 Roles 59
3.2.3 Activities 59
3.3 DiMAT Knowledge Acquisition Framework (Di^{KAF}) 67
3.3.1 Overview 68

3.3.2 Roles 68
3.3.3 Activities 68
3.4 DiMAT Materials Environmental and Cost Life Cycle Assessment ($Di^{MEC\cdot LCA}$) 84
3.4.1 Overview 84
3.4.2 Roles 85
3.4.3 Activities 86
3.5 DiMAT Materials Design Framework (Di^{MDF}) 97
3.5.1 Overview 97
3.5.2 Roles 98
3.5.3 Activities 98
3.6 DiMAT Materials Modeler (Di^{MM}) 104
3.6.1 Overview 104
3.6.2 Roles 104
3.6.3 Activities 106
3.7 DiMAT Materials Designer (Di^{MD}) 116
3.7.1 Overview 117
3.7.2 Roles 117
3.7.3 Activities 117
3.8 DiMAT Materials Mechanical Properties Simulator (Di^{MMS}) 122
3.8.1 Overview 122
3.8.2 Roles 123
3.8.3 Activities 124
3.9 DiMAT Materials Processing Simulator (Di^{MPS}) 129
3.9.1 Overview 130
3.9.2 Roles 130
3.9.3 Activities 131
3.10 DiMAT Digital Twin for Process Control (DiDTPC) 139
3.10.1 Overview 140

3.10.2 Roles 140
3.10.3 Activities 141
4 FUNCTIONAL VIEWPOINT 159
4.1 FUNCTIONAL VIEWPOINT approach for DiMAT 159
4.1.1 Functional Viewpoint Concepts 159
4.1.2 Functional Viewpoint Methodology 159
4.1.3 Diagram of components 160
4.2 DiMAT Cloud Materials Database (Di^{CMDB}) 164
4.2.1 General architecture 164
4.2.2 Functional components 164
4.2.3 Interactions 168
4.2.4 Mapping of functional components to IIRA functional domain 173
4.3 DiMAT Knowledge Acquisition Framework (DI^{KAF}) 174
4.3.1 General architecture 174
4.3.2 Functional components 174
4.3.3 Interactions 176
4.3.4 Mapping of functional components to IIRA functional domain 179
4.4 DiMAT Materials Environmental and Cost Life Cycle Assessment ($Di^{MEC\cdot LCA}$) 180
4.4.1 General Architecture 180
4.4.2 Functional Components 181
4.4.3 Interactions 183
4.4.4 Mapping of functional components to IIRA functional domain 189
4.5 DiMAT Materials Design Framework (DI^{MDF}) 189
4.5.1 General architecture 189
4.5.2 Functional components 190
4.5.3 Interactions 192
4.5.4 Mapping of functional components to IIRA functional domain 193
4.6 DiMAT Materials Modeler (DI^{MM}) 194

4.6.1 General Architecture 194
4.6.2 Functional components 194
4.6.3 Interactions 197
4.6.4 Mapping of functional components to IIRA functional domain 198
4.7 DiMAT Materials Designer (DI^{MD}) 198
4.7.1 General architecture 198
4.7.2 Functional components 199
4.7.3 Interactions 201
4.7.4 Mapping of functional components to IIRA functional domain 203
4.8 DiMAT Materials Mechanical Properties Simulator (DI^{MMS}) 204
4.8.1 General Architecture 204
4.8.2 Functional components 204
4.8.3 Interactions 207
4.8.4 Mapping of functional components to IIRA functional domain 210
4.9 DiMAT Materials Processing Simulator (DI^{MPS}) 211
4.9.1 General Architecture 211
4.9.2 Functional components 211
4.9.3 Interactions 214
4.9.4 Mapping of functional components to IIRA functional domain 217
4.10 DiMAT Digital Twin for Process Control (DI^{DTPC}) 218
4.10.1 General Architecture 218
4.10.2 Functional components 218
4.10.3 Interactions 222
4.10.4 Mapping of functional components to IIRA functional domain 226
5 IMPLEMENTATION VIEWPOINT 226
5.1 Implementation Viewpoint approach for DiMAT 227
5.1.1 Implementation Viewpoint Concepts 227
5.1.2 Implementation Viewpoint Methodology 227

5.2 Dimat cloud materials database (Di^{CMDB}) 228
5.2.1 Internal Architecture of Toolkits 228
5.2.2 Required components 229
5.2.3 Technologies employed 231
5.2.4 Implementation maps 231
5.3 Dimat knowledge acquisition framework (Di^{KAF}) 233
5.3.1 Internal Architecture of Toolkits 233
5.3.2 Required components 233
5.3.3 Technologies employed 236
5.3.4 Implementation maps 237
5.4 Dimat materials environmental and cost life assessment (DI$^{MEC-LCA}$) 237
5.4.1 Internal Architecture of Toolkits 237
5.4.2 Required components 238
5.4.3 Technologies employed 242
5.4.4 Implementation maps 243
5.5 Dimat Materials Design Framework (Di^{MDF}) 243
5.5.1 Internal Architecture of Toolkits 243
5.5.2 Required components 244
5.5.3 Technologies employed 246
5.5.4 Implementation maps 247
5.6 Dimat Materials Modelling and Behaviour Prediction (Di^{MM}) 248
5.6.1 Internal Architecture of Toolkits 248
5.6.2 Required components 249
5.6.3 Technologies employed 252
5.6.4 Implementation maps 252
5.7 Dimat Materials Design, Characterization and Synthesis (Di^{MD}) 253
5.7.1 Internal Architecture of Toolkits 253
5.7.2 Required components 254

5.7.3 Technologies employed 256
5.7.4 Implementation maps 257
5.8 Dimat materials mechanical properties simulator (Di^{MMS}) 257
5.8.1 Internal Architecture of Toolkits 257
5.8.2 Required components 258
5.8.3 Technologies employed 260
5.8.4 Implementation maps 261
5.9 Dimat Materials Processing Simulation (Di^{MPS}) 262
5.9.1 Internal Architecture of Toolkits 262
5.9.2 Required components 263
5.9.3 Technologies employed 264
5.9.4 Implementation maps 265
5.10 Dimat Digital Twin for Process Control (Di^{DTPC}) 265
5.10.1 Internal Architecture of Toolkits 265
5.10.2 Required components 267
5.10.3 Technologies employed 272
5.10.4 Implementation maps 273
6 CONCLUSIONS 274
7 REFERENCES 275
APPENDIX A 276
APPENDIX B 278

LIST OF FIGURES

FIGURE 1: IIRA ARCHITECTURE FRAMEWORK 23
FIGURE 2: DIMAT BUSINESS VIEWPOINT 27
FIGURE 3. DIMAT SUITES AND TOOLKITS 30
FIGURE 4: USAGE VIEWPOINT LAYER 52
FIGURE 5: KEY CONCEPTS ON THE USAGE VIEWPOINT 53
FIGURE 6: EXAMPLE USE CASE DIAGRAM 54
FIGURE 7: EXAMPLE ACTIVITY DIAGRAM 56
FIGURE 8: USE CASE DIAGRAM FROM DI^{CMDB} 58
FIGURE 9: WORKFLOW "UPLOAD A NEW DATASET" OF DI^{CMDB} 60
FIGURE 10: WORKFLOW "FIND A SPECIFIC PROCESS" OF DI^{CMDB} 63
FIGURE 11: WORKFLOW "QUERYING DATA VIA THE PYTHON SDK" OF DI^{CMDB} 66
FIGURE 12: USE CASE DIAGRAM OF DI^{KAF} 68
FIGURE 13: WORKFLOW "GET VISUALIZATION OF DATA" OF DI^{KAF} 70
FIGURE 14: WORKFLOW "ADD NEW DATA SOURCE" OF DI^{KAF} 75
FIGURE 15: WORKFLOW "MODIFY KNOWLEDGE GRAPH" OF DI^{KAF} 79
FIGURE 16: WORKFLOW "CREATE ACCOUNT" OF DI^{KAF} 82
FIGURE 17. USE CASE DIAGRAM OF $DI^{MEC-LCA}$ 85
FIGURE 18: WORKFLOW "STORE USER INPUT" OF $DI^{MEC-LCA}$ 86
FIGURE 19: MOCKUP "STORE USER INPUT" OF $DI^{MEC-LCA}$ 87
FIGURE 20: WORKFLOW "VISUALIZE LCA RESULTS" OF $DI^{MEC-LCA}$ 88
FIGURE 21: MOCKUP "VISUALIZE LCA RESULTS" OF $DI^{MEC-LCA}$ 89
FIGURE 22: WORKFLOW "VISUALIZE KPIS" OF $DI^{MEC-LCA}$ 90
FIGURE 23: MOCKUP “VISUALIZE KPIS” OF $DI^{MEC-LCA}$ 91
FIGURE 24: WORKFLOW "DOWNLOAD DATA" OF $DI^{MEC-LCA}$ 92
FIGURE 25: MOCKUP "DOWNLOAD DATA" OF $DI^{MEC-LCA}$ 93
FIGURE 26: WORKFLOW "LOG IN" OF $DI^{MEC-LCA}$ 94
FIGURE 27: MOCKUP "LOG IN" OF $DI^{MEC-LCA}$ 95

FIGURE 28: WORKFLOW "MODIFY VISUALIZATION" OF DI$^{MEC-LCA}$ 96
FIGURE 29: MOCKUP "MODIFY VISUALIZATION" OF $^{DIMEC-LCA}$ 97
FIGURE 30: USE CASE DIAGRAM OF DI^{MDF} 97
FIGURE 31: WORKFLOW "SEARCH FOR RECOMMENDATIONS" OF DI^{MDF} 99
FIGURE 32: MOCKUP "ADD NEW INFORMATION" OF DI^{MDF} 102
FIGURE 33: USE CASE DIAGRAM OF DI^{MM} 105
FIGURE 34: WORKFLOW "PREDICT MATERIAL BEHAVIOUR" OF DI^{MM} 107
FIGURE 35: WORKFLOW "SUGGEST OPTIMAL CONFIGURATIONS" OF DI^{MM} 110
FIGURE 36. WORKFLOW "ANALYSE CAUSAL INFERENCES" OF DI^{MM} 112
FIGURE 37: WORKFLOW "GENERATE REPORTS" OF DI^{MM} 115
FIGURE 38: USE CASE OF DI^{MD} 117
FIGURE 39: WORKFLOW "DEFINE NEW MATERIAL" OF DI^{MD} 118
FIGURE 40: WORKFLOW "REVIEW THE RESULTS" OF DI^{MD} 121
FIGURE 41: USE CASE OF DI^{MMS} 123
FIGURE 42: WORKFLOW "PREDICTION OF MECHANICAL PROPERTIES" OF DI^{MMS} 125
FIGURE 43: WORKFLOW "OPTIMIZATION" OF DI^{MMS} 128
FIGURE 44: USE CASE DIAGRAM DI^{MPS} 130
FIGURE 45: WORKFLOW "PREDICT MATERIAL BEHAVIOUR UNDER PROCESSING CONDITIONS" OF DI^{MPS} 132
FIGURE 46: WORKFLOW "OPTIMIZATION OF THE MATERIAL TRANSFORMATION PROCESSES AND PREDICTION CONFIGURATION" OF DI^{MPS} 135
FIGURE 47: WORKFLOW "CORRELATION ANALYSIS AND RESULTS VALIDATION WITH DATABASE INFO" OF DI^{MPS} 138
FIGURE 48: USE CASE DIAGRAM OF DI^{DTPC} 140
FIGURE 49: WORKFLOW "SEND COMMAND TO DEVICE" OF DI^{DTPC} 142
FIGURE 50: WORKFLOW "VIEW DIGITAL TWIN STATUS" OF DI^{DTPC} 145
FIGURE 51: WORKFLOW “DEVICE REGISTRATION” OF DI^{DTPC} 148
FIGURE 52: WORKFLOW "MONITOR DTPC OPERATION" OF DI^{DTPC} 151
FIGURE 53: WORKFLOW "ANALYSE TIME-SERIES DATA" OF DI^{DTPC} 156
FIGURE 54: EXAMPLE DIAGRAM 160

FIGURE 55: EXAMPLE OF SEQUENCE DIAGRAM 162
FIGURE 56: FUNCTIONAL DOMAINS 162
FIGURE 57: CLOUD MATERIALS DATABASE DIAGRAM 164
FIGURE 58: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT " CREATE A KLTEM" .. 168
FIGURE 59: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "UPLOAD DATASETS" 169
FIGURE 60: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "VISUALIZE KLTEMS" 170
FIGURE 61: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "DOWNLOAD DATA". 170
FIGURE 62: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "MODIFY METADATA" 171
FIGURE 63: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "SEARCH DATASPACE" 172
FIGURE 64: SEQUENCE DIAGRAM OF THE DI^{CMDB} TOOLKIT COMPONENT "CHECK USER ACCESS" 173
FIGURE 65: KNOWLEDGE ACQUISITION FRAMEWORK DIAGRAM 174
FIGURE 66: SEQUENCE DIAGRAM OF THE DI^{KAF} TOOLKIT COMPONENT "GET VISUALIZATION OF DATA" 177
FIGURE 67: SEQUENCE DIAGRAM OF THE DI^{KAF} TOOLKIT COMPONENT "ADD NEW DATA SOURCE" 178
FIGURE 68: SEQUENCE DIAGRAM OF THE DI^{KAF} TOOLKIT COMPONENT "MODIFY KNOWLEDGE GRAPH" 178
FIGURE 69: SEQUENCE DIAGRAM OF THE DI^{KAF} TOOLKIT COMPONENT "CREATE ACCOUNT".... 179
FIGURE 70: MATERIALS ENVIRONMENTAL AND COST LIFE CYCLE ASSESSMENT DIAGRAM 180
FIGURE 71: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "STORE USER INFORMATION" 184
FIGURE 72: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "STORE USER INFORMATION 2" 185
FIGURE 73: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "VISUALIZE LCA RESULTS" 186
FIGURE 74: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "VISUALIZE KPIS"... 186
FIGURE 75: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "DOWNLOAD DATA" 187
FIGURE 76: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "LOG IN" 188

FIGURE 77: SEQUENCE DIAGRAM OF THE DI$^{MEC-LCA}$ TOOLKIT COMPONENT "MODIFY VISUALIZATION" 188
FIGURE 78: MATERIALS DESIGN FRAMEWORK DIAGRAM 189
FIGURE 79: SEQUENCE DIAGRAM OF THE DI^{MDF} TOOLKIT COMPONENT "SEARCH CONNECTION" 192
FIGURE 80: SEQUENCE DIAGRAM OF THE DI^{MDF} TOOLKIT COMPONENT "ADD CONNECTION" . 193
FIGURE 81: MATERIALS MODELER DIAGRAM 194
FIGURE 82: MATERIALS DESIGNER DIAGRAM 198
FIGURE 83: SEQUENCE DIAGRAM OF THE DI^{MD} TOOLKIT COMPONENT "LOG IN" 202
FIGURE 84: SEQUENCE DIAGRAM OF THE DI^{MD} TOOLKIT COMPONENT "DEFINE NEW MATERIAL" 203
FIGURE 85: MATERIALS MECHANICAL PROPERTIES SIMULATOR DIAGRAM 204
FIGURE 86: SEQUENCE DIAGRAM OF THE DI^{MMS} TOOLKIT COMPONENT "LOG IN" 208
FIGURE 87: SEQUENCE DIAGRAM OF THE DI^{MMS} TOOLKIT COMPONENT "PREDICT MECHANICAL PROPERTIES" 209
FIGURE 88: SEQUENCE DIAGRAM OF THE DI^{MMS} TOOLKIT COMPONENT "OPTIMIZATION" 210
FIGURE 89: MATERIALS PROCESSING SIMULATOR DIAGRAM 211
FIGURE 90: SEQUENCE DIAGRAM OF THE DI^{MPS} TOOLKIT COMPONENT "LOG IN" 215
FIGURE 91: SEQUENCE DIAGRAM OF THE DI^{MPS} TOOLKIT COMPONENT "PREDICT MANUFACTURING PROCESS PARAMETERS" 216
FIGURE 92: SEQUENCE DIAGRAM OF THE DI^{MPS} TOOLKIT COMPONENT "OPTIMIZATION" 216
FIGURE 93: SEQUENCE DIAGRAM OF THE DI^{MPS} TOOLKIT COMPONENT "STORE USER INPUT AND OUTPUT" 217
FIGURE 94: DIGITAL TWIN FOR PROCESS CONTROL DIAGRAM 218
FIGURE 95: SEQUENCE DIAGRAM OF THE DI^{DTPC} TOOLKIT COMPONENT "SEND COMMAND TO DEVICE" 223
FIGURE 96: SEQUENCE DIAGRAM OF THE DI^{DTPC} TOOLKIT COMPONENT "VIEW DIGITAL TWIN STATUS" 223
FIGURE 97: SEQUENCE DIAGRAM OF THE DI^{DTPC} TOOLKIT COMPONENT "DEVICE REGISTRATION" 224
FIGURE 98: SEQUENCE DIAGRAM OF THE DI^{DTPC} TOOLKIT COMPONENT "MONITOR DTPC OPERATION" 225

FIGURE 99: SEQUENCE DIAGRAM OF THE DI^{DTPC} TOOLKIT COMPONENT "TIME SERIES DATA ANALYSIS" 226
FIGURE 100: CLOUD MATERIALS DATABASE COMPONENT MAP 229
FIGURE 101: DATA UPLOAD IMPLEMENTATION MAP OF DI^{CMDB} 232
FIGURE 102: DATA QUERY VIA SDK IMPLEMENTATION MAP OF DI^{CMDB} 232
FIGURE 103: KNOWLEDGE ACQUISITION FRAMEWORK COMPONENT MAP 233
FIGURE 104: IMPLEMENTATION MAP OF DI^{KAF} 237
FIGURE 105: TECHNOLOGIES OF $DI^{MEC\text{-}LCA}$ 238
FIGURE 106: IMPLEMENTATION MAP OF $DI^{MEC\text{-}LCA}$ 243
FIGURE 107: MATERIALS DESING FRAMEWORK COMPONENT MAP 244
FIGURE 108: IMPLEMENTATION MAP OF DI^{MDB} 247
FIGURE 109: MATERIALS MODELLING AND BEHAVIOUR PREDICTION COMPONENT MAP 248
FIGURE 110: DATA FLOWS OF DI^{MM} 250
FIGURE 111: CLOUD-COMPUTING PLATFORMS OF DI^{MM} 252
FIGURE 112: IMPLEMENTATION MAP OF DI^{MM} 253
FIGURE 113: MATERIALS DESIGN, CHARAZTERIZATION AND SYNTHESIS OF DI^{MD} 254
FIGURE 114: IMPLEMENTATION MAP OF DI^{MD} 257
FIGURE 115: MATERIALS MECHANICAL PROPERTIES SIMULATOR COMPONENT MAP 258
FIGURE 116: IMPLEMENTATION MAP OF DI^{MMS} 262
FIGURE 117: INTERNAL ARCHITECTURE OF DI^{MPS} 262
FIGURE 118: IMPLEMENTATION MAP DI^{MPS} 265
FIGURE 119: DIGITAL TWIN FOR PROCESS CONTROL COMPONENT MAP 266
FIGURE 120: IMPLEMENTATION MAP OF DI^{DTPC} 273

LIST OF TABLES

TABLE 1. SURVEY STRUCTURE 29
TABLE 2. STAKEHOLDERS' INTEREST ON TOOLKITS 31
TABLE 3. BUSINESS DECISION MAKERS' RESPONSES TO TOOLKIT DI^{CMDB} 33
TABLE 4. BUSINESS DECISION MAKERS' TO TOOLKIT DI^{KAF} 34
TABLE 5. BUSINESS DECISION MAKERS' RESPONSES TO TOOLKIT $DI^{MEC\text{-}LCA}$ 35
TABLE 6. BUSINESS DECISION MAKERS' RESPONSES TO TOOL DI^{MDF} 36
TABLE 7. BUSINESS DECISION MAKERS' RESPONSES TO TOOL DI^{MM} 37
TABLE 8. BUSINESS DECISION MAKERS' RESPONSES TO TOOL DI^{MD} 37
TABLE 9. BUSINESS DECISION MAKERS' RESPONSES TO TOOL DI^{MMS} 38
TABLE 10. BUSINESS DECISION MAKERS' RESPONSES TO TOOLKIT DI^{MPS} 39
TABLE 11. BUSINESS DECISION MAKERS' RESPONSES TO TOOLKIT DI^{DTPC} 41
TABLE 12. TECHNICAL LEADERS' RESPONSES TO TOOLKIT DI^{CMDB} 42
TABLE 13. TECHNICAL LEADERS' RESPONSES TO TOOLKIT DI^{KAF} 43
TABLE 14. TECHNICAL LEADERS' RESPONSES TO TOOLKIT $DI^{MEC\text{-}LCA}$ 44
TABLE 15. TECHNICAL LEADERS' RESPONSES TO TOOL DI^{CMDB} 45
TABLE 16. TECHNICAL LEADERS' RESPONSES TO TOOL DI^{MM} 46
TABLE 17. TECHNICAL LEADERS' RESPONSES TO TOOL DI^{MD} 46
TABLE 18. TECHNICAL LEADERS' RESPONSES TO TOOLKIT DI^{MMS} 47
TABLE 19. TECHNICAL LEADERS' RESPONSES TO TOOLKIT DI^{MPS} 48
TABLE 20. TECHNICAL LEADERS' RESPONSES TO TOOLKIT DI^{DTPC} 49
TABLE 21. TOOLKITS IN ORDER FROM HIGHEST TO LOWEST AVERAGE SCORE 50
TABLE 22. SCORE OF EACH SUITE AS THE SUM OF THE AVERAGE SCORES OF THE TOOLKITS 51
TABLE 23. DI^{CMDB} MOCKUPS ACTIVITY "UPLOAD A NEW DATASET" 62
TABLE 24. DI^{CMDB} MOCKUP ACTIVITY "FIND A SPECIFIC PROCESS" 64
TABLE 25. DI^{KAF} MOCKUP ACTIVITY "GET VISUALIZATION OF DATA" 74
TABLE 26. DI^{KAF} MOCKUP ACTIVITY "ADD NEW DATA SOURCE" 78
TABLE 27: DI^{KAF} MOCKUP ACTIVITY " MODIFY KNOWLEDGE GRAPH" 80

TABLE 28. DI^{KAF} MOCKUP ACTIVITY "CREATE ACCOUNT" 84
TABLE 29. DI^{MDF} MOCKUPS ACTIVITY "SEARCH FOR RECOMMENDATIONS" 101
TABLE 30. DI^{MDF} MOCKUP ACTIVITY "ADD NEW INFORMATION" 103
TABLE 31. DI^{MM} MOCKUP ACTIVITY "PREDICT MATERIAL BEHAVIOUR" 109
TABLE 32. DI^{MM} MOCKUP ACTIVITY "SUGGEST OPTIMAL CONFIGURATIONS" 111
TABLE 33. DI^{MM} MOCKUP ACTIVITY "ANALYSE CAUSAL INFERENCES" 113
TABLE 34. DI^{MM} MOCKUP ACTIVITY "GENERATE REPORTS" 116
TABLE 35. DI^{MD} MOCKUP ACTIVITY "DEFINE NEW MATERIAL" 120
TABLE 36. DI^{MD} MOCKUP ACTIVITY "REVIEW THE RESULTS" 122
TABLE 37. DI^{MMS} MOCKUP ACTIVITY "PREDICTION OF MECHANICAL PROPERTIES" 127
TABLE 38. DI^{MMS} MOCKUP ACTIVITY "OPTIMIZATION" 129
TABLE 39. DI^{MPS} MOCKUP ACTIVITY "PREDICT MATERIAL BEHAVIOR UNDER PROCESSING CONDITIONS" 134
TABLE 40. DI^{MPS} MOCKUP ACTIVITY "OPTIMIZATION OF THE MATERIAL TRANSFORMATION PROCESSES AND PREDICTION CONFIGURATION" 137
TABLE 41. DI^{MPS} MOCKUP ACTIVITY "CORRELATION ANALYSIS AND RESULTS VALIDATION WITH DATABASE INFO" 139
TABLE 42. DI^{DTPC} MOCKUP ACTIVITY "SEND COMMAND TO DEVICE" 144
TABLE 43: DI^{DTPC} MOCKUP ACTIVITY "VIEW DIGITAL TWIN STATUS" 147
TABLE 44. DI^{DTPC} MOCKUP ACTIVITY "DEVICE REGISTRATION" 150
TABLE 45. DI^{DTPC} MOCKUP ACTIVITY "MONITOR DTPC OPERATION" 155
TABLE 46. DI^{DTPC} MOCKUP ACTIVITY "ANALYSE TIME-SERIES DATA" 158
TABLE 47. EXAMPLE TABLE FUNCTIONAL COMPONENT 161
TABLE 48. EXAMPLE TABLE FUNCTIONAL DOMAIN 163
TABLE 49. GUI OF THE FUNCTIONAL COMPONENT OF THE DICMDB TOOLKIT 165
TABLE 50. KLTEM SERVICE OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 166
TABLE 51. USER VALIDATION OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 166
TABLE 52. DATA UPLOAD OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 166
TABLE 53. DATA DOWNLOAD OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 167
TABLE 54. DATA SEARCH OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 167

TABLE 55. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI^{CMDB} TOOLKIT 167
TABLE 56. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{CMDB} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 173
TABLE 57. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{KAF} TOOLKIT 175
TABLE 58. REGISTRATION AND LOG IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DI^{KAF} TOOLKIT 175
TABLE 59. DATA HANDLING OF THE FUNCTIONAL COMPONENT OF THE DI^{KAF} TOOLKIT 176
TABLE 60. GRAPH DATABASE OF THE FUNCTIONAL COMPONENT OF THE DI^{KAF} TOOLKIT 176
TABLE 61. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{KAF} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 179
TABLE 62. GUI OF THE FUNCTIONAL COMPONENT OF THE DI$^{MEC\text{-}LCA}$ 181
TABLE 63. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DI$^{MEC\text{-}LCA}$ 182
TABLE 64. LOGIN IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DI$^{MEC\text{-}LCA}$ 182
TABLE 65. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI$^{MEC\text{-}LCA}$ 183
TABLE 66. DATA INSERTION API OF THE FUNCTIONAL COMPONENT OF THE DI$^{MEC\text{-}LCA}$ 183
TABLE 67. MAPPING OF FUNCTIONAL COMPONENTS OF DI$^{MEC\text{-}LCA}$ TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 189
TABLE 68. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{MDF} TOOLKIT 190
TABLE 69. MDF SERVICE OF THE FUNCTIONAL COMPONENT OF THE DI^{MDF} TOOLKIT 190
TABLE 70. USER VALIDATION OF THE FUNCTIONAL COMPONENT OF THE DI^{MDF} TOOLKIT 191
TABLE 71. TOOLKIT QUERY OF THE FUNCTIONAL COMPONENT OF THE DI^{MDF} TOOLKIT 191
TABLE 72. RELATIONS MANAGER OF THE FUNCTIONAL COMPONENT OF THE DI^{MDF} TOOLKIT . 192
TABLE 73. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{MDF} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 193
TABLE 74. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{MM} TOOLKIT 195
TABLE 75. AI ALGORITHMS OF THE FUNCTIONAL COMPONENT OF THE DI^{MM} TOOLKIT..... 195
TABLE 76. DATA ANALYZER OF THE FUNCTIONAL COMPONENT OF THE DI^{MM} TOOLKIT..... 196
TABLE 77. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI^{MM} TOOLKIT 196
TABLE 78. DATA INSERTION API OF THE FUNCTIONAL COMPONENT OF THE DI^{MM} TOOLKIT..... 197

TABLE 79. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{MM} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 198

TABLE 80. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{MD} TOOLKIT 199

TABLE 81. LOG IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DI^{MD} TOOLKIT 199

TABLE 82. SIMULATION DATABASE OF THE FUNCTIONAL COMPONENT OF THE DI^{MD} TOOLKIT 200

TABLE 83. SIMULATION SCHEDULER OF THE FUNCTIONAL COMPONENT OF THE DI^{MD} TOOLKIT 200

TABLE 84. CALCULATION CODE OF THE FUNCTIONAL COMPONENT OF THE DI^{MD} TOOLKIT 201

TABLE 85. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{MD} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 203

TABLE 86. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT 205

TABLE 87. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT 206

TABLE 88. LOG IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT..... 206

TABLE 89. PREDICT MECHANICAL PROPERTIES API OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT 206

TABLE 90. OPTIMIZATION API OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT..... 207

TABLE 91. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI^{MMS} TOOLKIT 207

TABLE 92. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{MMS} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 210

TABLE 93. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT 212

TABLE 94. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT..... 213

TABLE 95. LOG IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT..... 213

TABLE 96. PREDICT MANUFACTURING PROCESS PARAMETERS OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT..... 213

TABLE 97. OPTIMIZATION RESULTS OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT 214

TABLE 98. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI^{MPS} TOOLKIT..... 214

TABLE 99. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{MPS} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 217

TABLE 100. GUI OF THE FUNCTIONAL COMPONENT OF THE DI^{DTPC} TOOLKIT 219

TABLE 101. LOGIN VALIDATION OF THE FUNCTIONAL COMPONENT OF THE DIDTPC TOOLKIT 219
TABLE 102. IOT DEVICE GATEWAY OF THE FUNCTIONAL COMPONENT OF THE DI^{DTPC} TOOLKIT 220
TABLE 103. SHORT-TERM DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DI^{DTPC} TOOLKIT 221
TABLE 104. CLOUD SERVICE OF THE FUNCTIONAL COMPONENT OF THE DI^{DTPC} TOOLKIT 221
TABLE 105. VIRTUAL FUNCTIONS OF THE FUNCTIONAL COMPONENT OF THE DI^{DTPC} TOOLKIT 221
TABLE 106. MAPPING OF FUNCTIONAL COMPONENTS OF DI^{DTPC} TOOLKIT WITH IIRA FUNCTIONAL DOMAIN 226
TABLE 107. DATA FLOW OF DI^{CMDB} 230
TABLE 108. DATA FLOW OF DI^{KAF} 234
TABLE 109. DATA ANALYTICS OF DI^{KAF} 235
TABLE 110. DATA FLOW OF $DI^{MEC-LCA}$ 240
TABLE 111. DATA FLOW OF DI^{MDB} 246
TABLE 112. DATA ANALYTICS OF DI^{MDB} 246
TABLE 113. DATA FLOW OF DI^{MM} 251
TABLE 114. DATA FLOW OF DI^{MD} 255
TABLE 115. DATA FLOW OF DI^{MMS} 259
TABLE 116. DATA FLOW OF DI^{DTPC} 271
TABLE 117. DATA ANALYST OF DI^{DTPC} 271
TABLE 118. CHANGES IN THE DELIVERABLE BASED ON THE REVIEWERS' COMMENTS 276
TABLE 119. DEGREE OF INTEREST IN THE TOOLKITS 279

ABBREVIATIONS

API	Application Programming Interface
CMDB	Cloud Materials Database
DTPC	Digital Twin for Process Control
FE	Finite Elements
IIRA	Industrial Internet Reference Architecture
IP	Internet Protocol
KAF	Knowledge Acquisition Framework
KG	Knowledge Graph
KPI	Key Performance Indicator
KPIs	Key Performance Indicators
LCA	Life Cycle Assessment
MD	Materials Designer
MDF	Materials Design Framework
MEC-LCA	Materials Environmental and Cost Life Cycle Assessment
ML	Machine Learning
MM	Materials Modeler
MMS	Materials Mechanical Properties Simulator
MPS	Materials Processing Simulator
ROI	Return on Investment
SME	Smart Manufacturing Environments
TCP	Transmission Control Protocol

1 METHODOLOGICAL APPROACH

Throughout this section, we are going to see the framework design approach that will be used in the DiMAT project. To do this, we will analyze the views, where they come from and why they are interesting. Then, we will develop a plan of interaction between them to offer validity at all levels. Finally, we will develop the methodology to be followed in this deliverable.

1.1 INTRODUCTION

The DiMAT project aims to create a digital platform providing open modelling, simulation, and optimization tools with a special focus on smart manufacturing environments (SME) to improve the effectiveness of material design and ensure high quality, sustainability, and competitiveness of manufacturing processes. To achieve this, it is necessary to define an appropriate architecture and to consider certain characteristics. The different components will be delivered as Software-as-a-Service, based on a modular design, and following best practices to ensure maintainability and scalability. Security will be considered to ensure data confidentiality, integrity, and availability. The architectural work presented here will guide the development of all DiMAT suites to ensure a unified approach where the different perspectives and requirements of the stakeholders involved can be met.

1.2 VIEWPOINTS

The core principle of the IIRA methodology revolves around viewpoints, enabling different architects, developers, and engineers to address different design challenges. To achieve this, the design process is divided into four phases that help to define stakeholder perspectives. This division establishes a systematic order that reflects a pattern of interactions between viewpoints, as decisions made at higher levels impose requirements on those at lower levels [1].

The DiMAT approach reflects this methodology by basing the definition of solutions on these views in an iterative approach, as detailed below. The DiMAT approach incorporates the IIRA methodology by using an iterative process based on the different viewpoints to define solutions.

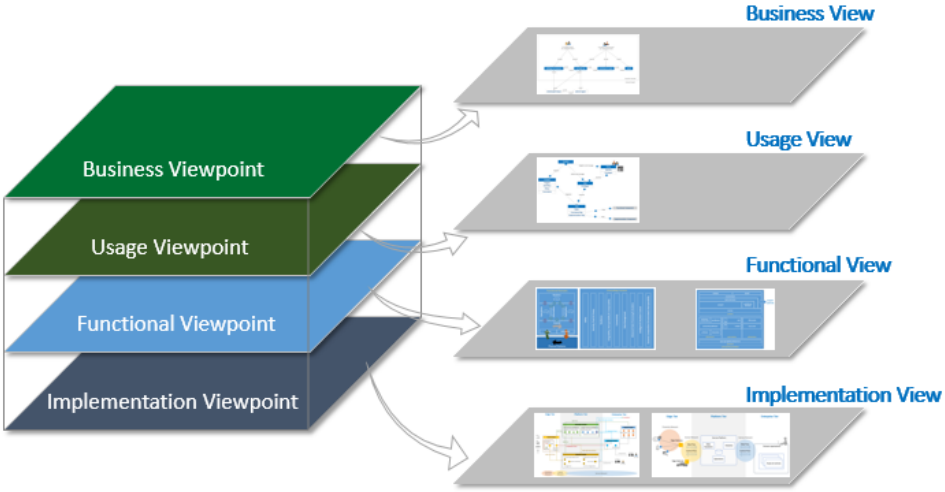


Figure 1: IIRA Architecture Framework

These viewpoints, namely the Business Viewpoint, the Usage Viewpoint, the Functional Viewpoint, and the Implementation viewpoint, serve as the basis for structuring the DiMAT-RA framework, tailored explicitly for smart manufacturing processes (Figure 1). The following summary outlines the key characteristics of these viewpoints as identified by the IIRA:

- **The Business Viewpoint:** The main objective of this task is to provide a business point of view within the DiMAT design activities to avoid the risk of 'technology-centric' development. This will allow us to include in the (early) design the requirements and needs closer to the real operational needs. To this end, the task will focus on framing the vision, values, and key objectives of the DiMAT architecture from a business, regulatory and stakeholder perspective. In parallel, it will also identify the capabilities of the consortium's technical team and perform a mapping exercise to the business requirements. To increase the effectiveness of the task, targeted input from experienced business partners close to the industry will be gathered, for example, through a dedicated questionnaire or interview, in a similar way to product development.

- **The Usage Viewpoint:** The development of this task focuses on the structuring of the DiMAT framework. To do this, we will identify several key points that the IIRA reference architecture presents to us. These are the tasks, roles, activities, and parts of DiMAT that focus on software development and human intervention. Once the previous elements have been identified, we will go one step further to define both the functional map, the specifications of the roles responsible for the tasks, and the implementation maps. For each of the tasks, the triggers for each activity are defined, considering the workflows, effects, and constraints.

- **The Functional Viewpoint:** This task will decompose DiMAT Framework into its Control, Operations, Information, Application, and Business domains and identify the

data, decision and command/request flows that circulate between them. Once this decomposition has been performed, it will elaborate on the control, coordination and orchestration exercised from each of these domains and the different typical operations from these domains.

- **The Implementation Viewpoint:** The main objective of this task is to define the internal structure of the DiMAT solutions. To describe the interconnection between the different components of the solutions and to define the technologies to be used for their development. To define the interconnection and communication of the solutions, the Application Programming Interfaces (API) will be specified. On the other hand, it is important to stress that this viewpoint will consider all the data obtained from the different viewpoints. From the business layer to the activities identified in use, including the functional information of these solutions. In this sense, the Edge Computing, Microservice Applications and Function as a Service (FaaS) platforms are fundamental for the implementation of the DiMAT toolsets described in WP4, WP5 and WP6.

1.3 DIMAT RA: VIEWPOINTS

Following the IIRA methodology, the final iteration of the DiMAT RA integrates essential outlooks (referred to as "viewpoints" in DiMAT) for intelligent manufacturing processes. The DiMAT Reference Architecture adopts a four-viewpoint structure to facilitate an iterative approach to addressing architectural challenges during its inception. This section outlines the methodology for establishing the four viewpoints (Business, Usability, Functional and Implementation) and provides a concise overview of their primary objectives and deliverables.

- The **Business Viewpoint** describes the stakeholders involved in the system's development, deployment, and operation, along with their business vision and objectives. Contextual factors, including general business and regulatory issues, are considered within this viewpoint.
- The **Usage Viewpoint** details the practical use of the system. This involves outlining sequences of activities performed by human actors and/or logical components (e.g., system or system components).
- The **Functional Viewpoint** specifies the system's functionalities by describing the functional components that make up the system, their interfaces, and interactions. It also includes interactions with external logical modules (e.g., external subsystems).

- The **Implementation Viewpoint** includes the implementation technologies used for the functional components, their lifecycle information and how communication between them is realized.

The interaction pattern between viewpoints establishes a hierarchical relationship: decisions made in higher-level viewpoints guide and impose requirements on the viewpoints below them. At the same time, considerations within lower-level viewpoints validate and, in some cases, lead to revisions in the analysis and potentially influence decisions in the viewpoints above.

The initial version of the DiMAT Reference Architecture served as the basis for defining the fundamental viewpoints while contributing to their review and subsequent release of the final version.

The development of each viewpoint followed an incremental and iterative approach, with several iterations anticipated. Each sprint involved successive refinements and increments. At the end of each sprint, a validation session was held to release the viewpoints and gather feedback. In addition, a retrospective session was held to identify lessons learned and improvements for the next iteration of each viewpoint.

2 BUSINESS VIEWPOINT

The main objective of this task is to provide, within the DiMAT design activities, a business point of view to avoid the risk of a 'technology-centric' development. This will allow to incorporate, already in the (early) design, the requirements and needs which are closer to real-world, operational needs. To this end, the task will focus on framing the vision, values, and key objectives of the DIMAT Architecture from the business, regulatory and stakeholders' point of view. In parallel, it will also identify the capabilities of the consortium technical team and carry out a mapping exercise to the business requirements. To enhance the effectiveness of the task, focused input from experienced business partners closer to the industry will be captured, for instance, through the submission of a dedicated questionnaire or interview, in a similar way as product development is carried out.

2.1 BUSINESS VIEWPOINT APPROACH FOR DIMAT

2.1.1 Business Viewpoint Concepts

As stated above, the DiMAT project aims to create a digital platform that offers open modelling, simulation, and optimization tools with a special focus on SMEs. Nevertheless, this development must be aligned with the business vision and objectives. Thus, the main objective of the business viewpoint is to avoid the risk of a 'technology-centric' development and to incorporate, already in the (early) design, the requirements and needs closer to real-world operational needs.

The DiMAT proposal includes three suits and three toolkits in each suit. Therefore, it will be necessary to know the business perspective for each of these toolkits. The task will focus on framing (for each toolkit) the **vision**, **values**, and **key objectives** of the DIMAT Architecture from the **business, regulatory and stakeholders' point of view**. In parallel, it will also **identify the capabilities** of the consortium technical team and carry out a **mapping exercise to the business requirements.** For this purpose**, input from experienced business partners** closer to the industry **will be captured** through the submission of a dedicated questionnaire or interview, in a similar way as product development is carried out. Two different groups of decision-makers in each partner will be asked (Figure 2):

1. Business decision-makers
2. Technical leaders

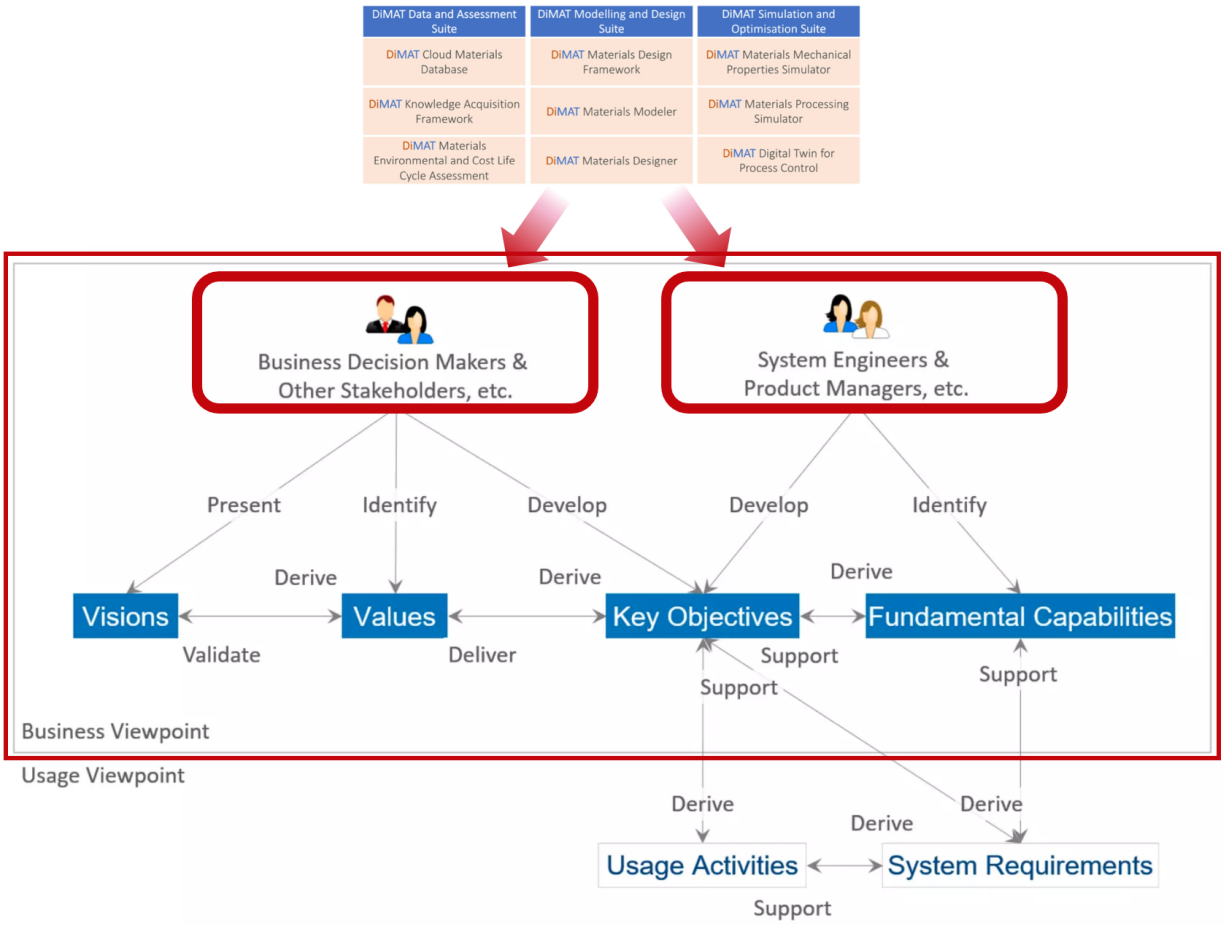


Figure 2: DiMAT Business Viewpoint

2.1.2 Business Viewpoint Methodology

The methodology for completing the task has been defined with the following steps:

- Firstly, the identification of stakeholders. The companies interested in the project must provide the business perspective, and the first step is to identify them.
- Secondly, stakeholders are asked for their business perspective on each toolkit from the business decision-makers and technical leaders' domains. Thus, business decision-makers are asked to identify the vision, values, and key objectives, and technical leaders are asked to identify the new capabilities for the organization.

It is important to note that stakeholders may be interested in some toolkits rather than in others. In this sense, we only seek to obtain information on the toolkits in which the stakeholder is interested. Schematically, the steps are as follows:

- Stakeholders' identification
- For each stakeholder and each toolkit (they are interested in):
  - Extract information from the Business Decision Makers Domain
  - Extract information from the Technical Leaders' Domain

2.1.3 Questionnaire (Form) Design

A questionnaire has been designed to collect information regarding the requirements closer to the real business needs from the perspectives of business decision-makers and technical leaders. The questionnaire comprises four sections in addition to the survey instructions. These sections are as follows:

- Identification of Respondents.
- Toolkit Selection.
- Business Perspective.
- Technical Perspective.

Table 1 displays these sections and the information required in each of them. Note that two members of each stakeholder should complete the survey. Both should jointly fill out the "Toolkit Selection" section. Subsequently, the business decision-maker should complete the "Business Perspective" section according to the information provided in the previous section. Finally, the technical leader should fill out the "Technical Perspective" section based on the input provided by their counterpart in the earlier sections.

Section	Question/Text		Answer type
Survey instructions	Welcome, survey duration, survey objective...		Text
Identification	Organization		Open answer
	Sector		Open answer
	Business contact	Name	Open answer
		Job position	Open answer
		E-mail	Open answer
	Technical contact	Name	Open answer

		Job position	Open answer
		E-mail	Open answer
Toolkit selection	Introduction to DiMAT Suites and Toolkits		Text
	Interest per toolkit		5-point Likert Scale
Business perspective	Description of information to be collected (vision, values, key objectives, and process to focus on)		Text
	Toolkit table (one per interested toolkit)	Toolkit name	Open answer
		Vision	Open answer
		Values	Open answer
		Key objectives	Open answer
		Processes to focus on	Open answer
Technical perspective	Description of information to be collected (key objectives, fundamental capabilities, and alignment)		Text
	Toolkit table (one per interested toolkit	Toolkit name	Open answer
		Technical key objectives	Open answer
		Fundamental capabilities	Open answer
		Alignment	Open answer

Table 1. Survey structure

The full Survey model sent to the stakeholders can be referred to in the annex to this document.

2.2 DIMAT BUSINESS VIEWPOINT

In this section, the stakeholders identified for their participation in developing the business viewpoint are presented in subsection 2.1.1. Additionally, the information collected through the surveys filled out by the stakeholders for the toolkits they are interested in is presented. The nine toolkits defined by the DiMAT project are grouped into three suites (Figure 3). The information collected in the surveys is organized according to the different suites in subsections 2.2.3 (business perspective) and 2.2.4 (technical perspective).

DiMAT Data and Assessment Suite	DiMAT Modelling and Design Suite	DiMAT Simulation and Optimisation Suite
DiMAT Cloud Materials Database	DiMAT Materials Design Framework	DiMAT Materials Mechanical Properties Simulator
DiMAT Knowledge Acquisition Framework	DiMAT Materials Modeler	DiMAT Materials Processing Simulator
DiMAT Materials Environmental and Cost Life Cycle Assessment	DiMAT Materials Designer	DiMAT Digital Twin for Process Control

Figure 3. DiMAT suites and toolkits

2.2.1 Stakeholder identification

The stakeholders (organizations) involved are collected and classified here. Within the first iteration of the Business viewpoint survey, the following organizations have been approached and have provided their responses:

- ACCELIGENCE LTD
- Cetma Composites Srl
- Hegla-Hanic
- Imerys Graphite&Carbon Switzerland Ltd
- NaturePlast

All these organizations are interested in the project and have been asked for information from the business perspective. In this sense, the two groups previously identified have been asked to participate in the survey: Business Decision Makers (C-Level or upper) and Technical Leaders (System designers, Production Engineers, Product Managers, etc.).

2.2.2 Stakeholders' interest in toolkits

Table 2 displays the interest shown by the stakeholders in each of the toolkits through the survey. Based on these results, it is evident that the stakeholders have a higher interest in the "Data and Assessment" and "Simulation and Optimization" suites, with the toolkit "Materials Environmental and Cost Life Cycle Assessment" generating the most interest among them.

		Natureplast					IGC					Hegla					CETMA					ACCELI					TECNORED					
Suite/Toolkit		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	Score
Data and assessment Suite	Cloud Material Database				x			x						x			x							x						x		2.8
	Knowledge Acquisition Framework				x			x							x		x								x						x	3.3
	Materials Environmental and Cost Life Cycle Assessment					x					x					x					x				x					x		4.6
Modelling and Design Suite	Material Design Framework		x					x						x					x					x						x		2.8
	Materials Modeler			x				x				x							x					x					x			2.5
	Materials Designer	x									x	x							x			x							x			2.3
Simulation and Optimisation Suite	Materials Mechanical Properties Simulator					x		x							x				x						x			x				3.3
	Materials Processing Simulator					x				x						x					x		x								x	4.1
	Digital Twin for Process Control		x								x					x					x			x				x				3.6

Table 2. Stakeholders' interest on toolkits

In the following sections, each stakeholder responds to the parts of the survey corresponding to the toolkits they have rated with a score equal to or greater than 3. It is important to note that once the project is underway, more people are aware of it in the participating organizations. Also, the progress of the project means that the participants in it have greater clarity on specific aspects of the project.

Thus, while the initial vision of the project—presented in Deliverable 2.1 (DiMAT Project Vision)—was based on insights exclusively from those involved in the project at an early stage, this deliverable expands the perspective by incorporating input from a broader group

of external stakeholders, specifically Business Decision Makers and Technical Leaders. These insights do not imply any changes or adjustments to the project's defined metrics but rather reflect the expectations and perceived benefits identified by external stakeholders regarding the toolkits.

The document presents expectations gathered through a business questionnaire conducted with these external stakeholders. While some participants in the questionnaire are employed by companies involved in the project's pilots, they are not directly engaged in the project itself. As such, they were identified as external stakeholders, providing insights from a business perspective rather than technical project implementation. These stakeholders, holding roles such as Deputy General Manager, Application Director, R&D Manager, and CEO, were not necessarily aware of the predefined KPIs but provided their expectations regarding the toolkits' potential impact on their business operations.

For example, companies involved in Pilot 1 expressed interest in additional toolkits not initially highlighted in Deliverable 2.1, based on their evolving business needs and market considerations. Similarly, while the Project Vision in D2.1 anticipated a reduction in time-to-market by 45%, Business Decision Makers—based on their own business viewpoints—provided a different expectation of potential impact. These figures represent their perspectives on how the toolkits might benefit their operations rather than a revision of the project's defined KPIs or metrics.

For clarification, the values presented in the questionnaire responses reflect the expected benefits of the toolkits as perceived by the stakeholders. These responses to the questionnaires were completely open-ended, allowing participants to freely express their expectations, objectives and evaluations without any pre-defined restrictions. The results obtained should not be interpreted as actual results derived from the implementation of the project in the pilots

2.2.3 Business Decision Makers Domain

2.2.3.1 DiMAT Data and Assessment Suite

2.2.3.1.1 Toolkit 1 Cloud Material Database – Di^{CMDB}

The DiMAT Cloud Materials Database is a system for storing, sharing, and exploration of relevant material data for the material design, processing, and manufacturing processes. The system follows the FAIR-principles and important features like the traceability of data along

different kind of operations, data sovereignty, and data safety. The system is based on semantic technologies to handle the manifold of data sourced and possibilities for data exploration. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Easy access to relevant material data. This information allows companies to develop new formulations or to increase dynamism to select and offer products with the features required by the clients.
Values	Improved understanding, time saving and adaptation to market requirements.
Key Objectives (KO)	Increase productivity by 10-20%. Reduction of errors in data by 25%. Increase security of the data by 25%. Launching 1-2 products per year with different materials.

Table 3. Business decision makers' responses to toolkit Di^{CMDB}

Synthesis:

Four out of six respondents have indicated interest in Toolkit 1, marking scores of 3, two of them, and 4 out of 5 the others two. The organizations interested in the Cloud Material Database anticipate that collection and identification of relevant data will be eased through the introduction of the toolkit. Additionally, they highlight the increase of personnel productivity as a key objective and benefit resulting from the implementation of the toolkit. Two out of the four respondents express quantitative expectations for this improvement, ranging from a 10% to a 20% increase. One organization focuses more on market requirements alignment opportunities provided through the tool, as well as production processes optimization.

2.2.3.1.2 Toolkit 2 Knowledge Acquisition Framework – Di^{KAF}

The DiMAT Knowledge Acquisition Framework is a toolkit for representing and managing information related to the materials along with their characteristics and their relationships in the form of a Knowledge Graph (KG). Advanced information retrieval, analysis and visualisation techniques will be supported over the KG for revealing correlations among the available data. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Improving the efficiency to evaluate the information of the previous toolkit addressed to the same objective. Applications: Expansion to new prototypes, identification of relevant material for formulation development and other possibilities.
Values	Time saving, in-house knowledge (reduce dependences with 3rd parties), identification of correlations or creation of new options.
Key Objectives (KO)	Improvement of material data safety by 25%. Increase of personnel productivity 10-20%. Error reduction by 25%. Cost reduction by 5%. Timesaving by 5%. Delivery time by 7%. Launching 1-2 products per year with different materials.

Table 4. Business decision makers' to toolkit Di^{KAF}

Synthesis:

Four out of six respondents have identified interest in Toolkit 2 Knowledge Acquisition Framework marking three of them a score of 4 and one a score of 5 out of 5. The connection of DIKAF and DICMDB is highly visible in the responses of the organizations, where they expect that while the first tool will provide the data, the second one will provide the analytical feature serving the users to read and utilize it in the best way possible. Another similarity identified within the respondents' answers is the link to increased productivity as a key objective of two out of four respondents with a range of 10% to 20%.

2.2.3.1.3 Toolkit 3 Materials Environmental and Cost Life Cycle Assessment – $Di^{MEC-LCA}$

The DiMAT Materials Environmental and Cost Life Cycle Assessment is a tool for providing a high-level assessment on the environmental and economic impact of the pilot cases. The tool will be user friendly and will be based on key mid-point indicators and simple assessment calculations models. Finally, the tool will provide a graphical representation of the assessment results. Business decision-makers have identified the following visions, values, and key objectives:

Vision	To have an advanced tool to evaluate and optimize costs and environmental impact of own production. Impact for a more sustainable and energy-saving industry. Launch of new products aligned with company vision to reduce GHG emissions. Strengthening of service proposition. Provide sustainability data in sales proposals with real back up data.
Values	Increase the performance/cost ratio of own products. Increase of turnover and reduction of environmental impact. Inhouse knowledge (reduce dependences with 3rd parties)
Key Objectives (KO)	Reduction of environmental impact by 20-30%. Reduction of manufacturing cost by 20%. Reduction energy consumption by 20%. Reduction of material waste by 20%. Reduction of material economic impact by 20%. Reduction of scope1 and scope2 GHG emissions by 2030 by 42%. Increase in turnover of 5%. Reduction of environmental impact by 15-25%.

Table 5. Business decision makers' responses to toolkit Di$^{MEC-LCA}$

Synthesis:

All six responding organizations have identified toolkit 3 Materials Environmental and Cost Life Cycle Assessment as being of interest to their activities. Four out of these organizations gave a maximum score of 5 to the selected tool and the other two gave a score of 4. All of them point out to the reduction of environmental and economic impact among their personal key objectives to be reached through the implementation of the toolkit. Expectations vary between 20% and 42% of reduced impact depending on the identified indicator. Additionally, they perceive the utilization of the toolkit, and its sustainability impact would also contribute to an enhanced selling proposition, opportunities for the development of new sustainable models and with it an increased turnover.

2.2.3.2 DiMAT Modelling and Design Suit

2.2.3.2.1 Toolkit 4 DiMAT Materials Design Framework – Di^{MDF}

The DIMAT Materials Design Framework is an ontology-based open knowledge system to support the material design process, an App running on DIMAT Open Cloud Materials Database to provide data, information, and wisdom relevant for the material design process. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Speed-up the new material research process. Use existing knowledge.
Values	Reduce the required number of physical experiments with materials. Accurate decision-making.
Key Objectives (KO)	Drop to about 50% the research of new properties in materials. Increase in personnel productivity by 20%. Error reduction by 15%.

Table 6. Business decision makers' responses to tool Di^{MDF}

Synthesis:

Four out of six respondents have identified Toolkit 4 as relevant for their operations, providing three of them with a score of 3 and one with a score of 4 out of 5. The possibility of reducing the number of physical experiments is highlighted, which facilitates greater agility in the new material research process.

2.2.3.2.2 Toolkit 5 DiMAT Materials Modeler – Di^{MM}

The DiMAT Materials Modeler is a toolkit to assist discovering and designing competitive materials rapidly and effectively by modelling a range of phenomena present in materials data, by predicting materials properties and behaviour under different conditions, and by estimating or suggesting materials with improved properties with respect to certain requirements via robust AI. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Being able to provide improved materials to the market. Predict behavior of material through modelling. Expansion of business activities/new services

Values	Adding value to the products. Time saving. Inhouse knowledge (reduce dependences with 3rd parties)
Key Objectives (KO)	Increase productivity by 10-15%. Cost reduction by 5%. Reduction of environmental impact by 15%.

Table 7. Business decision makers' responses to tool Di^{MM}

Synthesis:

Four out of six respondents have identified interest in Toolkit 5 DiMAT Material Modeler marking a score of 3 out of 5. Respondents point out the benefits concerning the opportunity for materials' predictability, hence efficiency, reduced costs and improved products brought to market. Their expectations in terms of increased performance and productivity are by 10-15%.

2.2.3.2.3 Toolkit 6 Materials Designer – Di^{MD}

The DiMAT Materials Designer is a toolkit for designing materials in terms of mechanical properties, light fastness, thermal properties, rheology, etc. and predict their behavior during and after its processing, and for the analysis of the influence of the internal structure of the material and how its properties can be improved or modified by adding additives and nanoparticles. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Controlling the economic and environmental impact of the material in products. Speed-up the new product development process and for reduce the required number of physical samples and application tests
Values	Information control over our products. Reduce the number of experiments and time-to-market.
Key Objectives (KO)	Reduction of workload to acquire this information by 80%. Reduction in experiments by 25%. Reduction of time-to-market by 20%.

Table 8. Business decision makers' responses to tool Di^{MD}

Synthesis:

Three out of six respondents have indicated interest in this tool, marking two of them a score of 3 and one a score of 5 out of 5. Respondents emphasize the enhancement of its ability to control material impacts, optimize information management, decrease the resources needed for acquiring the respective information, and reduce time to market.

2.2.3.3 DiMAT Simulation and Optimisation Suite

2.2.3.3.1 Toolkit 7 DiMAT Materials Mechanical Properties Simulator Di^{MMS}

The DiMAT Materials Mechanical Properties Simulator is a toolkit for determining numerically mechanical properties such as stiffness, tensile strength, plasticity, viscoelastic and viscoplastic properties, damage, fracture, fatigue, etc. The macro mechanical behaviour will be linked with the microstructure and the constitutive equations of the components. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Expansion to larger prototypes for special applications. Use existing knowledge for multiple Use-Cases. Shorten formulation and compound development time.
Values	Inhouse knowledge, reduce dependences with 3rd parties. Accuracy; precision. Time saving and increase of turnover.
Key Objectives (KO)	Cost reduction by 8%. Increased performance by 5-20%. Delivery time by 7%. Improvement of material thermal properties by 10%. Error reduction by 15%. Increase in turnover by 5-10%.

Table 9. Business decision makers' responses to tool Di^{MMS}

Synthesis:

Four out of six respondents share interest in the implementation of toolkit 7 DiMAT Materials Mechanical Properties Simulator, providing one of them a score of 3, two a scores of 4, and

one a score of 5 out of 5. Each of them recognizes a different vision and value connected to the improvement of the respective company's operation after utilization of the toolkit, however, all of them share a common objective centered around improving productivity and cost-reduction.

2.2.3.3.2 Toolkit 8 Materials Processing Simulator Di^{MPS}

The DiMAT Materials Processing Simulator is a toolkit for determining manufacturing conditions and concepts while simulating their application, results, and requirements, in each one of the materials, processes, and processing conditions found at the DiMAT Open Cloud Materials Database. Business decision-makers have identified the following visions, values, and key objective:

Vision	Having an advanced tool to optimize process parameters reducing the number of experimental activities required to develop new manufacturing processes. Glass bending in one stroke/ first attempt. Speed up the new product development process and reduce the required number of physical samples and application tests. Will shorten formulation and compound development time.
Values	This toolkit will increase the competitiveness of the company, making the development and launch of new products easier. Reduce R&D costs and time-to-market. Increase Accuracy, punctuality. Reduce the number of experiments. Increase in turnover. Reduce experimental workload.
Key Objectives (KO)	R&D costs reduction by 50%. Time to market reduction by 20-30%. Reduction of prototyping procedures by 25-30%. Increase on-time completion by 20%. Reduction in experiments by 25%. Increase in productivity by 10-15%. Increase in turnover by 5-10%.

Table 10. Business decision makers' responses to toolkit Di^{MPS}

Synthesis:

Among the respondents, five out of six have recognized the relevance of toolkit 8 to their business operations, with three respondents giving it a score of 5 out of 5, and two rating it 4. Commonalities among the respondents include a shared objective to reduce costs, improve time-to-market, and enhance productivity within their respective domains through the implementation of the toolkit. They all emphasize the importance of efficiency and achieving substantial reductions or improvements in various aspects of their processes. Overall, the respondents share a collective objective to streamlining operations, reducing experimental activities, and achieving specific targets to enhance their organizational performance.

2.2.3.3.3 Toolkit 9 Digital Twin for Process Control – Di^{DTPC}

The DiMAT Digital Twin for Process Control is a digital counterpart of the IoT devices considered within DiMAT that provides a set of abstractions, virtualized functions, and APIs to support real and simulated manufacturing processes. Business decision-makers have identified the following visions, values, and key objectives:

Vision	Having an advanced tool to monitor process parameters giving information on the factors that can influence the performance of composite structures. Digital glass bending is as easy as possible. Identify key process parameters allowing a continuous improvement in terms of product quality and process efficiency. Ease and strengthening of process control.
Values	This toolkit will increase the competitiveness of the company by reducing the manufacturing costs. Increase the performance and reduce the scraps. On point, accurate, 3D, graphically. Higher product quality and higher efficiency. Increase in process control.
Key Objectives (KO)	Reduction of production scraps by 20%. Reduce energy consumption by 15%. Reduce material waste by 20%. Reduction of prototyping procedures by 30%.

	Increase of on-time completion by 20%. Increase in performance/personnel productivity by 20%. Errors reduction by 10-25%. Improvements of material designs by 25%. Energy saving by 25%. Reduction in process control analysis by 10%.

Table 11. Business decision makers' responses to toolkit Di^{DTPC}

Synthesis:

Four out of six respondents have indicated interest in Toolkit 9 Digital Twin for process control with three of them respondents giving it a score of 5 out of 5, and one rating it 3. All respondents share a vision of enhancing process control, performance, and efficiency within their respective domains through the adoption of the toolkit. They value improvements in different aspects such as reducing energy consumption, waste, errors, and enhancing product quality. Key objectives include reducing scraps, enhancing process control, and collecting relevant process data. The respondents collectively emphasize the importance of manufacturing optimization processes and cost savings achievement with the support of the toolkit.

2.2.4 Technical Leaders Domain

2.2.4.1 DiMAT Data and Assessment Suite

The Data and Assessment Suite is designed to enhance material data safety and traceability, promote the usage of materials derived from renewable resources and boost personnel digital competency. Additionally, it aims to decrease material design costs, economic and environmental impact of materials, and shorten the time to market.

