Ref. Ares(2025)846896 - 04/02/2025

DiMAT

D3.4 VIEWPOINTS V2

04/02/2025

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

D3.4 VIEWPOINTS V2

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](https://cordis.europa.eu/project/id/101091496)		
Work package	WP3 - DESIGN: DiMAT Framework Design		
Deliverable	D3.3 – DiMAT Viewpoints v2		
Task	T3.2 – Business Viewpoint T3.3 – Usage Viewpoint T3.4 – Functional Viewpoint T3.5 – Implementation Viewpoint		
Due date	01/10/2024		
Submission date	04/02/2025		
Nature	Report	Dissemination level	Public
Deliverable lead	2-UPV		

Version	1.3
Authors	Miguel A. Mateo-Casalí (UPV), Andres Boza (UPV), Ana Esteso Álvarez (UPV), Llanos Cuenca (UPV), Mareva Alemany (UPV), Raul Poler (UPV), Konstantinos Tsitseklis (NTUA), Jorge Doménech (AITEX)
Contributions	Konstantinos Tsitseklis (NTUA), Jorge Doménech (AITEX)
Reviewers	Anastasios Zafeiropoulos (NTUA), Christian Grunewald (DIN)
Abstract	This document presents the second version of the different views of the DiMAT Reference Architecture (RA). Four views (Business, Usability, Functional and Implementation) have been updated and more information has been added.
Keywords	IIRA, Viewpoints, Reference Architecture, Design

Document Revision History

Version	Date	Description of change	List of contributor(s)
0.1	19-May-2023	ToC	UPV
0.2	25-May-2024	1st Draft available for internal review	UPV, AITEX, NTUA, CETMA, ROPARDO
0.3	04-Jun-202	Internal Review	NTUA, DIN
0.4	10-Jun-2024	2nd Draft addressing the comments from internal reviewers	UPV
1.0	28-Jun-2024	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	29-Jan-2025	Additional update of the deliverable according to PO/reviewers' comments and quality check	CERTH, UPV
1.3	04-Feb-2025	Final quality check and issue of final document	CERTH, UPV

DISCLAIMER

COPYRIGHT NOTICE

EXECUTIVE SUMMARY

This document presents the second version of the DiMAT solution design phase viewpoints. These viewpoints are Business, Usability, Functional, and Implementation.

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 would interact with the solutions, identifying several key points from a usability perspective, such as tasks, activities, or functions, always considering the human factor that would interact with these tools.

The third part of this deliverable analysed the functional part of the DiMAT solutions. For this purpose, and focusing on the development part of the solutions, their structure, interrelations, and interfaces were considered.

Finally, in the implementation part, we technically described the different components of the DiMAT solutions and how they were interconnected and selected 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 128 can be found in the appendix, providing an overview of the revisions and additions for easy reference.

TABLE OF CONTENTS

1 METHODOLOGICAL APPROACH 21
2 INTRODUCTION 25
2.1 Viewpoints 25
2.2 DiMAT RA: Viewpoints 27
3 BUSINESS VIEWPOINT 28
3.1 Business viewpoint 1st approach for DiMAT 28
3.1.1 Business Viewpoint Concepts, Methodology and Questionnaire 28
3.1.2 DiMAT Business Viewpoint 1st Approach 30
3.2 Business viewpoint 2nd approach 32
3.2.1 Business Viewpoint Methodology 32
3.2.2 Business Viewpoint Questionnaire 32
3.3 DIMAT RESULTS Business viewpoint 2nd approach 34
3.3.1 Suitability and Expected Business Impact 35
3.3.2 Difficulties for their implementation 47
3.3.3 Costs perspective 57
3.4 DIMAT Business viewpoint 2nd approach CONCLUSIONS 58
4 USAGE VIEWPOINT 60
4.1 Usage VIEWPOINT approach for DiMAT 60
4.1.1 Usage Viewpoint Concepts 60
4.1.2 Usage Viewpoint Methodology 61
4.1.3 Overview 61
4.1.4 Roles 61
4.1.5 Activities 62
4.1.6 Functional map 63

4.2 DiMAT Cloud Materials Database (Di^{CMDB}) 65
4.2.1 Activities 65
4.3 DiMAT Knowledge Acquisition Framework (DiKAF) 70
4.3.1 Activities 70
4.4 DiMAT Materials Environmental and Cost Life Cycle Assessment ($Di^{MEC \cdot LCA}$) 75
4.4.1 Overview 75
4.4.2 Activities 76
4.5 DiMAT Materials Design Framework (Di^{MDF}) 87
4.5.1 Activities 87
4.6 DiMAT Materials Modeler (DIMM) 92
4.6.1 Activities 93
4.7 DiMAT Materials Designer (Di^{MD}) 99
4.7.1 Roles 99
4.7.2 Activities 99
4.8 DiMAT Materials Mechanical Properties Simulator (DI^{MMS}) 104
4.8.1 Overview 104
4.8.2 Roles 104
4.8.3 Activities 105
4.9 DiMAT Materials Processing Simulator (Di^{MPS}) 107
4.9.1 Overview 108
4.9.2 Roles 108
4.9.3 Activities 108
4.10 DiMAT Digital Twin for Process Control (Di^{DTPC}) 112
4.10.1 Activities 112
5 FUNCTIONAL VIEWPOINT 119
5.1 FUNCTIONAL VIEWPOINT approach for DiMAT 119
5.1.1 Functional Viewpoint Concepts 119
5.1.2 Functional Viewpoint Methodology 119

5.1.3 General Architecture....120
5.1.4 Functional components 120
5.1.5 Interactions 121
5.1.6 Mapping of functional components to IIRA functional domain....121
5.1.7 Functional components, functional structure and requirements interactions 121
5.1.8 [Functional component name] 121
5.2 DiMAT Cloud Materials Database (Di^{CMDB}) 124
5.2.1 GUI 124
5.2.2 KItem service....125
5.2.3 User validation 127
5.3 DiMAT Knowledge Acquisition Framework (DI^{KAF})....137
5.3.1 Functional components, functional structure and requirements interactions 137
5.4 DiMAT Materials Environmental and Cost Life Cycle Assessment (DI$^{MEC \cdot LCA}$) 142
5.4.1 General Architecture....142
5.4.2 Functional components 142
5.4.3 Interactions 144
5.4.4 Mapping of functional components to IIRA functional domain....147
5.4.5 Functional components, functional structure and requirements interactions 148
5.5 DiMAT Materials Design Framework (DI^{MDF}) 154
5.5.1 General architecture....154
5.5.2 Functional components, functional structure and requirements interactions 155
5.6 DiMAT Materials Modeler (DI^{MM})....163
5.6.1 Functional components, functional structure and requirements interactions 163
5.7 DiMAT Materials Designer (DI^{MD}) 173
5.7.1 Functional components, functional structure and requirements interactions 173
5.8 DiMAT Materials Mechanical Properties Simulator (DI^{MMS}) 177
5.8.1 General Architecture....177
5.8.2 Functional components 177

5.8.3 Interactions 179
5.8.4 Functional components, functional structure and requirements interactions 179
5.9 DiMAT Materials Processing Simulator (DI^{MpS}) 185
5.9.1 General Architecture.. 185
5.9.2 Functional components 185
5.9.3 Interactions 188
5.9.4 Mapping of functional components to IIRA functional domain............................. 188
5.9.5 Functional components, functional structure and requirements interactions 189
5.10 DiMAT Digital Twin for Process Control (DI^{dtpc}).. 195
6 IMPLEMENTATION VIEWPOINT... 202
6.1 Implementation Viewpoint approach for DiMAT 202
6.1.1 Implementation Viewpoint Concepts.. 202
6.1.2 Implementation Viewpoint Methodology.. 203
6.2 DiMAT Cloud Materials Database (Di^{CMDB}) 207
6.2.1 Data handling.. 207
6.2.2 Technologies employed.. 209
6.2.3 Implementation maps... 209
6.2.4 User Manuals 210
6.2.5 Deployment PATTERN.. 210
6.3 Dimat knowledge acquisition framework (Di^{KAF}) 211
6.3.1 Data handling.. 212
6.3.2 Technologies employed.. 213
6.3.3 Implementation maps... 214
6.3.4 User Manuals 215
6.3.5 Deployment PATTERN.. 215
6.4 Dimat materials environmental and cost life assessment (DI$^{MEC-LCA}$) 216
6.4.1 Data handling.. 217
6.4.2 Technologies employed.. 217

6.4.3 Implementation maps 218
6.4.4 User Manuals 218
6.4.5 Deployment PATTERN 219
6.5 Dimat Materials Design Framework (Di^{MDF}) 220
6.5.1 Technologies employed 220
6.5.2 Implementation maps 221
6.5.3 User Manuals 222
6.5.4 Deployment PATTERN 223
6.6 Dimat Materials Modelling and Behaviour Prediction (Di^{MM}) 223
6.6.1 Data handling 224
6.6.2 Technologies employed 225
6.6.3 Implementation maps 226
6.6.4 User Manuals 227
6.7 Dimat Materials Design, Characterization and Synthesis (Di^{MD}) 228
6.7.1 Data handling 229
6.7.2 Technologies employed 231
6.7.3 Implementation maps 232
6.7.4 User Manuals 233
6.8 Dimat materials mechanical properties simulator (Di^{MMS}) 235
6.8.1 Data handling 236
6.8.2 Technologies employed 237
6.8.3 Implementation maps 238
6.8.4 User Manuals 238
6.8.5 Deployment PATTERN 240
6.9 Dimat Materials Processing Simulation (Di^{MPS}) 240
6.9.1 Data handling 241
6.9.2 Technologies employed 242
6.9.3 Implementation maps 243

6.9.4 User Manuals 244
6.9.5 Deployment PATTERN.. 245
6.10 Dimat Digital Twin for Process Control (Di^{DTPC}).. 246
6.10.1 Data handling... 247
6.10.2 Technologies employed.. 248
6.10.3 Implementation maps 249
6.10.4 User Manuals 250
6.10.5 Deployment PATTERN.. 252
7 CONCLUSIONS... 254
8 REFERENCES.. 255
APPENDIX A.. 256
APPENDIX B.. 258
TOOLKIT SELECTION 258
BUSINESS PERSPECTIVE.. 260
TECHNICAL PERSPECTIVE 261

LIST OF FIGURES

FIGURE 1. IIRA ARCHITECTURE FRAMEWORK........ 25
FIGURE 2. DIMAT BUSINESS VIEWPOINT 29
FIGURE 3. DIMAT SUITES AND TOOLKITS........ 30
FIGURE 4. IMPORTANCE-POTENTIAL QUADRANTS 36
FIGURE 5. EXPRESSIONS OF INTEREST IN SUITE 1........ 36
FIGURE 6. EXPRESSIONS OF INTEREST IN SUITE 1 BY TYPE OF MATERIAL........ 37
FIGURE 7. AVERAGE IMPROVEMENT POTENTIAL........ 38
FIGURE 8. INTEREST IN SUITE 1........ 39
FIGURE 9. EXPRESSIONS OF INTEREST IN SUITE 2........ 39
FIGURE 10. EXPRESSIONS OF INTEREST IN SUITE 2 BY TYPE OF MATERIAL........ 40
FIGURE 11. INTEREST IN SUITE 2........ 41
FIGURE 12. EXPRESSIONS OF INTEREST IN SUITE 3 42
FIGURE 13. EXPRESSIONS OF INTEREST IN SUITE 3 BY TYPE OF MATERIAL........ 43
FIGURE 14. INTEREST IN SUITE 3........ 44
FIGURE 15. INTEREST IN SUITES........ 44
FIGURE 16. COMPANIES NEEDS THAT CAN BE COVERED BY SUITES. 46
FIGURE 17. ORGANIZATIONAL DIFFICULTIES BY SUITES........ 49
FIGURE 18. ORGANIZATIONAL DIFFICULTIES BY SUITES AND COMPANY SIZE........ 52
FIGURE 19. TECHNICAL DIFFICULTIES BY SUITES. 54
FIGURE 20. TECHNICAL DIFFICULTIES BY SUITES AND COMPANY SIZE........ 56
FIGURE 21. USAGE VIEWPOINT LAYER........ 60
FIGURE 22. USE CASE DIAGRAM OF DI$^{MEC\text{-}LCA}$ 76
FIGURE 23. MOCKUP “STORE USER INPUT " OF DI$^{MEC\text{-}LCA}$ 77
FIGURE 24. MOCKUP "VISUALIZE LCA RESULTS OF DI$^{MEC\text{-}LCA}$........ 78
FIGURE 25. MOCKUP “VISUALIZE KPIS” OF DI$^{MEC\text{-}LCA}$........ 80
FIGURE 26. MOCKUP "DOWNLOAD DATA" OF DI$^{MEC\text{-}LCA}$ 82
FIGURE 27. MOCKUP “LOG IN" OF DI$^{MEC\text{-}LCA}$........ 83

Co-funded by
the European Union

FIGURE 28. MOCKUP “MODIFY VISUALIZATION " OF DI$^{MEC-LCA}$ 84
FIGURE 29. MOCKUP “EDIT DATA INLINE" OF DI$^{MEC-LCA}$ 86
FIGURE 30. MOCKUP HOME PAGE OF DI^{MD} 100
FIGURE 31. MOCKUP MATERIAL DEFINITION OF DI^{MD} 100
FIGURE 32. MOCKUP RESULTS REVIEW OF DI^{MD} 102
FIGURE 33. GENERAL ARCHITECTURE EXAMPLE 120
FIGURE 34. MATERIALS ENVIRONMENTAL AND COST LIFE CYCLE ASSESSMENT DIAGRAM 142
FIGURE 35. SEQUENCE DIAGRAM OF THE DIMEC-LCA TOOLKIT COMPONENT “UPLOAD DATA” 145
FIGURE 36. SEQUENCE DIAGRAM OF THE DIMEC-LCA TOOLKIT COMPONENT "VISUALIZE LCA RESULTS" 145
FIGURE 37. SEQUENCE DIAGRAM OF THE DIMEC-LCA TOOLKIT COMPONENT "VISUALIZE KPIS" 146
FIGURE 38. SEQUENCE DIAGRAM OF THE DIMEC-LCA TOOLKIT COMPONENT "LOG IN" 147
FIGURE 39: MATERIALS DESIGN FRAMEWORK DIAGRAM 154
FIGURE 40: MATERIALS MECHANICAL PROPERTIES SIMULATOR DIAGRAM 177
FIGURE 41. MATERIALS PROCESSING SIMULATOR DIAGRAM 185
FIGURE 42. INTERNAL ARCHITECTURE EXAMPLE 204
FIGURE 43. INTERNAL ARCHITECTURE DICMDB 207
FIGURE 44. INTERNAL ARCHITECTURE DIKAF 211
FIGURE 45. IMPLEMENTATION MAPS DIKAF 214
FIGURE 46. INTERNAL ARCHITECTURE DIMECLCA 216
FIGURE 47. IMPLEMENTATION MAPS DIMECLCA 218
FIGURE 48. INTERNAL ARCHITECTURE DIMM 223
FIGURE 49. INTERNAL ARCHITECTURE DIMD 229
FIGURE 50. IMPLEMENTATION MAP DIMD 232
FIGURE 51. INTERNAL ARCHITECTURE DIMMS 235
FIGURE 52. IMPLEMENTATION MAP OF DI^{MMS} 238
FIGURE 53. INTERNAL ARCHITECTURE DIMPS 241
FIGURE 54. IMPLEMENTATION MAP OF DIMPS 244

Co-funded by
the European Union

FIGURE 55. INTERNAL ARCHITECTURE DIDTPC.......... 247
FIGURE 56. IMPLEMENTATION MAP DIDTPC.......... 250
FIGURE 57. IMPLEMENTATION MAP 2 DIDTPC.......... 250

LIST OF TABLES

TABLE 1. COMPARISON OF D3.3 AND D3.4 22
TABLE 2. UPDATES TO DELIVERABLE 3.4 COMPARED TO DELIVERABLE 3.3 23
TABLE 3. SURVEY 34
TABLE 4. INTEREST PER SUITE 34
TABLE 5. TYPE OF MATERIAL SUITE1 37
TABLE 6. TYPE OF MATERIAL SUITE2 40
TABLE 7. TYPE OF MATERIAL SUITE3 42
TABLE 8. SUMMARY BY TYPE OF MATERIAL 45
TABLE 9. FUNCTIONAL VIEWPOINT ARCHITECTURE DIAGRAM 64
TABLE 10. FUNCTIONS 65
TABLE 11. FV ARCHITECTURE DICMDB - ACTIVITY 1 66
TABLE 12. FV ARCHITECTURE DICMDB - ACTIVITY 2 67
TABLE 13. FV ARCHITECTURE DICMDB - ACTIVITY 3 69
TABLE 14. FV FUNCTIONS DICMDB 70
TABLE 15. FV ARCHITECTURE DIKAF - ACTIVITY 1 71
TABLE 16. FV ARCHITECTURE DIKAF - ACTIVITY 2 72
TABLE 17. FV ARCHITECTURE DIKAF - ACTIVITY 3 73
TABLE 18. FV ARCHITECTURE DIKAF - ACTIVITY 4 74
TABLE 19. FV FUNCTIONS DIKAF 75
TABLE 20. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 1 78
TABLE 21. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 2 79
TABLE 22. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 3 81
TABLE 23. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 4 82
TABLE 24. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 5 84
TABLE 25. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 6 85
TABLE 26. FV ARCHITECTURE DIMEC-LCA - ACTIVITY 7 86
TABLE 27. FV FUNCTIONS DIMEC-LCA 87

TABLE 28. FV ARCHITECTURE DIMDF - ACTIVITY 1 88
TABLE 29. FV ARCHITECTURE DIMDF - ACTIVITY 2 90
TABLE 30. FV ARCHITECTURE DIMDF - ACTIVITY 3 91
TABLE 31. FV FUNCTIONS DIMDF 92
TABLE 32. FV ARCHITECTURE DIMM - ACTIVITY 1 94
TABLE 33. FV ARCHITECTURE DIMM - ACTIVITY 2 96
TABLE 34. FV ARCHITECTURE DIMM - ACTIVITY 3 97
TABLE 35. FV ARCHITECTURE DIMM - ACTIVITY 4 98
TABLE 36. FV FUNCTIONS DIMM 99
TABLE 37. MOCKUP HOME PAGE & MATERIAL DEFINITION OF DI^{MD} 100
TABLE 38. FV ARCHITECTURE DIMD - ACTIVITY 1 101
TABLE 39. MOCKUP RESULTS REVIEW 102
TABLE 40. FV ARCHITECTURE DIMD - ACTIVITY 2 103
TABLE 41. FV FUNCTIONS DIMD 103
TABLE 42. FV ARCHITECTURE DIMMS - ACTIVITY 1 105
TABLE 43. FV ARCHITECTURE DIMMS - ACTIVITY 2 106
TABLE 44. FV FUNCTIONS DIMMS 107
TABLE 45. FV ARCHITECTURE DIMPS - ACTIVITY 1 109
TABLE 46. FV ARCHITECTURE DIMPS - ACTIVITY 2 110
TABLE 47. FV ARCHITECTURE DIMPS - ACTIVITY 3 111
TABLE 48. FV FUNCTIONS DIMPS 112
TABLE 49. FV ARCHITECTURE DIDTPC - ACTIVITY 1 113
TABLE 50. FV ARCHITECTURE DIDTPC - ACTIVITY 2 114
TABLE 51. FV ARCHITECTURE DIDTPC - ACTIVITY 3 115
TABLE 52. FV ARCHITECTURE DIDTPC - ACTIVITY 4 116
TABLE 53. FV ARCHITECTURE DIDTPC - ACTIVITY 5 117
TABLE 54. FV FUNCTIONS DIDTPC 118
TABLE 55. GENERAL TABLE COMPONENTS EXAMPLE 120
TABLE 56. FUNCTIONAL COMPONENT DIAGRAM 123

TABLE 57. GUI FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 125
TABLE 58. KITEM SERVICE FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 127
TABLE 59. USER VALIDATION FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 129
TABLE 60. DATA UPLOAD FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 131
TABLE 61. DATA DOWNLOAD FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 133
TABLE 62. DATA SEARCH FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 135
TABLE 63. DATA STORAGE FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 137
TABLE 64. KNOWLEDGE GRAPH AND VISUALIZATION FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 139
TABLE 65. LOG IN AND CREATE ACCOUNT FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 141
TABLE 66. GUI OF THE FUNCTIONAL COMPONENT OF THE DIMEC-LCA 143
TABLE 67. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DIMEC-LCA 143
TABLE 68. IDENTITY AND ACCESS MANAGEMENT OF THE FUNCTIONAL COMPONENT OF THE DIMEC-LCA 144
TABLE 69. DATA SOURCE PLUGIN OF THE FUNCTIONAL COMPONENT OF THE DIMEC-LCA......... 144
TABLE 70. MAPPING OF THE FUNCTIONAL COMPONENT OF THE DIMEC-LCA......... 147
TABLE 71. STORE DATA FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 149
TABLE 72. VISUALIZE RESULTS & DOWNLOAD DATA FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 151
TABLE 73. LOG IN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS... 153
TABLE 74. USER VALIDATION AND AUTHENTICATION FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 157
TABLE 75. EXPLORATION APP FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS......... 159

TABLE 76. CORRELATION APP FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....161
TABLE 77. MATERIALS RELATION APP FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....163
TABLE 78. GUI FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....165
TABLE 79. AI ALGORITHMS FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....167
TABLE 80. DATA ANALYZER FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....169
TABLE 81. DATA STORAGE FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....171
TABLE 82. DATA INSERTION FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....173
TABLE 83. FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....175
TABLE 84. LOG IN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....177
TABLE 85. GUI OF THE FUNCTIONAL COMPONENT OF THE DIMMS....178
TABLE 86. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DIMMS....178
TABLE 87. PREDICT MECHANICAL PROPERTIES API OF THE FUNCTIONAL COMPONENT OF THE DIMMS TOOLKIT....179
TABLE 88. STORE DATA FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 181
TABLE 89. VISUALIZATION AND DASHBOARDING FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....183
TABLE 90. LOG IN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS....185
TABLE 91. GUI OF THE FUNCTIONAL COMPONENT OF THE DIMPS....186
TABLE 92. VISUALIZATION AND DASHBOARDING OF THE FUNCTIONAL COMPONENT OF THE DIMPS....186
TABLE 93. LOG IN VALIDATOR OF THE FUNCTIONAL COMPONENT OF THE DIMPS....187
TABLE 94. PREDICT MANUFACTURING PROCESS PARAMETERS OF THE FUNCTIONAL COMPONENT OF THE DIMPS....187
TABLE 95. OPTIMIZATION RESULTS OF THE FUNCTIONAL COMPONENT OF THE DIMPS....187

Co-funded by the European Union

TABLE 96. DATA STORAGE OF THE FUNCTIONAL COMPONENT OF THE DIMPS 188
TABLE 97. MAPPING OF FUNCTIONAL COMPONENTS TO IIRA DIMPS 188
TABLE 98. STORE DATA FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 191
TABLE 99. VISUALIZA RESULTS & MODIFY VISUALIZATIONS & DOWNLOAD DATA FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 193
TABLE 100. LOG IN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 195
TABLE 101. DIGITAL TWIN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 199
TABLE 102. LOG IN FUNCTIONAL COMPONENT DIAGRAM, STRUCTURE AND REQUIREMENTS 201
TABLE 103. DATA FLOW EXAMPLE 205
TABLE 104. NECESSARY EQUIPMENT EXAMPLE 205
TABLE 105. API EXAMPLE 206
TABLE 106. DATA FLOWS DICMDB 208
TABLE 107. API DICMDB 211
TABLE 108. DATA FLOWS DIKAF 212
TABLE 109. DATA ANALYTICS DIKAF 213
TABLE 110. API DIKAF 216
TABLE 111. DATA FLOWS DIMECLCA 217
TABLE 112. API DIMECLCA 219
TABLE 113. NECESSARY EQUIPMENT DIMDF 222
TABLE 114. DATA FLOWS DIMM 224
TABLE 115. API DIMM 228
TABLE 116. DATA FLOWS DIMD 230
TABLE 117. API DIMD 234
TABLE 118. DATA FLOWS DIMMS 236
TABLE 119. USER MANUALS OF DIMMS 238
TABLE 120. API DIMMS 240
TABLE 121. DATA FLOW DIMPS 242
TABLE 122. USER MANUALS DIMPS 244

TABLE 123. API DIMPS 246
TABLE 124. DATA FLOWS DIDTPC 248
TABLE 125. DATA ANALYTICS DIDTPC 248
TABLE 126. USER MANUAL DIDTPC 251
TABLE 127. API DIDTPC 253
TABLE 128 CHANGES IN THE DELIVERABLE BASED ON THE REVIEWERS' COMMENTS 256

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
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.

To understand the evolution of the viewpoints within the DiMAT project, it is necessary to compare the updates between the previous version (D3.3) and the current version (D3.4). The viewpoints provide the backbone of the architecture, covering critical aspects such as business impact, user interaction, system functionality, and technical implementation. By reviewing the changes between these two versions, we can preciate how the framework has matured and adapted to project needs:

Viewpoint	Deliverable 3.3	Deliverable 3.4
Business Viewpoint	In D3.3, this perspective focused mainly on the basic identification of stakeholders and their business interests. Surveys were used to gather general information from stakeholders, but less emphasis was placed on implementation challenges or cost considerations.	In D3.4, the deliverable is extended by introducing a more detailed approach in two phases: one focusing on stakeholder identification and the other on business impact analysis, including implementation challenges and costs. It uses a combination of targeted questionnaires and interviews to gain deeper insight from partners, ensuring that business objectives are closely aligned with the technical objectives of the project. This version also places greater emphasis on aligning business requirements with technical capabilities, making the methodology more robust and comprehensive.
Usage Viewpoint	In D3.3, the deliverable provided a basic identification of tasks, roles and activities, with less emphasis on detailed workflows and fewer defined roles. It focused more on outlining general interactions	In this deliverable D3.4, toolkits will be updated as needed. If no modifications are required from the previous version (D3.3), this section will be excluded from the viewpoint, and the content will remain consistent with the initial deliverable submitted D3.3. Concretely,

	without going into detail about specific user activities.	D3.4 introduces new sections to enhance the toolkit's structural framework and operational scope, supplemented by schematic diagrams. As implied by their titles, these sections consist of a Functional Map and a Summary Matrix. The Functional Map aims to illustrate the toolkit's overarching architecture by aligning a functional perspective with detailed activity diagrams and accompanying explanations. The Summary Matrix, in turn, delineates the relationships between toolkit functions and activities in a tabular format, establishing a clear linkage between them.

Table 1. Comparison of D3.3 and D3.4

In this deliverable D3.4, toolkits will be updated as needed. If no modifications are required from the previous version (D3.3), this section will be excluded from the viewpoint, and the content will remain consistent with the initial deliverable submitted D3.3. Concretely, D3.4 introduces new sections to enhance the toolkit's structural framework and operational scope, supplemented by schematic diagrams. As implied by their titles, these sections consist of a Functional Map and a Summary Matrix. The Functional Map aims to illustrate the toolkit's overarching architecture by aligning a functional perspective with detailed activity diagrams and accompanying explanations. The Summary Matrix, in turn, delineates the relationships between toolkit functions and activities in a tabular format, establishing a clear linkage between them.

Functional Viewpoint	In D3.3 were identified the functional components of the system, including their interfaces and published operations, to facilitate a comprehensive understanding of system functionality. It detailed interactions between external elements and the system, as well as inter-component interactions, to clarify how these components collaborate within the overall architecture. Additionally, it included	In this deliverable D3.4, toolkits will be updated as needed. If no modifications are required from the previous version (D3.3), this section will be excluded from the viewpoint, and the content will remain consistent with the initial deliverable submitted D3.3. D3.4 decomposes the DiMAT framework into distinct functional components, clearly identifying data flows, decision points and interactions between different functional domains such as control,

Co-funded by the European Union

	a mapping of these identified functional components to the five primary functional domains defined by IIRA, ensuring alignment with established architectural standards and enhancing system interoperability and scalability. It was an initial exploration of the system architecture, outlining the main functions without delving into their interconnections.	operations and information. This version also expands the functional maps and relationships between components, providing a more granular understanding of how each part of the system works together. A new section has been added into D3.4, titled "Functional Components, Structural Architecture, and Requirements Interactions," was incorporated into this document to finalize the toolkit's architectural framework. Several technical diagrams were included: one illustrating the functional components, another detailing the functional structure, and a schema mapping the requirements linked to each functional component. As a result, version 3.4 provides a more comprehensive analysis of the system architecture, ensuring that all functional elements are well integrated and aligned with project objectives.
Implementation Viewpoint	D3.3 focused on describing key technologies and components, but did not provide in-depth details on implementation strategies or data processing. It was an introductory analysis that provided a high-level overview of the technologies to be used, without going into specifics such as the use of APIs or detailed deployment patterns.	D3.4 presents a much more detailed and refined methodology. It focuses on defining the specific technologies required for the implementation of DiMAT solutions, including comprehensive deployment patterns and data handling strategies. This version also emphasises the use of APIs for interfacing between components, providing detailed technical guidelines to ensure scalability and maintainability. The deliverable provides a complete blueprint for effectively implementing and integrating system components, making it much more practical and actionable than its predecessor.

Table 2. Updates to Deliverable 3.4 compared to Deliverable 3.3

This comparison (Table 1) reveals that the new version 3.4 offers a more comprehensive approach to the viewpoints, integrating more detailed information about the technical components. Each viewpoint has evolved to better address the challenges of implementation and align with the project's business objectives and stakeholder needs. This iterative improvement in the viewpoints ensures a stronger alignment between the project's goals and the technological solutions being developed, creating a robust framework that facilitates the design, development, and deployment of DiMAT 's digital modeling and simulation tools.

2 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 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.

2.1 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.

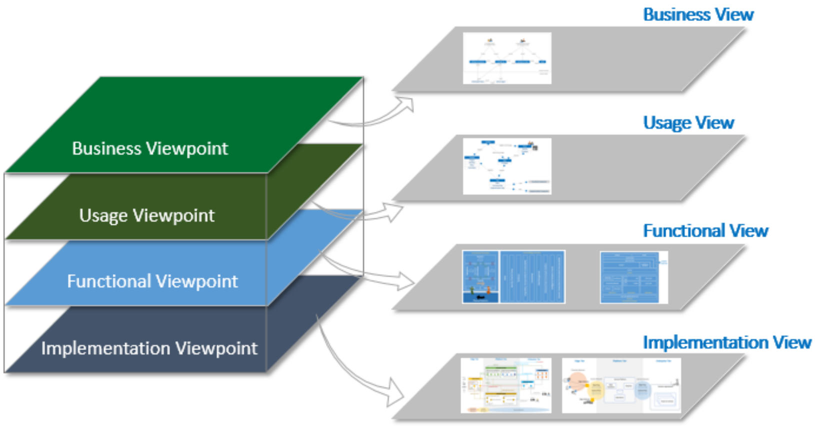


Figure 1. IIRA Architecture Framework

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 solutions' different components and define the technologies to be used for their development. The Application Programming Interfaces (API) will be specified to define the interconnection and communication of the solutions. 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 implementing the DiMAT toolsets described in WP4, WP5 and WP6.

2.2 DIMAT RA: VIEWPOINTS

Following the IIRA methodology, the final iteration of the DiMAT RA integrates essential perspectives (referred to as "viewpoints" in DiMAT) for intelligent manufacturing processes. The DiMAT Reference Architecture uses 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 [2].

- The **business viewpoint** details 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 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 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.

3 BUSINESS VIEWPOINT

As mentioned previously, 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. The main objective of the BUSINESS VIEWPOINT is to provide, within the DiMAT design activities, the business's vision, values, and objectives to avoid the risk of a 'technology-centric' development. Thus, the requirements closer to the real world are identified and included in the design of the DiMAT Architecture.

The DiMAT Architecture is very complex, and its design requires a natural evolution. However, the business viewpoint must be considered in all these processes to relate to the real requirements. This circumstance was considered in the tasks of the DiMAT project, and a first analysis of the business viewpoint was included in month 9 of the project and a second analysis in month 18.

3.1 BUSINESS VIEWPOINT 1ST APPROACH FOR DIMAT

3.1.1 Business Viewpoint Concepts, Methodology and Questionnaire

The DiMAT proposal includes three suites and three toolkits in each suite. Therefore, it will be necessary to know the business perspective for each of these toolkits. The task 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 **was 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 were asked (Figure 2):

1. Business decision-makers
2. Technical leaders.

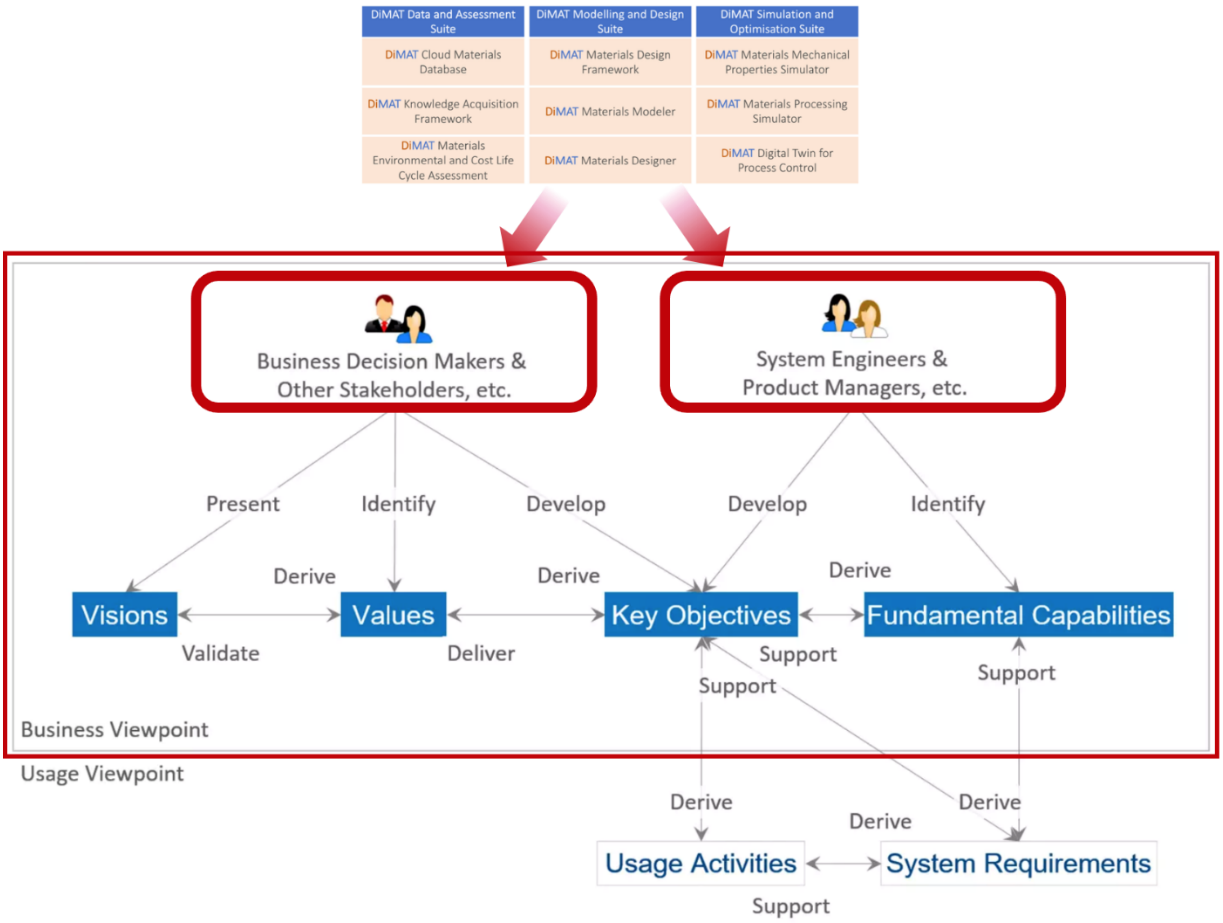


Figure 2. DiMAT Business Viewpoint

The **Business Viewpoint Methodology** for completing the task was 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 sought to obtain information on the toolkits in which the stakeholder is interested.

A **questionnaire** was 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 comprised four sections in addition to the survey instructions. These sections were as follows:

1. Identification of Respondents.
2. Toolkit Selection.
3. Business Perspective.
4. Technical Perspective.

The full survey model sent to the stakeholders can be consulted in the annexe section of the first deliverable.

3.1.2 DiMAT Business Viewpoint 1st Approach

The nine toolkits defined by the DiMAT project are grouped into three suites (Figure 3).

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

Five **organizations** (stakeholders) were identified for their participation in developing the business viewpoint of these nine toolkits. 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.).

The complete analysis can be consulted in the deliverable 3, and a **summary** of the analysis is presented below. The business perspective has made it possible to identify the suites and toolkits of greatest interest to stakeholders. The DiMAT Materials Environmental and Cost Life Cycle Assessment toolkit (DiMEC·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 (DiMPS) is of great interest to most stakeholders. On the other hand, the DiMAT Material Designer (DiMD) toolkit is the least interesting, from a global perspective. From the perspective of the suites, 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.

Co-funded by
the European Union

3.2 BUSINESS VIEWPOINT 2ND APPROACH

This new approach focuses on the following main aspects:

1. Extension in the number of stakeholders to obtain a wide range of opinions.
2. Focus on suites.
3. Suitability and Expected Business Impact.
4. Identification of difficulties in their implementation.

The methodology proposed to carry out this 2nd approximation is presented in section 2.2.1 Business Viewpoint Methodology, and the design of the questionnaire used for the survey is presented in the appendix.

3.2.1 Business Viewpoint Methodology

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

Firstly, the identification of stakeholders. This approach extends the evaluation of the suites beyond the pilot companies involved in the project. The aim is to know the usefulness of the suites for companies in the same and/or in similar industrial sectors. To do this, we seek to expand the number of relevant stakeholders to be involved.

Secondly, stakeholders are asked for their business perspective on each suite. The business decision-makers domain addresses the suitability of the suite and expected impact for the business, and the technical leaders' domain addresses the difficulties in their implementation.

It is important to note that stakeholders may be interested in some suites and not in others. In this sense, we only seek to obtain information on the suites in which the stakeholder is interested. In a schematic way, the steps are as follows:

1. Stakeholders' identification.
2. For each stakeholder and each suite (they are interested in):
   2.1. Extract information on their suitability and expected impact.
   2.2. Extract information on their difficulties.

3.2.2 Business Viewpoint Questionnaire

A questionnaire has been designed to collect this new information regarding the stakeholders’ business perspective. The questionnaire is composed of four sections, in addition to the survey instructions. These sections are as follows: 1. Identification of Respondents, 2. Suite Selection, 3. Suitability for the business, 4. Difficulties, 5. Costs perspective. Table 1 displays these sections and the information required in each of them.

Section	**Question/Text**	**Answer type**
Survey instructions	Welcome, survey duration, survey objective...	Text
Identification	Company Sector	Close answer
	Company Size	
	Company Location	Close answer
Suite selection	Introduction to DiMAT Suites	Text
	Interest per suite	Yes /No
Suitability and expected business impact	How important for your business are the processes that are impacted by the suite?	5-point Likert Scale
	How much can the process be improved by implementing Suite X?	5-point Likert Scale
	Can you list a maximum of 3 processes that will be highly impacted by the Suite in order of importance? (e.g. raw material selection, material design, material production, quality control)	
	Which companies' needs does Suite X address?	Close answer
Difficulties	Mark the **organizational** difficulties that your company could have when implementing Suite X	Close answer
	Mark the **technical** difficulties that your company could have when implementing Suite 1.	Close answer

Costs	What monthly cost per company (unlimited users) would you be willing to pay for Suite 1?	Close answer

Table 3. Survey

3.3 DIMAT RESULTS BUSINESS VIEWPOINT 2ND APPROACH

The participation of the companies in surveys is, in general, very difficult. Two actions were prepared to obtain answers.

- **Action 1.** A webinar was launched for the diffusion of the project and its suites. The webinar was published on YouTube (https://www.youtube.com/watch?v=jrDYfZPXChI).
- **Action 2.** E-mail diffusion to companies in the material sector from databases of F6S and SUPSI.

Finally, 34 answers from different companies have been collected. They were asked about the three suites, and their interest in each suite was variable. Some were interested in all suites, and others were interested in only one suite. The distribution is shown in the following table:

Interest per Suite		
Suite1	Suite2	Suite3
29 answers	21 answers	23 answers
85%	61%	67%

Table 4. Interest per Suite

More than 60% are interested in the suits, highlighting Suite 1 with 85% of interest.

The analysis of these answers has been structured in the following main aspects according to the structure of the survey:

1. Suitability and expected impact.
2. Difficulties for their implementation.
3. Cost perspective.

3.3.1 Suitability and Expected Business Impact

3.3.1.1 Asses of interest and impact

Throughout the surveys, we have obtained information about the following questions:

- "How much can the process be improved by implementing Suite X?" This question evaluates the direct impact that the Suite can have on efficiency, productivity, or any other aspect of the process in question.
- How important for your business are the processes that are impacted by the suite?" Evaluate how critical the process affected by the Suite is to the operational and strategic success of the company.

Thus, the questionnaire responses can be interpreted as follows:

- If the importance of the process is high and the expected improvement is equally high, this strongly supports investment in the Suite since the expected improvement is aligned with the importance of the process to the business. (Investment)
- If the importance of the process is low and the expected improvement is also low, the need to invest in the Suite could be questioned since the expected improvement does not justify the investment in a low-importance process. (Questionable)
- If the importance of the process is high but the expected improvement is low, it may be necessary to review the Suite implementation strategy or consider other alternatives that may provide a more substantial improvement in a business-critical process. (Reconsider)
- If the importance of the process is low but the expected improvement is high, a careful evaluation is required to determine if the investment in the Suite is justified, considering the potential impact on other processes, the overall cost-benefit, and the possibility of reallocation. of resources. (Evaluation)

Graphically, it can be presented as follows (Figure 4):

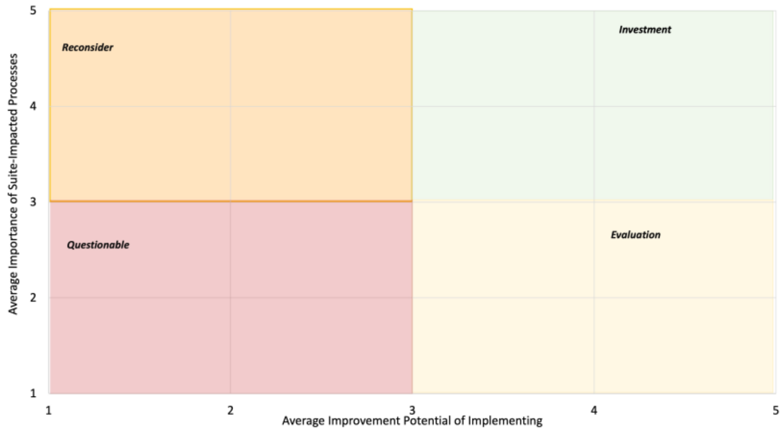


Figure 4. Importance-Potential quadrants

3.3.1.2 SUITE1 analysis

We have received 29 answers about Suite 1. The distribution of the 29 responses is represented as follows (Figure 5). The number indicates how many times the same value is repeated.

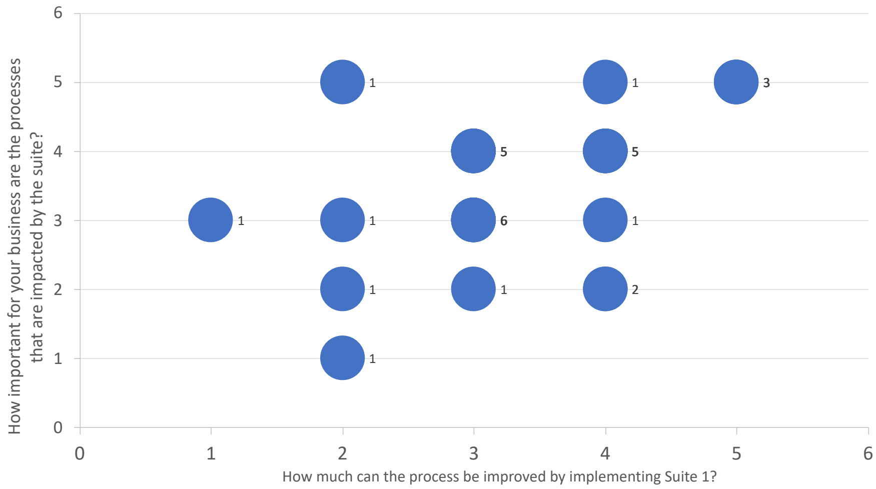


Figure 5. Expressions of interest in Suite 1

The results are very dispersed. Despite this, we observe that the most repeated situations are those in which the importance and improvement values are (3,3), (3,4), and (4,5).

3.3.1.2.1 According to the type of material:

Type of material	*Interested in Suite1*
Composite materials	6
Glass	2
Metals and alloys	1
Other	9
Polymers (plastics)	8
TOTAL	***29***

Table 5. Type of material Suite1

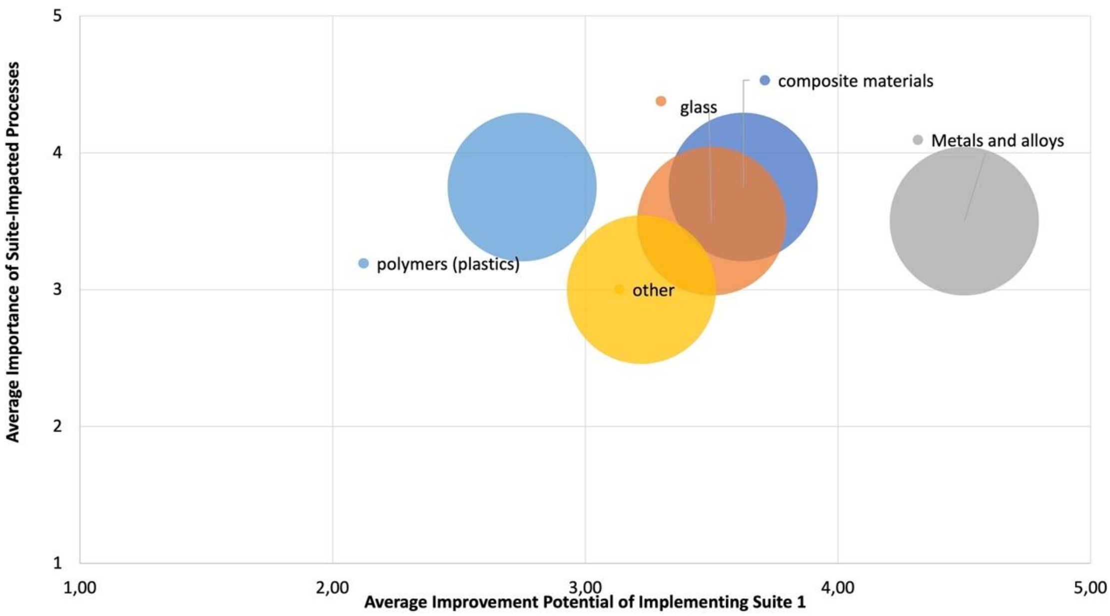


Figure 6. Expressions of interest in Suite 1 by type of material

According to the proposed analysis, companies of metals and alloys, glass, and composite materials can **invest** in Suite1 because the expected improvement is high, and the process impacted by the Suite is also important for the organisation.