2.2.4.1.1 Toolkit 1 DiMAT Cloud Materials Database – Di^{CMDB}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Increase in personnel productivity by 15-20%. Improve material data safety by 25%. Reduce errors by 25%. Reduce risk by 15%. Launch 1-2 products per year with different materials.
Fundamental capabilities	To quickly collect and get comprehensive and wide information from previous and new materials. To target faster right components for formulations among the various materials, in agreement with targeted properties.
Alignment	Yes

Table 12. Technical leaders' responses to toolkit Di^{CMDB}

Synthesis:

This toolkit has received interest from four of the six respondents. From the technical point of view, key objectives related to personnel productivity, data safety and error and risk reduction are defined, the first three objectives being aligned with those defined from the business point of view. For this purpose, the fundamental capabilities provided by the toolkit would be related to the safe and quick collection of information and its exploitation to determine the best components to be used according to a set of desired properties.

2.2.4.1.2 Toolkit 2 DiMAT Knowledge Acquisition Framework – Di^{KAF}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Increase productivity by 7%. Increase personnel productivity by 15-20%.

	Improve material data safety by 25%. Reduce errors by 25%. Reduce risk by 15%. Save time by 8%. Increase delivery time by 8%. Launch 1-2 products per year with different materials.
Fundamental capabilities	Identifying correlations in relevant data. Reducing model design iterations through improved component behavior prediction and understanding material-property relationships. To target faster right components for formulations among the various materials, in agreement with targeted properties. Easy way of accessing and representing the characteristics of the materials in the KG form.
Alignment	Yes

Table 13. Technical leaders' responses to toolkit Di^{KAF}

Synthesis:

This toolkit has received interest from four of the six respondents. From the technical point of view, stakeholders define key objectives related to productivity, data safety, error and risk reduction and time saving. These objectives are aligned with the objectives defined by business decision-makers, although the technical leaders set slightly more ambitious objectives. The fundamental capabilities are related to the analysis of data for rapid prediction of material behavior and for the identification of the best component or material to be used according to the desired properties.

2.2.4.1.3 Toolkit 3 DiMAT Materials Environmental and Cost Life Cycle Assessment – $Di^{MEC\cdot LCA}$

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Increase personnel productivity by 15%. Increase user performance by 30%.

	Reduce costs by 5%. Improve environmental performance by 7-25%. Decrease GHG emissions by 42%. Getting to know the features of 75% of products.
Fundamental capabilities	Definition of product requirements. Evaluation and comparison of the environmental and economic performance of design options (direct costs, indirect costs, scraps, waste). Reliable calculation of potential economic and environmental savings. Ensure minimal cost and environmental impact. Export data on the environmental impact of the new products.
Alignment	Yes

Table 14. Technical leaders' responses to toolkit $Di^{MEC\text{-}LCA}$

Synthesis:

The stakeholders have shown a great interest in this toolkit, scoring it with 4 or 5 points out of 5. The technical key objectives defined by the stakeholders are related to personal productivity, user performance, costs, and environmental performance. These objectives are fully aligned with those defined from the business point of view. As for the fundamental capabilities, they are related to the evaluation and reduction of economic and environmental cost in product design and to the definition of product requirements.

2.2.4.2 DiMAT Modelling and Design Suite

Modelling and Design Suite aims to enhance material designs and optimize the return on investment (ROI) for personnel training. It's also designed to reduce material design errors and the use of materials during the design and modelling phase, and to boost personnel productivity.

2.2.4.2.1 Toolkit 4 DiMAT Materials Design Framework – Di^{CMDB}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Increase personnel productivity by 20%. Reduce errors by 15%. Translate the ontology-based system to real materials properties saving up to 50% of regular research time.
Fundamental capabilities	Synchronize with connected toolkits to coordinate which data is queried and where it is stored. Obtain interesting properties with the combination of existing materials in databases.
Alignment	Yes

Table 15. Technical leaders' responses to tool Di^{CMDB}

Synthesis:

Four of the six stakeholders showed interest in this toolkit. The technical key objectives are related to errors reduction, time savings and increased personnel productivity, which coincide exactly with the key objectives proposed by the business decision-makers. The fundamental capabilities are related to the synchronization and storage of existing data and its analysis to find out interesting properties that could be obtained by combining existing materials.

2.2.4.2.2 Toolkit 5 DiMAT Materials Modeler – Di^{MM}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Cost reduction by 10%. Reduction in environmental impact by 10%. Increase performance by 10%.
Fundamental capabilities	To predict in advance mechanical and rheological properties depending on composition and processing parameters. Design competitive materials avoiding errors, saving costs and gaining quickness.

Alignment	Yes

Table 16. Technical leaders' responses to tool Di^{MM}

Synthesis:

Four of the six respondents have a relative interest in this toolkit (with a score of 3 out of 5). The key technical objectives defined by the technical leaders are related to cost reduction, environmental impact reduction and performance improvement. These objectives are aligned with the key objectives set by business decision makers, although the requirements given to each of them vary according to the profile of the respondent (business or technical). The key capabilities are related to the prediction of material properties, which would enable error avoidance, cost savings and speed gains.

2.2.4.2.3 Toolkit 6 DiMAT Materials Designer – Di^{MD}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Reduction of required number of physical samples and application tests by 25%. Reduction of workload to acquire information by 60-80%.
Fundamental capabilities	Predict results from limited number of experiments. Reduce steps to get the assessment on the environmental and economic impact of the possible new materials.
Alignment	Yes

Table 17. Technical leaders' responses to tool Di^{MD}

Synthesis:

This toolkit has received interest from half of the respondents. The technical key objectives are related to the reduction of workload to acquire information and the reduction of number of required samples and tests. In line with these objectives, the prediction of results from a limited number of experiments and the reduction of steps required to analyze the information are fundamental capacities.

2.2.4.3 DiMAT Simulation and Optimization Suite

DiMAT Simulation and Optimization Suite is designed to enhance material mechanical properties and performance, increase on-time completion rates, and improve material operational characteristics. It also aims to reduce the amount of material needed for testing and prototyping procedures and to cut down material production costs.

2.2.4.3.1 Toolkit 7 DiMAT Materials Mechanical Properties Simulator – Di^{MMS}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Reduction of costs by 6-15% Increase performance by 5-10%. Increase delivery time by 7%. Reduce errors by 15%. Increase personnel productivity by 20%.
Fundamental capabilities	Reduction of trials required to reach the objectives. Prediction of reliable obtainable mechanical and rheological properties when proposing formulations.
Alignment	Yes

Table 18. Technical leaders' responses to toolkit Di^{MMS}

Synthesis:

This toolkit receives interest from four of the six respondents. The technical key objectives are related to reducing costs and errors, and increasing performance, delivery time and personnel productivity. Key objectives overlap for business decision-makers and technical leaders, although the values set for each objective vary. The fundamental capabilities relate to reducing trials to meet objectives and predicting reliable properties when proposing formulations.

2.2.4.3.2 Toolkit 8 DiMAT Materials Processing Simulator Di^{MPS}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Cost reduction by 15%. Increase in performance by 10%. Increase of on-time completion by 20%. Increase of user experience by 30%. Increase the capability to launch the products of new materials by 25-35%. Reduction in environmental impact by 10%. Reduction of required physical samples and application tests by 25%. Definition of optimized parameters (temperature and pressure) for composite manufacturing processes.
Fundamental capabilities	Use of rheological and physical properties of resin matrix to determine optimized pressure and temperature profiles for manufacturing composite structures. Utilization of reliable calculations and production feedback to predict outcomes accurately. Achievement of predictions from a limited number of experiments. Prediction of properties achievable with an optimal set of parameters in twin-screw extrusion for different formulations.
Alignment	Yes

Table 19. Technical leaders' responses to toolkit Di^{MPS}

Synthesis:

Most respondents find this toolkit interesting (5 out of 6). The key objectives for technical leaders are related to cost reduction, environmental impact, number of samples needed and increase of user experience, on-time completion, and the ability to launch new products. In this case, the key objectives are slightly different for technical leaders and business decision-makers. The fundamental capabilities are therefore related to the prediction of product properties, which allows the reduction of the number of tests and the correct configuration of the manufacturing environment.

2.2.4.3.3 Toolkit 9 DiMAT Digital Twin for Process Control – Di^{DTPC}

Technical leaders have identified the following technical key objectives, capabilities and whether the new capabilities are aligned with the identified business and technical key objectives:

Technical key objectives	Reduction of errors by 25%. Reduction of specific energy consumption by 25%. Reduction of process control analysis by 10%. Increase of on-time completion by 20%. Increase in performance/personnel productivity by 20%. Increase user experience by 30%.
Fundamental capabilities	Implementation of an alert system to identify when certain process parameters may lead to increased scraps. Management, testing and prediction of the behavior of new products before being created. Optimization of process parameters to minimize specific energy consumption and determining the key parameters for process control analysis. Evaluation of the influence of process parameters and other factors (material properties, environmental data) on performance of composite production.
Alignment	Yes

Table 20. Technical leaders' responses to toolkit Di^{DTPC}

Synthesis:

Four of the six respondents show interest in the toolkit, three of them giving it the highest possible rating. The technical leaders set key objectives related to reducing errors, energy consumption and process control analysis, and increasing on-time completion, productivity, and user experience. Many of the objectives defined by technical leaders coincide with those defined by business decision-makers. The fundamental capabilities are related to optimizing process parameters, implementing an alert system that helps identify which process parameters will lead to scraps, and predicting the behaviour of new products before they are created.

2.3 DIMAT BUSINESS VIEWPOINT CONCLUSIONS

The business perspective has made it possible to identify the suites and toolkits of greatest interest to stakeholders. In addition, it has enabled the identification of the specific aspects of improvement that stakeholders expect with the vision and values, and the quantification of these values with the Key Objectives. To identify the toolkits of greatest interest, the following table presents the toolkits in order from highest to lowest average score obtained:

	Natureplast					IGC					Hegla					CETMA					ACCELI					TECNORED					
Toolkit	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	Score
Materials Environmental and Cost Life Cycle Assessment					x					x					x					x				x					x		4,67
Materials Processing Simulator					x				x						x					x		x							x		4,17
Digital Twin for Process Control		x								x					x					x			x				x				3,67
Knowledge Acquisition Framework				x			x							x		x								x						x	3,33
Materials Mechanical Properties Simulator					x		x							x				x						x			x				3,33
Cloud Material Database				x			x						x			x							x						x		2,83
Material Design Framework		x					x						x					x					x						x		2,83
Materials Modeler			x				x				x							x					x					x			2,50
Materials Designer	x									x	x							x			x							x			2,33

Table 21. Toolkits in order from highest to lowest average score

The DiMAT Materials Environmental and Cost Life Cycle Assessment toolkit ($Di^{MEC\cdot LCA}$) is the one that shows the most interest from a business perspective. All stakeholders rate this toolkit with a high level of interest. In second place, the DiMAT Material Processing Simulator toolkit (Di^{MPS}) is of great interest to most stakeholders.

On the other hand, the DiMAT Material Designer (Di^{MD}) toolkit is the least interesting, from a global perspective, with only three of the stakeholders showing interest. From the perspective of the suites, the following table shows the score of each suite as the sum of the average scores of the toolkits that comprise it:

		Natureplast					IGC					Hegla					CETMA					ACCELI					TECNORED						
Suite/Toolkit		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	Score	
Data and assessment Suite	Cloud Material Database				x			x						x			x							x						x		2,83	10,8
	Knowledge Acquisition Framework				x			x							x		x								x						x	3,33	
	Materials Environmental and Cost Life Cycle Assessment					x					x					x					x				x					x		4,67	
Modelling and Design Suite	Material Design Framework		x					x						x					x					x						x		2,83	7,6
	Materials Modeler			x				x				x							x					x					x			2,50	
	Materials Designer	x									x	x							x			x							x			2,33	
Simulation and Optimisation Suite	Materials Mechanical Properties Simulator					x		x							x				x						x			x				3,33	11,1
	Materials Processing Simulator					x				x						x					x		x							x		4,17	
	Digital Twin for Process Control		x								x					x					x			x				x				3,67	

Table **22.** Score of each suite as the sum of the average scores of the toolkits

In this case, the DiMAT Simulation and Optimization Suite is the most interesting from a business perspective, followed by the DiMAT Data and Assessment Suite. However, it can be concluded that the DiMAT Modelling and Design Suite raises less interest among the stakeholders surveyed. Regarding the specific aspects of improvement provided by the suites, it can be indicated that:

- DiMAT Data and Assessment Suite. The improvement in access to information in a faster way and reducing errors for prototyping and in the design of new materials is the greatest interest of the stakeholders. It highlights the importance of the Materials Environmental and Cost Life Cycle Assessment that encompasses multiple objectives such as waste reduction, emission reduction or energy consumption reduction.
- DiMAT Modelling and Design Suite. The reduction in the number of physical experiments required for the development of new materials and the speed of introduction of new materials is perceived to be of great interest in reducing time-to-market, having greater control over the process, and offering better and faster solutions to the market.
- DiMAT Simulation and Optimization Suite. The improvement of competitiveness with cost reduction, reduction of time-to-market and above all a better control of the process parameters are the factors of interest that stand out, and that mainly lead to an improvement in term of product quality and process efficiency.

3 USER VIEWPOINT

This task will oversee defining how to implement the capabilities and structure of the DiMAT Framework. To do this, we will identify several key points that the IIRA reference architecture presents to us. These are the tasks, roles, activities, and parts of DiMAT that focus on software development and human intervention. Once the previous elements have been identified, we will go one step further to define both the functional map, the specifications of the roles responsible for the tasks, and the implementation maps. For each of the tasks, the triggers for each activity are defined, considering the workflows, effects, and constraints.

3.1 USAGE VIEWPOINT APPROACH FOR DIMAT

3.1.1 Usage Viewpoint Concepts

The Usage Viewpoint will build on the objectives previously established in the Business Viewpoint to identify the contexts and scenarios in which the system will be used. This will include the entities and actions involved in such use. The activities derived from this analysis will be used to define the system requirements and in turn guide its design, development, implementation, deployment, operation, and improvement process (**Figure 4**). These subsequent elements will be integrated into the functional approach of the system. In this section we will focus on getting an overview of the actors involved, from the stakeholders to the roles used, the activities performed, and the concrete tasks carried out.

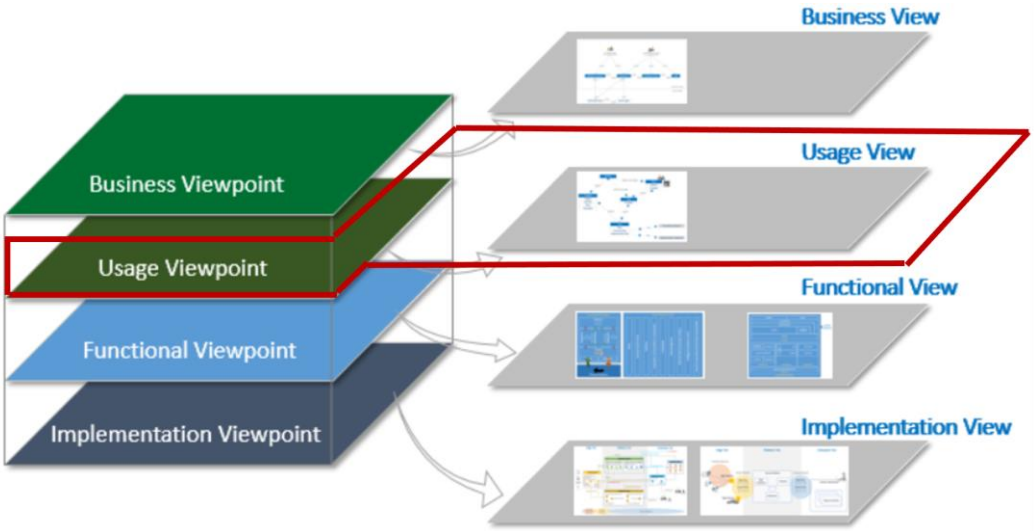


Figure 4: Usage Viewpoint layer

3.1.2 Usage Viewpoint Methodology

Thanks to the analysis of the business and technical perspectives from the business viewpoint, it is possible to define the usage activities and system requirements from the usage perspective. At this point, three essential elements can be identified from the business viewpoint: Roles, activities and tasks.

The usage viewpoint acts as a guide for the development of the functional and implementation viewpoints. Through this, several tasks can be defined, which are considered as the basic unit of work, such as the execution of an operation, the transfer of data or the actions of a part. Every task will consist of a role or roles responsible for carrying out the task. A role represents a set of capabilities that an entity assumes to participate in the execution of certain tasks and is generally associated with certain security properties, such as authorisations, privileges, and permissions. Roles can be assumed by one or several parties.

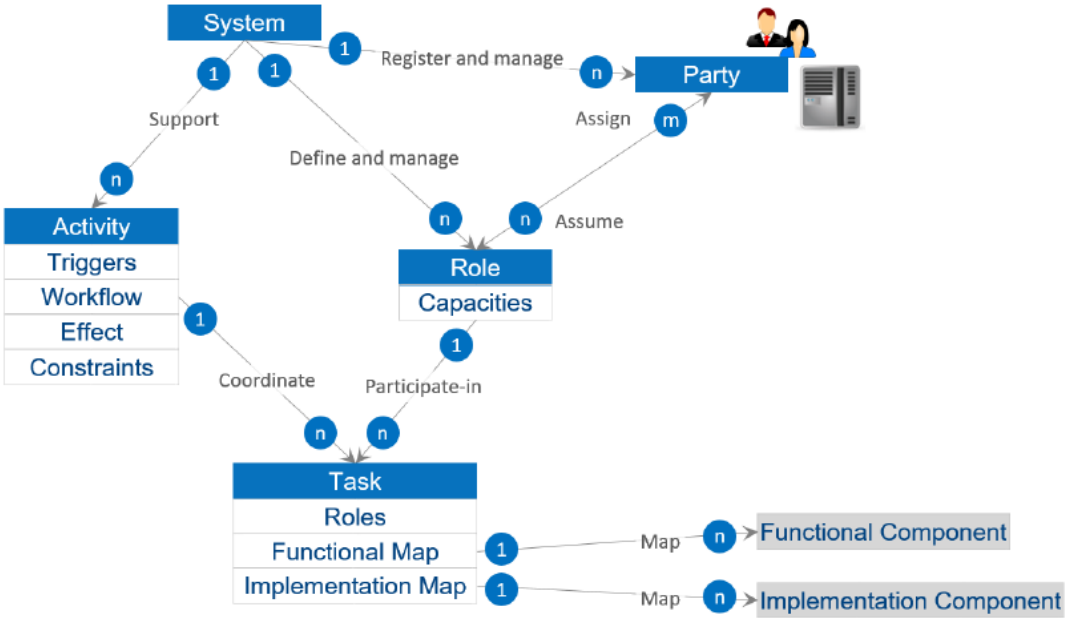


Figure 5: Key concepts on the Usage viewpoint

To develop this part of the deliverable, we will start by making a use case diagram, which will allow us to identify which actors interact with the solutions. Secondly, we will analyse the roles of the different actors interacting with the solution, allowing us to define the activities. Finally, we will analyse in depth the tasks that will be carried out to develop the activities defined above.

3.1.2.1 Usage Case Diagram Actors

Describe in a use case diagram the actors (roles) of the toolkit and the main use cases (activities). This diagram may serve as an overview for the toolkit's main functionalities, as well as the main parties involved.

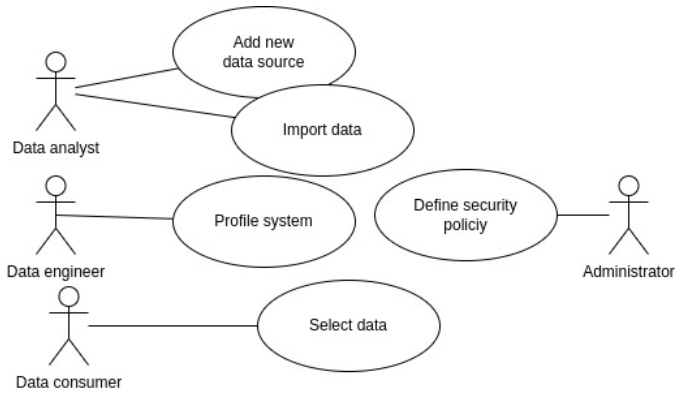


Figure 6: Example Use Case Diagram

3.1.2.1 Roles

Describe the different roles involved in the toolkit. Each role includes a collection of capacities for using the system. Example:

- **Data analyst** (or a data analysis team): responsible for defining new data sources, data source to internal storage mappings and triggering import data processes.
- **Data engineer**: responsible for monitoring and optimizing internal storage subsystems.
- **Data consumer**: any external entity (human being or automated system) which selects data from the toolkit.
- **Administrator:** responsible for defining security policies, as well as assigning roles and credentials to the different parties.

3.1.2.2 Activities

Here all the activities (main processes of the toolkit) should be included. They might also be defined using a card, template or similar. Given the complexity of certain actions, it is essential to provide users with clear and comprehensive guidance. While this deliverable focuses on the design and preparation of solutions, guidelines and manuals will be

developed in parallel within Work Packages 4, 5, and 6 of the projects. These resources will offer detailed information on installation, system integration, and effective usage of the solutions. To facilitate effective navigation of the toolkit, we will provide clear instructions to guide users through each process, ensuring they follow the necessary steps. Additionally, we will include examples to illustrate common best practices, providing practical guidance for users. These guidelines will not only simplify complex tasks but also enhance the overall user experience.

3.1.2.2.0 Represents a high-level process

An activity represents a high-level process of the toolkit. An overall description of the activity is required here. All activity subcomponents (triggers, workflow, effects, constraints, mockups) should be included in subsections. Example of activity **"Add new data source":** The data analyst defines a new data source for obtaining information about materials.

3.1.2.2.1 Triggers

An activity gets triggered by some role/s, or/and because some conditions are met. Example of activity **"Add new data source"**: The user explicitly selects the option "Add new data source".

3.1.2.2.2 Workflow

A process involves the execution of several tasks. These tasks may execute sequentially, in parallel, conditionally, repeatedly, etc. It can be summarized using a UML activity diagram.

Example of activity **"Add new data source":** Activity diagram:

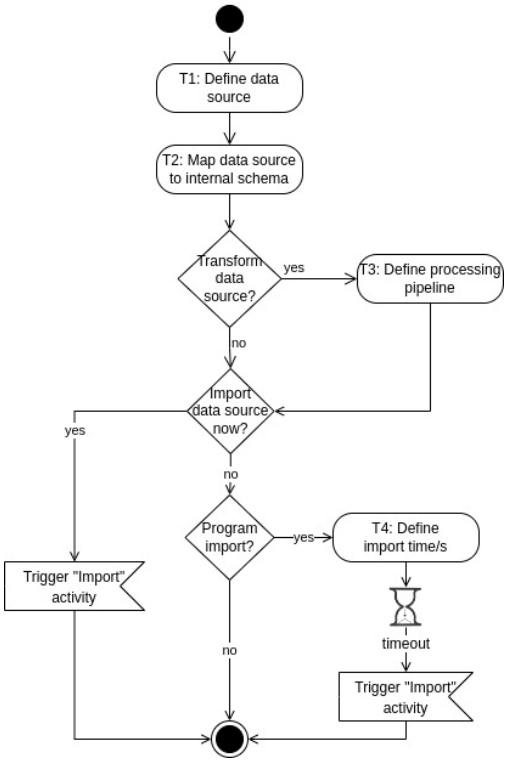


Figure 7: Example Activity diagram

Tasks from **"Add new data source":**

- Task T1 "Define data source":
  - The user specifies the data source URL, type, required credentials, etc.
- Task T2 "Map data source to internal schema":
  - The data source tables, fields, etc. are mapped to the internal database schema.
- Task T3 "Define processing pipeline":
  - Define the processing steps the input data source must go through before inserting data into the internal storage system. These processing steps may be considered long batch operations (e.g. removals, castings, joins, etc.)
- Task T4 "Define import time/s":
  - Program the import operation for a later time.
  - Import operations may take place once or periodically, to keep data synchronized with external sources.

3.1.2.2.3 Effects

The activity may involve some permanent effects in the system. Example of activity **"Add new data source":**

- The data source gets registered into the system, along with the required steps to transform the data before storing it into the system.
- If the import is promptly required, then the activity "Import" is requested.
- If the import is programmed for a later time, then the activity "import" is requested sometime later.

3.1.2.2.4 Constraints

Here we should list all the constraints that must be preserved in the system. Example of activity **"Add new data source"***:*

- The data source, along with the specified transformations must meet the internal database schema.

3.1.2.2.5 Mockups

This section presents the mockups of the user interface, showcasing the windows that different user roles will interact with during the activity. These mockups help illustrate the interaction between various software components and the user interface. The design and prototyping tool FIGMA [2] were used to create and develop these mockups.

The user interface will maintain consistency across the various toolkits, not only in terms of structure and front-end layout but also in visual identity, including typography, color schemes, and iconography, all of which are standardized as per the design guidelines set by the project's technical manager. This ensures a cohesive user experience across the three DiMAT suites. All interfaces will be developed using REACT [3], leveraging its component-based architecture for reusability and modularity. This approach facilitates a scalable and maintainable codebase, enabling uniform interaction patterns and seamless integration between the different toolkits. React's virtual DOM and efficient state management will contribute to optimized performance across diverse toolkits, further enhancing user experience consistency.

3.2 DIMAT CLOUD MATERIALS DATABASE (DI^{CMDB})

The Cloud Materials Data Base (CMDB) offers homogeneous access to heterogenous data storage. Processes, linked to a CHADA and MODA definition, will be represented, and their different steps connected to material data. Upon uploading a certain file, the CMDB offers different data processing options to extract and generate semantic and relational data from the files. All data is then stored and interconnected for future retrieval.

3.2.1 Overview

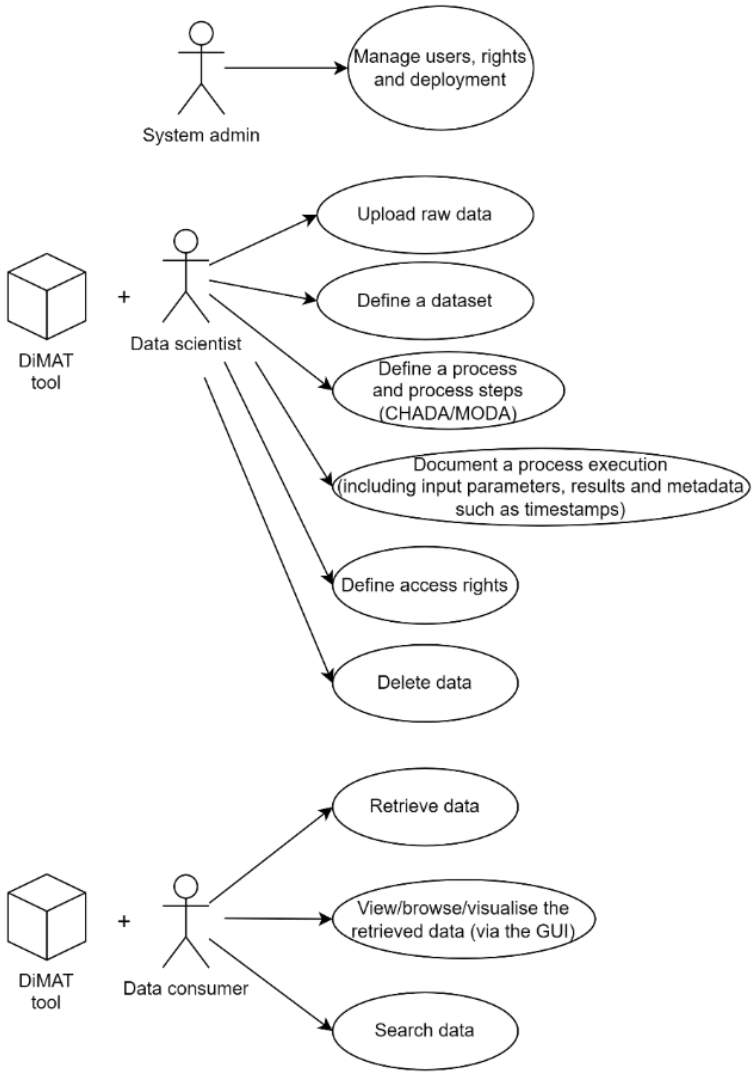


Figure 8: Use Case diagram from Di^{CMDB}

3.2.2 Roles

- **System admin:** Has full control over the system and its management.
- **Data scientist:** Generates data that will be stored in the CMDB.
- **Data consumer:** Browses the data stored in the CMDB.

3.2.3 Activities

Users should be able to manage the access rights of their uploaded data (e.g., public access, restricted access to user groups, etc.)

3.2.3.1 Activity 1 – Upload a new dataset

This activity covers how users interact with the CMDB to upload new datasets.

3.2.3.1.0 Activity 1 - Triggers

A user or a software tool (includes other DiMAT toolkits) decides to store some data in the CMDB.

3.2.3.1.1 Activity 1 - Workflow

Datasets can be standalone or connected to a process or process step.

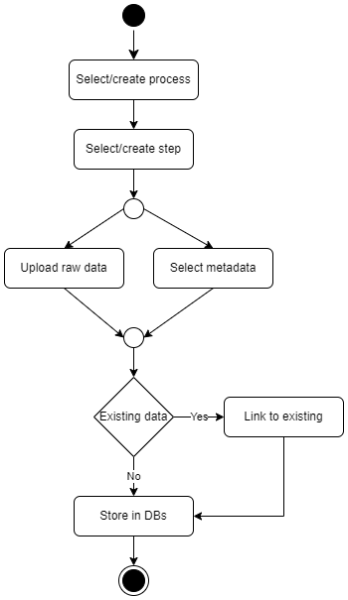


Figure 9: Workflow "Upload a new dataset" of Di[CMDB]

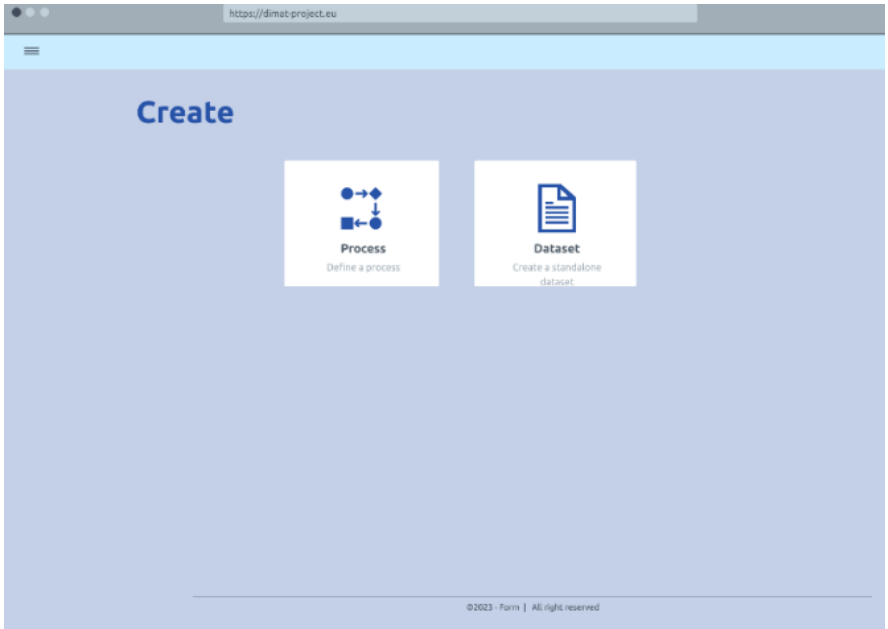
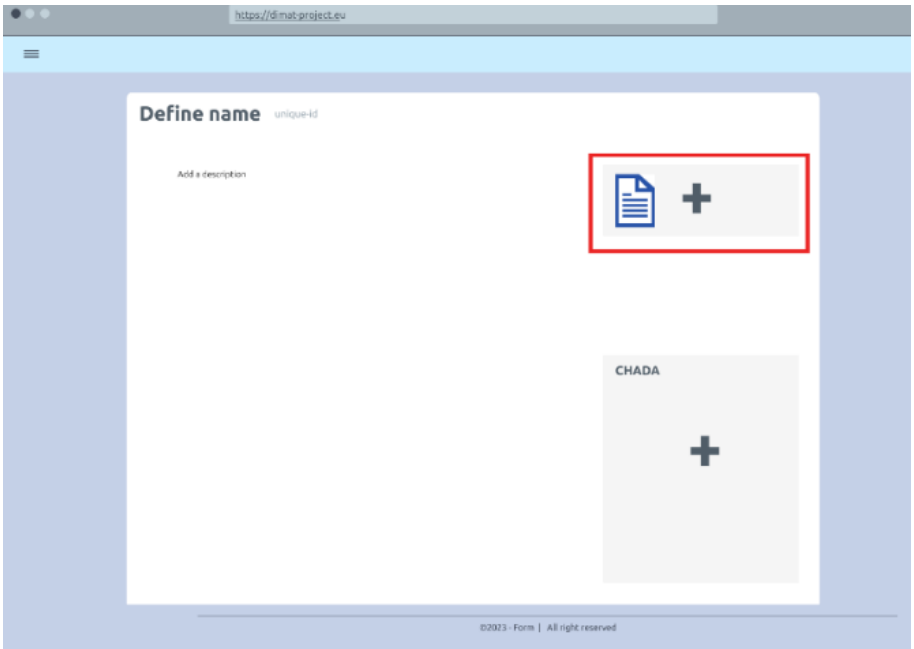
3.2.3.1.2 Activity 1 - Effects

- A new dataset element is created, and the files stored.
- The dataset is now findable via search.
- Additional metadata is stored for provenance.
- CHADA/MODA information is specified as well.

3.2.3.1.3 Activity 1 - Constraints

- Users must be logged in.
- If the data is connected to a process or step, they must have enough rights to do so.
- The number of files per dataset could be limited.

3.2.3.1.4 Activity 1 - Mockups

Function	Mockup
Create a new knowledge item	
Add files to a knowledge item	

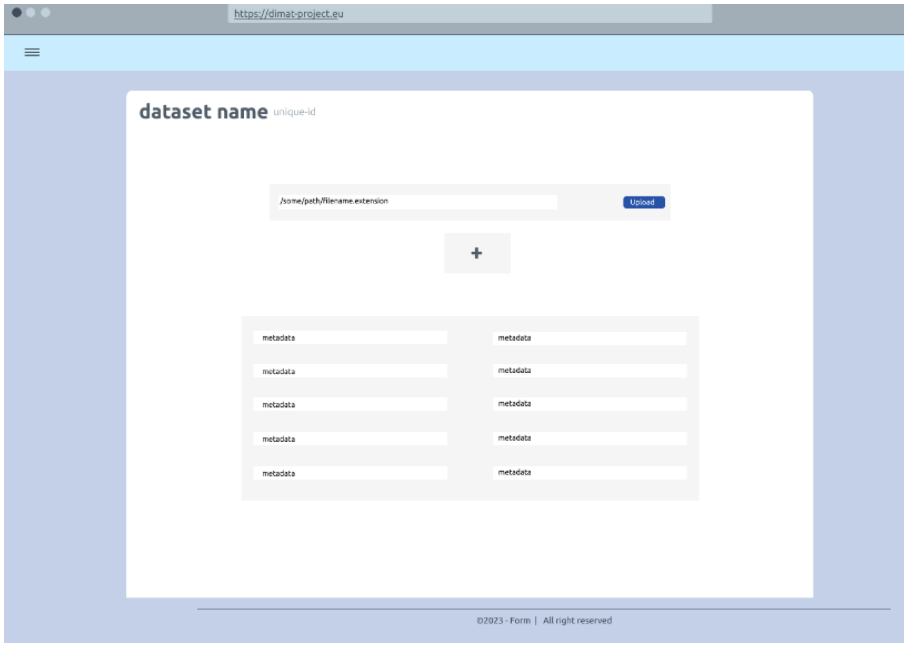
Add metadata to a knowledge item	

Table 23. Di^{CMDB} Mockups activity "Upload a new dataset"

3.2.3.2 Activity 2 – Find a specific process

The other side of data storage is data exploration. Users should be able to navigate all the stored data and find meaningful and helpful results.

3.2.3.2.0 Activity 2 – Triggers

A user wants to find all the information regarding a specific instance of a process or find a dataset.

3.2.3.2.1 Activity 2 – Workflow

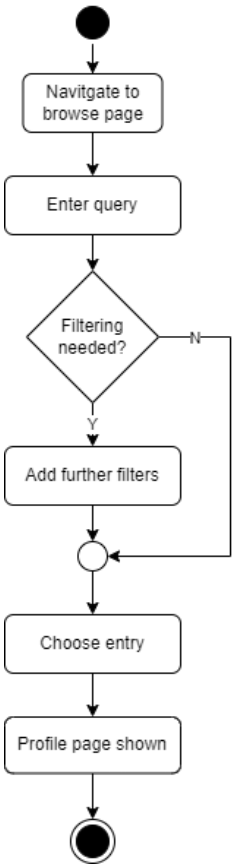


Figure 10: Workflow "Find a specific process" of Di^{CMDB}

3.2.3.2.2 Activity 2 – Effects

The user finds the desired result among all the data stored and can access the contained information. This includes CHADA or MODA specifications.

3.2.3.2.3 Activity 2 – Constraints

- User must be logged in.
- In order to view a certain dataset/process, the user must have enough rights.

3.2.3.2.4 Activity 2 - Mockups

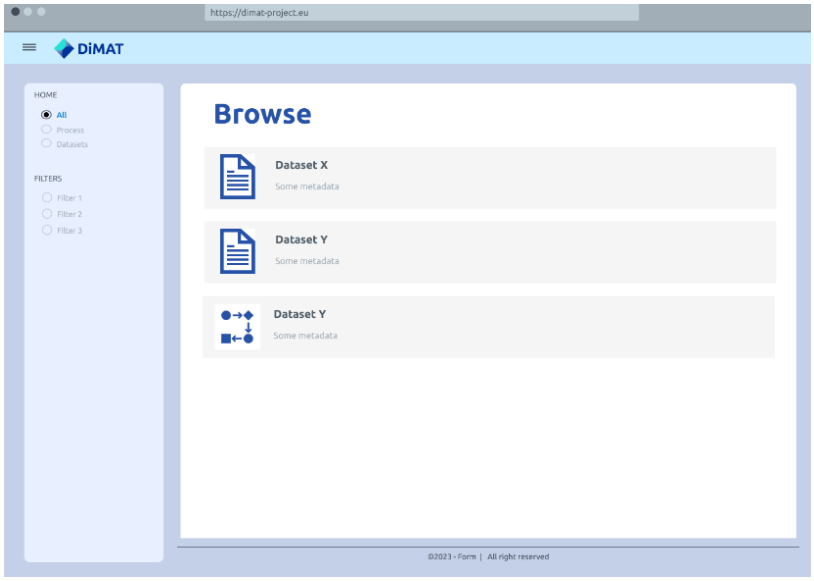
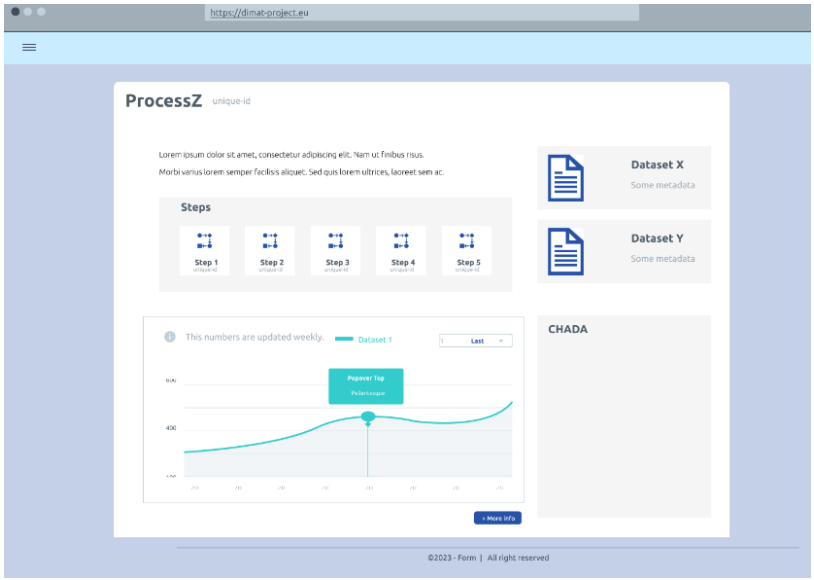
Function	Mockup
Browse all available knowledge items	
View a specific knowledge item	

Table 24. Di^{CMDB} Mockup activity "Find a specific process"

3.2.3.3 Activity 3 – Querying data via the Python SDK

Users and machines (e.g., other toolkits) can connect to the CMDB in order to access available data.

3.2.3.3.0 Activity 3 – Triggers

A user wants to access the files in a dataset, or a toolkit requires some information from the CMDB.

3.2.3.3.1 Activity 3 – Workflow

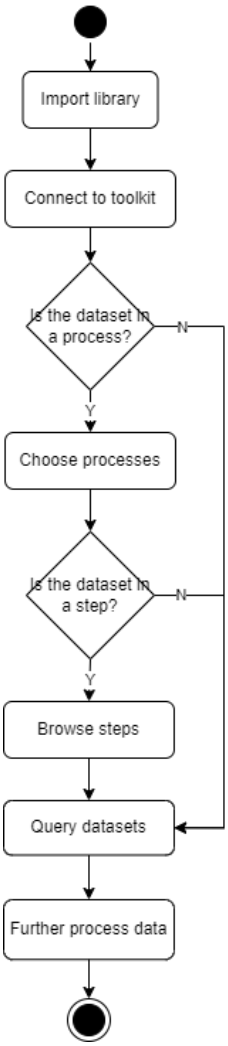


Figure 11: Workflow "Querying data via the Python SDK" of Di^{CMDB}

Connecting to the toolkit requires importing the library in the script and specifying the required parameters.

3.2.3.3.2 Activity 3 – Effects

Users and machines can interact with the toolkit as part of a script and directly access the data without needing to use a GUI.

3.2.3.3.3 Activity 3 – Constraints

- Users must be logged in.
- Users must provide an appropriate token to interact with the toolkit.

3.2.3.3.4 Activity 3 - Mockups

The interaction in this case is not via a GUI, but via scripting. The code would look something like:

```
from dimat.toolkits import cmdb

my_cmdb = cmdb.connect("https://dimat-project.eu/cmdb", "a_secret_token")
process = my_cmdb.get_process("process_id")

dataset = process.get_dataset("datset_id")
dataset.get_raw_data()
```

3.3 DIMAT KNOWLEDGE ACQUISITION FRAMEWORK (DI^{KAF})

The DiMAT KAF toolkit’s objective is to represent knowledge about materials in a structured manner. Besides storing just information about different materials, it also represents relationships among them. The KAF toolkit is based on the development of a Knowledge Graph (KG). The entities of the knowledge graph are going to be concepts like materials and manufacturing processes and the relationships joining these entities reveal links between these concepts. The KG is going to be constructed by employing existing material ontologies (e.g., the EMMO) as well as knowledge deriving from the DiMAT partners. Besides querying the KG for information, the KAF toolkit will support operations such as recommendations on the selection of materials and visualization.

3.3.1 Overview

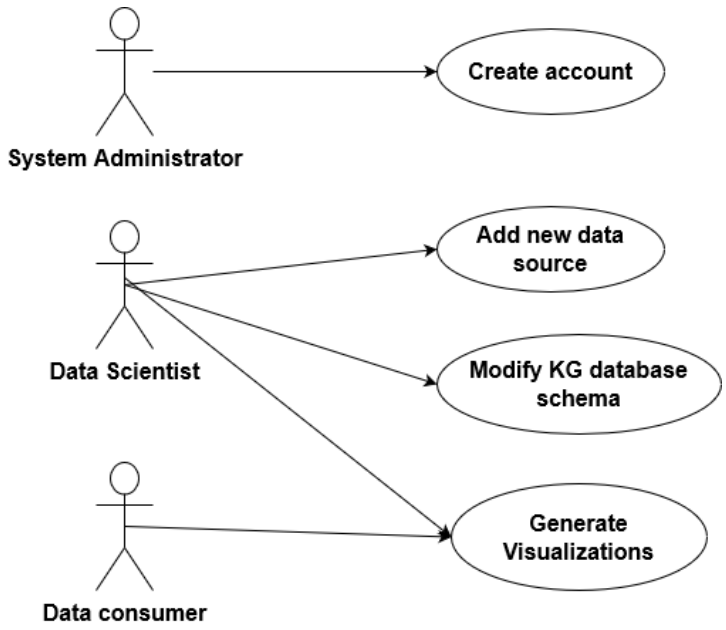


Figure 12: Use Case Diagram of Di[KAF]

3.3.2 Roles

The roles specified for this toolkit are the following:

- **System administrator:** responsible for supervising the smooth operation of the toolkit, adding or removing eligible users of the toolkit.
- **Data scientist:** person with knowledge on the field of materials, capable of identifying new external data sources that can be incorporated into the toolkit and managing the knowledge graph database schema when updates are necessary.
- **Data consumer**: a stakeholder with interest in accessing the toolkit in order to take advantage of its functionalities, such as a materials scientist or engineer, a researcher in the field of either databases or materials.

3.3.3 Activities

Below the main activities of the toolkit are described in detail. These activities are:

- Get visualization of data.
- Add new data source.
- Modify Knowledge Graph.

- Create account.

3.3.3.1 Activity 1- "Get visualization of data"

This activity describes the process of a user requesting to visualize specific data concerning processes or materials stored in the KAF toolkit, for exploration and analysis, deriving insights that can aid decision-making or drive further investigations.

3.3.3.1.0 Activity 1 – Triggers

Data Consumer provides the toolkit with the type and parameters of data they wish to access.

3.3.3.1.1 Activity 1 – Workflow

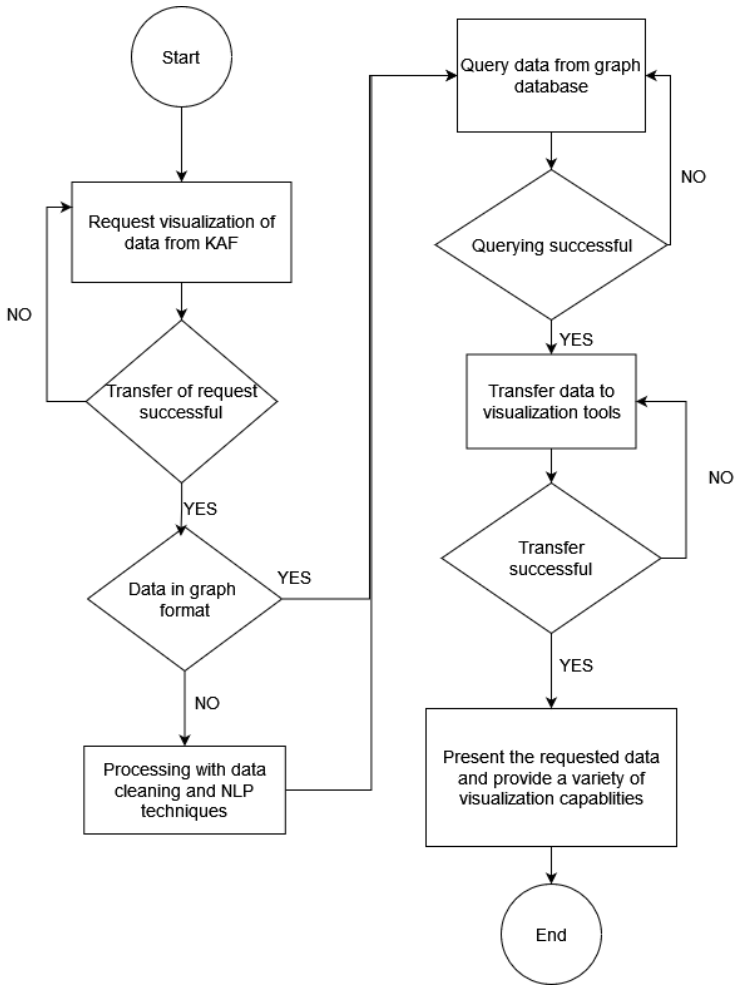


Figure 13: Workflow "Get visualization of Data" of Di^{KAF}

3.3.3.1.2 Activity 1 – Effects

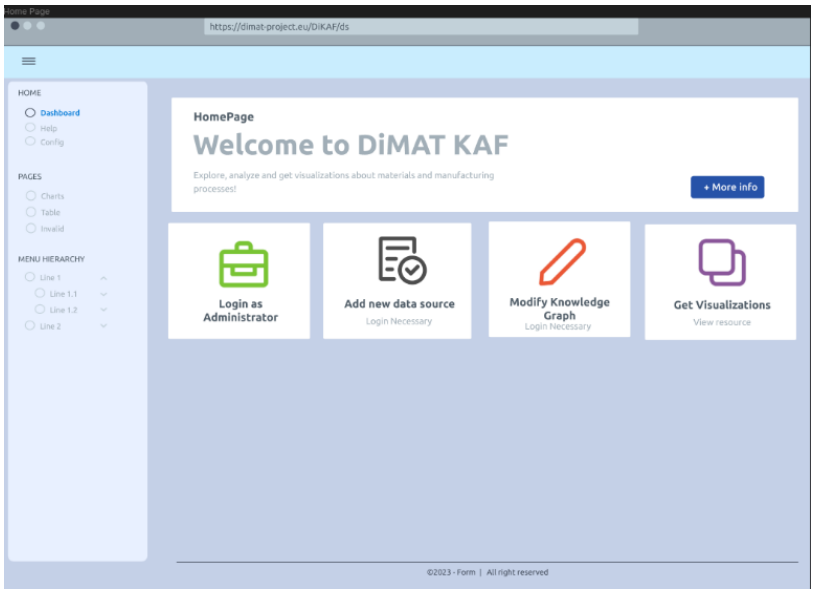
Display data to the user along with visualization capabilities.

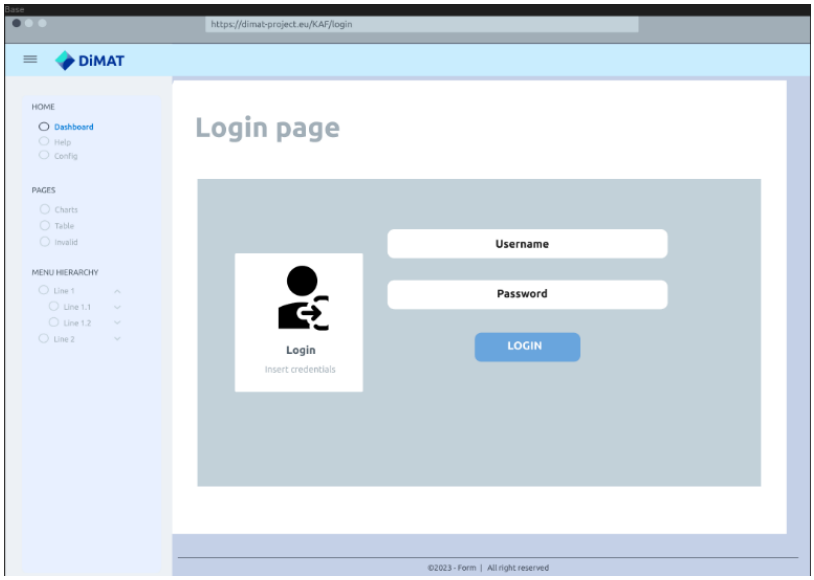
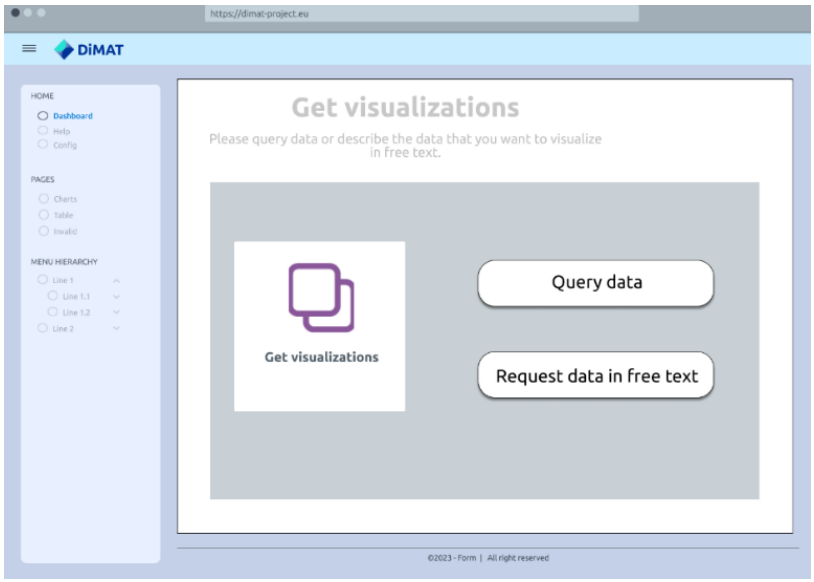
3.3.3.1.3 Activity 1 – Constraints

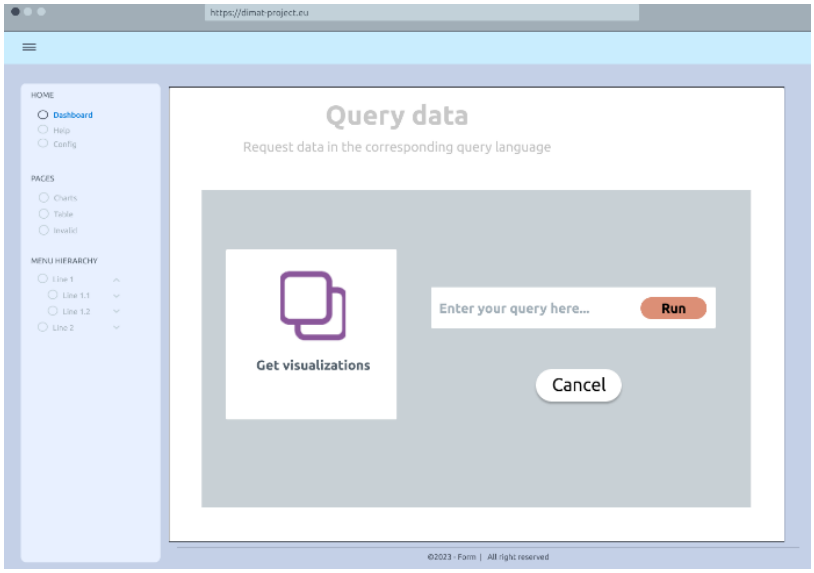
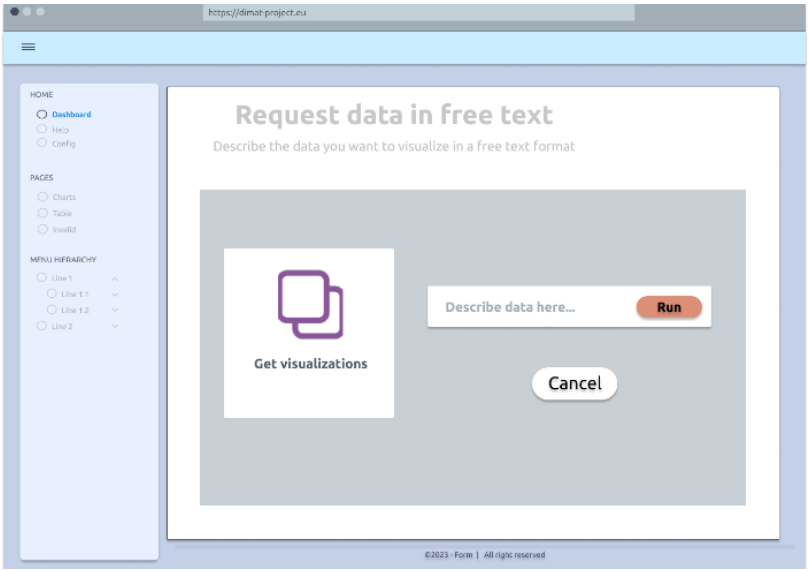
- The user should attempt to access the type of data already stored in the KAF toolkit.

- Data should be in an appropriate format to be handled by Neo4j database and visualization toolkits.
- All data must respect privacy and security rules.

3.3.3.1.4 Activity 1 – Mockups

Function	Mockup
View Homepage	

View Login page	
Select method of querying	

Provide Query	
Provide free text	

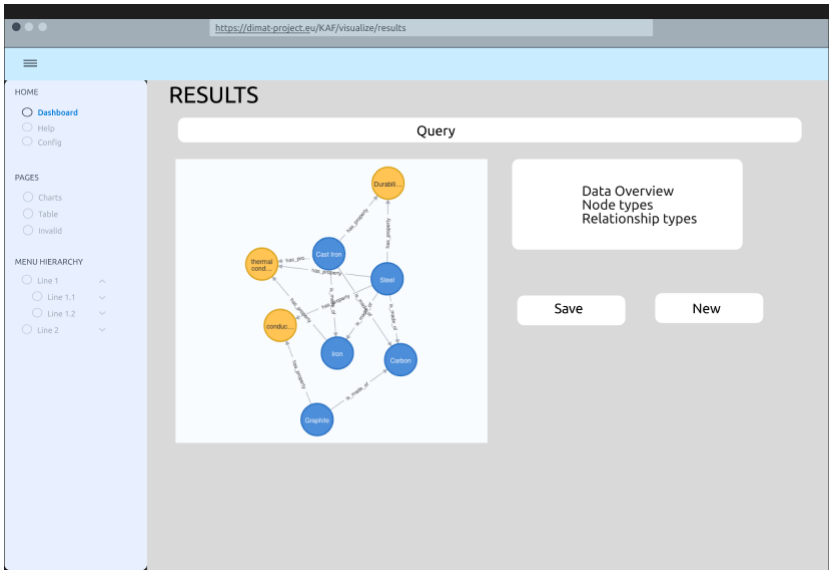
Display visualization results	

Table 25. Di[KAF] Mockup Activity "Get visualization of data"

3.3.3.2 Activity 2 - “Add new data source”

This activity concerns the inclusion of new data to the KG of the KAF toolkit. A data scientist can identify such suitable sources of data deriving from ontologies, experiments, or any other sources.

3.3.3.2.0 Activity 2 – Triggers

The data scientist provides the details of the dataset (URI, type, size) to the toolkit so it can connect and fetch the data.

3.3.3.2.1 Activity 2 – Workflow

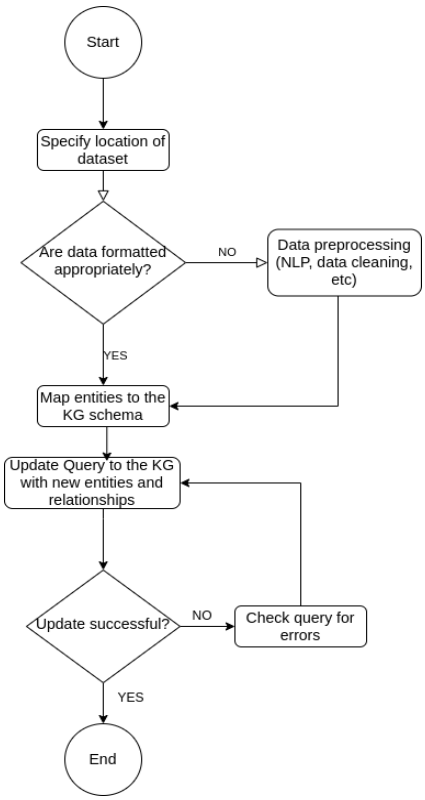


Figure 14: Workflow "Add new data source" of Di^{KAF}

3.3.3.2.2 Activity 2 – Effects

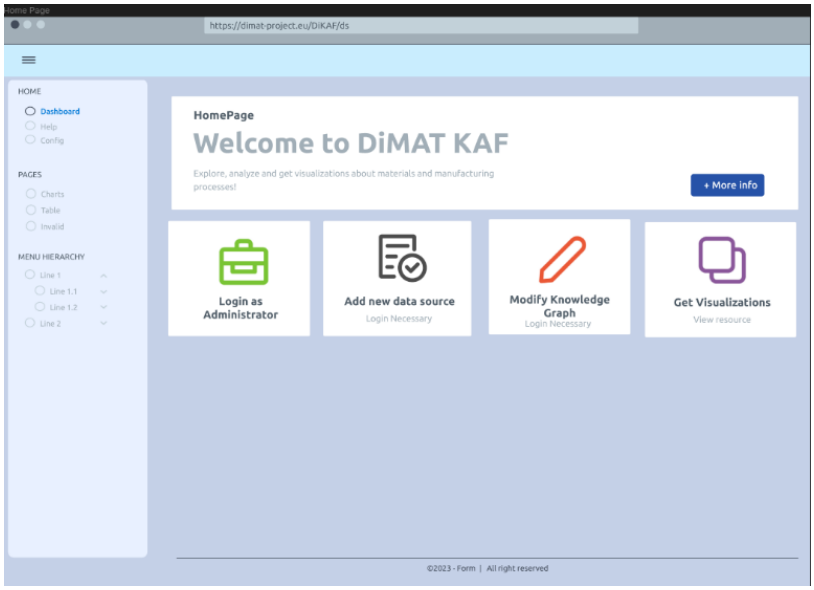
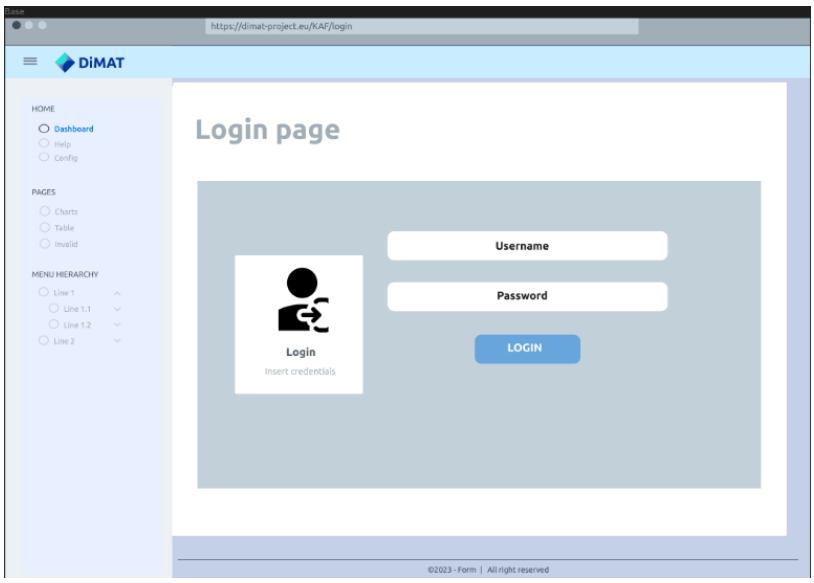
New data is incorporated now to the KG, which is enriched with new entities and relationships.

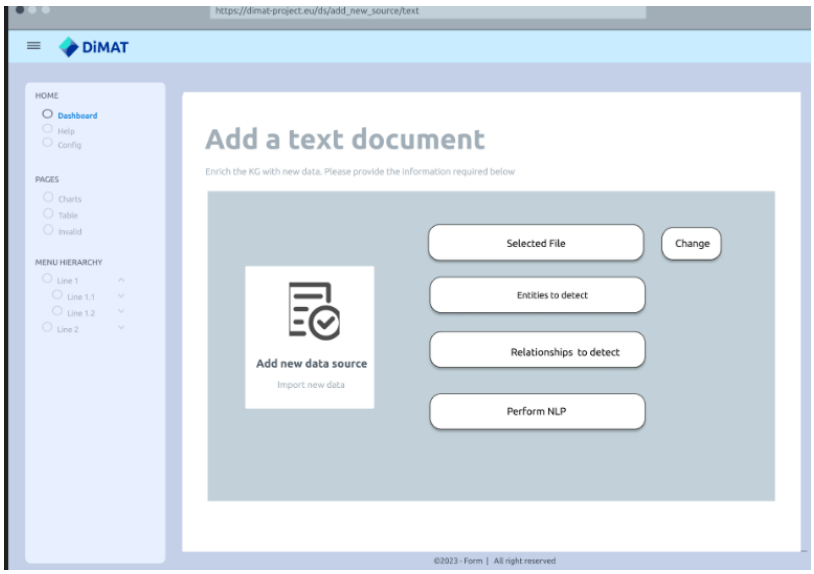
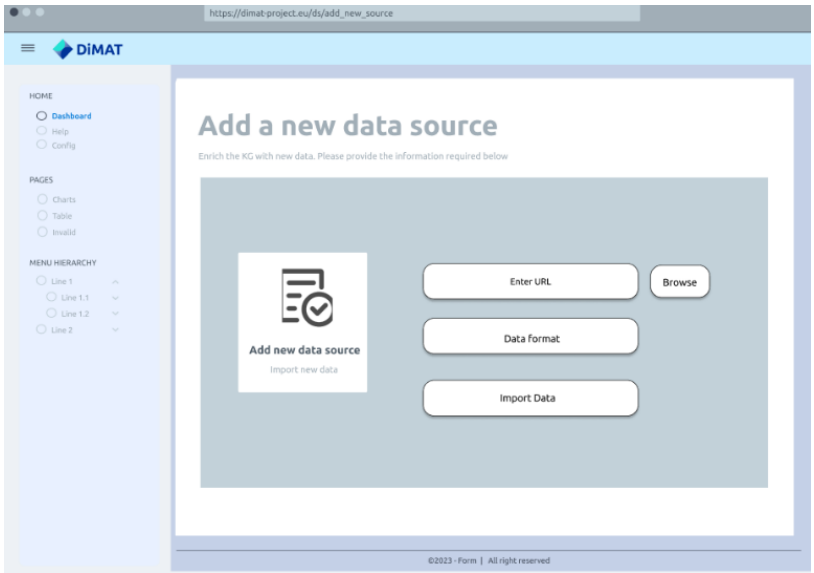
3.3.3.2.3 Activity 2 – Constraints

- The data to be added must be in suitable formats (e.g., RDF, JSON, CSV, etc.).
- The data must be semantically correct. This means that their inclusion must be meaningful to the KAF toolkit.
- Data should be compliant with privacy regulations and not expose sensitive information (anonymized data, data publicly available, etc.).

- Data sources should be credible to ensure the KAF toolkit's quality of suggestions, data analytics and visualizations.

3.3.3.2.4 Activity 2 – Mockups

Function	Mockup
View Homepage	
View Login page	

Provide text document details	
Upload data file	

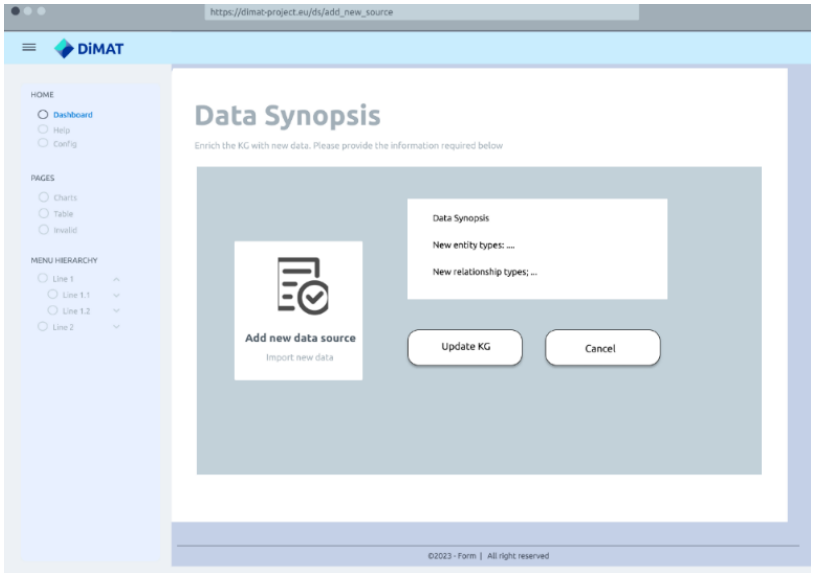
View data synopsis	

Table 26. Di^{KAF} Mockup Activity "Add new data source"

3.3.3.3 Activity 3 - “Modify Knowledge Graph”

Data scientists can modify the knowledge graph when updates are needed or to correct any mistakes.

3.3.3.3.0 Activity 3 – Triggers

The data scientist forms a query for updating the knowledge graph.

3.3.3.3.1 Activity 3 – Workflow

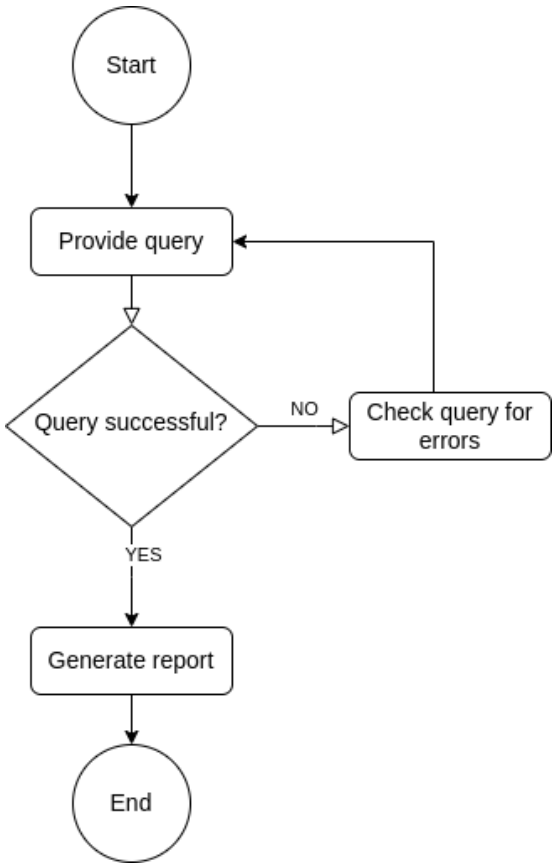


Figure 15: workflow "Modify Knowledge Graph" of Di^{KAF}

3.3.3.3.2 Activity 3 – Effects

Existing data are modified to reflect new knowledge about materials and their processes as well as correct any errors or omissions.

3.3.3.3.3 Activity 3 - Constraints

- The user should attempt to access the type of data already stored in the KAF toolkit.
- The user's queries should respect the data privacy concerns.
- The updates should be semantically correct and respect the KG database schema.

3.3.3.3.4 Activity 3 – Mockups

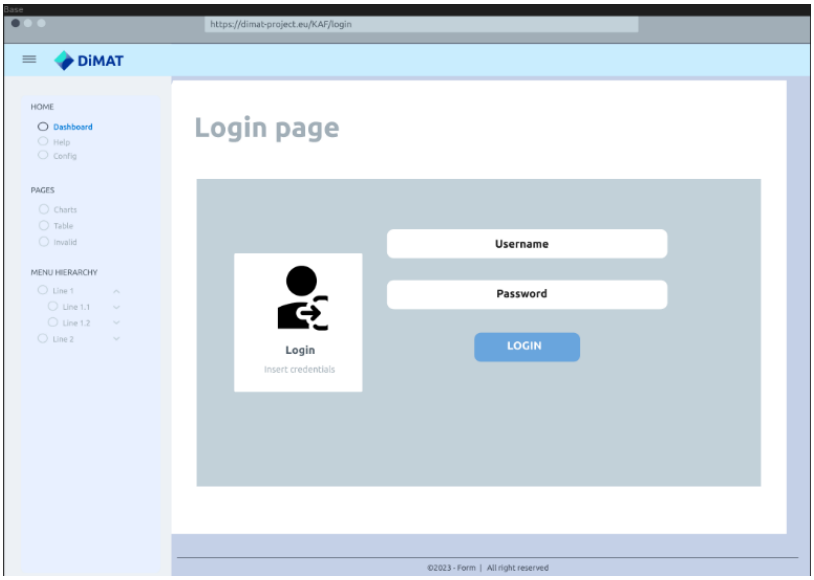
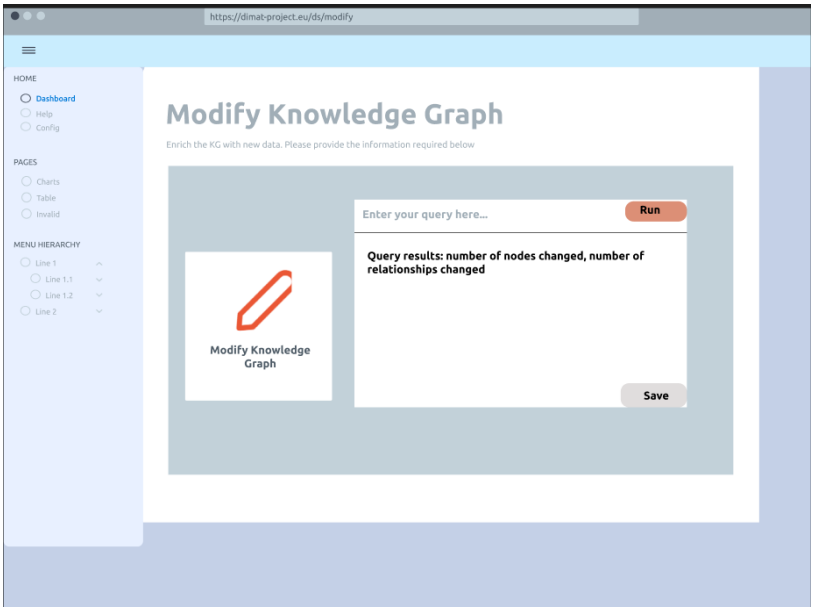
Function	Mockup
View Login page	
Provide query	

Table 27: Di^{KAF} Mockup Activity " Modify Knowledge Graph"

3.3.3.4 Activity 4 - “Create Account”

The System Administrator is responsible for creating and managing the accounts for the end users that can use the KAF, considering both data scientists and data consumers. Proper access rights are going to be provided according to the end user type.

3.3.3.4.0 Activity 4 – Triggers

For usage of the KAF by the DiMAT partners, requests for creation of new users will be directed to NTUA, at least for the initial development and experimentation phase. Following, a registration page will be made available where data consumers may be registered and acquire access to targeted KAF services. Data population mechanisms will be accessible for usage and/or update and maintenance by data scientists. The proper access rights will be allocated by the system administrator.

3.3.3.4.1 Activity 4 – Workflow

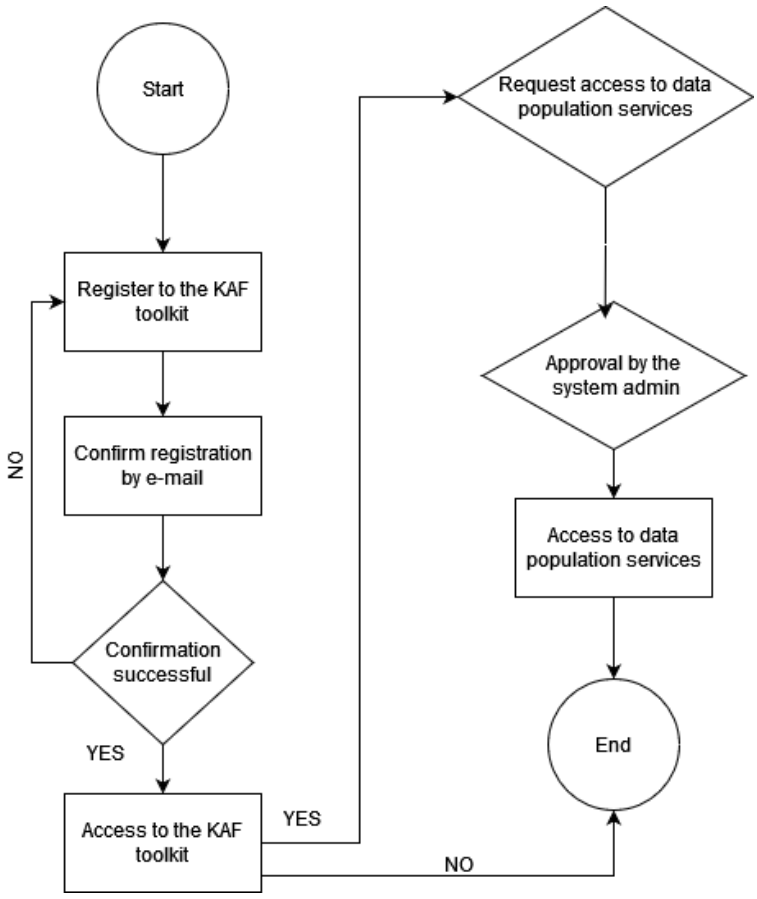


Figure 16: Workflow "Create Account" of Di^{KAF}

The registration workflow is going to consist of the following steps:

- Register to the KAF toolkit.
- Confirm registration by e-mail.
- Get access to the KAF toolkit as data consumer.
- Optional: Request access to data population services.
- Approval to be provided by the system administrator.

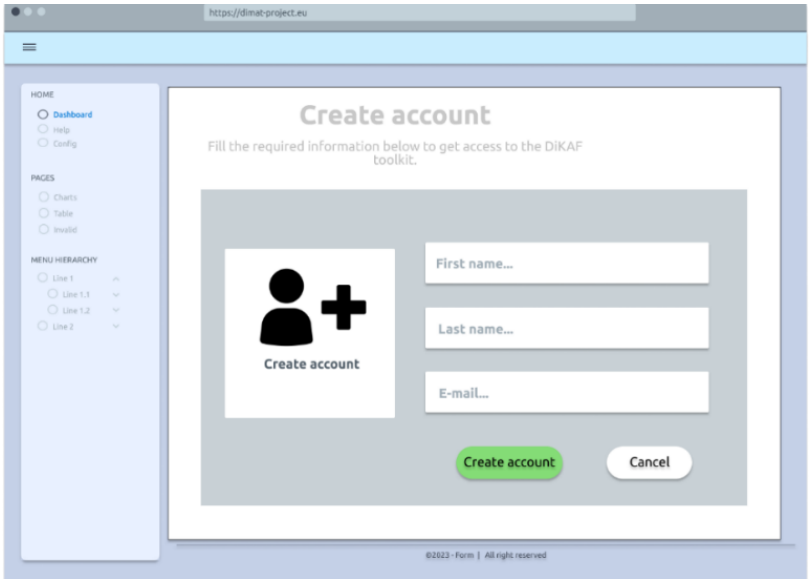
3.3.3.4.2 Activity 4 – Effects

A list with all the registered users in the KAF toolkit will be maintained, while access rights will be managed by the system administrator.

3.3.3.4.3 Activity 4 - Constraints

Access to the available data will be in accordance with their type (e.g., open data, private data).

3.3.3.4.4 Activity 4 – Mockups

Function	Mockup
Provide account information	

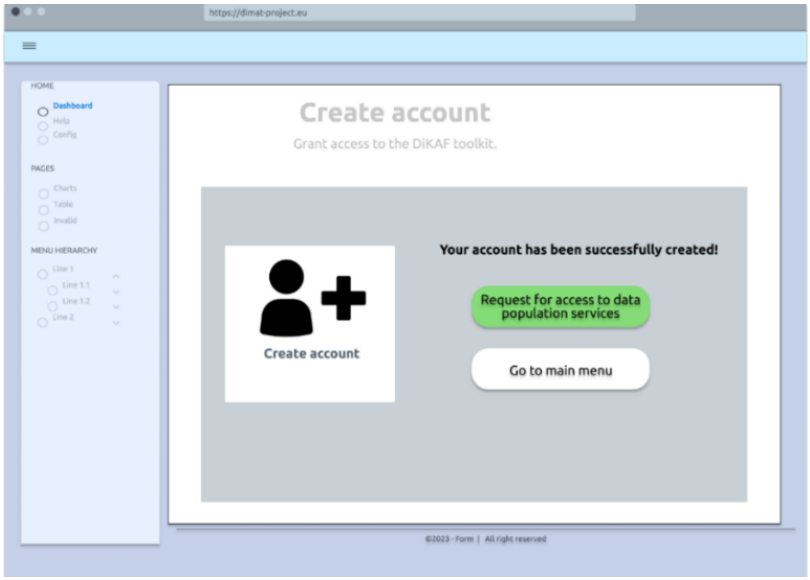
Request administrator privileges	

Table 28. Di^{KAF} Mockup Activity "Create Account"

3.4 DIMAT MATERIALS ENVIRONMENTAL AND COST LIFE CYCLE ASSESSMENT (DI$^{MEC\cdot LCA}$)

The Materials Environmental and Cost Life Cycle Assessment (MEC-LCA) provides a high-level assessment of the environmental and economic impact of the pilot use cases through the graphical representation of results. The DiMAT MEC-LCA tool will provide user-intelligible outputs based on user input, rather than real-time data or complex sets of data. The users, such as product engineers or decision-makers, will be able to review indicators of the environmental and cost LCAs, as well as information on the pilots' KPIs (WP2). Information on core life cycle assessment issues, such as identification of environmental and financial hotspots, evaluation of design options performance, and benchmarking will be visualized through the platform's UI.

3.4.1 Overview

Below is the use case diagram for the MEC-LCA toolkit, presenting the main actors of the toolkit, as per the roles identified in the chapter below and the main use cases or activities.

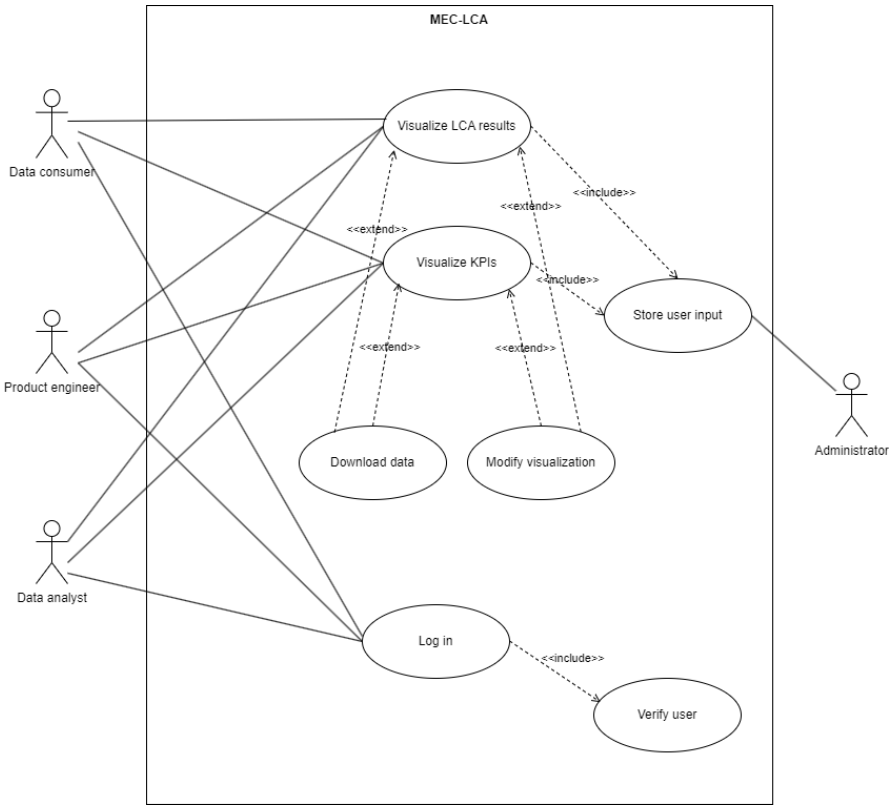


Figure 17. Use Case diagram of Di$^{MEC-LCA}$

3.4.2 Roles

- **Data consumer (Manager and Decision Makers Team):** In charge of defining business strategy and goals for technological solutions.
- **Data analyst (Research, Engineering and Product Design Team):** In charge of data analysis and creating data products to extract value and gather insights from the data about ongoing processes to assist decision-making.
- **Product engineer (Research, Engineering and Product Design Team):** In charge of products development and defining product requirements.
- **Administrator (IT team):** In this context, the solution provider (developers, maintainers), in charge of the code, configuration and making the toolkit available to users.

3.4.3 Activities

3.4.3.1 Activity 1 – "Store user input"

This activity focuses on storing user input – that is all the relevant data which shall be visualized through the toolkit.

3.4.3.1.0 Activity 1 - Triggers

On user input.

3.4.3.1.1 Activity 1 – Workflow

Activity diagram for A1 "Store user input" :

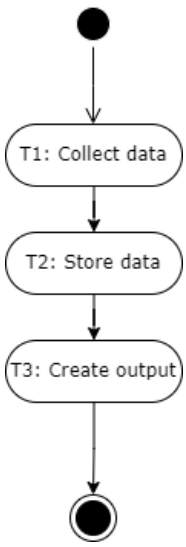


Figure 18: Workflow "Store user input" of Di$^{MEC-LCA}$

Tasks for A1 "Store user input":

- Task T1 "Collect data":
  - Data is collected from the user/s (pilots).
- Task T2 "Store data":
  - Data is stored in a database for further use.

- Task T3 “Create output”:
  - Output is created by the system.

3.4.3.1.2 Activity 1 - Effects

User input is stored in database and output is generated in a user-intelligible format.

3.4.3.1.3 Activity 1 - Constraints

The initiation of this process is contingent upon receiving user input.

3.4.3.1.4 Activity 1 - Mockups

As indicated in the Implementation Viewpoint (T3.5), Grafana will be used and customized as needed to fulfill all the toolkit’s use cases. The screenshots included in the mockup subchapters have been captured during an initial analysis of the system. The final view of the dashboards/UI is expected to differ from what is shown here.

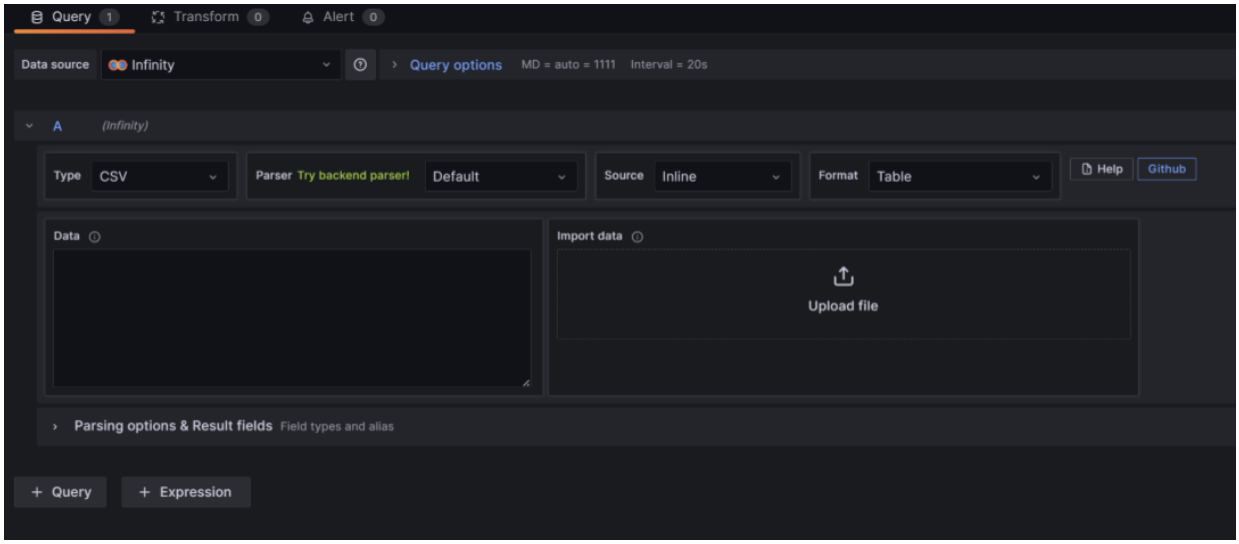


Figure 19: Mockup "Store user input" of Di$^{MEC\text{-}LCA}$

3.4.3.2 Activity 2 - Visualize LCA results

This activity aims to provide informative visualizations related to core LCA issues.

3.4.3.2.0 Activity 2 – Triggers

Upon user contact with the platform's user interface.

3.4.3.2.1 Activity 2 – Workflow

Activity diagram for A2 "Visualize LCA results":

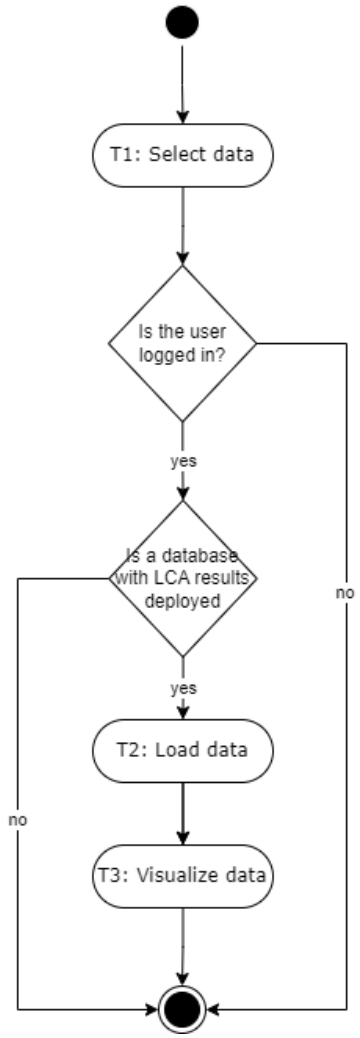


Figure 20: Workflow "Visualize LCA results" of Di$^{MEC-LCA}$

Tasks for A2 "Visualize LCA results":

- Task T1 "Select data":
  - The user selects the appropriate result to be visualized.

- Task T2 “Load data”:
  - Data is loaded from a database.
- Task T3 “Visualize data”:
  - Data is visualized.

3.4.3.2.2 Activity 2 – Effects

A visual representation of LCA results is generated.

3.4.3.2.3 Activity 2 – Constraints

A database concerning core LCA results should be deployed, and the user should have already completed the login process.

3.4.3.2.4 Activity 2 - Mockups

For the visualisations below, test LCA data was used. More visualisation options are available through Grafana.

Figure 21: Mockup "Visualize LCA results" of Di[MEC-LCA]

3.4.3.3 Activity 3 - Visualize KPIs

This activity aims to present relevant KPIs, initially collected from the pilots via T2.3.

3.4.3.3.0 Activity 3 - Triggers

Upon user contact with the platform's user interface.

3.4.3.3.1 Activity 3 - Workflow

The activity diagram and the tasks for diagram for A3 "Visualize KPIs" is the same as the previous one for "Visualize LCA results" as the activity is identical except for the type of data.

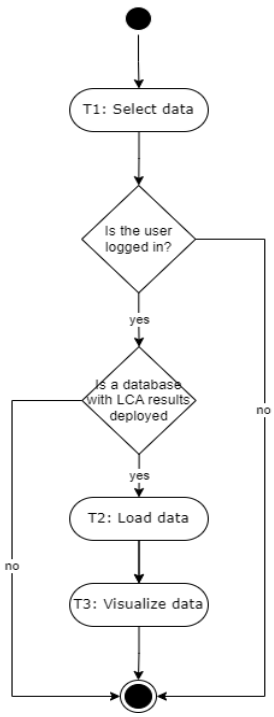


Figure 22: Workflow "Visualize KPIs" of Di$^{MEC-LCA}$

Tasks for A3 "Visualize KPIs":

- Task T1 "Select data":
- The user selects the appropriate result to be visualized.
- Task T2 "Load data":

    - Data is loaded from a database.
- Task T3 “Visualize data”:
    - Data is visualized.

3.4.3.3.2 Activity 3 - Effects

A representation of KPIs is generated.

3.4.3.3.3 Activity 3 - Constraints

A database concerning KPIs should be deployed, and the user should have already completed the login process.

3.4.3.3.4 Activity 3 - Mockups

The screenshot below shows some possible visualisations which can be used to display KPIs (among many more available). The screenshot was captured from Grafana’s sandbox environment [4].

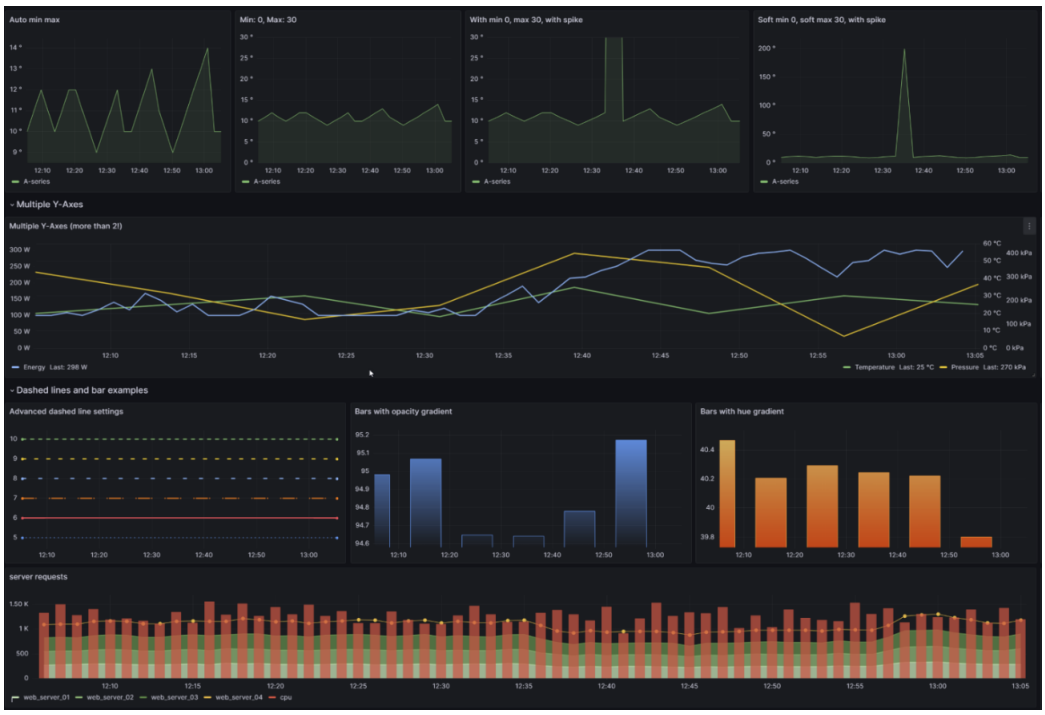


Figure 23: Mockup “Visualize KPIs” of Di[MEC-LCA]

3.4.3.4 Activity 4 - Download data

This activity aims to download data representations.

3.4.3.4.0 Activity 4 - Triggers

Upon user contact with the platform's user interface.

3.4.3.4.1 Activity 4 - Workflow

Activity diagram for A4 "Download data":

Tasks for A4 "Download data":

- Task T1 "Select data":
- The user selects the appropriate result to be downloaded.
- Task T2 "Download data":
- Data is downloaded.

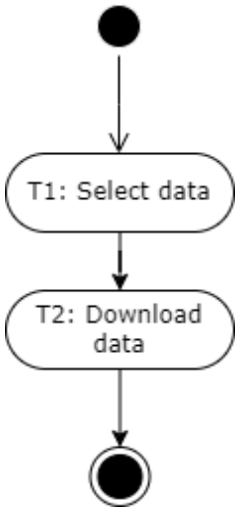


Figure 24: Workflow "Download data" of Di$^{MEC-LCA}$

3.4.3.4.2 Activity 4 - Effects

Data is downloaded.

3.4.3.4.3 Activity 4 - Constraints

Data should have already been presented in the UI.

3.4.3.4.4 Activity 4 - Mockups

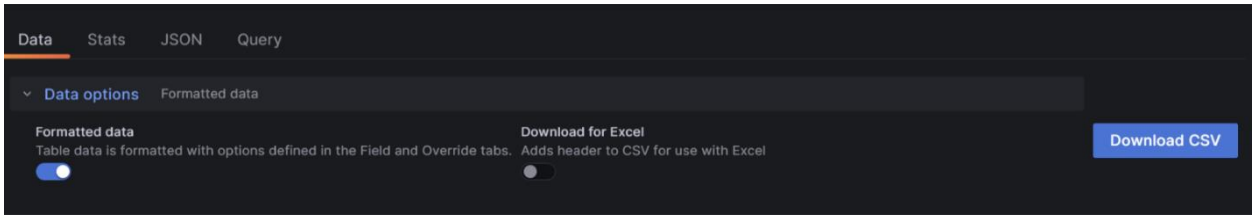


Figure 25: Mockup "Download data" of Di$^{MEC-LCA}$

3.4.3.5 Activity 5 - Log in

This activity aims to establish a streamlined process for users to log-in and access the system. It involves the verification of the user's identity and authorization to access specific data.

3.4.3.5.0 Activity 5 - Triggers

Upon user contact with the platform's user interface.

3.4.3.5.1 Activity 5 - Workflow

Activity diagram for A5 "Log in":

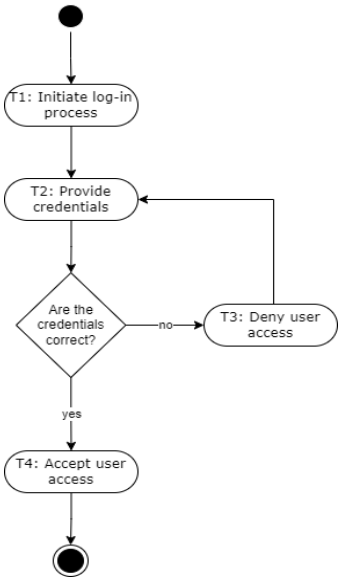


Figure 26: Workflow "Log in" of Di$^{MEC-LCA}$

Tasks for A5 “Log in process”:

- Task T1 “Initiate log-in process”:
- The user initiates a log-in process.
- Task T2 “Provide credentials”:
- The user provides his unique credentials, including a username and password.
- Task T3 “Deny user access”:
- The user is denied access, due to incorrect credentials.
- Task T4 “Accept user access”:
- The user is granted access.

3.4.3.5.2 Activity 5 - Effects

User accesses the system.

3.4.3.5.3 Activity 5 - Constraints

Prior registration in the system is required for the user.

3.4.3.5.4 Activity 5 - Mockups

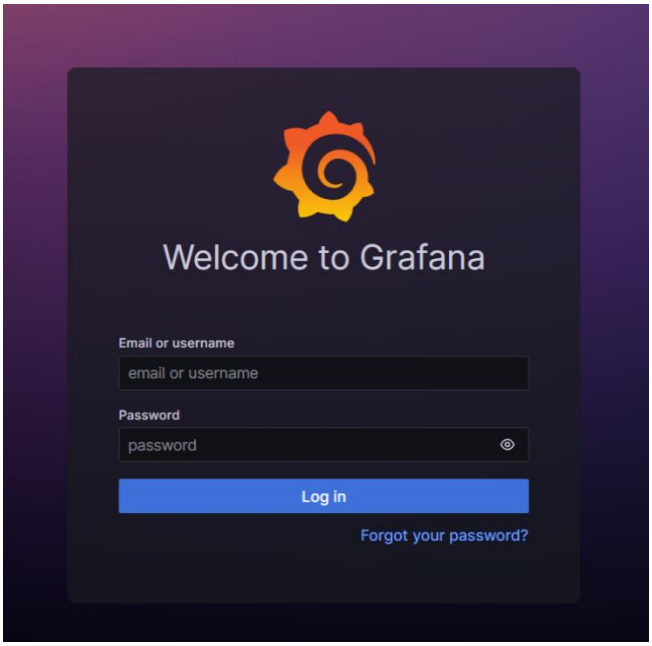


Figure 27: Mockup "Log in" of Di$^{MEC-LCA}$

3.4.3.6 Activity 6 - Modify visualization

This activity aims to enable the user to interact with the charts/graphs presented in the platform, for example through zooming, filtering, choosing a variable or range.

3.4.3.6.0 Activity 6 - Triggers

Upon user contact with the platform's user interface.

3.4.3.6.1 Activity 6 - Workflow

Activity diagram for A6 "Interact with data":

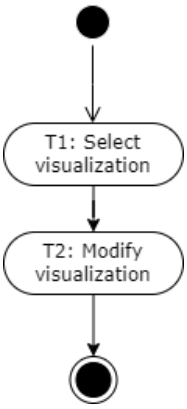


Figure 28: Workflow "Modify visualization" of Di$^{MEC-LCA}$

Tasks for A6 “Interact with data”:

- Task T1 “Select visualization”:
- The user chooses the visualization to be modified.
- Task T2 “Modify visualization”:
- Visualization is modified.

3.4.3.6.2 Activity 6 - Effects

Selected visualization is modified.

3.4.3.6.3 Activity 6 - Constraints

Some data should have already been presented.

3.4.3.6.4 Activity 6 - Mockups

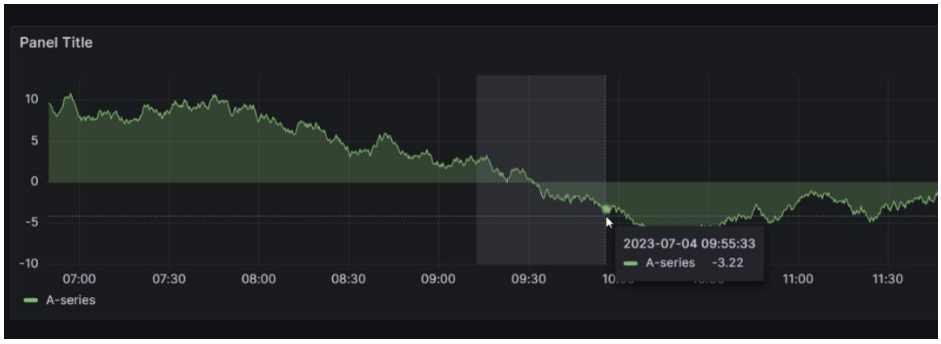


Figure 29: Mockup "Modify visualization" of DiMEC-LCA

3.5 DIMAT MATERIALS DESIGN FRAMEWORK (DI^{MDF})

The Materials Design Framework is a toolkit that leverages the knowledge stored in the CMDB and other toolkits to provide users with suggestions of relevant data, tools, digital twins, and other information based on their query. This will result in a more informed and efficient material design, where available relevant resources such as data, tools, and digital twins are easily findable.

3.5.1 Overview

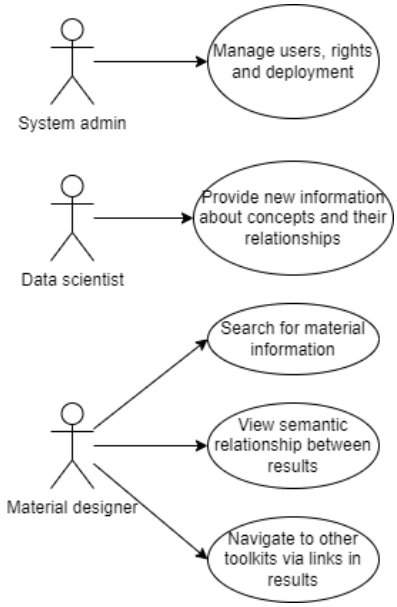


Figure 30: Use Case diagram of Di^{MDF}

3.5.2 Roles

- **System admin:** Has full control over the system and its management.
- **Data scientist:** Generates data and connections that will be added to the MDF.
- **Material designer:** Search for relevant material information and connections on the MDF.

3.5.3 Activities

3.5.3.1 Activity 1 – Search for recommendations

Users want to get information about materials, LCA, simulation and modelling tools and digital twins based on some input information.

3.5.3.1.0 Activity 1 - Triggers

A user knows some information about a material, the manufacturing process and/or the component usage and is looking for further information.

3.5.3.1.1 Activity 1 - Workflow

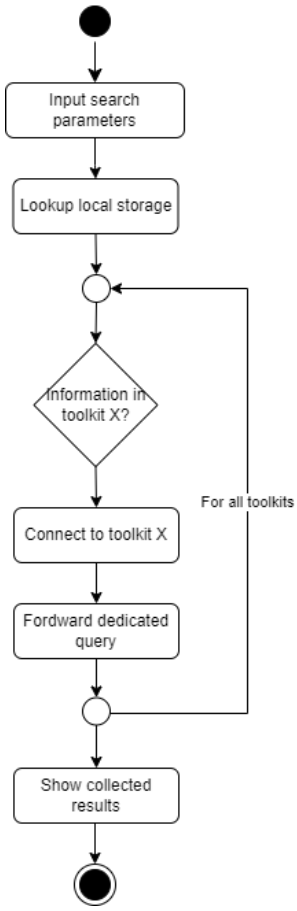


Figure 31: Workflow "Search for recommendations" of Di^{MDF}

The internal data storage is queried for relevant connections based on the user input. These connections are used to forward requests to the corresponding toolkits for additional information. All data from all sources is then gathered and shown to the user in a homogeneous way.

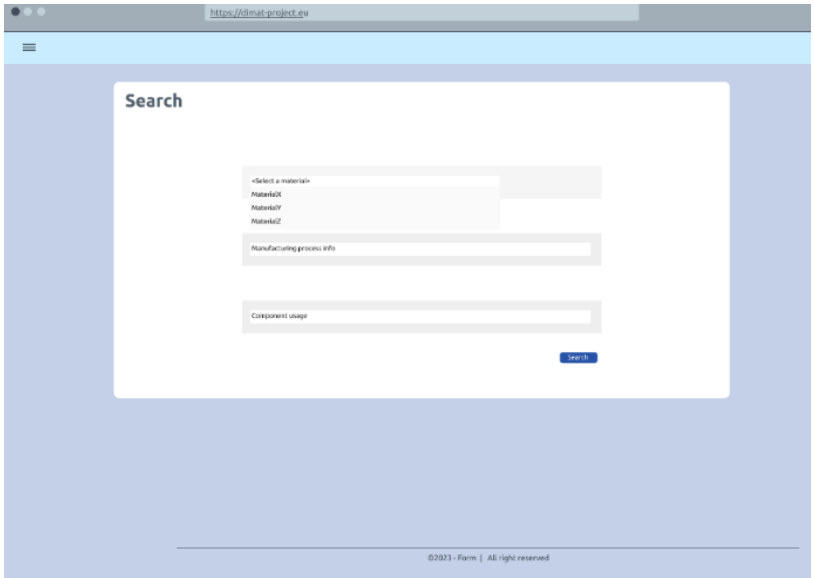
3.5.3.1.2 Activity 1 - Effects

Suggestions for relevant information and connections are shown to the user.

3.5.3.1.3 Activity 1 – Constraints

- All toolkits used to gather information should be up and running.
- Toolkits should have a standard query and data models so information can be asked for.

3.5.3.1.4 Activity 1 - Mockups

Function	Mockup
Search input	

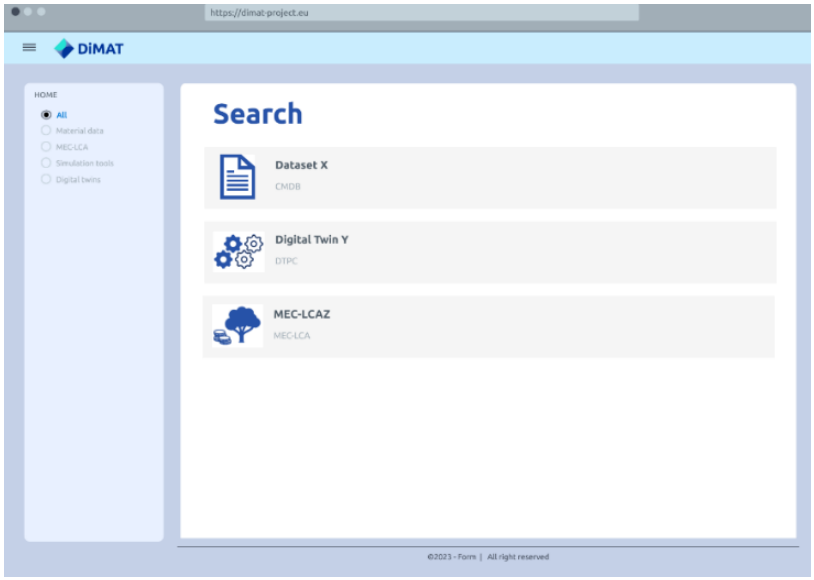
View of the recommendation search results	

Table 29. Di^{MDF} Mockups Activity "Search for recommendations"

3.5.3.2 Activity 2 – Add new information

The toolkit should be dynamic and support integration of new materials and material classes.

3.5.3.2.0 Activity 2 – Triggers

A new simulation tool is compatible for a certain material class, and the toolkit is updated to include this connection.

3.5.3.2.1 Activity 2 – Workflow

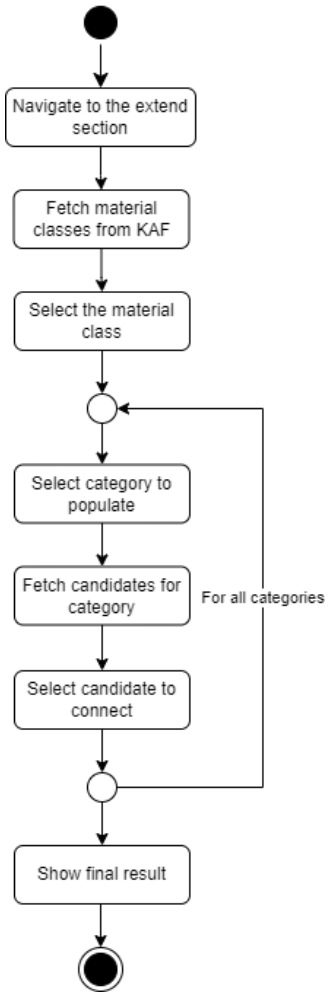


Figure 32: Mockup "Add new information" of Di[MDF]

3.5.3.2.2 Activity 2 – Effects

Both items are now connected and suggested on future searches.

3.5.3.2.3 Activity 2 – Constraints

The user must be logged in and have enough rights to connect items.

3.5.3.2.4 Activity 2 - Mockups

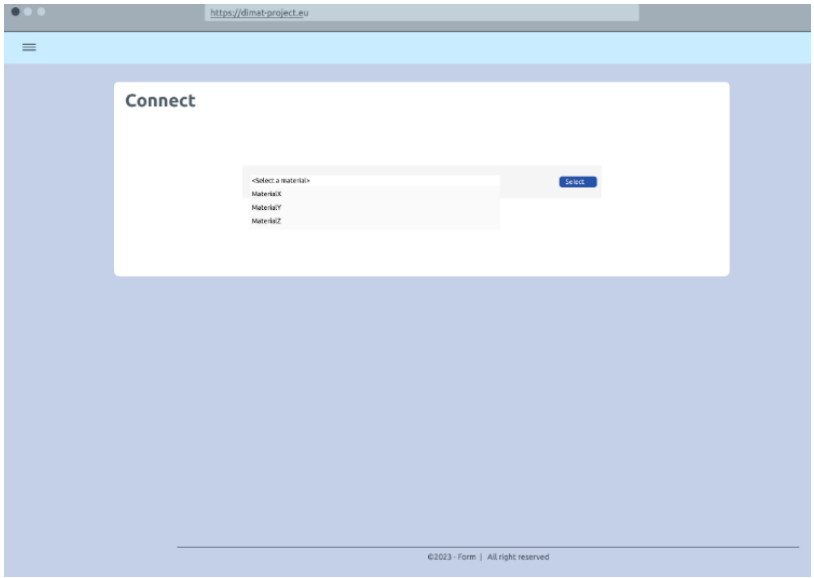
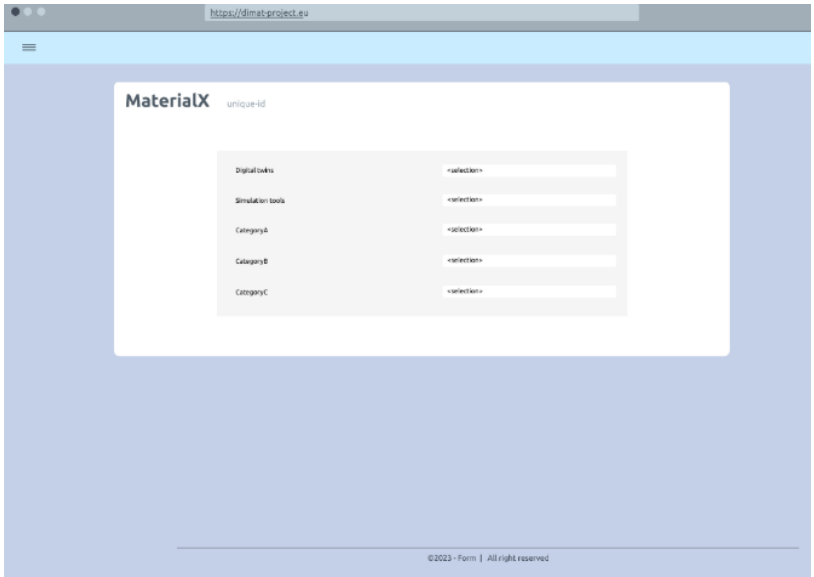
Function	Mockup
Material selection	
Connections to the selected material	

Table 30. Di^{MDF} Mockup Activity "Add new information"

3.6 DIMAT MATERIALS MODELER (DI^{MM})

The DiMAT Materials Modeler (MM) is a toolkit able to predict materials properties and behavior under different conditions during manufacturing processes, based on material structure, composition, etc., while inferring and suggesting optimal configurations. For instance, the DiMAT MM will be able to provide critical insights about which parameters/conditions affect the product quality during manufacturing on a microscopic scale. Moreover, the efficient material behavior prediction may provide control capabilities in microstructure mechanisms. For example, material damage or failure related behaviors can be predicted in time, and thus, permanent problems can be prevented via the application of the optimal configurations proposed by the DiMAT MM. The DiMAT MM solution will exploit robust AI (e.g., explainable AI, ML, DL), adopting causal inference approaches able to explain complex correlation patterns among interrelated phenomena, i.e., latent relationships between material structure and properties. The AI robustness will be ensured by testing the algorithms on different types of datasets, e.g., case study data, real industrial data, data collected from different industrial environments, etc. This solution will predict the material properties based on their structure and composition. Moreover, explainable AI will support converting the objective into understanding the underlying mechanisms and physical laws governing materials, leading to interpretable, trustworthy, and highly accurate estimations. All these actions aim to support increasing the production design rate, while decreasing the production cost.

3.6.1 Overview

Materials Scientists and Manufacturing Engineers use the toolkit to predict material behavior and suggest optimal configurations. Materials Scientists, with their deeper understanding of material structure and composition, also analyze causal inferences. Both roles, along with Production Managers, use the generated reports for decision-making.

3.6.2 Roles

- **Materials Scientist:** This role involves leveraging the toolkit for predicting material behavior based on various factors like material structure and composition. The materials scientist also uses the toolkit to analyze causal inferences and understand the complex relationships among interrelated phenomena.
- **Manufacturing Engineer:** The manufacturing engineer uses the DiMAT MM toolkit to optimize manufacturing processes. This involves using the toolkit to predict material behavior and to apply suggested optimal configurations in the manufacturing processes. The manufacturing engineer can also use the toolkit's report generation function for data-driven decision-making.

- **Production Manager/Data Consumer:** The production manager/data consumer oversees the manufacturing process and makes critical decisions. This role also involves monitoring the toolkit's data handling capabilities and ensuring it's working efficiently with various datasets. They utilize the toolkit's report generation capabilities to gain insights into material behaviors and manufacturing configurations, supporting their decision-making process.
- **System Administrator:** The system administrator is responsible for managing the toolkit's security measures. This includes defining security policies, as well as assigning roles and credentials to the different parties. They also ensure that the toolkit is well-maintained and up-to-date.

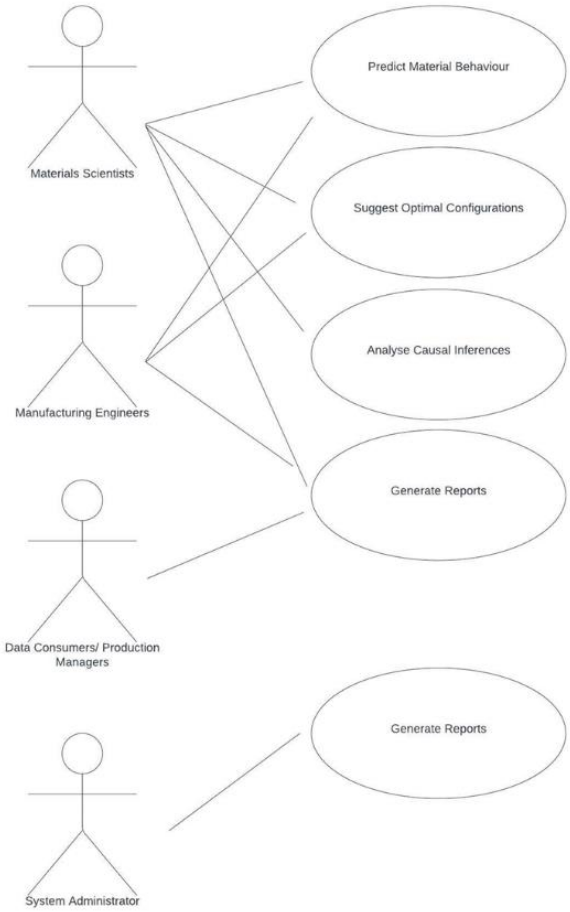


Figure 33: Use Case diagram of Di^{MM}

3.6.3 Activities

3.6.3.1 Activity 1 – “Predict Material Behaviour”

This activity represents the process of predicting material behavior using the DiMAT MM toolkit. The activity involves inputting relevant data about the material and processing conditions into the toolkit, which then uses its built-in algorithms to predict how the material will behave under different conditions.

3.6.3.1.0 Activity 1 - Triggers

The activity is triggered when the Materials Scientist or Manufacturing Engineer selects the option "Predict Material Behavior" in the toolkit's interface.

3.6.3.1.1 Activity 1 - Workflow

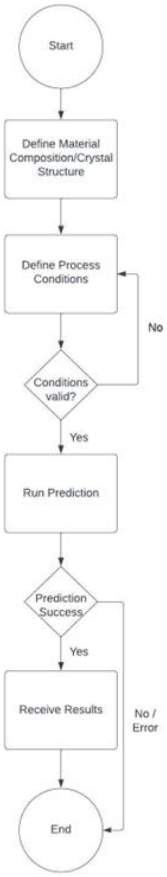


Figure 34: Workflow "Predict Material Behaviour" of Di^{MM}

3.6.3.1.2 Activity 1 - Effects

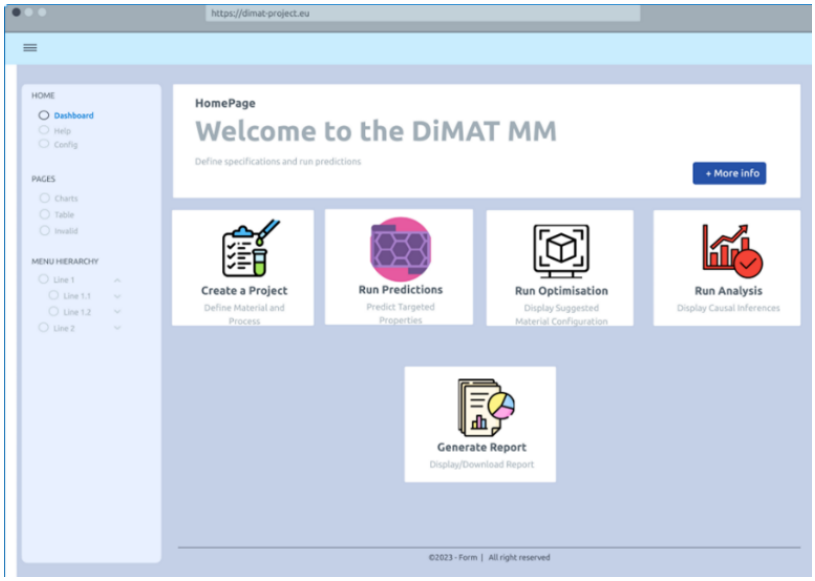
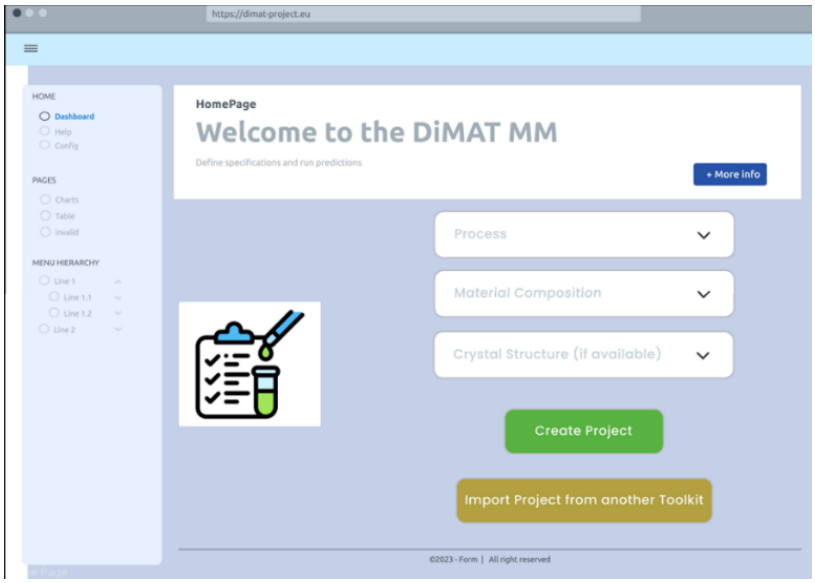
- The toolkit produces a prediction of material behavior, which is recorded in the system.

3.6.3.1.3 Activity 1 - Constraints

- The data entered about the material and conditions must be within the parameters that the toolkit can process.

- All data entered must comply with any existing data privacy and security regulations.

3.6.3.1.4 Activity 1 - Mockups

Function	Mockup
View Home Page	
Project Creation	

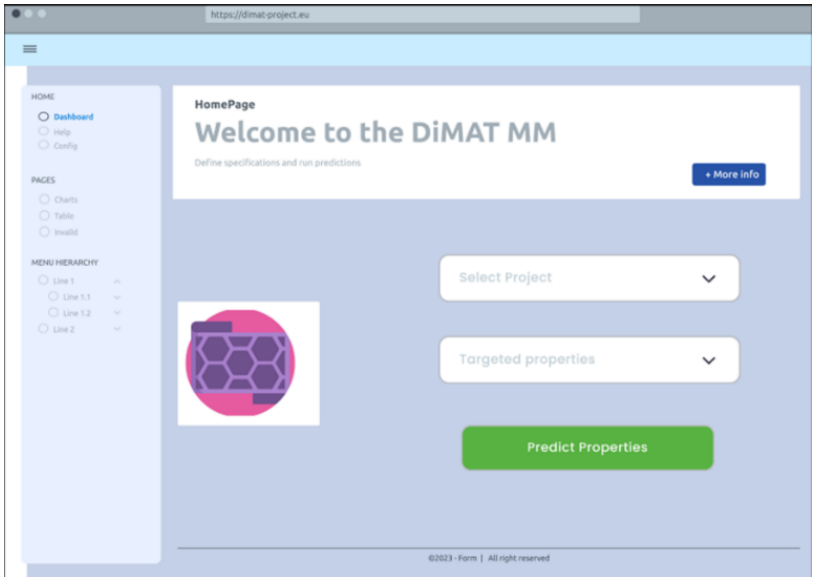
Predict Properties	

Table 31. Di^{MM} Mockup Activity "Predict Material Behaviour"

3.6.3.2 Activity 2 – "Suggest Optimal Configurations"

The toolkit provides material composition and crystal structure configuration suggestions to improve the material's performance under specified conditions. This helps to optimize the use of materials, potentially leading to improved product quality, sustainability, and cost-effectiveness.

3.6.3.2.0 Activity 2 – Triggers

This activity is typically triggered by the Materials Scientist or Manufacturing Engineer after analyzing the predicted material behavior and deciding to explore optimal material configurations.

3.6.3.2.1 Activity 2 – Workflow

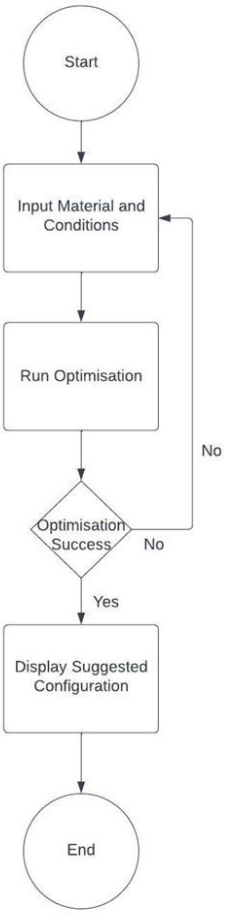


Figure 35: Workflow "Suggest Optimal Configurations" of Di^{MM}

3.6.3.2.2 Activity 2 – Effects

- The toolkit displays the optimal material composition and crystal structure configurations for the user to consider.
- The system logs the optimization run and its results.

3.6.3.2.3 Activity 2 – Constraints

- The optimization algorithm and material database must be up-to-date and accurate.
- The user inputs must be within valid and acceptable ranges.

3.6.3.2.4 Activity 2 - Mockups

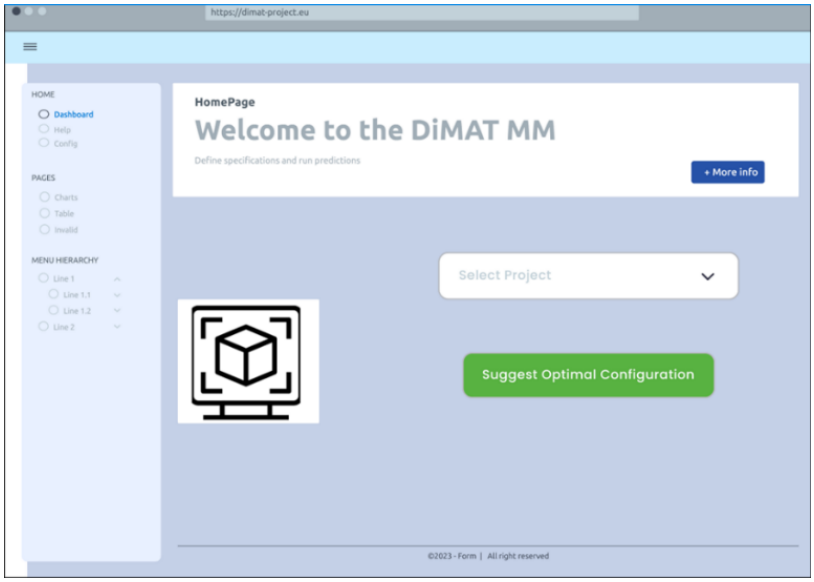
Function	Mockup
Suggest Optimal Configuration	

Table 32. Di^{MM} Mockup Activity "Suggest Optimal Configurations"

3.6.3.3 Activity 3 – "Analyse Causal Inferences"

The toolkit utilizes robust AI techniques to analyze and understand complex correlations and causal relationships between different factors, such as material structure, composition, and properties, and their influence on material behavior.

3.6.3.3.0 Activity 3 – Triggers

This activity is typically triggered by the Materials Scientist or Manufacturing Engineer when there's a need to understand the underlying mechanisms and physical laws governing materials, leading to interpretable, trustworthy, and highly accurate estimations.

3.6.3.3.1 Activity 3 – Workflow

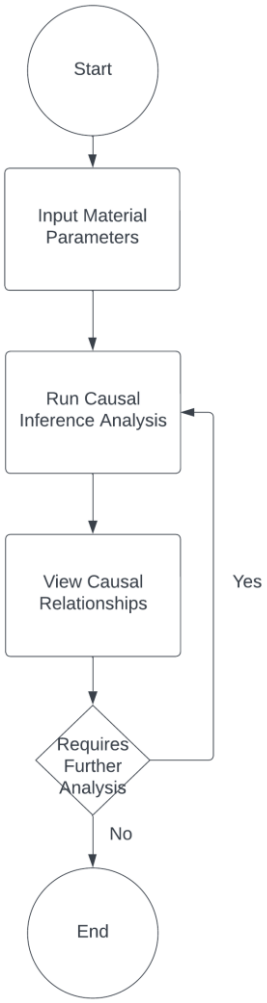


Figure 36. Workflow "Analyse Causal Inferences" of Di^{MM}

3.6.3.3.2 Activity 3 – Effects

- The toolkit displays the causal inferences for the user to understand the underlying mechanisms and physical laws governing the materials.
- The system logs the causal analysis run and its results.

3.6.3.3.3 Activity 3 – Constraints

- The causal analysis algorithm and material database must be up-to-date and accurate.
- The user input must be within valid and acceptable ranges.