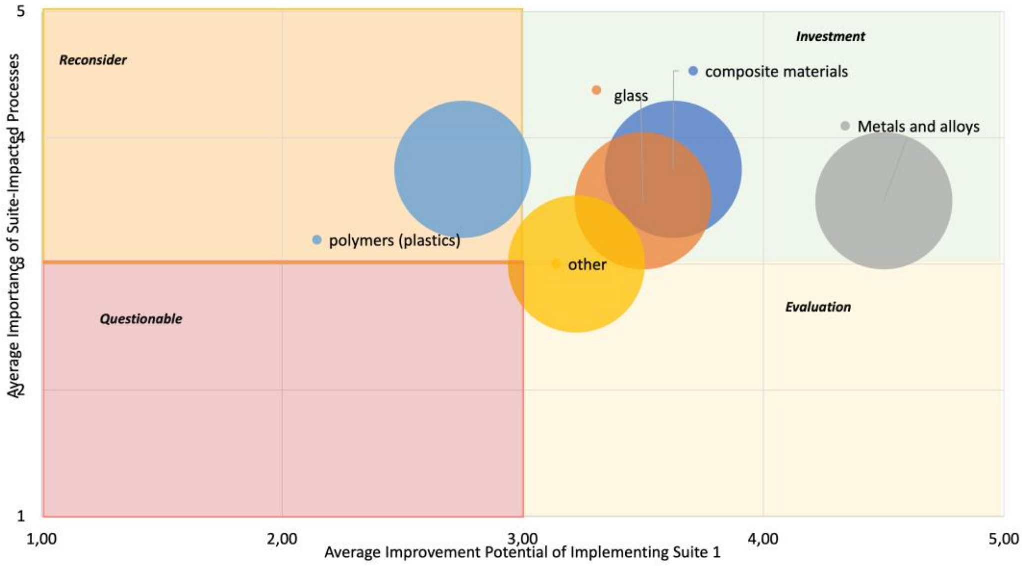


Figure 7. Average Improvement Potential

Instead, the polymer company should **reconsider** investing in Suite1 because it does not anticipate favourable outcomes, given the process's critical importance to the company.
For the other type of material, the average improvement potential is slightly greater than the average importance of the impacted process, so a more detailed **evaluation** could be done to analyse the suitability of the investment better.

3.3.1.2.2 Grouped by Suite

The following chart shows the aggregate analysis for suite 1. In general, for suite 1, the expected average improvement is slightly lower than the average importance of the process. However, the medium-high value obtained can justify the investment in most cases.

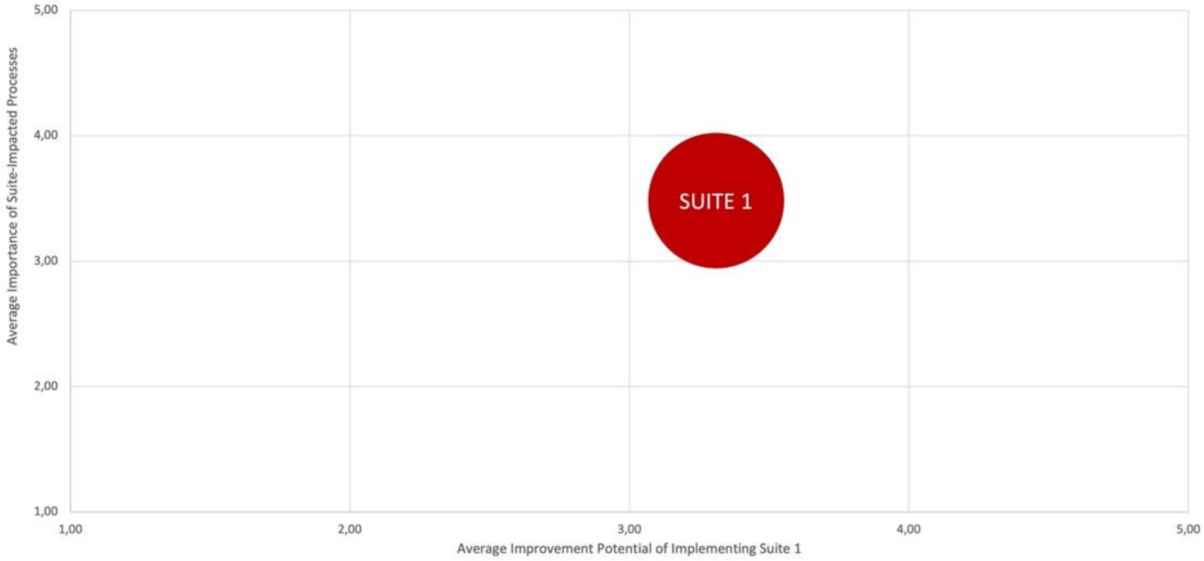


Figure 8. Interest in Suite 1

3.3.1.3 SUITE2 analysis

We have achieved 21 answers interested in Suite2. The distribution of the 21 responses is represented as follows (Figure 9). The number indicates how many times the same value is repeated.

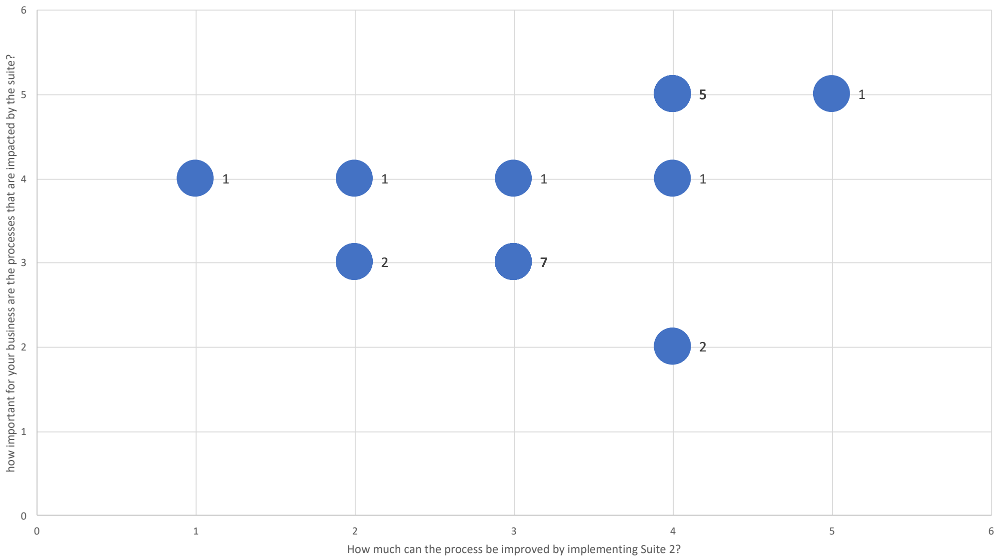


Figure 9. Expressions of interest in Suite 2

We observe that the situations that are most repeated are those in which the importance and improvement values with Suite2 are (3,3) and (5,5).

3.3.1.3.1 According to the type of material:

Type of material	*Interested in Suite2*
Composite materials	8
Glass	2
Metals and alloys	1
Other	7
Polymers (plastics)	5
TOTAL	***21***

Table 6. Type of material Suite2

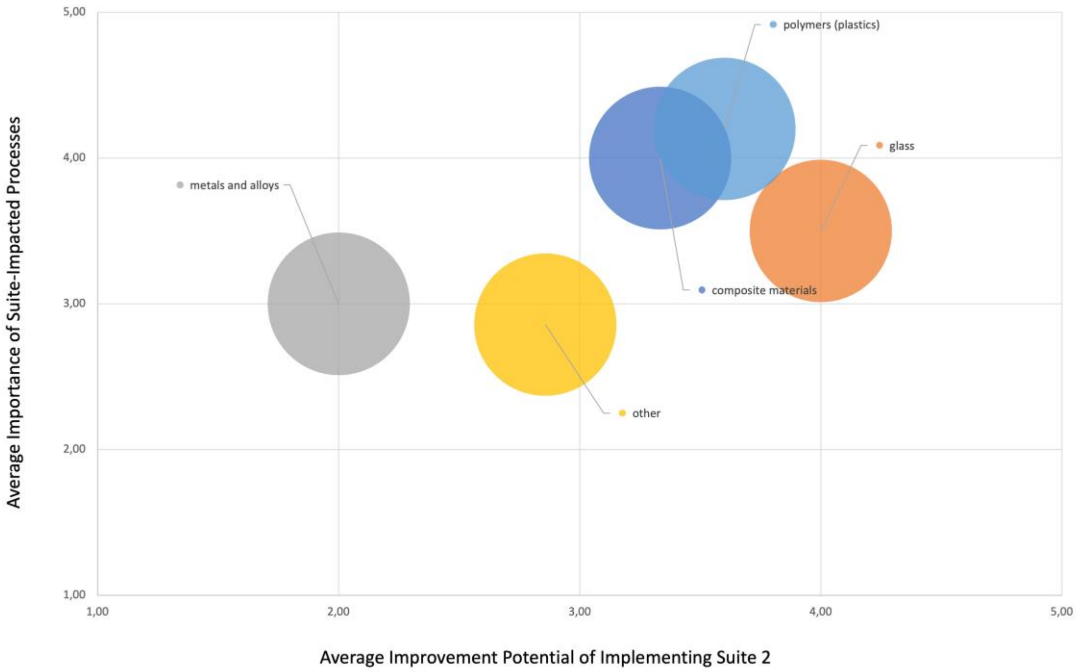


Figure 10. Expressions of interest in Suite 2 by type of material

According to the proposed analysis, polymers, glass and composite materials, companies can invest in Suite2 because the expected improvement is big, and the processes impacted by the Suite are also important for the organization. In the case of metals and alloys, the values should **reconsider** and **question** investing in Suite2 because it does not anticipate favourable outcomes, given the process's importance to the company. The *other* type of material company is near to **questioning** investing in Suite2 to determine if the investment in the Suite is justified.

3.3.1.3.2 Grouped by Suite

The following chart shows the aggregate analysis for Suite 2. In general, for Suite 2, the expected average improvement is slightly lower than the average importance of the process. However, the value obtained can justify the investment in most cases.

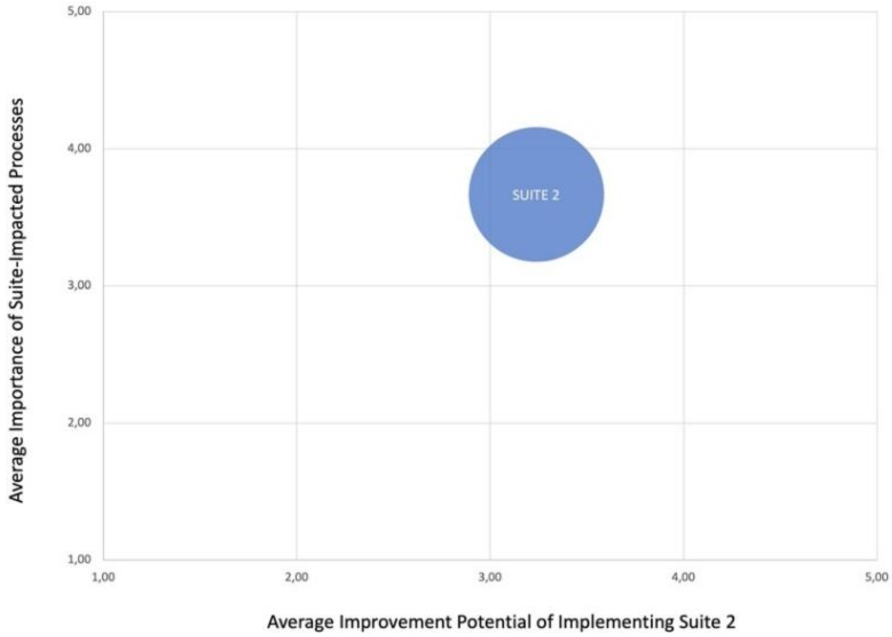


Figure 11. Interest in Suite 2

3.3.1.4 SUITE3 analysis

We have achieved 23 answers interested in Suite3. The distribution of the 23 responses is represented as follows (Chart 12). The number indicates how many times the same value is repeated.

Co-funded by the European Union

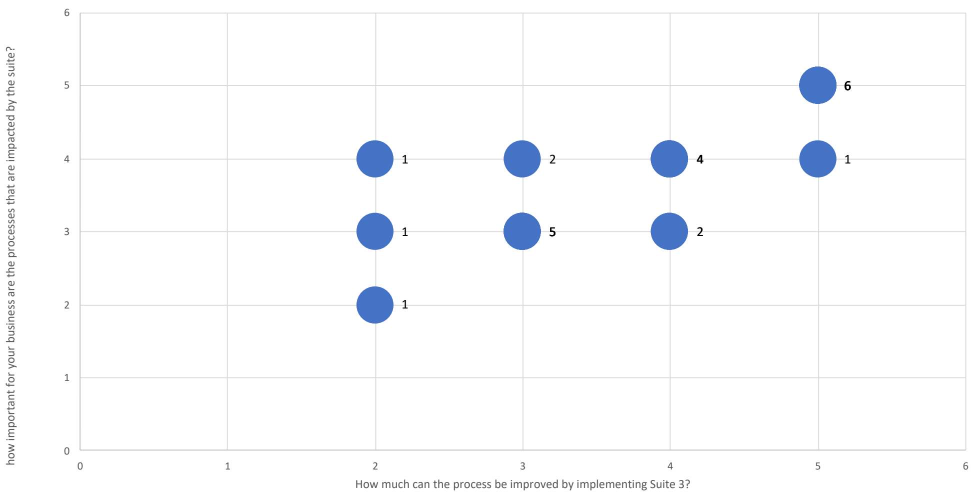


Figure 12. Expressions of interest in Suite 3

We observe that the situations that are most repeated are those in which the importance and improvement values with Suite3 are (5,5), (4,4) and (3,3); which shows a high alignment. The data is also more concentrated.

3.3.1.4.1 According to the type of material:

Type of material	***Interested in Suite3***
Composite materials	6
Glass	2
Metals and alloys	2
Other	8
Polymers (plastics)	5
TOTAL	***23***

Table 7. Type of material Suite3

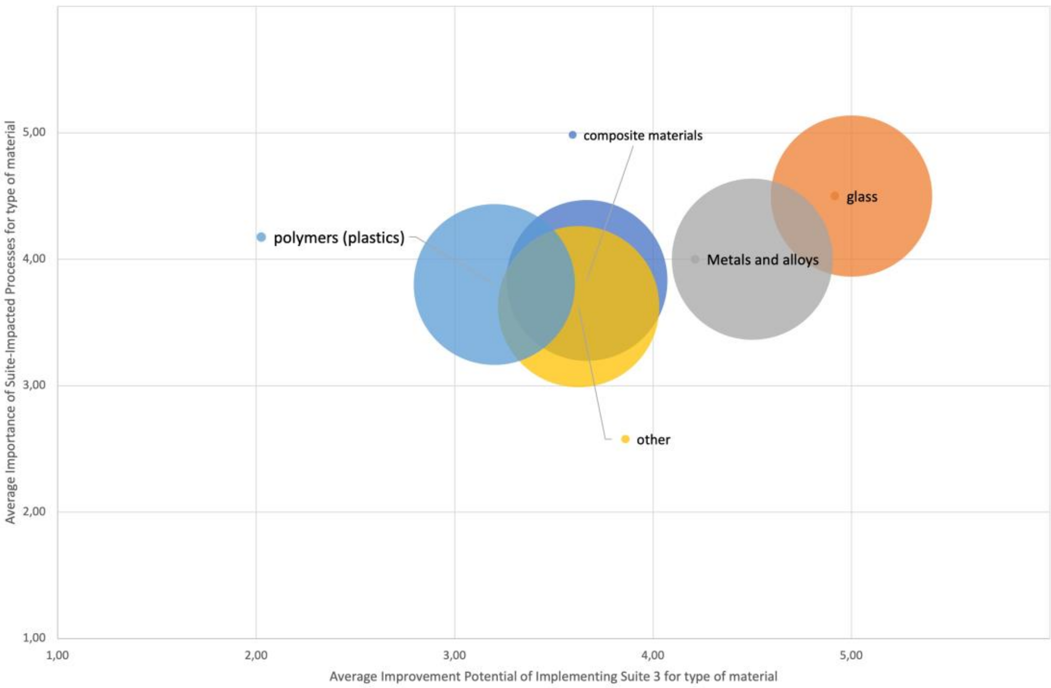


Figure 13. Expressions of interest in Suite 3 by type of material

Compared to the previous results, we can see that Suite3 is more appropriate for glass materials, with a high level of expected improvement for an important process, so it could be strongly recommended that **investment** be made. It is like *Metals and alloys* (although with a lower expected impact). For composite materials, polymers, and other materials, it can be recommended to invest in Suite3. The expected improvement, in these cases, is slightly lower, although aligned with the importance of the process.

3.3.1.4.2 Grouped by Suite

The following chart shows the aggregate analysis for suite 3. The value of the expected improvement and the importance of the process are practically the same value in this case, which may indicate an optimal alignment between both metrics, reflecting the best possible situation. In this case, the investment in the Suite 3 would be perfectly aligned with the needs and priorities of the company, which suggests that the decision to implement the Suite would be highly recommended.

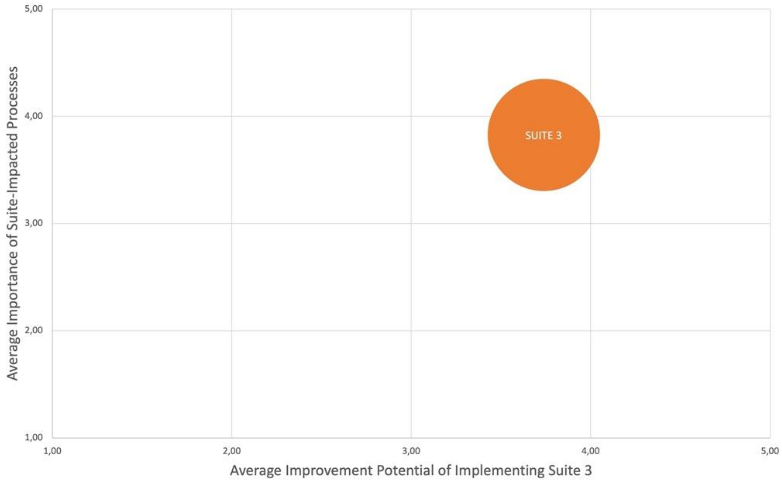


Figure 14. Interest in Suite 3

3.3.1.4.3 Comparative analysis of outcomes across suites

Suite3 achieves the highest value in process improvement and demonstrates stronger alignment with business importance. Also, Suite1 presents an adequate balance between process improvement and business importance. Suite2 shows a high level of process importance to the business (compared to Suite3 and Suite1). This indicates that the processes impacted by Suite2 are significant to the business. However, the expected improvement is slightly smaller in relation to the other two Suites respectively. This suggests that, although the business process is crucial, the expected improvement may not be as substantial as in the other Suites.

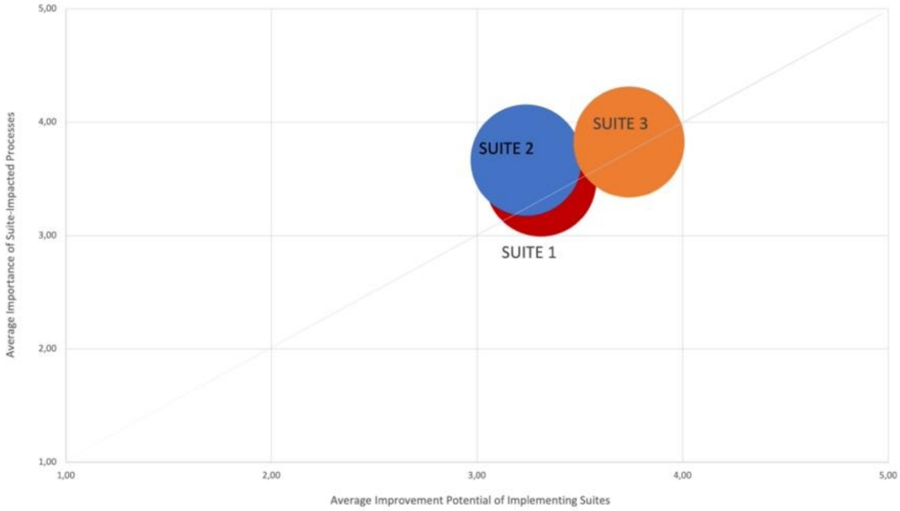


Figure 15. Interest in Suites

Co-funded by the European Union

3.3.1.4.4 Summary by type of material

Type of material	Suite1	Suite2	Suite3
Composite materials	Investment	Investment	Investment
Glass	Investment	Investment	Investment
Metals and alloys	Investment	Reconsider Questionable	Investment
Other	Evaluate	Questionable	Investment
Polymers (plastics)	Reconsider	Investment	Investment

Table 8. Summary by type of material

3.3.1.5 Needs Covered by the Suites

Figure 16 represents the needs that could be covered by the different Suites according to the surveyed companies. It shows the percentage of surveyed companies that have selected each predefined need: Brand reputation, Customer satisfaction, financial health, Human capital, Intellectual property, Operational Efficiency, Profitability, Revenue Growth, Risk management, and Scalability.

Co-funded by the European Union

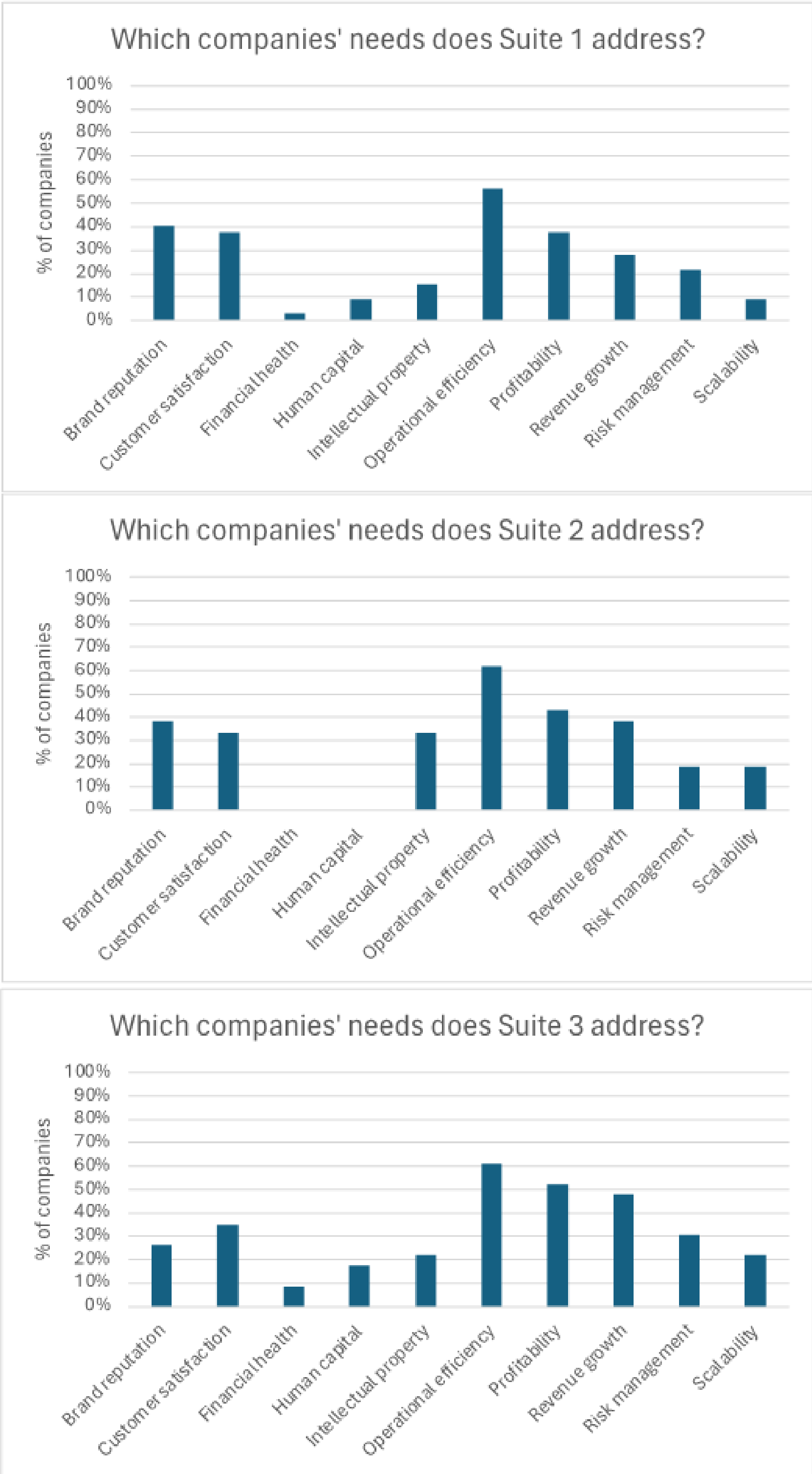


Figure 16. Companies needs that can be covered by Suites.

According to the responses of the companies surveyed, Suite 1 stands out for its ability to improve Operational efficiency. In addition, it is appreciated for its positive impact on Brand reputation and Customer satisfaction, demonstrating its potential to strengthen customer relationships and corporate image. The areas of Profitability and Revenue growth are also highly valued. However, the other needs receive moderate attention, with financial health being the least considered by respondents.

Suite 2 shows a similar response profile to Suite 1, with a high value placed on its impact on Operational efficiency, followed by its influence on Brand reputation, Customer satisfaction, Profitability and Revenue growth. In addition, this suite stands out for its potential coverage of Intellectual property. Importantly, no respondent considers this suite to contribute to financial health or Human capital.

Suite 3 is notable for its impact on Operational efficiency, Profitability and Revenue growth, which are the needs most highlighted by respondents and underlines its potential to contribute to the economic success of companies. There is also a significant impact on Brand reputation, Customer satisfaction, Risk management and Scalability.

When comparing the responses obtained for the three suites, interesting patterns are identified in how companies perceive the impact of these tools on various needs. Although there are small differences among the suites, the patterns are similar. It stands out that all suites are perceived as effective in improving Operational efficiency, this being the most selected need by respondents. In addition, financial health and Human capital are consistently the least selected across all suites.

3.3.2 Difficulties for their implementation

The difficulties that companies may encounter in implementing the different Suites have been identified and quantified by differentiating between organizational and technical difficulties.

3.3.2.1 Organizational difficulties

Figure 17 shows, for each of the suites, the difficulty estimated by the respondents according to Resistance to change, Lack of training, Lack of alignment with business processes, Insufficient budget, Lack of leadership, and Others when implementing the Suites. Each of them shows the minimum value (which in all cases is 0 - no difficulty), the average value and the maximum value given by the respondents.

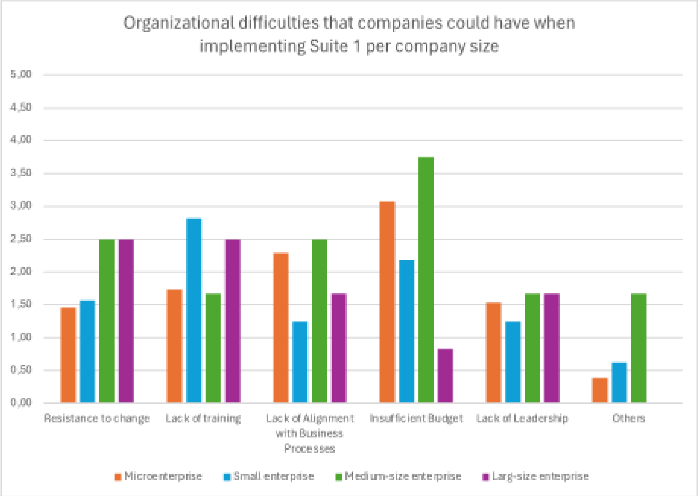
Figure 17. Organizational difficulties by Suites.

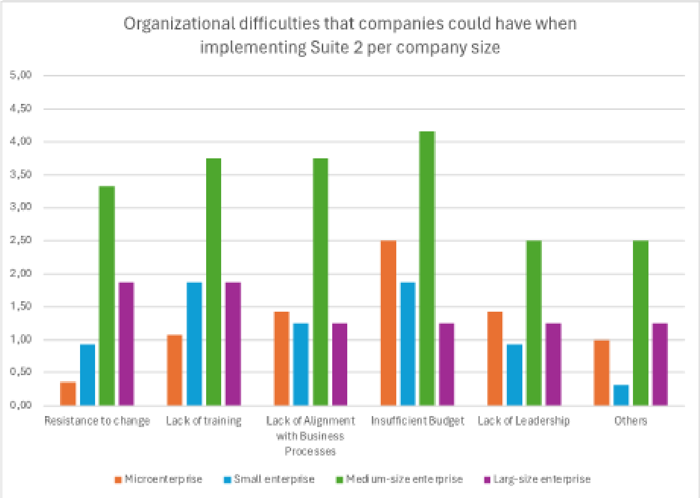
In implementing **Suite 1,** all organizational difficulties have been identified as significant (value 5) by at least one respondent. The mean values for each difficulty indicate that *Insufficient Budget (2.5)* is perceived as the most organizational difficulty, followed by *Lack of Training (2.18).* In addition, concerns about *Lack of Alignment with Business Processes (1.90), Resistance to Change (1.75)* and *Lack of Leadership (1.53)* are minor but relevant, suggesting the importance of effective integration, change management, and leadership during the transition. In the Other category, which has the lowest mean (0.56), of those who selected this option, only one specified that the difficulty is "Time and money for implementation". This indicates that the main categories cover most of the concerns but also highlights the relevance of considering implementation time as a critical factor.

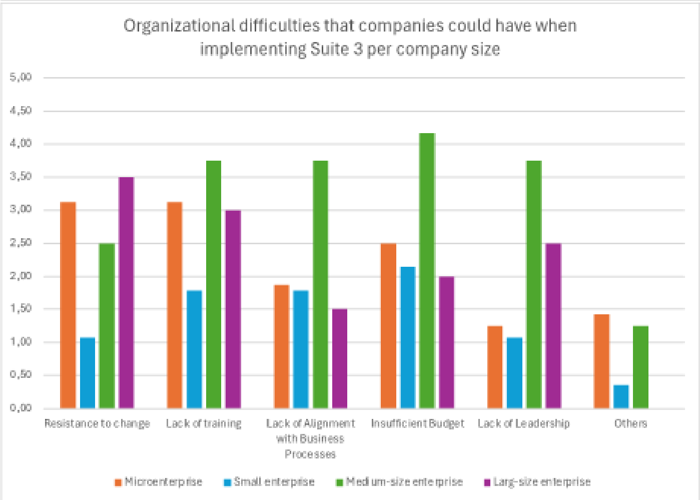
For **Suite 2**, the maximum values reached as high as 5.0 in most organizational difficulties, indicating that certain respondents perceive some of them as very severe. Average scores indicate that *Insufficient Budget (2.27)* is the primary barrier encountered for implementation, emphasizing the critical need for effective financial management. This is closely followed by Lack of Training (1.79), which underscores the necessity for detailed and targeted training sessions to enable proficient use of the suite. Concerns such as *Lack of Alignment with Business Processes (1.55), Lack of Leadership (1.31)* and *Resistance to Change (1.25),* though less pronounced, reflect important areas for management focus to ensure smooth integration and user adoption. Notably, the *Others* category shows a mean of 0.88, with no specific additional difficulties cited, which suggests that the identified categories comprehensively address the primary concerns.

Regarding **Suite 3,** survey responses show that all organizational difficulties have been rated very severe (5) by at least one respondent. *Lack of Training* stands out with the highest average of 2.73, underscoring the essential need for comprehensive training programs to ensure staff can effectively utilize Suite 3. This is closely followed by Resistance to Change and Insufficient Budget, averaging 2.5. *Lack of Alignment with Business Processes (1.93)* and *Lack of Leadership* (1.70) difficulties are less critical but still demand attention. Additionally, the *Others* category, with the lowest average (0.79), included a specific mention of "Material database" by one respondent.

The analysis of organizational difficulties across Suites 1, 2, and 3 reveals that a common theme across all suites is the high concern related to Insufficient Budget, indicating that financial constraints are a universal barrier to successful implementation. Similarly, Lack of Training consistently emerges as a critical issue, stressing the need for adequate educational support to maximize the use of these suites. Differences are noted primarily in the perceived severity of challenges, such as Resistance to Change and Lack of Leadership, with Suite 3 experiencing higher concerns related to change resistance, possibly due to its specific features or the contexts in which it is deployed.


Organizational difficulties that companies could have when implementing Suite 1 per company size
5,00
4,50
4,00
3,50
3,00
2,50
2,00
1,50
1,00
0,50
0,00
Resistance to change
Lack of training
Lack of Alignment with Business Processes
Insufficient Budget
Lack of Leadership
Others
Microenterprise
Small enterprise
Medium-size enterprise
Larg-size enterprise


Organizational difficulties that companies could have when implementing Suite 2 per company size
5,00
4,50
4,00
3,50
3,00
2,50
2,00
1,50
1,00
0,50
0,00
Resistance to change
Lack of training
Lack of Alignment with Business Processes
Insufficient Budget
Lack of Leadership
Others
Microenterprise
Small enterprise
Medium-size enterprise
Larg-size enterprise


Organizational difficulties that companies could have when implementing Suite 3 per company size
5,00
4,50
4,00
3,50
3,00
2,50
2,00
1,50
1,00
0,50
0,00
Resistance to change
Lack of training
Lack of Alignment with Business Processes
Insufficient Budget
Lack of Leadership
Others
Microenterprise
Small enterprise
Medium-size enterprise
Larg-size enterprise

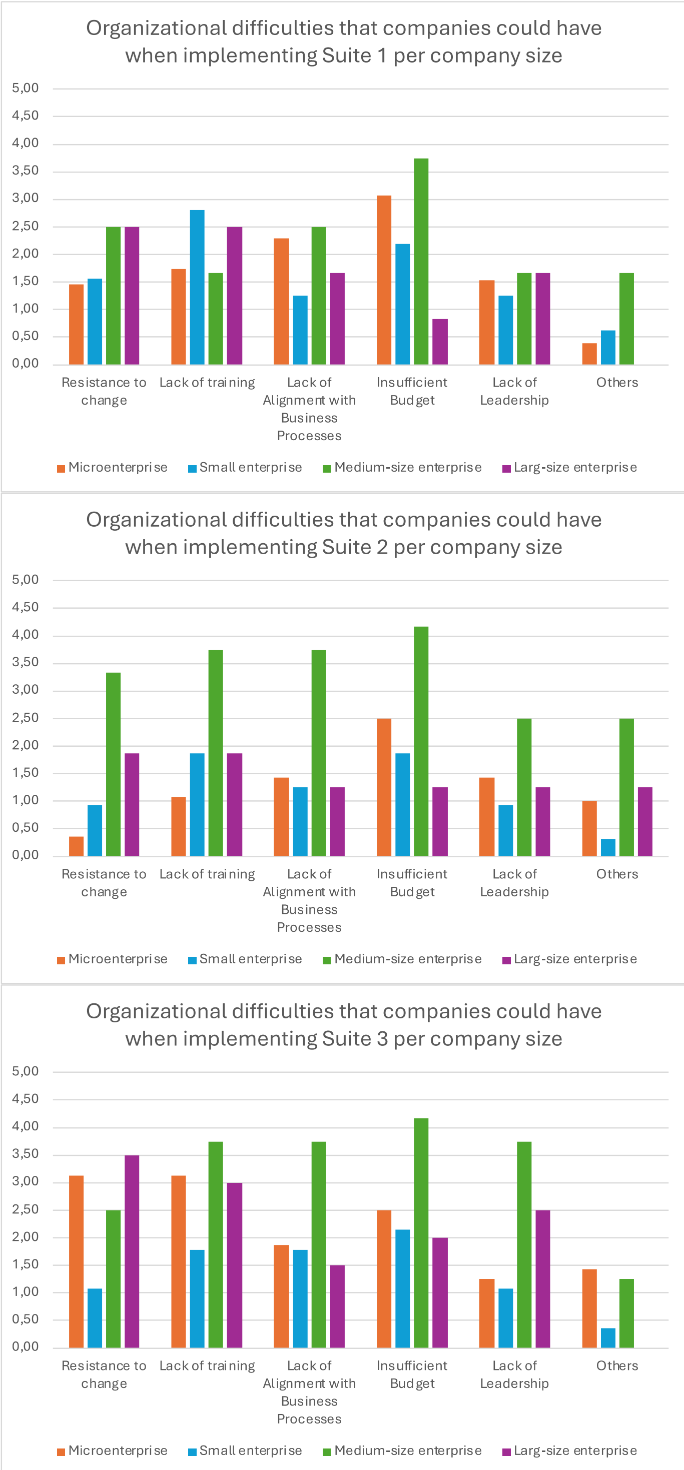


Figure 18. Organizational difficulties by Suites and company size.

In addition, Figure 19 shows the level of difficulty estimated by respondents for companies of different sizes. With them, we can determine if the organizational difficulties perceived by the companies are the same regardless of size or if, on the contrary, they are different, being able to find different types of potential customers.

The analysis of organizational difficulties in implementing the Suites shows variations in perception according to company size. In **Suite 1**, Micro and medium-sized companies indicate a high level of concern for *Insufficient Budget*, with means of 3.08 and 3.75, respectively, while large companies show less concern, with a mean of 0.83. Resistance to Change and *Lack of Alignment with Business Processes* are seen as more problematic in medium and large companies, both with a mean of 2.50. On the other hand, *Lack of Training* is a greater obstacle in small companies, with a mean of 2.81. In contrast, *Lack of Leadership* shows moderate concerns in all size companies, and in the *Others* category, medium-sized companies report specific challenges with a mean of 1.67, while large companies report no additional problems.

In **Suite 2**, midsize companies report the greatest concern in several categories, with Lack of Training and *Lack of Alignment with Business Processes* reaching a mean of 3.75, indicating significant training and process integration challenges. Insufficient Budget is also more problematic for medium-sized companies, with a mean of 4.17, while large companies show less concern, with a mean of 1.25. On the other hand, *Resistance to Change* is particularly noticeable in medium-sized companies, with a mean of 3.33. Micro and small companies report fewer difficulties, although *Lack of Leadership* is more relevant in micro-companies, with a mean of 1.43.

In **Suite 3** implementation, *Resistance to Change* and *Lack of Training* is particularly problematic for micro companies, with mean scores of 3.13. For medium companies, Insufficient *Budget and Lack of Alignment with Business Processes* stand out as the biggest challenges, with scores of 4.17 and 3.75, respectively, reflecting difficulties in aligning the suite with existing business processes and budget constraints. Large companies show moderate concern in several areas, but *Lack of Leadership* scores a mean of 3.75, suggesting that effective leadership is a critical challenge in larger-scale companies. Interestingly, in the *Others* category, only micro and medium-sized firms report additional difficulties with scores of 1.43 and 1.25, respectively, while large firms indicate no additional concerns.

Co-funded by
the European Union

3.3.2.2 Technical difficulties

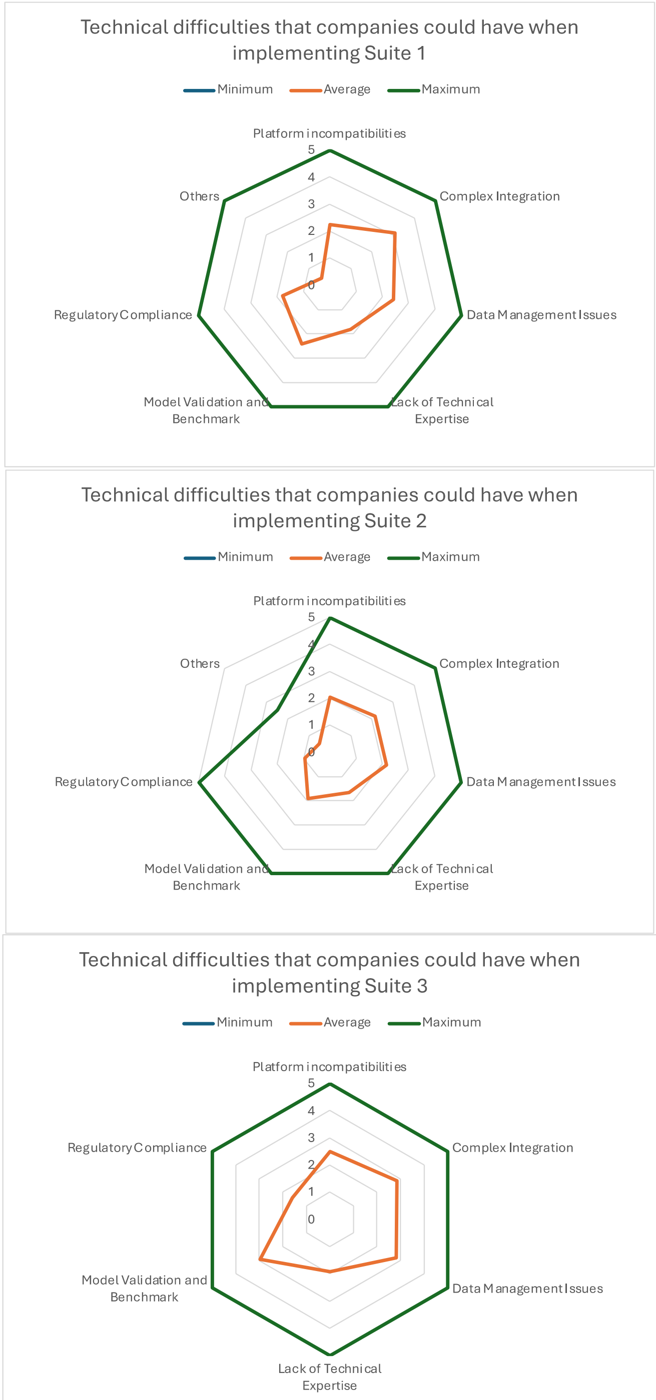


Figure 19. Technical difficulties by Suites.

Figure 20 shows the difficulty the respondents estimated according to Platform incompatibilities, Complex integration, Data management issues, Lack of technical expertise, Model validation and benchmark, Regulatory compliance, and Others when implementing the Suites. Each of them shows the minimum value (which in all cases is 0 - no difficulty), the average value and the maximum value given by the respondents.

In **Suite 1**, all technical issues have been rated with a maximum of 5.0, indicating that they are perceived as extremely difficult by some of the companies. *Complex integration* is the most important technical difficulty, with an average score of 3.1, indicating that integrating the suite with existing systems represents a considerable challenge. It is closely followed by *Data management issues, Model validation and benchmark*, and *Platform incompatibilities*, with mean scores of 2.42, 2.41 and 2.25, respectively. Lesser concerns include *Regulatory compliance* and *Lack of technical expertise*, with mean scores of 1.79 and 1.81, suggesting that while these are challenges, they are more manageable in the application context. In the *Others* category, which received the lowest mean score (0.4), respondents specified challenges such as "budget" and "the way we create and store data," indicating that there are specific budget and data management concerns that also require attention.

For **Suite 2**, Data management issues emerge as the most prominent technical problem, with an average score of 2.16. It is followed by *Complex Integration*, with an average of 2.14, which points to difficulties in integrating the suite with existing systems and processes. *Platform incompatibilities* also present notable challenges, with an average of 2.05, reflecting compatibility issues that can affect efficient implementation. Other technical issues, such as Model validation and benchmark and Lack of technical expertise, have mean scores of 1.90 and 1.67, respectively, indicating moderate concerns about model validation and the availability of adequate technical expertise. *Regulatory compliance* shows the lowest technical concern, with a mean score of 0.95. The *Others* category has a low mean score of 0.5 and no respondent specified additional difficulties.

In **Suite 3**, where all technical difficulties also reach maximum values of 5.0*, Model validation and benchmark, Complex integration*, and *Data management issues* are the most significant, with mean scores of 2.95, 2.84 and 2.83, respectively. Platform incompatibilities, with a mean of 2.5; *Lack of technical expertise*, with a mean score of 1.94; and *Regulatory compliance*, with a mean of 1.59, reflect relevant but less critical concerns. In the Others category, which has a mean of 0.79, one respondent specified *'Customised database'* as a difficulty. Thus, *Complex integration and Data management* issues emerge as consistently prominent problems in all suites, reflecting the universality of these challenges in technology integration and complex data management. In addition, Figure 20 shows the level of difficulty estimated by respondents for companies of different sizes. With them, we can determine if the technical difficulties perceived by the companies are the same regardless of size or if, on the contrary, they are different, being able to find different types of potential customers.

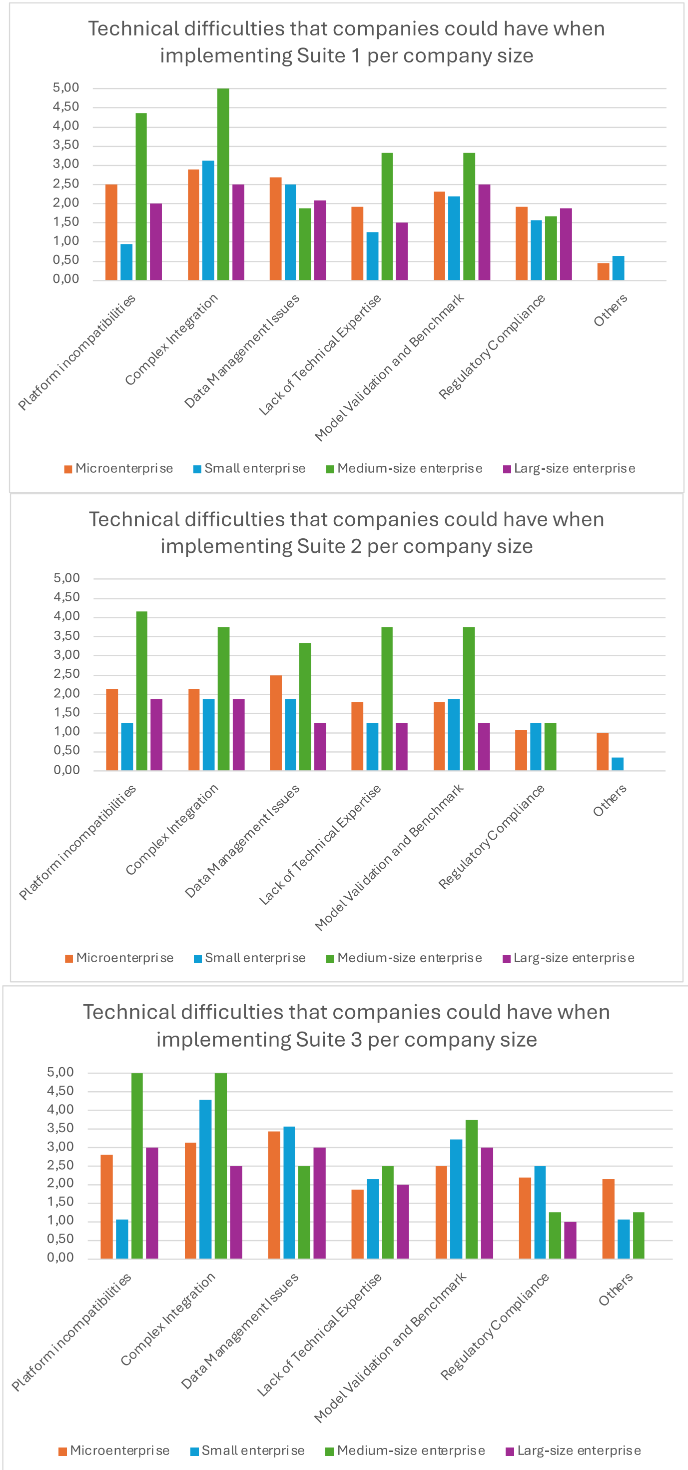


Figure 20. Technical difficulties by Suites and company size.

In **Suite 1**, medium-sized companies perceive that they have greater technical difficulties for its implementation, highlighting the high values given to *Complex Integration* (5), *Platform Incompatibilities* (4.38), *Model validation and benchmark* (3.33), and *Lack of Technical Expertise* (3.33). On the other hand, the responses given by companies of different sizes when evaluating technical difficulties show a similar pattern. Both micro, small, and large companies seem to perceive the technological difficulties of implementing this Suite as low or medium.

For **Suite 2**, it is again the medium-sized companies that perceive the greatest difficulty, with high scores on the Platform Incompatibilities (4.17), *Complex Integration* (3.75), *Lack of Technical Expertise* (3.75), *Model validation and benchmark* (3.75), and *Data management issues* (3.33). Both micro, small, and large companies seem to perceive the technological difficulties of implementing this Suite as low or medium.

Regarding **Suite 3**, it can be seen how companies of all sizes consider *Complex Integration*, Model validation and benchmarking as some of the technological difficulties of this Suite. Additionally, it reveals considerable difficulties in *Data Management Issues* for small and medium-sized companies, with mean scores of 3.57 and 3.57, respectively, indicating significant challenges in data management that are less pronounced in large companies.

3.3.3 Costs perspective

The assessment of price by companies can be closely related to the importance of the processes impacted by the Suite and its potential for improvement. If the affected processes are deemed highly important for operational and strategic success, they will likely pay a higher price for a solution that significantly enhances their performance. Similarly, if implementing the Suite holds considerable potential for improvement in these critical processes, companies may perceive greater value in the investment, positively influencing their willingness to pay a higher monthly cost for using the software. Ultimately, based on its importance and improvement potential, the perceived value that companies attribute to the Suite can significantly impact pricing strategy and product acceptance in the market.

The relationship between price valuation and company size can vary depending on several factors. Generally, larger companies may have greater financial capacity to invest in technological solutions, potentially increasing their willingness to pay higher prices for products offering significant benefits. Thus, a positive relationship may exist between company size and willingness to pay higher prices for the Suite, particularly if they perceive substantial value based on process importance and improvement potential. However, it's also important to consider that larger companies may have more complex organizational structures and elaborate processes, increasing their cost sensitivity and making them more demanding regarding value for money. In this case, there could be a negative relationship between company size and willingness to pay higher prices if they feel the Suite doesn't justify the additional cost compared to other alternatives.

In summary, the relationship between company size and willingness to pay for the Suite can vary depending on perceived value, financial capacity, and organizational complexity and cannot be easily generalized.

3.4 DIMAT BUSINESS VIEWPOINT 2ND APPROACH CONCLUSIONS

From a business perspective, the suites that raised stakeholder interest were analysed. This is an extended approach to the first analysis included in deliverable 3.3. Regarding the main aspects identified in this analysis about the suites, it can be indicated that:

A) Relating to the assets of interest and impact.

   Companies of *metals and alloys, glass, and composite materials* show potential for investment in Suite1 because the expected improvement is high, and the process impacted by the Suite is also important for the organization.

B) This investment potential appears for Suite 2 in companies of polymers, glass and composite materials, and for Suite 3 in companies of *glass and metal and alloys mainly, but also in companies of composite materials, polymers and other materials.*

C) Relating to the needs covered by the suites.

D) The responses obtained for the three suites show that all suites are perceived as effective in improving *Operational efficiency*. This is the most selected need by respondents in all of them. Also, *Profitability* and *Revenue Growth* are needs that they highlight that can be covered by Suites. In addition, financial health and Human capital are consistently the least selected across all suites.

E) Relating to organizational difficulties for their implementation.

   A common theme across all suites is the trouble related to *Insufficient Budget,* indicating that financial constraints are a universal barrier to successful implementation. In the same way, *Lack of Training* arises as a critical issue to maximize the use of these suites. Moreover, the suits present differences mainly in relation to Resistance to *Change and Lack of Leadership*, whereas Suite3 presents higher change resistance, possibly due to its specific features or the contexts in which it is deployed.

   <u>Micro companies</u> demonstrate significant concern regarding *Insufficient Budget* in Suite 1. In Suite 2, *Lack of Training* and *Lack of Alignment with Business Processes* are

notable for both categories. Suite 3 implementation reveals *Resistance to Change* and *Lack of Training* as major challenges. In Suite 1, *Lack of Training* is a greater obstacle for small companies, with a mean of 2.81. In Suite 2, small companies report fewer difficulties overall, though *Lack of Leadership* is more relevant. Medium companies express concern for I*nsufficient Budget and Lack of Alignment with Business Processes* in Suite 1. In Suite 2, they show the greatest concern in *the Lack of Training and the Lack of Alignment with Business Processes.* Suite 3 implementation highlights Insufficient Budget and Lack of Alignment with Business Processes as major challenges. Large companies in Suite 1 exhibit less concern regarding *Insufficient Budget*. In Suite 2, they show less concern for *Insufficient Budget*. In Suite 3, the Lack of Leadership is notable. Insufficient Budget is a significant concern across all company sizes, particularly for micro and medium-sized companies. *Lack of Training* is also a common challenge, especially for micro and small companies. *Resistance to Change* is particularly noticeable among medium-sized companies. *Lack of Leadership* emerges as a critical challenge for large companies.

F) Relating to technical difficulties for their implementation.

The analysis reveals that in Suite 1, *Complex integration* stands out as the most critical issue, closely followed by *Data management issues*, *Model validation and benchmark*, and *Platform incompatibilities*. Regulatory compliance and *Lack of technical expertise* present lesser concerns. In Suite 2, Data management issues are the most prominent, followed by Complex Integration and Platform incompatibilities. Suite 3 shares similar technical difficulties with Suite 1. Consistently across all suites, *Complex integration* and *Data management issues* emerge as primary challenges, emphasizing the widespread complexity in technology integration and data management.

Small companies perceive medium or low technological difficulties for Suite 1 and 2 but significant challenges in Data Management Issues in Suite3. Micro companies share a similar perception with small companies. Medium-sized companies perceive the greatest difficulty in all suites, particularly in Suite 1 and 2. Finally, Large companies perceive technological difficulties as low to medium across all suites. They share similar challenges with medium-sized companies, particularly in *Complex Integration*, *Platform Incompatibilities*, *Lack of Technical Expertise, Model validation and benchmark,* and *Data management issues.*

G) Relating to the cost perspective.

Companies are willing to pay a higher price for a suite when they identify high importance of the processes impacted by the Suite and a high potential improvement. Also, the larger the company's size, the larger the money willing to pay. Although some exceptions appear in some large companies.

4 USAGE 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. 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 the functional map, the specifications of the roles responsible for the tasks, and the implementation maps. Each task's triggers for each activity are defined, considering the workflows, effects, and constraints.

4.1 USAGE VIEWPOINT APPROACH FOR DIMAT

4.1.1 Usage Viewpoint Concepts

The Usage Viewpoint will build on the objectives 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 define the system requirements and 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.

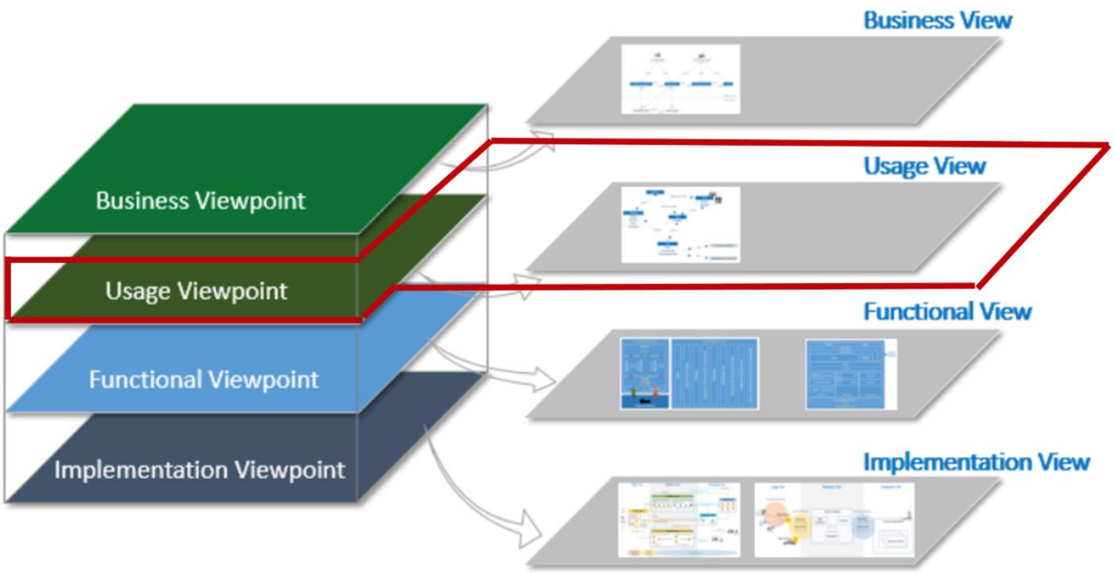


Figure 21. Usage Viewpoint layer

Co-funded by the European Union

4.1.2 Usage Viewpoint Methodology

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 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 executing certain tasks and is generally associated with certain security properties, such as authorizations, privileges, and permissions. Roles can be assumed by one or several parties. 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. In this second delivery of D3.3, toolkits may be updated if the toolkit requires it. If the toolkit has no changes from the previous delivery, this section is not included in the viewpoint and will remain the same as in the first deliverable submitted in M9.
In addition, this document includes new sections to complete the toolkit's architecture and activities with schematic diagrams, as their names indicate, consisting of a functional map and a summary matrix. In the Functional Map, the idea is to represent the toolkit's General Architecture, correlating a functional viewpoint with the activity's diagrams and their explanation, and in the Summary Matrix, it determines through a table the connections between toolkit functions and activities.

4.1.3 Overview

This section briefly resumes the general information about each toolkit, as explained in the first document version **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if the toolkit requires it. If the toolkit has no changes from the previous delivery, this section is not included in the viewpoint and the Overview section will remain the same as in the first deliverable submitted in M9.

4.1.4 Roles

Describes the different roles involved in the toolkit. Each role includes a collection of capacities for using the system. In this second delivery of D3.3, toolkits may be updated if required. In that instance, there are no changes from the previous delivery. This section is not included in this document, and the Roles section will remain the same as in the first submission of M9.

4.1.5 Activities

Here, all the toolkit activities should be included; they are explained in the first document version **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may update it in case the toolkit requires it. If there are no changes from the previous delivery, this section is not included in this document, and the Activities section will remain the same as in M9.

4.1.5.1 Triggers

One trigger is a specific action that starts a process or causes a specific function. An example of a trigger is the button that a user explicitly selects to start an event, like "Add new data source". All trigger information is explained in detail in the first submission of **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if the toolkit requires it. If there are no changes from the previous delivery, this section is not included in this document, and the Triggers section will remain the same as in the first delivery of M9.

4.1.5.2 Workflow

Workflows represent graphically the execution of several tasks through an activity diagram. These tasks can be shown sequentially, in parallel, conditionally, repeatedly, etc. UML Activity Diagrams were defined in the first delivery **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if the toolkit requires it. If there are no changes from the previous delivery, this section is not included in this document, and the Workflow section will remain the same as in the first delivery of M9.

4.1.5.3 Effects

Each activity could register some permanent effects in the system. All about effects information is explained in detail in the first document **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if required. In that instance, there are no changes from the previous delivery. This section is not included in this document, and the Effects section will remain the same as in the first deliverable of M9.

4.1.5.4 Constraints

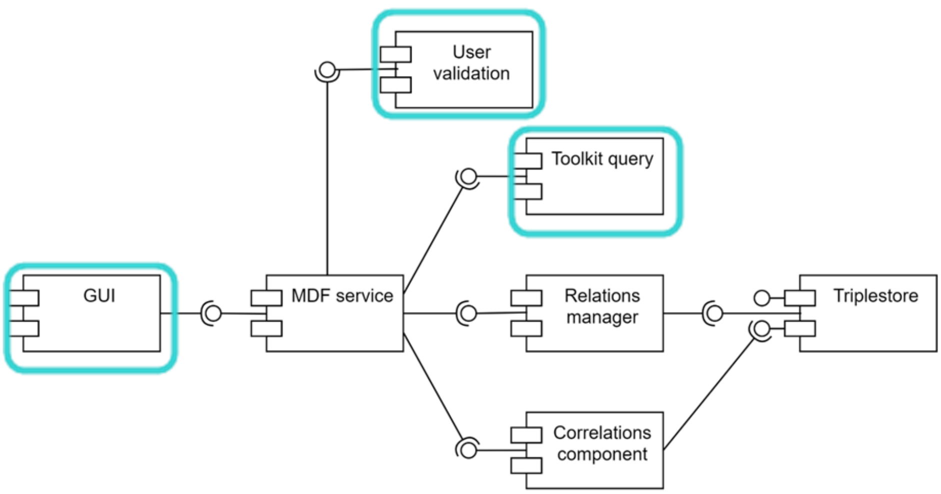
This section will show all the constraints that should be considered by the system. The list of constraints information is explained in detail in the first document, **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if required. In that instance, there are no changes from the previous delivery. This section is not included in this document, and the Constraints section will remain the same as in the first deliverable of M9.

4.1.5.5 Mock-ups

In this part, all mock-ups will appear designed to visualize, for example, the homepage screen, login page and display of visualization results page. The mock-up collection is detailed in the first document, **D3.3 VIEWPOINTS**. In this second delivery of D3.3, toolkits may be updated if required. In that instance, there are no changes from the previous delivery. This section is not included in this document, and the Mock-up section will remain the same as in the first deliverable of M9.

4.1.6 Functional map

This section is new from the first delivery and includes a table representing the toolkit's General Architecture. This General Architecture is split into Activities, adding a specific activity diagram, and accompanied by an explanation of the relationship between the activity and its functions, ensuring a better understanding.

<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">This section must include the General Architecture diagram of the toolkit included in the D3.3. VIEWPOINTS submitted in M9. Below, an example of a General Architecture diagram is shown.

</td></tr>
<tr><th>Activity diagram</th><th>Explanation</th></tr>
</table>

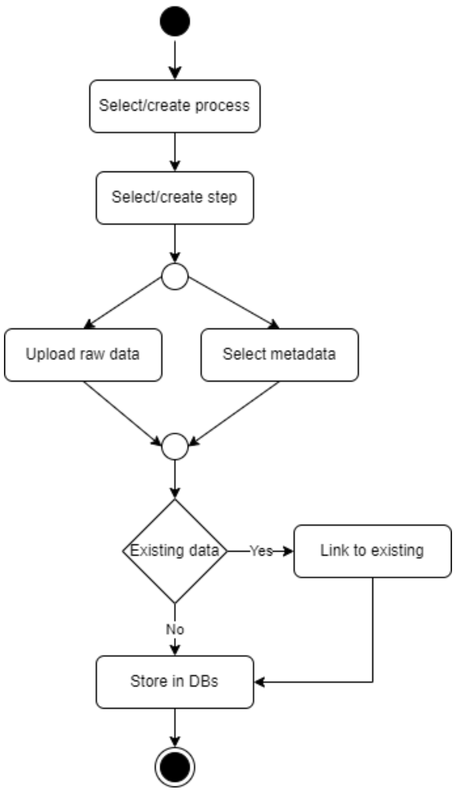
This cell must show a workflow of an activity, based on an illustration that represents the different functions of the activity linked by lines. An example is shown below: 	This cell is at the same level as the Activity diagram cell, because it briefly describes which function dependencies are drawn in the diagram and which is the objective of each function.

Table 9. Functional Viewpoint Architecture Diagram

4.1.6.1 Summary Matrix

This section is also new from the first delivery and must include a table that provides a concise summary and a classification of the functions required in the diagram. Each function is associated with the corresponding activity, indicated by an “X” in the respective cells. The rows represent the toolkit activities, while the columns correspond to functions shown in the diagram. This table format allows a more organized view of understanding the relationship between functions and activities.

		FUNCTIONS			
		FUNCTION	FUNCTION	FUNCTION	FUNCTION
ACTIVITIES	ACTIVITY	X	X	X	X
	ACTIVITY			X	X
	ACTIVITY			X	X
	ACTIVITY	X	X		

Table 10. Functions

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

The Cloud Materials Data Base (CMDB) offers homogeneous access to heterogeneous data storage. Processes linked to a CHADA and MODA definition and their different steps connected to material data will be represented. 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.

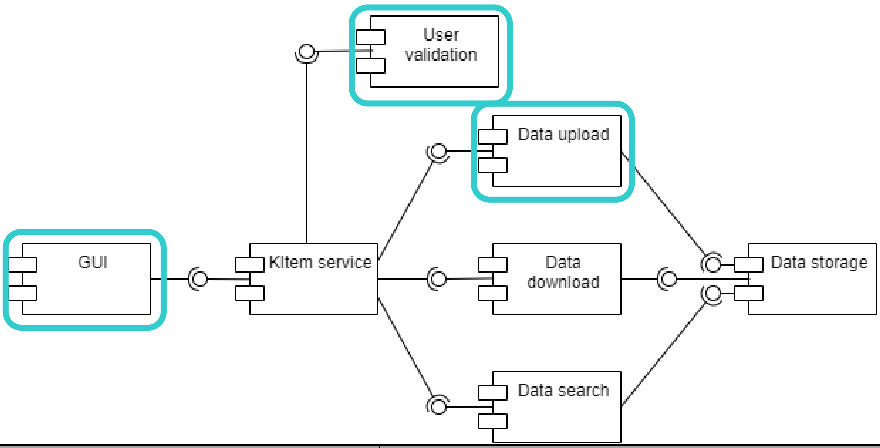
4.2.1 Activities

4.2.1.1 Activity 1

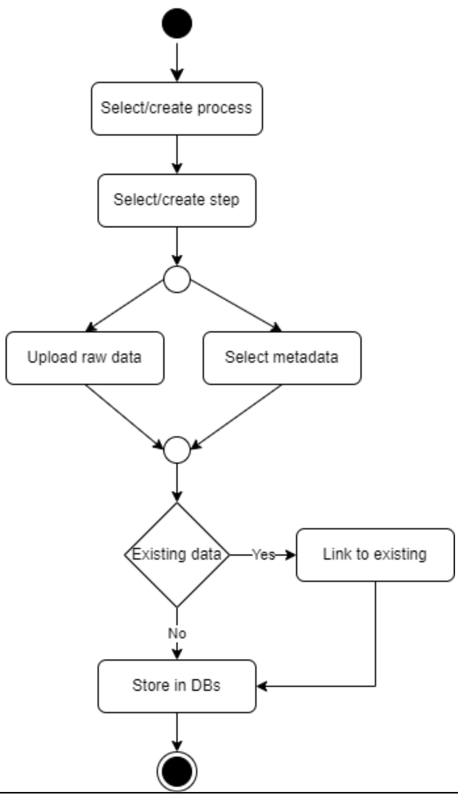
This activity covers how users interact with the CMDB to upload new datasets.

4.2.1.1.1 Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">



</td></tr>
<tr><th>Activity 1 diagram</th><th>Explanation</th></tr>
<tr><td>



</td><td>

Activity 1 links with the function "Data Upload" because the objective is to generate new knowledge, which involves adding new files to the CMDB.

"GUI" and "User validation" are needed for a basic, secure interaction.

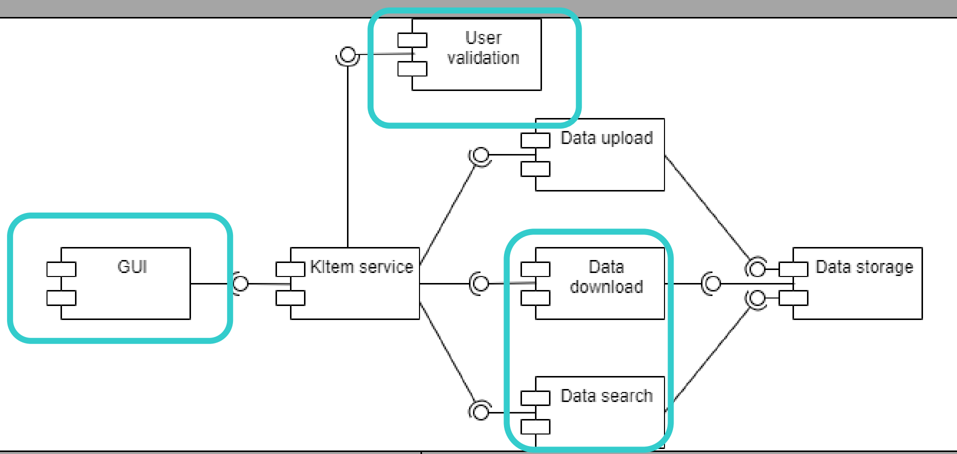
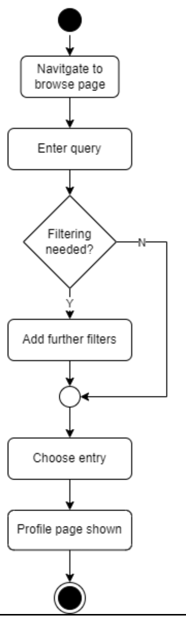
</td></tr>
</table>

Table 11. FV Architecture DiCMDB - Activity 1

4.2.1.2 Activity 2

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.

4.2.1.2.1 Activity 2 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 2 links with functions “Data search” and potentially “Data download” because the objective is to browse the existing data, which means searching through it and optionally downloading some relevant data found.

“GUI” and “User validation” are needed for a basic, secure interaction.</td></tr>
</table>

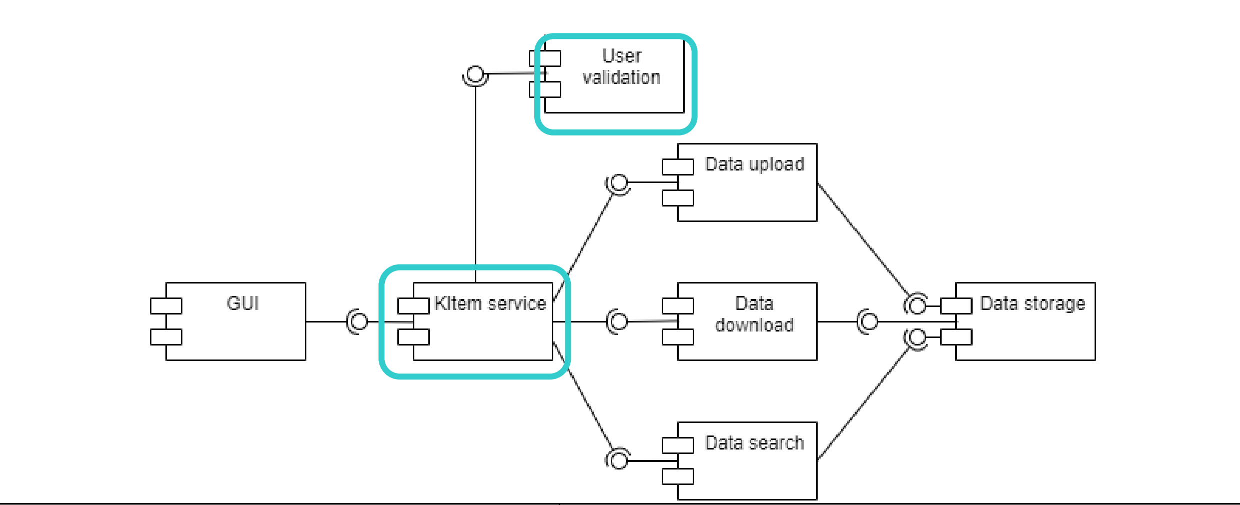
Table 12. FV Architecture DiCMDB - Activity 2

4.2.1.3 Activity 3

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

4.2.1.3.1 Activity 3 – Functional map

Functional Viewpoint Architecture Diagram	



Activity 3 diagram	Explanation

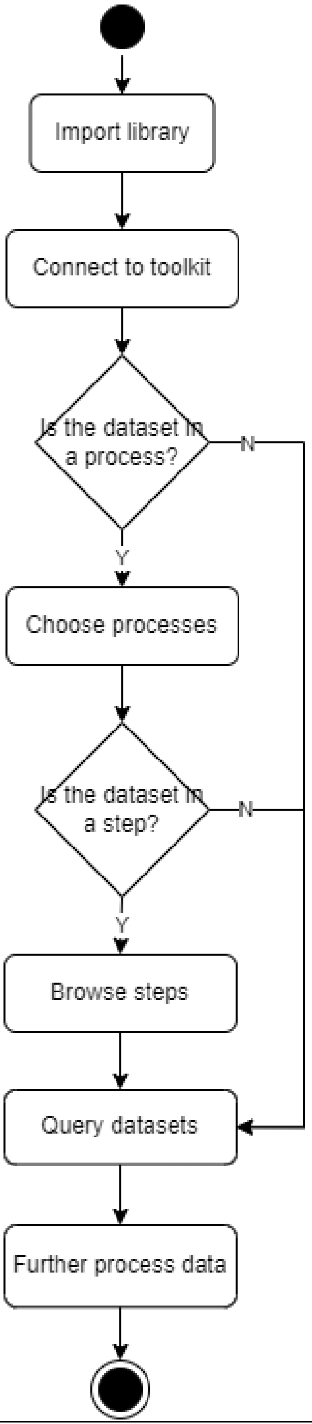
	Activity 3 links with the general function “kItem service” because the PythonSDK enables the same operations as the GUI does, but through Python scripts. “User validation” is also required when interacting via scripts (not just via GUI)

Table 13. FV Architecture DiCMDB - Activity 3

4.2.1.3.2 Summary matrix

		FUNCTIONS						
		GUI	KITEM SERVICE	USER VALIDATION	DATA UPLOAD	DATA DOWNLOAD	DATA SEARCH	DATA STORAGE
ACTIVITIES	ACTIVITY 1	X		X	X			
	ACTIVITY 2	X		X		X	X	
	ACTIVITY 3		X	X				

Table 14. FV Functions DiCMDB

4.3 DIMAT KNOWLEDGE ACQUISITION FRAMEWORK (DIKAF)

The DiMAT KAF toolkit's objective is to represent knowledge about materials in a structured manner. Besides storing information about different materials, it also represents relationships among them. The KAF toolkit is based on developing a Knowledge Graph (KG). The entities of the knowledge graph will be concepts like materials and manufacturing processes and the relationships joining these entities reveal links between these concepts. The KG will be constructed by employing existing material ontologies (e.g., the EMMO) and 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.

4.3.1 Activities

4.3.1.1 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.

4.3.1.1.1 Activity 1- Functional map

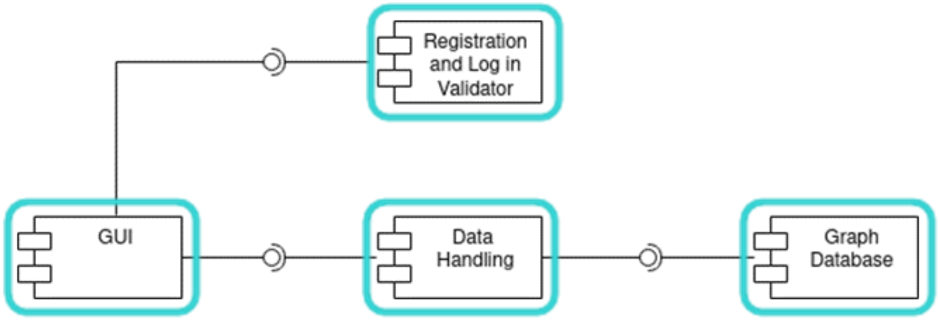
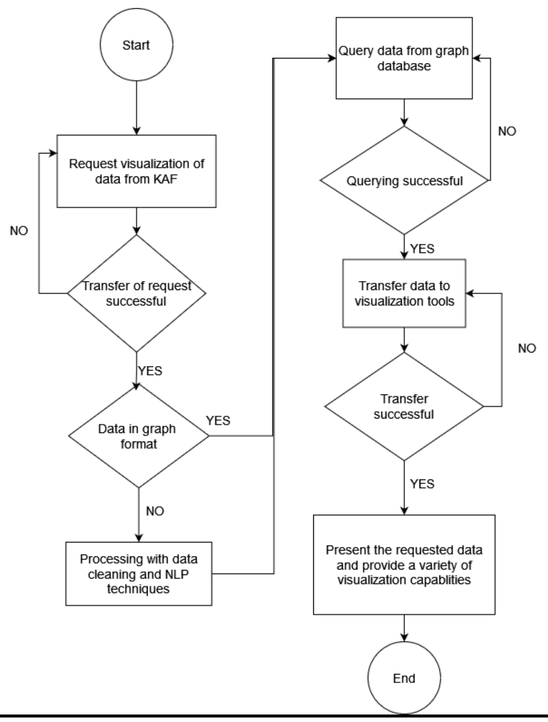
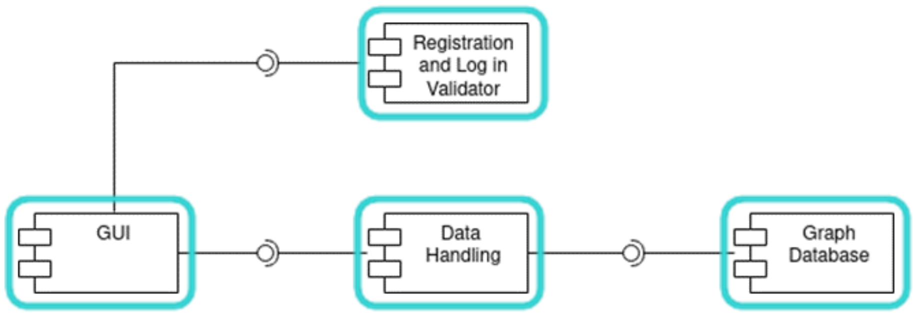
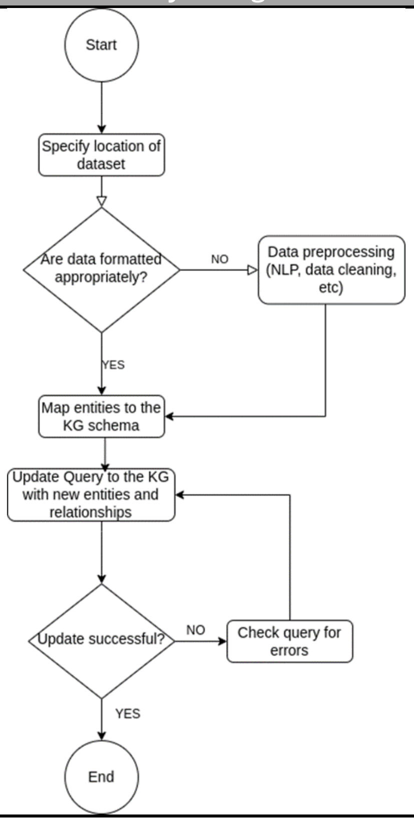
Functional Viewpoint architecture diagram	
	
Activity 1 diagram	**Explanation**
	Activity 1 links with the “Registration and Login validator” function to provide secure access to the toolkit’s services. The main part of the activity is the “data handling” function, which processes the data with cleaning or NLP techniques, and then the “Graph Database” function to retrieve the necessary data to be visualized.

Table 15. FV Architecture DiKAF - Activity 1

4.3.1.2 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 data sources deriving from ontologies, experiments, or any other sources.

4.3.1.2.1 Activity 2 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 2 links with the function "Registration and Login validator" to provide secure access to the toolkit's services.

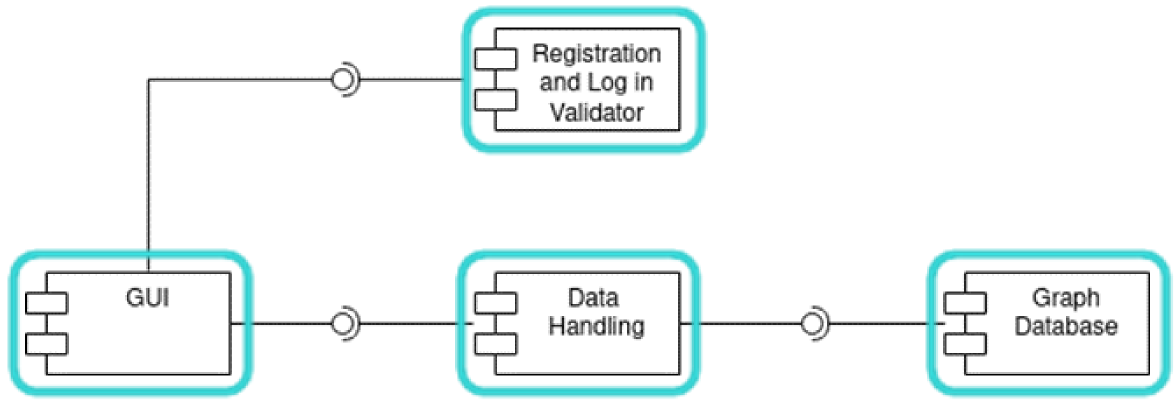
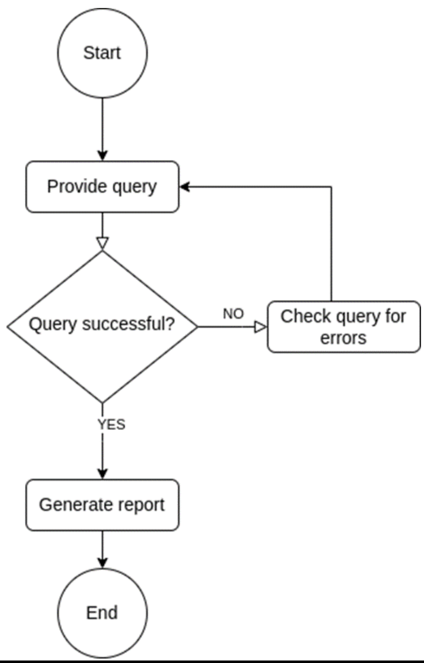
The main part of the activity is the "data handling" function so that the data from the new source are formatted appropriately (data cleaning, NLP, etc.) and then the "Graph Database" function to update the KG with the new entities and relationships.</td></tr>
</table>

Table 16. FV Architecture DiKAF - Activity 2

4.3.1.2.2 Activity 3 – "Modify Knowledge Graph"

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

4.3.1.2.3 Activity 3 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 3 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 3 links with function “Registration and Login validator” to provide secure access to the toolkit’s services.

The main parts of the activity are firstly the “data handling” function to provide the query and then the “Graph Database” function to modify the KG accordingly.</td></tr>
</table>

Table 17. FV Architecture DiKAF - Activity 3

4.3.1.3 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 consumers. Proper access rights will be provided according to the end user type.

4.3.1.3.1 Activity 4 – Functional map

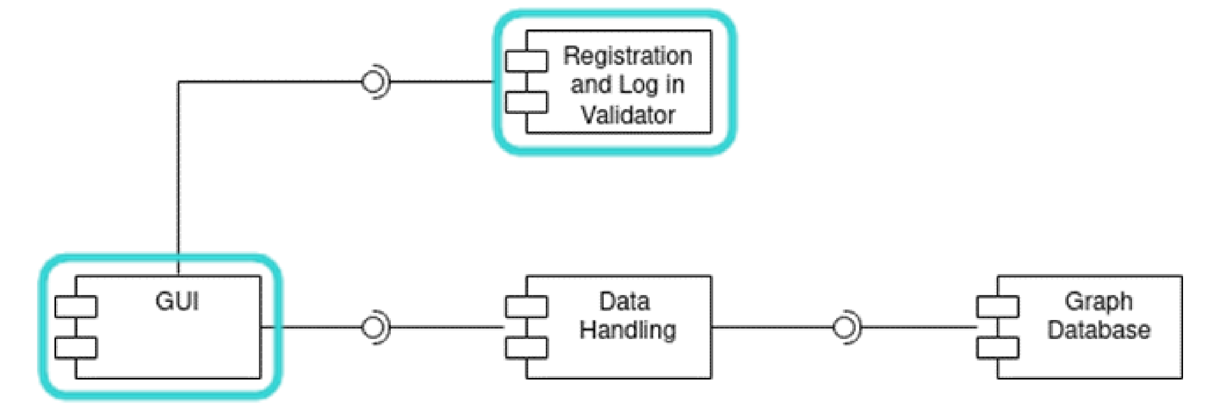
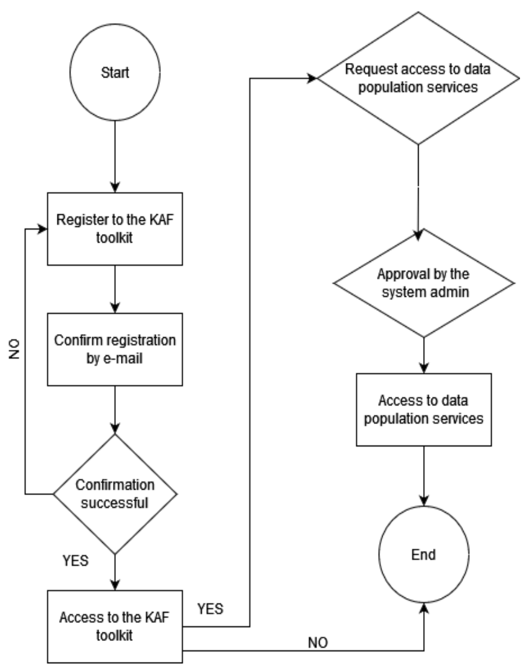
<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 4 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 4 links with the function "Registration and Login validator" for the user to create a new account, perform and confirm the registration and then perhaps request access to data population services, which he will be granted or denied.</td></tr>
</table>

Table 18. FV Architecture DiKAF - Activity 4

4.3.1.4 Summary Matrix

		FUNCTIONS			
		GUI	REGISTRATION AND LOG-IN VALIDATOR	DATA HANDLING	GRAPH DATABASE
ACTIVITIES	ACTIVITY 1	X	X	X	X
	ACTIVITY 2	X	X	X	X
	ACTIVITY 3	X	X	X	X
	ACTIVITY 4	X	X		

Table 19. FV Functions DiKAF

4.4 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 data sets. The users, such as product engineers or decision-makers, can review indicators of the environmental and cost LCAs and information on the pilots' KPIs (WP2). The platform's UI will visualise information on core life cycle assessment issues, such as identifying environmental and financial hotspots, evaluating design options performance, and benchmarking.

4.4.1 Overview

Below is the use case diagram for the MEC-LCA toolkit, updated from v1, presenting the main actors of the toolkit, as per the roles identified previously, and the main use cases or activities. The updates include the addition of: "Edit data inline", "Insert or import data" and "Grant access to specific resources".

Figure 22. Use Case diagram of Di$^{MEC-LCA}$

4.4.2 Activities

The main activities of the toolkit comprise:

- Activity 1 “Store user input”
- Activity 2 “Visualize LCA results”
- Activity 3 “Visualize KPIs”
- Activity 4 “Download data”
- Activity 5 “Log in”
- Activity 6 “Modify visualization”
- Activity 7 “Edit data inline”

Activity 7 is a new addition since the last version of the Viewpoints.

4.4.2.1 Activity 1 – Store user input

4.4.2.1.1 Activity 1 - Mockups

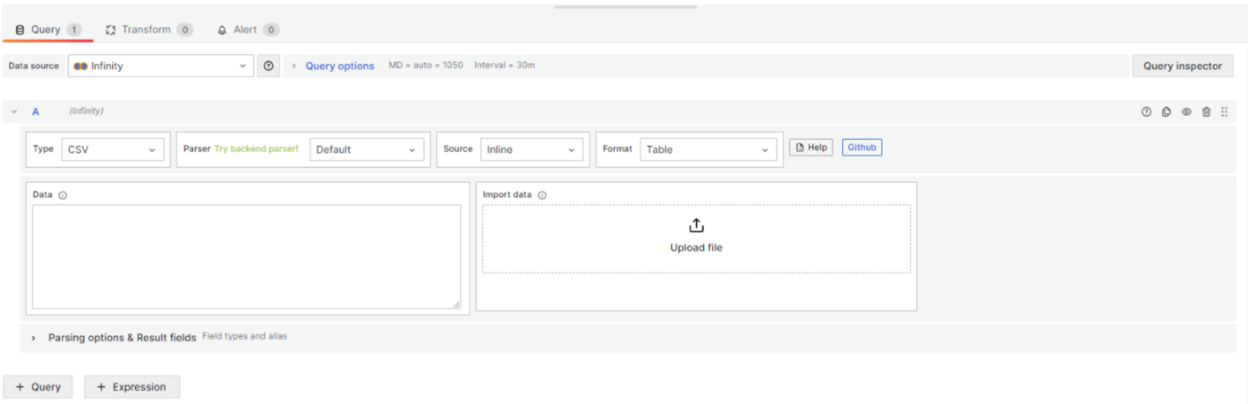
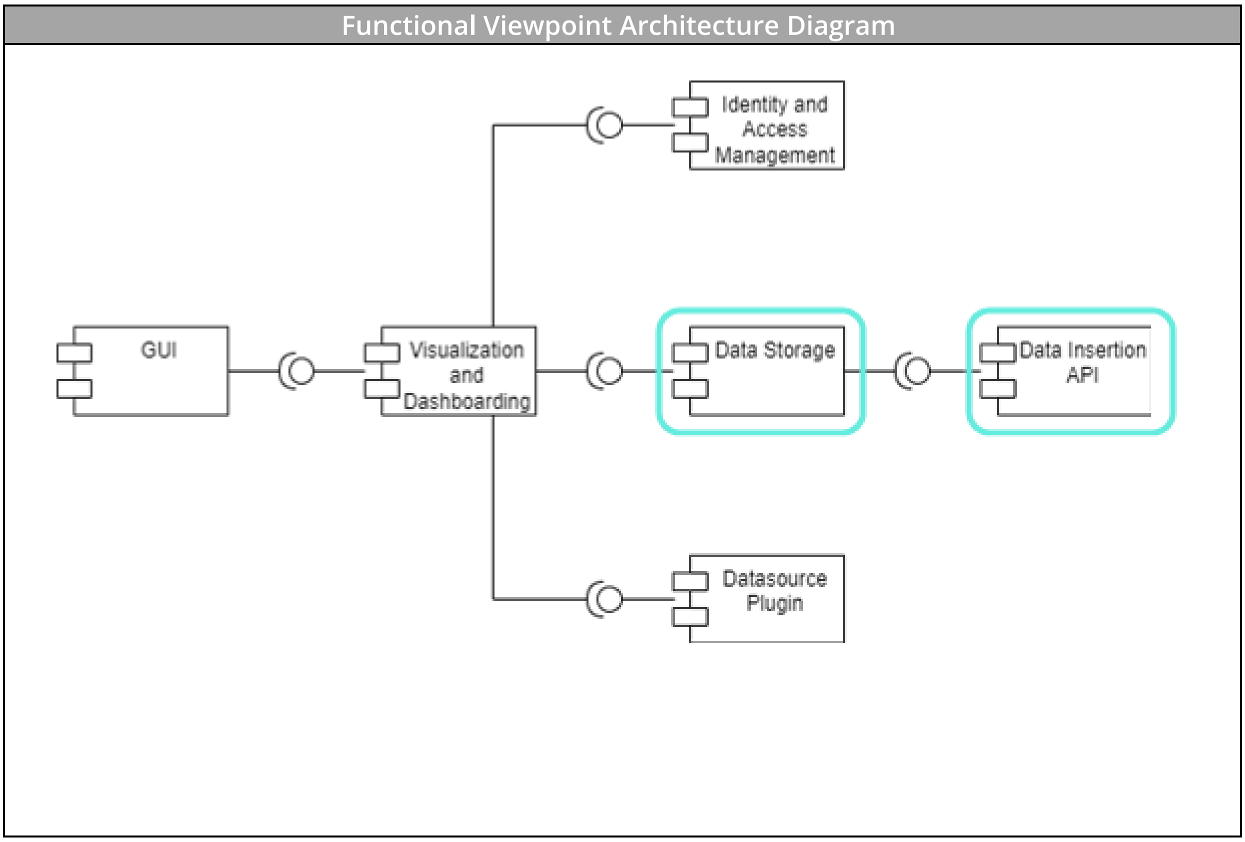


Figure 23. Mockup "Store user input " of Di$^{MEC-LCA}$

4.4.2.1.2 Activity 1 - Functional map



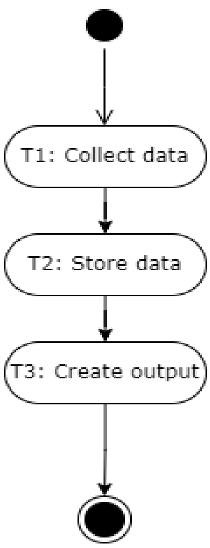
Activity 1 diagram	Explanation
	Activity 1 links with function “Data Storage”, because the objective is to store user-provided data into a storage system. “Data Insertion API” function is also required because it serves as the interface through which data is inserted into the storage system.

Table 20. FV Architecture DiMEC-LCA - Activity 1

4.4.2.2 Activity 2 - Visualize LCA results

4.4.2.2.1 Activity 2 - Mockups

The visualizations provided in the mock-up below are using dummy data as the LCA analysis is due to begin later in the project (T7.5).