3.6.3.3.4 Activity 3 – Mockups

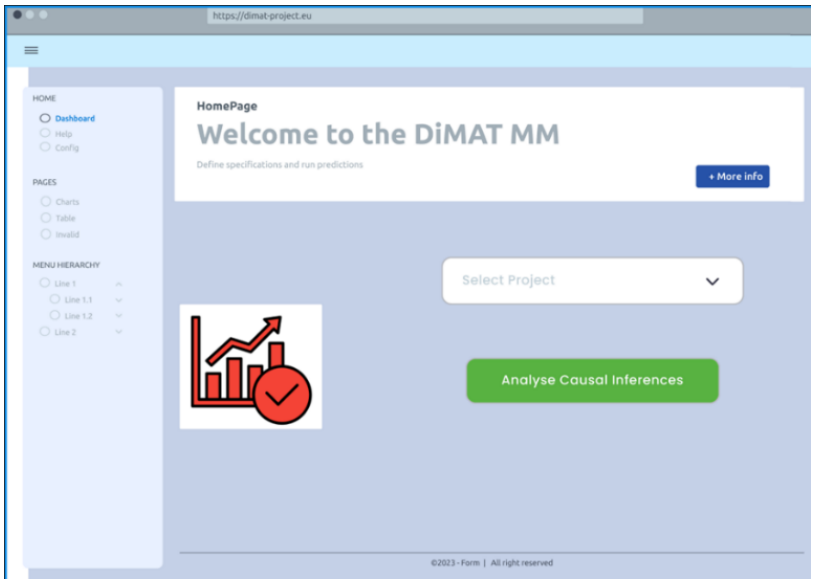
Function	Mockup
Analyse Causal Inferences	

Table 33. Di^{MM} Mockup Activity "Analyse Causal Inferences"

3.6.3.4 Activity 4 – "Generate Reports"

The toolkit allows users to generate comprehensive reports that summarize and present the outcomes of the material behavior prediction, optimal configuration suggestions, and causal inference analyses. These reports aid in the decision-making process for materials design and manufacturing.

3.6.3.4.0 Activity 4 – Triggers

This activity is typically triggered by the Materials Scientist, Manufacturing Engineer, or Data Consumers/Production Managers when they wish to consolidate, review, or use the results of the toolkit's analyses for their respective purposes.

3.6.3.4.1 Activity 4 – Workflow

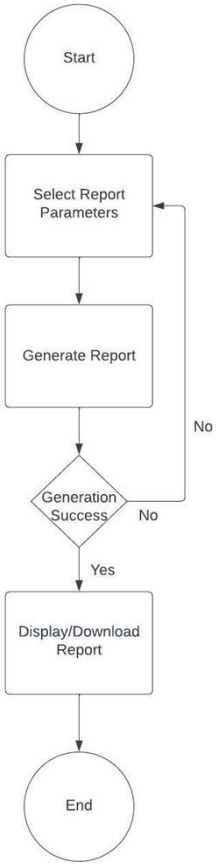


Figure 37: Workflow "Generate Reports" of Di[MM]

3.6.3.4.2 Activity 4 – Effects

- The toolkit generates a comprehensive report summarizing the outcomes of the conducted analyses.
- The user can view and/or download the report for further review and decision-making.

3.6.3.4.3 Activity 4 – Constraints

- The data used for generating the report must be already existing, accurate and up-to-date.
- The report generation feature must be operational and free of bugs.

3.6.3.4.4 Activity 4 - Mockups

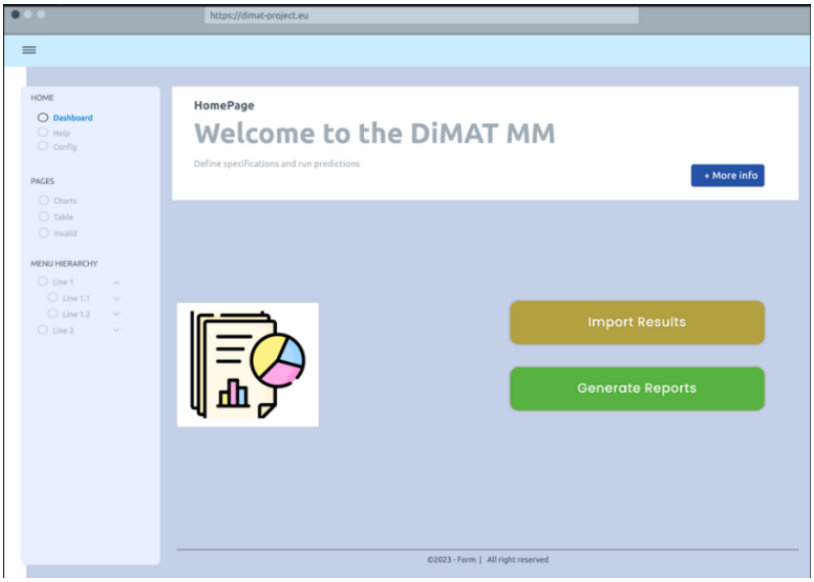
Function	Mockup
Generate Reports	

Table 34. Di^{MM} Mockup Activity "Generate Reports"

3.7 DIMAT MATERIALS DESIGNER (DI^{MD})

The DiMAT Material Designer is a tool dedicated to the calculation of the mechanical and thermal properties of non-homogenous materials, with a specific emphasis on composite materials.

The tool encompasses a state-of-the-art Finite Element code able to execute virtual experiments to characterize the new materials starting from the properties of the base materials and their microstructural arrangement.

3.7.1 Overview

Simulation Engineers, in charge of the design of a specific component, use the toolkit to accurately predict the material properties of complex materials that they want to employ for the new components that are being developed.

3.7.2 Roles

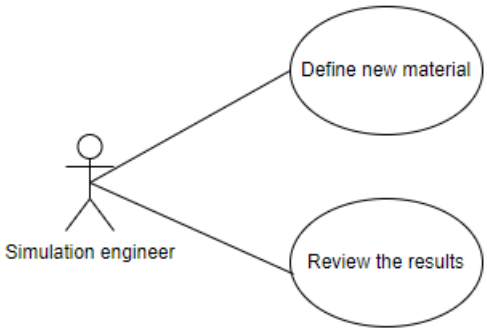


Figure 38: Use Case of Di^{MD}

- **Simulation engineer:** The simulation engineer uses the DiMAT MD toolkit to calculate the mechanical and thermal properties of a non-homogenous and complex materials defining as inputs the base components with all their properties and the microstructure of the new material. The toolkit provides the requested information and offer the possibility to compare the results to other materials previously analyzed.

3.7.3 Activities

3.7.3.1 Activity 1 – "Define new material"

This activity is the process of defining the characteristics of the new material to be analyzed. All the relevant information about the base material and the geometric description of the microstructure are inserted by the user and flagged to be sent to the calculation code for the solution.

Once a new configuration is defined to be analyzed, data are stored in an internal database and the execution of a new virtual experiment is requested to the calculation code.

3.7.3.1.0 Activity 1 - Triggers

The activity is triggered when the Simulation Engineer selects the option "Define new material" in the interface of the toolkit.

3.7.3.1.1 Activity 1 - Workflow

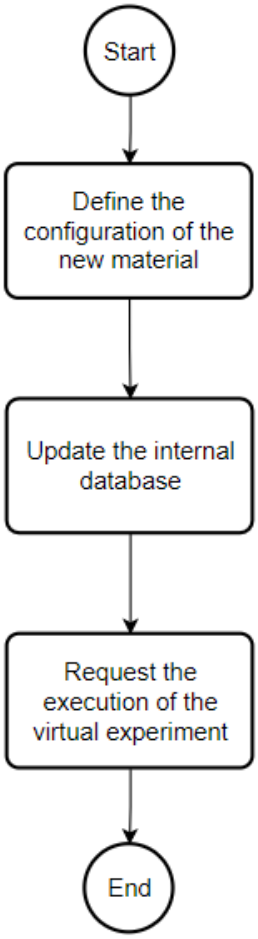


Figure 39: Workflow "Define new material" of Di^{MD}

3.7.3.1.2 Activity 1 - Effects

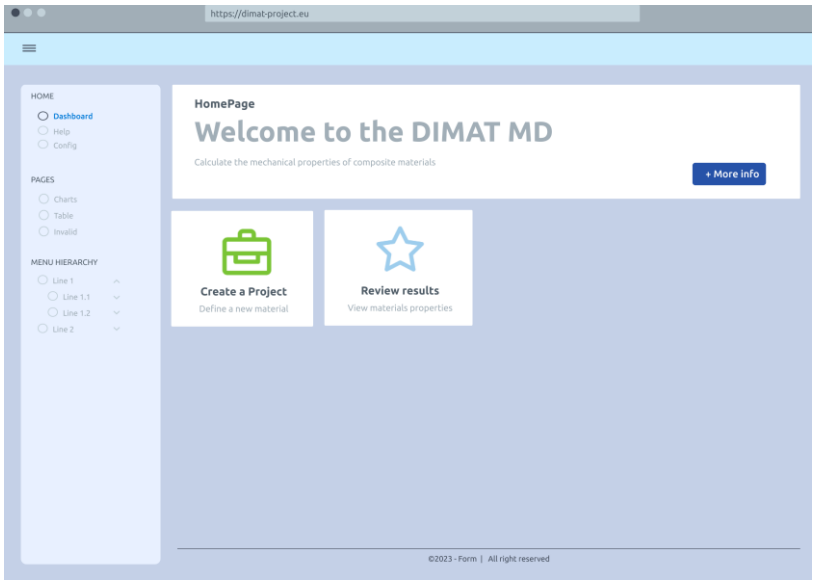
- The material definition is stored in the internal database.

- A new virtual experiment is required for the calculation code.

3.7.3.1.3 Activity 1 - Constraints

- Inputs should be compatible with the calculation code capabilities.

3.7.3.1.4 Activity 1 - Mockups

Function	Mockup
Home page	

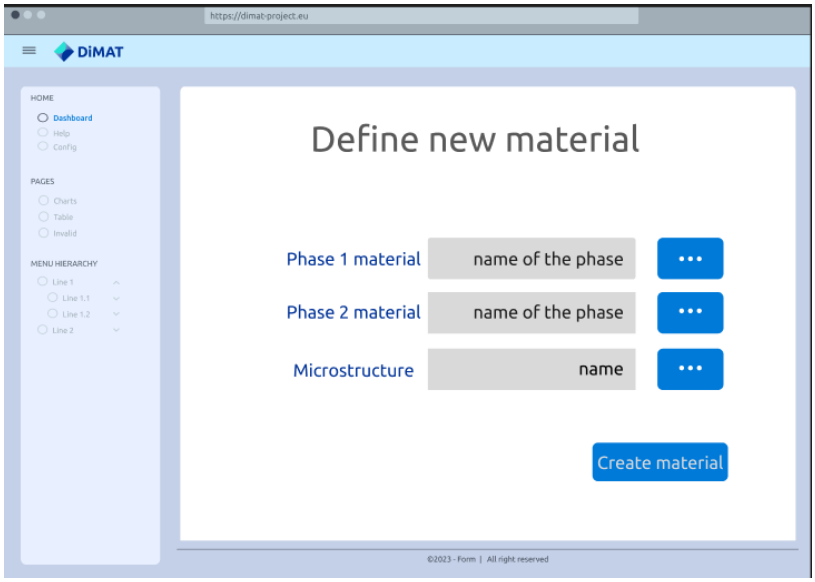
Material definition	

Table 35. Di^{MD} Mockup Activity "Define new material"

3.7.3.2 Activity 2 – "Review the results"

This activity is the process of inspecting the results of a calculation.

The user asks for the results of a submitted calculation via the internal database. If the simulation has been completed, data are transferred from the calculation code to the toolkit and the user can inspect them and confront them with those of a precedent configuration.

3.7.3.2.0 Activity 2 – Triggers

The activity is triggered when the user selects one of the submitted calculations in the internal database and asks to retrieve the results.

3.7.3.2.1 Activity 2 – Workflow

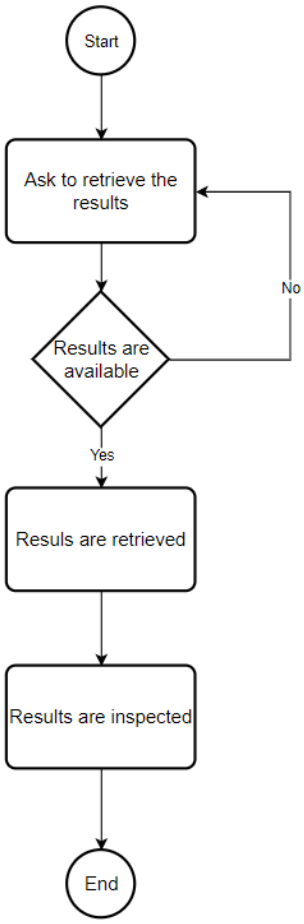


Figure 40: Workflow "Review the results" of Di[MD]

3.7.3.2.2 Activity 2 – Effects

- The internal database is updated with the results of the completed virtual experiment.

3.7.3.2.3 Activity 2 – Constraints

- The simulation must be completed to retrieve the data. If it is not the case, results are not transferred and not presented to the user who must wait until the completion of the calculation.

3.7.3.2.4 Activity 2 - Mockups

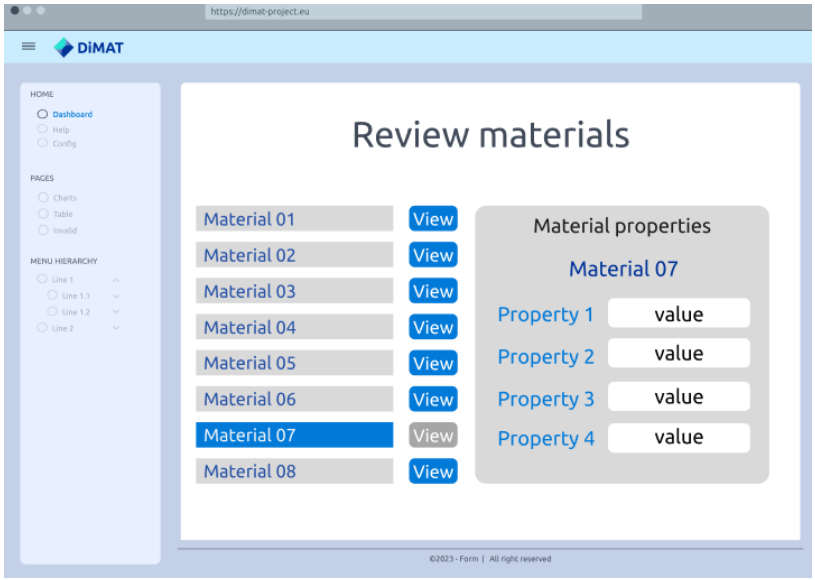
Function	Mockup
Results review	

Table 36. Di^{MD} Mockup Activity "Review the results"

3.8 DIMAT MATERIALS MECHANICAL PROPERTIES SIMULATOR (DI^{MMS})

The objective of the DiMAT MMS toolkit is to accurately predict materials mechanical properties. The input of the toolkit will be the material component fractions, compounds, microstructure parameters, as porosity or grain size. The toolkit generates outputs that include elastic properties, strength, and rheological properties of the material.

3.8.1 Overview

Materials Scientists and Manufacturing Engineers use the toolkit to define an optimization constraint or objective function based on materials mechanical properties. The toolkit will be used along or integrated in a more complex optimization problem for decision-making.

3.8.2 Roles

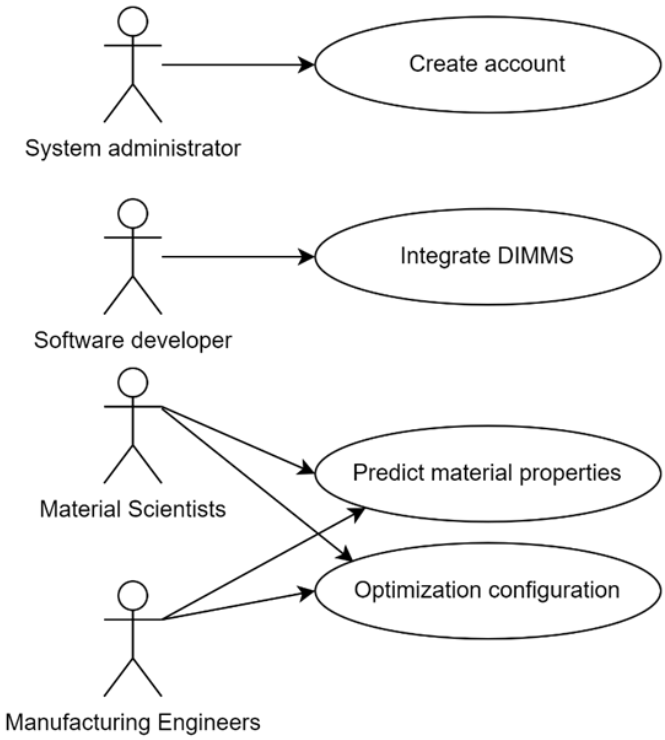


Figure 41: Use Case of Di^{MMS}

- **Materials Scientist:** This role entails utilizing the toolkit to predict material mechanical properties as a function of the composition, fractions, and microstructure parameters. The toolkit will be also used to define the optimization problem, formulating constraints and objective functions.
- **Manufacturing Engineer:** The manufacturing engineer uses the DiMMS toolkit for optimization purposes. The toolkit can be used independently or integrated into more sophisticated software to suggest optimal material compositions or microstructures. Furthermore, the manufacturing engineer leverages the toolkit for more efficient decision-making.
- **Software developer:** The software developer will integrate the DIMMS toolkit into a, optimization software, ensuring seamless integration and compatibility between the components.

- **System Administrator:** The administrator will be responsible for supervising the operation of the toolkit and will define security policies, adding or removing eligible users.

3.8.3 Activities

3.8.3.1 Activity 1 – "Prediction of Mechanical Properties"

This activity encompasses the process of predicting material mechanical properties with the DIMMS toolkit. The toolkit receives as input the composition, fraction of components, microstructure data, and calculates the elastic properties, strength, or rheological properties.

3.8.3.1.0 Activity 1 - Triggers

The activity is triggered by a Materials Scientific or a Manufacturing Engineer. The user explicitly selects the option "Run Prediction".

3.8.3.1.1 Activity 1 - Workflow

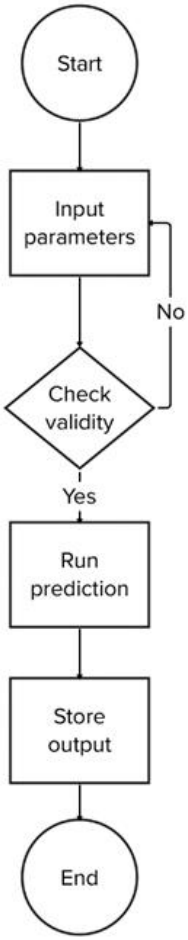


Figure 42: Workflow "Prediction of Mechanical Properties" of Di^{MMS}

3.8.3.1.2 Activity 1 - Effects

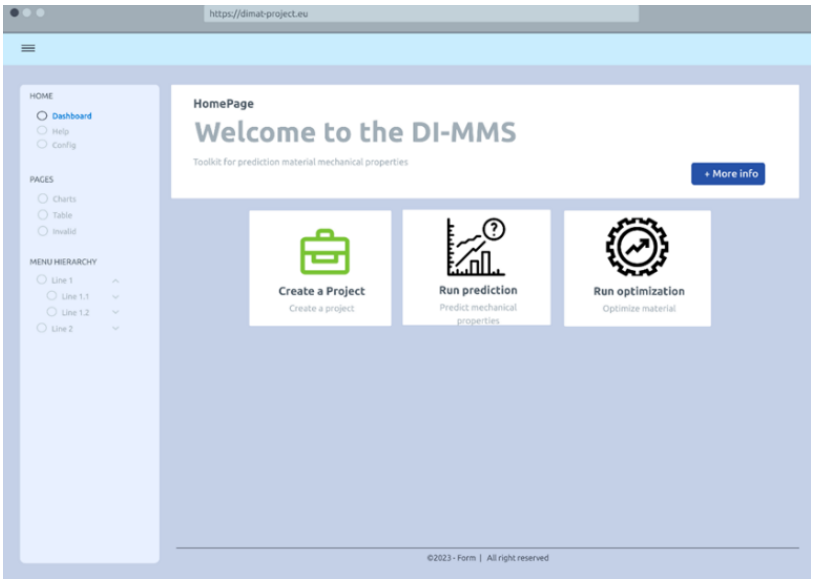
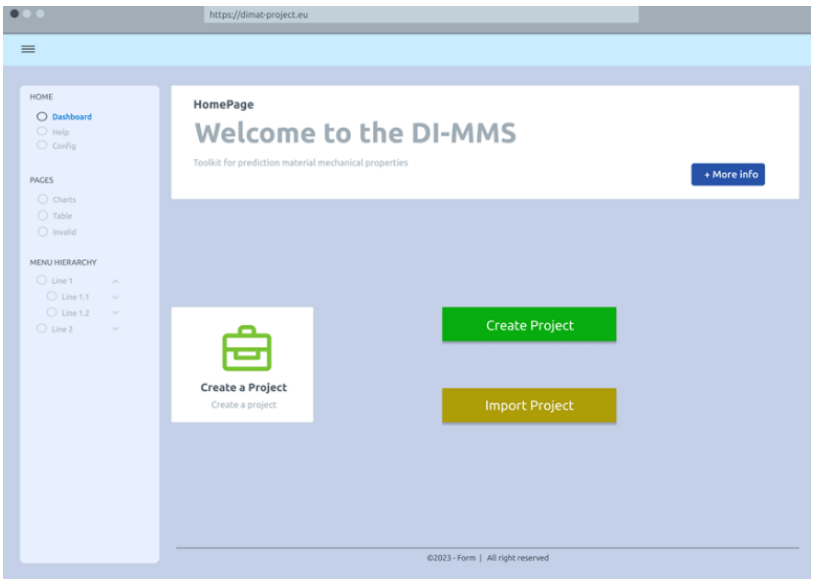
The toolkit produces a prediction of material mechanical property, which is recorded in the system.

3.8.3.1.3 Activity 1 - Constraints

- The user inputs must fall within valid range to obtain a validated prediction.

- The format of the input and the output should be standardized and consistent to facilitate the utilization.

3.8.3.1.4 Activity 1 - Mockups

Function	Mockup
View Home Page	
Project Creation	

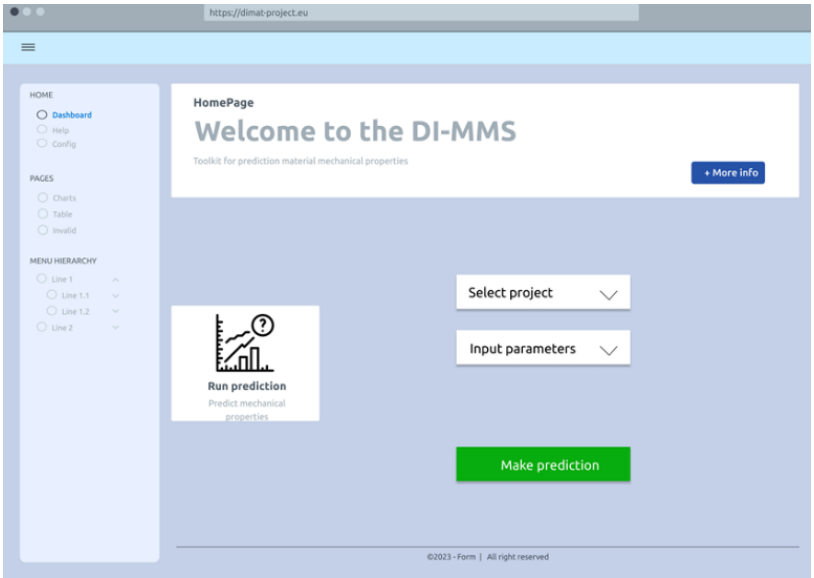
Make prediction	

Table 37. Di^{MMS} Mockup Activity "Prediction of Mechanical Properties"

3.8.3.2 Activity 2 – “Optimization”

The toolkit provides the optimization of material composition, fraction and microstructure parameters based on target mechanical material properties.

3.8.3.2.0 Activity 2 – Triggers

The activity is triggered by a Materials Scientific or a Manufacturing Engineer. The user explicitly selects the option “Run optimization”.

3.8.3.2.1 Activity 2 – Workflow

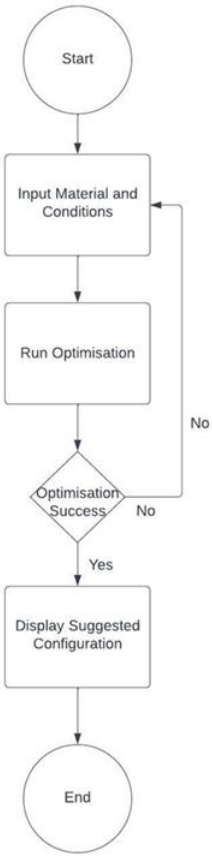


Figure 43: Workflow "Optimization" of Di^{MMS}

3.8.3.2.2 Activity 2 – Effects

- The toolkit displays the optimal material configuration corresponding to the target mechanical property.
- The system runs the optimization save the results.

3.8.3.2.3 Activity 2 – Constraints

- The user inputs must fall within valid range to obtain a validated inverse prediction or optimization.

- The format of the input and the output should be standardized and consistent to facilitate the utilization.

3.8.3.2.4 Activity 2 - Mockups

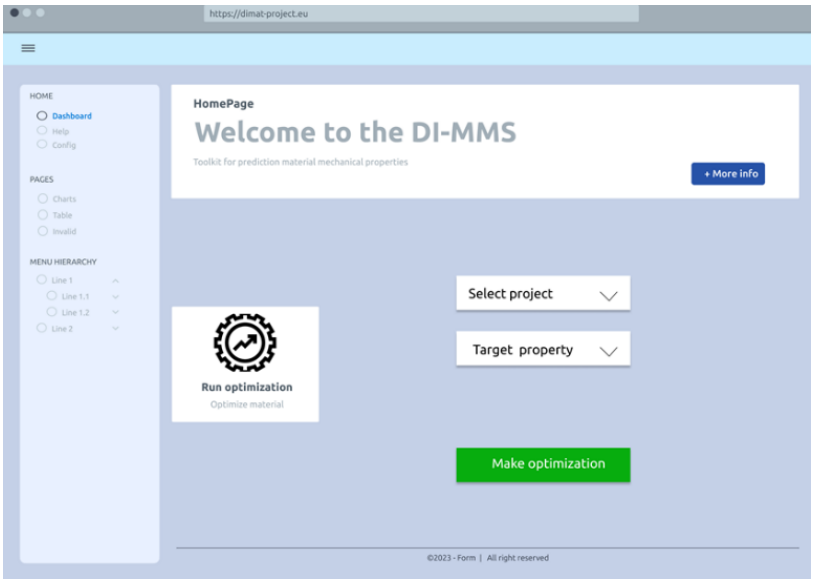
Function	Mockup
Optimization	

Table 38. Di^{MMS} Mockup Activity "Optimization"

3.9 DIMAT MATERIALS PROCESSING SIMULATOR (DI^{MPS})

The DiMAT materials processing simulator (MPS) is a toolkit developed to recreate the conditions of different material transformation processes and the behavior of materials under industrial processing conditions.

The toolkit allows the user to carry out a processing simulation of the most common manufacturing processes used to process each material. First, the development of a geometry that coincides with the shape of the material at the end of the transformation process. Then, the boundary conditions will be added, as well as the main parameters. These conditions are the input corresponding to the parameters and conditions used in processing the materials, e.g., temperature, pressure, material properties, etc. Before performing the simulation, the geometry will be mesh, and if necessary, the mesh will be refined for better results. Finally, results will be able to simulate the principal parameters like pressure to avoid processing problems that may lead to defects in the material.

Also, one of the main reasons for simulating manufacturing processes is to analyze the different materials' behavior without having to go through the production process line to perform such tests. In this way, material transformation processes can be optimized, reducing costs and production times.

3.9.1 Overview

The toolkit could be used by Manufacturing Engineers, Process Scientists, and Process Engineers to predict material behavior under production conditions and optimize materials transformation processes. Also, Materials Scientists could use the toolkit to anticipate possible defects during materials processing and use the output to select the correct material.

3.9.2 Roles

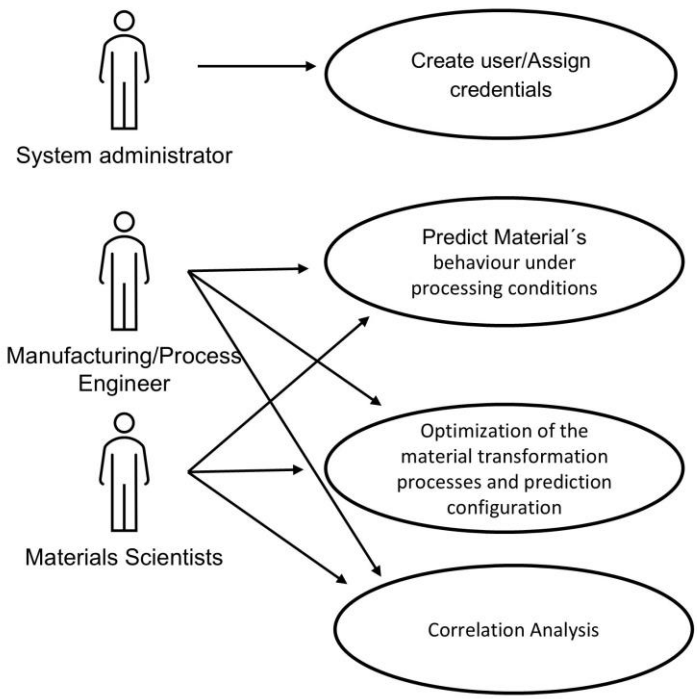


Figure 44: Use Case diagram Di^{MPS}

- System Administrator: The system administrator will manage the toolkit system, create new users and assign credentials and keep up-to-date the toolkit database.

- Manufacturing/Process Engineer: manufacturing engineers can use the toolkit to optimize manufacturing processes, parameters like temperature, velocity, or pressure. The toolkit can predict material behaviour and optimize the parameters to optimize the transformation process.
- Materials Scientists: this type of professional can use the toolkit to perform tests and the behaviour of the materials under processing conditions. These could select the correct material to perform the transformation process after testing the materials by performing simulations only by entering the material properties and processing conditions.

3.9.3 Activities

3.9.3.1 Activity 1- Predict Material Behaviour under processing conditions

The toolkit utilizes combined algorithms from several software to analyse and understand materials' behaviour considering different factors, such as type of material and transformation process, material properties, and processing conditions.

3.9.3.1.0 Activity 1 – Triggers

The activity is triggered when the Materials Scientist or Manufacturing Engineer selects the option "Predict Material Behaviour under processing conditions" in the toolkit's interface.

3.9.3.1.1 Activity 1 – Workflow

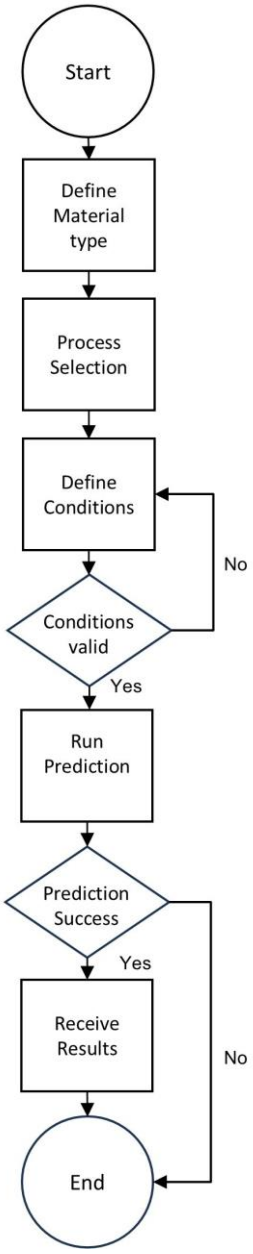


Figure 45: Workflow "Predict Material Behaviour under processing conditions" of Di^{MPS}

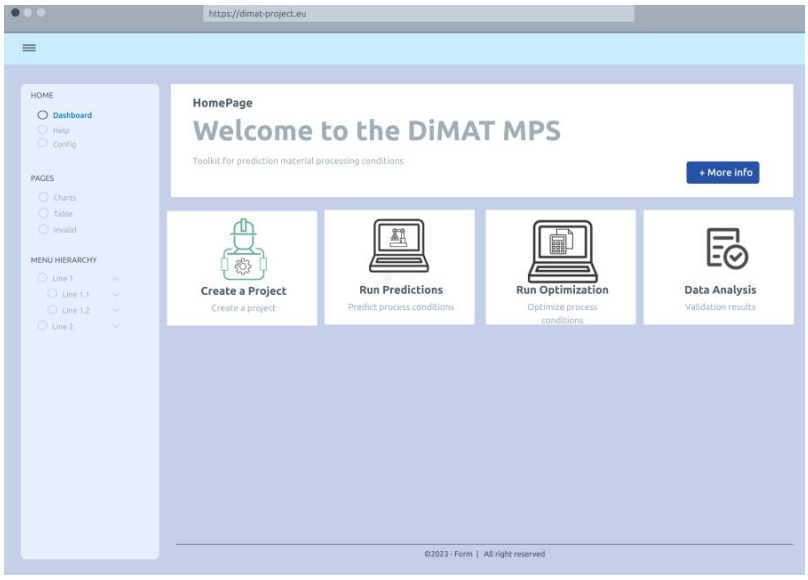
3.9.3.1.2 Activity 1 - Effects

The toolkit produces a prediction of Predict Material Behaviour under processing conditions, which is recorded in the system.

3.9.3.1.3 Activity 1 - Constraints

The data entered about the material and conditions must be within the parameters that the toolkit can process, and within valid and acceptable ranges. All data entered must comply with any existing data privacy and security regulations.

3.9.3.1.4 Activity 1 - Mockups

Function	Mockup
Home Page	

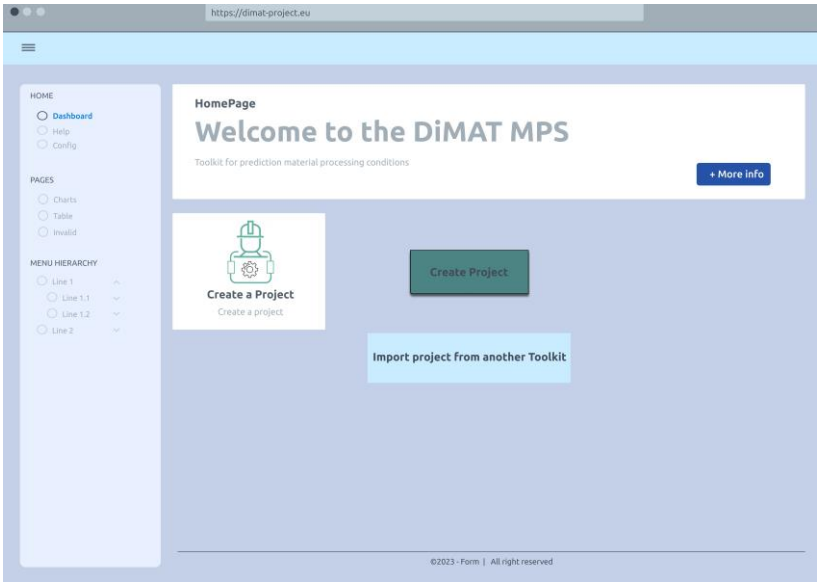
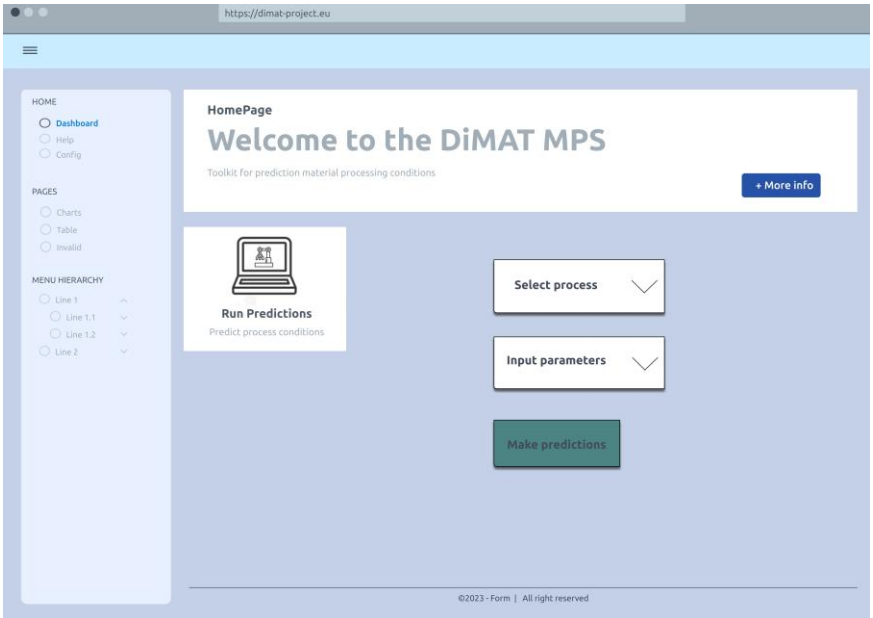
Project Creation	
Create Predicion	

Table 39. Di^{MPS} Mockup Activity "Predict Material Behavior under processing conditions"

3.9.3.2 Activity 2- Optimization of the material transformation processes and prediction configuration

The toolkit composes an optimization path based on the transformation process, material

properties, and processing conditions, targeting the suitable output.

3.9.3.2.0 Activity 2 – Triggers

The activity is triggered when the Materials Scientist or Manufacturing Engineer selects the option " Optimization of the material transformation processes " in the toolkit's interface.

3.9.3.2.1 Activity 2 – Workflow

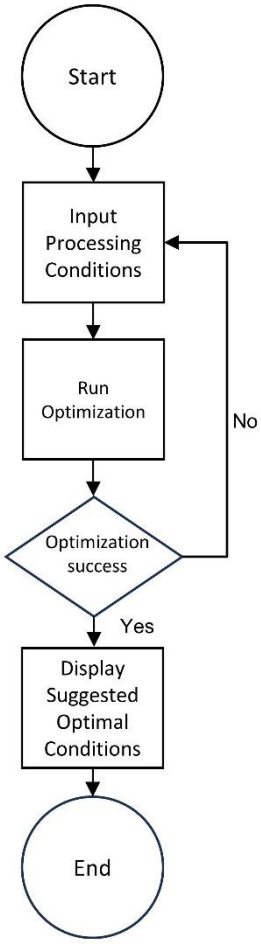


Figure 46: Workflow "Optimization of the material transformation processes and prediction configuration" of Di^{MPS}

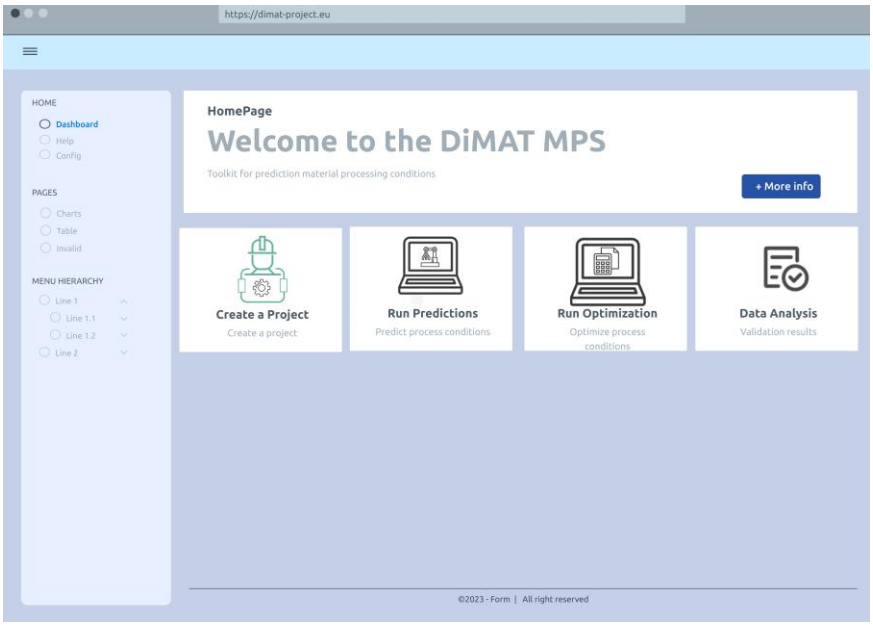
3.9.3.2.2 Activity 2 – Effects

The toolkit optimisation path provides a processing configuration corresponding to refining the parameters for the transformation processes.

3.9.3.2.3 Activity 2 – Constraints

The input and output formats and ranges should be standardized and consistent with the materials' characteristics. All data entered must comply with any existing data privacy and security regulations.

3.9.3.2.4 Activity 2 - Mockups

Function	Mockup
Home Page	

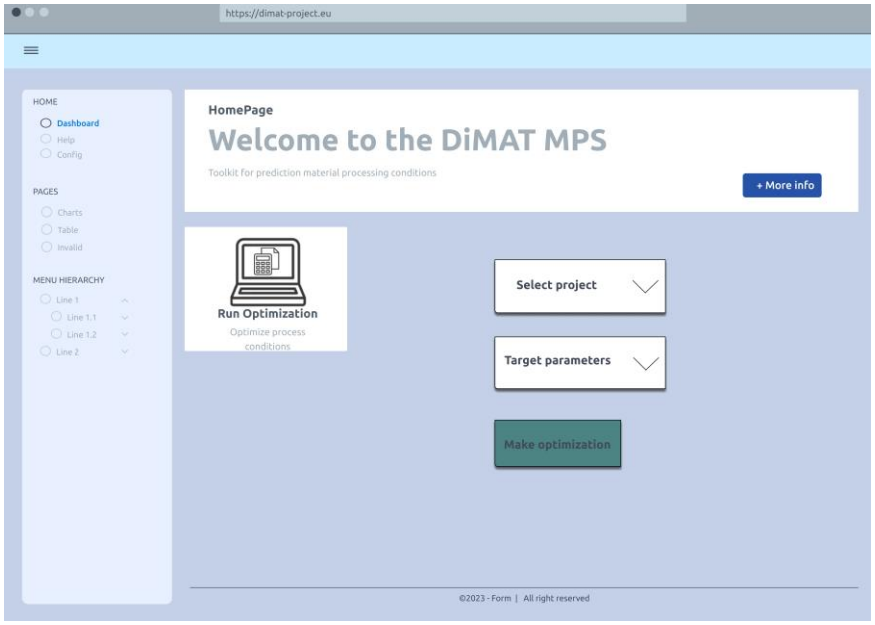
Optimization	

Table 40. Di^{MPS} Mockup Activity "Optimization of the material transformation processes and prediction configuration"

3.9.3.3 Activity 3 - Correlation Analysis and results validation with database info

The toolkit utilizes database info allowing a comparison between previous results and new predictions to analyse and validate the material's behaviour under processing conditions.

3.9.3.3.0 Activity 3 – Triggers

The activity is triggered when the Materials Scientist or Manufacturing Engineer selects the option "Correlation Analysis" in the toolkit's interface.

3.9.3.3.1 Activity 3 – Workflow

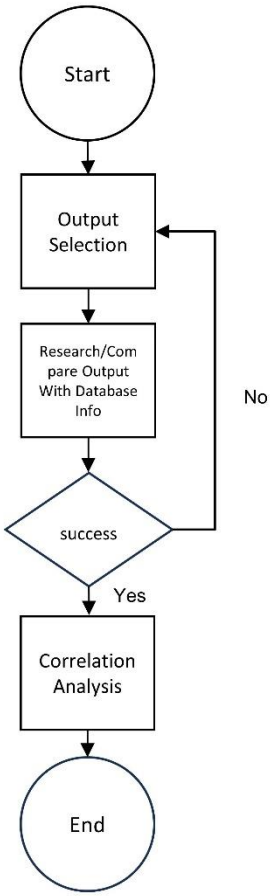


Figure 47: Workflow "Correlation Analysis and results validation with database info" of Di^{MPS}

3.9.3.3.2 Activity 3 – Effects

The toolkit produces a correlation and comparison analysis between previous results (uploaded in the database) and new predictions, which are recorded in the system.

3.9.3.3.3 Activity 3 – Constraints

The input and output formats and ranges should be standardized and consistent with the

materials' characteristics. All data entered must comply with any existing data privacy and security regulations.

3.9.3.3.4 Activity 3 - Mockups

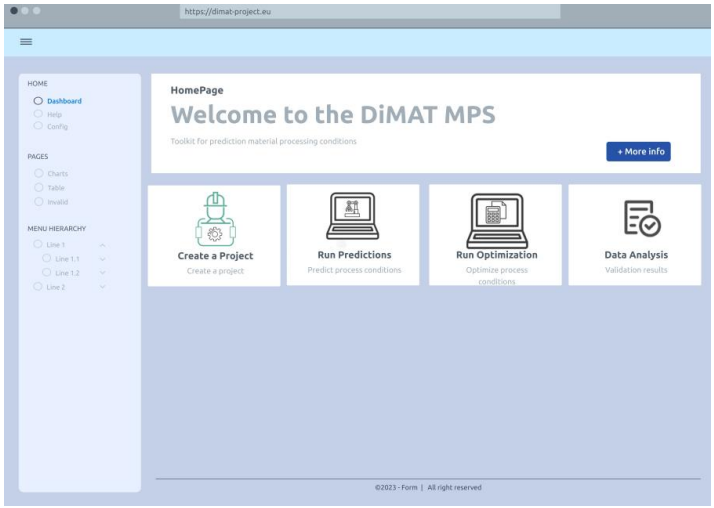
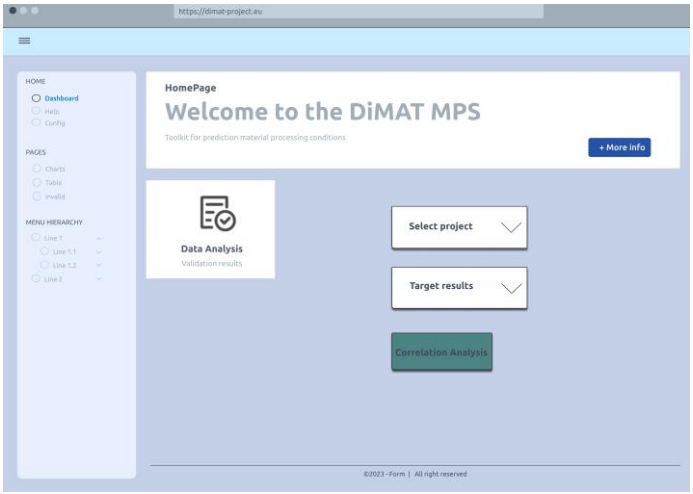
Function	Mockup
Home Page	
Validation and data Analysis	

Table 41. Di^{MPS} Mockup Activity "Correlation Analysis and results validation with database info"

3.10 DIMAT DIGITAL TWIN FOR PROCESS CONTROL (DIDTPC)

The DTPC toolkit aims to construct, using open-source software, digital twins as abstractions of the main IoT devices. The toolkit will be able to interact with the physical devices, but it

will also use the virtualized functions that are considered in the DiMAT Simulation and Optimization Suite as well as specialized simulation software. The DTPC will offer modular components for access control to the IoT devices as well as self-configuration and self-management functionalities. Moreover, different communication protocols will be supported (e.g., HTTP, MQTT, etc.) and open APIs will be provided for the users interacting with the toolkit.

3.10.1 Overview

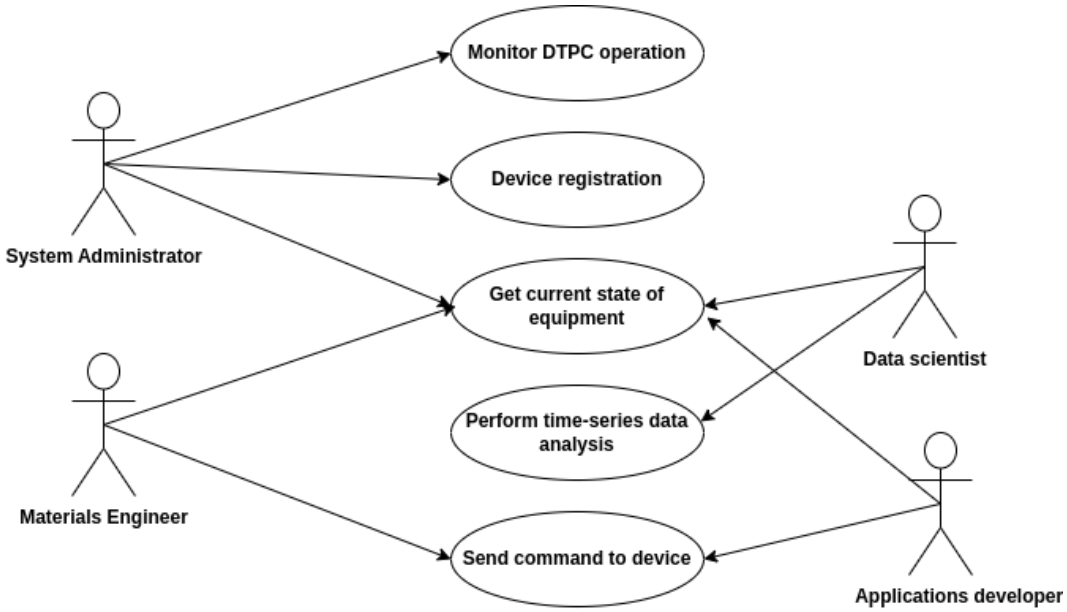


Figure 48: Use Case diagram of Di^{DTPC}

3.10.2 Roles

The DiMAT's DTPC toolkit will be used by users that assume different roles such as:

- **System administrator:** responsible for the continuous operation of the DTPC toolkit, can add new equipment to be "twinned" to the platform and perform health-checks, also responsible for addressing security concerns.
- **Data scientist:** responsible for analyzing the collected data that are fed to the DTPC via the appropriate functions and reaching conclusions about the manufacturing processes.
- **Materials engineer:** expert on the operation of the equipment and the employed materials, responsible for sending explicit commands through the DTPC to the physical equipment, when necessary, can recommend simulation software to be used.

- **Applications developer:** responsible for developing applications that utilize the digital twins managed in the toolkit.

3.10.3 Activities

3.10.3.1 Activity 1 - "Send command to device"

This activity describes the steps to be followed to send a command through the DTPC toolkit to apply an action to a physical device. The action is enforced by an IoT actuator.

3.10.3.1.0 Activity 1 – Triggers

The Material Engineer or the Applications Developer selects the "Send command" option along with the required parameters from the toolkit's interface.

3.10.3.1.1 Activity 1 – Workflow

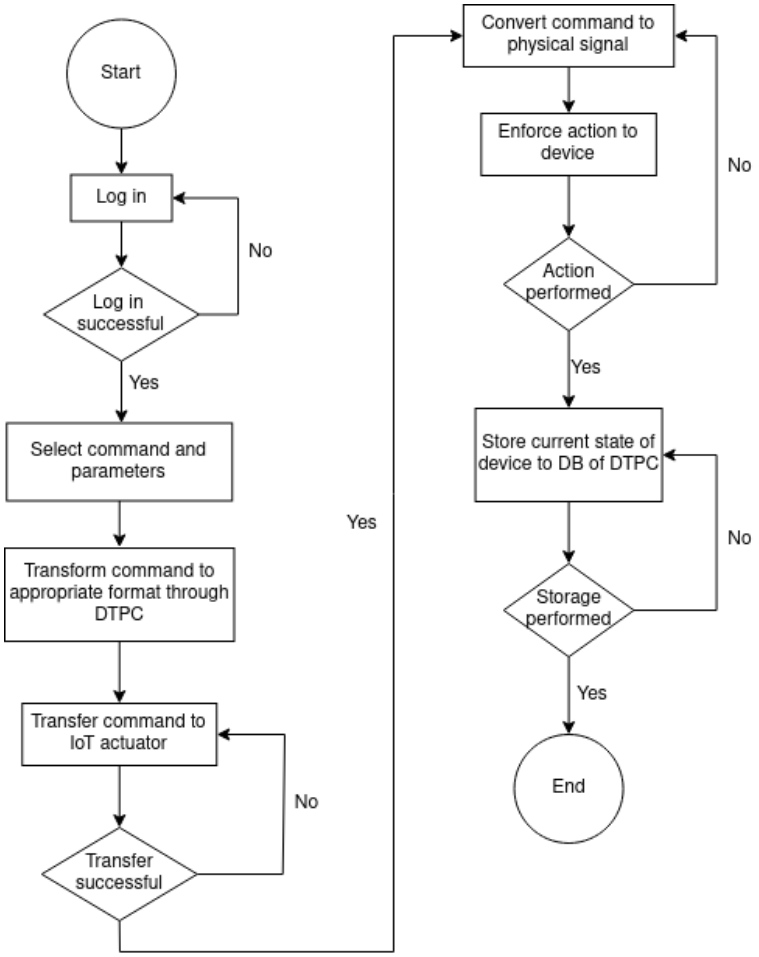


Figure 49: Workflow "Send command to device" of Di^{DTPC}

3.10.3.1.2 Activity 1 – Effects

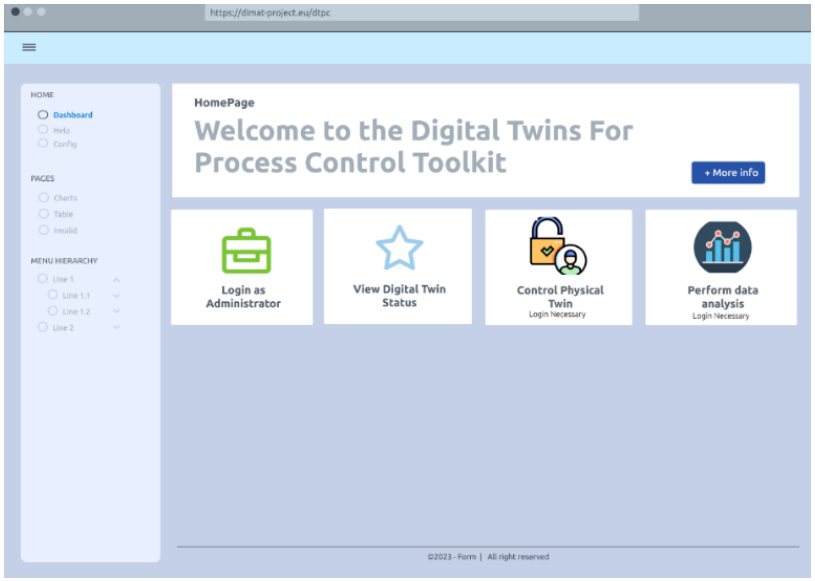
Action is performed to the device and its new state is stored in the database of the Digital Twin.

3.10.3.1.3 Activity 1 – Constraints

- The command and its parameters must be within the ranges that the toolkit can process.

- The digital twin should be able to communicate with the IoT actuator and the actuator with the physical equipment.
- It should be ensured that the command and parameters will not cause undesirable effects to the physical device.
- The interaction between the Digital Twin and the IoT device has to be in accordance with the applied security and privacy policies. For instance, the communication of the Digital Twin and the IoT device may be done over HTTPS, or part of the data has to be processed at the edge or the IoT part of the infrastructure due to privacy-related locality constraints.
- The data concerning the state of device must be compliant with the internal schema of the Digital Twin's database.

3.10.3.1.4 Activity 1 – Mockups

Function	Mockup
View Homepage	

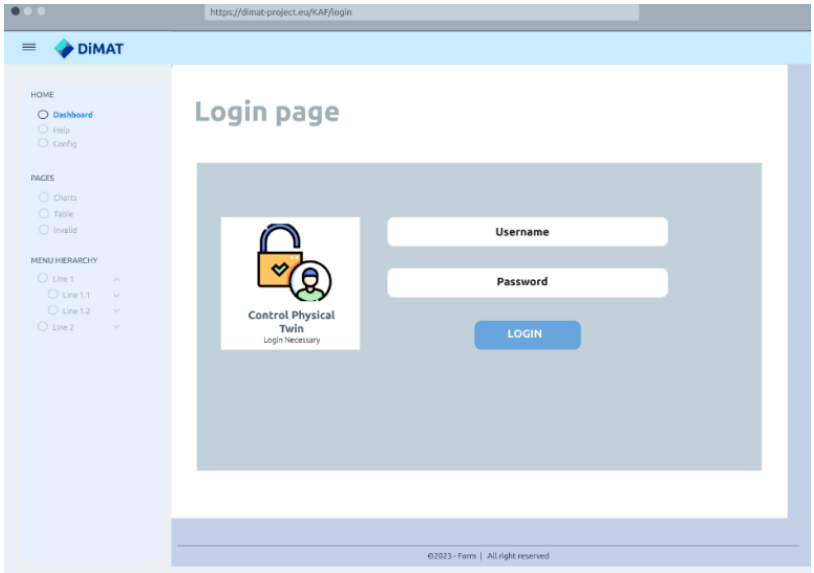
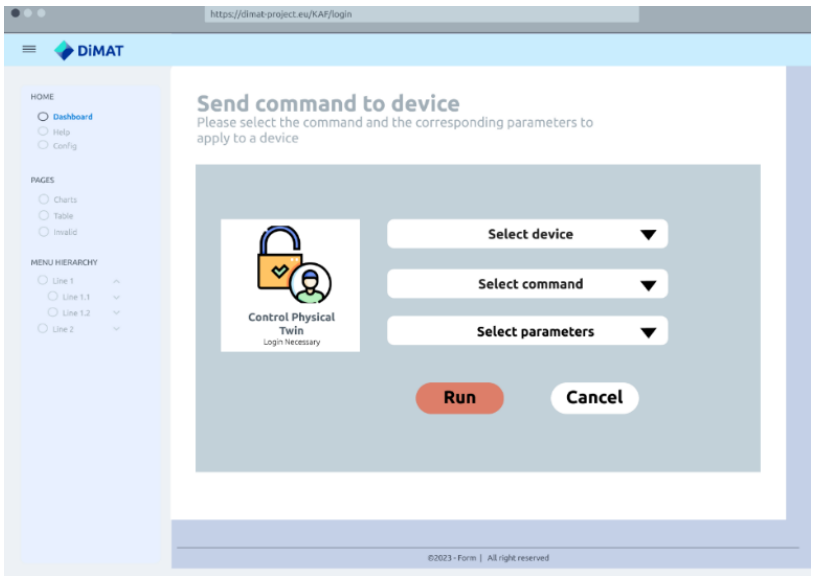
View Login page	
Select command and parameters	

Table 42. Di^{DTPC} Mockup Activity "Send command to device"

3.10.3.2 Activity 2 - “View Digital Twin Status”

This is an activity that can be performed by all users interacting with the DTPC toolkit and describes the process of getting the current state of a device that the Digital Twin handles, for health check or data analysis purposes.

3.10.3.2.0 Activity 2 – Triggers

All users can request access to the current state of the device through the toolkit's interface.

3.10.3.2.1 Activity 2 – Workflow

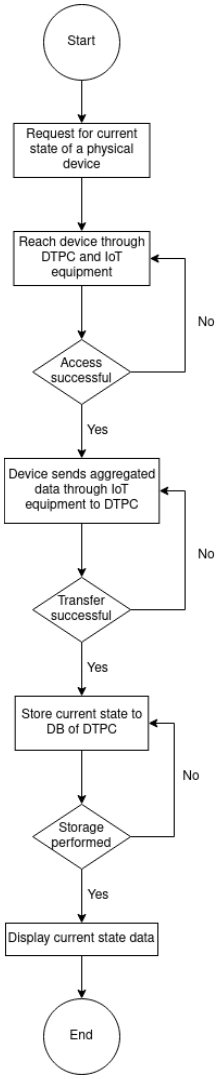


Figure 50: Workflow "View Digital Twin Status" of Di^{DTPC}

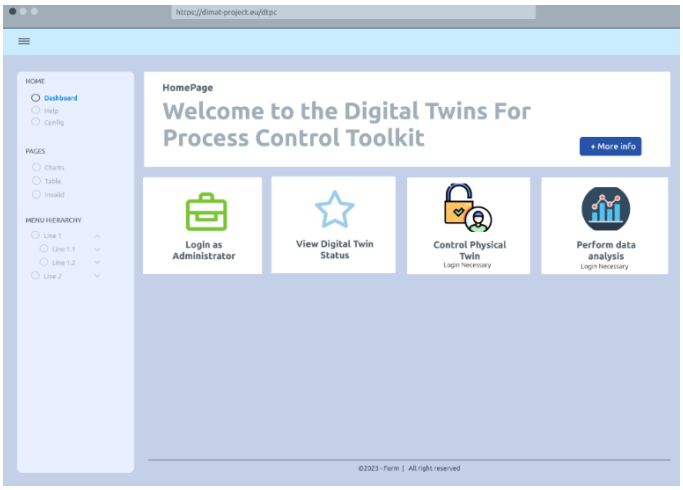
3.10.3.2.2 Activity 2 – Effects

Display device's current state data to the user and store it to the internal database of the Digital Twin.

3.10.3.2.3 Activity 2 – Constraints

- The data stored in the Digital Twin's database must be compliant with its internal schema.
- The digital twin should be able to communicate with the physical equipment.
- Data should be displayed to the user in a human readable form.
- The interaction between the end users and the Digital Twin has to be in accordance with the applied authentication, authorization and security policies. The specification of the exact policies depends on the application scenario and are mainly provided by the vertical industry where the Digital Twin is targeted to.

3.10.3.2.4 Activity 2 - Mockups

Function	Mockup
View Homepage	

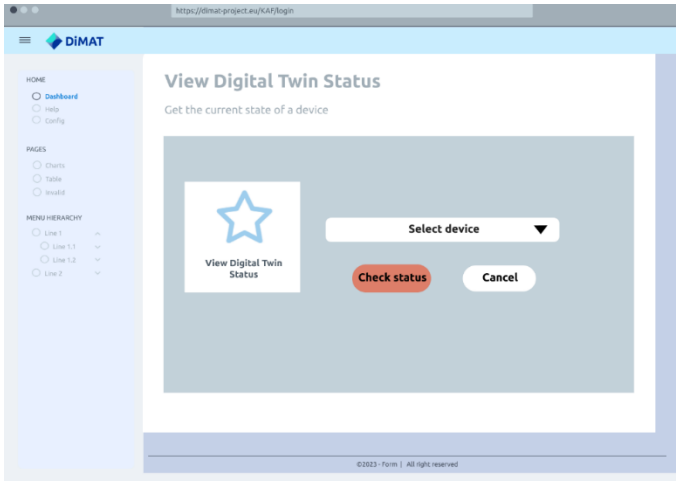
Select Digital Twin to check	

Table 43: Di^{DTPC} Mockup Activity "View Digital Twin Status"

3.10.3.3 Activity 3 - “Device Registration”

In this activity the system’s administrator identifies a new device that needs to be included in the developed digital twin. The new device may be a small device added to the existing equipment (e.g., a sensor, an IoT device in general, etc.) or brand-new equipment.

3.10.3.3.0 Activity 3 – Triggers

The administrator logs in to the platform and selects the add new equipment option.