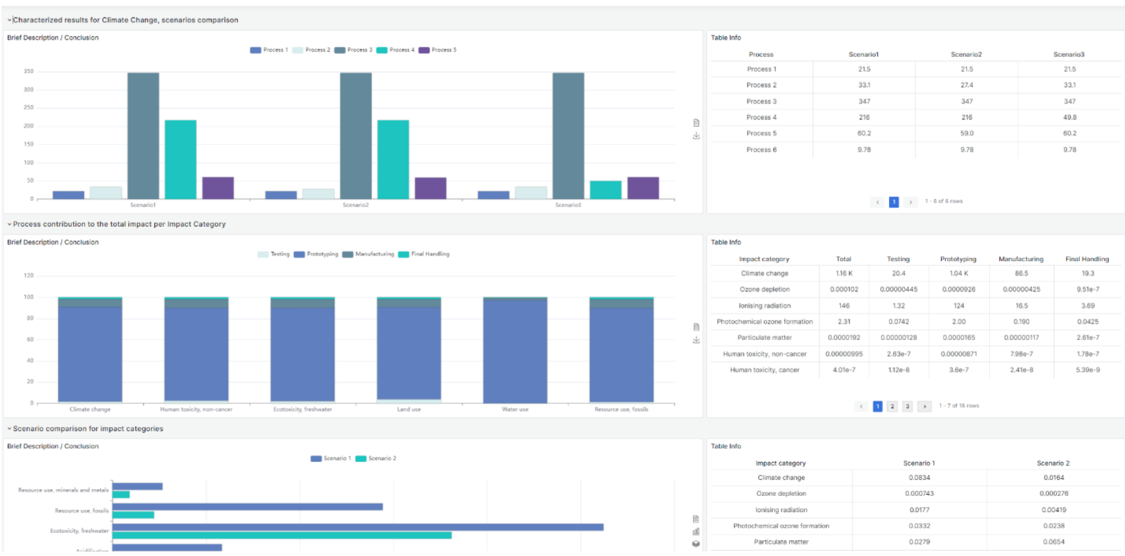
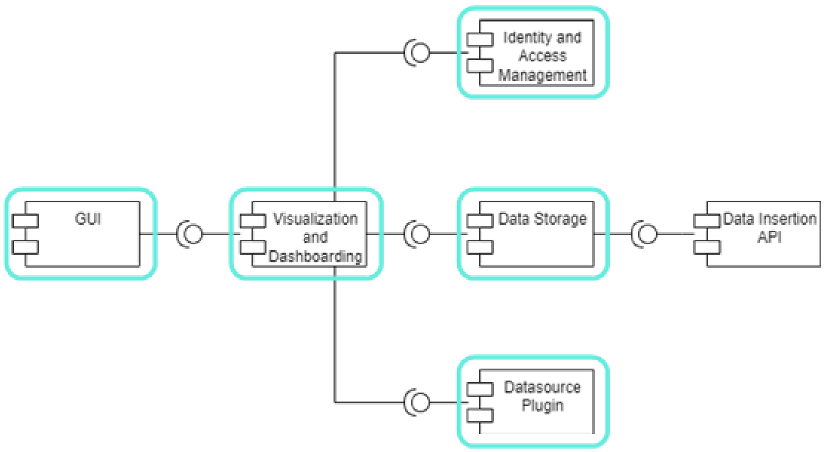
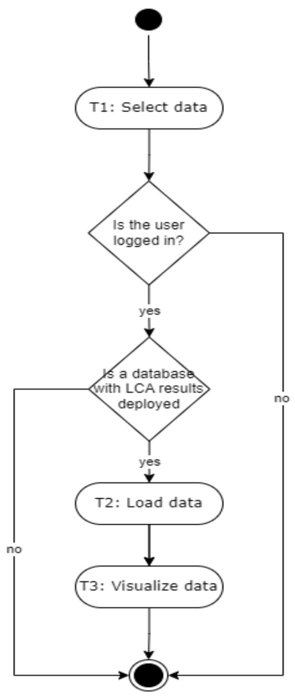


Figure 24. Mockup "Visualize LCA results of Di$^{MEC-LCA}$

4.4.2.2.2 Activity 2 - Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 2 links with functions "GUI" and "Visualization and Dashboarding", because the objective is to view LCA (and LCC) results. This requires an interface where users can view data in various visual formats, such as charts and tables, which are provided by the "Visualization and Dashboarding" function.

Activity 2 links also with functions "Data Storage" and "Datasource Plugin", because LCA results can be received from either of these functions.

"Identity and Access Management" function is also required because it ensures that only authorized users can access certain features and data within the system.</td></tr>
</table>

Table 21. FV Architecture DiMEC-LCA - Activity 2

4.4.2.3 Activity 3 - Visualize KPIs

4.4.2.3.1 Activity 3 - Mockups

The KPI dashboard for Pilot Glass is used as an example.

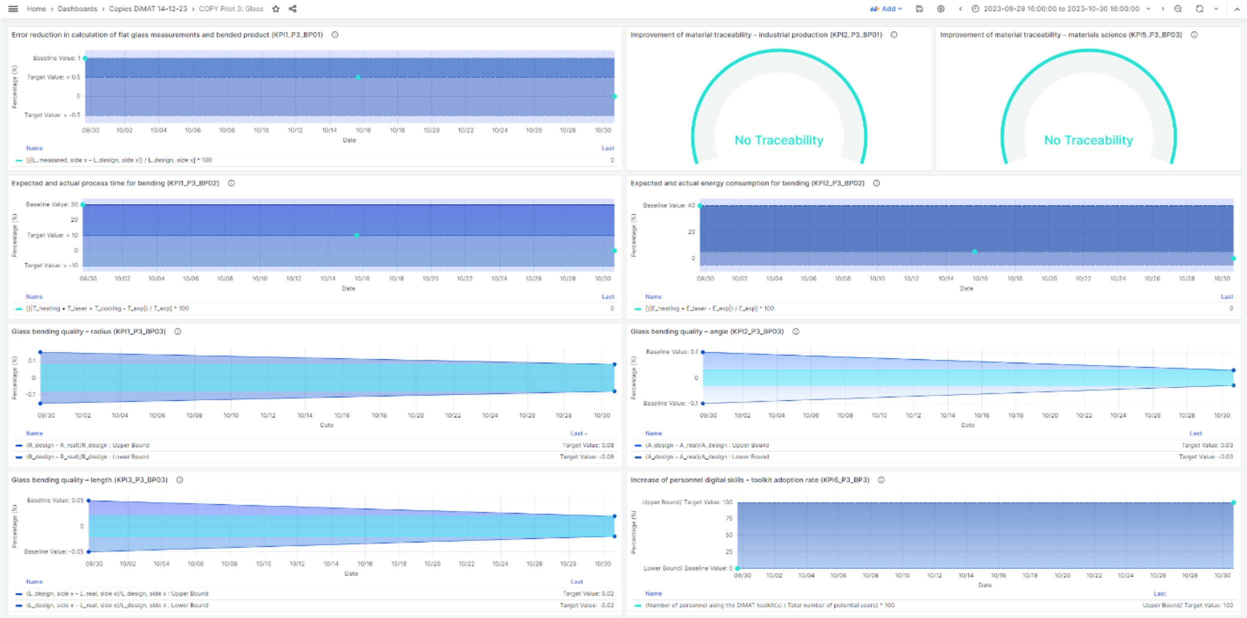
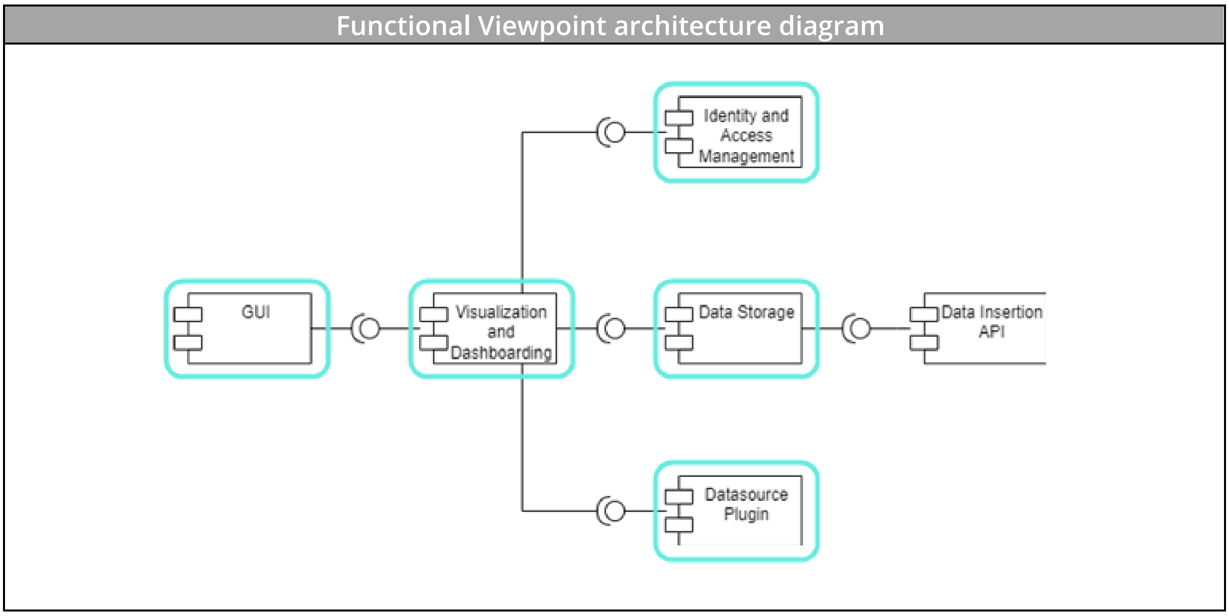


Figure 25. Mockup "Visualize KPIs" of Di[MEC-LCA]

4.4.2.3.2 Activity 3 - Functional map



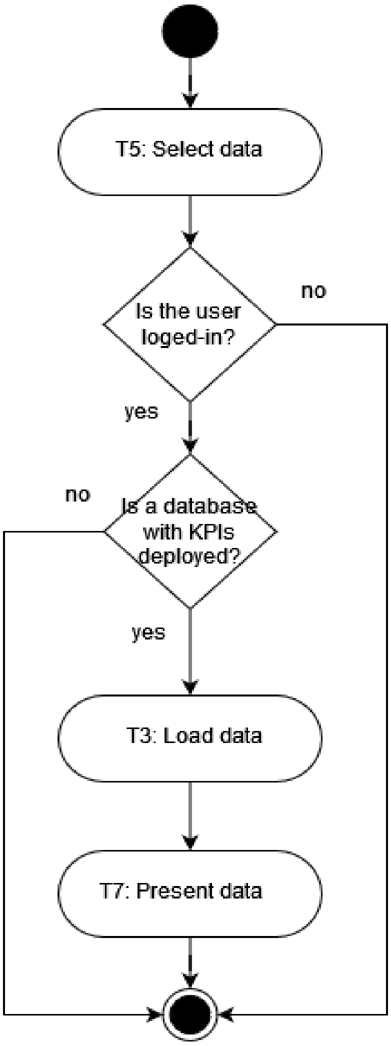
Activity 3 diagram	Explanation
	Same as Activity 2.

Table 22. FV Architecture DiMEC-LCA - Activity 3

4.4.2.4 Activity 4 - Download data

4.4.2.4.1 Activity 4 - Mockups

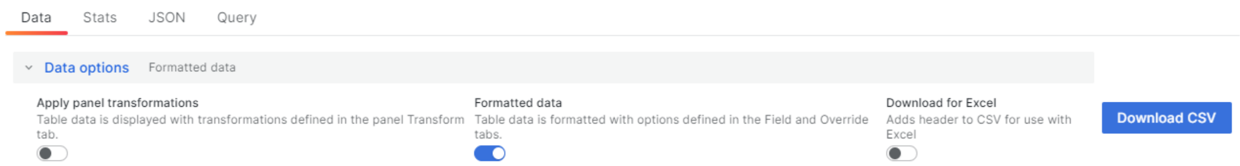


Figure 26. Mockup "Download data" of Di$^{MEC-LCA}$

4.4.2.4.2 Activity 4 - Functional map

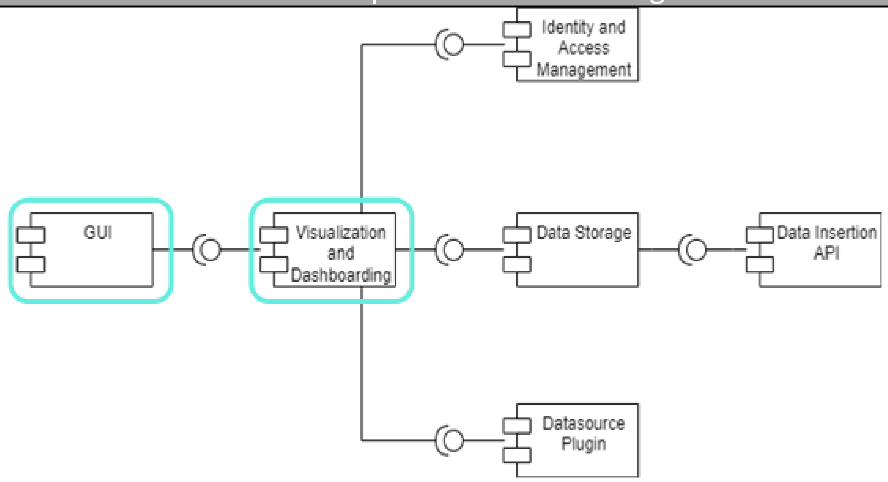
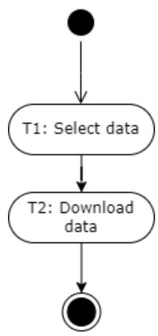
Functional Viewpoint architecture diagram	
	
Activity 4 diagram	**Explanation**
	Activity 4 links with function "GUI", because the objective is to download data, and this requires an interface where users can initiate and manage data downloads. "Visualization and Dashboarding" function are also required, because it provides the tools to manage data downloads.

Table 23. FV Architecture DiMEC-LCA - Activity 4

4.4.2.5 Activity 5 - Log in

4.4.2.5.1 Activity 5 - Mockups

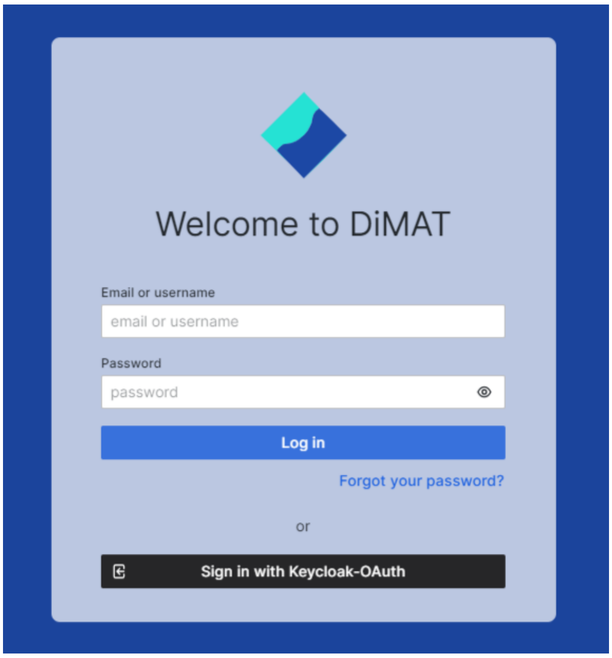
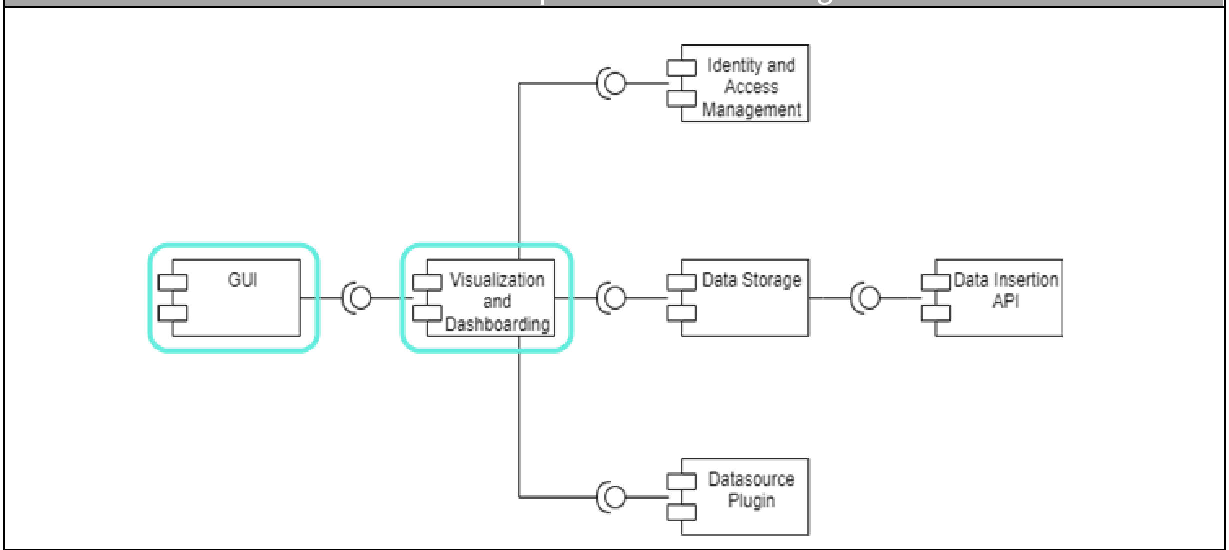
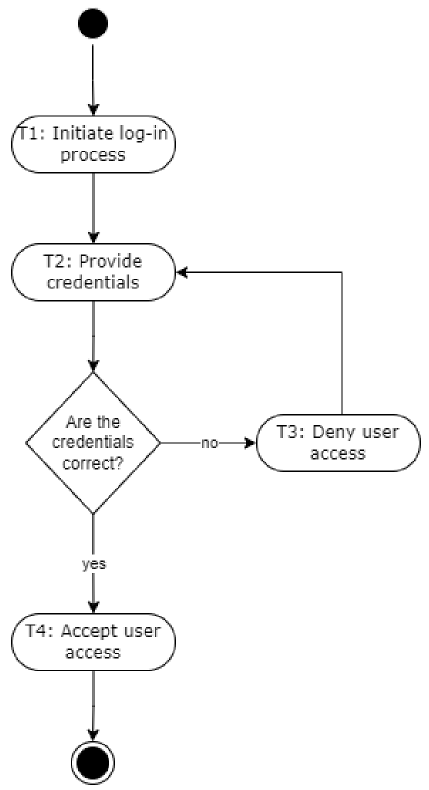


Figure 27. Mockup “Log in" of Di[MEC-LCA]

4.4.2.5.2 Activity 5 - Functional map

Functional Viewpoint architecture diagram



<table>
<tr><th>Activity 5 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 5 links with function “GUI”, because the objective is for users to access the system, and this requires an interface where users can enter their credentials.

“Visualization and Dashboarding” function are also required, because upon successful log-in the users gain access to the resources of the “Visualization and Dashboarding” function.</td></tr>
</table>

Table 24. FV Architecture DiMEC-LCA - Activity 5

4.4.2.6 Activity 6 – Modify Visualization

4.4.2.6.1 Activity 6 - Mockups

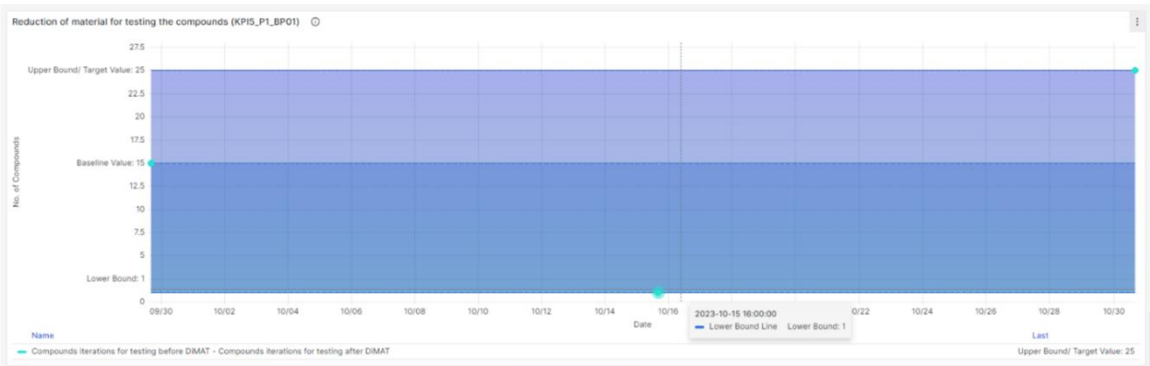


Figure 28. Mockup “Modify visualization " of Di$^{MEC-LCA}$

4.4.2.6.2 Activity 6 - Functional map

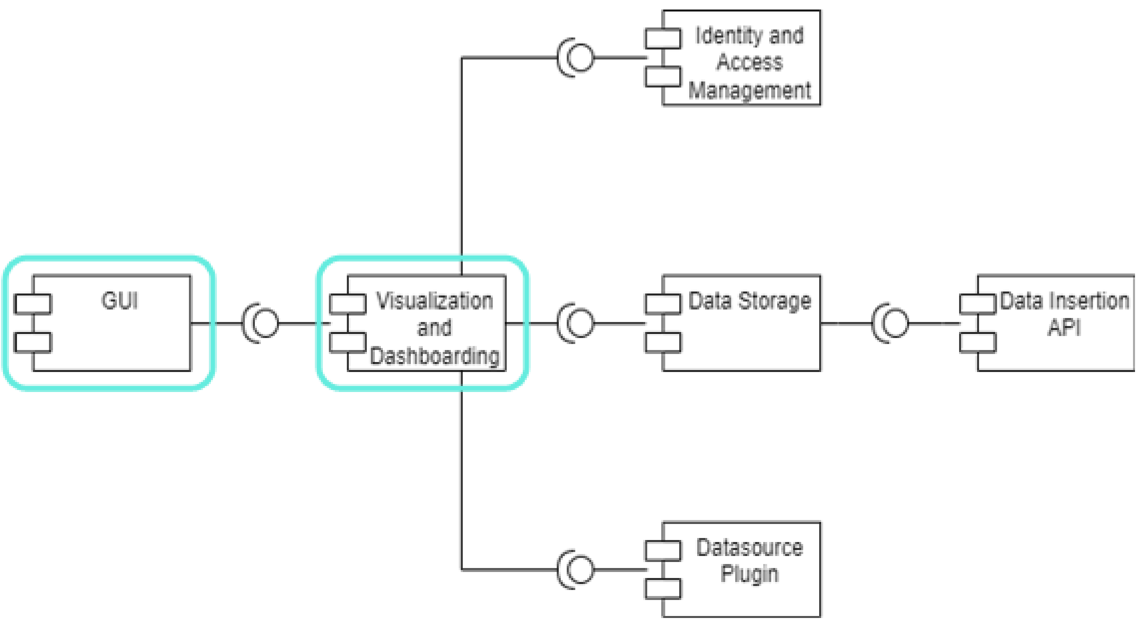
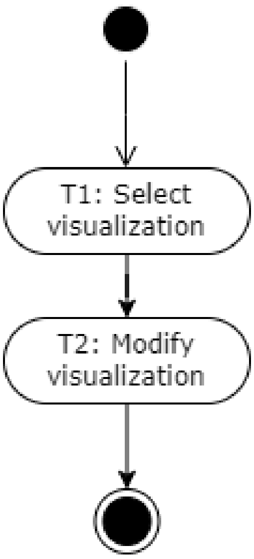
<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 6 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 6 links with functions “GUI” and “Visualization and Dashboarding”, because the objective is to modify the data presented, and this requires an interface where users can interact with the visualization’s tools, which are provided by the “Visualization and Dashboarding” function.</td></tr>
</table>

Table 25. FV Architecture DiMEC-LCA - Activity 6

4.4.2.6.3 Activity 7 - Mockups

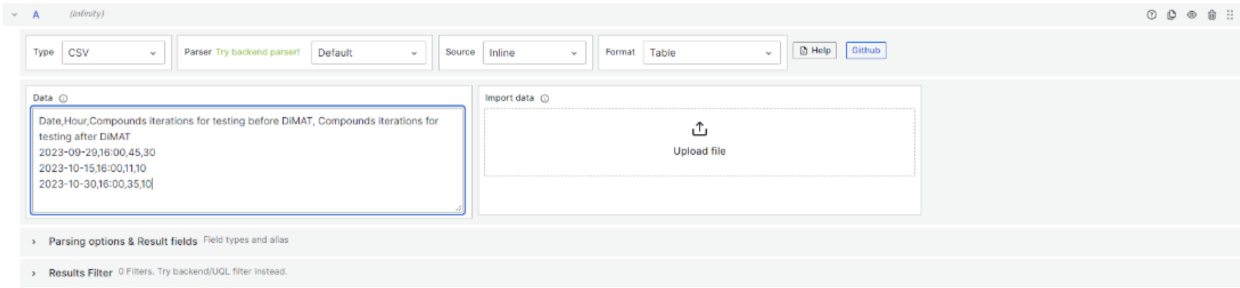


Figure 29. Mockup "Edit data inline" of Di$^{MEC-LCA}$

4.4.2.6.4 Activity 7 - Functional map

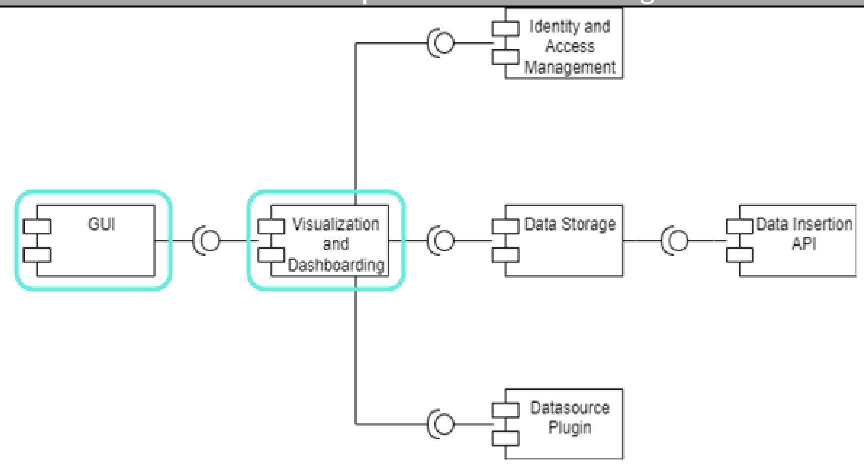
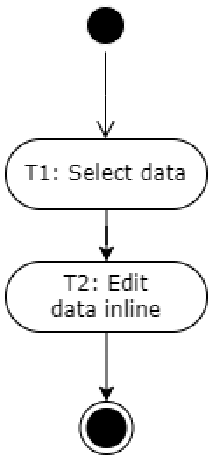
Functional Viewpoint architecture diagram	
	
Activity 7 diagram	**Explanation**
	Activity 7 links with function "GUI", because the objective is for users to interact with data directly within the interface. "Visualization and Dashboarding" function are also required, because it provides the tools to edit data inline.

Table 26. FV Architecture DiMEC-LCA - Activity 7

4.4.2.7 Summary matrix

		FUNCTIONS					
		GUI	VISUALIZATION AND DASHBOARDING	IDENTITY AND ACCESS MANAGEMENT	DATA SOURCE PLUGIN	DATA STORAGE	DATA INSERTION API
ACTIVITIES	ACTIVITY 1					X	X
	ACTIVITY 2	X	X	X	X	X	
	ACTIVITY 3	X	X	X	X	X	
	ACTIVITY 4	X	X				
	ACTIVITY 5	X	X				
	ACTIVITY 6	X	X				
	ACTIVITY 7	X	X				

Table 27. FV Functions DiMEC-LCA

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

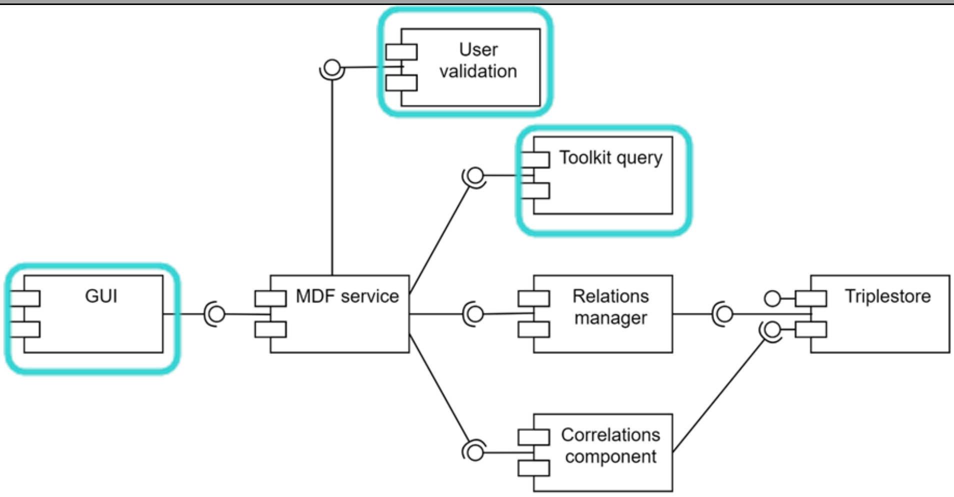
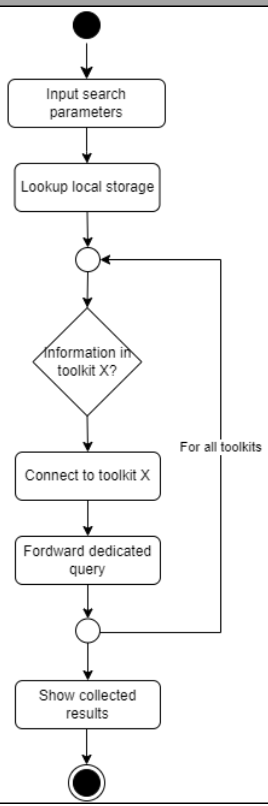
The Materials Design Framework is a toolkit that leverages the knowledge stored in the CMDB and other toolkits to provide users with suggestions of relevant data, tools, digital twins, and other information based on their query. This will result in a more informed and efficient material design, where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

4.5.1 Activities

4.5.1.1 Activity 1

A user wants to search for information contained in other toolkits.

4.5.1.1.1 Activity 1 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 1 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 1 links with the “Toolkit query” function, as it gathers information from other toolkits to present to the user.

“GUI” and “User validation” are needed for a basic, secure interaction.</td></tr>
</table>

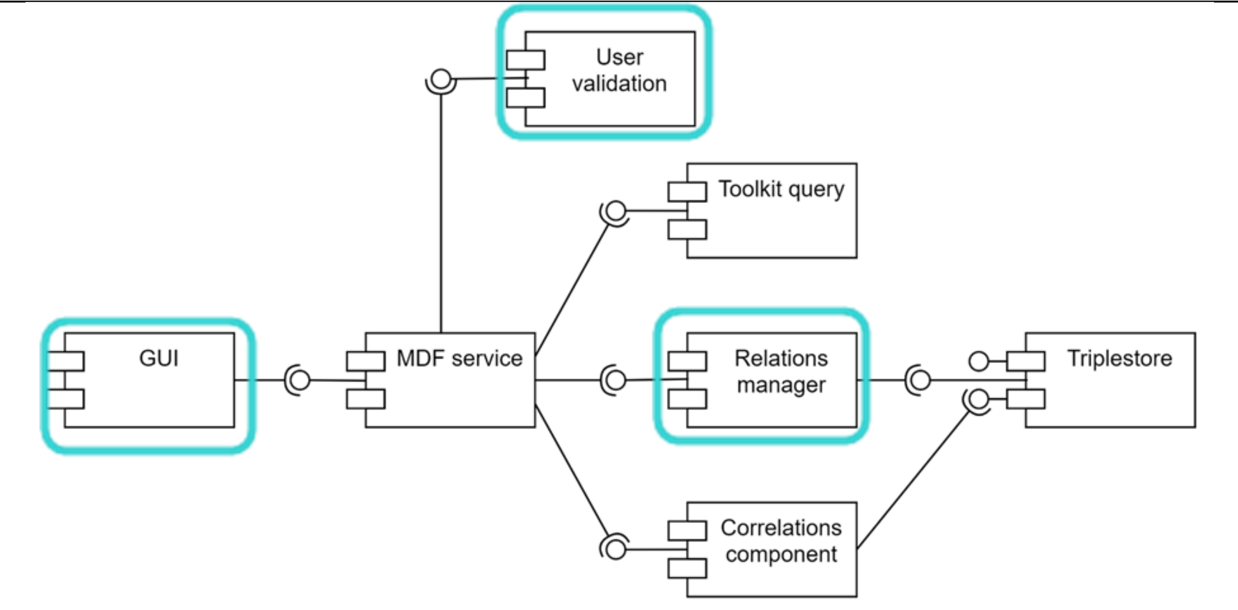
Table 28. FV Architecture DiMDF - Activity 1

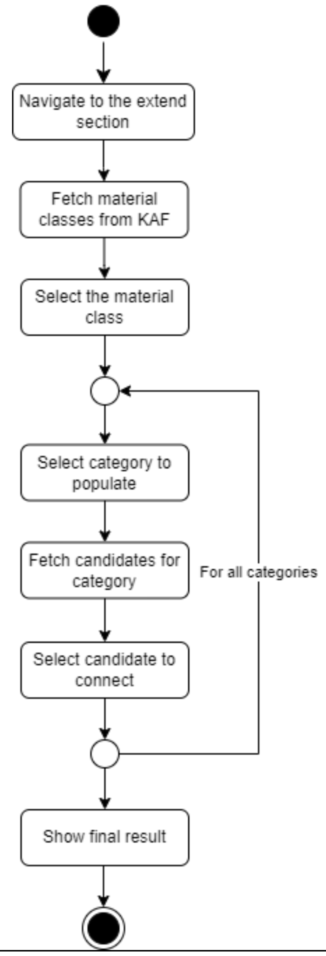
Co-funded by the European Union

4.5.1.2 Activity 2

A user needs help to find relevant materials relations for a material of interest.

4.5.1.2.1 Activity 2 – Functional map

Functional Viewpoint architecture diagram	
	
Activity 2 diagram	Explanation



Activity 2 links with the “Relations manager” function as it creates new connections with the KAF and stores the relevant information.

“GUI” and “User validation” are needed for a basic, secure interaction.

Table 29. FV Architecture DiMDF - Activity 2

4.5.1.3 Activity 3

A user wants to visualise or identify correlations contained in data that is available through the CMDB.

4.5.1.3.1 Activity 3 – Functional map

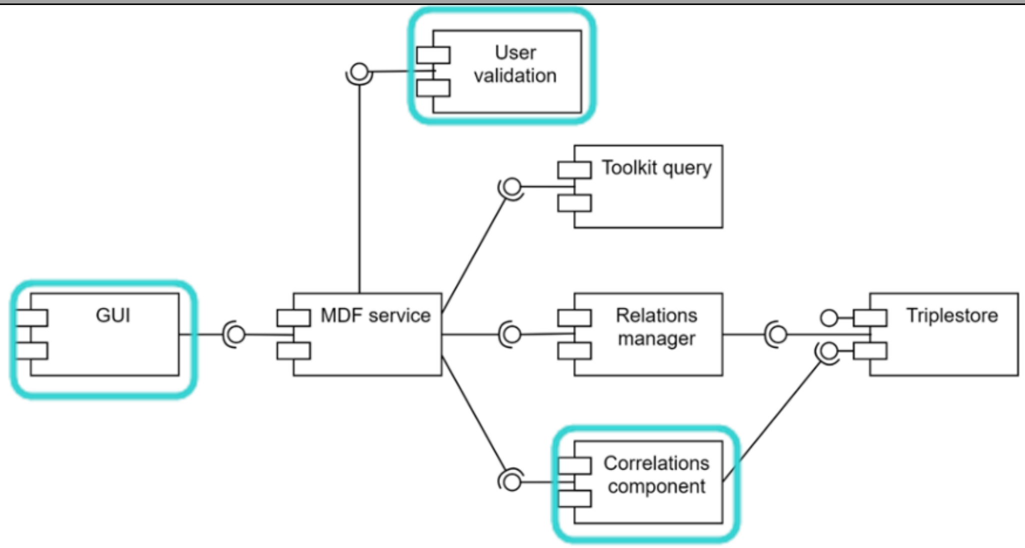
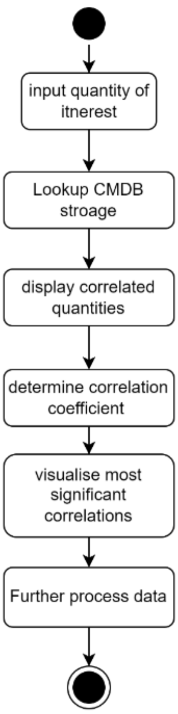
Functional Viewpoint architecture diagram	
	
Activity 3 diagram	**Explanation**
	Activity 3 links with the "Correlations component" function as it shows correlations hidden in the triple store. "GUI" and "User validation" are needed for a basic, secure interaction.

Table 30. FV Architecture DiMDF - Activity 3

4.5.1.4 Summary matrix

		FUNCTIONS					
		GUI	MDF SERVICE	USER VALIDATION	TOOLKIT QUERY	RELATIONS MANAGER	TRIPLESTORE
ACTIVITIES	ACTIVITY 1	X	X	X	X		
	ACTIVITY 2	X	X	X		X	X
	ACTIVITY 3	X	X	X	X		X

Table 31. FV Functions DiMDF

4.6 DIMAT MATERIALS MODELER (DIMM)

The DiMAT Materials Modeler (MM) is a toolkit able to predict materials' properties and behaviour 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 product quality during manufacturing on a microscopic scale. Moreover, efficient material behaviour prediction may provide control capabilities in microstructure mechanisms. For example, material damage or failure related behaviours can be predicted in time, and thus, permanent problems can be prevented by applying 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 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 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. All these actions aim to support increasing the production design rate while decreasing the production cost.

4.6.1 Activities

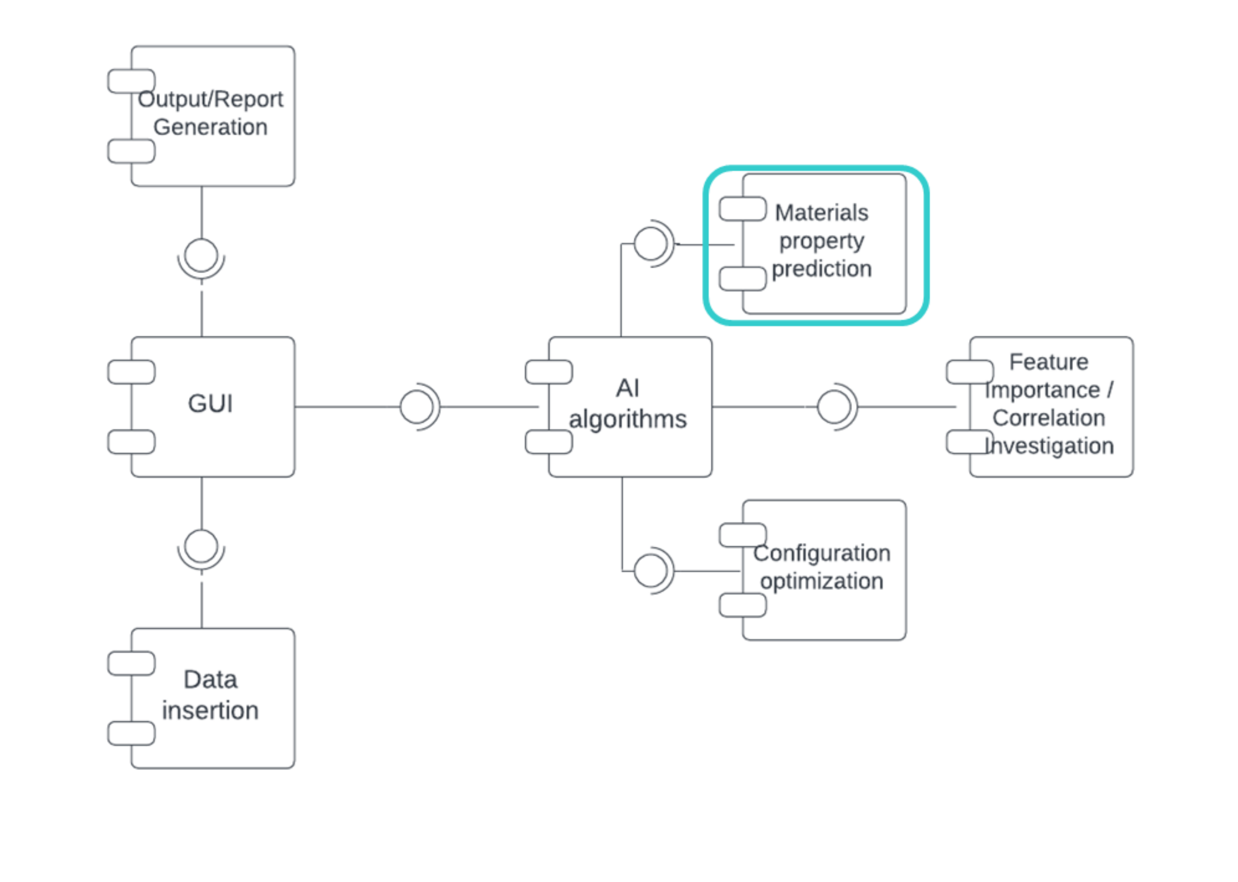
4.6.1.1 Activity 1 – "Predict Materials Properties"

This activity represents the process of predicting material properties 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.

4.6.1.1.1 Activity 1

4.6.1.1.2 Functional map

Functional Viewpoint architecture diagram



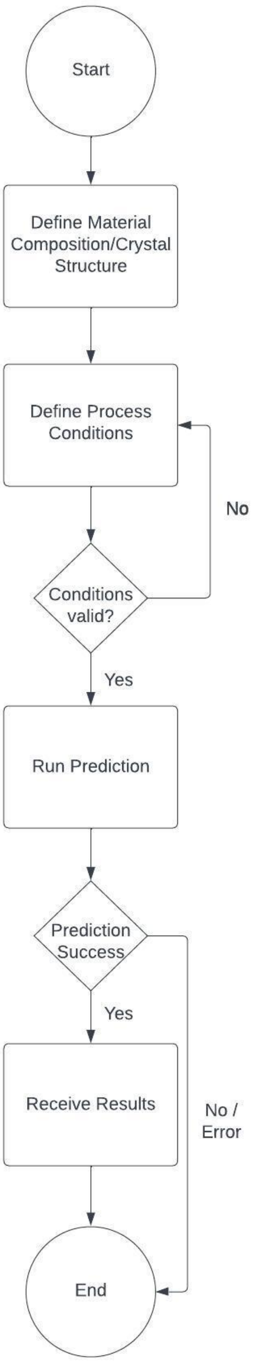
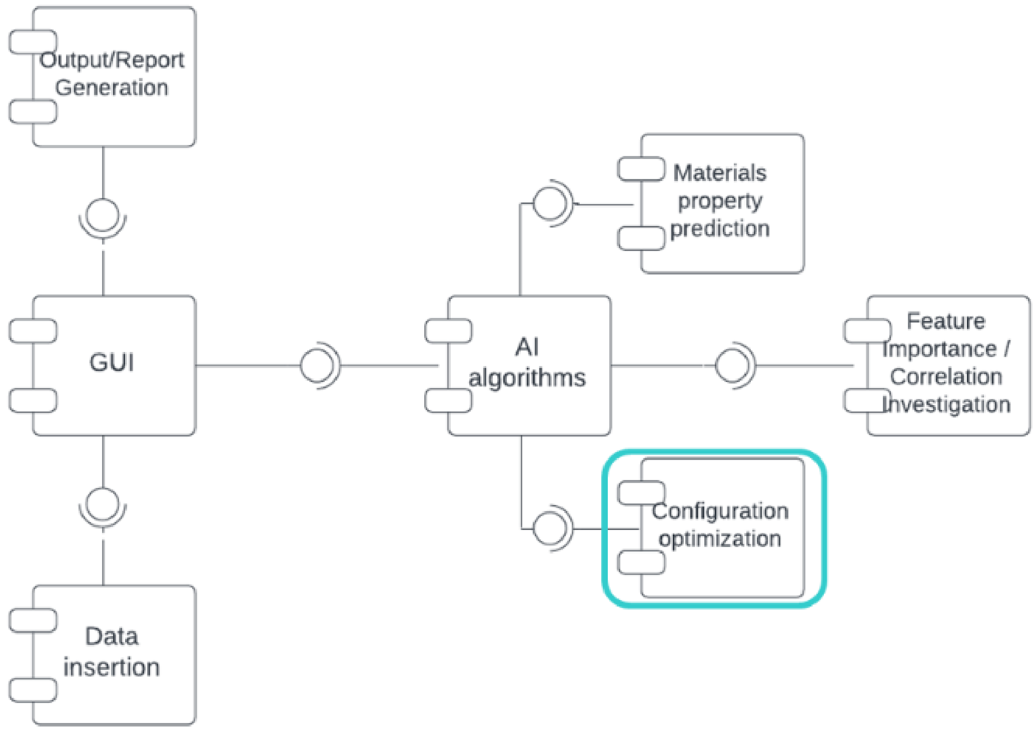
Activity 1 diagram	Explanation
	The Data Analyst develops the AI algorithms based on the data input from Product Engineers. The users interact with the Materials property prediction module to run models based on materials parameters with the objective of predicting target properties.

Table 32. FV Architecture DiMM - Activity 1

4.6.1.2 Activity 2 – “Suggest Optimal Configurations”

This activity 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.

4.6.1.2.1 Activity 2 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
</table>

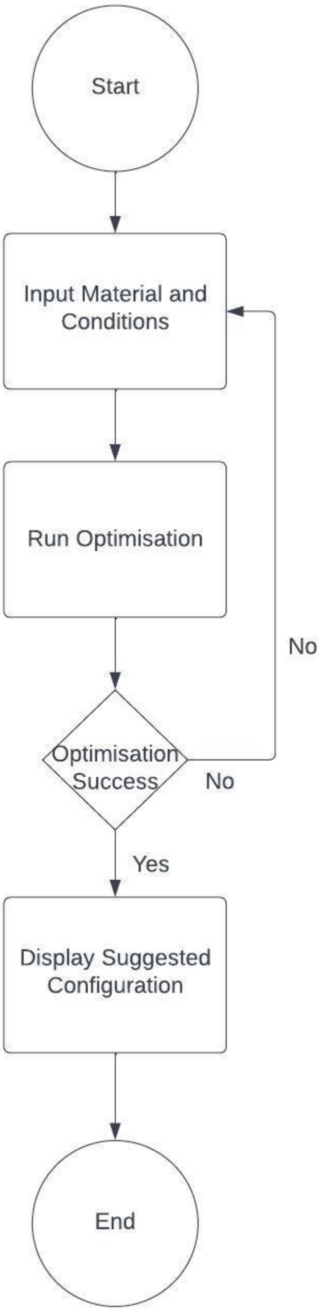
	The Data Analyst develops the AI algorithms based on the data input from Product Engineers. The users interact with the Configuration Optimization module to acquire suggested structure and/or composition based on the data provided.

Table 33. FV Architecture DiMM - Activity 2

4.6.1.3 Activity 3 – "Analyse Causal Inferences"

This activity utilizes AI techniques to analyse and understand complex correlations, data feature importances and causal relationships between different factors, such as material structure, composition, and properties, and their influence on material behaviour.

4.6.1.3.1 Activity 3 – Functional map

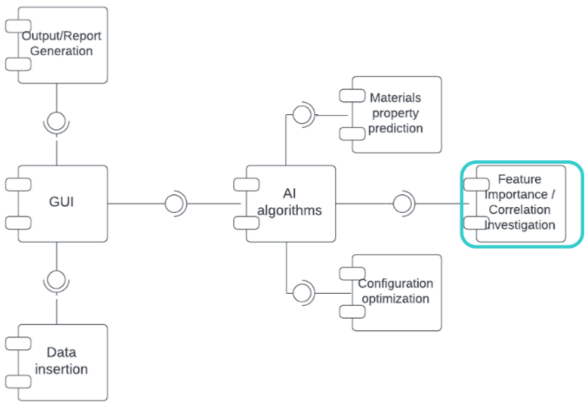
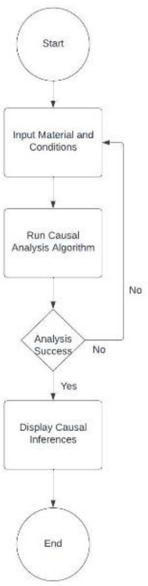
Functional Viewpoint architecture diagram	
	
Activity 3 diagram	**Explanation**
	The Data Analyst develops the AI algorithms based on the data input from Product Engineers. The users interact with the Feature Importance/Correlation Investigation module to obtain causal inferences, feature correlation and feature importances of the input data.

Table 34. FV Architecture DiMM - Activity 3

4.6.1.4 Activity 4 – "Output/Report Generation"

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

4.6.1.4.1 Activity 4 – Functional map

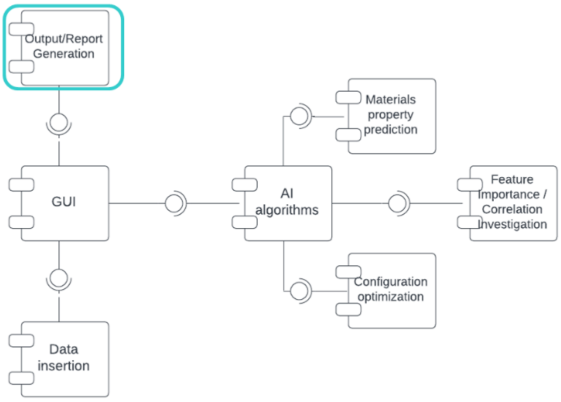
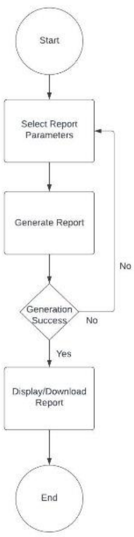
<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 4 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>The users interact with one of the three modules of the previous activities and obtain the results in the form of visual and raw output via the GUI.</td></tr>
</table>

Table 35. FV Architecture DiMM - Activity 4

4.6.1.5 Summary matrix

		GUI	OUTPUT/ REPORT GENERATION	AI ALGORITHMS	MATERIALS PROPERTY PREDICTION	CONFIGURATION OPTIMIZATION	FEATURE IMPORTANCE/CORRELATION INVESTIGATION
ACTIVITIES	ACTIVITY 1	X		X	X		
	ACTIVITY 2	X		X		X	
	ACTIVITY 3	X		X			X
	ACTIVITY 4	X	X				

Table 36. FV Functions DiMM

4.7 DIMAT MATERIALS DESIGNER (DI^{MD})

The DiMAT Material Designer is a tool dedicated to calculating 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.

4.7.1 Roles

A new role has been created since v1 of the document.

- **System administrator:** oversees the management of the toolkit.

4.7.2 Activities

4.7.2.1 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 materials, the geometric description of the

microstructure and the final layup of the composite 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.

4.7.2.1.1 Activity 1 - Triggers

The activity is triggered when the Simulation Engineer selects the option "Define new composite material" in the homepage of the toolkit or from the Composite panel.

4.7.2.1.2 Activity 1 - Mockups

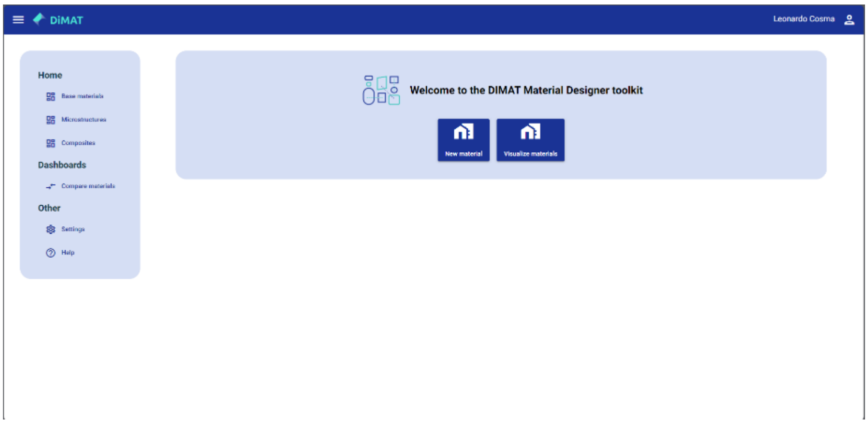
<table>
<tr><th>Function</th><th>Mockup</th></tr>
<tr><td>Home page</td><td>


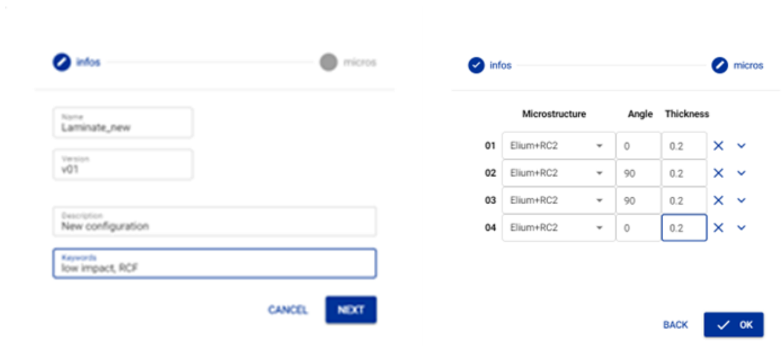
Figure 30. Mockup Home page of Di^{MD}
</td></tr>
<tr><td>Material definition</td><td>


Figure 31. Mockup Material definition of Di^{MD}
</td></tr>
</table>

Table 37. Mockup Home page & Material Definition of Di^{MD}

4.7.2.1.3 Activity 1 – Functional map

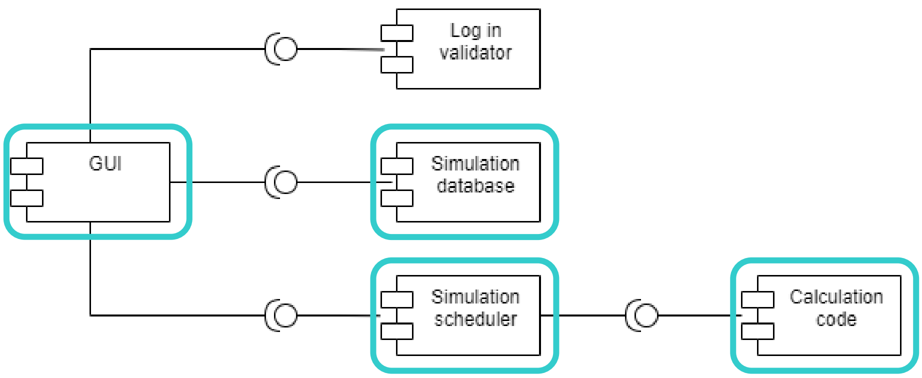
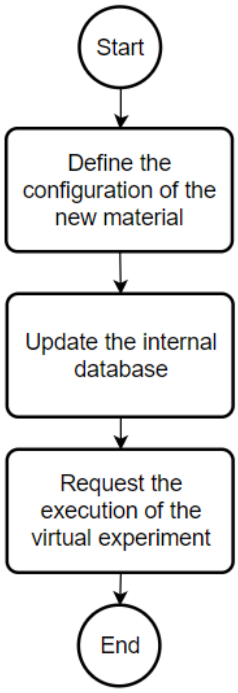
Functional Viewpoint architecture diagram	
	
Activity 1 diagram	**Explanation**
	The activity links with the Simulation database because data are stored when a new calculation is requested. At the same time, the Simulation Scheduler is involved in sending the data to the Calculation Code and ask for the execution of the virtual test to calculate the requested properties.

Table 38. FV Architecture DiMD - Activity 1

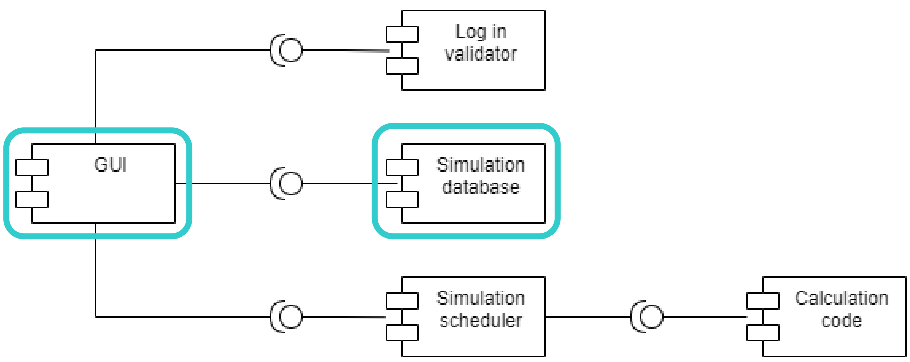
4.7.2.2 Activity 2 – "Review the results"

4.7.2.2.1 Activity 2 - Mockups

Function	Mockup
Results review	 **Figure 32.** Mockup Results review of Di^{MD}

Table 39. Mockup Results review

4.7.2.2.2 Activity 2 - Functional Map

Functional Viewpoint architecture diagram


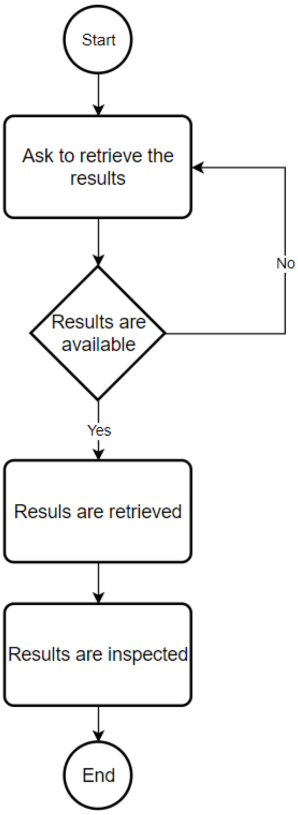
Activity 2 diagram	Explanation
	Results requested by the user are searched in the Simulation Database. If they're not yet available, they are not displayed. If results are available, they are shown in the UI.

Table 40. FV Architecture DiMD - Activity 2

4.7.2.3 Summary matrix

		GUI	LOG IN VALIDATOR	SIMULATION DATABASE	SIMULATION SCHEDULER	CALCULATION CODE
ACTIVITIES	ACTIVITY 1	X		X	X	X
	ACTIVITY 2	X		X		

Table 41. FV functions DiMD

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

The goal of the DiMAT MMS toolkit is to precisely predict the mechanical properties of materials. This toolkit considers various inputs, such as material component fractions, compound compositions, microstructure parameters (e.g., porosity, grain size), geometry details, and mechanical properties of the compounds involved. The toolkit generates outputs that include the material's elastic properties, strength, and rheological properties.

4.8.1 Overview

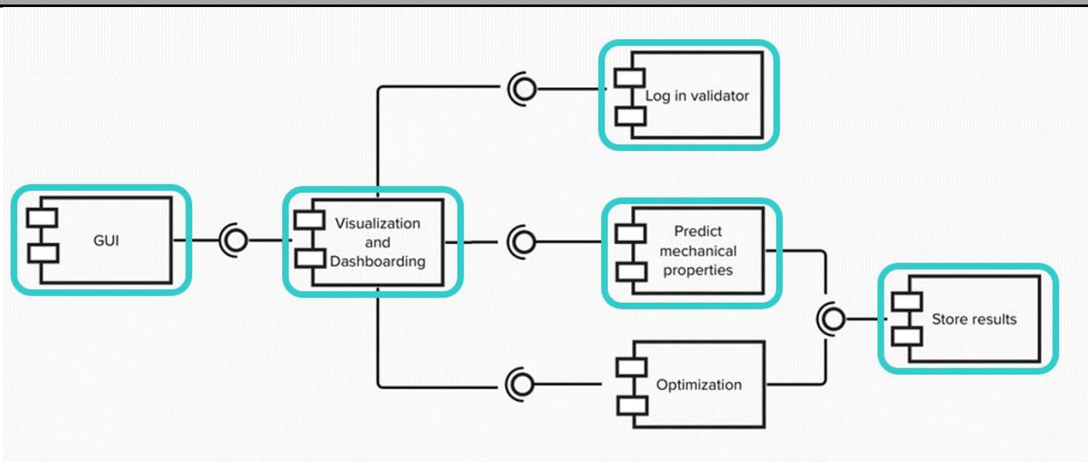
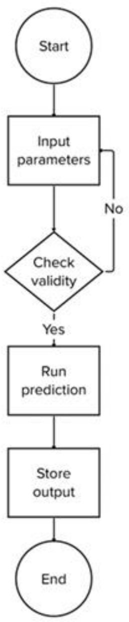
Materials Scientists and Manufacturing Engineers use this toolkit to predict the mechanical properties of materials based on their composition, microstructure, and compound properties. The toolkit enables users to predict and optimise material properties for specific applications.

4.8.2 Roles

- **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 also be used to define the optimization problem, formulating constraints and objective functions.
- **Manufacturing Engineer:** The manufacturing engineer uses the DiMMS toolkit for optimization. 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 supervise the toolkit's operation and define security policies, adding or removing eligible users.

4.8.3 Activities

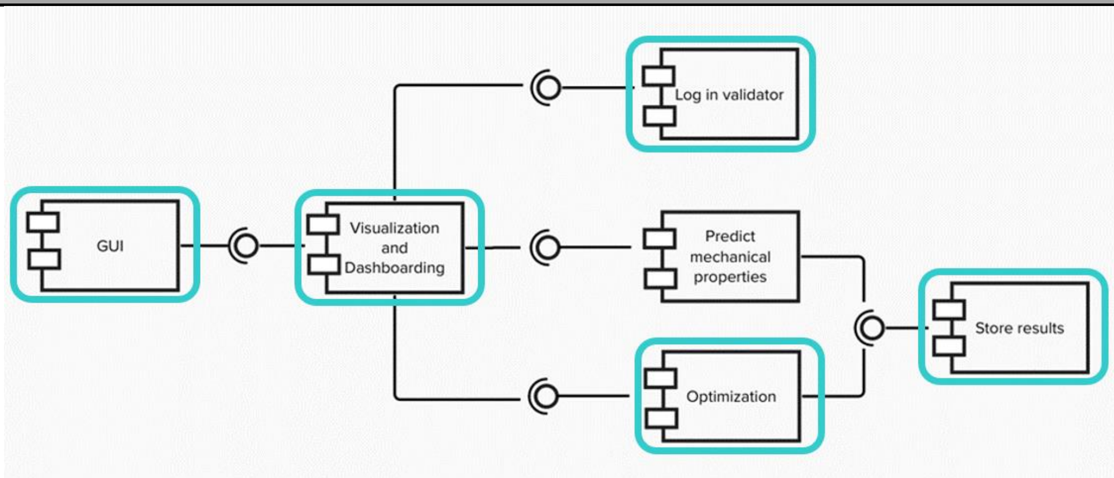
4.8.3.1.1 Activity 1 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 1 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 1 links with function “Predict mechanical properties”. The objective is to determine the mechanical properties associated a specific input.

“GUI” and “Log in validator” and “Visualization and Dashboarding” are needed for a graphical and secure interaction.</td></tr>
</table>

Table 42. FV Architecture DiMMS - Activity 1

4.8.3.1.2 Activity 2 – Functional map

<table>
<tr><th colspan="2">Functional Viewpoint architecture diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 2 links with function “Optimization”. The objective is to determine optimum composition or configuration for a given mechanical property, or range of mechanical properties.

“GUI” and “Log in validator” and “Visualization and Dashboarding” are needed for a graphical and secure interaction.</td></tr>
</table>

Table 43. FV Architecture DiMMS - Activity 2

4.8.3.2 Summary matrix

		FUNCTIONS					
		GUI	VISUALIZATION AND DAHSBOARDING	LOG IN VALIDATOR	PREDICT PROPERTIES	OPTIMIZATION	STORE RESULTS
ACTIVITIES	ACTIVITY 1	X	X	X	X		X
	ACTIVITY 2	X	X	X		X	X

Table 44. FV Functions DiMMS

4.9 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 behaviour 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' behaviour without going through the production process line to perform such tests. In this way, material transformation processes can be optimized, reducing costs and production times.

4.9.1 Overview

Manufacturing Engineers, Process Scientists, and Process Engineers use this toolkit to predict material behaviour under production conditions and optimize materials transformation processes. Also, Materials Scientists use the toolkit to anticipate defects during materials processing and use the output to select the correct material and parameters.

4.9.2 Roles

- **System Administrator:** The system administrator will manage the toolkit system, create new users assign credentials and keep up to date the toolkit database.
- **Manufacturing/Process Engineer:** manufacturing engineers can use the toolkit to optimize manufacturing processes, such as 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.

4.9.3 Activities

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.

Co-funded by
the European Union

4.9.3.1.1 Activity 1 – Functional Map

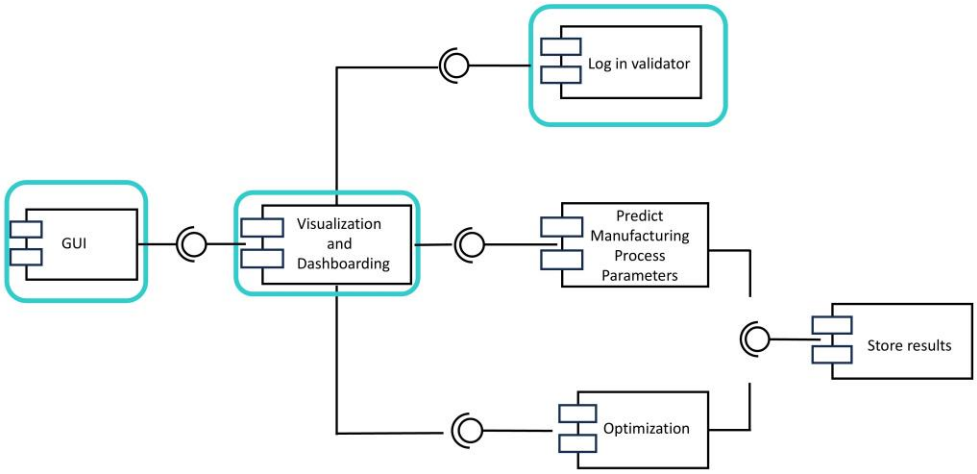
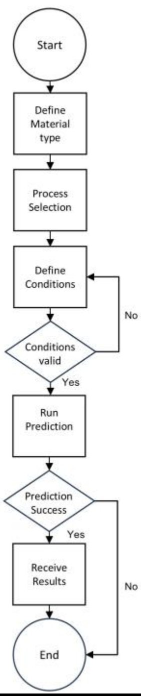
Functional Viewpoint architecture diagram	
	
Activity 1 diagram	**Explanation**
	Activity 1 links with the function "Visualization and Dashboarding" because the first step is to select the material process transformation, which involves geometry and boundary conditions, including the possibility of modifying these parameters. "GUI" and "Log in validator" are needed for a basic secure interaction.

Table 45. FV Architecture DiMPS - Activity 1

4.9.3.1.2 Activity 2 – Functional Map

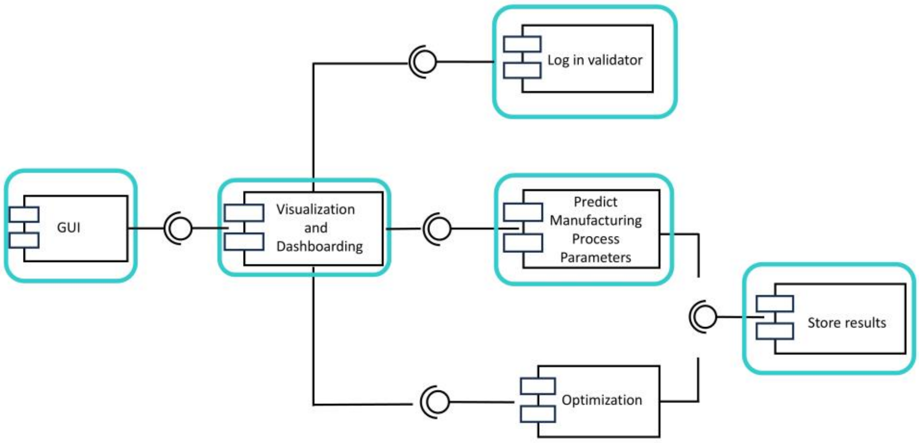
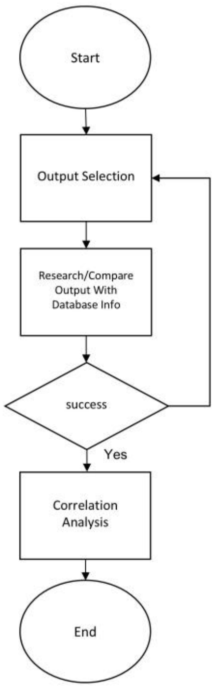
Functional Viewpoint Architecture Diagram	
	
Activity 2 diagram	**Explanation**
	Activity 2 links with functions "Predict Manufacturing Process Parameters" and "Store Results" because the objective is obtaining accurate data and testing new parameters to avoid interference in the production line. The generated new data will feed the database toolkit. At the same time this activity requires an interface where users can view the running simulation by the "Visualization and Dashboarding" function. "GUI" and "Log in validator" are needed for a basic secure interaction.

Table 46. FV Architecture DiMPS - Activity 2

4.9.3.1.3 Activity 3 – Functional Map

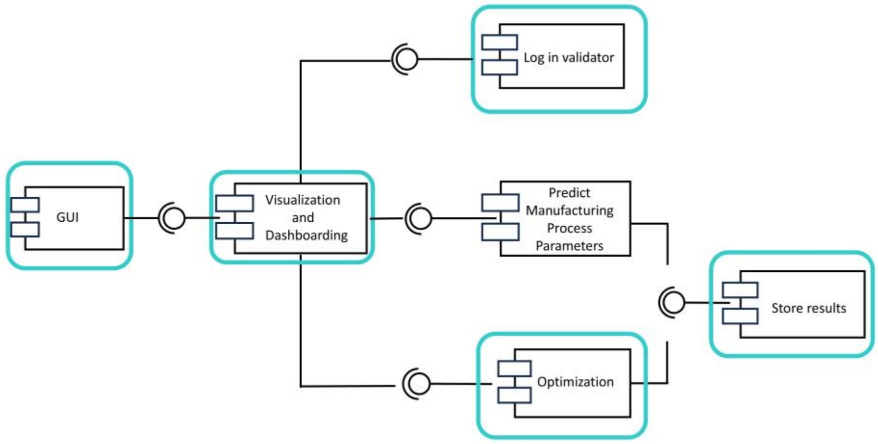
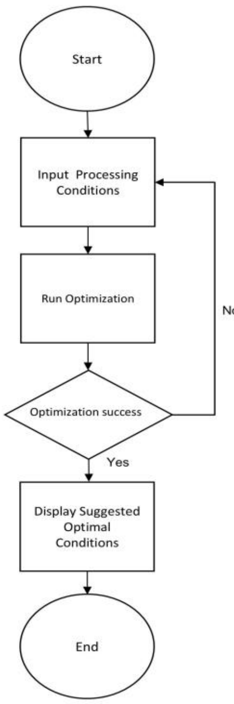
Functional Viewpoint Architecture Diagram	
	
Activity 3 diagram	**Explanation**
	Activity 3 links with functions “Optimization” and “Store Results” to use store data generated of predictions to accurately simulate the process and obtain specific input to determine accurate output. At the same time this activity requires an interface where users can view the running simulation by the “Visualization and Dashboarding” function. “GUI” and “Log in validator” are needed for a basic secure interaction.

Table 47. FV Architecture DiMPS - Activity 3

4.9.3.2 Summary matrix

		FUNCTIONS					
		GUI	VISUALIZATION AND DAHSBOARDING	LOG IN VALIDATOR	PREDICT PROCESS	OPTIMIZATION	STORE RESULTS
ACTIVITIES	ACTIVITY 1	X	X	X			
	ACTIVITY 2	X	X	X	X		X
	ACTIVITY 3	X	X	X		X	X

Table 48. FV Functions DiMPS

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

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 and 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.

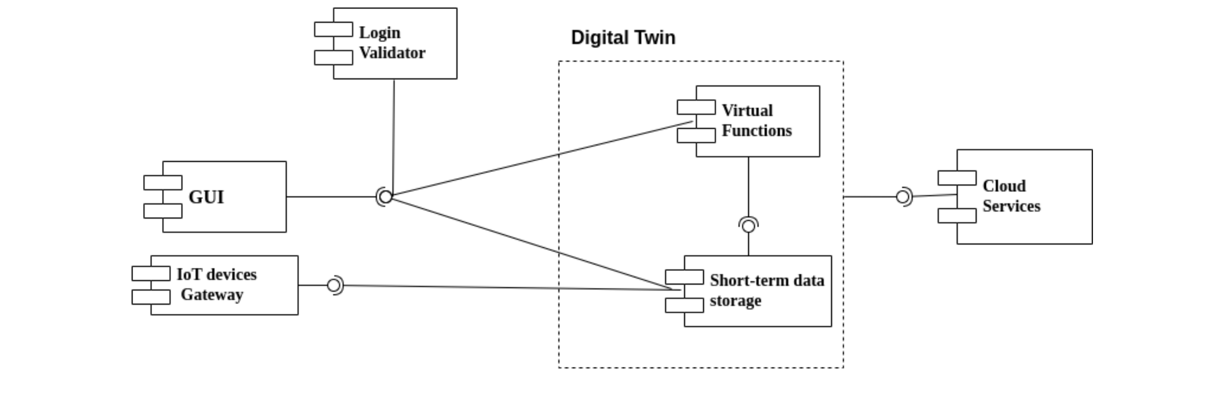
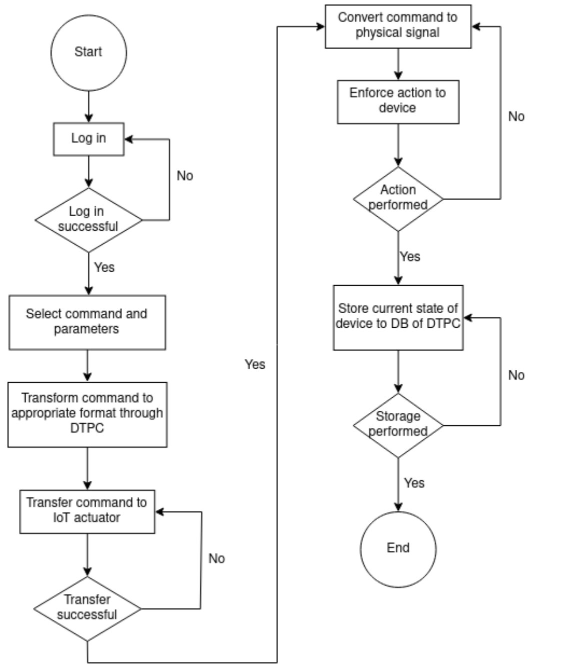
4.10.1 Activities

4.10.1.1 Activity 1 – Send command to device

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

4.10.1.1.1 Activity 1 – Functional map

Functional Viewpoint Architecture Diagram

<table>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 1 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 1 links with the function “Login validator” to provide secure access to the toolkit’s services.

The main part of the activity is the command send to the device. This is handled with the “Virtual Functions” function through an appropriately developed virtual function.

The activity also links with the function “short-term data storage” since the current state needs to be saved in the time-series database.</td></tr>
</table>

Table 49. FV Architecture DiDTPC - Activity 1

4.10.1.2 Activity 2 – View Digital Twin status

An essential activity of the DTPC toolkit. This activity refers to acquiring the corresponding physical twin's current state (measurements). This state can be employed in numerous operations (health checks, time-series analysis, etc.).

4.10.1.2.1 Activity 2 – Functional Maps

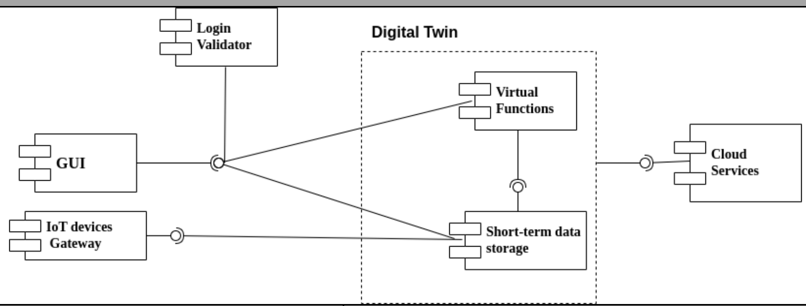
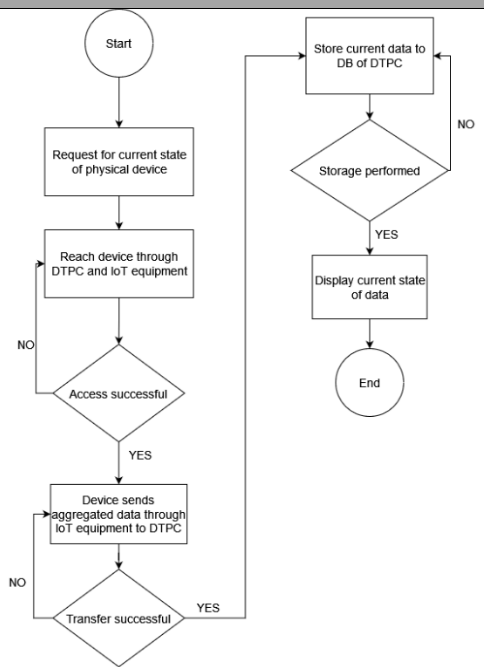
<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 2 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 2 links with the “Login validator” function to provide secure access to the toolkit’s services. The main part of the activity is the command sent to the device to retrieve the status of the equipment. This is handled with the “Virtual Functions” function through an appropriately designed function. Moreover, the IoT devices gateway can be used as an aggregation point in the case of a complex Digital Twin that needs to poll data from various data sources to give them as input to complex virtual functions. Moreover, the gateway can act as a point where some initial functions are applied to data (e.g., averaging, removal of corrupt data, etc.).
It also links with the function “short-term data storage” since the current state needs to be saved in the time-series database.</td></tr>
</table>

Table 50. FV Architecture DiDTPC - Activity 2

4.10.1.3 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.

4.10.1.3.1 Activity 3 – Functional Map

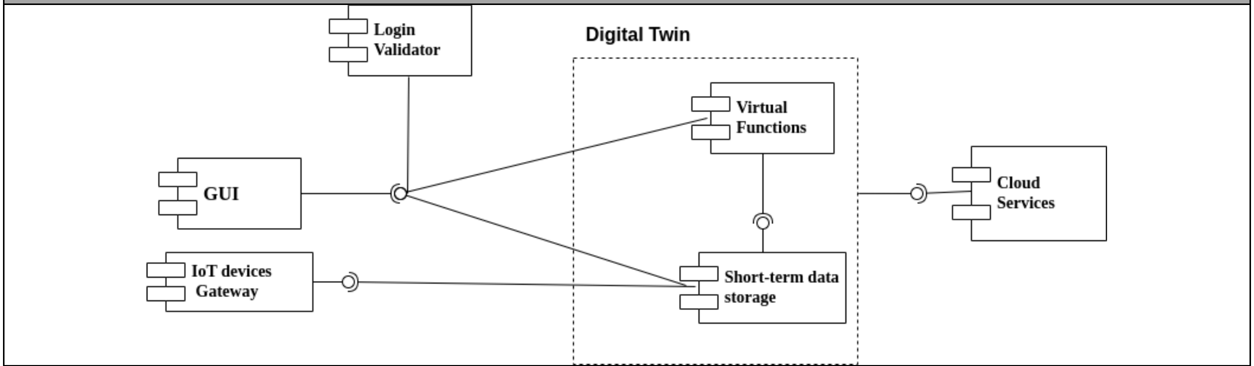
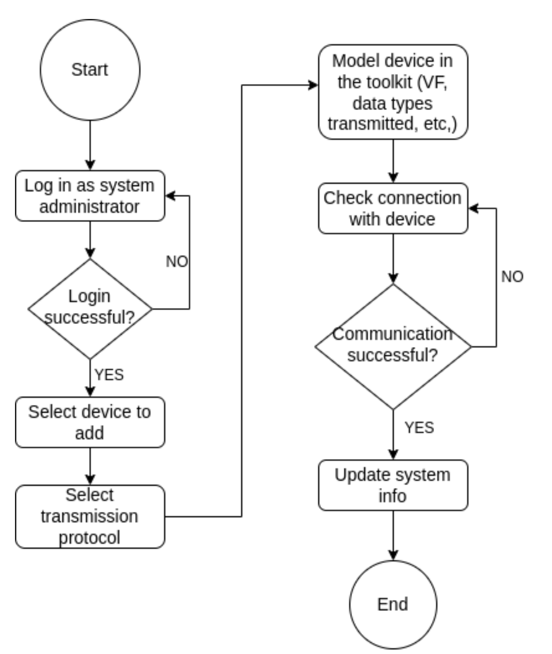
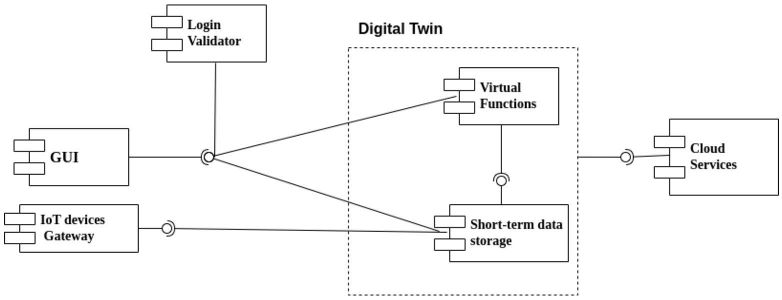
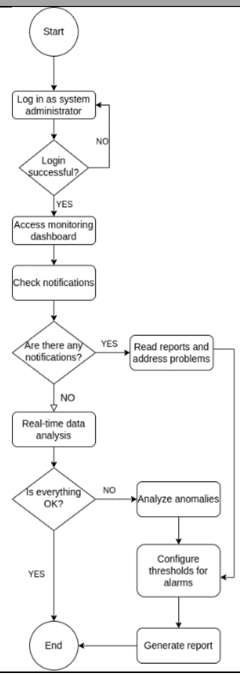
Functional Viewpoint Architecture Diagram	
	
Activity 3 diagram	**Explanation**
	Activity 3 links with the function "Login validator" to provide secure access to the toolkit's services. Only the administrator has rights to this activity. It also links with the "short-term data storage" function since the new device's information is stored in appropriately designed files specified in the Nephele VO software stack. Moreover, information about the resources measured from this device is required from the local time-series database employed.

Table 51. FV Architecture DiDTPC - Activity 3

4.10.1.4 Activity 4 – Monitor DTPC Operation

The system's administrator is responsible for the smooth operation of the toolkit and periodically performs health checks. The DTPC toolkit handles this activity by providing visualizations generated by Virtual functions and displayed in the GUI.

4.10.1.4.1 Activity 4 – Functional Maps

<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 4 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 4 links with the function "Login validator" to provide secure access to the toolkit's services. It is an activity that can be performed only by the toolkit administrator.

The GUI is necessary for providing access to visualizations/dashboards. Short-term storage is also crucial for handling data stored in the time-series database.

The "Virtual Functions" are necessary since real-time data analysis is developed as a virtual function responsible for deciding on the correct operation of the digital twin.</td></tr>
</table>

Table 52. FV Architecture DiDTPC - Activity 4

4.10.1.5 Activity 5 – Analyse time-series data

Time-series analysis is essential for the Digital Twin since a lot of the data that the DTPC toolkit handles are of this type (environmental measurements, measurements polled from machinery at fixed intervals, etc.)

4.10.1.5.1 Activity 5 – Functional Map

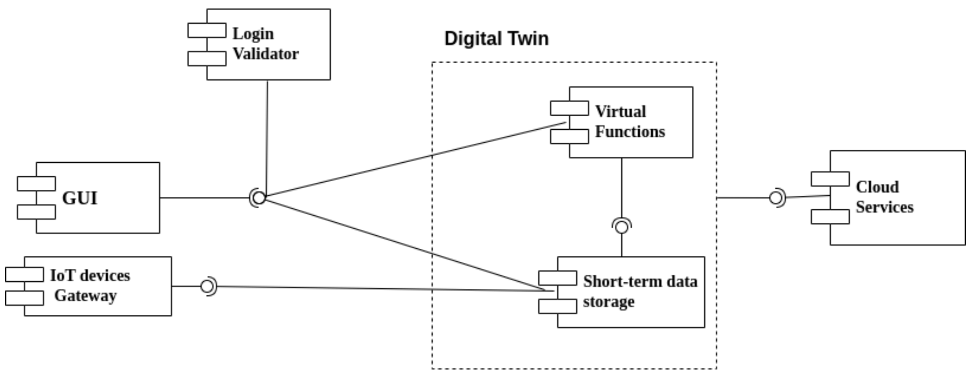
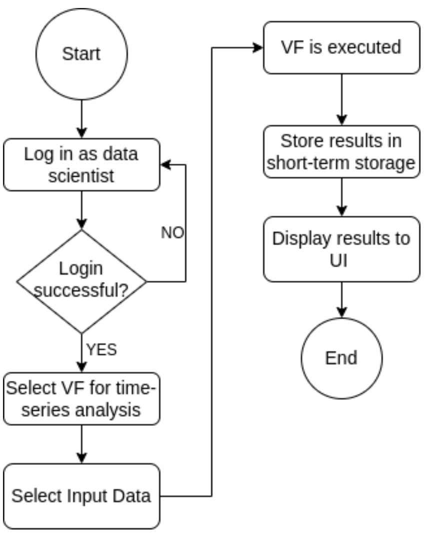
<table>
<tr><th colspan="2">Functional Viewpoint Architecture Diagram</th></tr>
<tr><td colspan="2">
</td></tr>
<tr><th>Activity 5 diagram</th><th>Explanation</th></tr>
<tr><td>
</td><td>Activity 5 links with the function “Login validator” to provide secure access to the toolkit’s services. It is an activity that can be performed only by the “data scientist” role of the toolkit.
The “Virtual Functions” are necessary since the user develops a specific function as one such functionality aimed for time-series analysis. If the VF is too heavy to be locally deployed, then the “Cloud services” function can also be used.
Short-term data storage is necessary for storing the results of the Virtual function, and finally, the GUI is responsible for demonstrating the results in figures/dashboards/tables, etc.</td></tr>
</table>

Table 53. FV Architecture DiDTPC - Activity 5

Summary Matrix

		FUNCTIONS					
		GUI	LOG IN VALIDATOR	IOT DEVICES GATEWAY	VIRTUAL FUNCTIONS	SHORT-TERM DATA STORAGE	CLOUD SERVICES
ACTIVITIES	ACTIVITY 1	X	X		X	X	
	ACTIVITY 2	X	X	X	X	X	
	ACTIVITY 3	X	X			X	
	ACTIVITY 4	X	X		X		X
	ACTIVITY 5	X	X		X	X	

Table 54. FV Functions DiDTPC

5 FUNCTIONAL VIEWPOINT

This task will break down the DiMAT Framework into its distinct domains: Control, Operations, Information, Application, and Business. We will identify the flows of data, decisions, and commands/requests circulating among these domains. Following this breakdown, we will detail the mechanisms of control, coordination, and orchestration applied within each domain, as well as the typical operations associated with them.

5.1 FUNCTIONAL VIEWPOINT APPROACH FOR DIMAT

5.1.1 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 [3]:

- Functional components along with their interfaces (published operations).
- Interactions between external elements and the system and between the functional components.

5.1.2 Functional Viewpoint Methodology

The methodology to complete this part of the deliverable is as follows: Firstly, a general architecture diagram will be created to identify the functional components included in the system, along with a brief description of each of them. Secondly, the roles involved in the different interactions will be analysed. Then, the functional components will be examined through a mapping process. Finally, this document includes in addition a new section, Functional components*, functional structure and requirements interactions* to complete the toolkit's architecture, and several diagrams will be included: one for the functional components, another for the functional structure, and a schema linking the requirements associated with the functional components.

In this second release of D3.3, the toolkits can be updated if the toolkit requires it. if there are no changes from the previous release, this section will not be included in the viewpoint and will remain the same as in the first release submitted in M9.

5.1.3 General Architecture

The general architecture of the toolkit is presented here with a Diagram of components, visually representing the main functions of the toolkit as they relate to each other. In the example diagram below, 6 different functions are identified and linked.

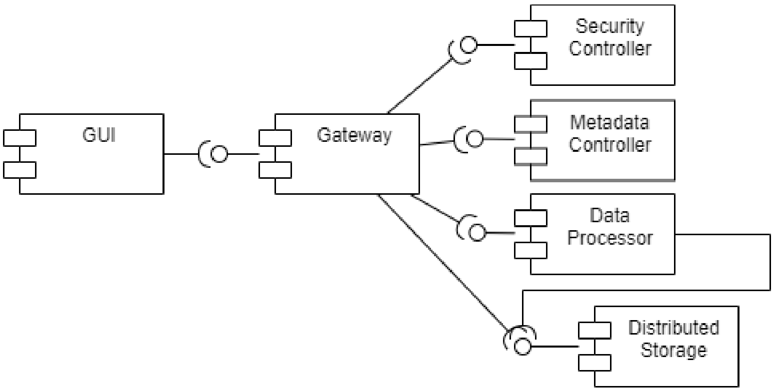


Figure 33. General Architecture example

The General Architecture was defined in the first release of **D3.3 VIEWPOINTS**. In this second release of D3.3, toolkits may or may not be updated, in which case the General Architecture, also known as the Diagram of components section, remains the same as in the first release (M9).

5.1.4 Functional components

This section extracts and defines each functional component from the General Architecture section. It consists of a template that provides information about the name of the functional component, a brief description of it, followed by the operations, inputs and outputs of the interface. The table in the example below describes how the information about a functional component should be organized.

Name:		
Description:		
INTERFACE		
Operation	***Inputs***	***Outputs***

Table 55. General Table Components Example

This table was defined for each functional component in each toolkit in the first release of **D3.3 VIEWPOINTS**. In this second release of D3.3, tables may be updated. However, if this section remains the same as in the first release of M9, it won't be included in this document.

5.1.5 Interactions

This section should contain 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. The roles identified in the Roles section of the Usage Viewpoint are used as a basis for explaining the Interactions section in the Functional Viewpoint. Similarly, each operation defined in each component should be added to its corresponding Interface - Operation cell corresponding of the previous table.

In this second release of D3.3, toolkits may be updated. However, if the Interactions section remains the same as in the first release of M9, it won't be included in this document.

5.1.6 Mapping of functional components to IIRA functional domain

This section focuses on assigning each functional component identified in the previous sections to the appropriate functional domain and a subdomain group.

In this second release of D3.3, toolkits may be updated. However, if the Mapping of functional components section remains the same as in the first release of M9, it won't be included in this document.

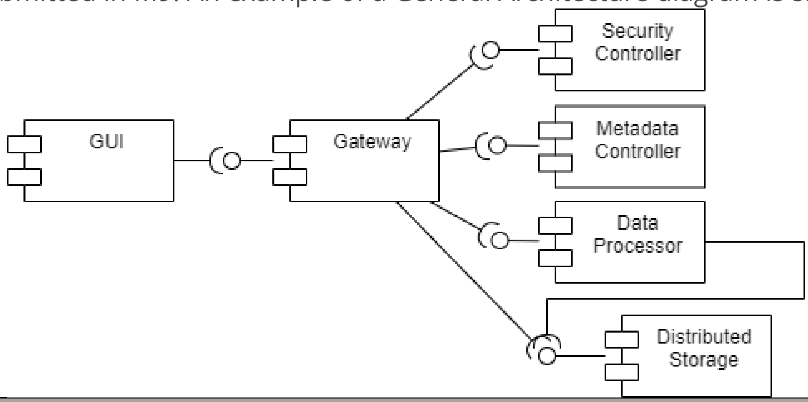
5.1.7 Functional components, functional structure and requirements interactions

This section presents a table with three different types of diagrams. The first one aims to visually identify the selected functional component of the General Architecture. The next one determines the system functions associated with the functional component already identified in the previous diagram. The last one concludes which requirements are necessary and are associated with the system functions selected before.

5.1.8 [Functional component name]

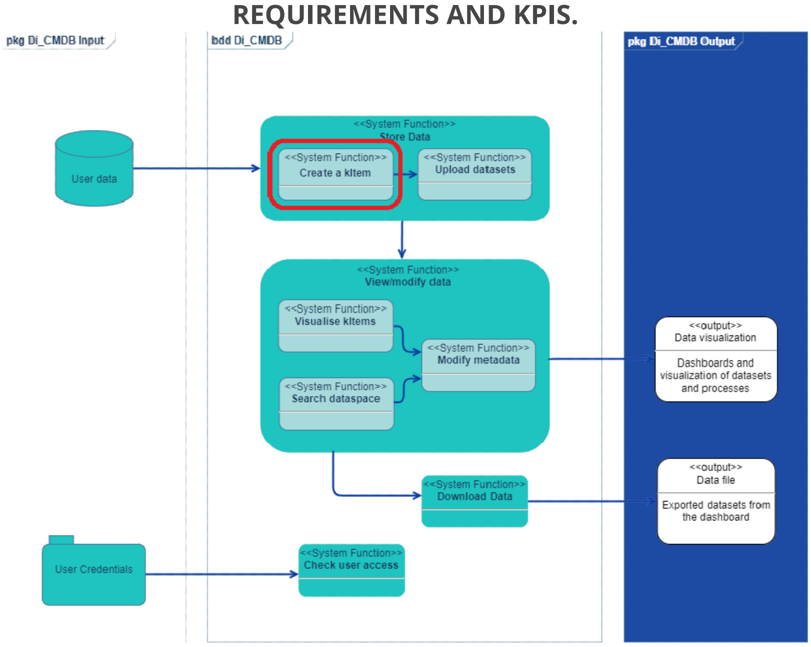
Functional component diagram

This section must include the General Architecture diagram of the toolkit included in the D3.3. VIEWPOINTS, submitted in M9. An example of a General Architecture diagram is shown below.