3.10.3.3.1 Activity 3 – Workflow

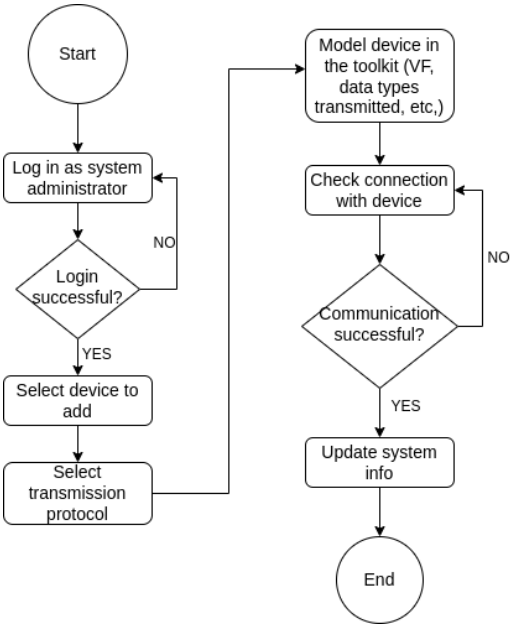


Figure 51: Workflow "Device Registration" of Di^{DTPC}

3.10.3.3.2 Activity 3 – Effects

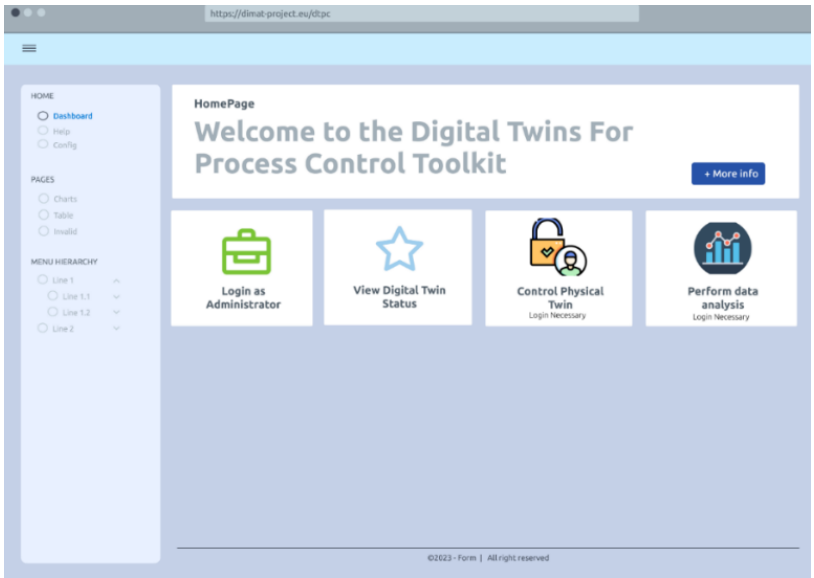
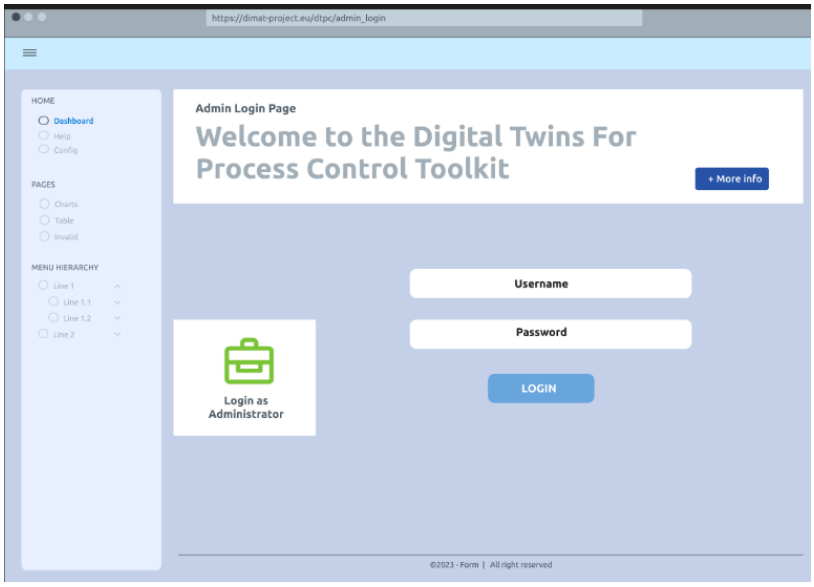
- The new device has been connected to the digital twin toolkit.
- The Digital Twin now represents the physical equipment more accurately.
- Virtual functions can have as inputs more real time-series data from the device's data transmissions.

3.10.3.3.3 Activity 3 – Constraints

- The only user that can perform this action is the system's administrator.
- The device must be able to either communicate with a gateway or transmit directly its data to the platform.
- The data transmitted should be in formats suitable to the ones that the DTPC toolkit can handle.
- Resource constraints of the DTPC platform should be respected, meaning there is a limit to the number of devices that can connect to the toolkit.

- The new device must be secured and unauthorized access to it should be prevented to maintain the toolkit's integrity as well as the safety of the equipment's operation.

3.10.3.3.4 Activity 3 – Mockups

Function	Mockup
View Homepage	
View Login page	

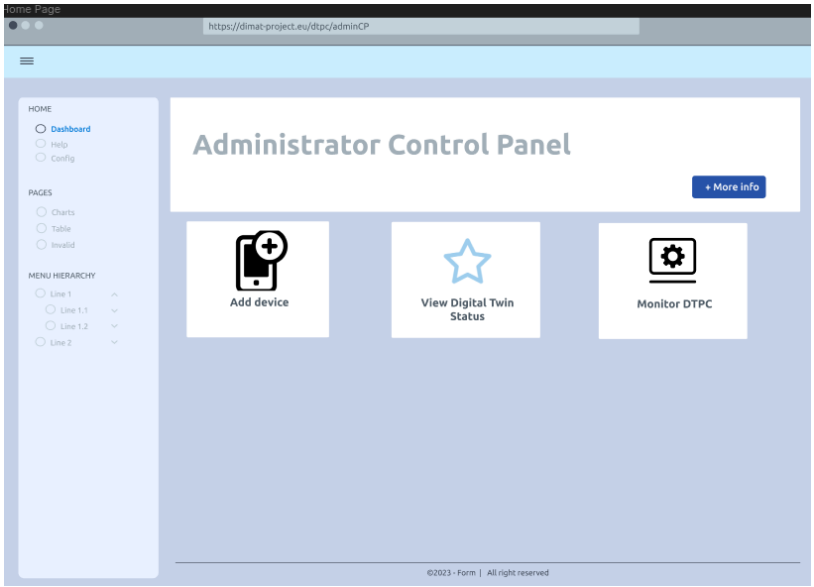
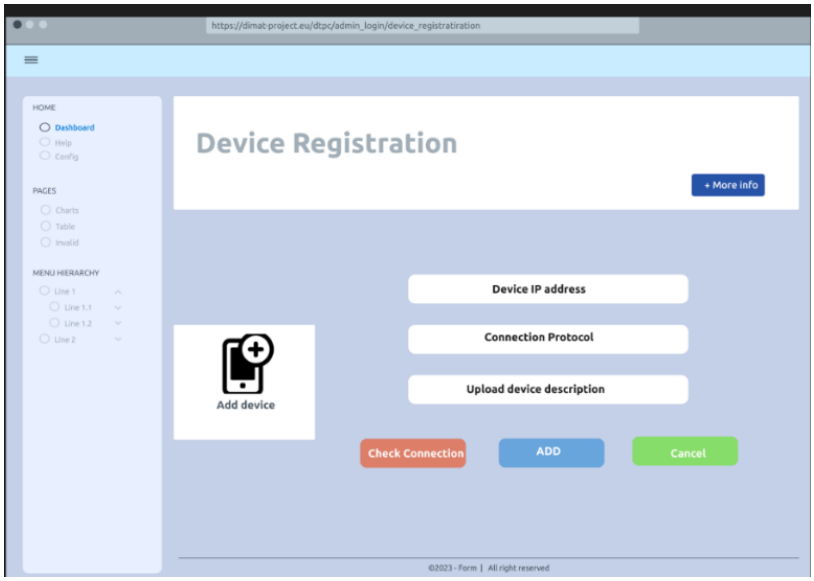
View Admin Control Panel	
Device Registration	

Table 44. Di^{DTPC} Mockup Activity "Device Registration"

3.10.3.4 Activity 4 - “Monitor DTPC operation”

The system’s administrator is responsible for the smooth operation of the toolkit and periodically performs health-checks.

3.10.3.4.0 Activity 4 – Triggers

The system's administrator logs in and selects the available option from the menu.

3.10.3.4.1 Activity 4 – Workflow

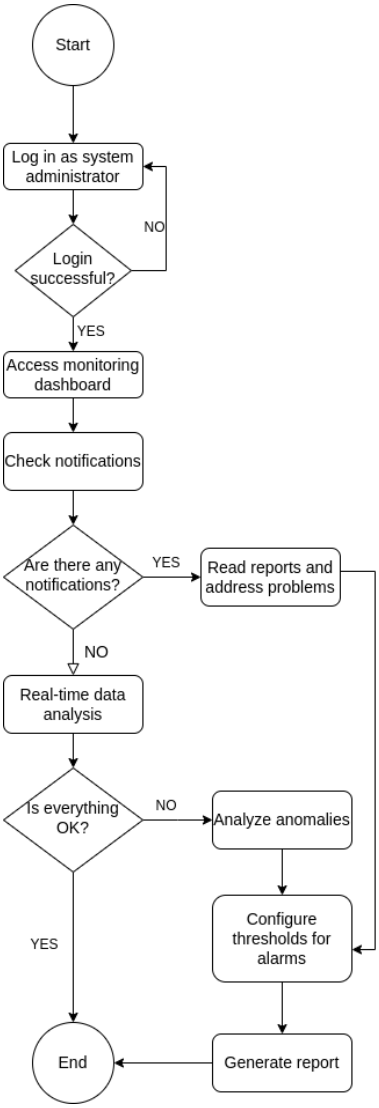


Figure 52: Workflow "Monitor DTPC operation" of Di^{DTPC}

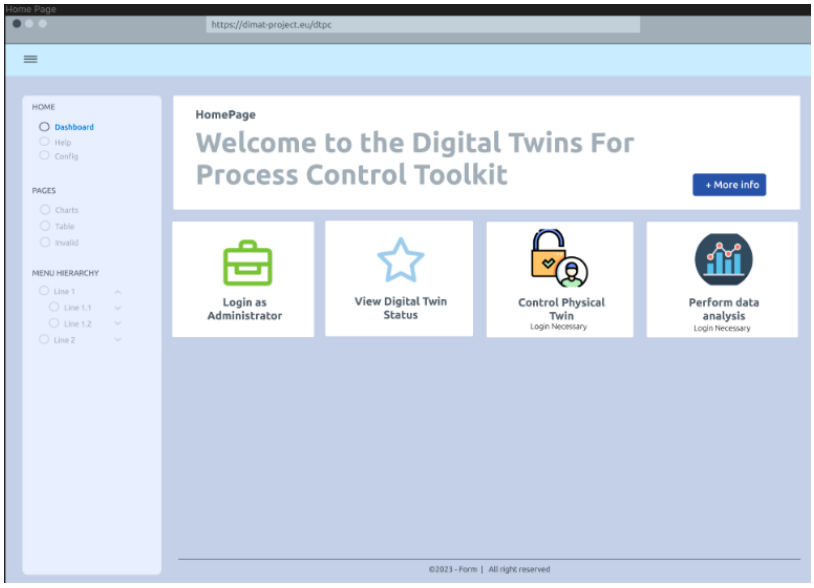
3.10.3.4.2 Activity 4 – Effects

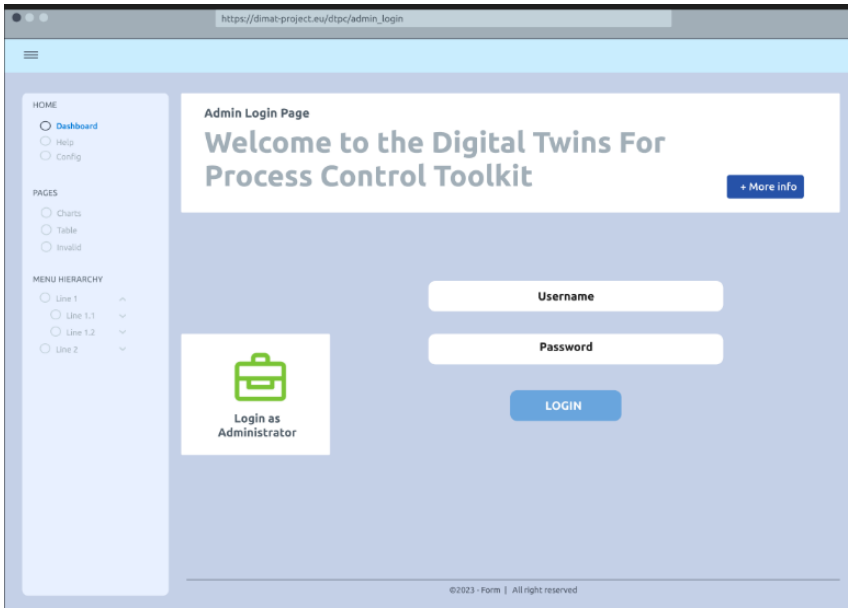
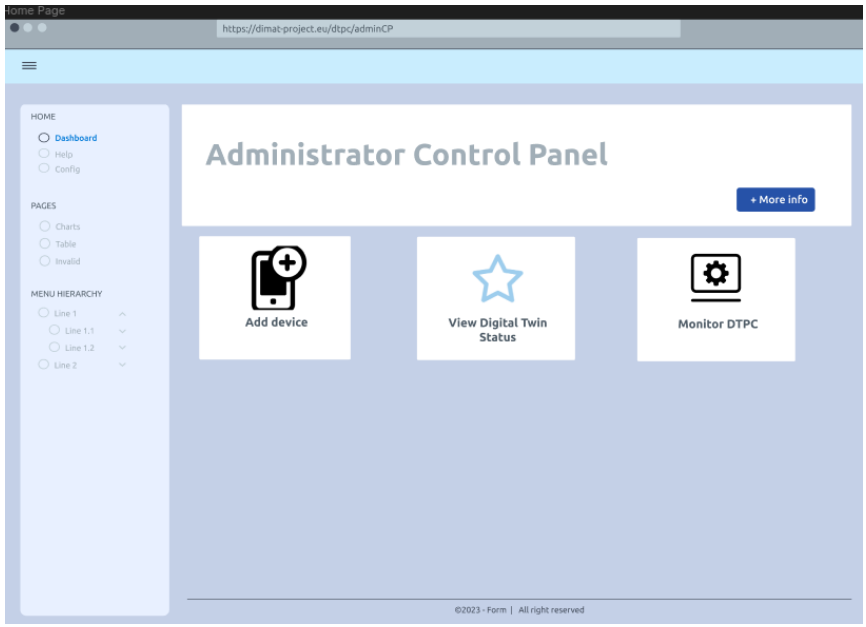
- System stability because of the correction of problems.
- Comprehensive reports about the system status.
- Timely maintenance of the toolkit and the equipment, detection of faulty equipment.

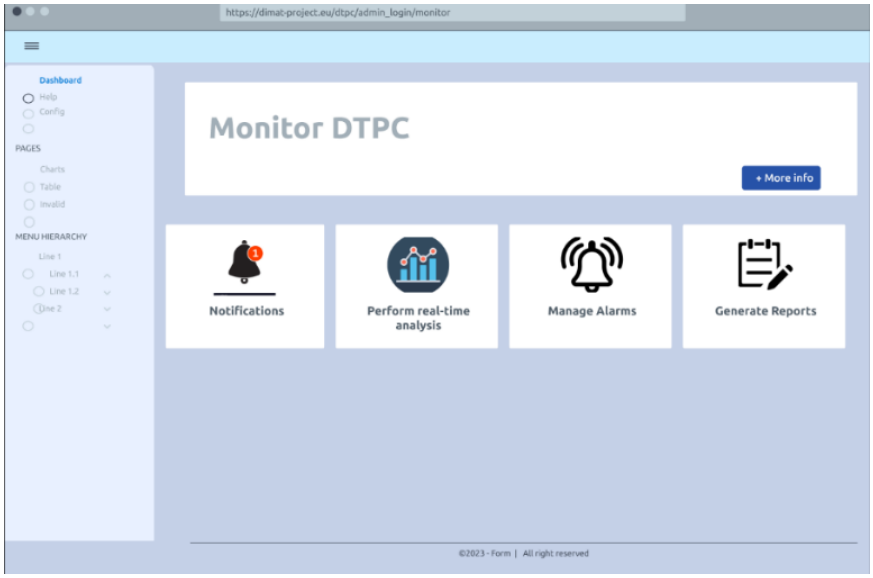
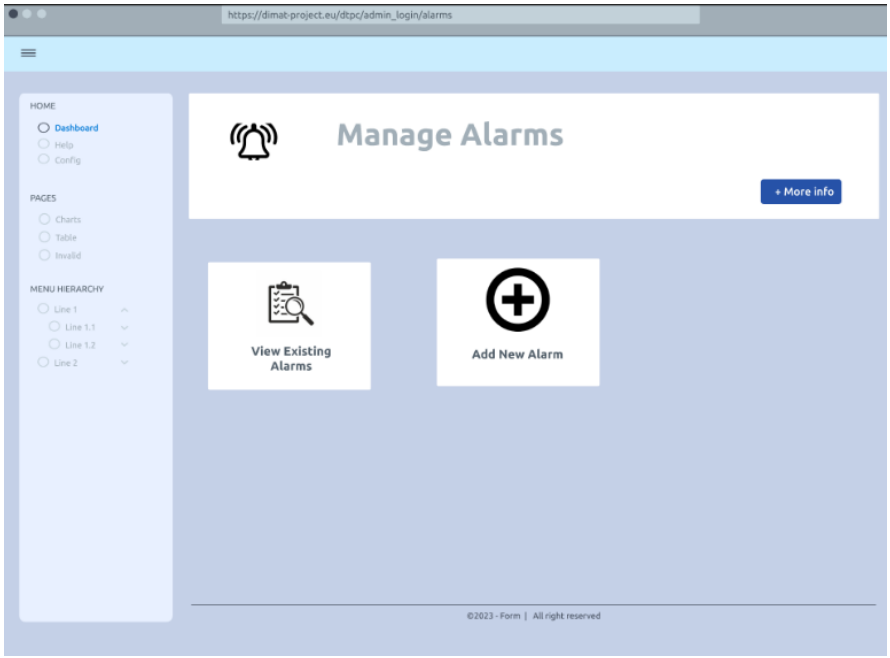
3.10.3.4.3 Activity 4 – Constraints

- The only user suitable for executing this activity is the system administrator.
- The digital twin should be able to communicate with the physical equipment.
- Thresholds should be carefully applied for setting alarms as to be useful and mitigate any "false alarms".

3.10.3.4.4 Activity 4 – Mockups

Function	Mockup
View Homepage	

View Login page	
View Administrator Control Panel	

View monitoring options	
Manage alarms	

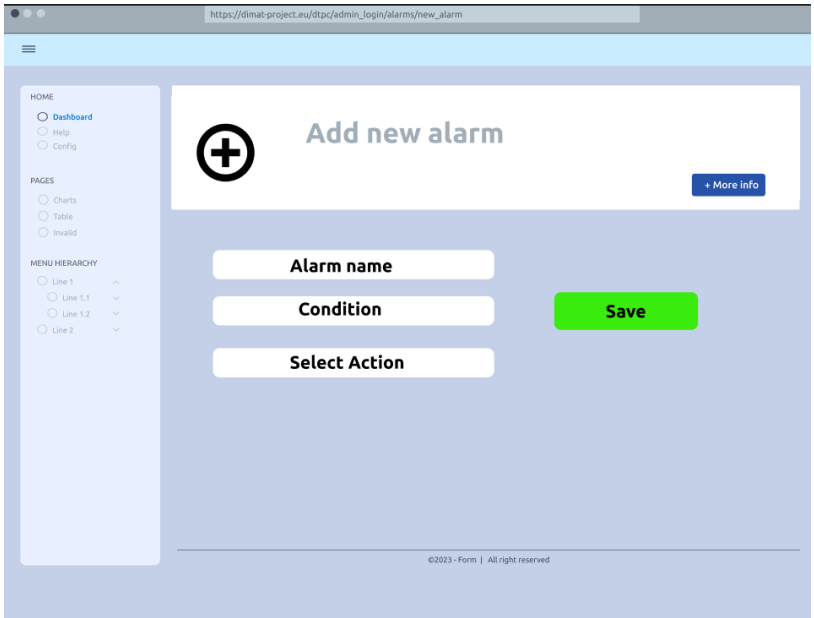
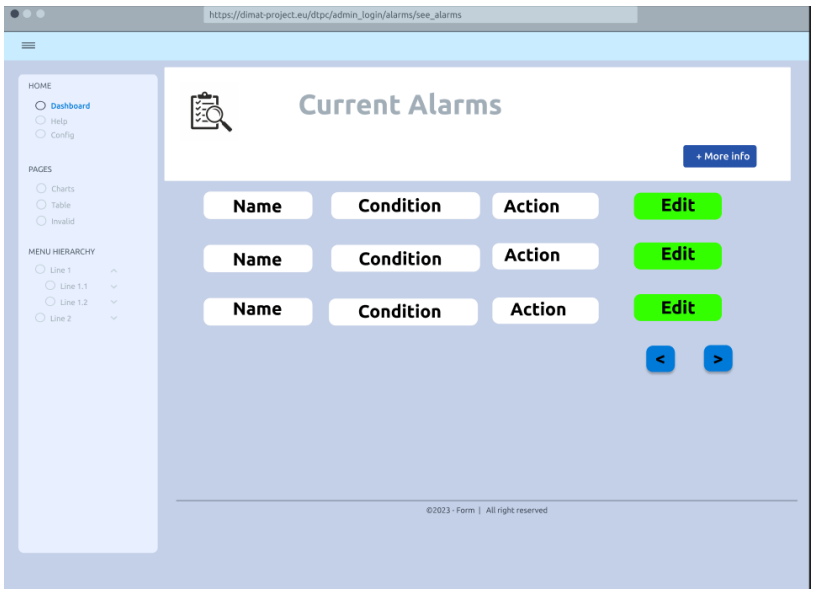
Add alarm	
View current alarms	

Table 45. Di^{DTPC} Mockup Activity "Monitor DTPC operation"

3.10.3.5 Activity 5 - “Analyse time-series data”

A set of time-series analysis functions are going to be supported by the DTPC, considering mainly forecasting and time series decomposition mechanisms. For instance, through

forecasting, predictive maintenance functionalities may be supported, while, through time series decomposition, specific patterns may be identified.

3.10.3.5.0 Activity 5 – Triggers

The triggering of a function that supports analysis of time-series data will be mostly software driven, following the business logic defined per application part. Multiple triggers may be applicable in a short time, considering the time sensitivity required in the analysis part.

3.10.3.5.1 Activity 5 – Workflow

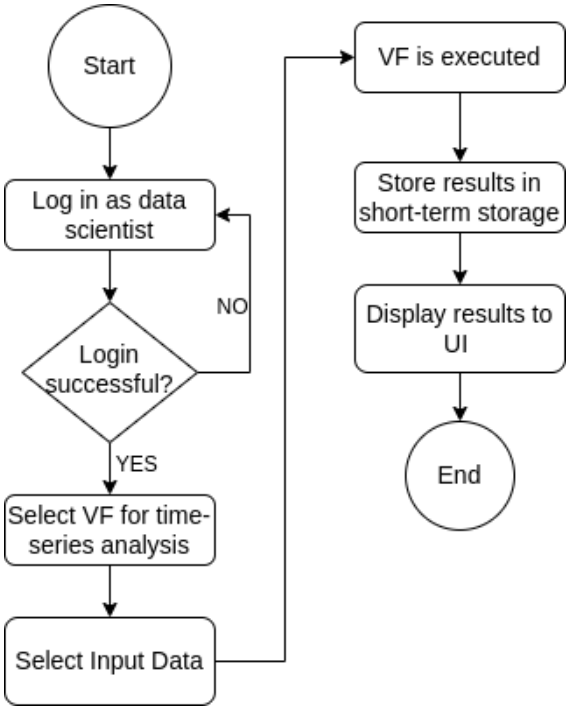


Figure 53: Workflow "Analyse time-series data" of Di^{DTPC}

An indicative workflow for the usage of a time-series data analysis process is:

- A time series data analysis function is developed to support specific business needs (e.g., predictive maintenance).
- The function is onboarded in the DTPC development instance that may be targeted to a specific IoT device, or a combination of IoT devices.

- The function is executed based on real-time (or offline) data collected by the set of IoT devices.
- The function can be triggered/consumed based on the application business logic or executed in periodic way.
- Based on the analysis results, further actions can be triggered.

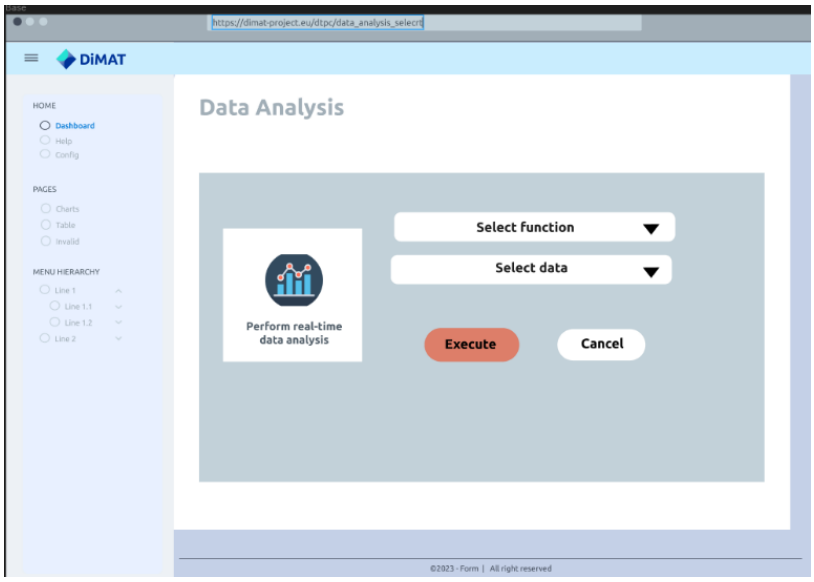
3.10.3.5.2 Activity 5 – Effects

Facilitation of decision making in various manufacturing processes.

3.10.3.5.3 Activity 5 – Constraints

There is a need for the provision of qualitative time-series data with proper timestamping and proper sampling procedures. Such data have also to be temporarily stored in the data store hosted by the DTPC.

3.10.3.5.4 Activity 5 – Mockups

Function	Mockup
Select virtual function and input data	

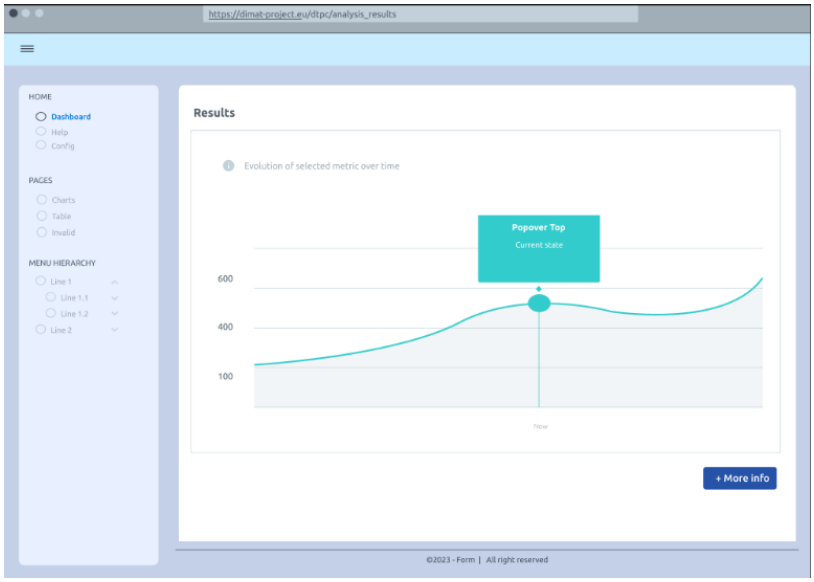
View analysis results	

Table 46. Di^{DTPC} Mockup Activity "Analyse time-series data"

4 FUNCTIONAL VIEWPOINT

This task will decompose DiMAT Framework into its Control, Operations, Information, Application, and Business domains and identify the data, decision and command/request flows that circulate between them. Once this decomposition has been performed, it will elaborate on the control, coordination and orchestration exercised from each of these domains and the different typical operations from these domains.

4.1 FUNCTIONAL VIEWPOINT APPROACH FOR DIMAT

4.1.1 Functional Viewpoint Concepts

According to IIRA: "The functional viewpoint focuses on the functional components in an IIoT system, their structure and interrelation, the interfaces and interactions between them, and the relation and interactions of the system with external elements in the environment, to support the usages and activities of the overall system." The following sections will focus on describing:

- Functional components along with their interfaces (published operations).
- Interactions between external elements, the system, and the functional components.
- Mapping the identified functional components to the IIRA five main functional domains.

4.1.2 Functional Viewpoint Methodology

The methodology to complete the task was defined with the following steps:

- Firstly, the development of a UML. With this diagram we will define the components of the DiMAT solutions. This will provide a first picture of the overall architecture of the solutions.
- Secondly, the different solution developers are asked to explain in more detail the functionality of each component that makes up their solution. For this purpose, a series of tables will be developed where each solution provider will be asked to describe the components and define the interfaces.
- Thirdly, the roles previously defined in the Usage Viewpoint will be aligned with the solution, and then several sequence diagrams will be developed to understand how the solution works.

4.1.3 Diagram of components

The general architecture of the toolkit is presented here. Appropriate tools to convey this information are UML component diagrams or package diagrams (Figure 116). In this case a component diagram is used. The diagram identifies 6 main functional components.

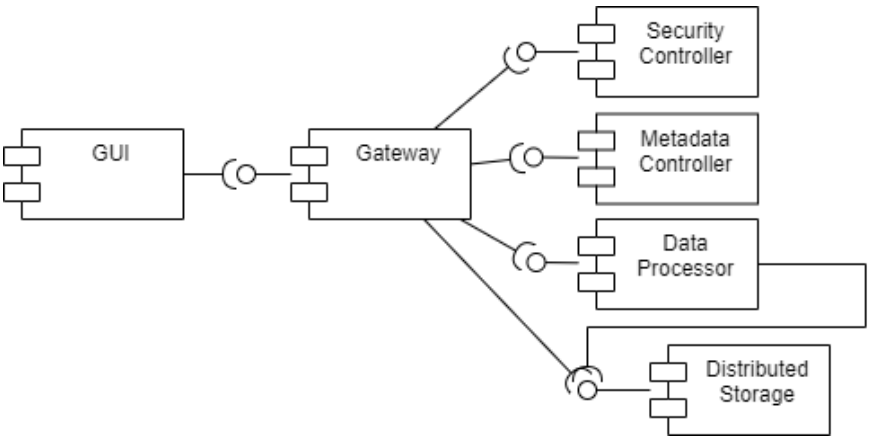


Figure 54: Example diagram

4.1.3.1 Functional components

This should describe the main functional components identified in the General Architecture section. Simple templates may be defined for this purpose. Each component's name, purpose and coarse-grained interface can be defined. The interface should include all the operations used in the Interactions section.

Name:	*Data Processor*	
Description:	*It implements data processing pipelines.* *What? Who? Where? Why?*	
Interface		
Operation	***Inputs***	***Outputs***
ImportData	- Data source info	

	- Data destination info - Processing pipeline spec	
ProcessData	- Input data stream to process - Processing pipeline spec	- Output stream processed

Table 47. Example Table functional component

4.1.3.2 Interactions

This section contains the main interactions of the system. Interactions are typically initiated by external actors, received by a public component, and followed by a chain of interactions between internal components to provide the required functionality.

Ideally, each use case identified in the Usage Viewpoint would translate into an interaction in the Functional Viewpoint. The UML sequence diagram is a natural tool for representing these interactions, where the participants represent either an external entity (user, system, etc.) or a component. The operations invoked on each component should be included in its interface specification in the previous section.

4.1.3.3 Sequence diagram

Below are two examples of how sequence diagrams are used. These diagrams are commonly used in software development to visualize and understand how different objects or parts of a system communicate and work together to perform a particular task or functionality.

In a sequence diagram, objects are represented by vertical rectangles and the interactions between them are shown by arrows indicating the flow of messages or method calls. Each object's own vertical lifeline shows its existence and state over time.

Sequence diagrams are handy for modelling the logic behind the processes and interactions in a system, helping developers and designers to understand and document how a system will work before implementing it. These diagrams are fundamental to the Unified Modelling Language (UML) notation, a standard widely used in the software development industry to represent and communicate complex system designs and structures.

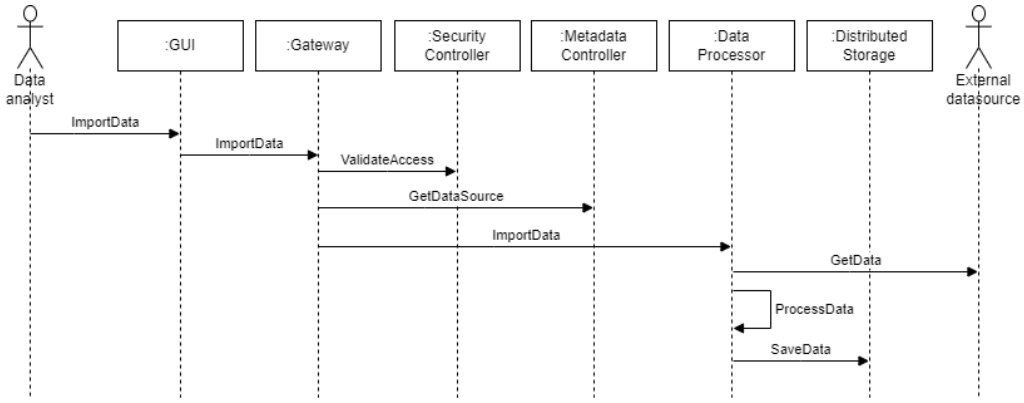


Figure 55: Example of sequence diagram

4.1.3.4 Mapping of functional components to IIRA functional domain

According to IIRA, typical IIOT systems decompose into the following five main functional domains. Functional components typically belong to one of these domains.

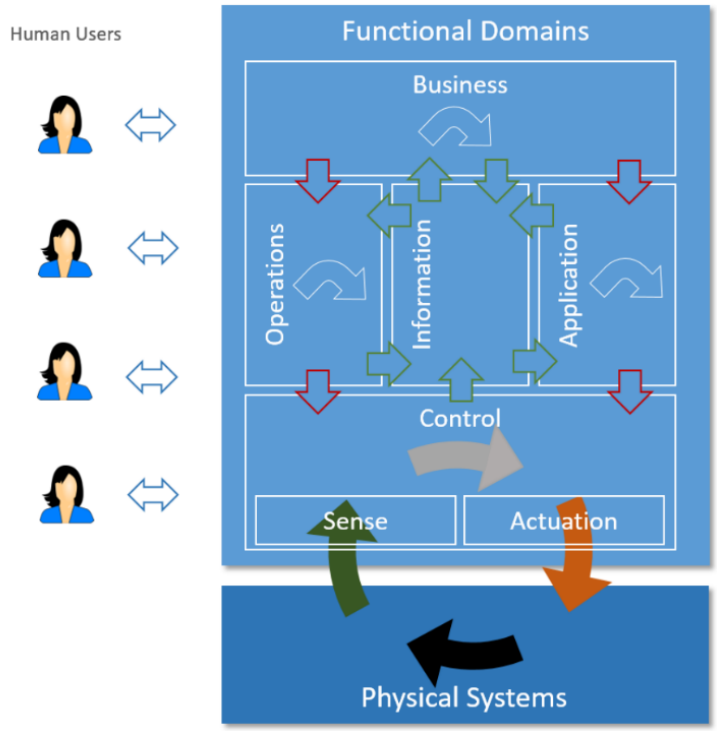


Figure 56: Functional Domains

- The application domain refers to functions that implement the logic of an application and perform specific functions related to a particular business. These functions operate at a high level and are optimized globally.
- The business domain focuses on the analysis of operations that affect both ends of Industrial Internet systems, considering both legacy and new systems. These functions include business processes and procedural activities.
- The control domain includes functions performed by industrial control systems. These functions include detailed data manipulation, rules and logic, and physical system control. Accuracy and timing are critical, and these systems may be geographically distributed.
- The information domain includes functions for collecting data from various domains, particularly the control domain and transforming, storing, modelling, and analyzing this data to provide high-level intelligence about the system. These functions complement those of the control domain, focusing on decision-making and long-term optimization.
- The operations domain is responsible for provisioning, managing, monitoring, and optimizing the systems in the control domain. Unlike traditional industrial control systems, the Industrial Internet seeks to optimize operations at multiple levels, opening opportunities to create greater value for the business and its customers.

Mapping each functional component identified in the previous sections to the corresponding functional domain and subdomain is desired.

Component	***Domain***	***Subdomain***
GUI	*Application*	*UI*
Gateway	*Application*	*API + Logic & Rules*
Security Controller	*Application*	*Logic & Rules*
Metadata Controller	*Information*	*Data*
Data Processor	*Information*	*Analysis*
Distributed Storage	*Information*	*Data*

Table 48. Example table functional domain

4.2 DIMAT CLOUD MATERIALS DATABASE (DI^{CMDB})

4.2.1 General architecture

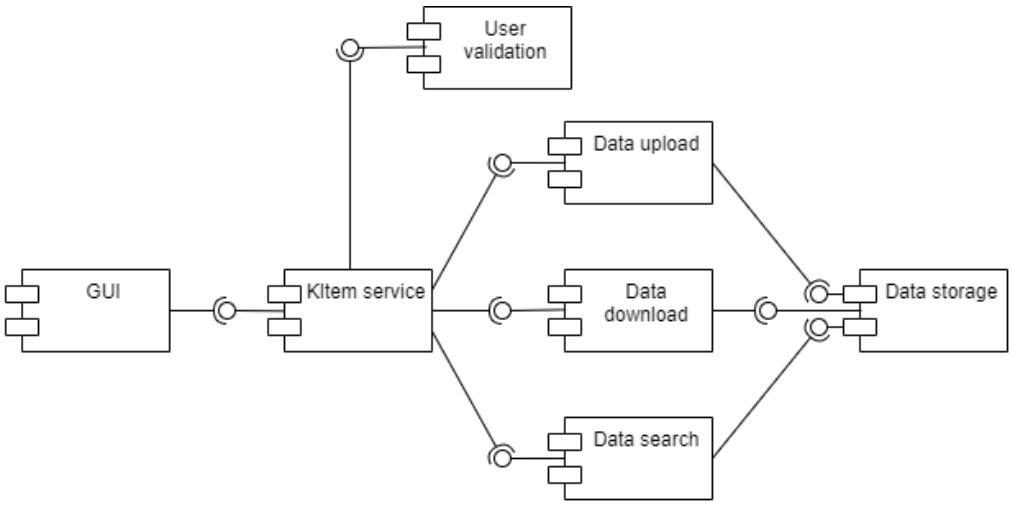


Figure 57: Cloud Materials Database diagram

The diagram illustrates how a user interacts with a system through a **Graphical User Interface (GUI)**, which connects to a service referred to as the **KItem service**. The KItem service manages several key functions: **user validation** (user authentication), ensuring that only authorized individuals gain access to the system; **data upload** and **data download**, allowing users to respectively submit and retrieve information; and **data search**, facilitating the retrieval of stored information. All data is housed within a **data storage** module, which supports the processes of uploading, downloading, and searching.

4.2.2 Functional components

The tables below describe the main functional components identified in the General Architecture of the CMDB toolkit.

4.2.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Dataspace.*	
Interface		
Operation	***Inputs***	***Outputs***

Create a kItem	- Process/dataset info	- New kItem
Upload datasets	- Files and metadata	- Populated dataset item
Visualize kItems	- Data to visualize	- Data visualized
Download data	- Data to download	- Data file
Modify metadata	- Modifications selected	- Metadata modified
Search dataspace	- Query + filters	- Matching results (processes and datasets)

Table 49. GUI of the functional component of the DiCMDB toolkit

4.2.2.2 KItem service

Name:	*KItem service*	
Description:	*Handles Datasets, Processes, and their steps*	
Interface		
Operation	***Inputs***	***Outputs***
Create a kItem	- Process/dataset info	- New kItem
Upload datasets	- Files and metadata	- Populated dataset item
Visualize kItems	- Data to visualize	- Data visualized
Download data	- Data to download	- Data file
Modify metadata	- Modifications selected	- Metadata modified

Search dataspace	- Query + filters	- Matching results (processes and datasets)

Table 50. Kltem service of the functional component of the Di^{CMDB} toolkit

4.2.2.3 User validation

Name:	*User validation*	
Description:	*Checks the users are logged in*	
Interface		
Operation	***Inputs***	***Outputs***
Check user access	- Credentials	- Access is granted

Table 51. User validation of the functional component of the Di^{CMDB} toolkit

4.2.2.4 Data upload

Name:	*Data upload*	
Description:	*Enables the upload of new data*	
Interface		
Operation	***Inputs***	***Outputs***
Upload files	- Files and metadata	- Success message

Table 52. Data upload of the functional component of the Di^{CMDB} toolkit

4.2.2.5 Data download

Name:	*Data download*
Description:	*Enables the download of existing data*

Interface		
Operation	***Inputs***	***Outputs***
Download data	- Data to download	- Data file

Table 53. Data download of the functional component of the Di^{CMDB} toolkit

4.2.2.6 Data search

Name:	*Data search*	
Description:	*Enables the search of existing data*	
Interface		
Operation	***Inputs***	***Outputs***
Search dataspace	- Query + filters	- Matching results

Table 54. Data search of the functional component of the Di^{CMDB} toolkit

4.2.2.7 Data storage

Name:	*Data storage*	
Description:	*Storage of the different types of data*	
Interface		
Operation	***Inputs***	***Outputs***
Upload files	- Files and metadata	- Success message
Download data	- Data to download	- Data file

Table 55. Data storage of the functional component of the Di^{CMDB} toolkit

Note that the data storage component represents an abstraction of different storages, namely relational, graph and object. For simplicity of understanding, they have been unified in this context.

4.2.3 Interactions

The relevant roles identified in the usage viewpoint for this toolkit include:

- **System administrator:** In charge of the deployment and management of the platform.
- **Data scientist:** Uploads and manages data generated from different sources and frames it inside a process (with the corresponding CHADA/MODA definition).
- **Data consumer:** Browses, searches, and retrieves the data stored in the CMDB.

4.2.3.1 Create a kItem

A data scientist navigates to the create section of the GUI and selects an option (process or dataset). After choosing a name, an initial blank kItem page is shown. This can then be filled by the user by providing more metadata and information, which will then be rendered in the updated kItem page.

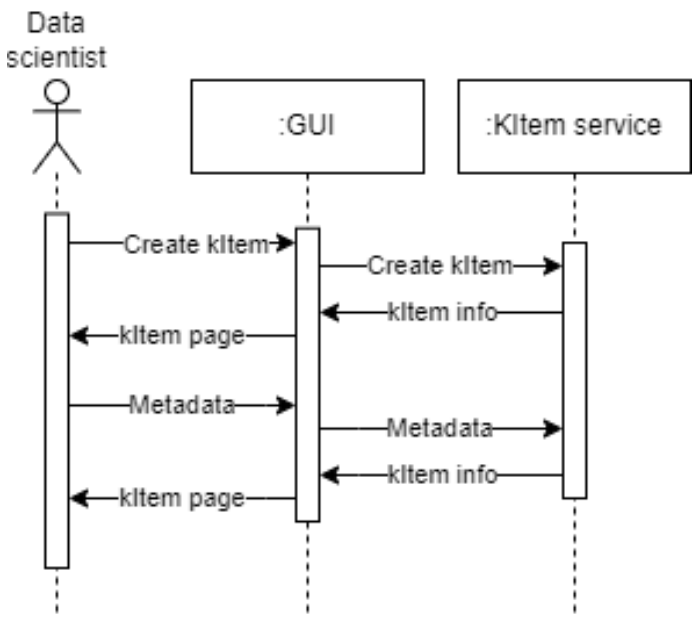


Figure 58: Sequence diagram of the Di^{CMDB} toolkit component " Create a KItem"

4.2.3.2 Upload datasets

For dataset kItems, files can be attached. A formular for metadata must be filled by the user in the GUI, together with the selection of the files to upload.

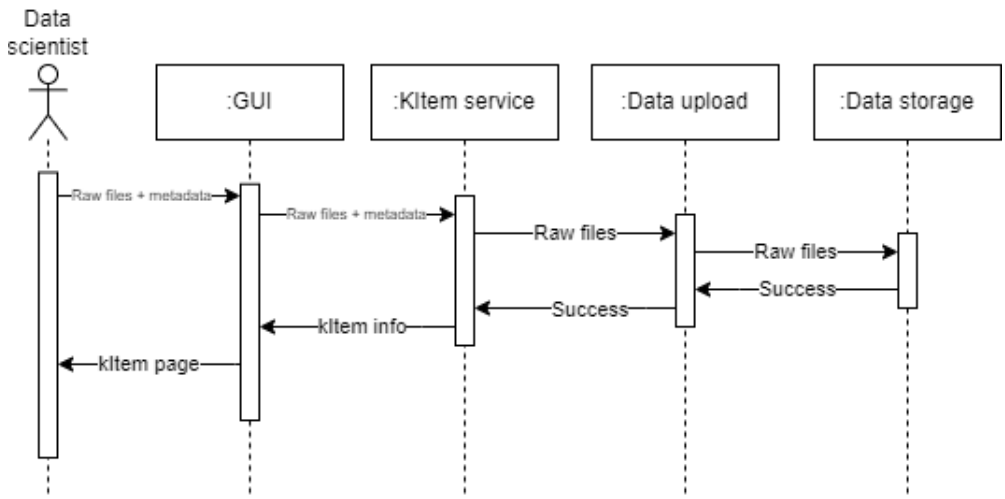


Figure 59: Sequence diagram of the Di^{CMDB} toolkit component "Upload datasets"

4.2.3.3 Visualise kItems

kItems (processes, steps, and datasets) are differentiated by unique identifiers. When a user selects a certain item in the frontend, the id is used for fetching the information from the data storage. Said information is rendered in the GUI for the user.

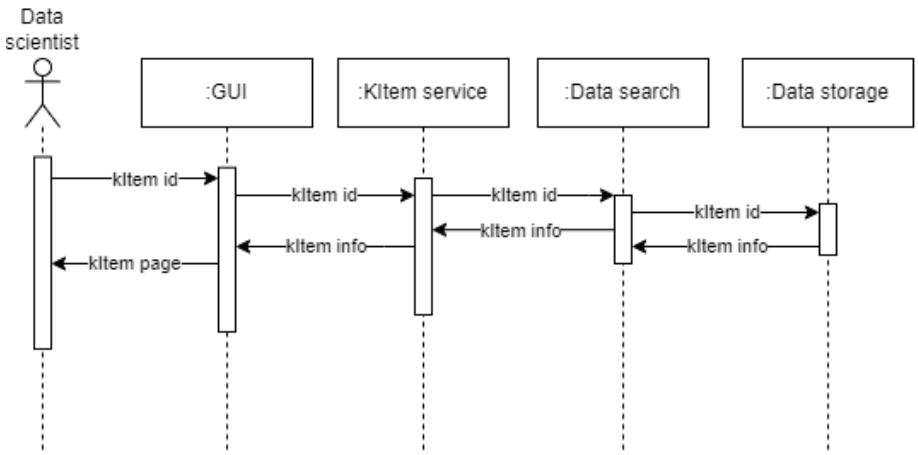


Figure 60: Sequence diagram of the Di^{CMDB} toolkit component "Visualize kItems"

4.2.3.4 Download data

The file selected by the data consumer is sent to the data download component, which fetches the relevant information from the data storage, and presents the file to the user.

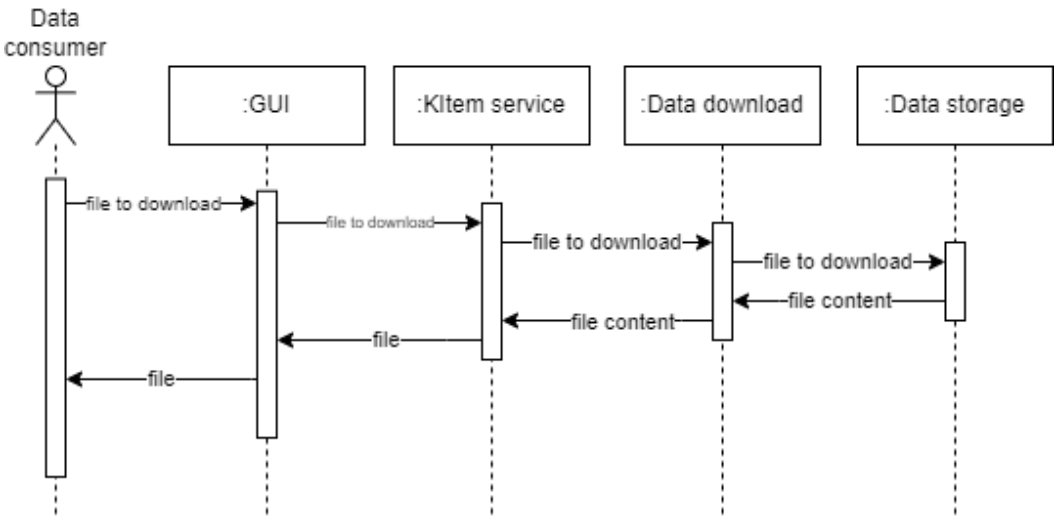


Figure 61: Sequence diagram of the Di^{CMDB} toolkit component "Download data"

4.2.3.5 Modify metadata

When the data scientist decides to change the metadata assigned to a knowledge item, the information is sent to the kItem service for storage, and the updated version rendered in the GUI.

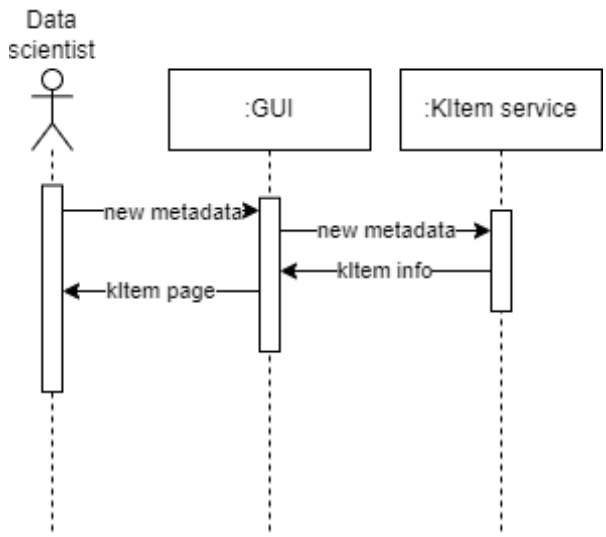


Figure 62: Sequence diagram of the Di^{CMDB} toolkit component "Modify metadata"

4.2.3.6 Search dataspace

The query with potential applied filters is sent to the kItem service, which identifies the matching kItems and returns them to the GUI to be rendered.

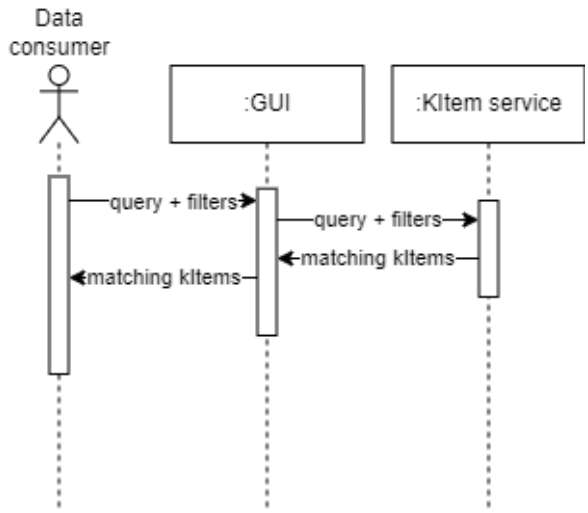


Figure 63: Sequence diagram of the Di^{CMDB} toolkit component "Search dataspace"

4.2.3.7 Check user Access

When a request arrives to the kItem service (backend), the first step is to check that it comes from a valid, active user. For this, the central User Authentication and Authorization instance is contacted. When the provided user information is correct, the kItem service proceeds with the request, returning the client appropriate response. Note that this interaction happens for all requests, even if they are not explicitly shown in all other graphs.

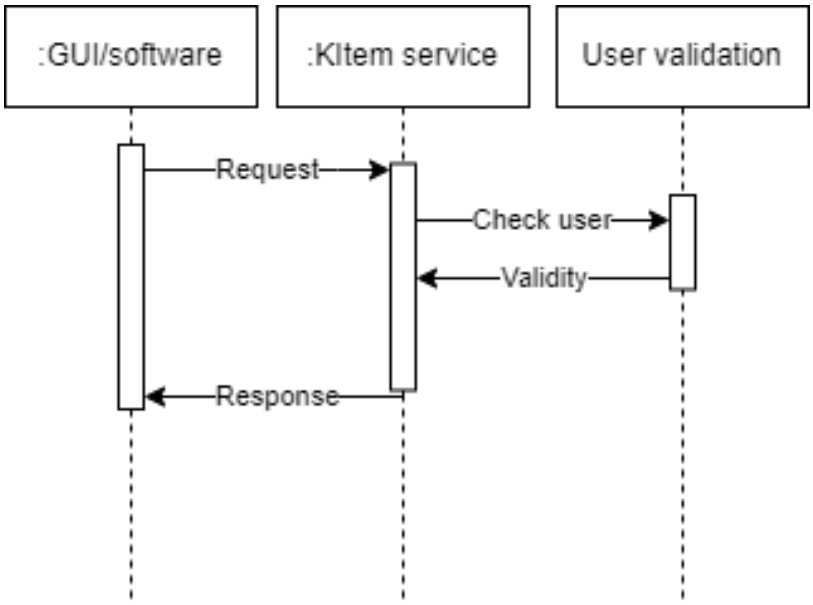


Figure 64: Sequence diagram of the Di^{CMDB} toolkit component "Check user access"

4.2.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
User validation	*Application*	*Logic & Rules*
Data upload	*Application*	*API + Logic & Rules*
Data download	*Application*	*API + Logic & Rules*
Data search	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*

Table 56. Mapping of Functional components of Di^{CMDB} toolkit with IIRA functional domain

4.3 DIMAT KNOWLEDGE ACQUISITION FRAMEWORK (DI^{KAF})

4.3.1 General architecture

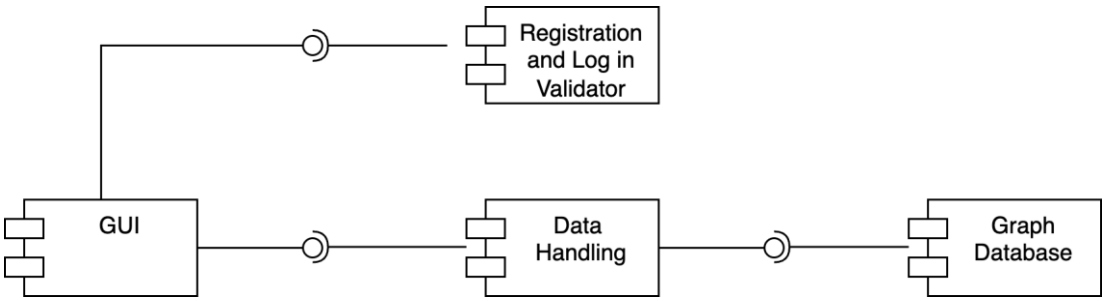


Figure 65: Knowledge Acquisition Framework Diagram

This diagram illustrates how a user interacts with a system through a Graphical User Interface (GUI), which is connected to the main module that is the Data Handling service. The Data Handling module is the central component which manages the processing, manipulation and visualization of data. The process begins with User Validation, implemented by the Registration and Log in Validator module, which ensures that only authorized users can access the system. Once validated, the system supports interactions such as data exploration and discovery, schema explanation, querying and viewing of visualizations, NLP processing and recommendations. All the above are enabled via interaction with the Graph Database component, where the data is stored and organized in a graph structure way, typically optimized for handling complex relationships between data entities.

4.3.2 Functional components

The tables below present the main functional components of the Di^{KAF} toolkit.

4.3.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the KAF toolkit*	
Interface		
Operation	***Inputs***	***Outputs***
Get visualization of data	- Data to visualize	- Data visualized

Add new data source	- Data source information (URI, type, size)	- Data source added
Modify Knowledge Graph	- Elements to modify	- Data modified
Create account	- Credentials	- Access is granted

Table 57. GUI of the functional component of the Di^{KAF} toolkit

4.3.2.2 Registration and Log in Validator

Name:	*Registration and Log in Validator*	
Description:	*Validates the credentials of a user and performs registrations.*	
Interface		
Operation	***Inputs***	***Outputs***
Create account	- Credentials	- Access is granted

Table 58. Registration and Log in Validator of the functional component of the Di^{KAF} toolkit

4.3.2.3 Data Handling

Name:	*Data Handling*	
Description:	*Manages toolkit related data supporting operations such as querying, data visualizations, NLP processing and recommendations*	
Interface		
Operation	***Inputs***	***Outputs***
Get visualization of data	- Data to visualize	- Data visualized
Add new data source	- Data source information (URI, type, size)	- Data source added

Modify Knowledge Graph	- Elements to modify	- Data modified

Table 59. Data Handling of the functional component of the Di^{KAF} toolkit

4.3.2.4 Graph Database

Name:	*Graph Database*	
Description:	*Stores and organizes data in a graph structure way*	
Interface		
Operation	***Inputs***	***Outputs***
Get visualization of data	- Data to visualize	- Data visualized
Add new data source	- Data source information (URI, type, size)	- Data source added
Modify Knowledge Graph	- Elements to modify	- Data modified

Table 60. Graph Database of the functional component of the Di^{KAF} toolkit

4.3.3 Interactions

In the Usage Viewpoint analysis, the following roles were identified:

- **System administrator:** responsible for supervising the smooth operation of the toolkit, adding, or removing eligible users of the toolkit.
- **Data scientist:** person with knowledge on the field of materials, capable of identifying new external data sources that can be incorporated into the toolkit and managing the knowledge graph database schema when updates are necessary.
- **Data consumer:** a stakeholder with interest in accessing the toolkit in order to take advantage of its functionalities, such as a materials scientist or engineer, a researcher in the field of either databases or materials.

4.3.3.1 Get visualization of data

The Data Consumer logs in to the platform by inserting their credentials to the Registration and Log in Validator component and requests through the GUI to visualize specific data stored in the platform. The request may be in the form of a query or a free text description of the desired data. In the latter case, the Data Handling mechanism performs NLP and data processing techniques to transform the request into an appropriate query interpretable from the Graph Database. The query is then directed to the Graph Database and the retrieved data is transmitted to the Data Handling mechanism to generate visual representation of the data along with visualization options. The results are finally, forwarded to the Data Consumer through the GUI.

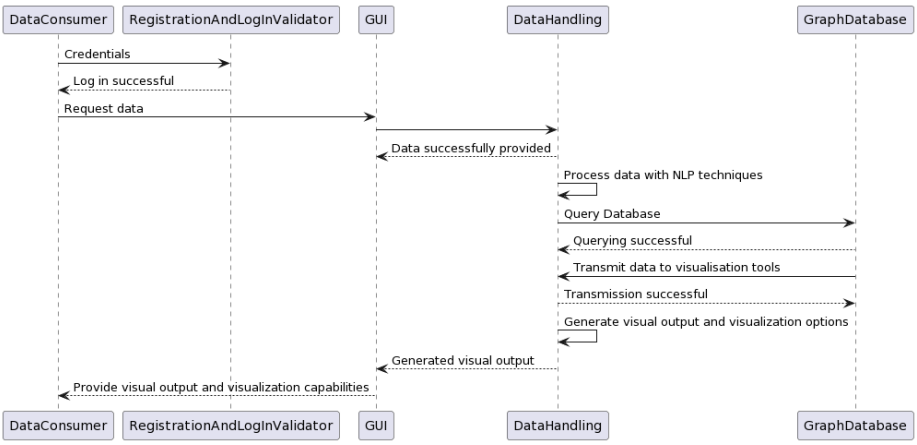


Figure 66: Sequence diagram of the Di^{KAF} toolkit component "Get visualization of data"

4.3.3.2 Add new data source

The “add new data source” activity is an activity that is executed by the data scientist role. The data scientist first logs in the platform and then selects from the GUI the option to add a new data source (to enrich the structured knowledge, update with recent experimental results, etc.). The user specifies the dataset information by detailing properties such as its location (online repository, locally stored, etc.), its size, type (e.g., RDF, CSV, JSON, text). Then, the data handling mechanism performs data preprocessing techniques if these are deemed necessary, for example NLP techniques in the case of free text. Then, the necessary query to update the KG with the new info originating from the new dataset is formed and directed to the knowledge graph database. Finally, the data scientist is informed about the successful addition of the new dataset.

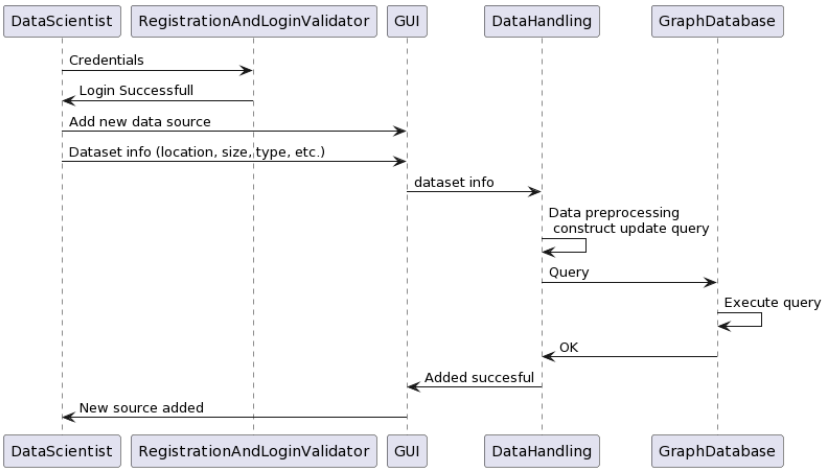


Figure 67: Sequence diagram of the Di^{KAF} toolkit component "Add new data source"

4.3.3.3 Modify Knowledge Graph

In this activity, the data scientist, after logging in with the corresponding credentials, selects from the GUI the option to modify the knowledge graph (KG) and provides the relevant query through a form provided by the GUI. The GUI connects to the data handling module which in turn passes the query to the graph database where it is executed. Then, messages about the successful update are passed through the components, reaching the data scientist.

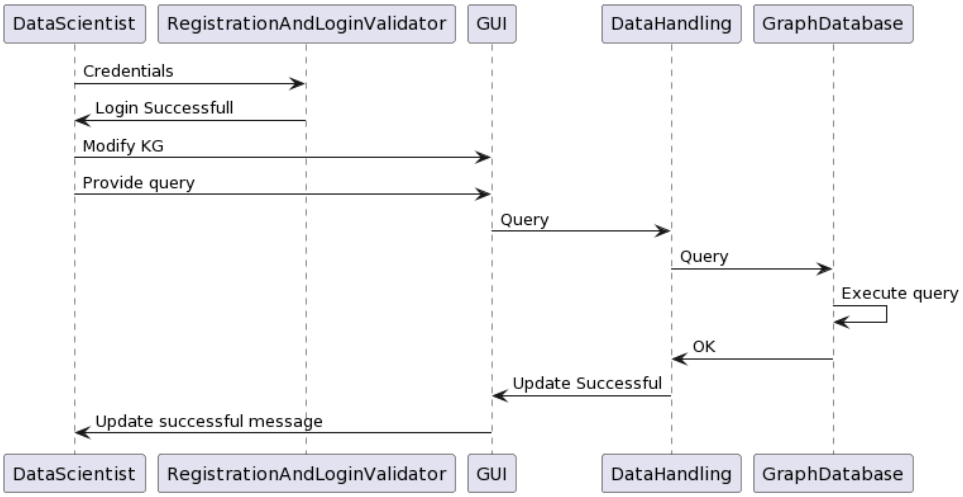


Figure 68: Sequence diagram of the Di^{KAF} toolkit component "Modify Knowledge Graph"

4.3.3.4 Create account

The Data Consumer or the Data Scientist provide their registration information to the Registration and Log in Validator and their request is forwarded to the System Administrator to approve the registration. The user is then informed about their successful registration to the platform and is granted access to the toolkit's functionalities. Optionally, the user could request access to data population services which should be then approved or not by the System Administrator through the Validator, following the same approach with the registration procedure.

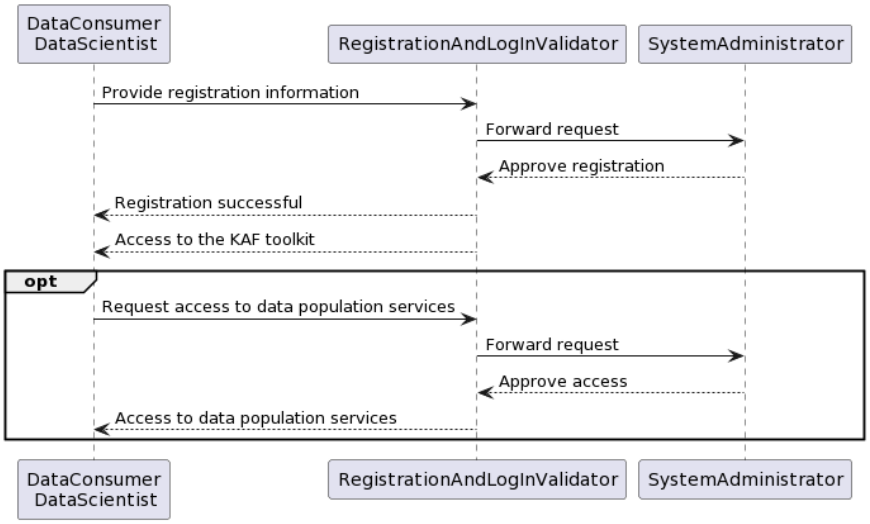


Figure 69: Sequence diagram of the Di^{KAF} toolkit component "Create account"

4.3.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
Registration and Log in Validator	*Application*	*Logic & Rules*
Data handling	*Application*	*API + Logic & Rules*
Graph Database	*Information*	*Data*

Table 61. Mapping of Functional components of Di^{KAF} toolkit with IIRA functional domain

4.4 DIMAT MATERIALS ENVIRONMENTAL AND COST LIFE CYCLE ASSESSMENT (DI$^{MEC\cdot LCA}$)

4.4.1 General Architecture

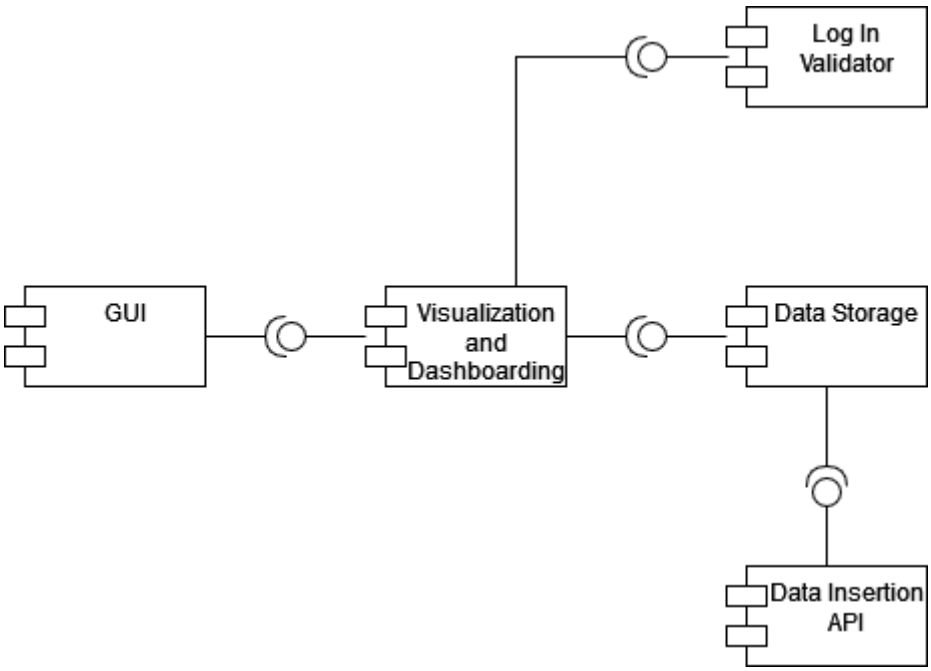


Figure 70: Materials Environmental and Cost Life Cycle Assessment Diagram

This diagram illustrates how a user interacts with a system through a **Graphical User Interface (GUI)**, which is connected to a **Visualization and Dashboarding** component. The **Visualization and Dashboarding** module acts as a central hub where users can view and interact with data in a visual format. This module is connected to three main components: the **Log In Validator**, which ensures that only authorized users can access the system; the **Data Storage** module, which stores the relevant data; and a **Data Insertion API**, which facilitates the addition of new data into the system.

The **Log in Validator** is responsible for managing user authentication and security. Once validated, users can access the **Visualization and Dashboarding** module, where data is retrieved from **Data Storage** and presented visually. Additionally, the **Data Insertion API** allows for the input of new data into the system, which is then stored and can be accessed through the dashboard.

4.4.2 Functional Components

The tables below describe the main functional components identified in the General Architecture of the MEC-LCA toolkit.

4.4.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Visualization and Dashboarding tool.*	
Interface		
Operation	***Inputs***	***Outputs***
Visualize LCA results	- LCA/LCC data	- LCA/LCC data visualized (charts and tables)
Visualize KPIs	- KPI data	- KPI data visualized (charts and tables)
Download data	- Data to download	- Data file (CSV, JSON or similar)
Log in	- Credentials	- Access is granted
Modify visualization	- Modifications selected	- Visualization modified

Table 62. GUI of the functional component of the Di$^{MEC-LCA}$

4.4.2.2 Visualization and dashboarding

Name:	*Visualization and Dashboarding*	
Description:	*It is a data visualization tool that displays dashboards, charts, and graphs to present data.*	
Interface		
Operation	***Inputs***	***Outputs***

Visualize LCA results	- LCA/LCC data	- LCA/LCC data visualized (charts and tables)
Visualize KPIs	- KPI data	- KPI data visualized (charts and tables)
Download data	- Data to download	- Data file (CSV, JSON or similar)
Log in	- Credentials	- Access is granted
Modify visualization	- Modifications selected	- Visualization modified

Table 63. Visualization and dashboarding of the functional component of the Di$^{MEC-LCA}$

4.4.2.3 Log in validator

Name:	*Log In Validator*	
Description:	*It checks the validity of log in credentials provided by a user attempting to log in.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted

Table 64. Login in validator of the functional component of the Di$^{MEC-LCA}$

4.4.2.4 Data storage

Name:	*Data Storage*	
Description:	*It stores and manages data via direct upload of files to the system.*	
Interface		
Operation	***Inputs***	***Outputs***

Store user input	- Data to store (environmental and cost LCA results, KPI measurements)	- Data is stored

Table 65. Data storage of the functional component of the Di$^{MEC-LCA}$

4.4.2.5 Data insertion API

Name:	*Data Insertion API*	
Description:	*It inserts data into the MySQL database.*	
Interface		
Operation	***Inputs***	***Outputs***
Store user input	- Data to store (environmental and cost LCA results, KPI measurements)	- Data inserted into the database

Table 66. Data insertion API of the functional component of the Di$^{MEC-LCA}$

4.4.3 Interactions

The relevant roles identified in the usage viewpoint for this toolkit include:

- **Data consumer (Manager and Decision Makers Team):** In charge of defining business strategy and goals for technological solutions.
- **Data analyst (Research, Engineering and Product Design Team):** In charge of data analysis and creating data products to extract value and gather insights from the data about ongoing processes in order to assist decision-making.
- **Product engineer (Research, Engineering and Product Design Team):** In charge of products development and defining product requirements.
- **Administrator (IT team):** In this context, the solution provider (developers, maintainers), in charge of the code, configuration and making the toolkit available to users.

4.4.3.1 Store user input

The Administrator interacts with the Data Insertion API (Laravel) to input data. The Data Insertion API captures the data imported by the Administrator and sends it to the Data Storage, to be stored. The Data Insertion API confirms the successful storage of the data in the Data Storage, generating a user-friendly output.

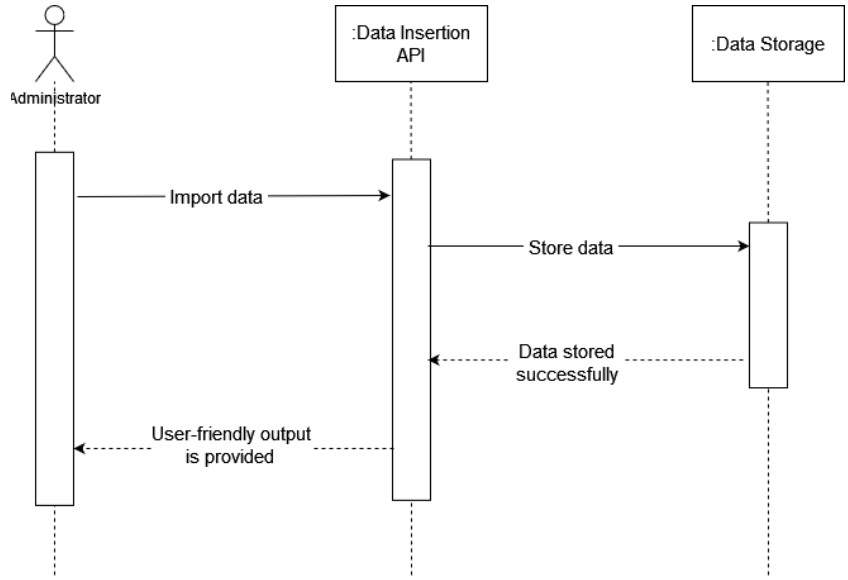


Figure 71: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Store user information"

Alternatively, the Administrator can also directly upload files and visualize the data through Grafana via a plugin without the need for additional database operation.

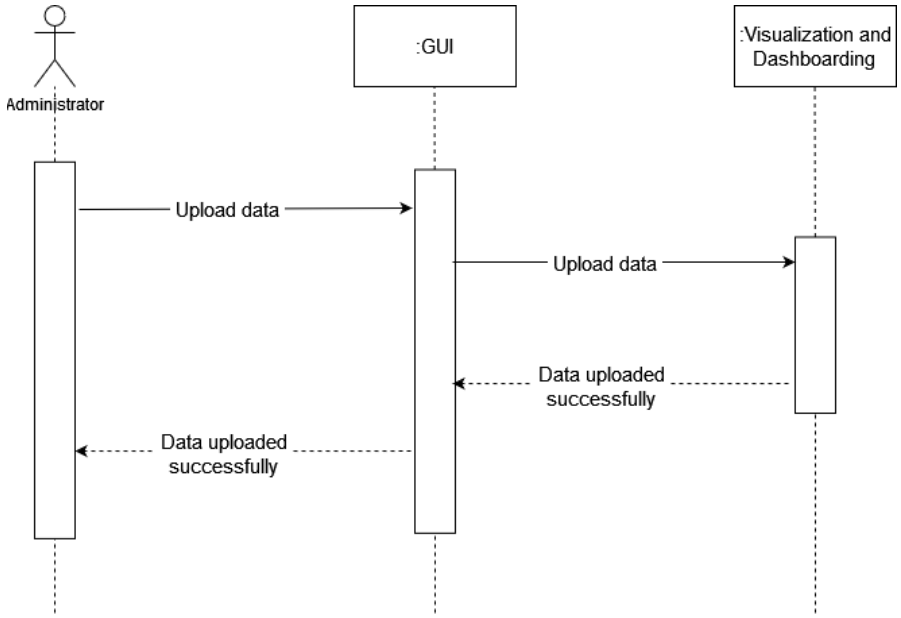


Figure 72: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Store user information 2"

4.4.3.2 Visualize LCA Results

The Data Consumer, the Product Engineer and the Data Analyst interact with the GUI to access the data visualization functionality and select specific data to visualize related LCA results. Initially, the Data Consumer, the Product Engineer and the Data Analyst submit their credentials via GUI to the Visualization and Dashboarding tool. The Log in Validator captures the provided credentials through the Visualization and Dashboarding tool and grants access. The GUI then captures the selected data and sends it to the Visualization and Dashboarding tool. The Visualization and Dashboarding tool receive the data selection from the GUI and validates the access. It also retrieves the corresponding LCA results from the Data Storage. The Visualization and Dashboarding tool generate the visual representation of LCA results. The Visualization and Dashboarding tool deliver the visual output to the GUI. The GUI presents the visual output to the Data Consumer, the Product Engineer, and the Data Analyst.

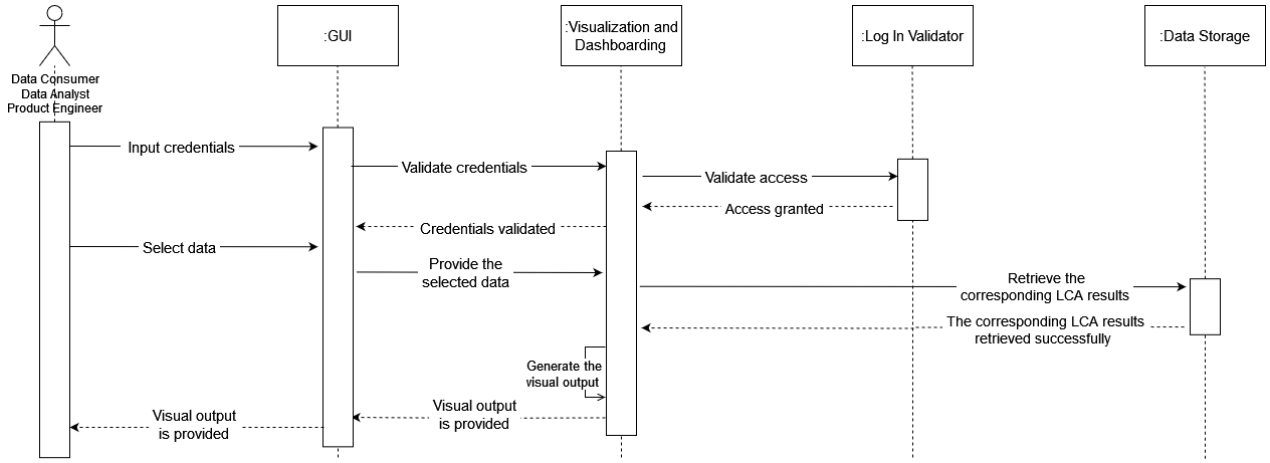


Figure 73: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Visualize LCA Results"

4.4.3.3 Visualize KPIs

For the visualization of the KPIs, the process follows the same steps as above.

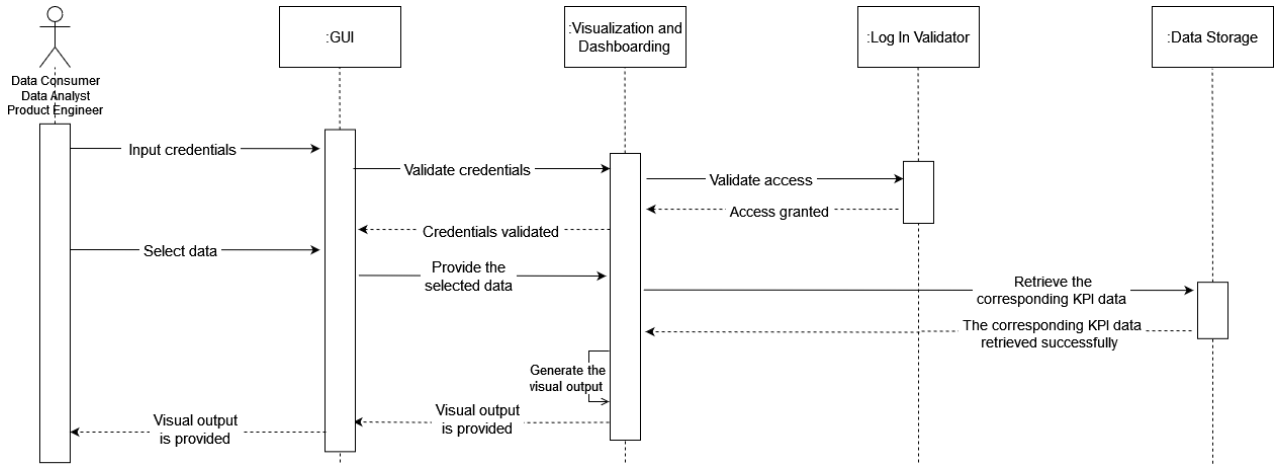


Figure 74: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Visualize KPIs"

4.4.3.4 Download Data

The Administrator, the Data Consumer, the Product Engineer, or the Data Analyst initiates a download request for specific data through the GUI. The Visualization and Dashboarding tool receive the download request. The Visualization and Dashboarding tool initiate the

automatic download process, packaging the selected data into a downloadable format (such as CSV). The Administrator, the Data Consumer, the Product Engineer or the Data Analyst receives the data file.

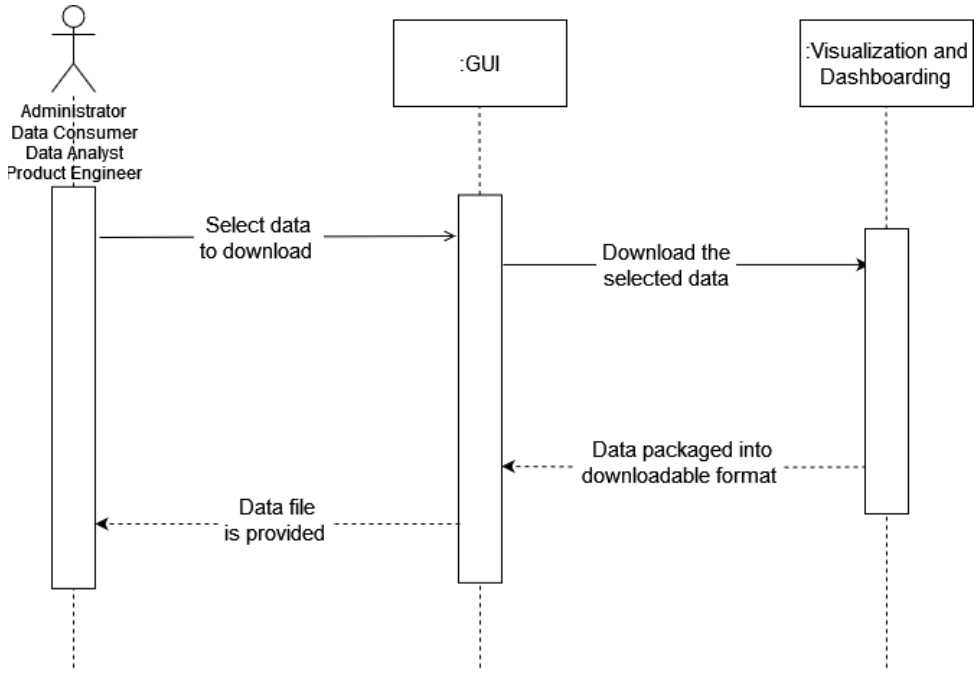


Figure 75: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Download Data"

4.4.3.5 Log in

The Administrator, the Data Consumer, the Product Engineer, or the Data Analyst navigate to the Visualization and Dashboarding tool's log-in page through the GUI, enter their unique credentials, such as username and password, into the designated fields, and submit the log-in information. The Visualization and Dashboarding tool receive the log-in request and verifies the entered credentials through the Log in Validator. If the credentials are valid, the Visualization and Dashboarding tool grants access.

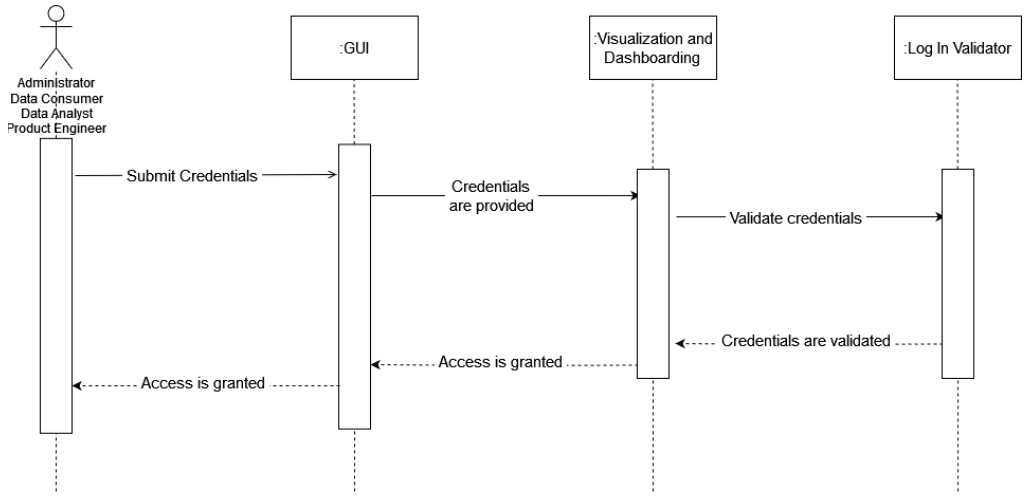


Figure 76: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Log in"

4.4.3.6 Modify Visualization

The Administrator, the Data Consumer, the Product Engineer, or the Data Analyst interact with the GUI that displays the visualization. The GUI provides options and controls for modifying the visualization (such as focusing on a specific variable or range). They select specific modifications to apply. The GUI captures the modification selections. The Visualization and Dashboarding tool receive the modification requests from the GUI and applies the requested modifications to the visualization. The GUI displays the modified visualization.

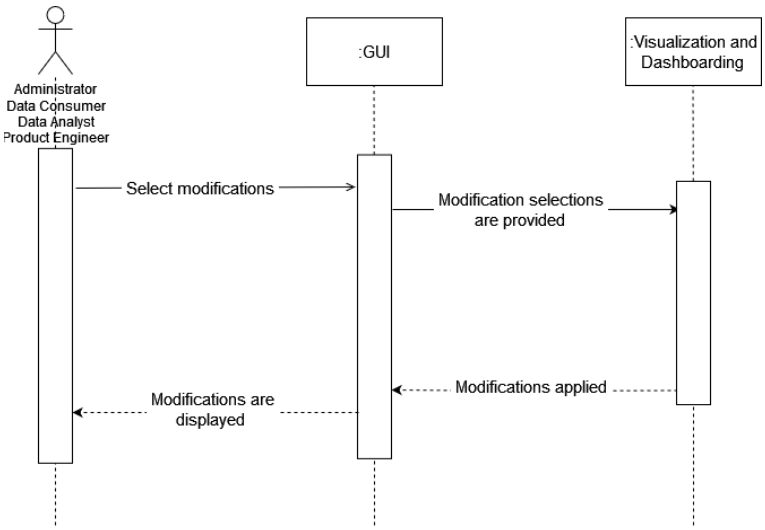


Figure 77: Sequence diagram of the Di$^{MEC-LCA}$ toolkit component "Modify Visualization"

4.4.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
Visualization and Dashboarding	*Application*	*Logic & Rules*
Log In Validator	*Application*	*Logic & Rules*
Data Insertion API	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*

Table 67. Mapping of Functional components of Di$^{MEC-LCA}$ toolkit with IIRA functional domain

4.5 DIMAT MATERIALS DESIGN FRAMEWORK (DI^{MDF})

4.5.1 General architecture

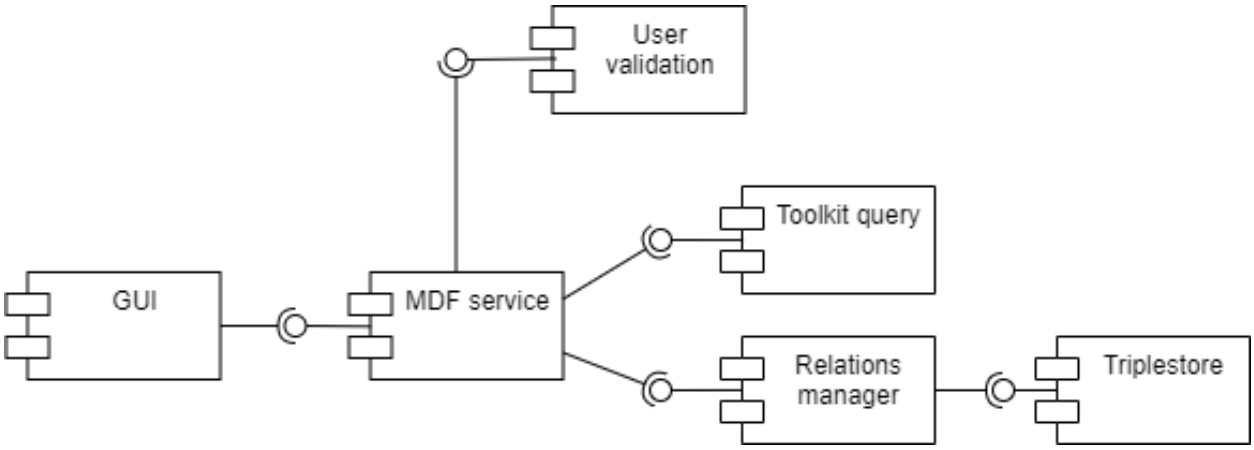


Figure 78: Materials Design Framework diagram

This diagram illustrates a system where the user interacts via a **Graphical User Interface (GUI)**, which is part of the MDF toolkit that acts as a central component managing the flow of data and operations. The process begins with **User Validation** (user authentication), which ensures that only authorized users access the system. Once validated, the system supports interactions such as querying and managing materials modelling relationships. The **Toolkit Query** component handles complex data queries, while the (**Materials**) **Relations Manager** make knowledge about materials relations accessible. Both components interact with a **Triplestore**, a semantics-based data storage for storing and retrieving data including the relationships between data.

4.5.2 Functional components

The tables below present the main functional components of the DiMDF toolkit.

4.5.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Dataspace.*	
Interface		
Operation	***Inputs***	***Outputs***
Search connections	- Query + filters	- Matching results (internal and from other toolkits)
Add connection	- Connecting items	- New connection

Table 68. GUI of the functional component of the Di^{MDF} toolkit

4.5.2.2 MDF service

Name:	*MDF service*	
Description:	*Handles the logic of the service*	
Interface		
Operation	***Inputs***	***Outputs***
Search connections	- Query + filters	- Matching results (internal and from other toolkits)
Add connection	- Connecting items	- New connection

Table 69. MDF Service of the functional component of the Di^{MDF} toolkit

4.5.2.3 User validation

Name:	*User validation*	
Description:	*Checks the users are logged in*	
Interface		
Operation	***Inputs***	***Outputs***
Check user access	- Credentials	- Access is granted

Table 70. User validation of the functional component of the Di^{MDF} toolkit

4.5.2.4 Toolkit query

Name:	*Toolkit query*	
Description:	*Connects to the APIs of other toolkits to collect information*	
Interface		
Operation	***Inputs***	***Outputs***
Fetch information	- Toolkit and data to query	- Data from the toolkit

Table 71. Toolkit query of the functional component of the Di^{MDF} toolkit

4.5.2.5 Relations manager

Name:	*Relations manager*	
Description:	*Stores information on data interconnections among toolkits*	
Interface		
Operation	***Inputs***	***Outputs***
Query connection	- Entity queried	- Related entities and their toolkits

Add connection	- Entities to connect	- Success message

Table 72. Relations manager of the functional component of the Di^{MDF} toolkit

4.5.3 Interactions

4.5.3.1 Search connections

The material designer provides some criteria in the search box of the MDF tool, which is sent to the backend service. Based on the search input, the triple store is searched by the relations manager for relevant toolkits, and the specific records or entities from those toolkits. This index is used for querying the toolkits for their actual information.

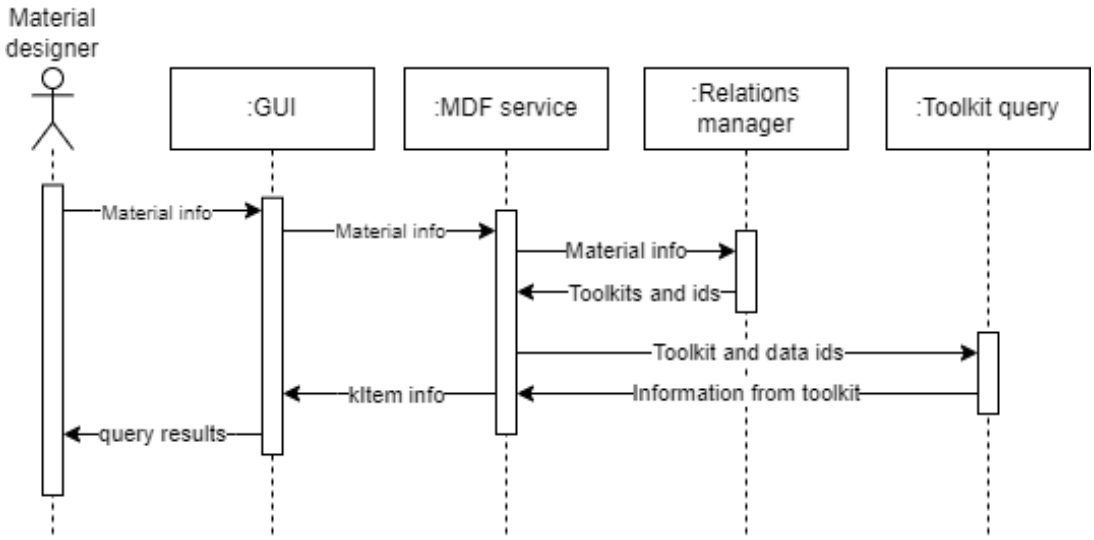


Figure 79: Sequence diagram of the Di^{MDF} toolkit component "Search connection"

4.5.3.2 Add connection

Data scientists can add new connections and relationships between the data in the different toolkits by browsing the records (fetched by the toolkit query) and adding new connections to the triple store via the relations manager.

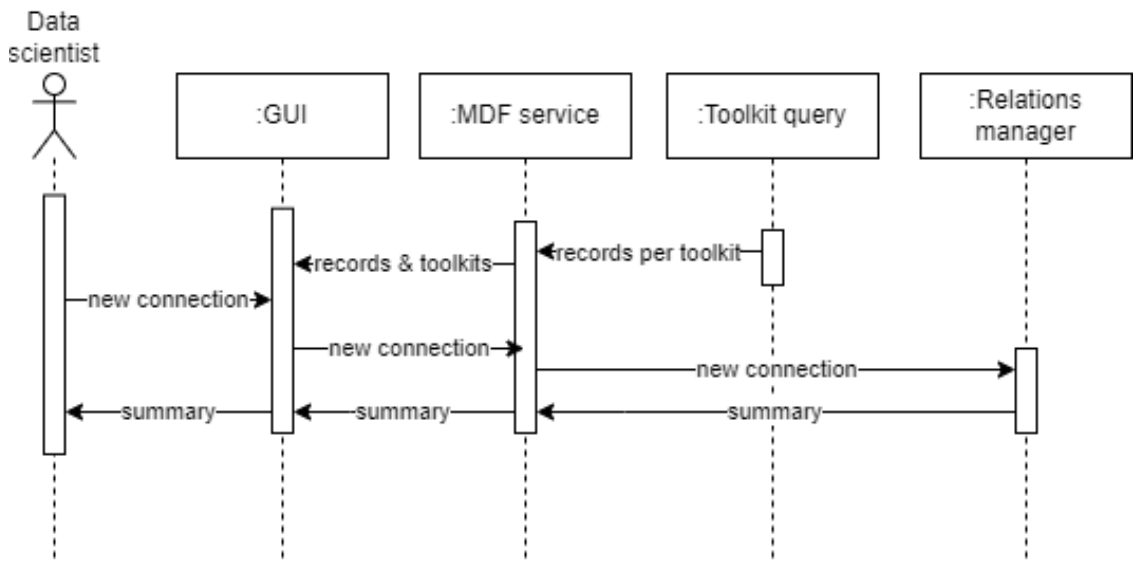


Figure 80: Sequence diagram of the Di^{MDF} toolkit component "Add Connection"

4.5.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
User validation	*Application*	*Logic & Rules*
MDF service	*Application*	*API + Logic & Rules*
Toolkit query	*Application*	*API + Logic & Rules*
Relations manager	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*

Table 73. Mapping of Functional components of Di^{MDF} toolkit with IIRA functional domain

4.6 DIMAT MATERIALS MODELER (DI^{MM})

4.6.1 General Architecture

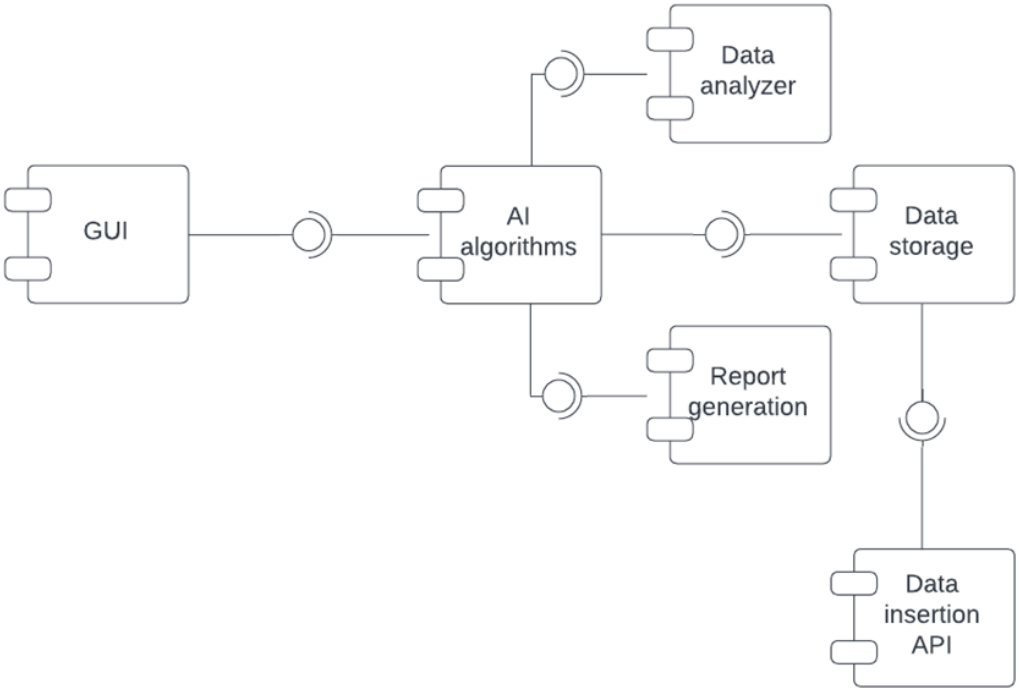


Figure 81: Materials Modeler diagram

This diagram illustrates a system where a user interacts through a **Graphical User Interface (GUI)**, which connects to the core of the system: the **AI algorithms** component. The **AI algorithms** module plays a pivotal role in processing and analyzing data. It is connected to several key components: the **Data Analyzer**, which processes the data to extract insights; **Data Storage**, where the processed data and any relevant information are stored; and the **Report Generation** module, which uses the analyzed data to generate comprehensive reports for the user. Additionally, there is a **Data Insertion API** that allows the system to receive new data, which is then processed by the AI algorithms and stored accordingly.

4.6.2 Functional components

The tables below describe the main functional components identified in the General Architecture of the MM toolkit.

4.6.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Dashboarding tool.*	
Interface		
Operation	***Inputs***	***Outputs***
Download data	- Data to download	- Data file
Log in	- Credentials	- Access is granted

Table 74. GUI of the functional component of the Di^{MM} toolkit

4.6.2.2 AI algorithms

Name:	*AI algorithms*	
Description:	*Various robust AI algorithms (e.g. explainable AI, ML, DL) to analyze data and generate predictions.*	
Interface		
Operation	***Inputs***	***Outputs***
Generate predictions	- Data to analyze	- Predicted results
Evaluate performance	- Prediction results	- Performance metrics
Adapt to new data	- New data	- Updated models

Table 75. AI algorithms of the functional component of the Di^{MM} toolkit

4.6.2.3 Data Analyzer

Name:	*Data analyzer*	
Description:	*Analyzes data to identify patterns and relationships.*	
Interface		
Operation	***Inputs***	***Outputs***
Analyze data	- Data to analyze	- Data analyzed
Visualize data	- Data to visualize	- Data visualized
Download data	- Data to download	- Data file

Table 76. Data analyzer of the functional component of the Di^{MM} toolkit

4.6.2.4 Data storage

Name:	*Data Storage*	
Description:	*It implements data storing and manages data in a structured and organized manner.*	
Interface		
Operation	***Inputs***	***Outputs***
Store data	- Data to store	- Data is stored
Retrieve data	- Query	- Corresponding data

Table 77. data storage of the functional component of the Di^{MM} toolkit

4.6.2.5 Data insertion API

Name:	*Data Insertion API*	
Description:	*It inserts data to the data storage.*	
Interface		
Operation	***Inputs***	***Outputs***
Store data	- Data to store	- User-friendly output

Table 78. Data insertion API of the functional component of the Di^{MM} toolkit

4.6.3 Interactions

The relevant roles identified in the usage viewpoint for this toolkit include:

- **Data analyst (Research, Engineering and Product Design Team):** In charge of analyzing data and developing AI models.
- **Product engineer (Research, Engineering and Product Design Team):** Defines and adjusts the parameters for the manufacturing process based on model predictions.
- **Administrator (IT team):** Ensures the system and data infrastructure are functioning properly and securely.

4.6.3.1 Data storage and management

The Administrator interacts with the Data Insertion API to input data. The Data Insertion API captures the data imported by the administrator and sends it to the Data Storage to be stored. The Data Insertion API confirms the successful storage of the data in the Data Storage, generating a user-friendly output.

4.6.3.2 AI algorithms

The Data Analyst develops the AI algorithms based on the data input from Product Engineers. The data is stored in the data storage through the Data insertion API. The AI algorithms adapt to new data and the models are updated.

4.6.3.3 Data analysis

The Data Analyst interacts with the Data Analyzer to conduct various analyses of the stored data. The results of the analyses are used to identify patterns and relationships. The Data Analyzer provides a visualization of the analyzed data to aid the Data Analyst in understanding the data patterns.

4.6.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
AI algorithms	*Application*	*Logic & Rules*
Data Analyzer	*Application*	*Logic & Rules*
Data Storage	*Information*	*Data*
Data Insertion API	*Application*	*Logic & Rules*

Table 79. Mapping of Functional components of Di^{MM} toolkit with IIRA functional domain

4.7 DIMAT MATERIALS DESIGNER (DI^{MD})

4.7.1 General architecture

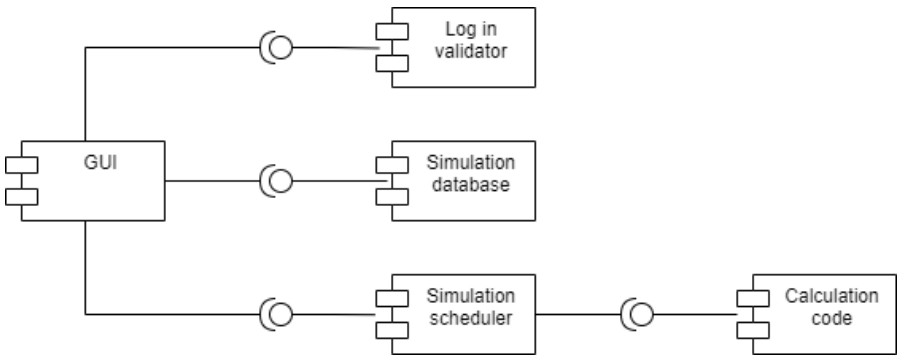


Figure 82: Materials Designer diagram

This diagram depicts a system where the user interacts through a **Graphical User Interface (GUI)**. The GUI is connected to three primary components. First, the **Log in Validator** ensures that only authorized users gain access to the system by verifying credentials. Second, the **Simulation Database** manages, and stores data related to simulations, ensuring that the system has access to the necessary information. Third, the **Simulation Scheduler** is responsible for managing and coordinating simulation tasks, and interacts with

the **Calculation Code** module sending all the necessary data for the required simulations. Finally, the **Calculation Code** executes the actual computational tasks.

4.7.2 Functional components

In this section the main functional components identified in the General Architecture of the MD toolkit are described.

4.7.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users input data and visualize results*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- User credentials	- Access is granted
Define a new material	- Material components data	- Mechanical properties of the new material

Table 80. GUI of the functional component of the Di^{MD} toolkit

4.7.2.2 Log in validator

Name:	*Log in validator*	
Description:	*Grant access to the toolkit usage*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- User credentials	- Access is granted

Table 81. Log in validator of the functional component of the Di^{MD} toolkit

4.7.2.3 Simulation database

Name:	*Simulation database*	
Description:	*Stores the materials definitions and their calculated properties*	
Interface		
Operation	***Inputs***	***Outputs***
Define new material	- Description of the new material	- Properties of the new material

Table 82. Simulation database of the functional component of the Di^{MD} toolkit

4.7.2.4 Simulation scheduler

Name:	*Simulation scheduler*	
Description:	*Prepares the input file for the execution of the virtual tests*	
Interface		
Operation	***Inputs***	***Outputs***
Define new material	- Description of the new material to be simulated	- Properties of the new material

Table 83. Simulation scheduler of the functional component of the Di^{MD} toolkit

4.7.2.5 Calculation code

Name:	*Calculation code*	
Description:	*Executes virtual test to calculate the properties of non-homogeneous materials*	
Interface		
Operation	***Inputs***	***Outputs***
Define new material	- Input file for the calculation	- Output file of the calculation

Table 84. Calculation code of the functional component of the Di^{MD} toolkit

4.7.3 Interactions

The usage viewpoint for this toolkit has identified the following relevant role:

- **Simulation engineer**, who uses the DiMAT MD toolkit to calculate the mechanical and thermal properties of a non-homogenous and complex materials defining as inputs the base components with all their properties and the microstructure of the new material.

4.7.3.1 Log in

The user inputs his/her credentials into the GUI. Data are sent to the log-in validator which, if the credentials are valid, grant the access to the toolkit to the user.

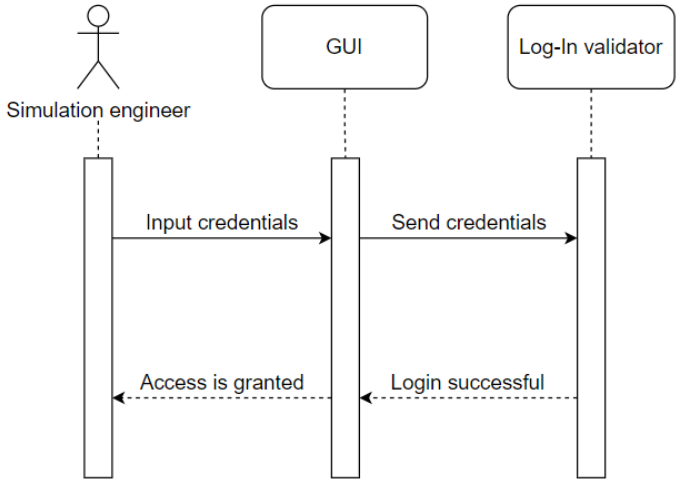


Figure 83: Sequence diagram of the Di^{MD} toolkit component "Log In"

4.7.3.2 Define new material

The user defines, through the provided graphical interface, the definition of the new non-homogeneous material in terms of its base components, their properties, and the desired internal microstructure. From the user interface, this data is provided to the internal simulation database which define a new record for the material and send all the compositional information to the simulation scheduler. This component oversees the communication with the calculation code; based on the material to be characterized, it prepares the necessary input file and send it to the calculation code, calling for its execution. When the calculation carried out by the calculation code ends, the output data is retrieved by the simulation scheduler which extract the requested data and stores them into the database. From the database, the properties of the new material can be finally consulted by the user.

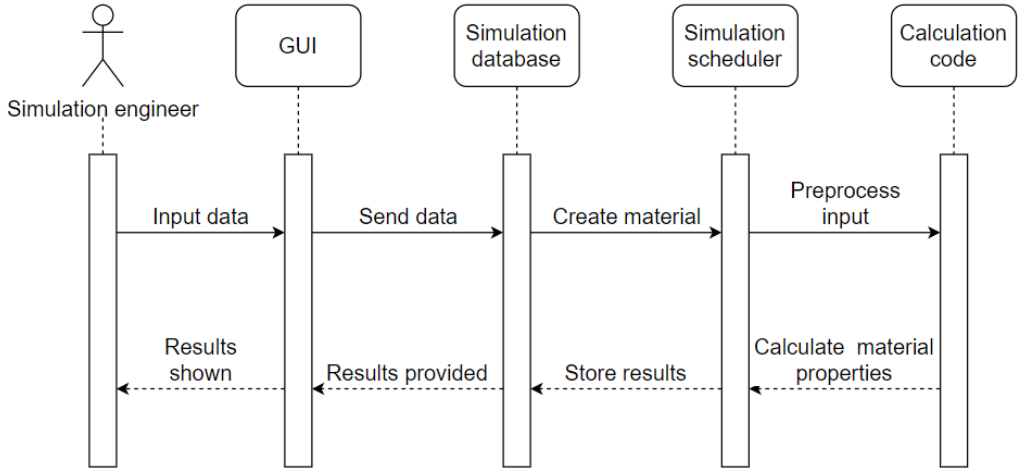


Figure 84: Sequence diagram of the Di^{MD} toolkit component "Define new material"

4.7.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*API & UI*
Log in validator	*Application*	*Logic & Rules*
Simulation database	*Information*	*Data Management*
Simulation scheduler	*Application*	*Logic & Rules*
Calculation code	*Application*	*Logic & Rules*

Table 85. Mapping of Functional components of Di^{MD} toolkit with IIRA functional domain

4.8 DIMAT MATERIALS MECHANICAL PROPERTIES SIMULATOR (DI^{MMS})

4.8.1 General Architecture

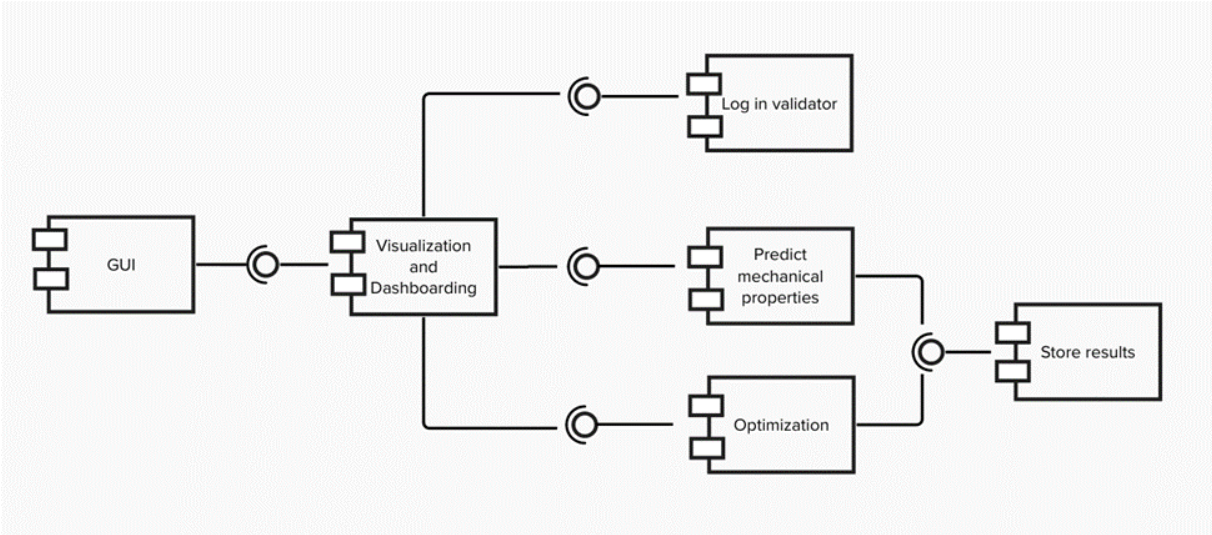


Figure 85: Materials Mechanical Properties Simulator diagram

This diagram represents a system where users interact via a **Graphical User Interface (GUI)**, which is linked to a **Visualization and Dashboarding** module. This central module allows users to visually interact with and manage data and system processes. The **Visualization and Dashboarding** component connect to several key functionalities: the **Log in Validator**, which ensures that only authorized users can access the system; the **Predict Mechanical Properties** module, which handles simulations or analyses to predict physical or mechanical characteristics of materials or components; and the **Optimization** module, responsible for refining or improving the results or processes. The outcomes of these predictions and optimizations are sent to the **Store Results** module, which securely stores the final data for later access or analysis.

Additionally, the Predict Mechanical Properties module will be linked to the CMDB toolkit to retrieve the initial material data from the material database.

4.8.2 Functional components

The tables below describe the main functional components identified in the General Architecture of the MMS toolkit.

4.8.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Visualization and Dashboarding tool.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted
Predict MMS properties	- Microstructure parameters and compound properties	- Mechanical properties
Optimize MMS properties	- Target mechanical properties	- Optimizes parameter
Store simulation results	- Data to store	- Data file

Table 86. GUI of the functional component of the Di^{MMS} toolkit

4.8.2.2 Visualization and dashboarding

Name:	*Visualization and Dashboarding*	
Description:	*It is a data visualization tool that displays dashboards, charts, and graphs to present data.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted
Predict MMS properties	- Microstructure parameters and compound properties	- Mechanical properties

Optimize MMS properties	- Target mechanical properties	- Optimize parameter

Table 87. Visualization and dashboarding of the functional component of the Di^{MMS} toolkit

4.8.2.3 Log in validator

Name:	*Log In Validator*	
Description:	*It checks the validity of log in credentials provided by a user attempting to log in.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted

Table 88. Log in validator of the functional component of the Di^{MMS} toolkit

4.8.2.4 Predict Mechanical Properties API

Name:	*Predict mechanical properties API*	
Description:	*Determine the mechanical properties associated to a microstructure*	
Interface		
Operation	***Inputs***	***Outputs***
Predict mechanical properties	- Microstructure parameters and compound properties	- Mechanical properties
Store simulation results	- Data to store	- Data file

Table 89. Predict Mechanical Properties API of the functional component of the Di^{MMS} toolkit

4.8.2.5 Optimization API

Name:	*Optimization API*	
Description:	*Optimize microstructure parameters*	
Interface		
Operation	***Inputs***	***Outputs***
Optimization	- Mechanical properties	- Microstructure parameters
Store simulation results	- Data to store	- Data file

Table 90. Optimization API of the functional component of the Di^{MMS} toolkit

4.8.2.6 Data storage

Name:	*Data Storage*	
Description:	*It implements data storing and manages data in a structured and organized manner.*	
Interface		
Operation	***Inputs***	***Outputs***
Store user input	- Data to store	- Data is stored
Predict mechanical properties	- Data to store	- Corresponding data
Optimization	- Data to store	- Corresponding data

Table 91. Data Storage of the functional component of the Di^{MMS} toolkit

4.8.3 Interactions

The usage viewpoint for this toolkit has identified the following relevant roles:

- **Administrator (IT team):** the Administrator is the solution provider responsible for managing the code, configurations, and ensuring the toolkit's availability for users.

- **Manufacturing engineer (Manager and Decision Makers Team).**
- **Material engineer (Research, Engineering and Product Design Team).**
- **Software developer (Research, Engineering and Product Design Team):** In charge of toolkit development, defining modelling workflows to predict mechanical properties regarding the input variables, performing the physical simulation, and constructing the surrogated models.

4.8.3.1 Log in

The Administrator, the Manufacturing engineer, the Material analyst, or the Software developer navigate to the Visualization and Dashboarding tool's log-in page through the GUI, enter their unique credentials and submit the log-in information. The tool processes the log-in request and verifies the entered credentials through the Log In Validator. If the credentials are valid, the Visualization and Dashboarding tool grants access, enabling users to utilize its functionalities.

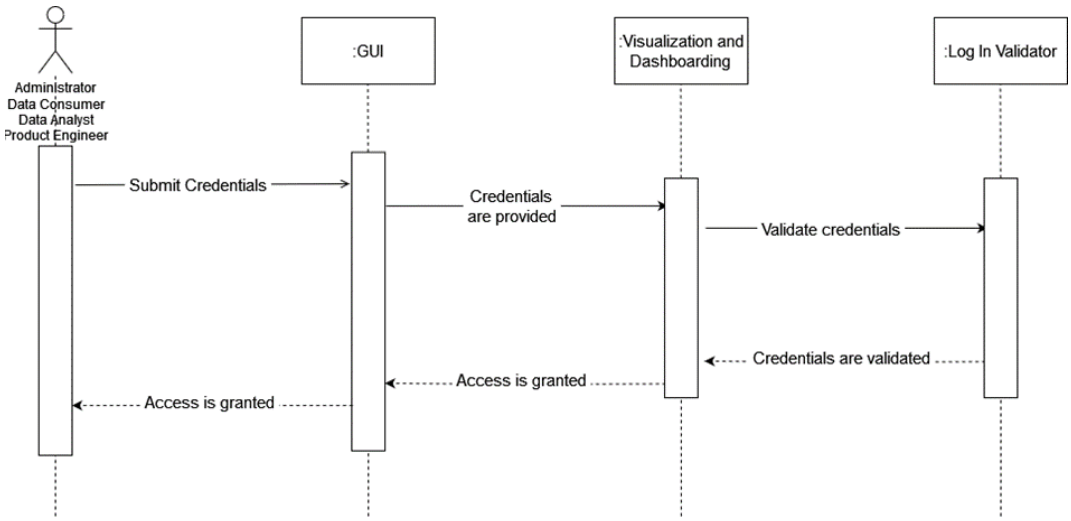


Figure 86: Sequence diagram of the Di^{MMS} toolkit component "Log in"

4.8.3.2 Predict mechanical properties.

The manufacturing engineer and the material engineer interact with the GUI to access the prediction of mechanical properties functionality. They input access data, the microstructure data, and the properties of the compounds. The GUI captures the input and sends this data to the Visualization and Dashboarding tool. The Visualization and Dashboarding tool receive

the input from the GUI, validates the access and send the information to the Prediction of Mechanical Properties API. The visualization and dashboarding tool also retrieve the corresponding mechanical properties results. The Prediction of Mechanical Properties tool send data to the Store Results.

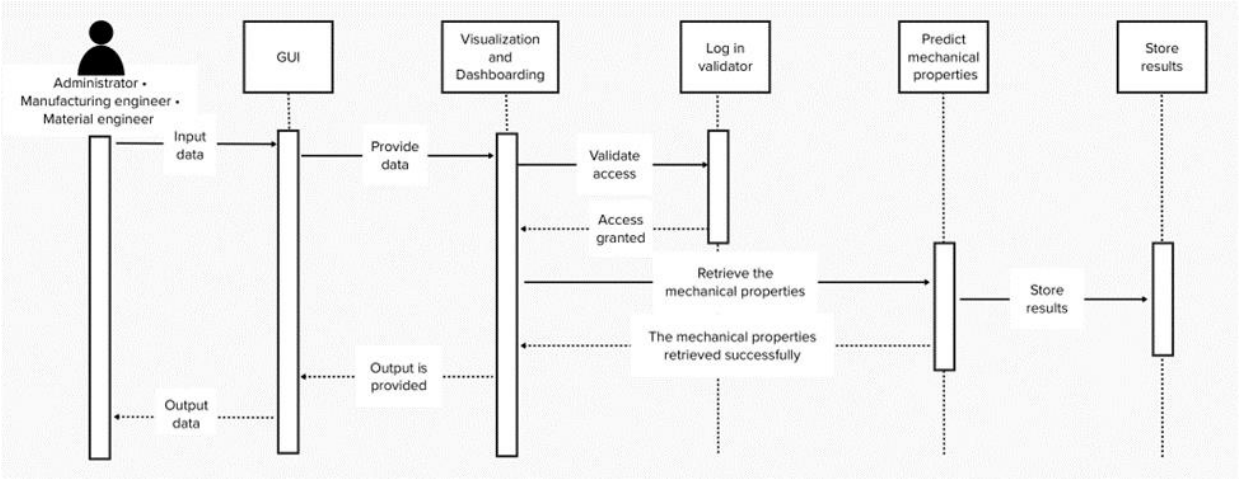


Figure 87: Sequence diagram of the Di^{MMS} toolkit component "Predict Mechanical Properties"

4.8.3.3 Optimization

The manufacturing engineer and the material engineer interact with the GUI to access the optimization functionality. They input access data and the targeted mechanical property. microstructure data, and the properties of the compounds. The GUI captures the input and sends this data to the Visualization and Dashboarding tool. The Visualization and Dashboarding tool receive the input from the GUI, validates the access and send the information to the Optimization API. The visualization and dashboarding tool also retrieve the corresponding optimum microstructural parameters. The optimum values are sent to the Store Results tool.

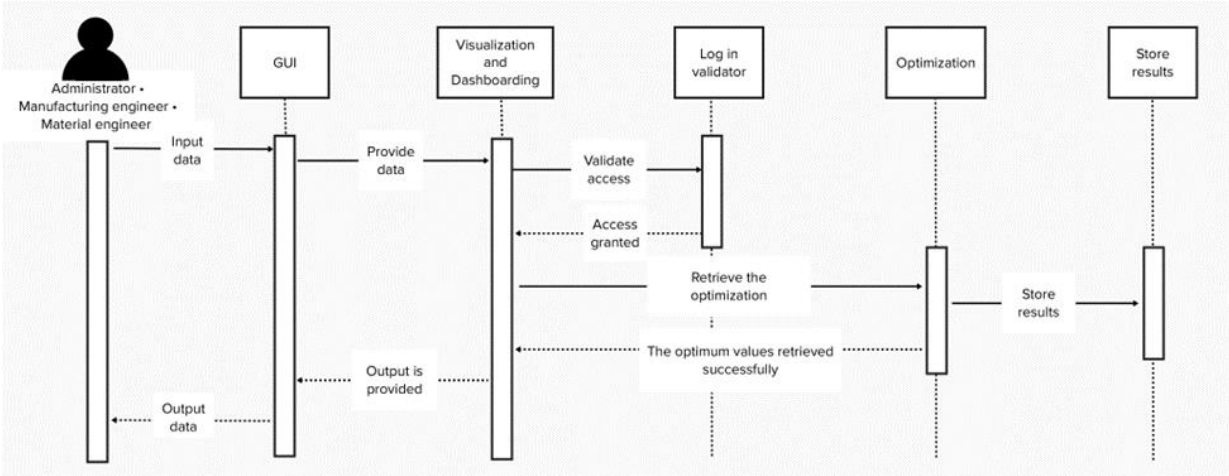


Figure 88: Sequence diagram of the Di^{MMS} toolkit component "Optimization"

4.8.3.4 Store user input and output

The Prediction of Mechanical Properties API and the Optimization API interact with the Store results tool. The results of the APIs in combination with the input are sent to the Data Storage, to be stored. The Store Result tool confirms the successful storage of the data.

4.8.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
Visualization and Dashboarding	*Application*	*Logic & Rules*
Log In Validator	*Application*	*Logic & Rules*
Predict mechanical properties API	*Application*	*API + Logic & Rules*
Optimization API	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*

Table 92. Mapping of Functional components of Di^{MMS} toolkit with IIRA functional domain

4.9 DIMAT MATERIALS PROCESSING SIMULATOR (DI^{MPS})

4.9.1 General Architecture

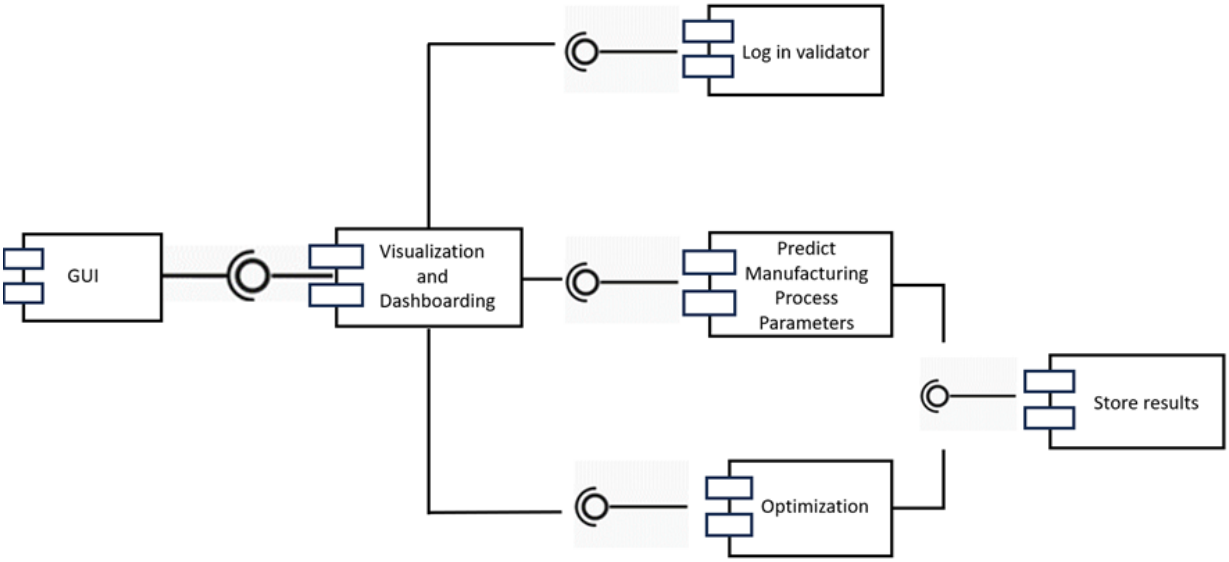


Figure 89: Materials Processing Simulator diagram

This diagram outlines a system where users interact via a **Graphical User Interface (GUI)**, which is connected to a **Visualization and Dashboarding** module. This module serves as the central interface unit for users to manage data and processes within the system. The **Visualization and Dashboarding** component connect to several key elements: the **Log in Validator**, which ensures that only authorized users can access the system; the **Predict Manufacturing Process Parameters** module, which analyses and predicts parameters related to manufacturing processes; and the **Optimization** module, which is responsible for refining or improving the processes outputs. Finally, the results from the predictions and optimizations are stored in the **Store Results** module, ensuring that all generated data is securely saved for future access.

4.9.2 Functional components

The tables below describe the main functional components identified in the General Architecture of the MPS toolkit.

4.9.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Visualization and Dashboarding tool.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted
Predict MPS parameters	- Materials properties and boundary conditions of manufacturing process	- Manufacturing process parameters and materials behaviour
Optimize MPS parameters	- Target process parameters and materials behaviour	- Optimizes parameter
Store simulation results	- Data to store	- Data file

Table 93. GUI of the functional component of the Di^{MPS} toolkit

4.9.2.2 Visualization and dashboarding

Name:	*Visualization and Dashboarding*	
Description:	*It is a data visualization tool that displays dashboards, charts, and graphs to present data.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted
Predict MPS parameters	- Materials properties and boundary conditions	- Manufacturing process parameters and materials behaviour

	of manufacturing process	
Optimize MPS parameters	- Target process parameters and materials behaviour	- Optimize parameter

Table 94. Visualization and dashboarding of the functional component of the Di^{MPS} toolkit

4.9.2.3 Log in validator

Name:	*Log In Validator*	
Description:	*It checks the validity of log in credentials provided by a user attempting to log in.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted

Table 95. Log in validator of the functional component of the Di^{MPS} toolkit

4.9.2.4 Predict Manufacturing Process Parameters

Name:	*Predict manufacturing process parameters*	
Description:	*Determine the manufacturing process parameters associated to type and properties of materials*	
Interface		
Operation	***Inputs***	***Outputs***
Predict manufacturing process parameters	- Materials properties and boundary conditions of manufacturing process	- Manufacturing process parameters and materials behaviour
Store simulation results	- Data to store	- Data file

Table 96. Predict Manufacturing Process Parameters of the functional component of the Di^{MPS} toolkit

4.9.2.5 Optimization Results

Name:	*Optimization Results*	
Description:	*Optimize manufacturing process parameters*	
Interface		
Operation	***Inputs***	***Outputs***
Optimization	- Process parameters and materials behaviour	- Optimize parameter
Store simulation results	- Data to store	- Data file

Table 97. Optimization Results of the functional component of the Di^{MPS} toolkit

4.9.2.6 Data Storage

Name:	*Data Storage*	
Description:	*It implements data storing and manages data in a structured and organized manner.*	
Interface		
Operation	***Inputs***	***Outputs***
Store user input	- Data to store	- Data is stored
Predict manufacturing process parameters	- Data to store	- Corresponding data
Optimization	- Data to store	- Corresponding data

Table 98. Data storage of the functional component of the Di^{MPS} toolkit

4.9.3 Interactions

The usage viewpoint for this toolkit has identified the following relevant roles:

- **System Administrator:** manage the toolkit system, create new users and assign credentials and keep up-to-date the toolkit database.
- **Manufacturing Engineering:** Production and maintenance management
- **Materials Scientists:** Research, Engineering Production and Product Design

4.9.3.1 Log in

System Administrator, Manufacturing engineer or Material scientists navigate to the Visualization and Dashboarding tool's log-in page through the GUI, enter their unique credentials and submit the log-in information. The tool processes the log-in request and verifies the entered credentials through the Log in Validator. If the credentials are valid, the Visualization and Dashboarding tool grants access, enabling users to utilize its functionalities.