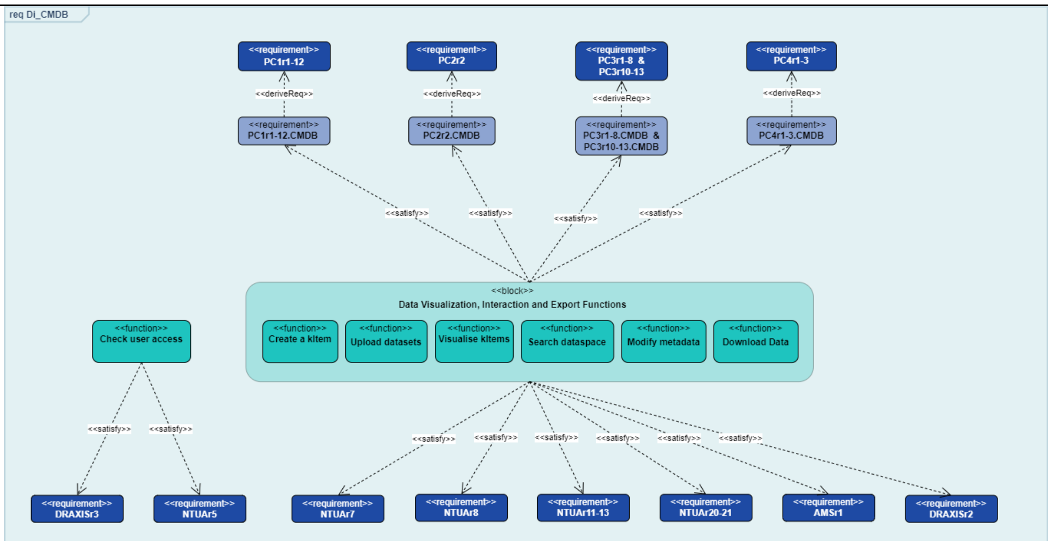
Functional structure diagram

This cell should contain the functional structure diagram defined in **D2.3 USE CASES SCENARIOS REQUIREMENTS AND KPIS.**



Requirements linked to the functional component

This cell should contain the functional structure diagram specified in **D2.3 USE CASES SCENARIOS REQUIREMENTS AND KPIS.**



Describe the connection between the functional component and the requirements

This part should describe the interactions, connections and relationships between the functional component and each requirement.

- PC1r1.CMDB: ###
- PC1r1.KAF: ###
- PC1r1.MECLCA: ###

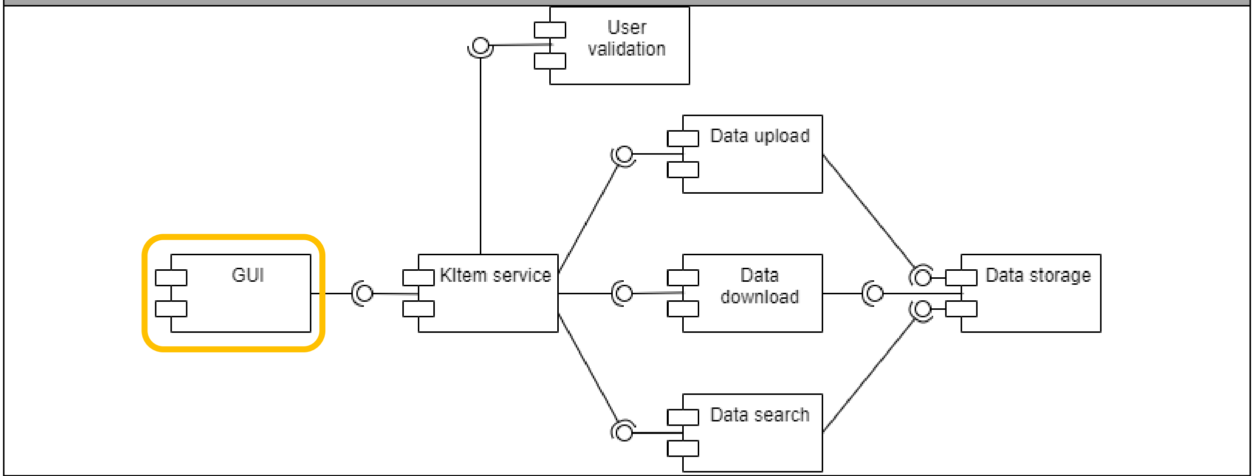
Table 56. Functional component diagram

Co-funded by the European Union

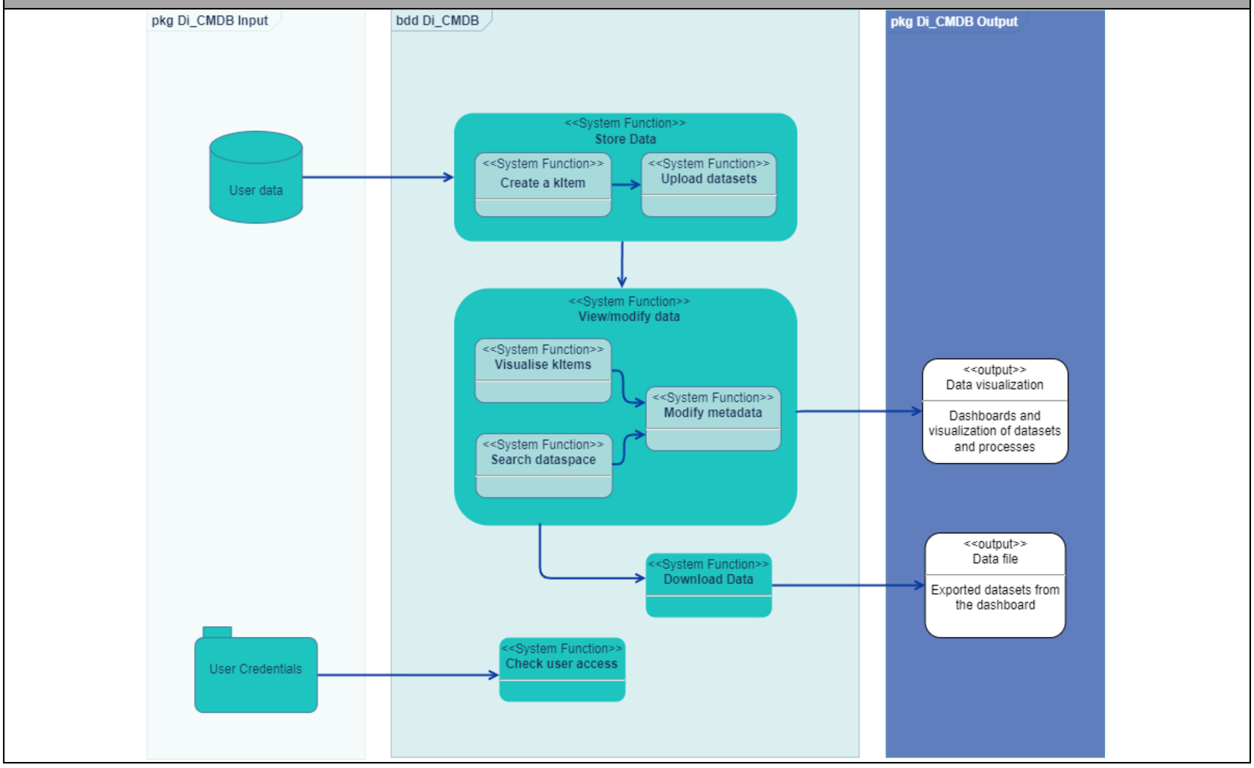
5.2 DIMAT CLOUD MATERIALS DATABASE (DI^{CMDB})

5.2.1 GUI

Functional component diagram



Functional structure diagram



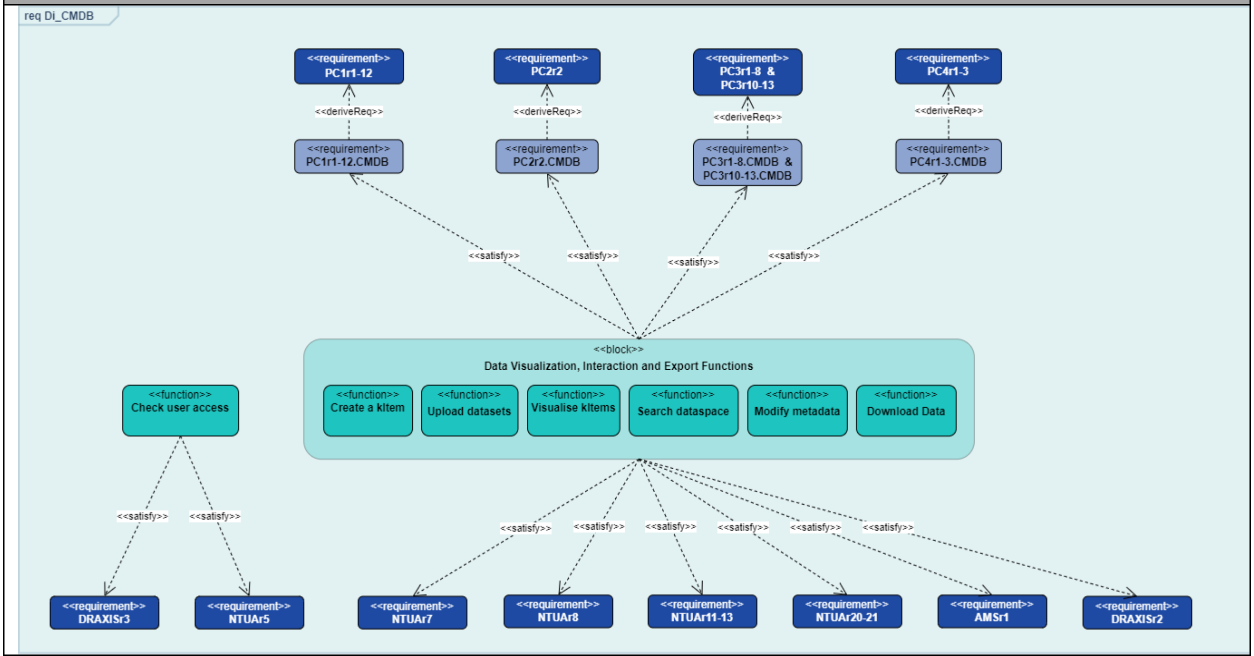
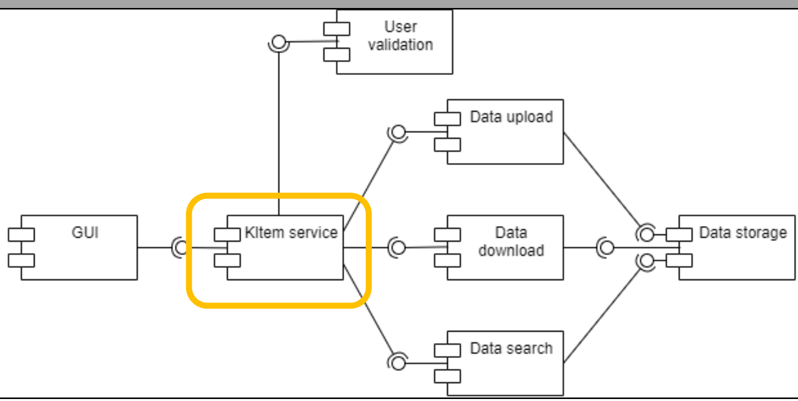
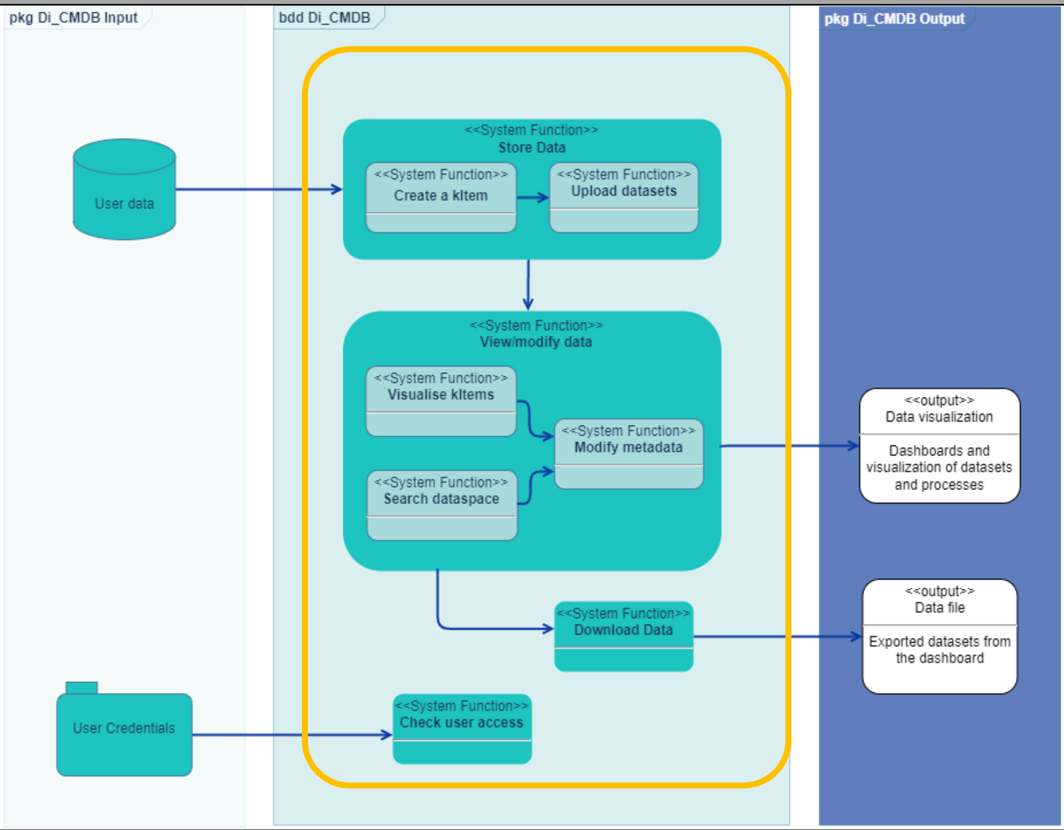
Requirements linked to the functional component

Describe the connection between the functional component and the requirements
• The GUI represents the standard way of interaction with the toolkit and as such, is needed to satisfy all the requirements.

Table 57. GUI functional component diagram, structure and requirements

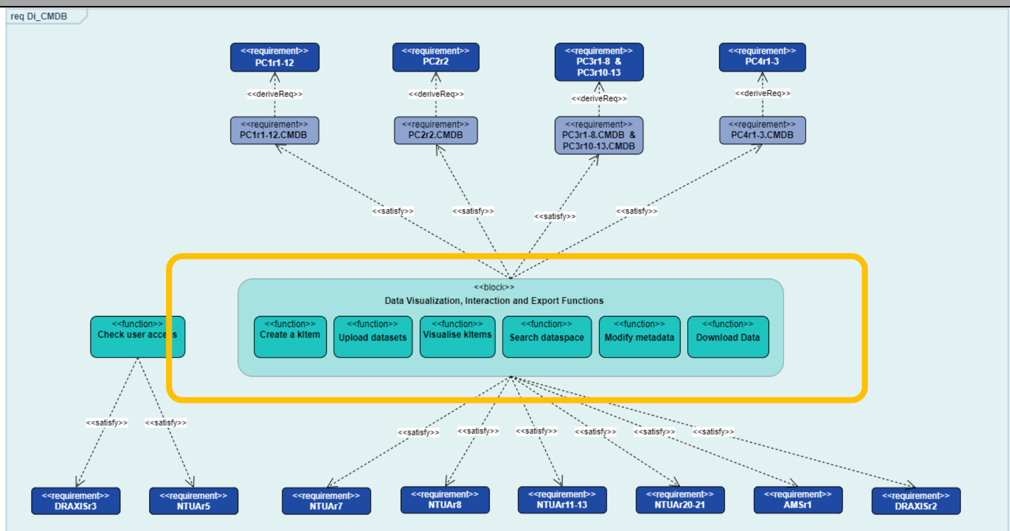
5.2.2 KItem service

Functional component diagram


Functional structure diagram



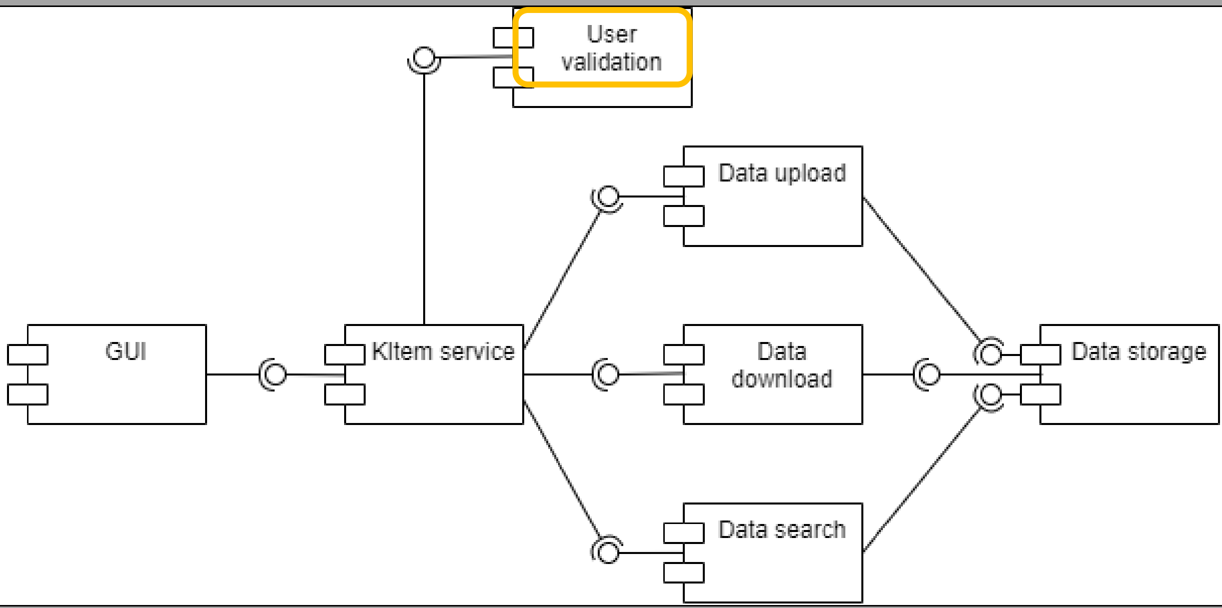
Requirements linked to the functional component

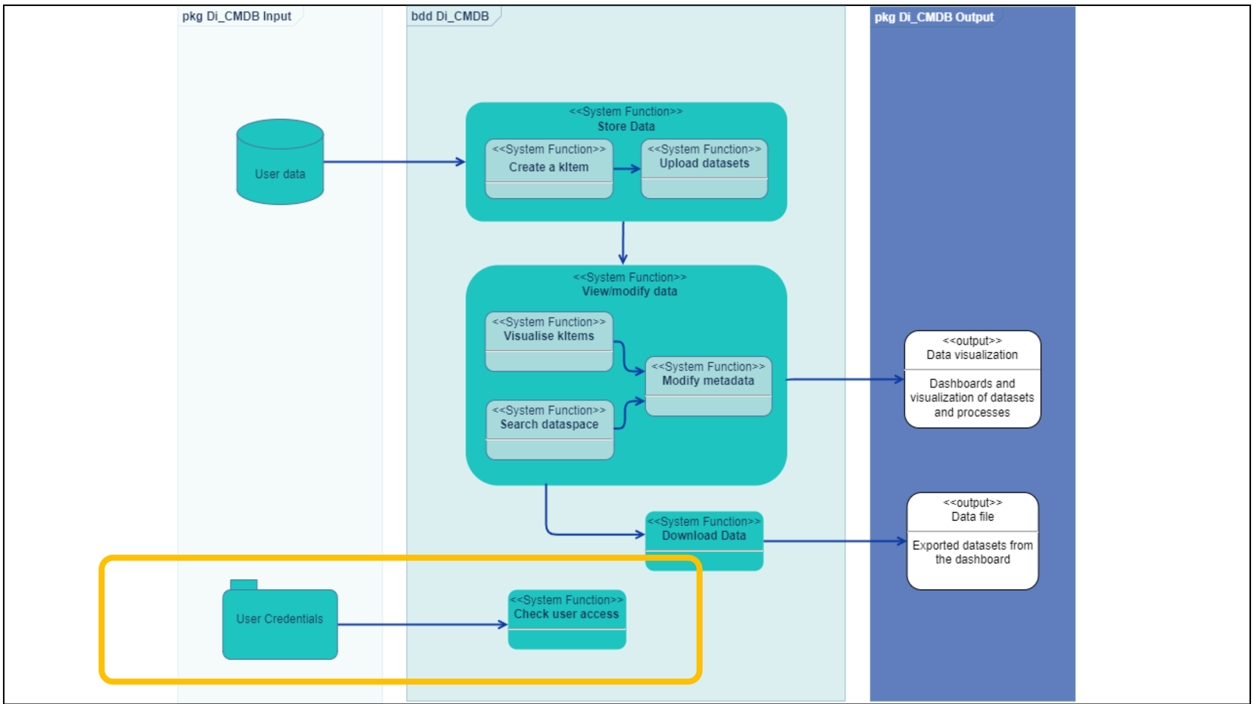


Describe the connection between the functional component and the requirements
The KItem service is the backbone of the CMDB and tackles all the requirements related to data storage, management and exploration: - PC1r1-12 - PC2r2 - PC2r1-8 & PC3r10-13 - PC4r1-3 - NTUAr7 - NTUAr8 - NTUAr11-13 - NTUAr20-21 - AMSr1 - DRAXISr2

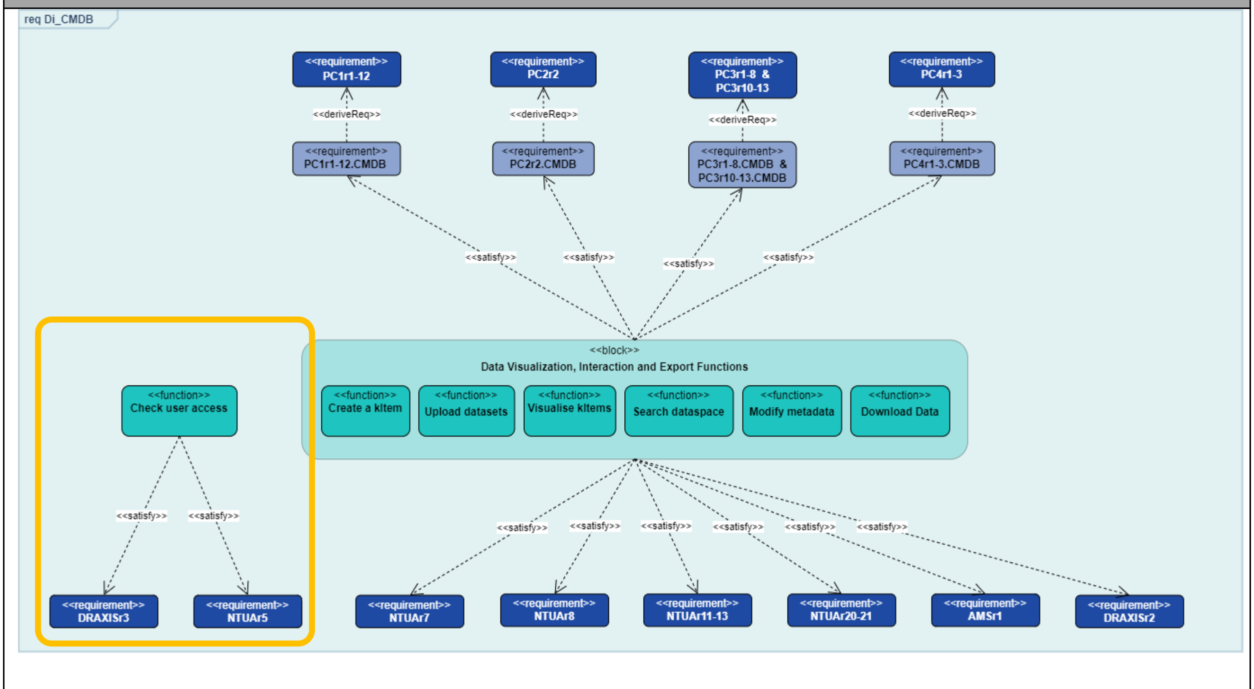
Table 58. Kitem service functional component diagram, structure and requirements

5.2.3 User validation

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



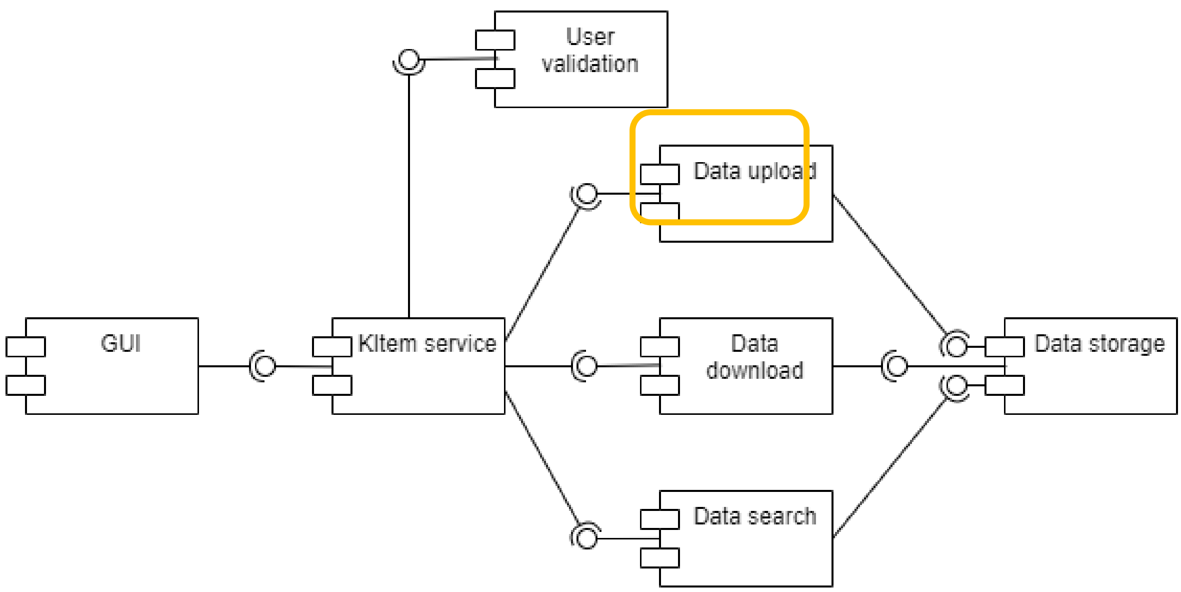
Describe the connection between the functional component and the requirements

This component takes care of the user authentication and authorization. As such, it will satisfy the following security requirements: - DRAXISr3 - NTUAr5 - NTUAr8

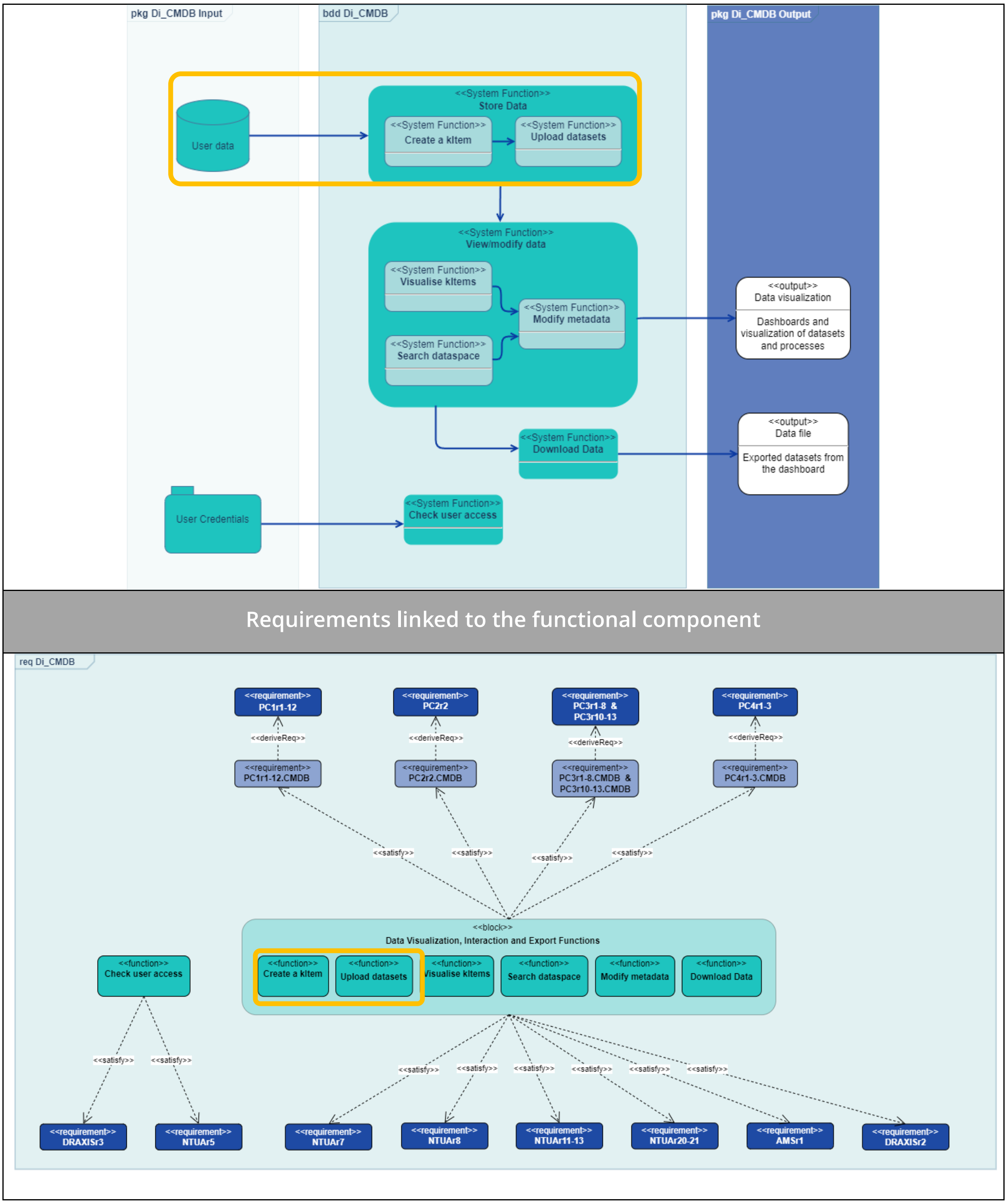
Table 59. User validation functional component diagram, structure and requirements

5.2.3.1 Data upload

Functional component diagram



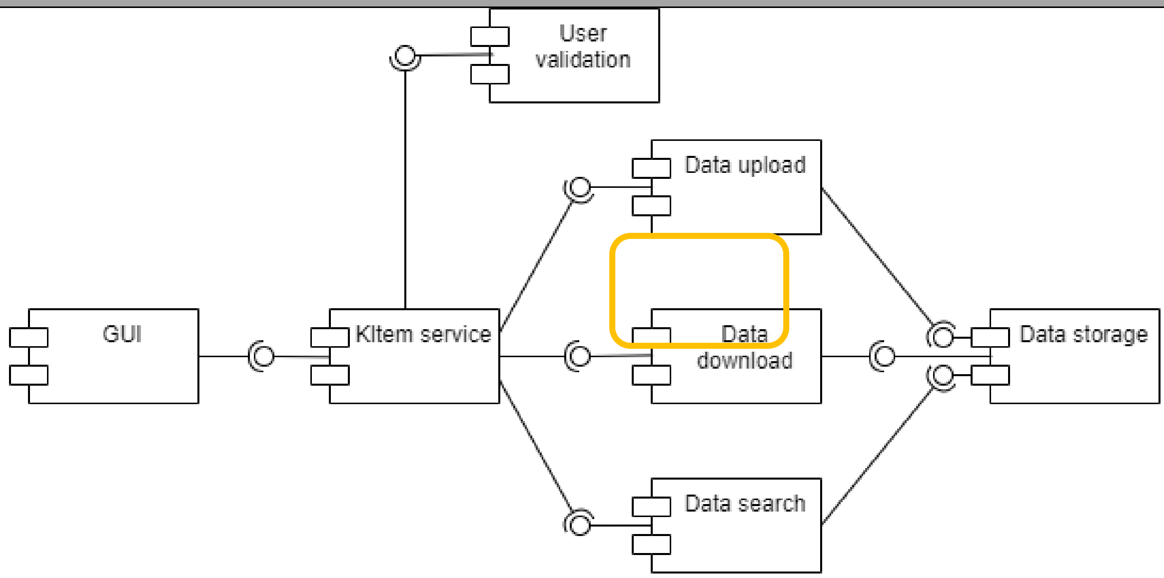
Functional structure diagram

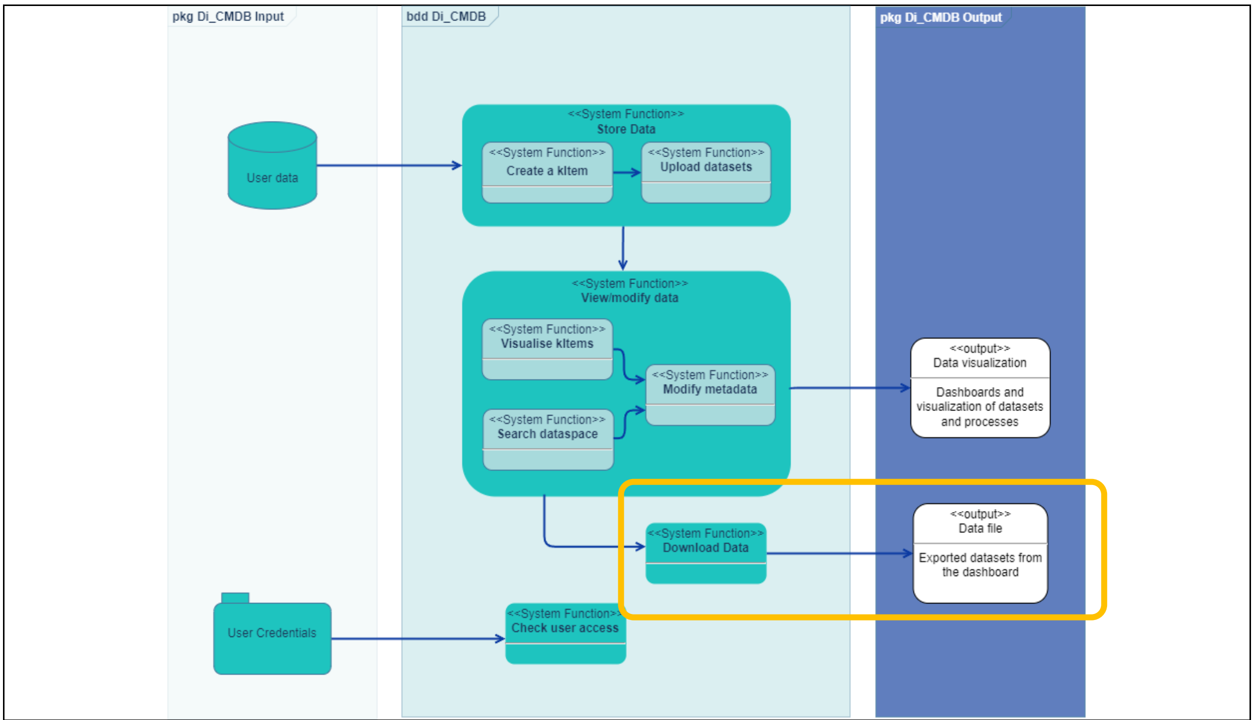

pkg Di_CMDB Input
bdd Di_CMDB
pkg Di_CMDB Output
User data
<<System Function>>
Store Data
<<System Function>>
Create a kItem
<<System Function>>
Upload datasets
<<System Function>>
View/modify data
<<System Function>>
Visualise kItems
<<System Function>>
Modify metadata
<<System Function>>
Search dataspace
<<output>>
Data visualization
Dashboards and visualization of datasets and processes
<<System Function>>
Download Data
<<output>>
Data file
Exported datasets from the dashboard
User Credentials
<<System Function>>
Check user access
Requirements linked to the functional component
req Di_CMDB
<<requirement>>
PC1r1-12
<<requirement>>
PC2r2
<<requirement>>
PC3r1-8 &
PC3r10-13
<<requirement>>
PC4r1-3
<<deriveReq>>
<<requirement>>
PC1r1-12.CMDB
<<requirement>>
PC2r2.CMDB
<<requirement>>
PC3r1-8.CMDB &
PC3r10-13.CMDB
<<requirement>>
PC4r1-3.CMDB
<<satisfy>>
<<block>>
Data Visualization, Interaction and Export Functions
<<function>>
Check user access
<<function>>
Create a kItem
<<function>>
Upload datasets
<<function>>
Visualise kItems
<<function>>
Search dataspace
<<function>>
Modify metadata
<<function>>
Download Data
<<requirement>>
DRAXISr3
<<requirement>>
NTUAr5
<<requirement>>
NTUAr7
<<requirement>>
NTUAr8
<<requirement>>
NTUAr11-13
<<requirement>>
NTUAr20-21
<<requirement>>
AMSr1
<<requirement>>
DRAXISr2

Describe the connection between the functional component and the requirements
The data upload component is related to the data integration tasks and will satisfy the following requirements: - PC1r1-12 - NTUAr12 - AMSr1 - DRAXISr2

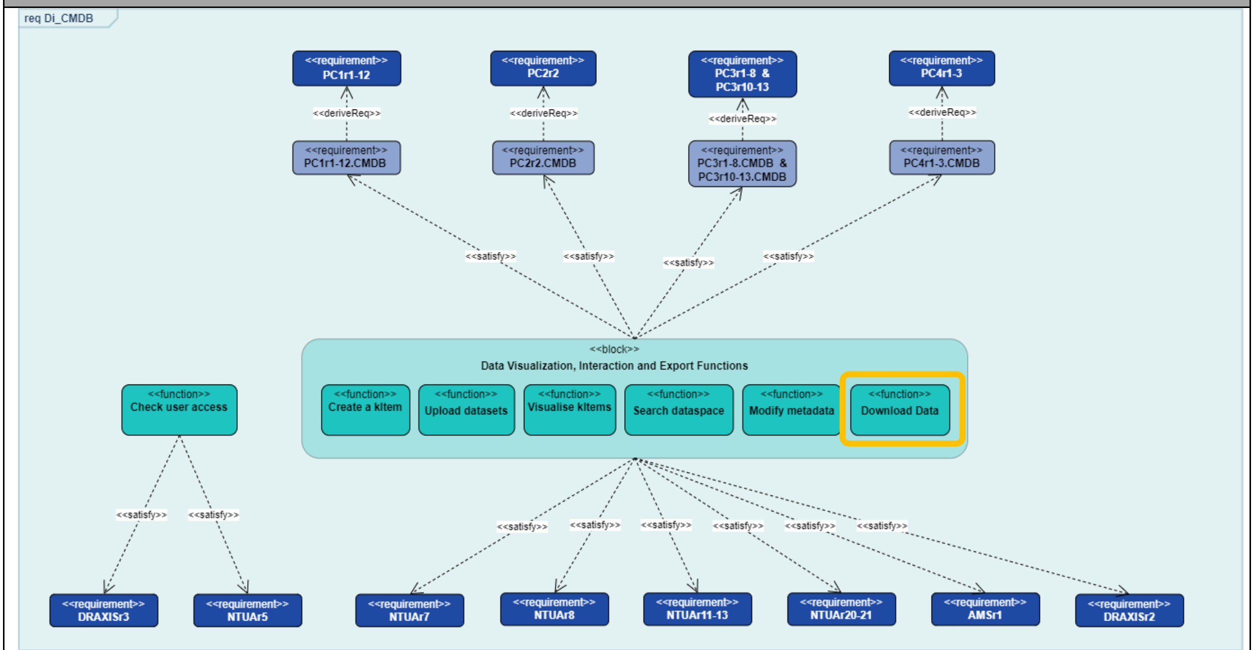
Table 60. Data upload functional component diagram, structure and requirements

5.2.3.2 Data download

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

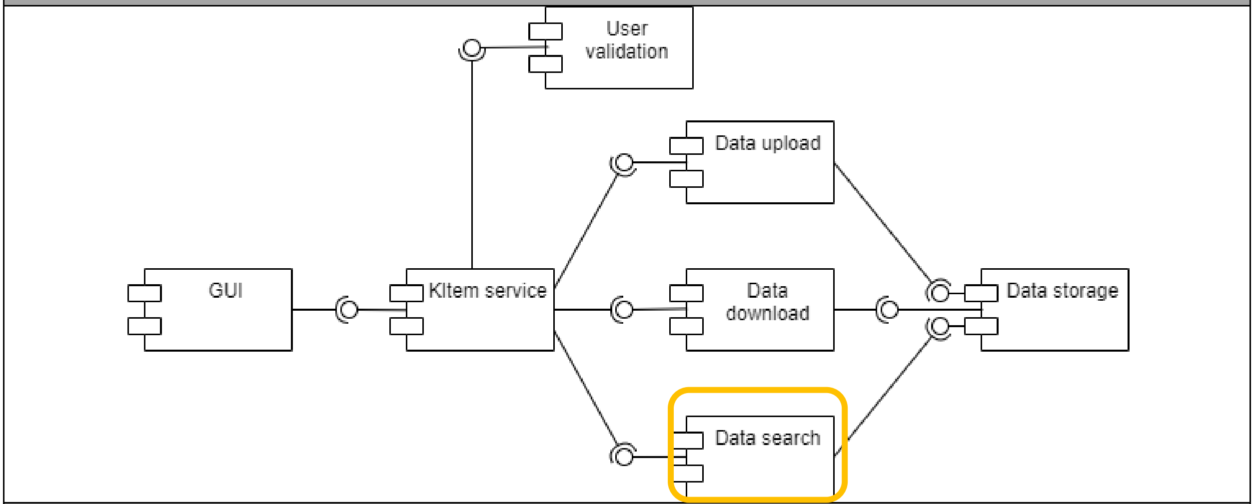


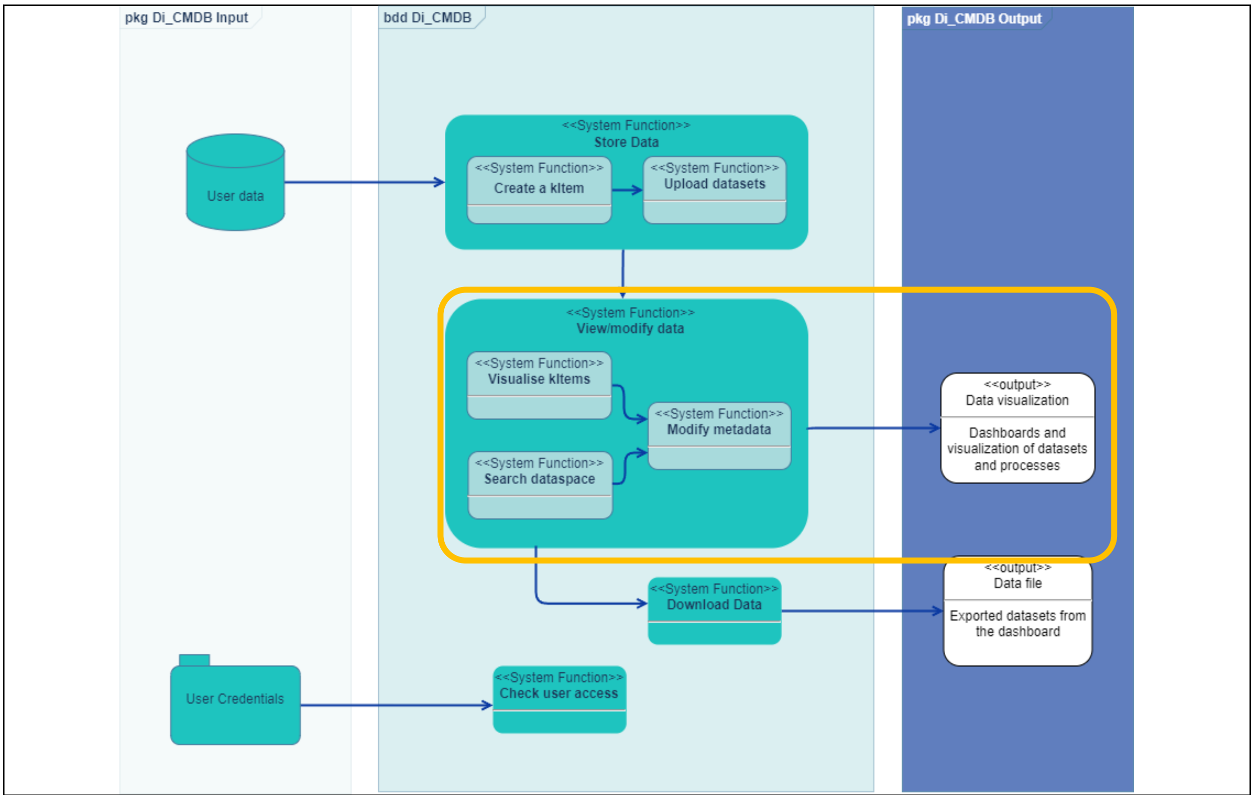
Describe the connection between the functional component and the requirements

The data download component is related to all the data export tasks and satisfies the following requirements: - PC1r1-12 - PC3r1-13 - PC4r1-3

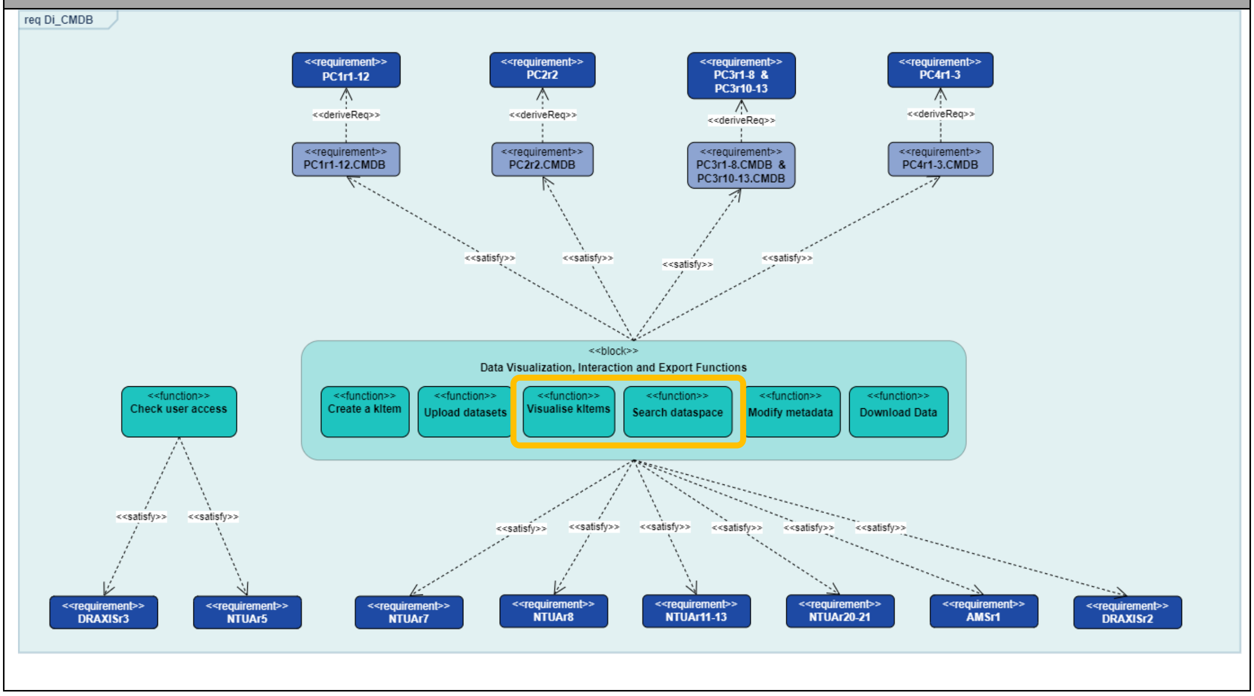
Table 61. Data download functional component diagram, structure and requirements

5.2.3.3 Data search

Functional component diagram

Functional structure diagram



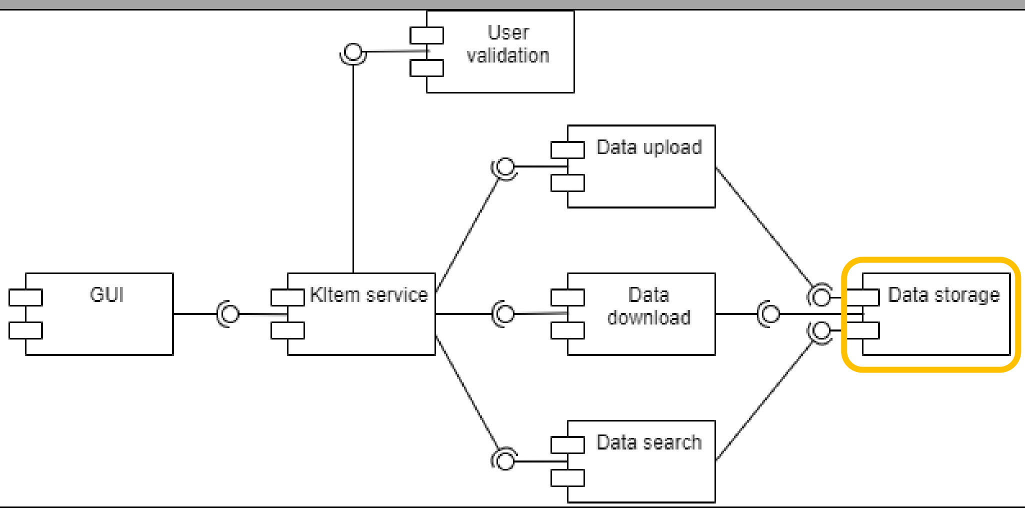
Requirements linked to the functional component

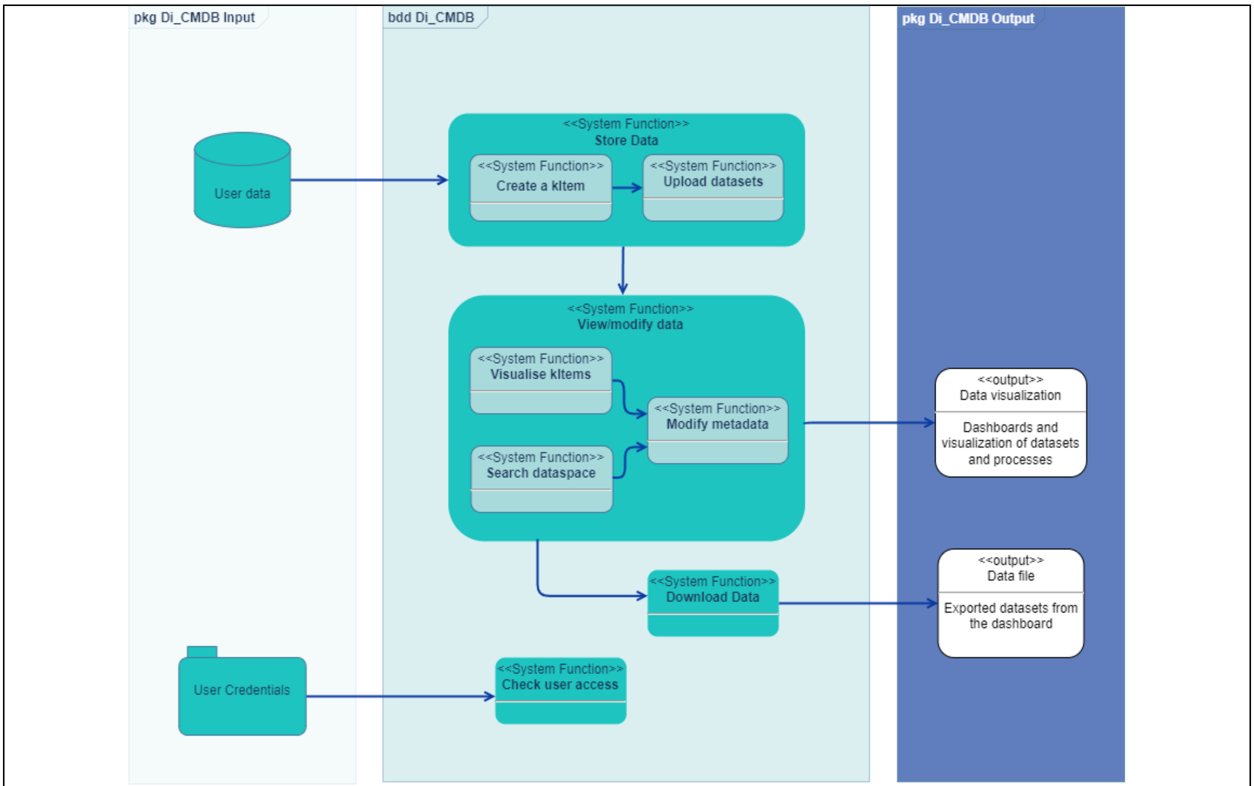


Describe the connection between the functional component and the requirements
The data search component is related to the data browsing tasks and satisfies the following requirements: - PC1r1-12 - PC4r1 - NTUAr11

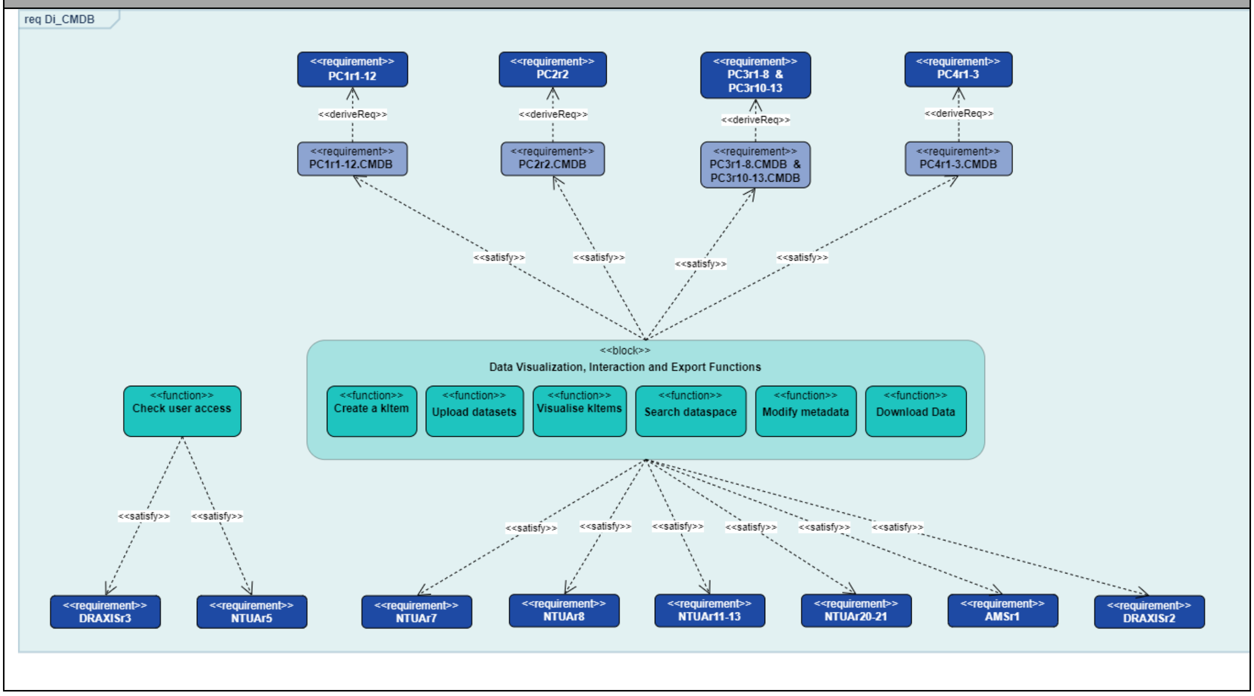
Table 62. Data search functional component diagram, structure and requirements

5.2.3.4 Data storage

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements
The data storage component handles should cover all the internal data saving and access requirements. - PC1r1-12 - PC2r2 - PC3r1-13 - PC4r1-3 - DRAXISr2

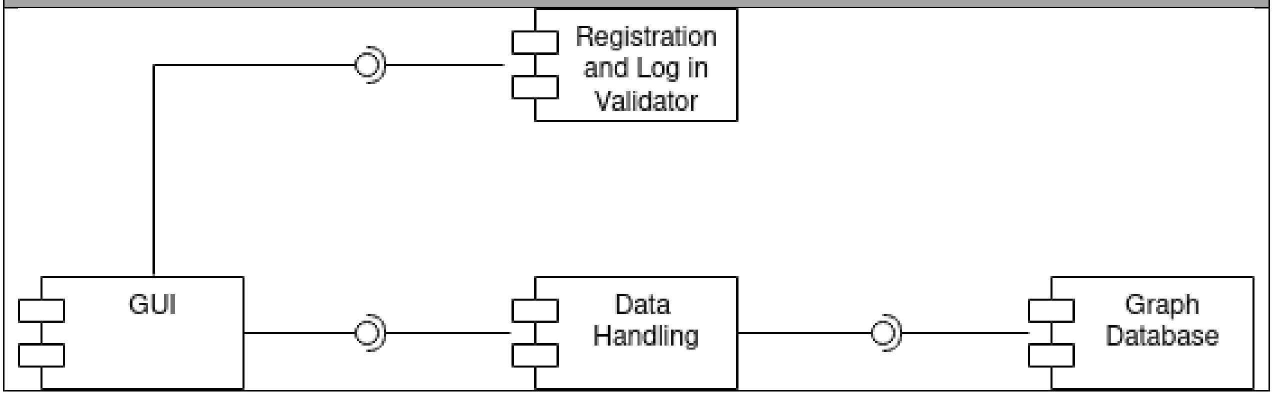
Table 63. Data storage functional component diagram, structure and requirements

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

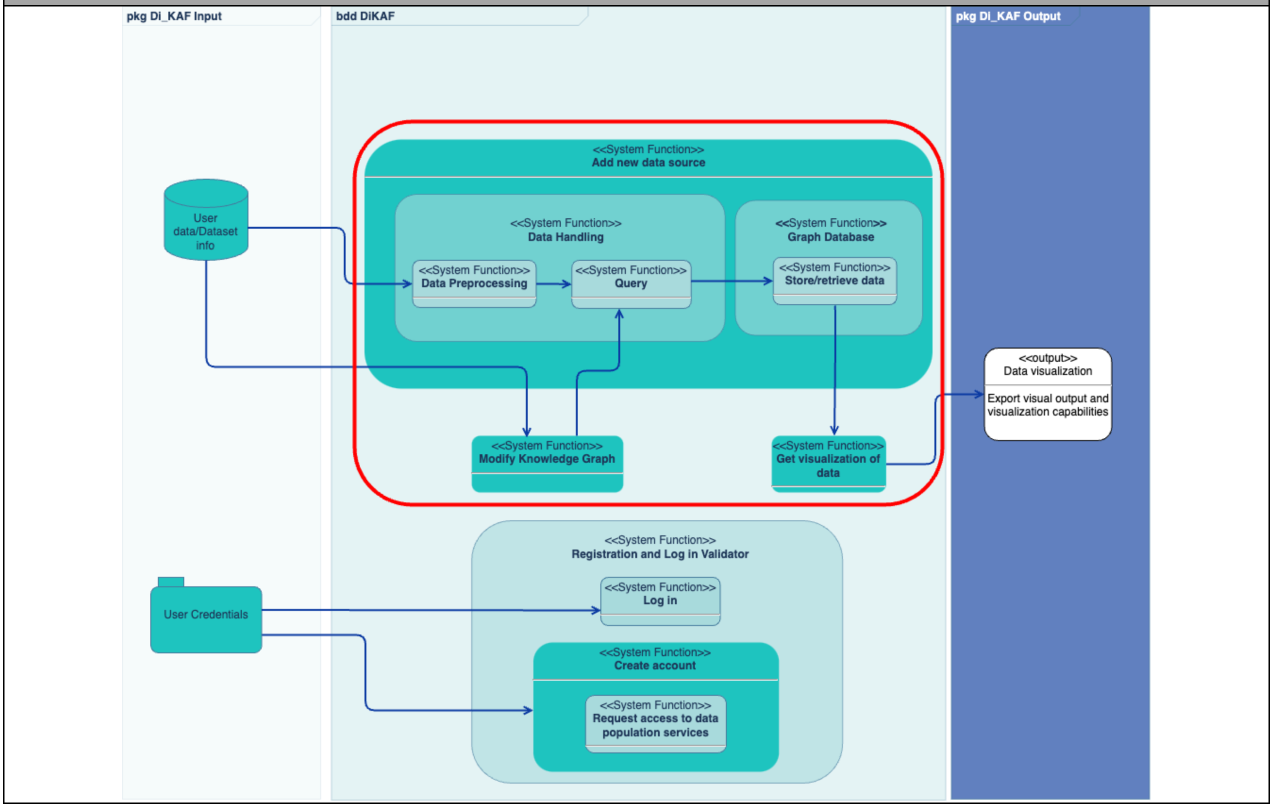
Concerning the KAF toolkit, the General Architecture and Functional components remain the same, as described in the first version of the Viewpoints' Deliverable. The only update pertains to the Registration and Log in Validator, which permits access according to users and roles defined through Keycloack. Users are granted access to specific resources and capabilities according to their access permissions.

5.3.1 Functional components, functional structure and requirements interactions

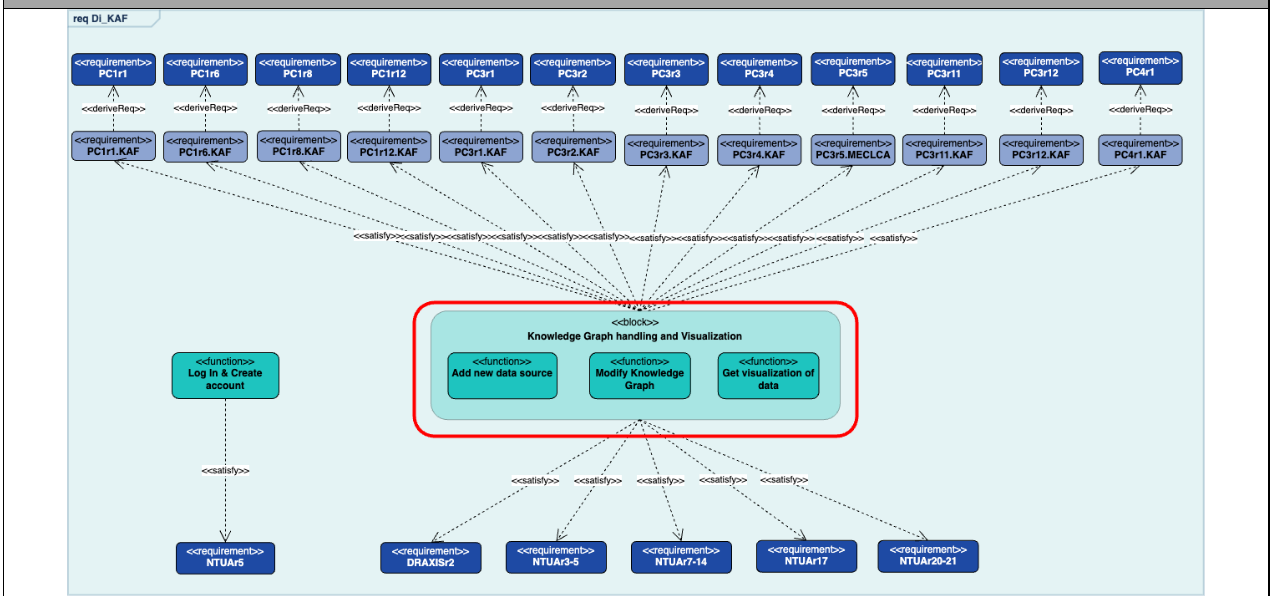
5.3.1.1 Knowledge Graph handling and Visualization

Functional component diagram


Functional structure diagram



Requirements linked to the functional component



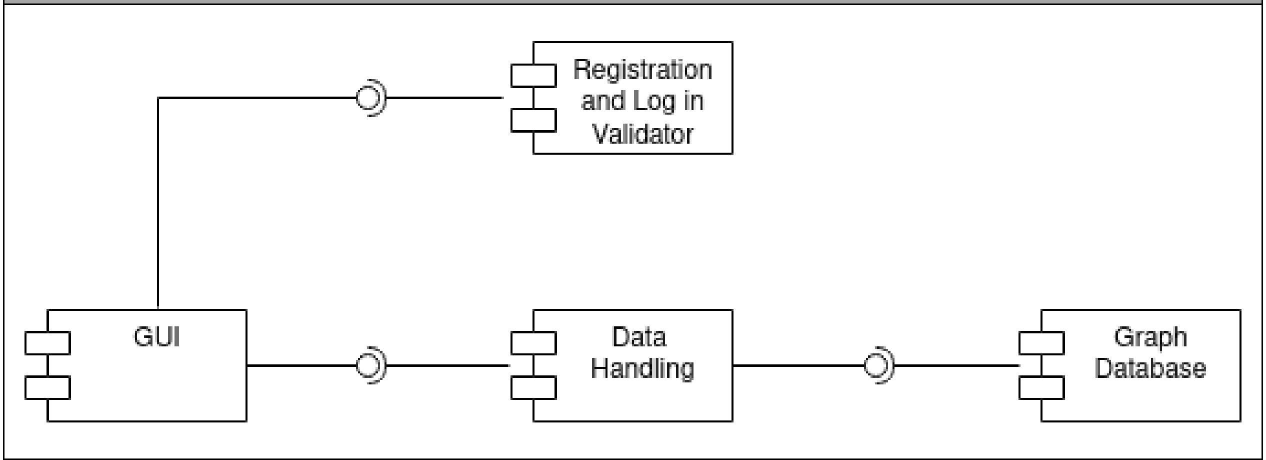
Describe the connection between the functional component and the requirements-
-DRAXISr2: Data collected from pilots regarding the KAF toolkit will be appropriately organized and maintained. -NTUAr10, NTUAr12, NTUAr14, NTUAr17: The Graph Database along with its Knowledge Graph schema should store and organize semantically correct data, provide suggestions concerning processes and materials and be aligned with other popular material related ontologies. The toolkit should also support multiple data formats. -NTUAr3, NTUAr8, NTUAr9: The toolkit will provide secure communication with at least the other 2 toolkits of the suite over channels supporting widely used protocols. -NTUAr13, NTUAr20: Materials and processing related data along with toolkit's implementation configurations should be updated when required and checked regularly to ensure smooth, correct and up-to-date operation of the toolkit. **-NTUAr7. NTUAr11, NTUAr21:** Data structure, visual representation and user's interaction with the toolkit should be documented in detail. Dashboards should provide visualization, modification, data addition and export capabilities, concerning materials and processes. **-NTUAr4:** KAF's simple and friendly UI ensures that users can visualize, modify and interact with results effectively and efficiently. **PC1r1.KAF, PC1r6.KAF, PC1r8.KAF, PC1r12.KAF, PC3r1.KAF, PC3r2.KAF, PC3r3.KAF, PC3r4.KAF, PC3r5.KAF, PC3r11.KAF, PC3r12.KAF, PC4r1.KAF,** (grouped): The user can query materials and industrial processes related data and receive suggestions about materials and their future state through the toolkit's UI.

Table 64. Knowledge Graph and Visualization functional component diagram, structure and requirements

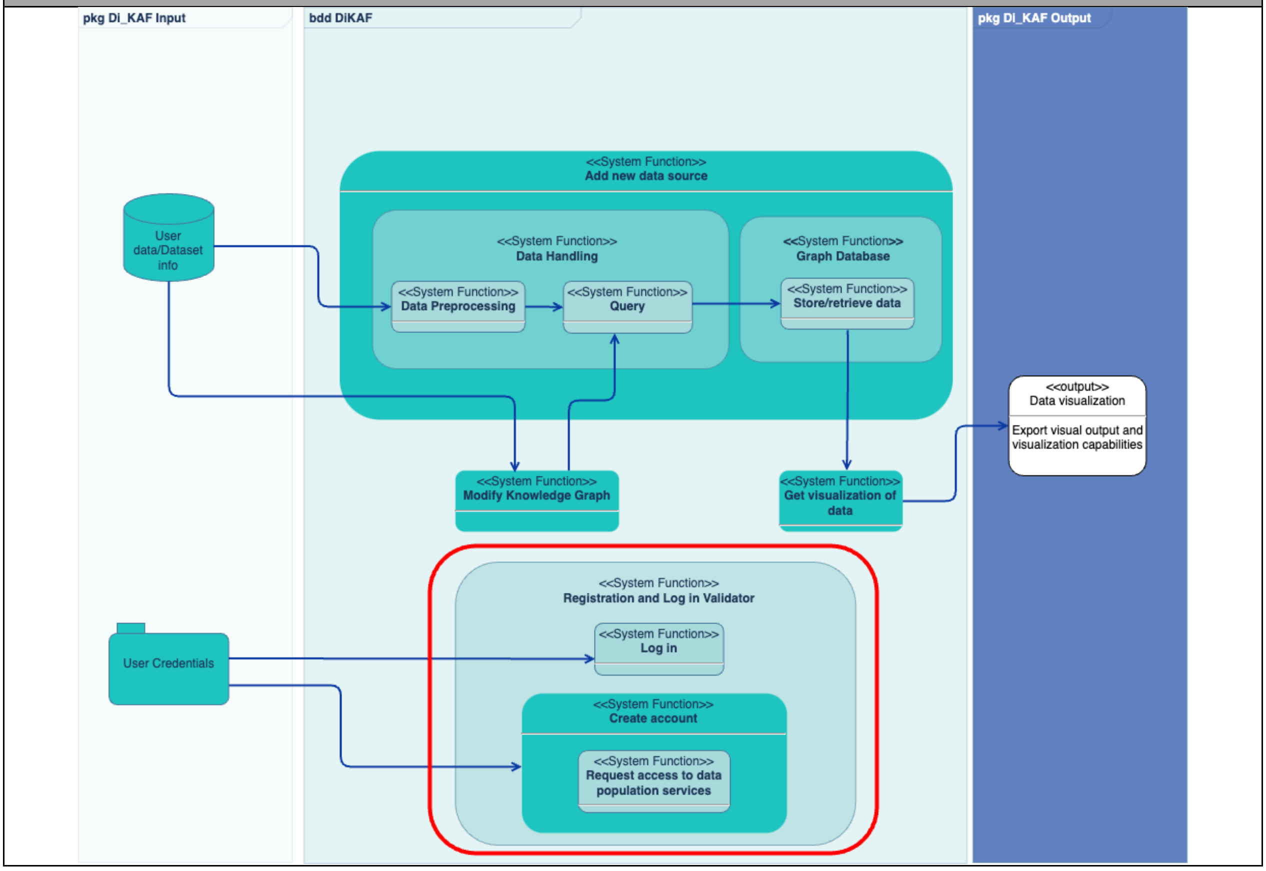
Co-funded by the European Union

5.3.1.2 Log in & Create account

Functional component diagram



Functional structure diagram



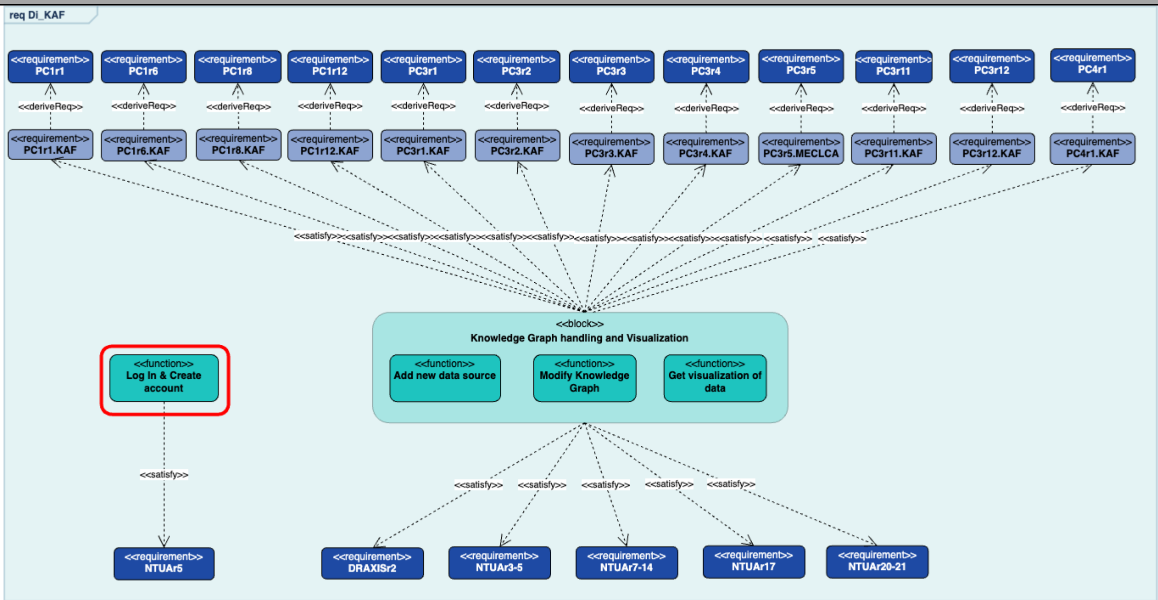
Requirements linked to the functional component

Describe the connection between the functional component and the requirements
- **NTUAr5** (*The toolkit shall be accessible only by authorised users):* Only authorized users shall be able to log in, view and edit the data available through the tool. To this end, the Keycloak software is essential.

Table 65. Log in and Create Account functional component diagram, structure and requirements

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

5.4.1 General Architecture

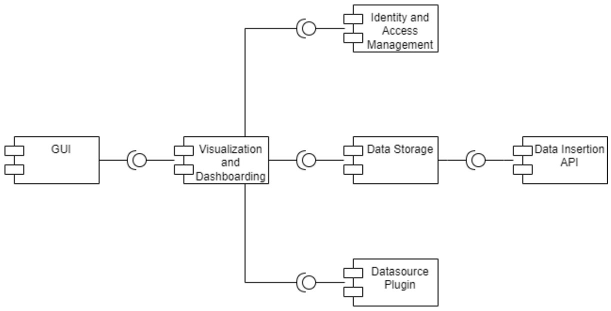


Figure 34. Materials Environmental and Cost Life Cycle Assessment Diagram

This diagram shows how users interact with the system through a simple **Graphical User Interface (GUI),** which connects to a central Visualization and Dashboarding module. This module allows users to view and interact with data in a visual format. It links to four key parts:

- The **Identity and Access Management** (previously Log In Validator), which checks that only authorized users get access.
- The **Data Storage** module, where data is securely kept and can be pulled up for viewing.
- The **Data Insertion API**, which lets users add new data into the storage.
- The **Datasource Plugin**, which allows users to directly upload data to the dashboards.

Once logged in, users can access the Visualization and Dashboarding module, where data from the Data Storage is displayed in a visual format. If users need to add more data to the storage, they can do it through the Data Insertion API, and the new data appears instantly on the dashboard. There's also the Datasource Plugin that allows users to upload data directly to the Visualization and Dashboarding module, which updates and shows up in real-time through the dashboards.

5.4.2 Functional components

The tables below describe updates to the main functional components identified in the General Architecture of the MEC-LCA toolkit. The functional components "Data storage" and "Data insertion API" remain the same as in version 1.

5.4.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Visualization and Dashboarding component.*	
	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 to specific resources (according to user's access permissions)
Modify visualization	- Modifications selected	- Visualization modified
Edit data inline	- Data to edit inline	- Data is edited inline

Table 66. GUI of the functional component of the DiMEC-LCA

5.4.2.2 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 to specific resources (according to user's access permissions)*
Modify visualization	- *Modifications selected*	- *Visualization modified*
Edit data inline	- *Data to edit inline*	- *Data is edited inline*

Table 67. Visualization and dashboarding of the functional component of the DiMEC-LCA

5.4.2.3 Identity and Access Management

Name:	*Identity and Access Management*	
Description:	*It manages user identities and controls access to Visualization and Dashboarding component's resources.*	
	Interface	
Operation	***Inputs***	***Outputs***
Log in	- Credentials	- Access is granted to specific resources (according to user's access permissions)

Table 68. Identity and Access Management of the functional component of the DiMEC-LCA

5.4.2.4 Data source plugin

Name:	*Data source Plugin*	
Description:	*It is used to upload data to the Visualization and Dashboarding component.*	
	Interface	
Operation	***Inputs***	***Outputs***
Insert or import data	- Data to insert or import (environmental and cost LCA results, KPI measurements)	- Data is imported
Edit data inline	- Data to edit inline	- Data is edited inline

Table 69. Data source plugin of the functional component of the DiMEC-LCA

5.4.3 Interactions

The sequence diagrams below describe updates to the main interactions identified for the MEC-LCA toolkit. The interactions "Store user input", "Download data" and "Modify visualization" remain the same as in version 1.

5.4.3.1 Upload data

The administrator can also upload files and visualize the data through Grafana via Data source Plugin (Infinity Plugin), without the need for additional database operation. Furthermore, the administrator can perform inline data editing via the Data source Plugin.

This only replaces the "Store user information 2" diagram in version 1, the rest of this interaction remains the same.

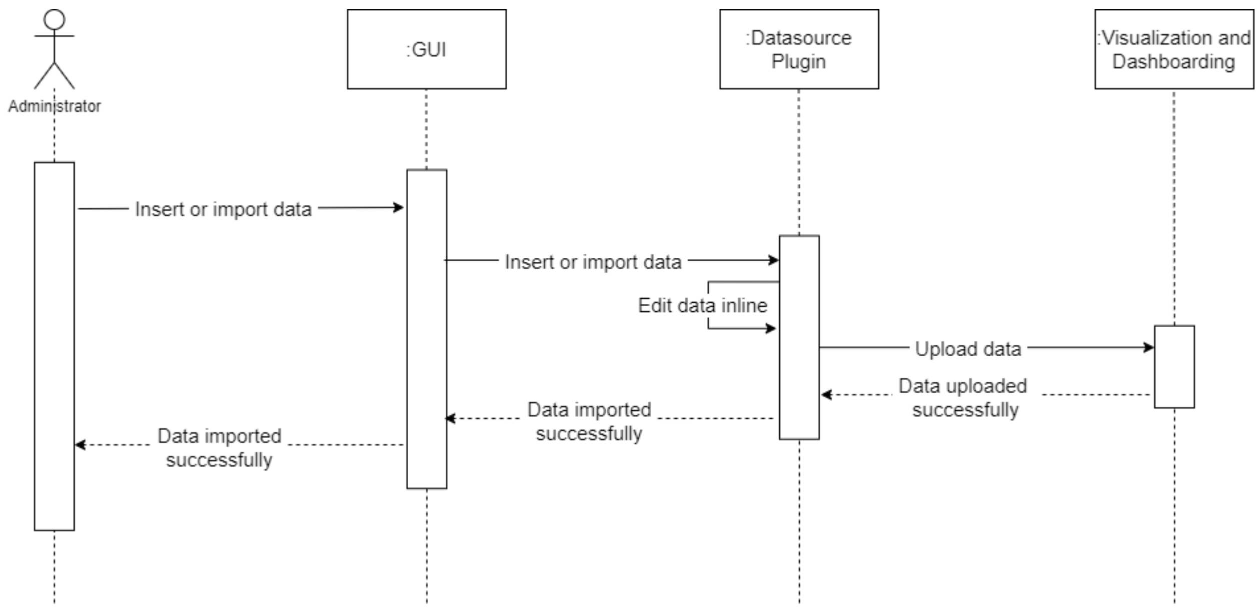


Figure 35. Sequence diagram of the DiMEC-LCA toolkit component "Upload data"

5.4.3.2 Visualize LCA results

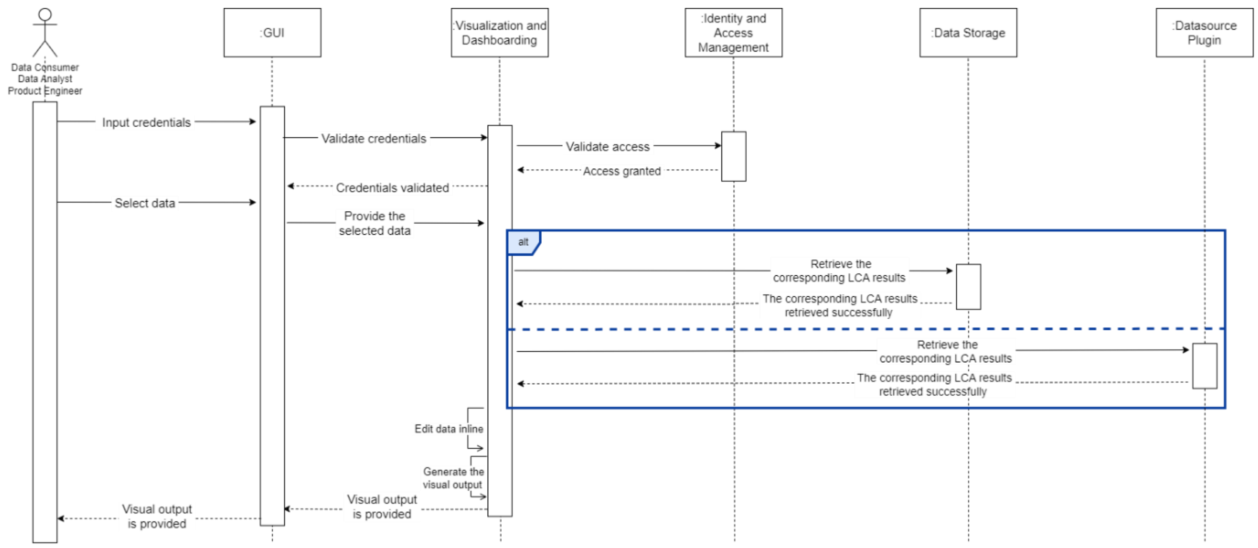


Figure 36. Sequence diagram of the DiMEC-LCA toolkit component "Visualize LCA Results"

5.4.3.3 Visualize KPIs

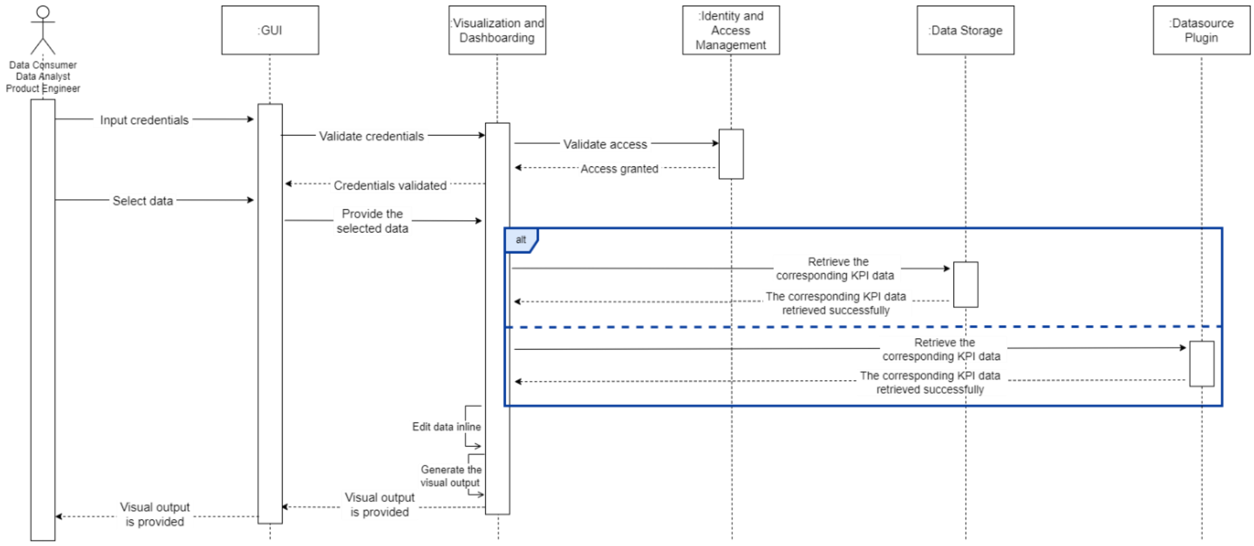


Figure 37. Sequence diagram of the DiMEC-LCA toolkit component "Visualize KPIs"

5.4.3.4 Log in

The Administrator, the Data Consumer, the Product Engineer or the Data Analyst navigate to the Visualization and Dashboarding component'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 component receive the log-in request and verifies the entered credentials through the Identity and Access Management component (Keycloack). If the credentials are valid, the Visualization and Dashboarding component grants access to specific resources, according to user's access permissions.

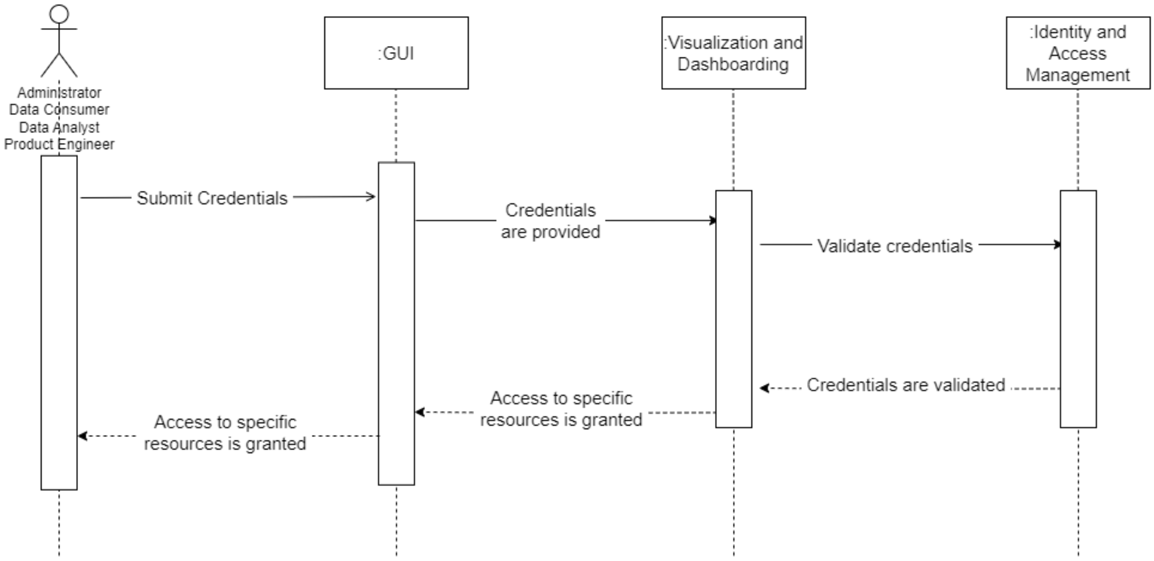


Figure 38. Sequence diagram of the DiMEC-LCA toolkit component "Log in"

5.4.4 Mapping of functional components to IIRA functional domain

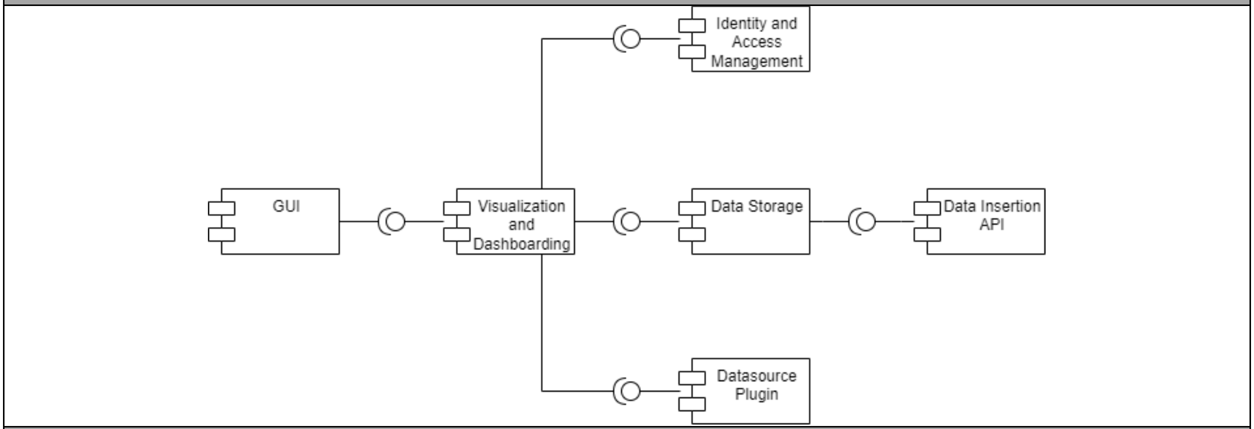
Component	***Domain***	***Subdomain***
GUI	*Application*	*UI*
Visualization and Dashboarding	*Application*	*Logic & Rules*
Identity and Access Management	*Application*	*Logic & Rules*
Data Insertion API	*Application*	*API + Logic & Rules*
Data Storage	*Information*	*Data*
Datasource Plugin	*Application*	*Logic & Rules*

Table 70. Mapping of the functional component of the DiMEC-LCA

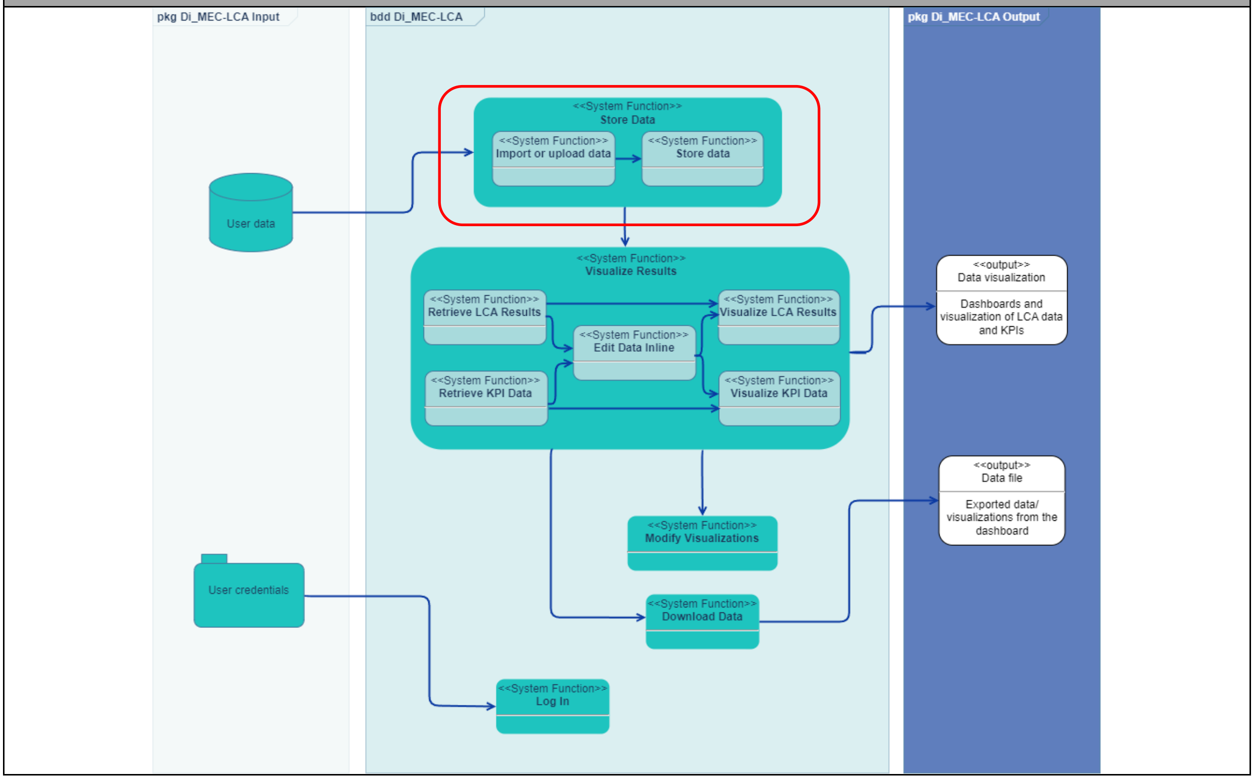
5.4.5 Functional components, functional structure and requirements interactions

5.4.5.1 Store Data

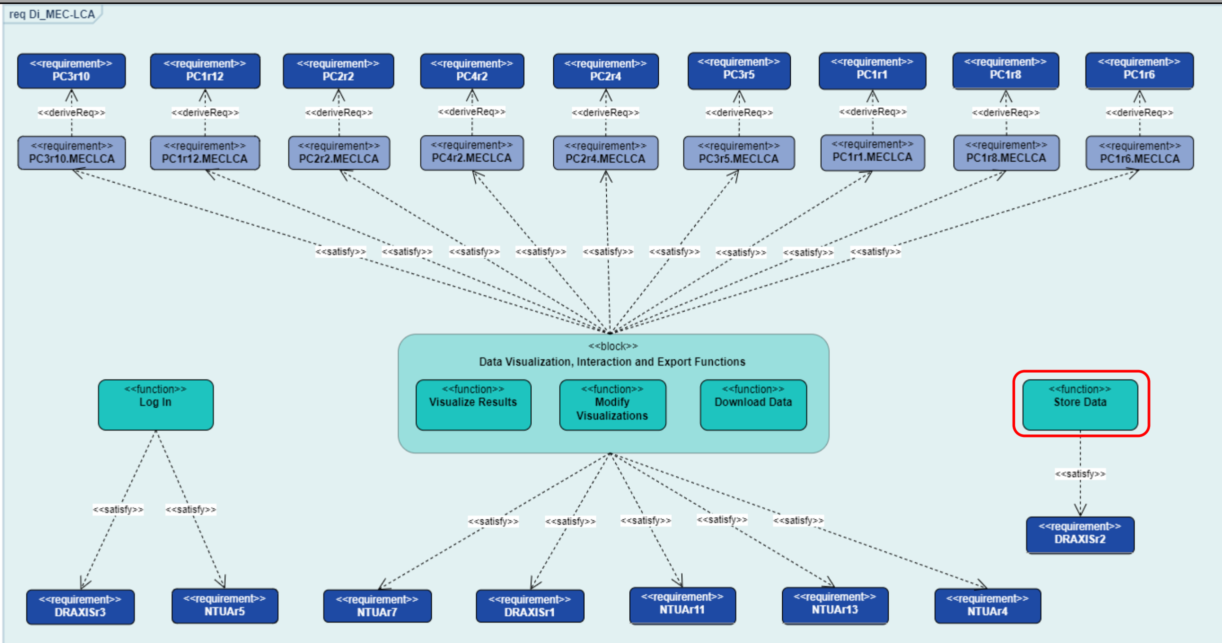
Functional component diagram



Functional structure diagram



Requirements linked to the functional component



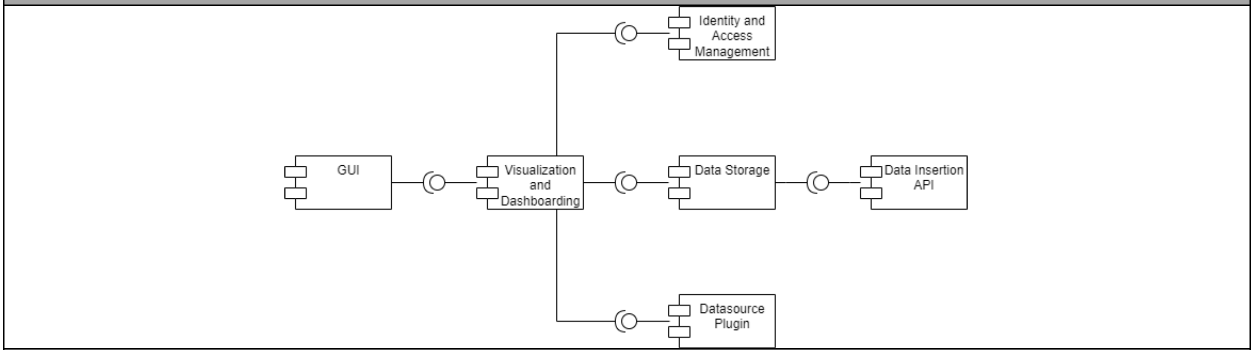
Describe the connection between the functional component and the requirements

- **DRAXISr2** (*Data shall be collected from the pilots and maintained in order to provide the required outputs*): Data collected in the context of T2.3 and T7.5 (Impact Assessment and KPI Validation of DiMAT Toolkits) which are to be used for MEC-LCA will be appropriately organized and maintained.