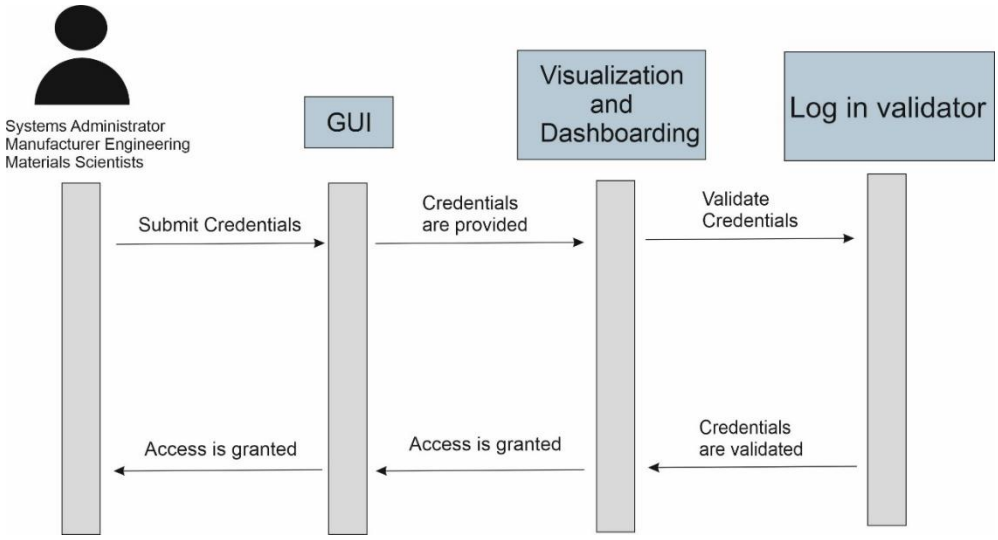


Figure 90: Sequence diagram of the Di^{MPS} toolkit component "Log in"

4.9.3.2 Predict Manufacturing Process Parameters

The manufacturing engineer and the material scientists interact with the GUI to access the prediction of manufacturing process parameters. They input access data, the material properties, and the boundary conditions of manufacturing process. The GUI captures the input and sends this data to the Visualization and Dashboarding tool. The Visualization and Dashboarding tool receive the input from the GUI, validates the access and sends the information to the Prediction of Manufacturing Process Parameters. The visualization and

dashboarding tool also retrieve corresponding process parameters results. The Manufacturing Process Parameters tool sends data to the Store Result.

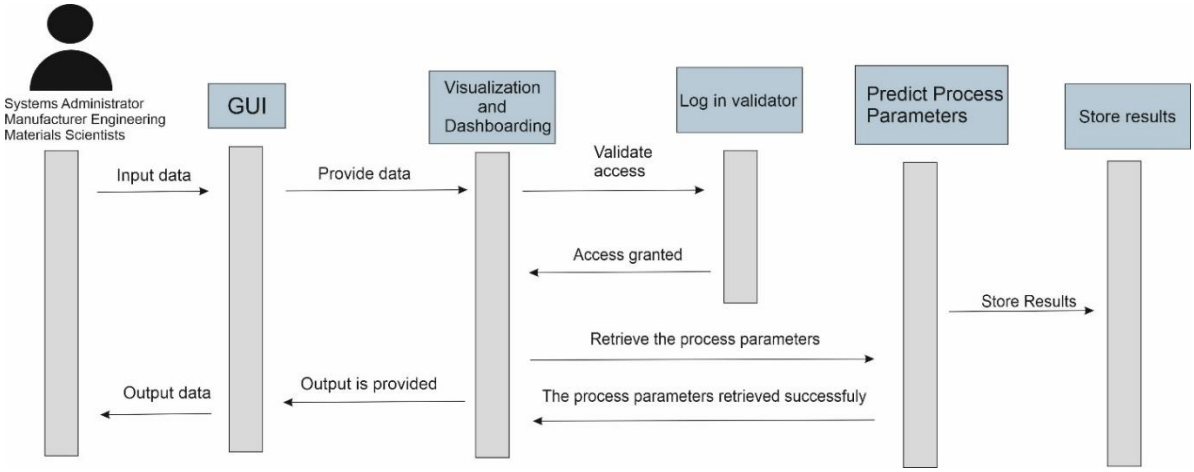


Figure 91: Sequence diagram of the Di^{MPS} toolkit component "Predict Manufacturing Process Parameters"

4.9.3.3 Optimization

The manufacturing engineer and the material scientist interact with the GUI to access the optimization functionality. They input access data and the targeted process parameters and materials behavior. The GUI captures the input and sends this data to the Visualization and Dashboarding tool. The Visualization and Dashboarding tool receive the input from the GUI, validates the access and send the information to the Optimization API. The visualization and dashboarding tool also retrieve the corresponding optimum process parameters. The optimum values are sent to the Store Results tool.

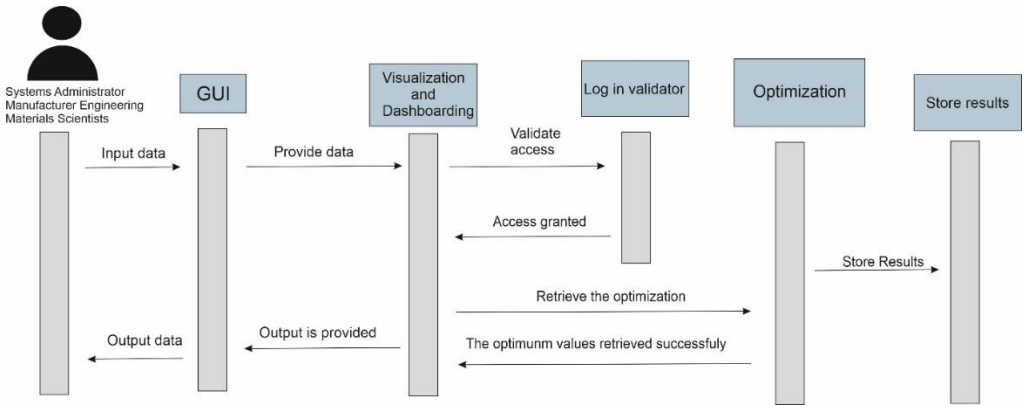


Figure 92: Sequence diagram of the Di^{MPS} toolkit component "Optimization"

4.9.3.4 Store user input and output

The Prediction Manufacturing of Process Parameters and Optimization interacts with the Store results tool. The obtained simulation results in combination with the input are sent to the Data Storage to be stored. The Store Result tool confirms the successful storage of the data.

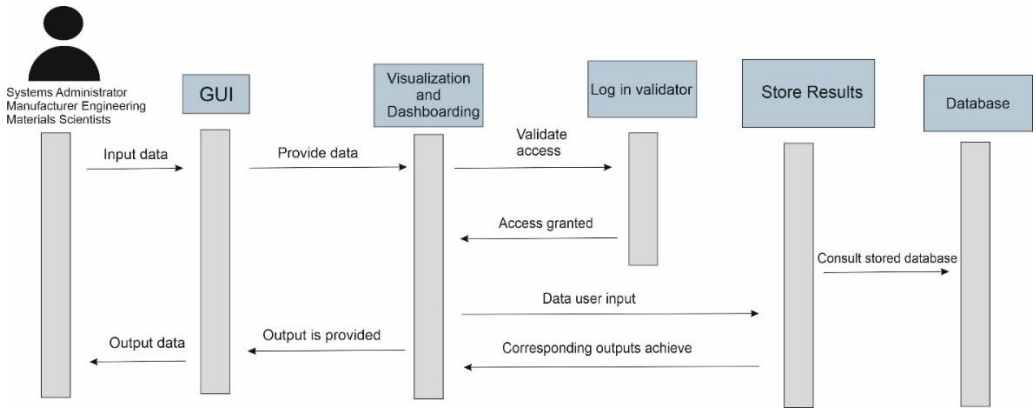


Figure 93: Sequence diagram of the Di^{MPS} toolkit component "Store user input and output"

4.9.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
Visualization and Dashboarding	*Application*	*Logic & Rules*
Log In Validator	*Application*	*Logic & Rules*
Predict manufacturing process parameters	*Application*	*API + Logic & Rules*
Optimization	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*

Table 99. Mapping of Functional components of Di^{MPS} toolkit with IIRA functional domain

4.10 DIMAT DIGITAL TWIN FOR PROCESS CONTROL (DI^{DTPC})

4.10.1 General Architecture

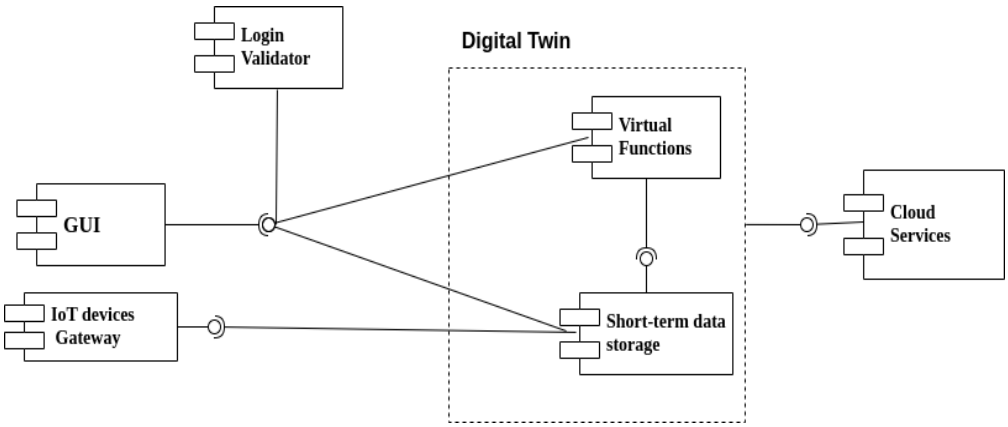


Figure 94: Digital Twin for Process Control diagram

This diagram represents a system where the user interacts through a Graphical User Interface (GUI), which in turn is connected to both the login validator and the main module of the Digital Twin (DT). The process begins with the Login validator for the purpose of authentication, which ensures that only authorized users can access the system. Once validated, the users can interact with the core of the system the Digital Twin, which is a virtual representation that mirrors the real-time state and functionality of the physical system. The GUI offers a number of services related to the Digital Twin such as enforcement of commands on the physical equipment, viewing of the DT's status, monitoring of the DT's operations and time-series data analysis. This core component interacts with the IoT devices Gateway module which is responsible for aggregating measurements from sensors/IoT boards on the physical equipment and enforcing actions to the physical twin through communication with actuators. The Virtual Functions component of the DTPC toolkit provides access to functions such as materials simulation software, future-state prediction and anomaly detection. The results of these virtual functions as well as time series data originating from the IoT devices can be stored in the Short-term Data Storage component for quick access and processing. Additionally, the system can interface with Cloud Services for long-term storage or advanced processing, enabling scalability and remote accessibility.

4.10.2 Functional components

The tables below present the main functional components of the DTPC toolkit.

4.10.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the DTPC toolkit.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted
Register Device	- Device id, IP address, transmission protocols, data types	- Device is registered
Send command to device	- Command type, command parameters	- Action is executed in the Physical Twin
View Digital Twin Status	- Device id	- Data stored and visualized
Monitor DTPC operation	- Device id to check	- Alerts, notifications, data reports
Time-series data analysis	- Virtual function, dataset for analysis	- Data stored and visualized

Table 100. GUI of the functional component of the Di^{DTPC} toolkit

4.10.2.2 Login Validator

Name:	*Log In Validator*	
Description:	*It checks the validity of log in credentials provided by a user attempting to log in.*	
Interface		
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted

Table 101. Login validation of the functional component of the DiDTPC toolkit

4.10.2.3 IoT device Gateway

Name:	*IoT device Gateway*	
Description:	*Responsible for aggregating measurements from sensors/IoT boards and enforcing actions to the physical twin through communication with actuators*	
Interface		
Operation	***Inputs***	***Outputs***
Register Device	- Device id, IP address, transmission protocols, data types	- Device is registered
Send command to device	- Command type, command parameters	- Action is executed in the Physical Twin
View Digital Twin Status	- Device id	- Data stored and visualized
Monitor DTPC operation	- Device id to check	- Alerts, notifications, data reports
Time-series data analysis	- Virtual function, dataset for analysis	- Data stored and visualized

Table 102. IoT device Gateway of the functional component of the Di^{DTPC} toolkit

4.10.2.4 Short-term Data Storage

Name:	*Short-term data storage*	
Description:	*A database suitable for storing time series data originating from the IoT devices as well as results of virtual functions*	
Interface		
Operation	***Inputs***	***Outputs***
View Digital Twin Status	- Device id	- Data stored and visualized

Monitor DTPC operation	- Device id to check	- Alerts, notifications, data reports
Time-series data analysis	- Virtual function, dataset for analysis	- Data stored and visualized

Table 103. Short-term Data Storage of the functional component of the Di^{DTPC} toolkit

4.10.2.5 Cloud services

Name:	*Cloud services*	
Description:	*Services running on cloud-based infrastructure, providing long term storage and access to functions that require high processing capabilities.*	
Interface		
Operation	***Inputs***	***Outputs***
Time-series data analysis	- Virtual function, dataset for analysis	- Data stored and visualized

Table 104. Cloud service of the functional component of the Di^{DTPC} toolkit

4.10.2.6 Virtual Functions

Name:	*Virtual Functions*	
Description:	*Provides access to functions such as materials simulation software, future-state prediction, anomaly detection, etc.*	
Interface		
Operation	***Inputs***	***Outputs***
Time-series data analysis	- Selected virtual function, dataset for analysis	- Data stored and visualized

Table 105. Virtual Functions of the functional component of the Di^{DTPC} toolkit

4.10.3 Interactions

In the Usage Viewpoint analysis, the following roles were identified:

- **System administrator:** responsible for the continuous operation of the DTPC toolkit, can add new equipment to be "twinned" to the platform and perform health-checks, also responsible for addressing security concerns.
- **Data scientist:** responsible for analyzing the collected data that are fed to the DTPC via the appropriate functions and reaching conclusions about the manufacturing processes.
- **Materials engineer:** expert on the operation of the equipment and the employed materials, responsible for sending explicit commands through the DTPC to the physical equipment, when necessary, can recommend simulation software to be used.
- **Applications developer:** responsible for developing applications that utilize the digital twins managed in the toolkit.

4.10.3.1 Send Command to Device

The Materials Engineer or the Applications Developer log in to the platform by providing their credentials to the Log in Validator component. Through the GUI, they select the command along with the required parameters to enforce an action to a device or a set of devices. This request is transformed into an appropriate data format by the Virtual Functions component and this information is passed from the GUI to the IoT Gateway which establishes the connection with the corresponding device(s). Then, the action is enforced on the physical equipment though an IoT actuator according to the user's command and the user is informed about the successful operation. Finally, the IoT Gateway transmits data concerning the new current state of the physical equipment to the short-term data storage to be stored.

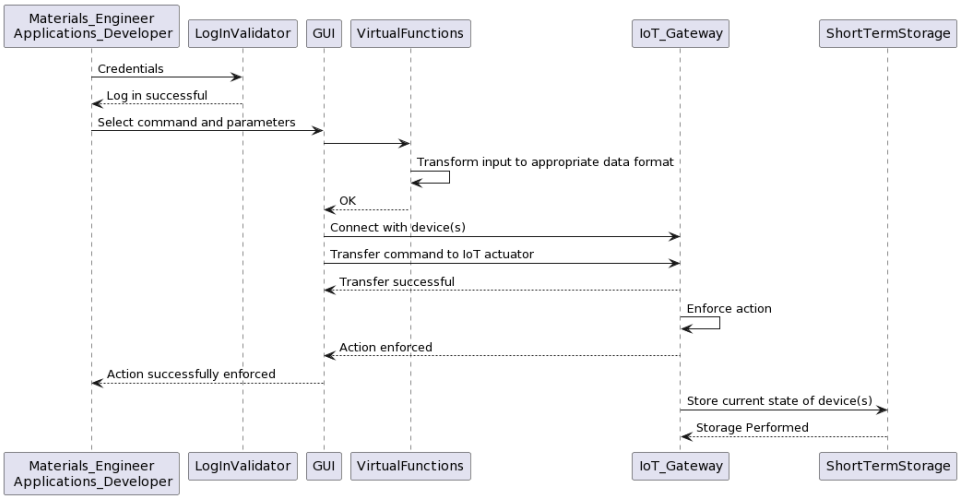


Figure 95: Sequence diagram of the Di^{DTPC} toolkit component "Send Command to Device"

4.10.3.2 View Digital Twin Status

Users from all the defined roles of the toolkit insert their credentials to the Log in Validator and through the toolkit's GUI, request for the current state of the device(s). The request is forwarded to the IoT Gateway, connection with the device(s) is established and the corresponding data of the equipment is fed back to the GUI and displayed to the user in a user-friendly form. Finally, this data is also stored in the short-term data storage, through the IoT Gateway, in the appropriate form.

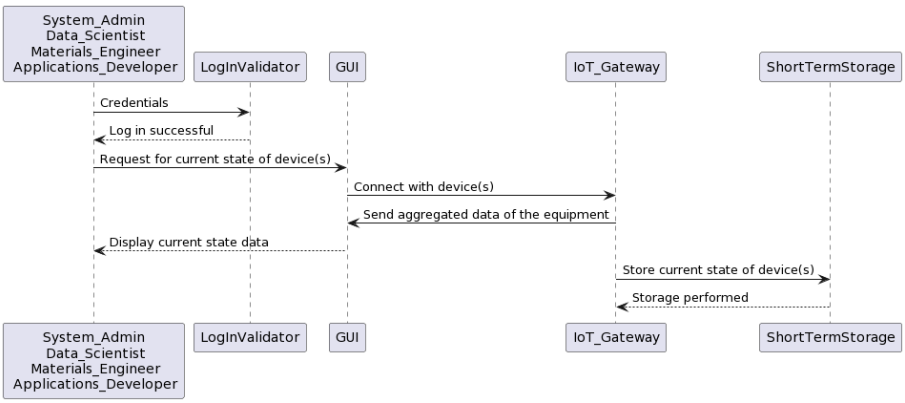


Figure 96: Sequence diagram of the Di^{DTPC} toolkit component "View Digital Twin Status"

4.10.3.3 Device Registration

For the device registration, first the user must be logged in with the role "System Administrator". After this is ensured, the system administrator selects from the GUI the "Add device" option and fills the provided forms with necessary information such as the device id, its IP address, the type of data provided by the device, etc. Then, this information is passed from the GUI to the IoT Gateway that is the module responsible for communicating with the IoT devices. The gateway tries to establish the connection to the device using the specified information and after it succeeds it passes an "OK" message to the GUI which is transformed to an appropriate message to the user informing them about the successful addition of the device. Then, the IoT Gateway begins storing data originating from the new device to the short-term storage database.

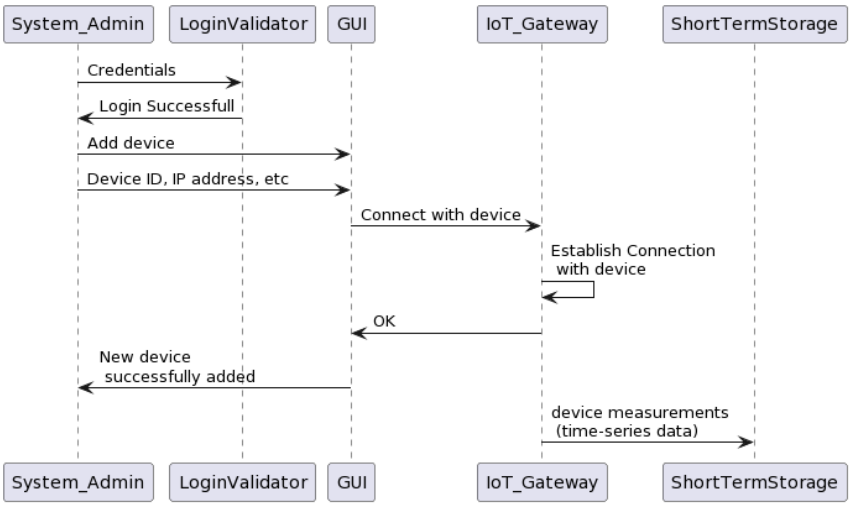


Figure 97: Sequence diagram of the Di^{DTPC} toolkit component "Device Registration"

4.10.3.4 Monitor DTPC operation

In this activity, the System Administrator after logging in the platform, selects from the menu displayed in the GUI, the option allowing them to monitor the DTPC. Then, the GUI connects to the short-term storage in order to retrieve notifications and display them to the user. The administrator may choose to follow more steps depending on the existence and the content of any notifications. Then, the GUI directs the IoT Gateway to connect with the devices associated with the DTPC. After, connections are established the measurements are fed to appropriate virtual functions that analyze the system behavior. For these to operate, more data may be required and for this purpose the short-term storage sends such data to the corresponding virtual functions. After the data analysis is completed, the results are

displayed through the GUI to the system administrator. If the administrator wishes, they can opt to configure alarms through the GUI and the new information is stored in the database.

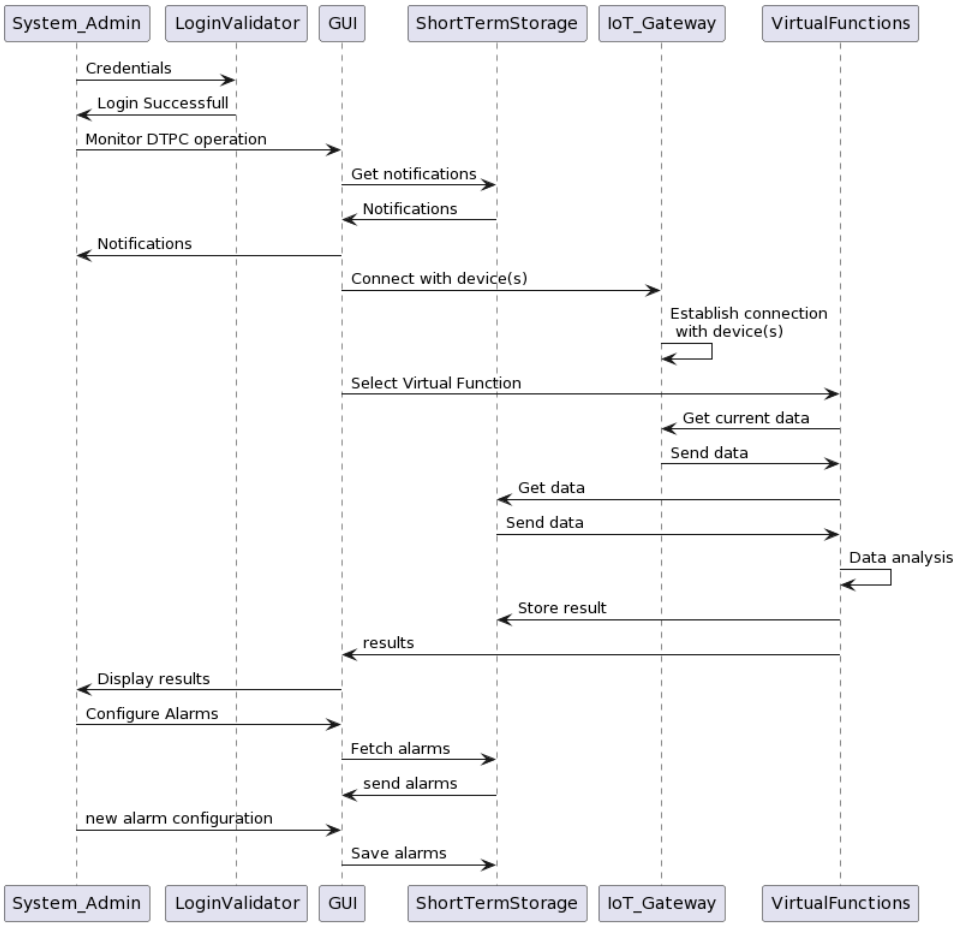


Figure 98: Sequence diagram of the Di^{DTPC} toolkit component "Monitor DTPC operation"

4.10.3.5 Time Series Data Analysis

The following UML diagram details the interaction of components for the “time-series data analysis” activity which is an abstraction of any Virtual Function call (materials or manufacturing simulation process, anomaly detection, future-state prediction etc.). The role responsible for executing this scenario is the “data scientist” that needs to be logged in the platform. Through the displayed GUI the data scientist selects which virtual function is needed. Then, the GUI connects with the Virtual Functions module, that in turn requests data both from the Short-term storage and the Cloud services (if necessary). Then, the function is

executed, and the results are displayed through the GUI, and they are stored in the short-term database.

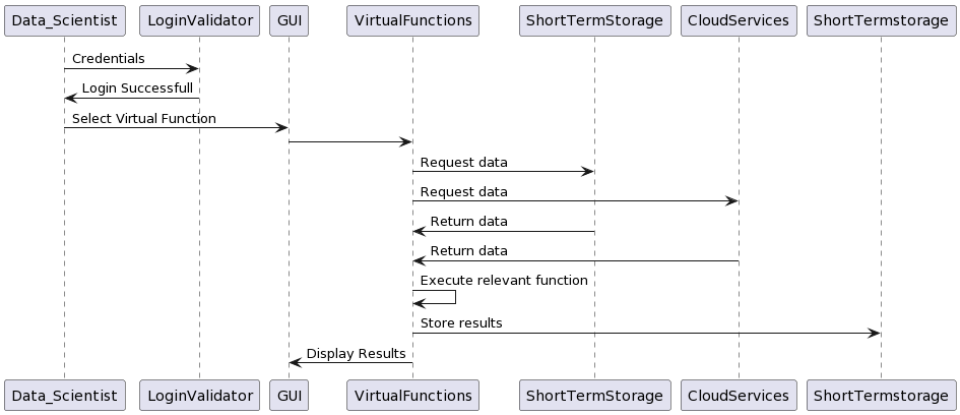


Figure 99: Sequence diagram of the Di^{DTPC} toolkit component "Time Series data Analysis"

4.10.4 Mapping of functional components to IIRA functional domain

Component	*Domain*	*Subdomain*
GUI	*Application*	*UI*
Log In Validator	*Application*	*Logic & Rules*
IoT Gateway	*Application*	*API + Logic & Rules*
Virtual Functions	*Application*	*API + Logic & Rules*
Cloud services	*Application*	*API + Logic & Rules*
Short-term data storage	*Information*	*Data*

Table 106. Mapping of Functional components of Di^{DTPC} toolkit with IIRA functional domain

5 IMPLEMENTATION VIEWPOINT

This task will describe, technically, the different components of the DiMAT Architecture. To describe the interconnection between the different components of the solutions and to define the technologies to be used for their development. To define the interconnection and communication of the solutions, the Application Programming Interfaces (API) will be specified. On the other hand, it is important to stress that this viewpoint will consider all the data obtained from the different viewpoints. From the business layer to the activities identified in use, including the functional information of these solutions. In this sense, the

Edge Computing, Microservice Applications and Function as a Service (FaaS) platforms are fundamental for the implementation of the DiMAT toolsets described in WP4, WP5 and WP6.

5.1 IMPLEMENTATION VIEWPOINT APPROACH FOR DIMAT

5.1.1 Implementation Viewpoint Concepts

The implementation viewpoint focuses on representing the technical part that affects the implementation of the solutions, considering both the technologies and the IoT systems. It also considers the implementation of the previously defined components and functionalities, as well as the activities defined in the previous viewpoints. Key facets of the implementation viewpoint include:

- **General Architecture:** It presents the distribution and configuration of the various components in DiMAT solutions. It also provides information on the relationships between the components and the chosen technology.
- **Technical Component Particulars**: It provides a complete overview of the components that make up the system, including interfaces, protocols, behaviour, and other relevant properties.
- **Implementation Mapping:** It provides a map to understand the interaction between the activities studied in the Usage Viewpoint and the components defined in the Functional Viewpoint. This visualization makes it easier to interpret how the system works.
- **System Characteristics:** Define the characteristics required for correctly implementing the system solutions.

The Implementation viewpoint in DiMAT is based on the Industrial Internet Reference Architecture (IIRA), ensuring that the technical representation of the system can effectively deliver the functions and activities stipulated by the Usage and Functional perspectives. In addition, it considers factors such as cost, time-to-market, business strategy, regulatory compliance, and the evolving technology landscape, all aligned with the guidance provided by the business perspective. Furthermore, the Implementation Viewpoint will furnish an intricate architectural framework grounded in cloud computing paradigms. Within this context, Edge computing, Microservice applications, and Function as a Service (FaaS) platforms play pivotal roles in realizing the software deliverables outlined within this document.

5.1.2 Implementation Viewpoint Methodology

The methodology to complete the task was defined with the following steps:

- Firstly, a diagram was developed to provide an internal picture of the solution's architecture.
- Secondly, a complete analysis of the component-level requirements of the solution will be developed. Aspects at the hardware level, data storage and a final section on data analysis will be considered. In this analysis section, the data flow, pre-processing, and final analysis will be analysed in detail.
- Thirdly, the technologies to develop the solution will be defined. The selection of the solutions follows the basic methodology presented in the D2.2 deliverable. The major attributes to be considered when selecting are the openness of the tools/libraries, the active communities, available documentation and processing requirements.
- Finally, a roadmap is drawn up to facilitate the implementation of the solutions.

5.2 DIMAT CLOUD MATERIALS DATABASE (DI^{CMDB})

5.2.1 Internal Architecture of Toolkits

The CMDB offers a homogeneous storage solution for heterogeneous data. Relational, graph and object storage solutions are available and orchestrated via a central backend. Users (both human and machines) interact with the platform via the GUI or the Python SDK, with the user authentication component checking their identity. Internal data processing tools allow the pre-processing of the data to extract relevant knowledge prior to its storage.

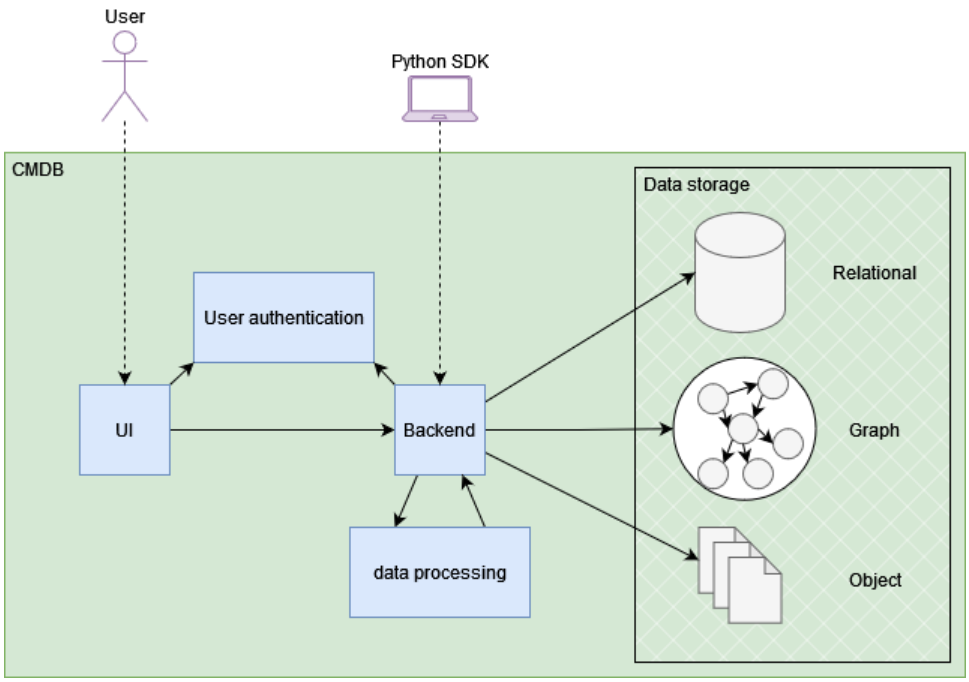


Figure 100: cloud materials database Component Map

5.2.2 Required components

5.2.2.1 Hardware Components

The only hardware requirement for deployment of the toolkit is an internet reachable server with enough available resources for handling the requests and sufficient storage for all the datasets that might be sent to the CMDB.

5.2.2.2 Data Storage

The core functionality of this toolkit is data storage. Three types of databases are required:

- **Relational:** Tabular data extracted from the processed uploaded data
- **Graph:** semantic data and metadata extracted from the raw file or defined by users
- **Object storage:** Storage of experimental raw files and other datasets

5.2.2.3 Data Analysis

5.2.2.3.1 Data Flows

- **External data to CMDB:** The external data uploaded is mainly made up of raw files. These will be pre-processed to extract meaningful data and metadata for easier access.
- **Direct queries to CMBD:** The stored data can be queried and browsed via the GUI.
- **CMDB to other toolkits:** Data stored on the CMDB will be relevant for other toolkits, such as KAF or MDF.

Data flows among components			Manner of transmission
Component 1	**Directionality**	**Component 2**	
External data	One directional	CMDB backend	APIs, HTTP
CMDB frontend (GUI)	Bi-directional	CMDB backend	HTTP
CMDB backend	Bi-directional	Other toolkits	APIs, HTTP

Table 107. Data Flow of Di^{CMDB}

5.2.2.3.2 Data Preprocessing

Most data stored to the CMDB will be in the form of raw data files. Based on the type, pre-processing can be carried out to identify the most relevant values. Additionally, certain metadata, such as operator or machine information can be identified and linked to other, previously uploaded data. These pre-processed data will be stored on the relational and graph databases.

5.2.2.3.3 Data Analytics

Similar to how data pre-processing can be done via Python scripts, further functionality to visualize and plot relevant data is included. The Python SDK library will also enable data exploration and extraction locally for users.

5.2.3 Technologies employed

The CMDB will be based on the Dataspace Management System (DSMS) developed in Fraunhofer IWM. The DSMS is a web platform that provides a novel approach for managing data in a smart way. This software solution enables the system to understand user and materials data and offers opportunities for automation. Users can write and publish apps and processing scripts that can access the stored data as well as semantic representation of the data via a friendly graphical user interface (GUI), REST API or Python SDK. The DSMS has been developed using standard technologies in Python and Angular. For data storage, tools like PostgreSQL can be employed as it handles large datasets and high workloads efficiently and it supports complex queries, joins, and subqueries. Additionally, Elasticsearch can be employed to support analyses of large datasets, both of these toolkits were examined as parts of the evaluated technologies mentioned in Deliverable 2.2.

5.2.4 Implementation maps

- **Data upload:** The raw data coming for instance from a machine in a lab, is uploaded via the GUI, and additional metadata inputted in the upload form. This is sent to the backend, which may carry out additional data processing operations to extract relevant information from within the raw files. All data is then stored in the different databases, with unique identifiers to ensure consistency.

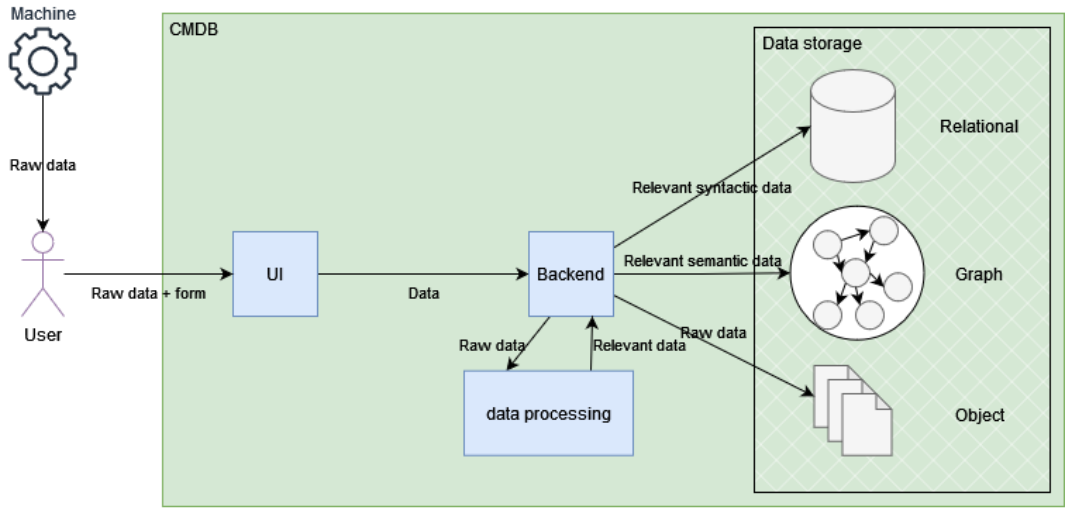


Figure 101: Data Upload Implementation Map of Di^{CMDB}

- **Data query via SDK:** In this example, the GUI is circumvented, and the backend contacted directly. This allows people and machines (other DiMAT toolkits, for instance) to interact with toolkits via Python scripts. The method, in this case a query, is translated into the appropriate HTTP REST API request and sent to the backend, which gathers the relevant data from the different storages and provides a response.

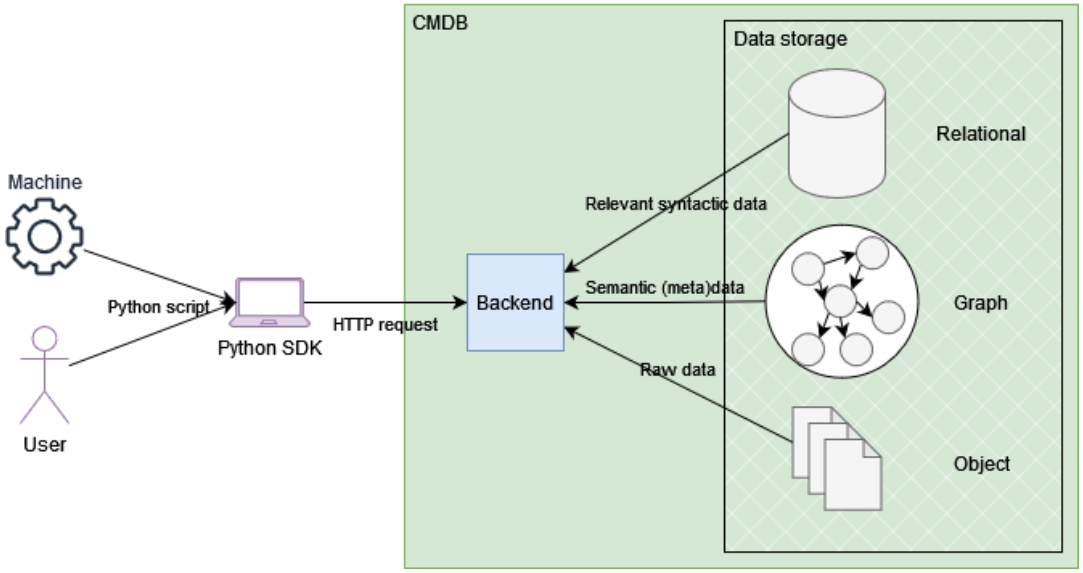


Figure 102: Data query via SDK Implementation Map of Di^{CMDB}

5.3 DIMAT KNOWLEDGE ACQUISITION FRAMEWORK (DI^{KAF})

5.3.1 Internal Architecture of Toolkits

The most central component of the KAF toolkit is the Graph Database (i.e., Neo4J or any other relevant solution). This database is populated with data coming from external sources that are documents describing material processes, RDF triples, or material ontologies, etc. These data, especially text from documents, may require preprocessing such as application of Natural Language Processing techniques. The users can interact with the toolkit via a suitably designed User Interface and request for different functions to be executed on the stored data. Such functionalities include querying, visualization, recommender systems and data analysis tasks in general. In addition, two of DiMAT's toolkits have a connection to the KAF toolkit. CMDB can be used to access data and populate the graph database, while DTPC can utilize the stored data for the operation of the toolkit's virtual functions.

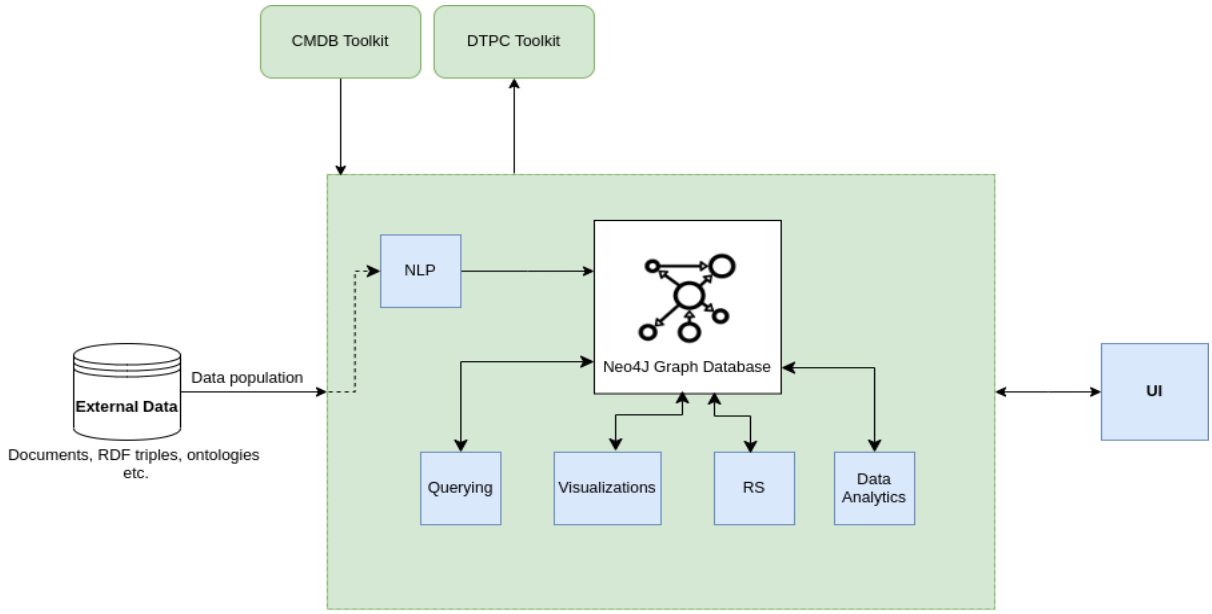


Figure 103: Knowledge Acquisition Framework Component Map

5.3.2 Required components

5.3.2.1 Hardware Components

- **Servers**: For ensuring the smooth operation of the toolkit, a server with high processing capabilities should be employed (enough CPU cores) capable of running both the software required for storing the data into the Knowledge Graph (KG) format and the software that implements the different functions that operate on the KG data

(e.g., visualization, recommender systems, etc.). Also, there should be enough storage space to maintain the Knowledge Graph.

5.3.2.2 Data Storage

The KAF related data are stored in the Knowledge Graph implemented with the Neo4J knowledge graph database.

5.3.2.3 Data Analysis

5.3.2.3.1 Data Flows

- **External data to KAF:** The external data may consist of text documents, structured data (e.g., RDF), material ontologies, etc. These data need to be transmitted to the KG in suitable formats that can be translated to KG format.
- **CMDB to KAF:** Some of the information stored into the CMDB toolkit's repositories may be beneficial for the purposes of DiKAF. The data must be transmitted in suitable formats as explained for the case of other external datasets.
- **KAF to DTPC:** Data revealing material relationships/requirements can be useful for the virtual functions defined in the DTPC toolkit.

In the table presented below examples of data flows are given alongside indicative manners of transmission.

Data flows among components			**Manner of transmission**
Component 1	**Directionality**	**Component 2**	
External data	One directional	Knowledge Graph	FTP, HTTP, APIs
Knowledge Graph	Bi-directional	Functions	Neo4J APIs
KAF	Bi-directional	User Interface	Neo4J APIs, HTTP

Table 108. Data Flow of Di^{KAF}

5.3.2.3.2 Data Preprocessing

Data preprocessing is necessary to have the external data in a format suitable to be included in the KG database. Some preprocessing techniques that might be employed include:

- **Data cleaning:** Removal of missing values, corrupted RDF triples, duplicate data, etc.
- **Extraction of entities and relationships:** For external data that are in the form of text documents, NLP techniques must be applied to identify the entities and the relationships that connect them.
- **Data transformation:** Convert the raw data into formats the Neo4J software can understand, such as RDF, CSV, etc.

5.3.2.3.3 Data Analytics

In the KAF toolkit data analysis is performed by functions operating on the Knowledge Graph. Some of these functions are described in the table provided below.

Software functions	Details
Visualization	Parts of the graph are visualized to give to the user a brief overview of the relationships between subsets of the graph's entities. Also, through suitably formed queries useful statistics from the graph can be obtained and visualized (e.g., number of specific entities, percentages, etc.).
Recommendation Systems	Recommender systems can be applied to suggest the optimal selection of activities for a process or the appropriate materials or suggest actions that lead to a specific goal (e.g., reduce cost of an operation).
NLP	Necessary techniques to extract entities and relationships from textual documents concerning materials.
Data analytics	Other data analysis algorithms that can be either implemented in Neo4J libraries like GDS or be custom developed for specific use cases.

Table 109. Data Analytics of Di^{KAF}

5.3.3 Technologies employed

Toolkits employed (e.g., Python Libraries, other software). The toolkits employed for implementing the aspects of KAF are software tools and libraries that are open-source, popular and with active communities surrounding them.

- Neo4j will be employed as a graph database system to create and manage the knowledge graph representing materials and manufacturing processes. As mentioned in the D2.2 deliverable, Neo4J is one of the most popular graph databases with active community, minimum requirements for deploying and an open-source version. Moreover, Neo4J displays better performance in large graph databases compared to ontologies management and reasoning tools.
- Tools for visualization of data, such as NeoDash and Neo4j Bloom. Which are tools that are built on top of the Neo4J database

- Python libraries for the implementation of the functions supporting the toolkit:
  - NLP libraries (e.g., spaCy). spaCy is one of the most popular libraries for building NLP applications. Its main advantages are the ease of use, the straightforward integration with other Python tools and the speed of its implementations, as discussed also in more detail in D2.2. Ampligraph to create graph embeddings of the knowledge graph elements. Ampligraph is an open-source software library, also examined in D2.2, popular among developers for knowledge graph embeddings offering implementations for many different algorithms that can be employed for KAF.
  - Libraries for Recommendation Systems (e.g., LightFM, PySClump). Python offers many such popular open-source libraries with viable requirements and well written documentation and active communities.

5.3.4 Implementation maps

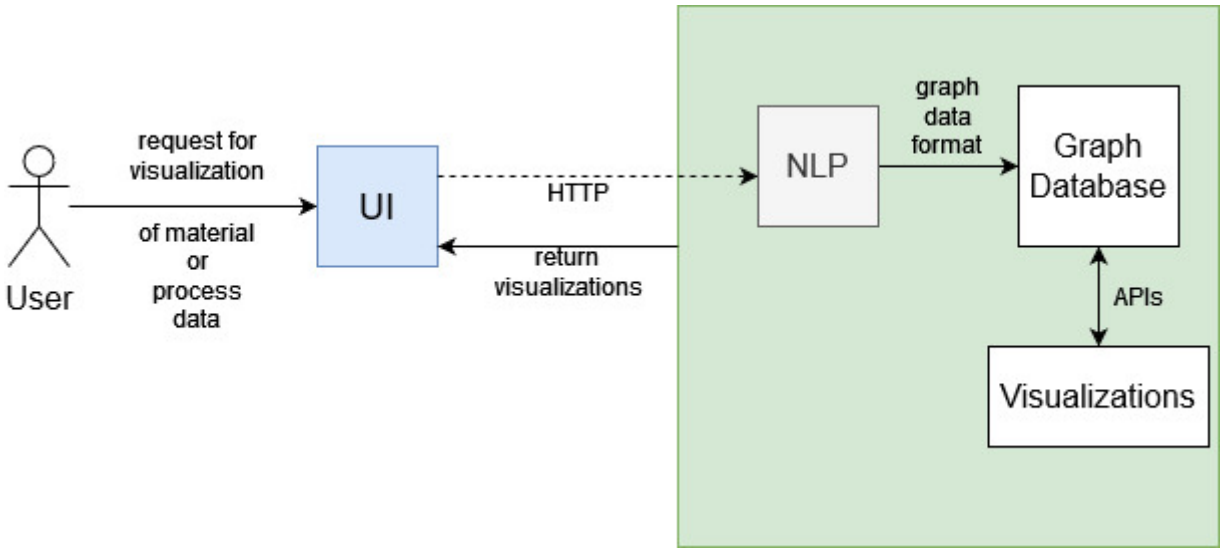


Figure 104: Implementation Map of Di^{KAF}

The figure above illustrates, in the form of a diagram, the implementation map of a user requesting to visualize specific data stored in the KAF toolkit. The user specifies the desired data to be further explored and analyzed, through a UI that communicates the request to the toolkit with an appropriate communication protocol, such as HTTP. An NLP component further processes the input in case data is not in a graph data format, and the processed input is passed to the Neo4j Database to perform the suitable queries. By employing Neo4j APIs that communicate with visualization tools, the results are returned and displayed to the user through the UI for further analysis.

5.4 DIMAT MATERIALS ENVIRONMENTAL AND COST LIFE ASSESSMENT (DI$^{MEC\text{-}LCA}$)

5.4.1 Internal Architecture of Toolkits

The system comprises several components that work together to facilitate the seamless flow of data. The CSV component serves as the data source, allowing users to upload raw data in CSV format. The Infinity plugin component integrates with Grafana, enabling users to effortlessly import and utilize CSV data for visualization and analysis. Grafana acts as the central visualization and dashboarding tool, offering users an intuitive interface to create, customize, and share visualizations based on the data from CSV files and the MySQL database. Laravel plays a critical role in the data flow by handling the integration between the MySQL database and Grafana, ensuring secure and efficient insertion of CSV data. The MySQL database serves as the backend storage and retrieval system, providing a robust

solution for storing and managing the uploaded data. Together, these components ensure a smooth and efficient data flow, enabling users to explore and analyze their data effectively within Grafana.

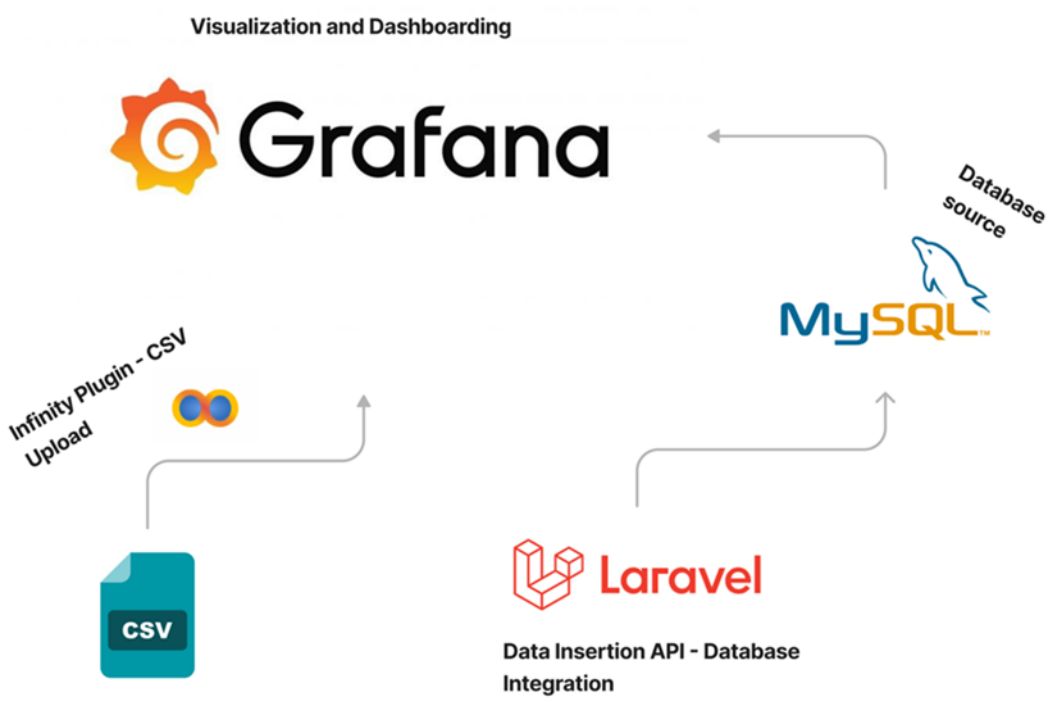


Figure 105: Technologies of Di$^{MEC-LCA}$

5.4.2 Required components

5.4.2.1 Hardware Components

The system architecture incorporates essential hardware components to ensure optimal performance and reliability. The Docker-based setup consists of three primary containers: Grafana, MySQL, and Laravel, all hosted on a dedicated server. The server features robust hardware specifications, including a high-performance processor and ample RAM. This enables efficient data processing, visualization, and data management within the containers. The server's network infrastructure maintains a stable and high-bandwidth connection, facilitating real-time data ingestion and seamless interactions between the containers. While the Laravel container is available, it is primarily utilized when there is a need to insert CSV data into the MySQL database for specific queries. Laravel provides a powerful framework

for handling file uploads and performing database operations, streamlining the insertion process. On the other hand, the system also offers a simpler approach using the Infinity plugin without relying on Laravel. With the plugin, users can directly upload the CSV file through Grafana and visualize the data without the need for additional database operations or integration with Laravel. This provides a user-friendly and straightforward way to display graphs and visualize the data from the specific CSV file.

By optimizing the hardware components on the dedicated server, the system delivers reliable performance, seamless data visualization, and provides the flexibility to utilize either Laravel for CSV data insertion or the Infinity plugin for direct CSV file upload and visualization in Grafana.

5.4.2.2 Data Storage

The Docker stack comprises three containers: Laravel, MySQL, and Grafana, with each container serving a specific role in the data storage process.

- MySQL acts as the primary data storage solution within the stack. It provides a robust and scalable relational database management system for storing structured data efficiently. The MySQL container ensures reliable data persistence and retrieval, facilitating seamless storage and retrieval of Life Cycle Assessment (LCA)/Key Performance Indicator (KPI) data.
- Laravel, utilized solely by the admin, enables the creation of seeders to populate the MySQL database with data. By leveraging Laravel's seeders, the admin can easily insert data into the MySQL database, setting up the foundation for subsequent data querying and analysis.
- Grafana, integrated with MySQL as a data source, allows the admin to run queries and perform advanced data analysis when needed. Additionally, the Infinity plugin simplifies data visualization by enabling the direct upload of CSV files into Grafana. The admin can leverage the plugin's capabilities to create dynamic graphs and visualize LCA/KPI results without relying on complex database queries.

In summary, the combination of Laravel, MySQL, and Grafana forms an efficient data storage solution within the Docker stack. Laravel streamlines the process of populating the MySQL database, MySQL ensures reliable data storage and retrieval, and Grafana, along with the Infinity plugin, provides flexible options for data querying, visualization, and analysis.

5.4.2.3 Data Analysis

5.4.2.3.1 Data Flows

Data flows within the Docker stack are designed to facilitate efficient handling and utilization of information, ensuring a streamlined process. The flow begins with the admin utilizing the Infinity plugin in Grafana. By uploading CSV files directly through the plugin, the admin can quickly generate dynamic graphs and visualize LCA/KPI results without the need for complex database queries. This provides a user-friendly and straightforward approach to data visualization. Alternatively, the admin can leverage Laravel for data insertion into the MySQL database. Using Laravel's seeders, the admin can populate the database with CSV files or other relevant data sources. This step ensures that the data is properly stored and readily available for retrieval and analysis. Once the data is stored in the MySQL database, Grafana connects to it as a data source, allowing the admin to run queries, retrieve specific datasets, and perform advanced data analysis if needed. Grafana's intuitive interface empowers the admin to explore and visualize the LCA/KPI results effectively. In summary, the data flows within the Docker stack prioritize ease of use and flexibility. The Infinity plugin in Grafana allows for quick data visualization, while Laravel facilitates data insertion into the MySQL database. Finally, Grafana serves as the platform for data retrieval, analysis, and visualization, enabling the admin to gain valuable insights and make informed decisions based on the LCA/KPI results.

Data flows among components			**Manner of transmission**
Component 1	**Directionality**	**Component 2**	
GUI	One directional	Grafana	HTTP
External data	One directional	Grafana	CSV plugin
Laravel	One directional	MySQL database	REST API
MySQL database	One directional	Grafana	MySQL query

Table 110. Data flow of Di$^{MEC-LCA}$

5.4.2.3.2 Data Preprocessing

In the Docker stack, users have the flexibility to preprocess the CSV files locally before utilizing them within the system. They can employ various data preprocessing techniques to modify and manipulate the data as per their specific requirements.

The system does not provide built-in data preprocessing functionalities, but users can leverage their preferred tools and methods to preprocess the CSV files externally. This may involve using local data preprocessing frameworks, programming languages, or software tools to clean, transform, and manipulate the data before uploading it into the system.

Data preprocessing is a critical step in ensuring data quality, addressing inconsistencies, and preparing the data for further analysis and visualization. By performing preprocessing techniques locally, users can tailor the data to their specific needs, enhancing the accuracy and reliability of the LCA/KPI results obtained through Grafana. The Docker stack empowers users to take control of their data preprocessing workflows, allowing them to derive more accurate insights and make informed decisions based on the processed data. Although the system does not include in-built data preprocessing capabilities, users have the freedom to preprocess the CSV files externally to meet their specific requirements. By providing users with the flexibility to perform data preprocessing locally, the Docker stack allows for customization and ensures that the data utilized within the system is carefully prepared to yield accurate and meaningful results.

5.4.2.3.3 Data Analytics

Data analytics plays a crucial role in the Docker stack, enabling users to gain valuable insights and extract meaningful information from the stored data. Using Grafana's powerful querying capabilities, users can perform advanced data analytics on the LCA/KPI data. Grafana connects to the MySQL database, allowing users to execute queries and retrieve specific datasets based on their analysis needs. By leveraging the query language supported by Grafana, users can filter, aggregate, and manipulate the data to uncover patterns, trends, and correlations. Additionally, Grafana provides a wide range of visualization options to present the analyzed data in a clear and intuitive manner. Users can create dynamic and interactive dashboards, charts, and graphs to visually represent the LCA/KPI results. This visual representation enables users to easily identify insights, identify anomalies, and communicate findings effectively.

Furthermore, the integration of the Infinity plugin expands the data analytics capabilities within Grafana. Users can upload CSV files directly into Grafana, utilizing the plugin's features for data manipulation and visualization. This allows for quick data exploration and analysis

without relying solely on database queries. By combining data analytics techniques with the comprehensive features offered by Grafana and the Infinity plugin, users can derive actionable insights and make data-driven decisions. The Docker stack provides a robust foundation for data analytics, facilitating a seamless and efficient process of extracting valuable information from the stored LCA/KPI data.

5.4.3 Technologies employed

5.4.3.1 Toolkits employed (e.g., Python Libraries, other software)

The Docker stack incorporates various tools and technologies to enhance data analytics capabilities, enabling efficient analysis and visualization of the stored data. Within the stack, Grafana provides the ability to analyze the data using a graphical interface. This is intuitive and allows multiple forms of configuration, be it graphs, tables or dashboards. Grafana empowers users to explore and analyze the LCA/KPI data, providing a comprehensive toolkit for visualizing trends, patterns, and anomalies, due to its flexibility and ability to handle large datasets seamlessly.

In addition, the MySQL container plays a crucial role as the database management system within the stack. It efficiently stores and retrieves the LCA/KPI data, enabling seamless data access for analysis purposes. MySQL was chosen for its reliability, speed, and its well-established role in data-driven applications. The structured query language (SQL) it uses is essential for performing complex data queries, which in turn integrate smoothly with Grafana for visualization.

While specific external toolkits or software may not be utilized within this Docker stack, Grafana's native features and the robust functionality of MySQL provide a comprehensive toolkit for data analytics. Users can leverage Grafana's built-in capabilities and the querying capabilities of MySQL to extract valuable insights and perform data analysis. Users can leverage Grafana’s visualization tools alongside MySQL’s querying capabilities to pull insights from the data. Docker ensures each tool can be deployed, maintained, and scaled independently, adding to the stack’s resilience and adaptability as data needs grow. Furthermore, Laravel is used to make data entry into MySQL smoother and simpler. With Laravel’s ORM (Eloquent), developers can manage and input data without using complicated SQL, which speeds up building and maintaining the data management process. Plus, Laravel’s built-in security features, like CSRF protection and defenses against SQL injection, keep the data safe and secure, which is relevant for sensitive LCA/KPI data.

Overall, the Docker stack incorporates Grafana, MySQL and Laravel as key components for data analytics, offering a powerful toolkit to explore, analyze, and visualize the LCA/KPI data.

Open-source technologies like Grafana, MySQL and Laravel offer significant community support, cost-effectiveness, and adaptability, allowing continuous improvement and customizability tailored to specific project needs without licensing restrictions.

For the LCA assessments themselves, which provide the results to MEC-LCA, the software OpenLCA is used, as prioritized in D2.2. This is done outside the system and the results are added to the data storage.

5.4.4 Implementation maps

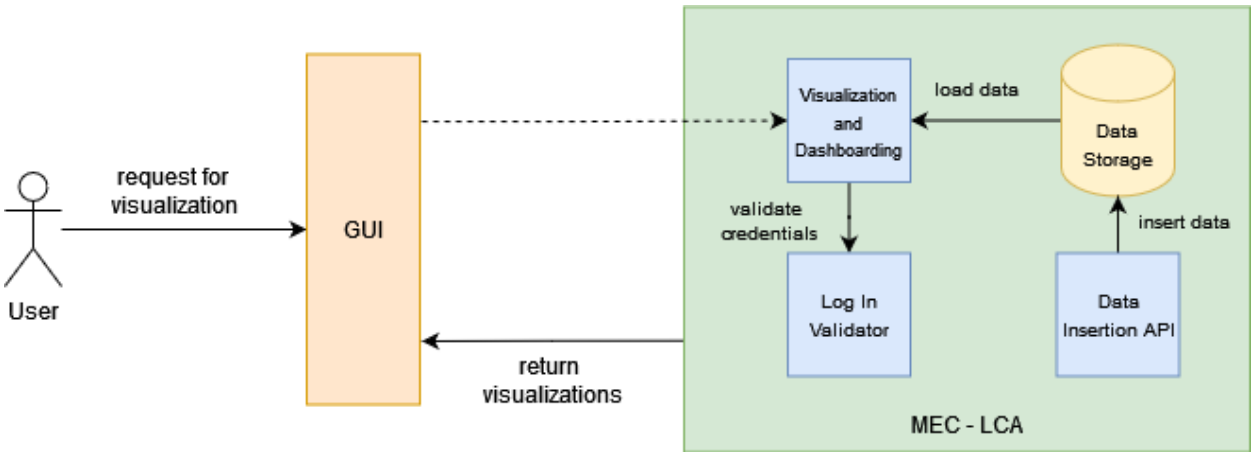


Figure 106: Implementation Map of Di$^{MEC-LCA}$

The diagram above gives a high-level view of a user's request to visualize data stored in the MEC-LCA tool. The user can specify the desired data through the GUI, which communicates the request to the tool. The Visualization and Dashboarding component receive the request and retrieves the necessary data from the Data Storage. Data can be collected by the admin either via direct upload of files into Grafana with the Infinity plugin or by leveraging Laravel for data insertion into the MySQL database through the Data Insertion API. Prior to viewing any content, the user's credentials must be validated. Finally, the results are returned and displayed to the user through the GUI.

5.5 DIMAT MATERIALS DESIGN FRAMEWORK (DI^{MDF})

5.5.1 Internal Architecture of Toolkits

The following diagram presents how the components of MDF relate with each other:

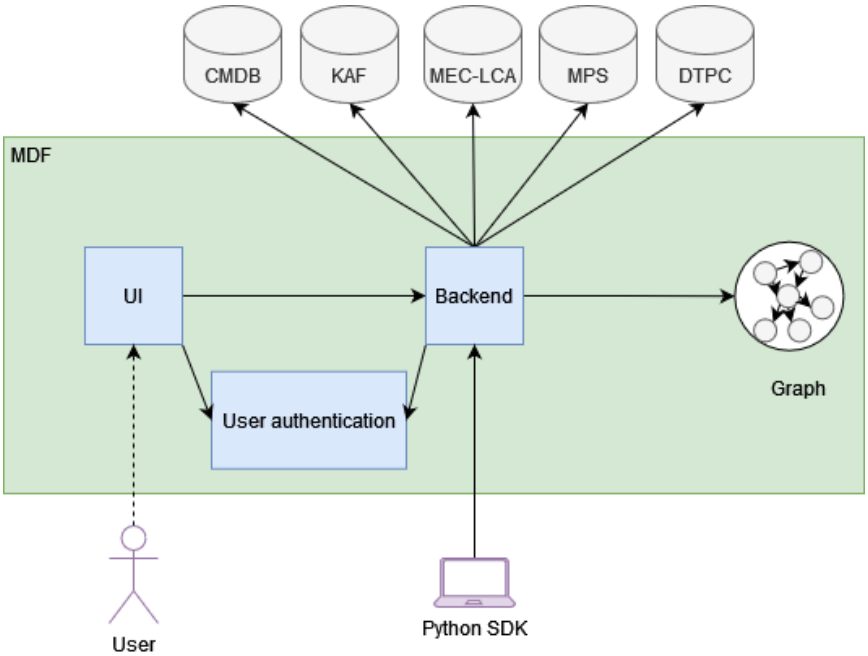


Figure 107: Materials Desing Framework Component Map

Since the tool focuses on the providing advanced search functionalities, the UI plays an important role as it should provide a user-friendly interface for specifying the search parameters and presenting the results in a useful manner. The backend will help collecting information from the other tools. For example, the CMDB may contain datasets that are relevant for the search, that will be retrieved by the MDF and be presented to the user. The graph database will maintain ontological information such as mapping to connect concepts coming from the ontologies employed by the other tools.

5.5.2 Required components

5.5.2.1 Hardware Components

The only hardware requirement for deployment of the toolkit is an internet reachable server with enough available resources for handling the search and forwarding requests to the other toolkits. Furthermore, some storage space will be required for toolkit specific data.

5.5.2.2 Data Storage

As an ontology-based toolkit, a triple store will be required to manage relevant semantic data, such as linking of concepts between ontologies of different tools. The triple store will

be powered by the free edition of AllegroGraph [5]. DiCMDB will also be used for data retrieval.

5.5.2.3 Data Analysis

5.5.2.3.1 Data Flows

Data exchange between MDF and other DiMAT's tools:

- **CMDB to MDF:** Relevant datasets (w.r.t. search criteria)
- **KAF to MDF:** Relevant material data
- **MEC-LCA to MDF:** Relevant environmental and life cycle data of materials
- **MM to MDF:** Relevant information about material properties
- **MD to MDF:** Relevant designs of new materials
- **MMS to MDF:** Relevant AI-based tools
- **MPS to MDF:** Relevant process information
- **DTPC to MDF:** Relevant digital twins

The MDF will collect information from the different tools of DiMAT to meet the search parameters given by the user. The following table summarizes the communication inside the MDF:

Data flows among components			**Manner of transmission**
Component 1	**Directionality**	**Component 2**	
UI	One directional	Backend	HTTP, REST API
Backend	One directional	DiMAT tools (CMDB, KAF, ...)	HTTP, REST API
Backend	One directional	Triplestore	AllegroGraph API (REST API, Python SDK)
UI	One directional	User authentication	HTTP, REST API via JavaScript libraries

Backend	One directional	User authentication	HTTP, REST API via Python libraries

Table 111. Data Flow of Di^{MDB}

5.5.2.3.2 Data Preprocessing

As part of processing a search query initiated by the user, data will be collected from the DiMAT tools. The following pre-processing steps will take place during the execution of the search:

- **Tool identification:** At the beginning of the search, MDF will analyze the query to detect which tools are relevant for the executing the search.
- **Data transformation:** The search results should be converted to a uniform representation that will allow an easy further processing of the data as well as an easy rendering of the results by the UI.

5.5.2.3.3 Data Analytics

The MDF will employ a few data analytics capabilities. There are listed in the following table:

Software functions	Details
Relation Extraction	The detection and classification of semantic relationship mentions between a set of artifacts that are stored within the different DiMAT's tools. Here the relations and mappings maintained in the triplestore of the MDF will come into play to power to discovery of the implicit links.
Ranking	The search results will be sorted according to ranking given to each result that indicates the relevance to the search query.

Table 112. Data Analytics of Di^{MDB}

5.5.3 Technologies employed

Internally, the Materials Design Framework will store the connections used for the search in a triplestore. The backend service will contact other toolkits and it will be developed in Python, with the FastAPI framework. The frontend will be written in Angular. Deployment will be done via Docker. Further possible technologies to be employed are machine learning tools, such as scikit-learn and tools for storing data like PostgreSQL. Both tools have an active

community and are publicly available, as also indicated in the relevant sections of the D2.2 deliverable.

5.5.4 Implementation maps

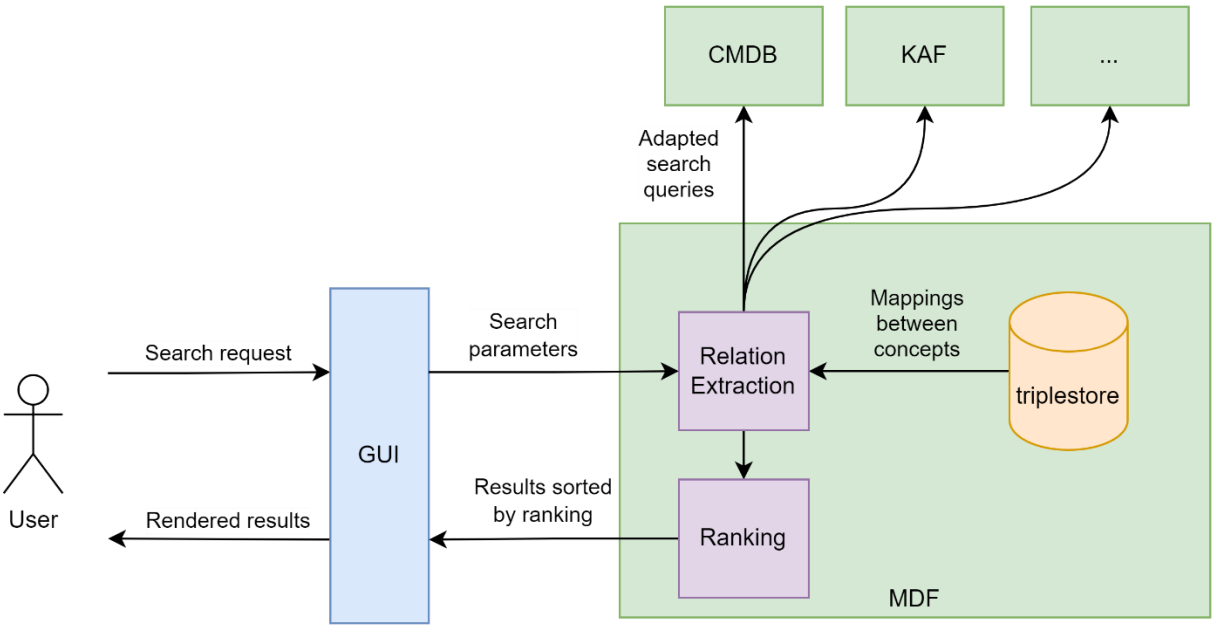


Figure 108: Implementation Map of Di^{MDB}

The above diagram shows another view for the implementation of the MDF. First, a user submits a search request via the GUI of the MDF, which will be sent to the Relation Extraction module. This component will send corresponding queries to the other DiMAT's tools, whose results will be linked using the mapping information stored in the triple store of the MDF. All results will be subsequently processed by a ranking module to sort the result according to the relevance of the search parameters. The sorted results will be sent back to the GUI to be rendered for the user.

5.6 DIMAT MATERIALS MODELLING AND BEHAVIOUR PREDICTION (DI^{MM})

5.6.1 Internal Architecture of Toolkits

The above image depicts the Materials Modeler architecture. It is designed around robust AI and causal inference approaches, providing a system that can predict, infer, and suggest optimal configurations for various materials under diverse conditions. The core components include a communication layer, a data collection and preprocessing layer, an AI analysis layer, and a prediction and inference layer. The communication layer of the Materials Modeler (MM) interfaces using RESTful APIs, enabling seamless interaction with MDF and CMDB databases for efficient data retrieval. It also enables bi-directional information flow.

Data is collected via the partners' submissions which involve the material properties and behaviours under different manufacturing conditions. This data is then pre-processed and analysed using AI algorithms, including explainable AI, machine learning (ML), and deep learning (DL). The AI analysis layer uses these algorithms to infer latent relationships between material structure and properties and suggest optimal configurations. The prediction and inference layer then provides critical insights and potentially control capabilities in microstructure mechanisms.

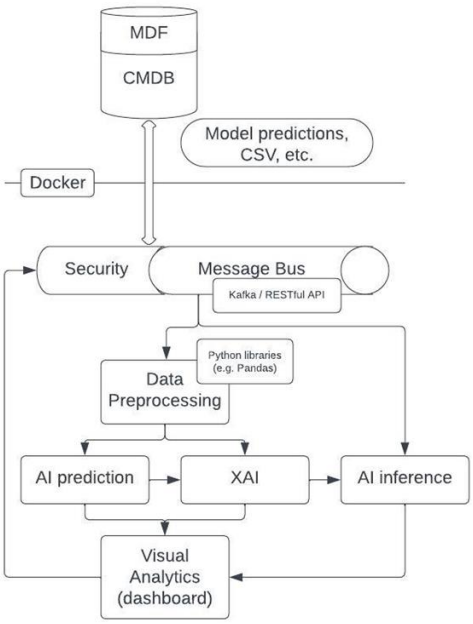


Figure 109: Materials Modelling and Behaviour Prediction Component Map

5.6.2 Required components

5.6.2.1 Hardware Components

For the effective implementation of the Materials Modeler (MM), the following hardware components are essential:

1. **Server:** High-performance server hardware, equipped with fast multi-core CPUs, is crucial for handling the complex computational load of AI/ML tasks. These servers will house the Docker containers that the MM operates within.
2. **GPU:** To optimize the performance of deep learning elements in the MM, the incorporation of powerful Graphics Processing Units (GPUs) is vital. Dockerized deep learning tasks can leverage these GPUs to significantly accelerate computational processes.
3. **Temporary Storage (RAM):** High-capacity, rapid-access Random Access Memory (RAM) is required to store intermediate data during computations and to ensure the efficient running of the MM software in its Docker environment.

All these components should be configured to support Docker-based deployment, promoting flexibility, scalability, and isolation of resources and processes.

5.6.2.2 Connectivity Devices (Gateways, Switches, etc)

These devices ensure that data collected from various sources is aggregated and transmitted to the central AI system for analysis. They also support bi-directional communication for control and configuration purposes. Each toolkit within the DiMAT suite will feature its own unique API, enabling users to interact seamlessly with each one. Each toolkit within the DiMAT suite will be equipped with its own distinct API, thus allowing users to interact with each individually or through the utilization of a comprehensive frontend dashboard. Additionally, these toolkits will have the capability to communicate interdependently, further streamlining the communication process.

5.6.2.3 Data Storage

Long-term storage, possibly cloud-based, is utilized for historical data and long-term pattern analysis. The Materials Database Framework, an ontology-based database constructed atop

the Cloud Materials Database, will facilitate efficient data exchange between each toolkit, with a particular focus on the Materials Modeler.

5.6.2.4 Data Analysis

5.6.2.4.1 Data Flows

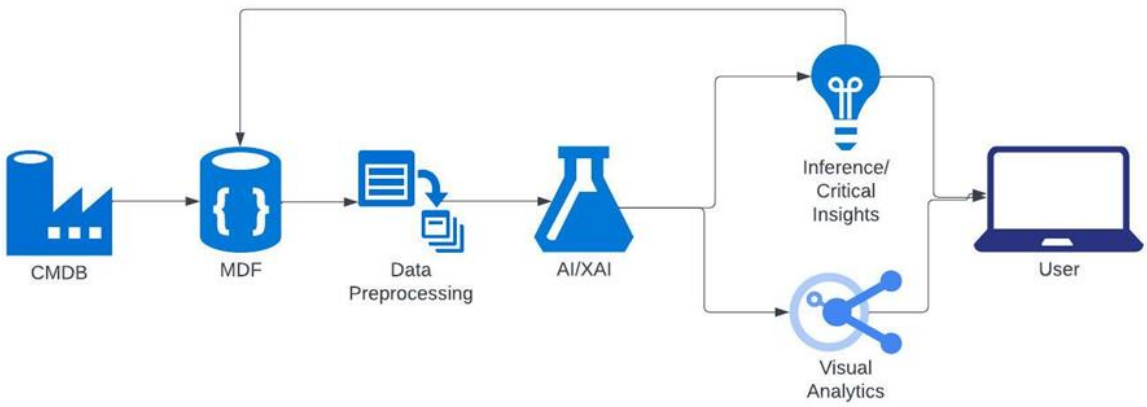


Figure 110: Data flows of Di^{MM}

Data is sourced from the collection stage, progressing through an extensive preprocessing phase before reaching the AI analysis layer. Here, bolstered by Explainable AI (XAI) algorithms, the data is processed to uncover clear decision-making properties. This analyzed data is then advanced to the prediction and inference layer. The resultant insights are conveyed to the user interface for visualization, designed to be comprehensive and interactive for effective interpretation. Finally, these data insights are recirculated back into the Materials Design Framework, making them accessible for future analytical runs and further enhancing the iterative learning process.

Data flows among components			Manner of transmission
Component 1	**Directionality**	**Component 2**	
Data Preprocessing	Uni-directional	AI/XAI	HTTP, Cloud storage APIs

AI/XAI	Uni-directional	Inference/Critical Insights – Visual Analytics	HTTP, Cloud storage APIs
Inference/Critical Insights	Bi-directional	Other toolkits - User	HTTP, Cloud storage APIs
Visual Analytics	Uni-directional	User	HTTP, Cloud storage APIs

Table 113. Data Flow of Di^{MM}

5.6.2.4.2 Data Preprocessing

Data preprocessing encompasses an array of procedures, including data cleansing, annotation, addressing missing or anomalous values, and formatting the data in an optimal structure for the AI algorithms. This process might entail rectifying inconsistencies, resolving discrepancies, and applying normalization techniques to ensure the data is suitable for further analysis. Particularly, the focus will be on handling raw data, which often requires more rigorous preprocessing. This will involve conventional methods of manipulating raw data such as imputation, transformation, and normalization. Imputation techniques are employed to fill in missing or incomplete data. Transformation methods, including scaling or binning, are used to change the distribution or relationship of the data to better fit model assumptions. Normalization procedures, such as min-max normalization or standardization, ensure the data is on a comparable scale and can improve the performance of certain algorithms. Once the data is suitably pre-processed, it serves as an input for the AI models. These models are designed to update themselves as new data becomes available, ensuring they evolve and improve over time, thereby enhancing their performance and accuracy.

5.6.2.4.3 Data Analytics

Utilising robust AI algorithms, the toolkit will analyse pre-processed data, predict material behaviour and properties, and uncover hidden relationships between material structures. They provide data-driven suggestions for optimal configurations. In addition to their robust predictive capabilities, it will feature explainable AI components. These elements make the decision-making processes transparent, enabling users to understand and trust the insights generated. The derived results offer critical insights and potential control capabilities for microstructure mechanisms. This AI-assisted analysis streamlines the interpretation of complex material behaviours, fostering advancements in materials science.

5.6.3 Technologies employed

5.6.3.1 Toolkits employed (e.g., Python Libraries, other software)

Python libraries for ML and DL, like TensorFlow, PyTorch, etc., can be used along with libraries for data preprocessing and analysis like Pandas, NumPy, Scikit-learn, etc.

The docker stack comprises of a Python backend utilising the tools mentioned like scikit-learn, Pandas, NumPy etc., the library streamlit for the frontend and HuggingFace transformers and models for the LLM support. Everything used in stack is open source and well maintained and they were selected and evaluated based on what is mentioned in Deliverable 2.2.

5.6.3.2 Cloud-computing platforms

Cloud platforms such as AWS, Google Cloud, or Azure offer feasible options for long-term data storage and executing computationally intensive AI models. Additionally, the project will also accommodate the functionality to leverage its own servers for these tasks.

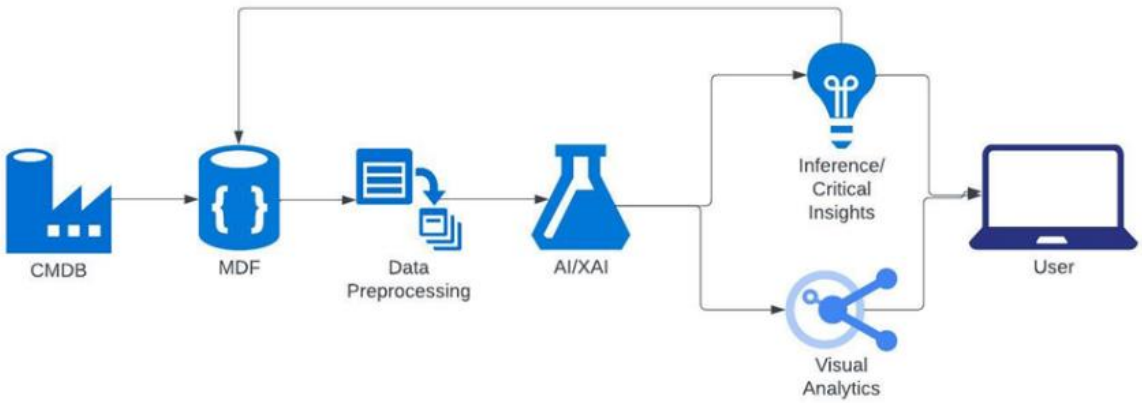


Figure 111: Cloud-computing platforms of Di^{MM}

5.6.4 Implementation maps

The implementation map, which also serves as the data flow diagram, starts with the data collection phase, where initial data is gathered. This data then proceeds to an in-depth preprocessing stage, setting the stage for more refined analysis. Next, it moves into the AI

analysis layer, where it is enhanced by Explainable AI (XAI) algorithms. These algorithms are integral to the process, providing intelligible decision-making properties and contributing to the accuracy and interpretability of the outcomes. Following the analysis, the processed data is channelled to the prediction and inference layer. This is where the final model outputs and insights are produced. The results are subsequently relayed to a user-friendly interface for visualization. Here, the insights are displayed in a comprehensive and interactive manner, fostering a clear and thorough understanding of the results. The last stage in the implementation map reflects the closing loop in the data flow process. It shows the channelling of these insights back into the Materials Design Framework, ensuring they are available for future iterations and continuous improvement of the system.

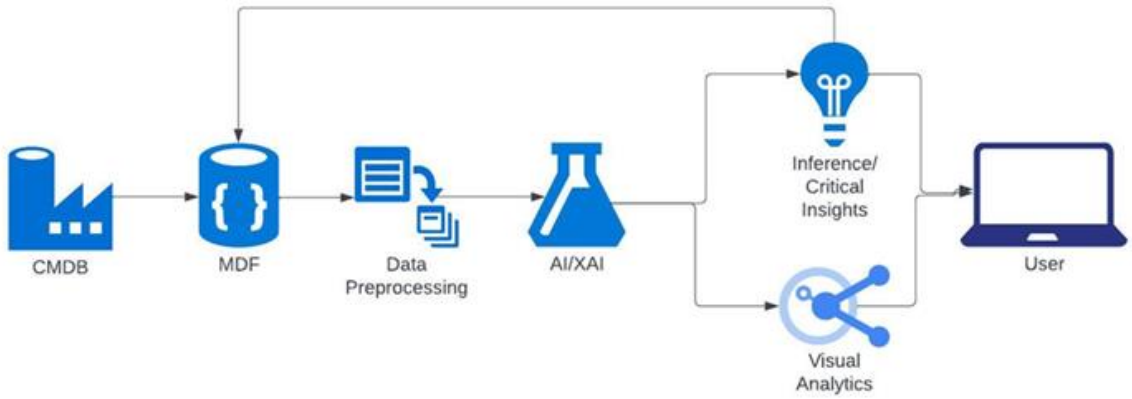


Figure 112: Implementation map of Di^{MM}

5.7 DIMAT MATERIALS DESIGN, CHARACTERIZATION AND SYNTHESIS (DI^{MD})

5.7.1 Internal Architecture of Toolkits

The function of the toolkit is to calculate the material properties of a composite material given its components, their properties, and a chosen microstructure. The core of the toolkit is a commercial code, Hexagon Digimat®, able to execute virtual experiments on a representative volume element of the composite and extract the requested properties from the results. A virtual experiment consists of Finite Element calculation executed on predefined models whose geometry is defined based on the requested microstructure.

The general scheme of the internal architecture of the toolkit can be thought as a classic client-server model. The user has direct interaction with the client component, which take

care of requesting, storing, and displaying data, while the actual calculation is executed by a server which is totally invisible to the user. The principal components of this toolkit are:

1. The user interface, where the user ask for material properties and the results are presented.
2. The internal database, which stores the input of the user and the results obtained.
3. The simulation scheduler, which manage the interaction between the database and the calculation code.
4. The calculation code, which executes the virtual experiments.

The first three components can be tough as part of the client, while the fourth one is the server. A graphic representation of the described architecture is shown in the next figure.

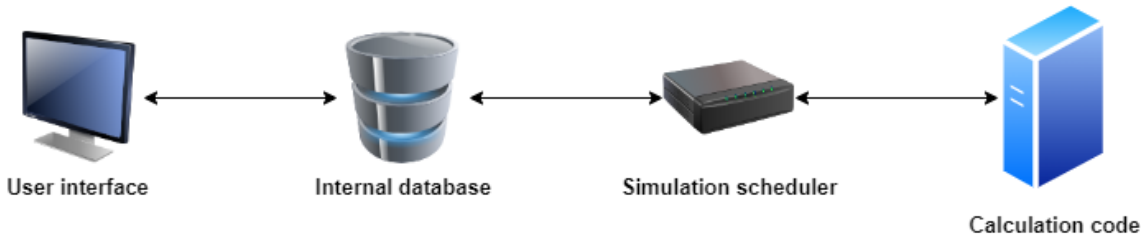


Figure 113: Materials Design, Charazterization and Synthesis of Di^{MD}

5.7.2 Required components

5.7.2.1 Hardware Components

The simplest scenario consists of just one single machine where all the software is installed and running. This is the case when the license of Digimat® code is locally available. If this is not the case, a more complex scenario is involved, which requires two machines:

- Calculation machine, where the Digimat® code is installed, and which oversees executing the FE calculations needed to retrieve the requested data.
- Client machine, where data input is requested to the user or data output is shown to him/her.

5.7.2.2 Connectivity Devices (Gateways, Switches, etc)

No specific connectivity devices are needed.

5.7.2.3 Data Storage

Different kind of data will be produced and stored in different ways.

- **Simulation data** produced by the FE calculation: the storage is local, on the calculation machine, and not permanent since data are erased once the elaborated data are presented to the user.
- **Simulation database:** this database stores the requested calculations and their results and is located on the client machine.
- **CMDB:** the properties of the material components or the final properties of the composite can be retrieved or uploaded from a CMDB instance.

5.7.2.4 Data Analysis

5.7.2.4.1 Data Flows

The data flow is bi-directional.

1. The simulation scheduler takes data from the simulation database, preprocess them to obtain a valid Digimat® input file and send the input data to the simulation code.
2. Output of the simulation code is taken from the simulation scheduler which translates the data into user-friendly output that can be stored in the database and presented to the user.

Optionally, material properties could be uploaded to or retrieved from a CMDB instance.

Data flows among components			**Manner of transmission**
Component 1	Directionality	Component 2	
Simulation scheduler	Bi-directional	Calculation code	HTTP, Cloud storage APIs
Material Designer	Bi-directional	Other toolkits (e.g. CMDB)	Custom developed APIs

Table 114. Data Flow of Di^{MD}

5.7.2.4.2 Data Preprocessing

The requested calculation is defined as a map-like data structure which defines all the necessary data. This map-like data structure must be translated to obtain a Digimat® input file, which is the only input file accepted by the calculation code. A dedicated Python function will take care of the input validation and the translation process.

5.7.2.4.3 Data Analytics

No data analysis is required. The UI allows the user to confront material properties of different analysed solutions stored in the internal database.

5.7.3 Technologies employed

5.7.3.1 Toolkits employed (e.g., Python Libraries, other software)

The choice of the employed technologies is not yet definitive for all the components of the toolkit and there are some evaluations ongoing.

For the toolkit infrastructure, a Python web framework for building APIs will be employed. Flask and FastAPI, both popular open-source projects with extensive documentation and active communities of users have been evaluated. In the end, FastAPI has been chosen given some advantages over Flask, like automatic input validation and its very high performances.

For the simulations database, the choice is not yet definitive. Sqlite3 python package and PostgreSQL are being evaluated. The evaluation is ongoing and is mostly based on two factors (see section 6.4.1 of the D2.2 deliverable): concurrency and scalability.

For the FE calculations, Hexagon Digimat® will be employed. Considering all the FE codes listed in section 7.3 of the D2.2 deliverable, Digimat has been chosen because it is specifically developed to investigate the behaviour of non-homogeneous materials.

For the User Interface, graphic libraries (like for example the Qt Framework) will be evaluated on the basis of ease of use and implementation.

5.7.3.2 Cloud-computing platforms

The possibility to host the calculation code on a cloud HPC system is to be evaluated. Model dimensions for the kind of calculations involved rarely require more than a simple

workstation to be executed. An instance of the CMDB can be considered as a cloud solution to archive and retrieve data.

5.7.4 Implementation maps

The next figure shows how the different components interact starting from the initial input of the user.

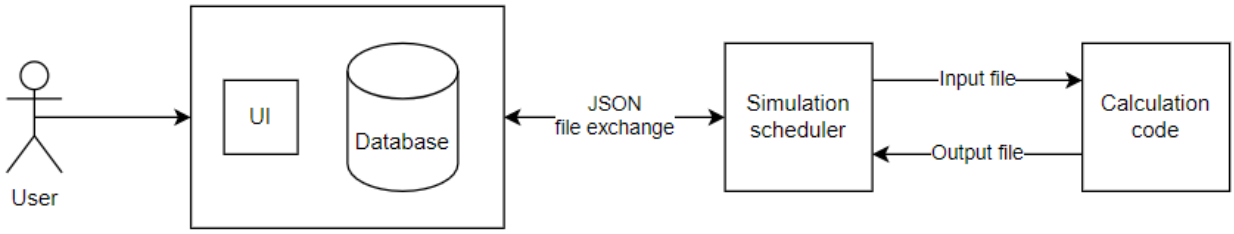


Figure 114: Implementation Map of Di^{MD}

Once the user asks for a specific material configuration to be analysed, the internal simulation database is updated adding the new asked simulation. Then, a json file describing the specific configuration to be analysed is sent to the simulation scheduler, which oversees the communication with the simulation code. The simulation scheduler pre-processes the received JSON file and produces a valid Digimat® input file which is then passed to the code whose execution is triggered. The scheduler, while waiting for other inputs from the user, can retrieve the output data generated by the simulation code and produce a structured JSONfile with the requested output data. The output json data is then processed and the requested material properties are added to the database in the specific record related to the simulation. In the end the user can review the results and, optionally, send them to an instance of the CMDB.

5.8 DIMAT MATERIALS MECHANICAL PROPERTIES SIMULATOR (DI^{MMS})

5.8.1 Internal Architecture of Toolkits

The surrogate model of the DMMS will link the input, compounds, fractions, micro-properties, microstructure with the output, the predicted mechanical properties. The surrogate model (SM) will be a simplified and computationally efficient representation of the physical models. The SM will capture the relationships between input variables and output responses of the original system and will operate in real time.

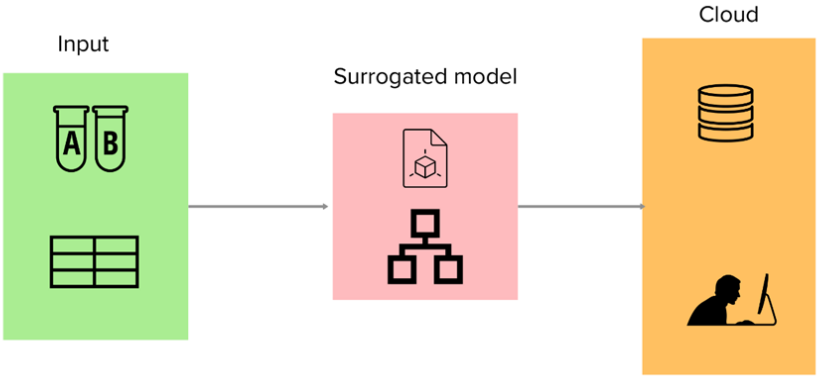


Figure 115: Materials mechanical Properties Simulator Component Map

5.8.2 Required components

5.8.2.1 Hardware Components

Various types of servers can be employed in the development of the mechanical models.

- **Database Servers:** to handle data storage related to the DMMS toolkit and provide mechanisms for querying and efficient access.
- **Servers for simulations:** DMMS toolkit will use dedicated servers for executing finite element and atomistic simulation models, implementing algorithms, and generating predictions of mechanical properties.

5.8.2.2 Connectivity Devices (Gateways, Switches, etc)

No specific connectivity devices are needed.

5.8.2.3 Data Storage

- **Simulation data:** the input and output of simulation should be stored in specialized storage systems such as simulation data management platforms.

Data for long-term storage should be better allocated into the Cloud and in collaboration with the DiMAT Cloud Materials Database.

5.8.2.4 Data Analysis

5.8.2.4.1 Data Flows

- **CMBD data to MMS:** Material data stored in will be used in MMS.
- **MMS to CMBD:** Relevant output simulations will be stored in CMBD.
- **MM to MMS:** Multiscale simulation of material properties could be based on previous MM models.
- **MMS to MPS:** MMS mechanical simulation output will be integrated into MPS.
- **MMS to DTPC:** MMS mechanical simulation output will be integrated into DTPC.
- **MMS platform and user:** The input data received from the user or API will be fed into a surrogate model, which establishes a connection between the input parameters and the corresponding mechanical property. The surrogate model acts as a bridge, effectively mapping the input data to the desired output value. Once this mapping is performed, the resulting mechanical property value will be stored in a database for further reference and utilization.

Data flows among components		
Component 1	**Directionality**	**Component 2**
CMBD	Bi-directional	MMS
MM	Bi-directional	MMS
MMS	one-directional	DTPC
MMS	one-directional	MPS
UI	Bi-directional	Cloud DB

Table 115. Data Flow of Di^{MMS}

5.8.2.4.2 Data Preprocessing

The input data format and numerical values will undergo thorough validation to prevent any abnormal functionality within the model. The physical models have extensively examined a

specific region in the parametric space; thus, it is crucial to ensure that the input falls within that defined range.

5.8.2.4.3 Data Analytics

The output data generated from the physical models will be carefully analyzed and integrated into a surrogate model using advanced techniques such as regression, artificial neural networks (ANN), support vector machine (SVM), or similar algorithms. These methods enable the extraction of valuable insights and patterns from the data, allowing us to create a robust surrogate model.

5.8.3 Technologies employed

Toolkits employed (e.g., Python Libraries, other software Python libraries will be used for the following functionalities:

- Machine-learning tasks: e.g., PyTorch, TensorFlow or scikit-learn offer powerful libraries for building surrogate models, which are required for real-time models. Libraries like scikit-learn provide powerful tools for support vector machines (SVMs) and dimensionality reduction (e.g., PCA), which can reduce the complexity of data without losing accuracy.
- Pandas will be used for efficient data manipulation, handling modelling inputs, outputs and simulation results.
- Matplotlib and Plotly will generate detailed visualizations of the modelling results.
- Numerical packages such as numpy and scipy will provide robust numerical computing capabilities required for performing scientific calculations

The next molecular dynamic software will be used:

- Molecular dynamic solver: LAMMPS and RadonPy, stated in D2.2, are efficient molecular dynamic solvers with the capability of large-scale simulations. These solvers are crucial in modeling material properties and behavior at the atomic scale, especially in polymer simulations. Custom Python scripts will be used to preprocess molecular dynamics simulations, ensuring setup and parameterization.

The next open-source finite element codes will be analyzed:

- Finite element preprocessor: Salome-MECA, TexGen, FreeCad, Gmesh, Python scripts. All these preprocessors are open source with various options for importing geometry model setup, and mesh generation. These open-source tools provide a

flexible and cost-effective solution to preprocess finite element models for both mechanical and thermal analysis.

- Finite element solver: Code-Aster, a powerful open-source solver for handling complex mechanical and thermal simulations. Its use will allow the project to run highly detailed and accurate simulations, particularly in advanced fields like structural integrity, heat transfer, and multi-physics analyses.
- Finite element postprocessor: Paraview is an advanced visualization tool for analyzing simulation results. Custom Python scripts can automate and extend the post-processing workflow, making it easier to analyze complex results and extract key insights efficiently.

5.8.3.1 Cloud-computing platforms

Databases dedicated to material mechanical properties or atomistic data.

- **MatWeb:** is a comprehensive materials property database for mechanical, thermal, electrical, and other material properties. It offers a wide range of materials, including metals, polymers, ceramics, composites, and more. The database is user-friendly and allows searching and filtering based on various material parameters.
- **NIST Materials Data Repository:** a collection of open-access data sets containing material properties.
- **Granta Materials Data:** a materials database with access to mechanical properties.
- **Open Material Property Search Engine:** Open-source platform that aggregates material data from various sources.
- **Materials Project:** a materials science platform with a collection of material data, including mechanical properties.

5.8.4 Implementation maps

The next figure provides a global overview of how the mechanical surrogate model is implemented. The starting point is the input from the user or an application that will provide the variables of the model. The data format and values of the variables will be checked, to avoid introducing an input out of the range for which the model has been trained. Then, the model map to the material property that will be the output of the process.

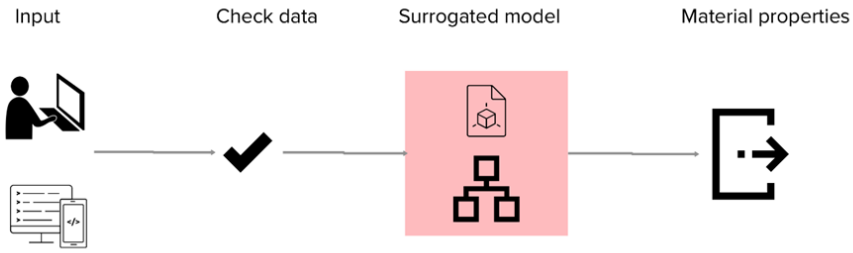


Figure 116: Implementation Map of Di^{MMS}

5.9 DIMAT MATERIALS PROCESSING SIMULATION (DI^{MPS})

5.9.1 Internal Architecture of Toolkits

DiMAT MPS, will use data of material transformation process, geometry, processing conditions and materials properties for recreate a virtual material processing, giving as output data that is important for the correct processing of the materials, such as presser, temperature, etc. This output will be used for increase database, and for optimize the transformation process, been the new parameters of the manufacturing, and feeding back into the process, to obtain new data.

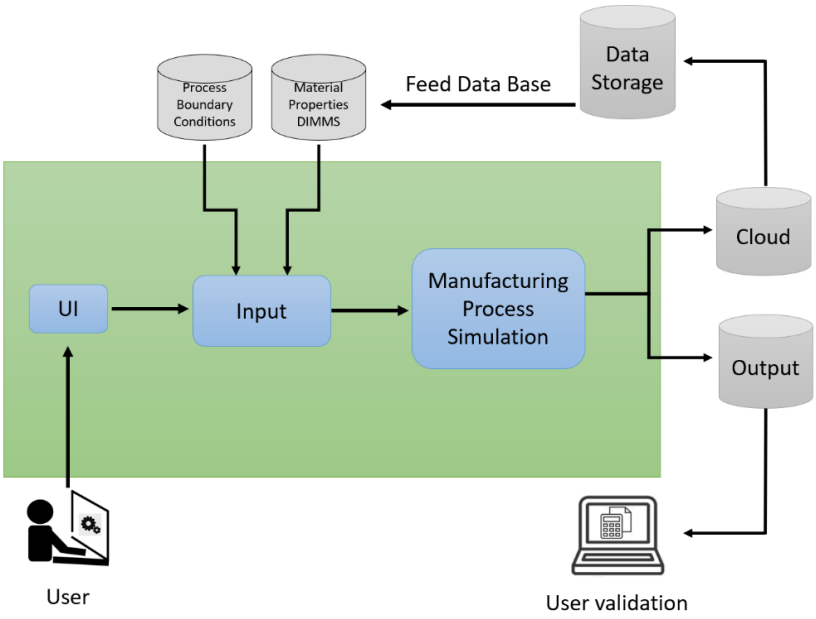


Figure 117: Internal Architecture of Di^{MPS}

5.9.2 Required components

5.9.2.1 Hardware Components

Different types of servers will be necessary for the correct development of the toolkit.

- **Database Servers:** this first type of database would be necessary to store the data related to the input of the simulations created for could be used as a base for new ones, and the output to create a database that facilitates the consultation of the results obtained.
- **Servers for simulation:** this second type of servers will be dedicated to executing finite elements calculations, as well as computational fluid dynamics, simulation process and generating predictions of material behaviour.

5.9.2.2 Data Storage

For a better understanding of the data, a cloud-based and collaborative approach with the other DiMAT project partners would be desirable. Providing the user with a complete and high value material study.

- **Simulation Data:** a repository of input; geometries most used in simulations, recommended mesh type, parameter conditions, as well as the output of previous simulations is considered necessary to facilitate the use of the toolkit.

5.9.2.3 Data Analysis

5.9.2.3.1 Data Flows

Depending on the type of material and transformation process the user will be asked to enter a type of input required for the process simulation. The computational fluid dynamics calculations will be selected depending on type of fluid, resulting in output with the principal parameter of process and the behaviour of the material during processing.

5.9.2.3.2 Data Preprocessing

Input data should be validated before pre-processing, to prevent a simulation with incorrect results. The simulation processes defined, as well as the materials processed will consist of

a predefined range of parameters that will correspond to the constraints and requirements of each process and material.

5.9.2.3.3 Data Analytics

The output data generated by the simulation and those obtained from the physical processes of material transformation will be used to feedback, the toolkit and the artificial neural networks. In this way, to achieve a better analysis of the results that allows to make predictions, creating a process simulator that supports the user.

5.9.3 Technologies employed

5.9.3.1 Toolkits employed (e.g., Python Libraries, other software)

- Python libraries will be used for the correct functionality:
  - Data management: Pandas, etc.
  - Visualizations generation: e.g., Matplotlib, etc.
  - Numerical packages: numpy, scipy.
- Machine-learning tasks: Pytorch, TensorFlow. As open-source libraries, they are among the most popular for their high level of documentation, high-performance computing with self-differentiation capability, and with the ability to be customizable and extensible.
- Finite element preprocessor: Salome-MECA, FreeCad, Elamx, PUMA. All these preprocessors are open source with a wide range of functions, including the capacity of importing and exporting models and geometry in different formats. In addition, they are freeware capable of being used in different platforms.
- Finite element solver: OpenFOAM, Calculix. These are high-performance toolboxes for numerical solvers with the capacity for handling computational fluid dynamics problems. Due to its characteristics and being an open-source software, it is freely available, which enables it to be backed by a vast number of users.
- Finite element postprocessor: Paraview. It is a license-free software with high adaptability and process efficiency that allows powerful visualization of simulation results, offering characteristics such as parallel processing and handling large amounts of data.

This software and technologies adhere to what was described in D2.2 for benchmarking. They are flexible and intuitive platforms for being operated and possess high data consistency.

5.9.3.2 Cloud-computing platforms

Databases dedicated to materials behavior or process parameters.

5.9.4 Implementation maps

Figure 118: Implementation Map Di^{MPS}

5.10 DIMAT DIGITAL TWIN FOR PROCESS CONTROL (DI^{DTPC})

5.10.1 Internal Architecture of Toolkits

The Virtual Object (VO) is considered as a virtual counterpart of an IoT device. It supports interaction with the IoT device and a set of IoT management functions. It can act as a Digital Twin (DT) based on the support of a set of virtual functions. The VO can be deployed at the edge part of the computing infrastructure.

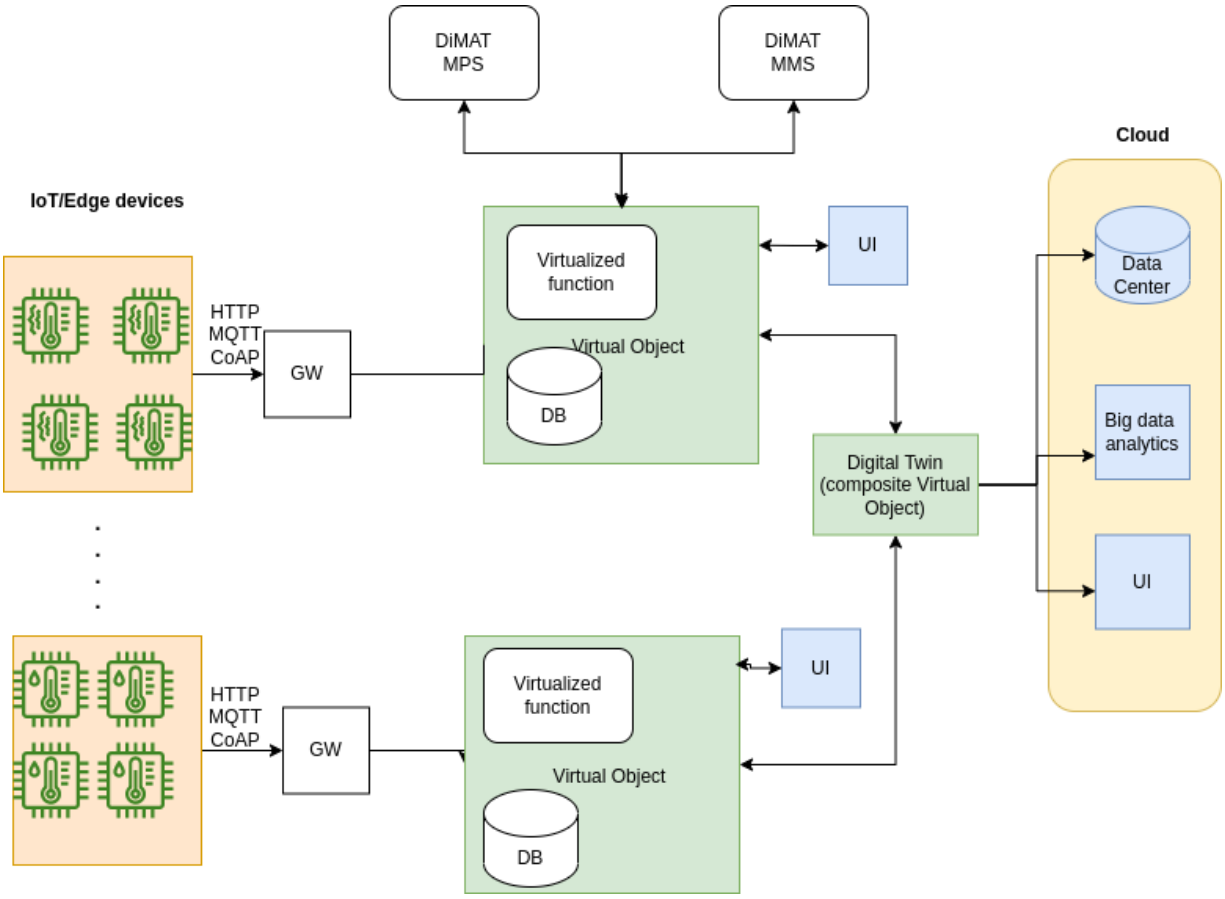


Figure 119: Digital Twin for Process Control Component Map

The above figure presents the internal architecture of the DTPC toolkit. The architecture is split into three distinct tiers each aiming to address a specific functionality. These tiers are the Network Edge, the platform, and the cloud layer. The devices embedded in the mechanical equipment generate measurements that aid in identifying the current state of the equipment (Physical Twin).

This data is either directly transmitted to the DT platform or aggregated via specialized network equipment. Then, they are given as input to the virtual functions specified in the DT platform. The current state of the equipment is stored in a local time-series DB such as InfluxDB. Human users can view information about the Digital Twin or issue commands through appropriately created User Interfaces (UI). Finally, Cloud based solutions offer long-term data storage as well as access to big data analytics algorithms and software libraries. Data flows among these components ensure the correct operation of the toolkit and assure the successful data analysis of the polled data from the edge as well as the control mechanisms for the physical twin.

5.10.2 Required components

5.10.2.1 Hardware Components

5.10.2.1.1 Edge Devices

- **Sensors** embedded in the Network Edge collect real-time data from the physical device concerning parameters such as temperature, humidity, pressure, energy consumption or environmental conditions and transmit them to the Digital Twin for analysis and simulation purposes. Through the continuous gathering of physical and environmental data, they enable real-time monitoring of the conditions, behavior, and performance of the physical system, detect anomalies and malfunctions and facilitate the prediction of the future state of the physical device. Moreover, they can apply some initial data processing of the collected information locally at the edge before transmitting them to the Digital Twin, reducing the consumption of the Platform resources and enhancing data quality.
- **Actuators** perform real-time and remote control of the physical device by triggering actions introduced by results and observations of the Digital Twin or by direct commands applied by a user of the system. They can correct detected anomalies and risks of the physical system though emergency instructions defined by the Platform and adjust functional and control parameters, to ensure safety, health and maintenance of the system. They can also manage energy consumption of the system and contribute to the improvement of simulation processes of the Digital Twin by enforcing the recommended actions to the physical twin, enabling the platform to compare the resulting performance of the physical device with the simulated behavior and reassess the employed methods and algorithms.
- **Programmable Logic Controllers (PLC)** are programmed and customized to control and monitor specific industrial and manufacturing processes and machinery and serve as an intermediate concerning the data exchange between the Digital Twin Platform and IoT devices such as sensors and actuators. Through their programmed control logic, they coordinate the performance of the Digital Twin and the physical system by transmitting input data from sensors to the Digital Twin and receiving instructions and control signals from the platform to determine the appropriate actions to be executed by actuators.
- **Radio-Frequency Identification (RFID)** tags offer unique identification of physical objects of the system, enabling real-time tracking and monitoring. By

combining RFID data with data collected from sensors, the Digital Twin is provided with a more accurate and synchronized representation of the physical object.

- **Raspberry Pi's** or other boards could also be employed for data acquisition, processing, and analytics purposes of sensor data at the edge and data transmission and bi-directional communication with the Digital Twin. They are also useful for rapid testing and prototyping of the Digital Twin concepts before deploying them at a larger scale.

5.10.2.1.2 Servers

Various types of servers can be employed in the development of the Digital Twin Platform.

- **Database Servers** can handle data for short and long-term storage related to the DTPC toolkit and provide mechanisms for querying and efficient access. They also support structured data representation, provide security and access control and high-performance management of big volumes of data.
- **Servers** for simulations are expected to be employed in the DTPC toolkit in order to run simulation models, implement algorithms and generate predictions concerning the behavior of the physical system.

5.10.2.2 Connectivity Devices

5.10.2.2.1 Gateways

Gateways serve as aggregation points of data collected from multiple sensors or other edge devices, providing preprocessing and initial filtering capabilities before transmitting them to the Digital Twin. They provide continuous bi-directional communication and reliable connection between the Edge and the Digital Twin Platform, support different communication protocols for interacting with the edge devices and enable the conversion of data into standard format compatible with the Digital Twin Platform. Gateways also contribute to the security of the system by supporting authentication and data encryption capabilities and could also be employed as short-term storage of data at the edge.

5.10.2.3 Data Storage

5.10.2.3.1 Short-term

- **Real-time data**, involving sensor measurements of the parameters, the current state of the physical system and short-term calculations of the virtual functions should be stored in local time-series databases inside the Digital Twin that are periodically updated.
- **Configuration data**, such as system settings, control strategies, rules detailing the operation of the system or model parameters should be stored in a short-term configuration management system that needs to be frequently updated to perform punctual operational adjustments.
- **Edge computing data**, such as data related to local analytics and actions performed at the edge, could be stored in gateways or edge servers for caching and quick accessing of data before transmitting them to the Digital Twin Platform. Temporary storage of data at the edge could also be employed when connection with the centralized system is unavailable or unreliable.

5.10.2.3.2 Long-term

- **Historical data**: Data previously stored in the local time-series DB of the Digital Twin are transferred after a predefined amount of time to a long-term data storage system such as a set of relational databases, data lakes or data warehouses.
- **Simulation data** concerning algorithms, parameters, results and configurations should be stored in specialized storage systems such as simulation data management platforms.

Data for long-term storage should be better allocated to the Cloud and in collaboration with the DiMAT Cloud Materials Database.

5.10.2.4 Data Analysis

5.10.2.4.1 Data Flows

- **Edge tier and Digital Twin Platform:** Data is generated through carefully embedded IoT devices in the equipment. These devices gather relevant

measurements that aid in describing the equipment's (Physical Twin) current state. These data are aggregated and transmitted via a gateway to the digital twin platform. Data is transmitted in a structured format (e.g., JSON, CSV, etc.). Besides these data flows, commands can be issued from the Virtual Object to actuators installed in the equipment for some action to take place (e.g., terminate process). These commands are also sent in structured format.

- **Digital Twin platform and cloud:** Data flows among the components of the DT platform. To begin with, data stored in the local DB of the platform are queried and fed to any Virtual Functions executed in the cloud to be analyzed. Vice versa, data produced by virtual functions of the cloud can be stored in the DB. Moreover, the local DB can offload the data in a cloud storage solution to enrich the historical data collected that may be required for virtual functions. Also, for computationally expensive tasks that are executed in cloud-based platforms, data should be fed from the Cloud based storage and potentially from the local DB through APIs.
- **Digital Twin platform and other DiMAT toolkits:** Data flows among the digital twin and other toolkits developed in DiMAT to support specialized virtual functions such as those to material processes simulation. Integration with the other two tools of the Simulation and Optimization Suite is necessary.

Data flows among components			**Manner of transmission**
Component 1	**Directionality**	**Component 2**	
IoT edge devices	Bi-directional	Gateway	MQTT, HTTP, CoAP
Gateway	Bi-directional	DB	MQTT, HTTP, CoAP, etc.
DB	Bi-directional	Virtual Functions	Custom APIs if hosted in the same equipment (e.g., server), Transmission protocols like MQTT.
Virtual Object	Bi-directional	User Interface (local or cloud based)	Queries to the database over suitable APIs
DB	One directional	Cloud DB	FTP, Cloud storage APIs
Virtual Object	Bi-directional	Cloud Analytics	APIs provided by such platforms, FTPs

Digital Twin	Bi-directional	Other toolkits	Custom developed APIs

Table 116. Data Flow of Di^{DTPC}

Below the data flows of the DiMAT's **DTPC** toolkit can be seen alongside the indicative ways they can be transmitted.

5.10.2.4.2 Data Preprocessing

The Gateway equipment may need to pre-process the data acquired from the IoT devices deployed in the edge-tier to remove possible null or corrupted values before transmitting them to the Digital Twin Platform.

5.10.2.4.3 Data Analytics

In the DTPC toolkit data analysis is performed in the virtual functions that can be executed locally in the platform and in cloud analytics platforms that offer more computational resources. In the table below some indicative functions relative to the Digital Twin's operation are described.

Virtual Function (local or cloud based)	Details
Real time analytics	Analysis of current state to predict future states, anomaly detection, assist in the equipment's maintenance.
Historical data analysis	Analysis of past data to discover correlations, extract information about optimal parameter selection, etc.
Materials Simulation	Access specialized software to simulate material processes. Outputs of these data can be used for setting parameters in the real equipment as well as executing simulation scenarios without the need to execute them in the real world.
Visualization functions	Access to both current and past data allows for creating figures and charts aiming to give the human user a better understanding of the operation of the twins.

Table 117. Data Analyst of Di^{DTPC}

5.10.3 Technologies employed

5.10.3.1 Toolkits employed (e.g., Python Libraries, other software)

The development of the DTPC toolkit will be based on open-source libraries, suitable for its goals, that exhibit strong performance and can handle the use-cases in which the toolkit will be employed. In the following, some indicative tools are presented alongside necessary justification for their selection.

- Virtual Object Software Stack (made available from the NEPHELE project), also evaluated in D2.2. Nephele is an open-source software stack, meaning that it has publicly available source code. Nephele's online repository statistics underline that the last commit was realized in September 2024 in a GitLab repository highlighting the fact that it is very well-mainteined Moreover, Nephele supports all following protocols: HTTP, WebSocket, CoAP, MQTT. Lastly, it is a lightweight software stack not at all demanding in requirements such as RAM and Proc/or Cores. Finally, Nephele VO is developed by NTUA, also one of DiMAT partners. Usage of Eclipse Ditto was also considered but despite the benefit of providing a user-interface, at the time it did not offer out of the box support for storing and handling time-series data contrary to Nephele.
- Python libraries can be used for developing algorithms for the following functionalities. All the mentioned python libraries are popular, with non-excessive computing requirements and are open-source:
    - Forecasting: e.g., Prophet, statsmodels, etc.
    - Visualizations generation: e.g., Matplotlib, seaborn, plotly, etc.
    - Machine-learning tasks: e.g., TensorFlow
- Materials analysis: pytmagen
- Time-series databases for storing the current state of the Physical Twin: e.g., InfluxDB. which is an open-source time series database used for storage and retrieval of time series. It is a very well regarded and widely used database with a very active online community and a constantly updated online repository. The repository's statistics mention 28.8k stars and 3.5k forks.
- SQL databases for storing configuration data (e.g., MySQL).
- Materials Simulation Software: e.g., OpenFOAM, SALOME, Digimat, etc. These are popular libraries that can be employed if required for specific use-cases of the pilots.
- User Interface design

5.10.3.2 Cloud-computing platforms

Platforms dedicated to analyzing materials related data in the cloud, which can be used for storing long-term data if required:

- **I4QBDA:** Optimization of Manufacturing processes through data-driven predictions.
- **OpenBabel:** Toolbox targeted to analysis of chemical related data.
- **RDKit:** Open-source cheminformatics and machine learning software.

5.10.4 Implementation maps

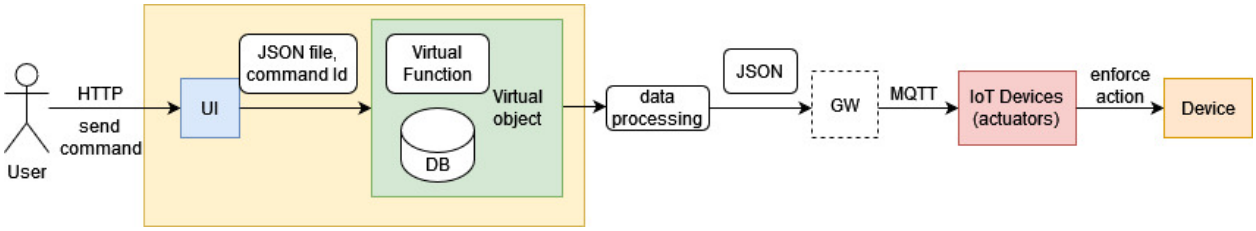


Figure 120: Implementation Map of Di^{DTPC}

The above figure provides a high-level overview of how a defined equipment control activity (send a command) issued by a user towards the physical twin (device) is implemented. First, the user interacts with the UI interface via an API such as HTTP and selects the appropriate option specifying the device and the command from a pre-defined list of available commands (e.g., shut-down, restart, increase speed, etc.). The UI passes along the information following a structured data format (JSON) and a suitable virtual function is employed to pass along the data (probably by adding more information to the JSON file, like IP addresses). Then, optionally, a Gateway receives the message and passes it along to the correct IoT device (I.e., an actuator) that enforces the corresponding action to the device.

6 CONCLUSIONS

The development of this deliverable comprises a set of tasks (T3.2, T3.3, T3.4 and T3.5) that aim to provide the design of the different viewpoints following the IIRA Reference Architecture. The viewpoints will provide a framework for iterative reflection on architectural issues that may arise during their conception, from the business point of view to the final implementation.

This deliverable has been developed in such a way that it serves as a starting reference point for the beginning of the development of the DiMAT project solutions. In such a way, it allows the initiation of decision-making both at the technical level and up to the interaction with the end user, allowing a better definition of the technical requirements of the solutions, which will indirectly influence the requirements of the pilot.

7 REFERENCES

[1] The Industrial Internet of Things - Volume G1: Reference Architecture. Tratto da Industrial Internet Consortium (2019): https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf

[2] https://www.figma.com/

[3] https://es.react.dev/

[4] https://play.grafana.org/

[5] https://allegrograph.com/allegrograph-editions/

APPENDIX A

Nº	Comment	
1º	Some figures are blurred and should be replaced by new ones with better quality (e.g. Figure 1, Figure 2, etc.).	
1.1	Figure 1 Change	See LIST OF FIGURES for pages.
1.2	Figure 2 Change	
1.3	Figure 4 Change	
1.4	Figure 12 Change	
1.5	Figure 13 Change	
1.6	Figure 16 Change	
1.7	Figure 39 Change	
1.8	Figure 40 Change	
1.9	Figure 44 Change	
1.10	Figure 45 Change	
1.11	Figure 83 Change	
1.12	Figure 84 Change	
1.13	Figure 104 Change	
1.14	Figure 105 Change	
1.15	Figure 106 Change	
2	Business viewpoint - it should be clarified if and how the Key Objectives (KO) relate to the KPIs. In case they are related, it should be better justified how will a Key Objective of Reduction of time-to-market by 20%-30% allow for a Reduction of time-to-market by 45% in Pilot 1.	Paragraph 2.2.2
3	User viewpoint - it is important to ensure that one of the main aspects of the architecture is kept (i.e.unified interface). Also, providing the users with	The necessary updates have been made to each of the solutions within the 'User Viewpoint' section considering security or other privacy elements (paragraphs 3.1.2.2, 3.1.2.2.5) In addition, consideration has been given to unifying the interface as far as possible for each of the solutions. For the specific phrase "all data must comply with security and privacy rules" explanation has been provided in paragraphs 3.10.3.1.3, 3.10.3.2.3 for the DTPC toolkit.

	appropriate guidance is key as the actions involve a high complexity level (e.g. selection of metadata, ranges for inputs, etc). In addition, it should be clarified what is meant by "All data must comply with security and privacy rules" and how and where this is checked.	
4	Functional viewpoint - and particularly the general architecture of the components, it is important to also reflect on the interactions between components.	In each of the solutions, within the 'General architecture' section (paragraphs 4.2.1, 4.3.1,4.4.1, 4.5.1,4.6.1,4.7.1, 4.8.1, 4.9.1, 4.10.1) the necessary updates have been made to reflect the interactions with the components of the reference architecture.
5	Implementation viewpoint it is not clear how where the technologies selected since some of them were not evaluated in D2.2 and some that were, are not being referred here.	The 'Technologies employed' section in each of the solutions has been updated and improved (paragraphs 5.1.2, 5.2.3, 5.3.3, 5.4.3, 5.5.3, 5.6.3, 5.7.3, 5.8.3, 5.9.3. 5.10.3) More specifically, for the technologies that were not mentioned in D2.2 there is proper justification for their selection based on the same criteria that the technologies included in D2.2 are evaluated.

Table 118. Changes in the deliverable based on the reviewers' comments

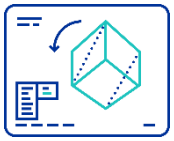
APPENDIX B

TOOLKIT SELECTION

The DiMAT Project is set to create three distinct DiMAT Suites. Each suite will contain three individual toolkits, making up a total of nine toolkits that represent the actionable outcomes of the project:

The **Data and Assessment Suite** is designed to enhance material data safety and traceability, promote the usage of materials derived from renewable resources and boost personnel digital competency. Additionally, it aims to decrease material design costs, economic and environmental impact of materials, and shorten the time to market. The Suite will contain the:

- DiMAT Cloud Materials Database – Di^{CMDB}: system for storing, sharing, and exploration of relevant material data for the material design, processing, and manufacturing processes.
- DiMAT Knowledge Acquisition Framework – $Di^{KAF:}$ toolkit for representing and managing information related to the materials along with their characteristics and their relationships in the form of a Knowledge Graph (KG).
- DiMAT Materials Environmental and Cost Life Cycle Assessment – $Di^{MEC\cdot LCA}$: tool for providing a high-level assessment on the environmental and economic impact of the pilot cases.



The **Modelling and Design Suite** aims to enhance material designs and optimize the return on investment (ROI) for personnel training. It's also designed to reduce material design errors and the use of materials during the design and modelling phase, and to boost personnel productivity. The Modelling and Design Suite will include:

- DiMAT Materials Design Framework – Di^{MDF}: ontology-based open knowledge system to support the material design process, an App running on DiMAT Open Cloud Materials.
- DiMAT Materials Modeler – Di^{MM}: toolkit to assist in discovering and designing competitive materials rapidly and effectively by modelling.
- DiMAT Materials Designer – Di^{MD}: tool for providing a high-level assessment on the environmental and economic impact of the pilot cases.

The DiMAT Simulation and Optimization Suite is designed to enhance material mechanical properties and performance, increase on-time completion rates and improve material operational characteristics. It also aims to reduce the amount of material needed for testing and prototyping

procedures and to cut down material production costs. The Simulation and Optimization Suite includes:

- DiMAT Materials Mechanical Properties Simulator Di^{MMS}: toolkit for determining numerically mechanical properties such as stiffness, tensile strength, plasticity, viscoelastic and viscoplastic properties, damage, fracture, fatigue, etc.
- DiMAT Materials Processing Simulator Di^{MPS}: toolkit for determining manufacturing conditions and concepts while simulating their application, results and requirements, in each one of the materials, processes, and processing conditions found at the DiMAT Open Cloud Materials Database.
- DiMAT Digital Twin for Process Control Di^{DTPC}: digital counterpart of the IoT devices considered within DiMAT that provides a set of abstractions, virtualized functions, and APIs to support real and simulated manufacturing processes.

Please indicate the degree of interest your organization has in the toolkits to be developed in the DiMAT project by marking with an X, where 1 is not interested at all and 5 is very interested.

Suite	Toolkit	1	2	3	4	5
Data and Assessment Suite	Cloud Material Database					
	Knowledge Acquisition Framework					
	Materials Environmental and Cost Life Cycle Assessment					
Modelling and Design Suite	Material Design Framework					
	Materials Modeler					
	Materials Designer					
Simulation and Optimisation Tool	Materials Mechanical Properties Simulator					
	Materials Processing Simulator					
	Digital Twin for Process Control					

Table 119. Degree of interest in the toolkits

BUSINESS PERSPECTIVE

The objective of this part is to identify the vision, values and key objectives for each of the toolkits selected with a mark higher than or equal to 3 in the previous part.

- Vision: Please describe the vision and future state of organization after implementing the Toolkit. How will the toolkit influence the strategy and competitiveness of organization?
- Values: What would be the value for the organization by implementing the toolkit?
- Key objectives: List of quantifiable high-level business outcomes expected of this toolkit in the context of delivering the values. For example: cost reduction by xx%, errors reduction by xx%, reduction of environmental impact by xx%, increase of personnel productivity by xx%, increase of performance by xx% ...
- Process to focus on: List of the processes involved in the achievement of the key objectives previously defined.

This part needs to be answered by a person with a business profile. Please, complete a table for each of these toolkits. Add more tables in case of being necessary.

Toolkit name:	
Vision:	
Values:	
Key objectives:	
Processes to focus on:	

Toolkit name:	
Vision:	
Values:	
Key objectives:	
Processes to focus on:	

Toolkit name:	
Vision:	
Values:	
Key objectives:	
Processes to focus on:	

Toolkit name:	
Vision:	
Values:	
Key objectives:	
Processes to focus on:	

Toolkit name:	
Vision:	
Values:	
Key objectives:	
Processes to focus on:	

TECHNICAL PERSPECTIVE

The objective of this part is to identify the technical key objectives and fundamental capabilities for each of the toolkits selected with a mark higher than 3 in the previous part.

- Key objectives: List of quantifiable technical outcomes expected of this toolkit in the context of delivering the values. For example: cost reduction by xx%, errors

reduction by xx%, reduction of environmental impact by xx%, increase of personnel productivity by xx%, increase of performance by xx% ...

- Fundamental capabilities: High-level specifications of the essential ability of the toolkit to complete the technical objectives tasks. Please, identify how implementation of the Toolkit may influence on new technical capabilities.
- Alignment: Do these new capabilities satisfy the previously identified business and technical key objectives?

This part needs to be answered by a person with a technical profile. Please, complete a table for each of these toolkits. Add more tables in case of being necessary.

Toolkit name:	
Technical key objectives:	
Fundamental capabilities:	
Alignment	

Toolkit name:	
Technical key objectives:	
Fundamental capabilities:	
Alignment	

Toolkit name:	
Technical key objectives:	
Fundamental capabilities:	
Alignment	

Toolkit name:	

Technical key objectives:	
Fundamental capabilities:	
Alignment	

Toolkit name:	
Technical key objectives:	
Fundamental capabilities:	
Alignment	