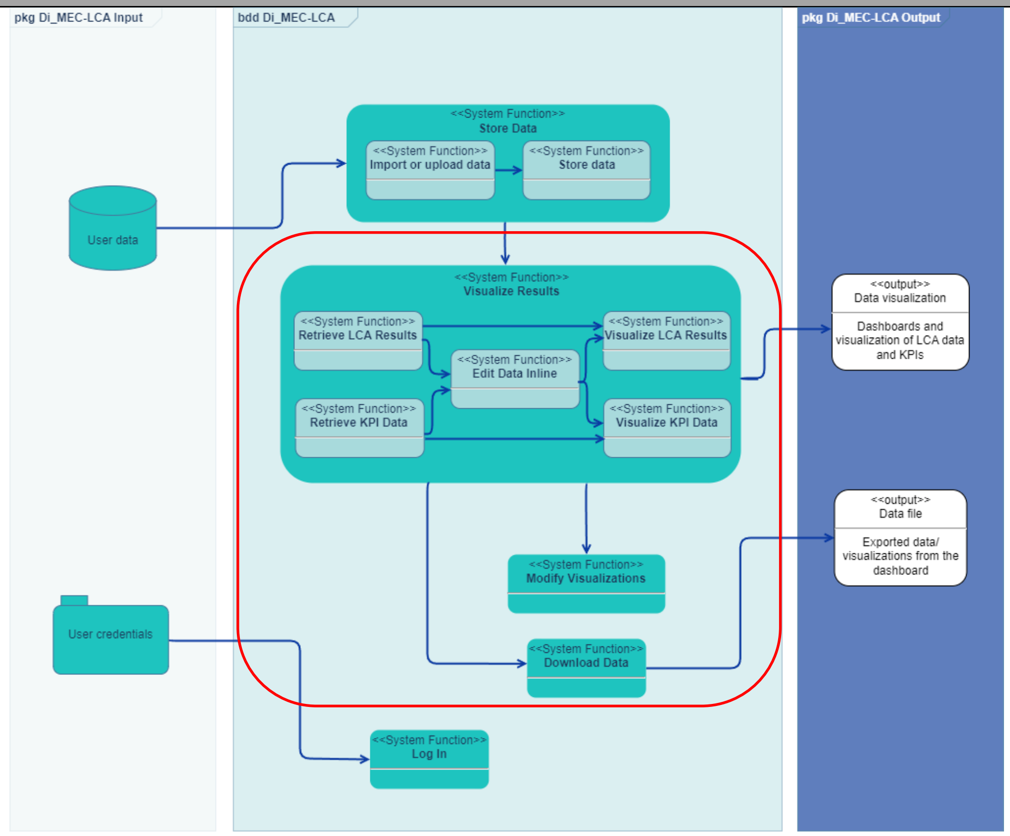
Table 71. Store data functional component diagram, structure and requirements

5.4.5.2 Visualize Results & Modify Visualizations & Download Data

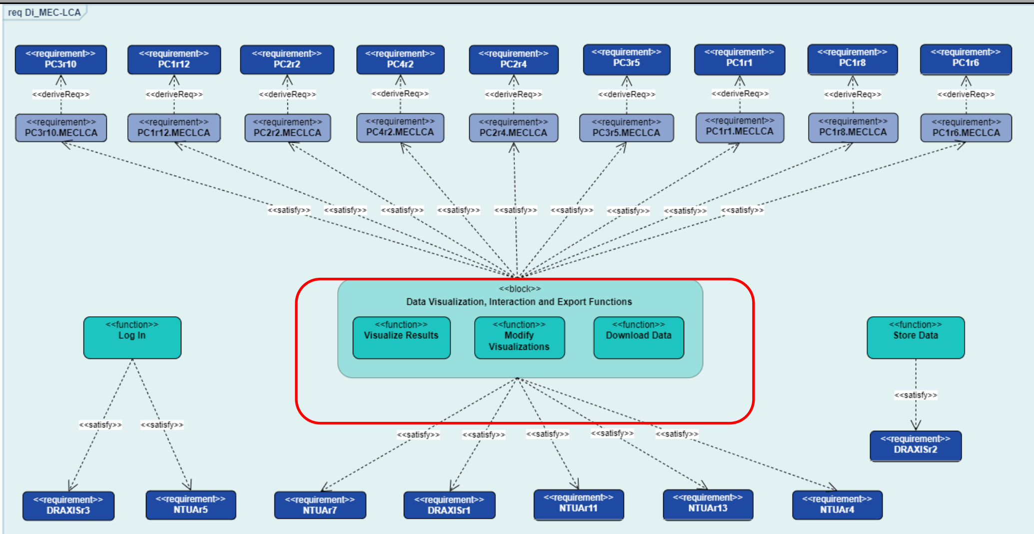
Functional component diagram



Functional structure diagram



Requirements linked to the functional component



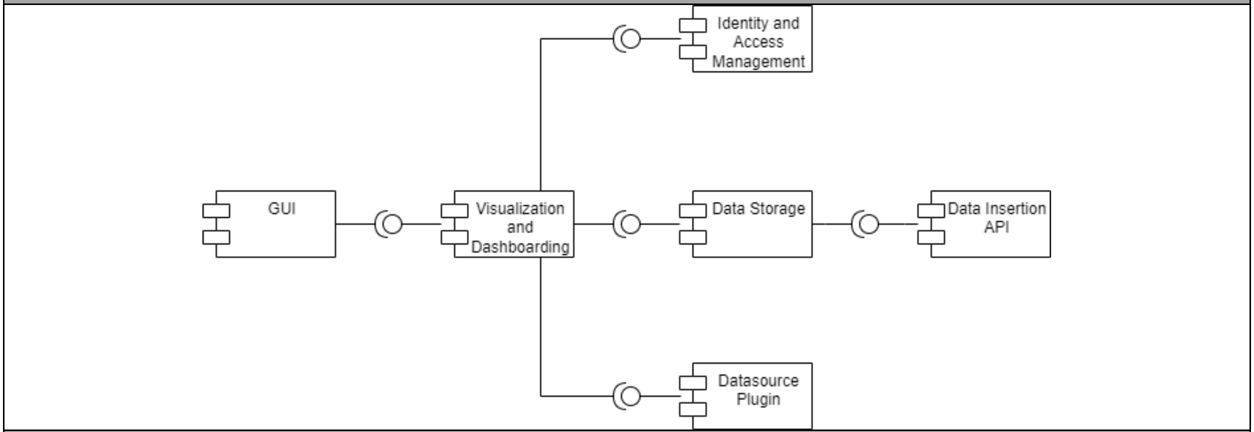
Describe the connection between the functional component and the requirements

- **NTUAr7** (*The toolkit shall provide sufficient documentation*): Visual representation, user interaction and data insights shall be effectively communicated and documented. Supplementary materials, such as demos and project deliverables, will guide users further.
- **DRAXISr1** *(The toolkit's outputs shall be based on user input, rather than real-time data)*: Visual representation of results that are based on user input from the pilots, rather than real-time data, such as the information needed to conduct the LCA/LCC or the measurements required for the KPI tracking.
- **NTUAr11** *(The toolkit shall support data visualisation through dashboards in the User Interface)*: Visual representation of data is delivered through dashboards as part of the toolkit. Data can be modified and exported through the dashboard environment as well.
- **NTUAr13** *(The toolkit shall be appropriately updated when necessary):* Data, plots and other information shall be appropriately updated when necessary.
- **NTUAr4** *(The toolkit shall offer an intuitive UI):* MEC-LCA's simple and intuitive UI ensures that users can understand, visualize and interact with results effectively and efficiently. **PC3r10.MECLCA, PC1r12.MECLCA, PC2r2.MECLCA, PC4r2.MECLCA, PC2r4.MECLCA, PC3r5.MECLCA, PC1r1.MECLCA, PC1r8.MECLCA, PC1r6.MECLCA** (grouped): The user can access, visualize, if necessary, modify (update or add) and export data derived from the LCA/LCC results regarding the impact of the various pilot processes.

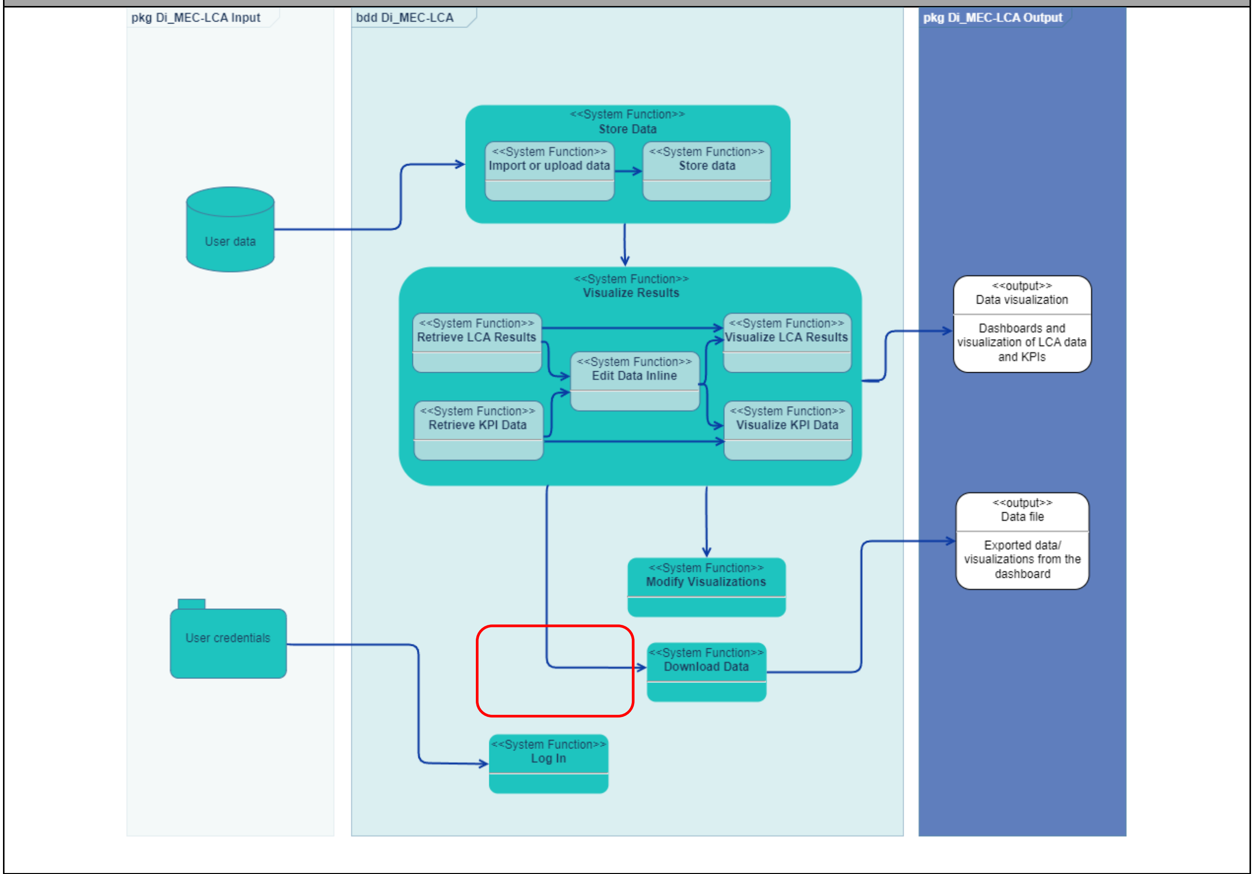
Table 72. Visualize Results & Download Data functional component diagram, structure and requirements

5.4.5.3 Log in

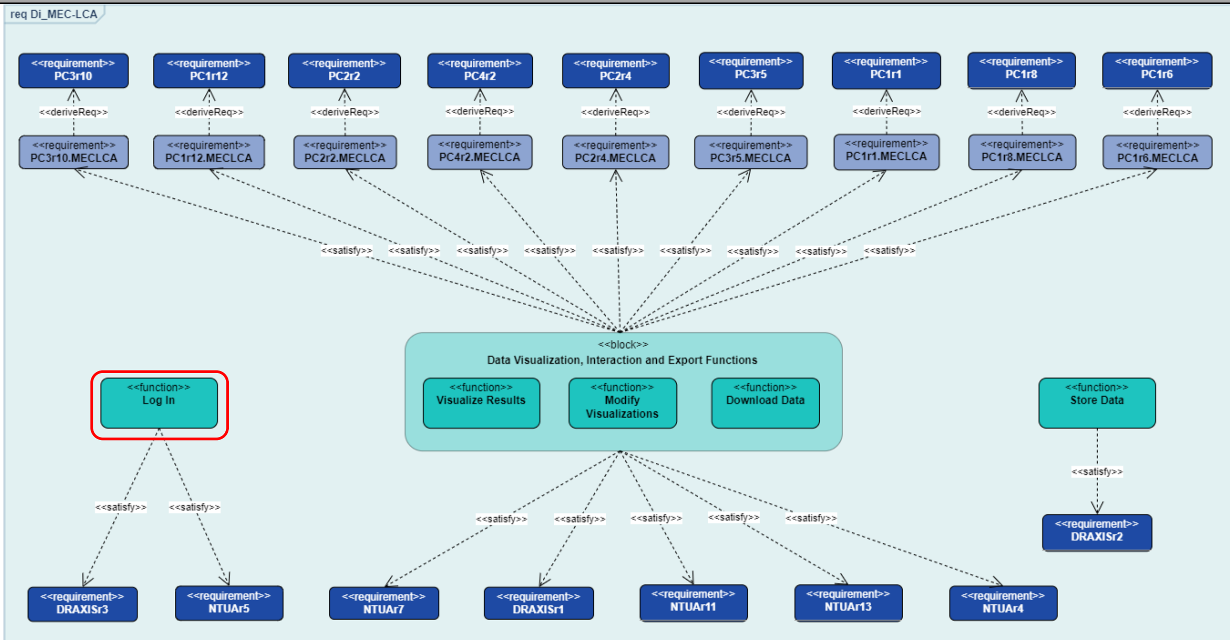
Functional component diagram



Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

- **DRAXISr3** *(The toolkits shall manage user access for every pilot partner*): Upon log in, users shall be able to access specific information, according to their rights.
- **NTUAr5** (*The toolkit shall be accessible only by authorized users):* Only authorized users shall be able to log in, view and edit the data available through the tool.

Table 73. Log in functional component diagram, structure and requirements

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

5.5.1 General architecture

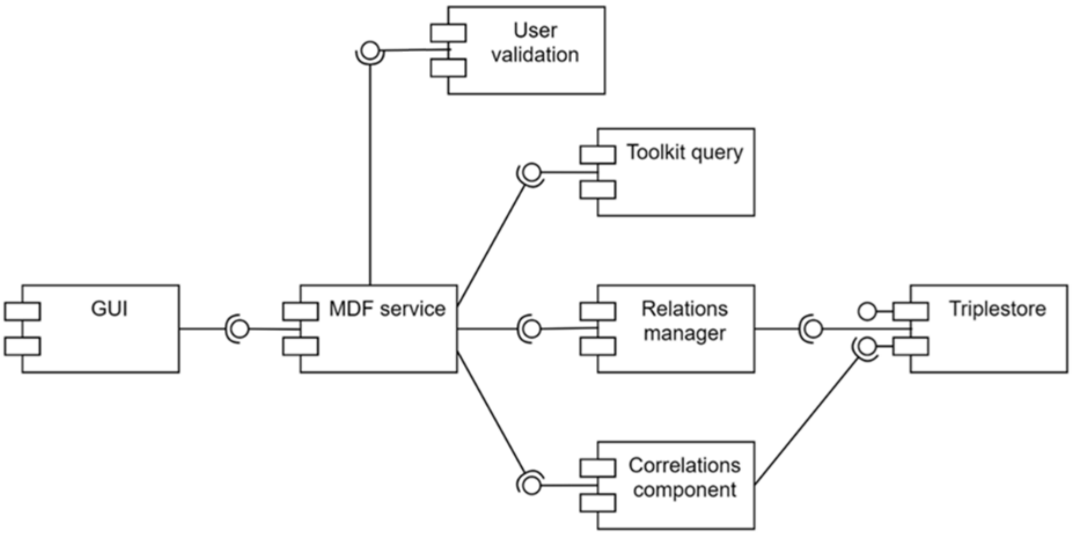
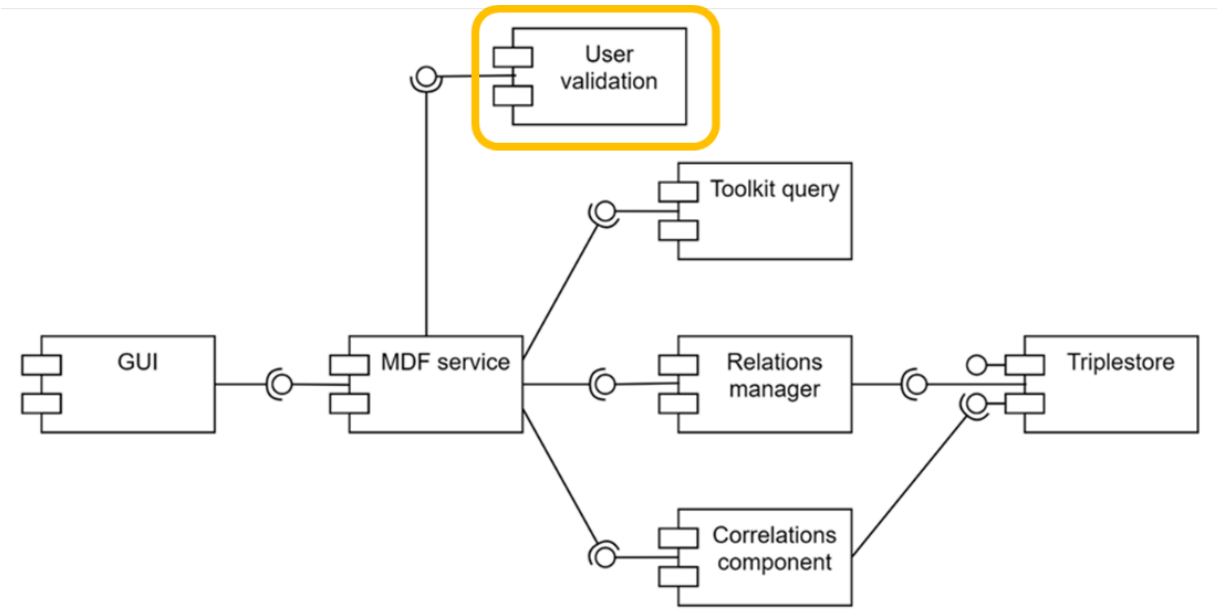


Figure 39: Materials Design Framework diagram

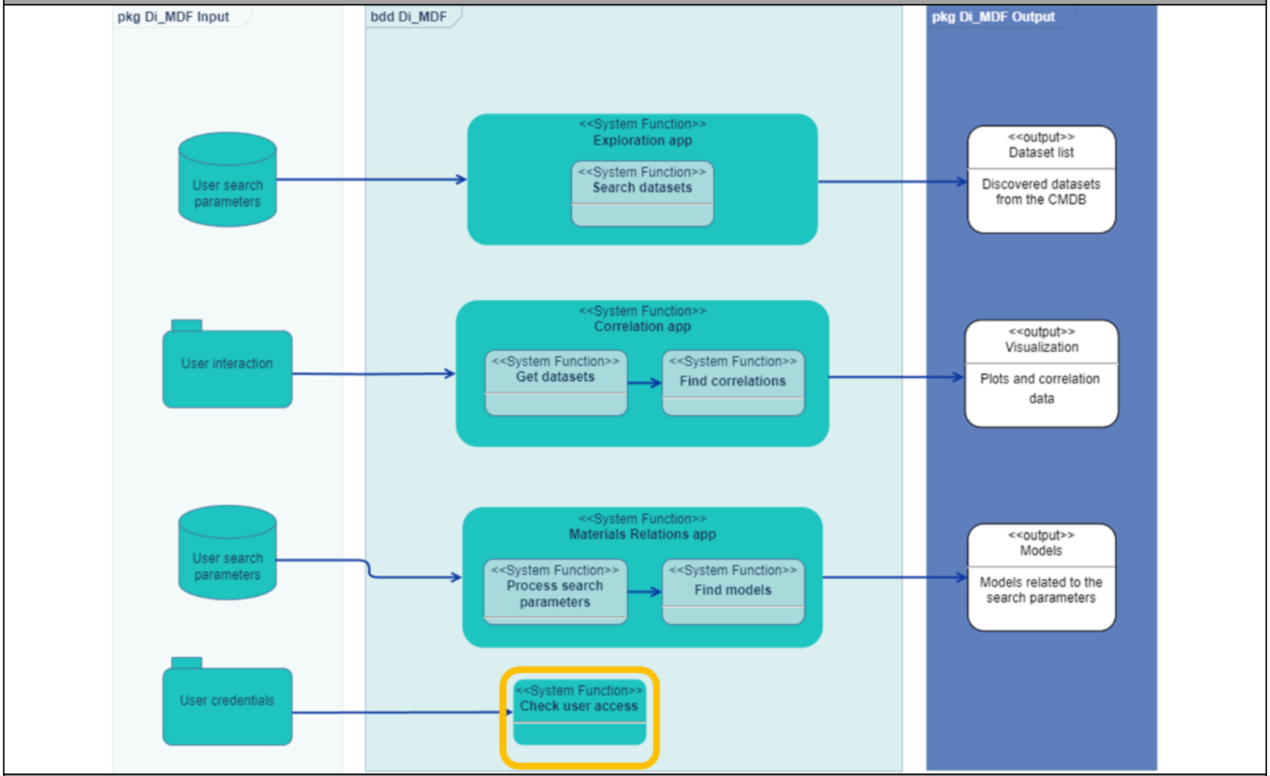
This diagram illustrates a system where the user interacts through a **Graphical User Interface (GUI),** which is part of the MDF toolkit that acts as a central component managing multiple functionalities. The process begins with **User Validation** (user authentication), which ensures that only authorized individuals can access the system. The MDF service connects to several modules: the **Toolkit Query** handles complex data queries, the (Material) **Relations Manager** makes knowledge about materials relations accessible, and the **Correlations Component** enables analyses of correlations between datasets accessible via CMDB. The three components interact with a **Triplestore**, a semantics-based data storage for storing and retrieving data including the relationships between data.

5.5.2 Functional components, functional structure and requirements interactions

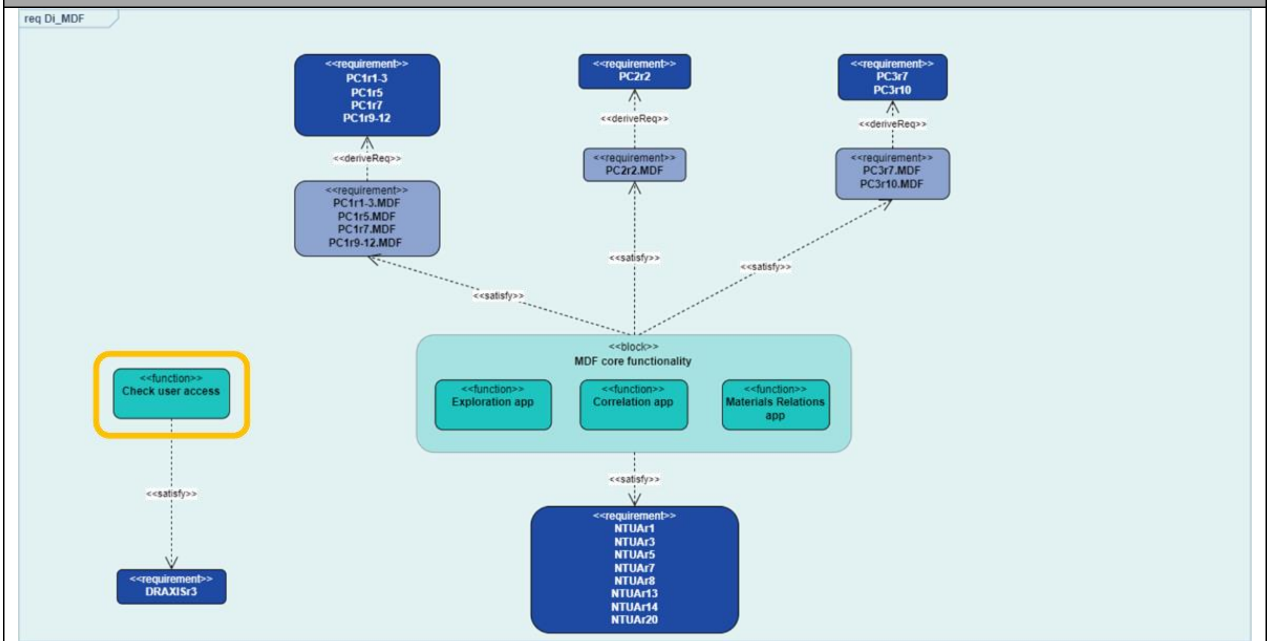
5.5.2.1 User validation and authentication

Functional component diagram


Functional structure diagram



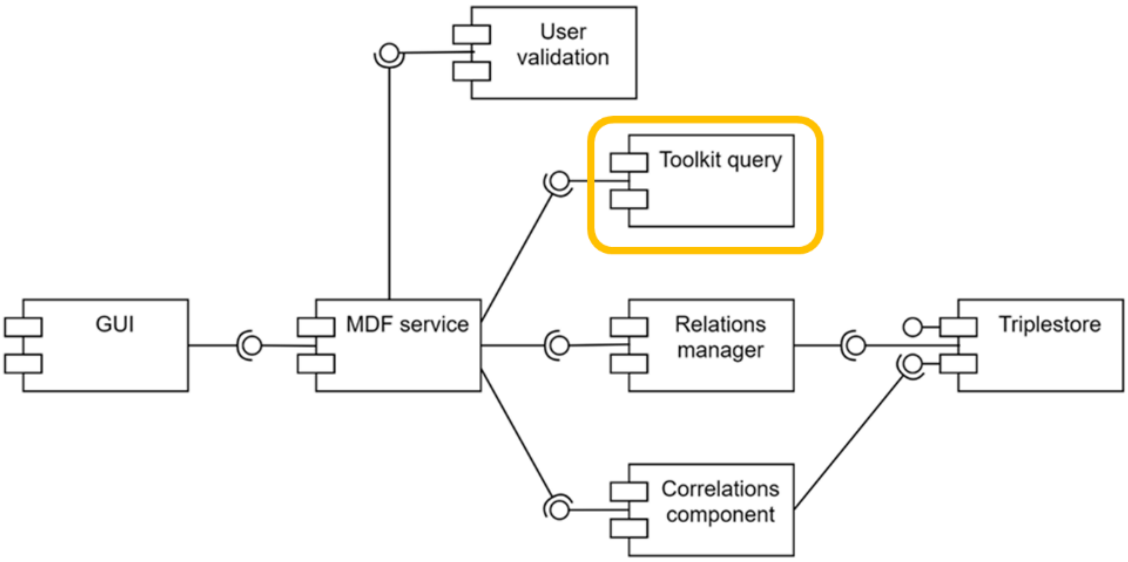
Requirements linked to the functional component

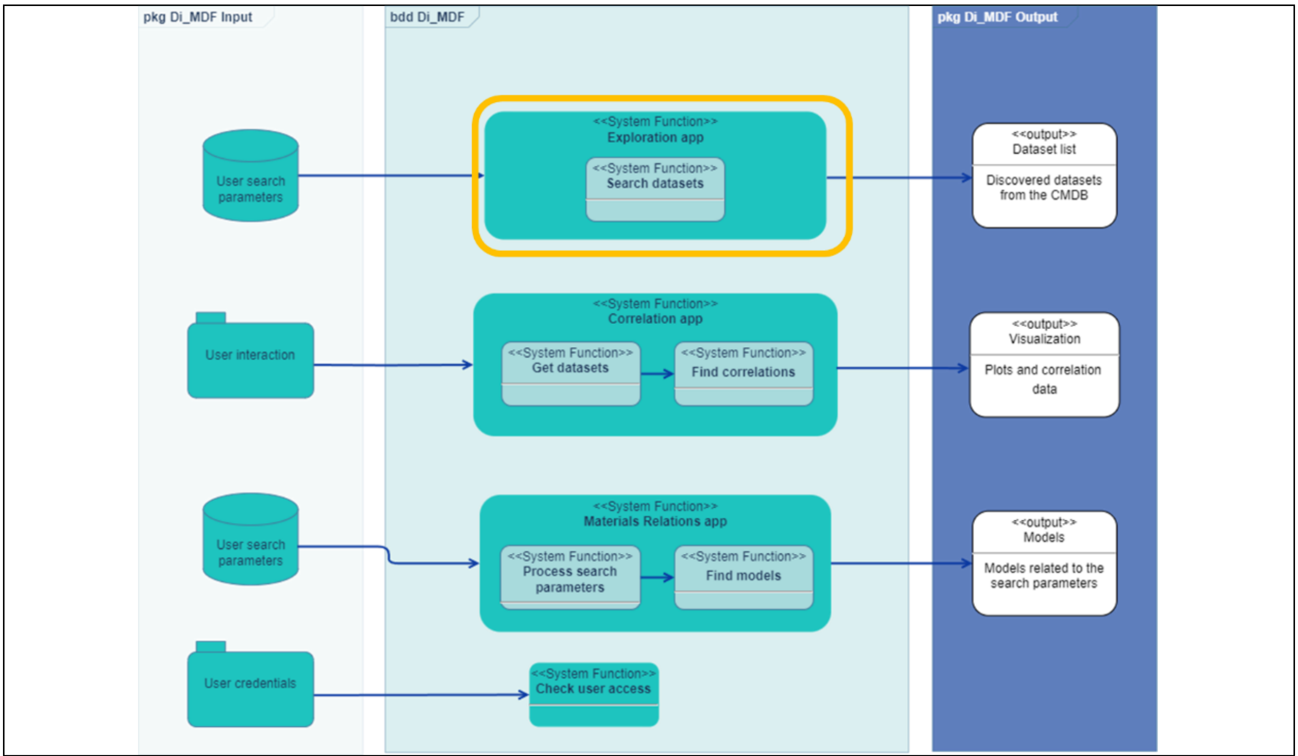


Describe the connection between the functional component and the requirements
This component takes care of the user authentication and authorization. As such, it will satisfy the following security requirements: - DRAXISr3 - NTUAr5 - NTUAr8

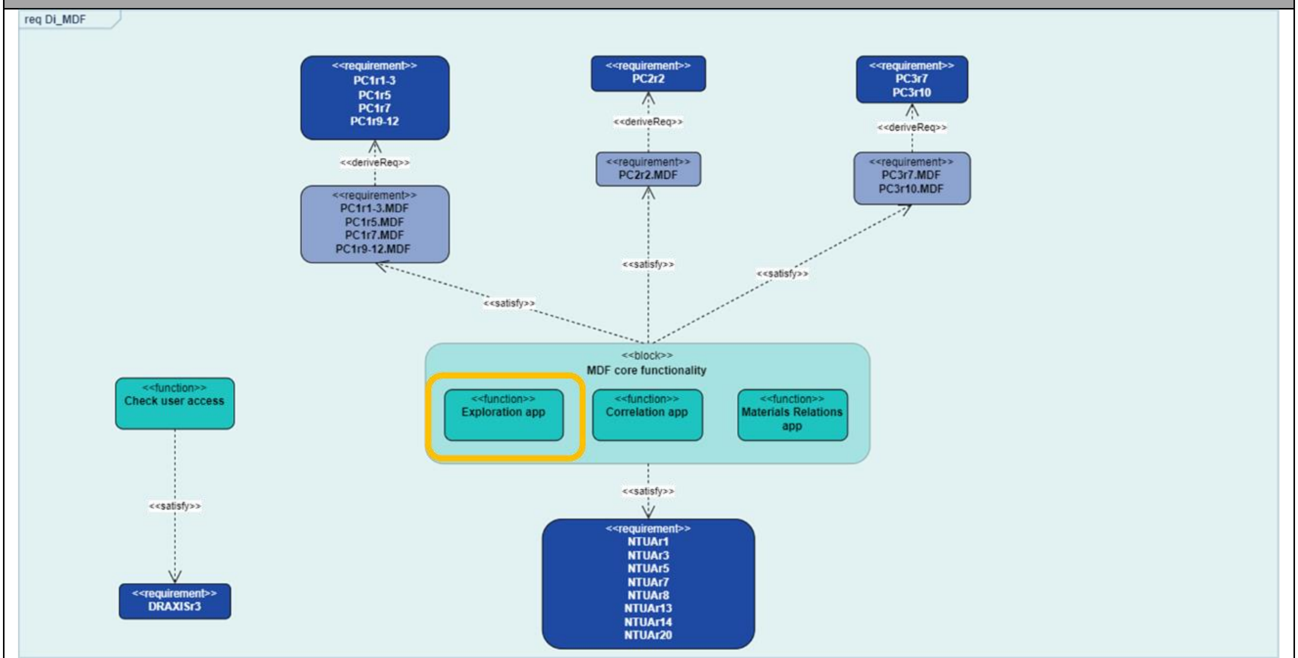
Table 74. User validation and authentication functional component diagram, structure and requirements

5.5.2.2 Exploration App

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

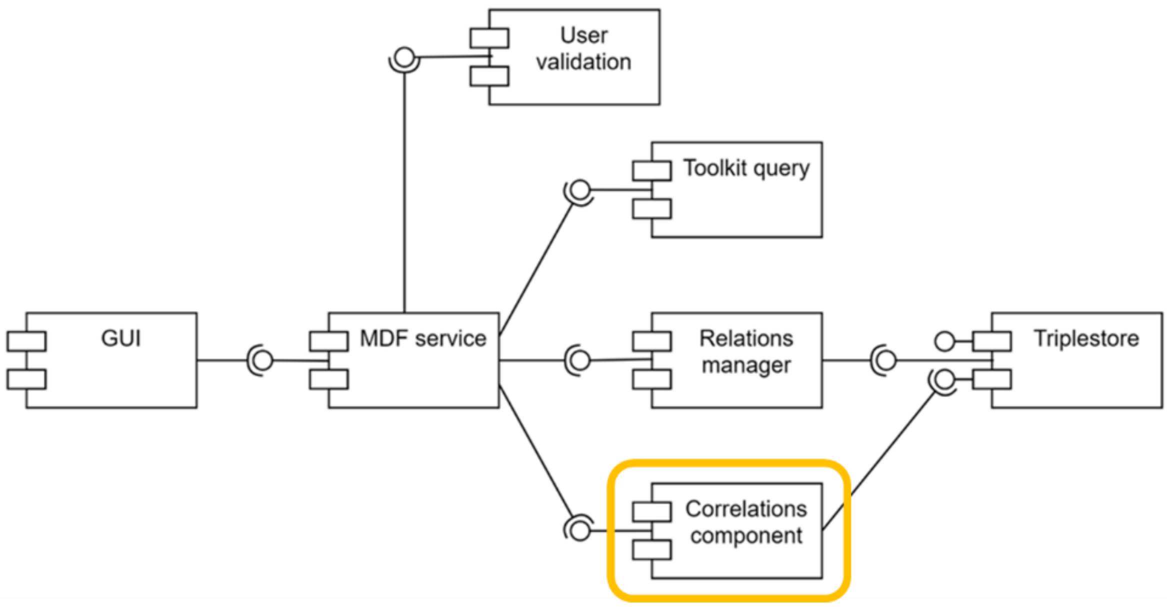


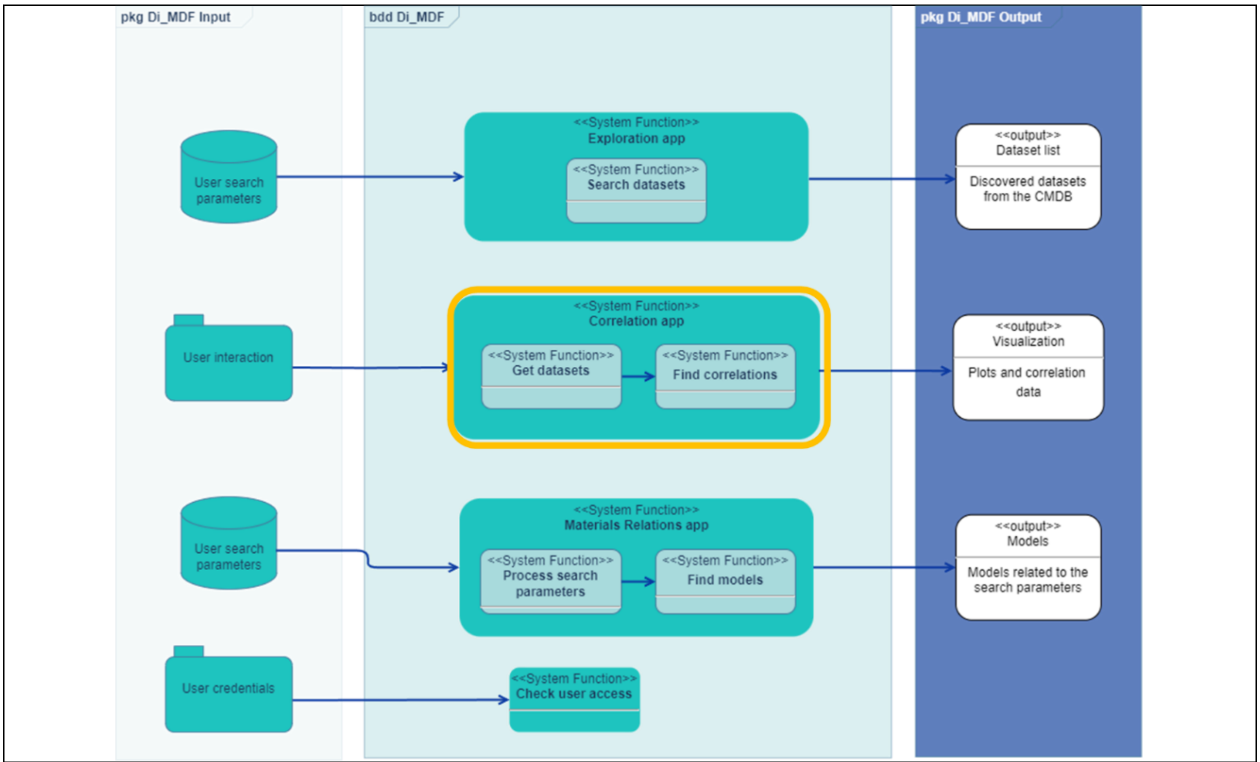
Describe the connection between the functional component and the requirements

- **PC1r1,2,9,10,11, PC3r7,10**: A search functionality is provided to browse all the existing toolkits for information about functionalities, datasets, models as well as properties and performance parameters (primary and secondary data). - **NTUAr1,3,20**: For browsing other toolkits, their individual availability is measured. Other toolkits will be accessed in real-time.

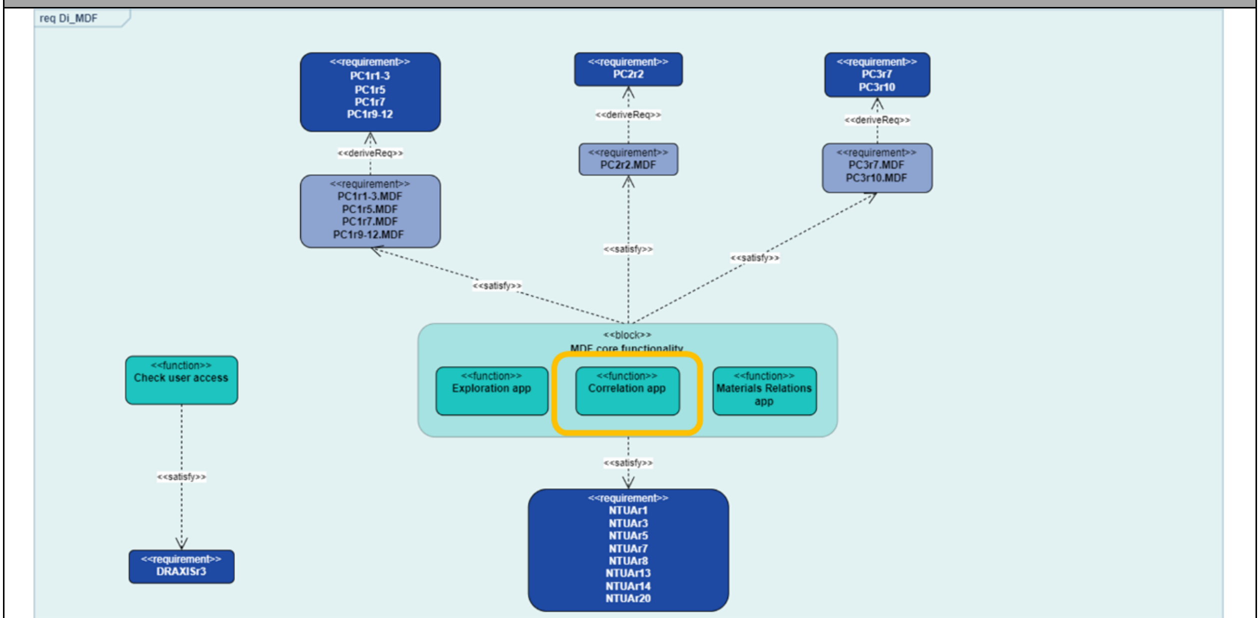
Table 75. Exploration App functional component diagram, structure and requirements

5.5.2.3 Correlation App

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

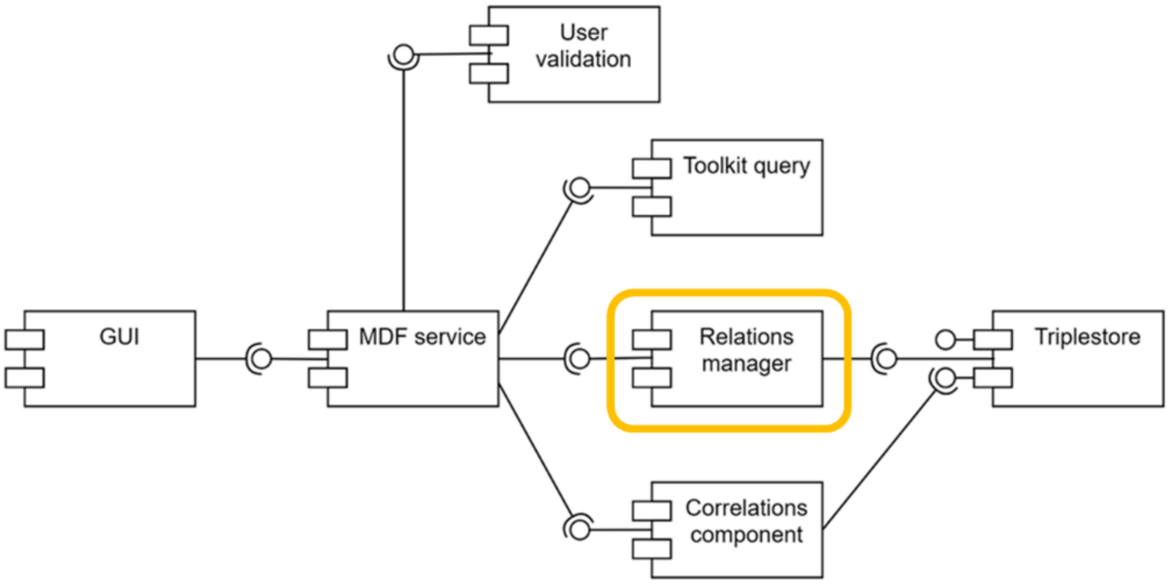


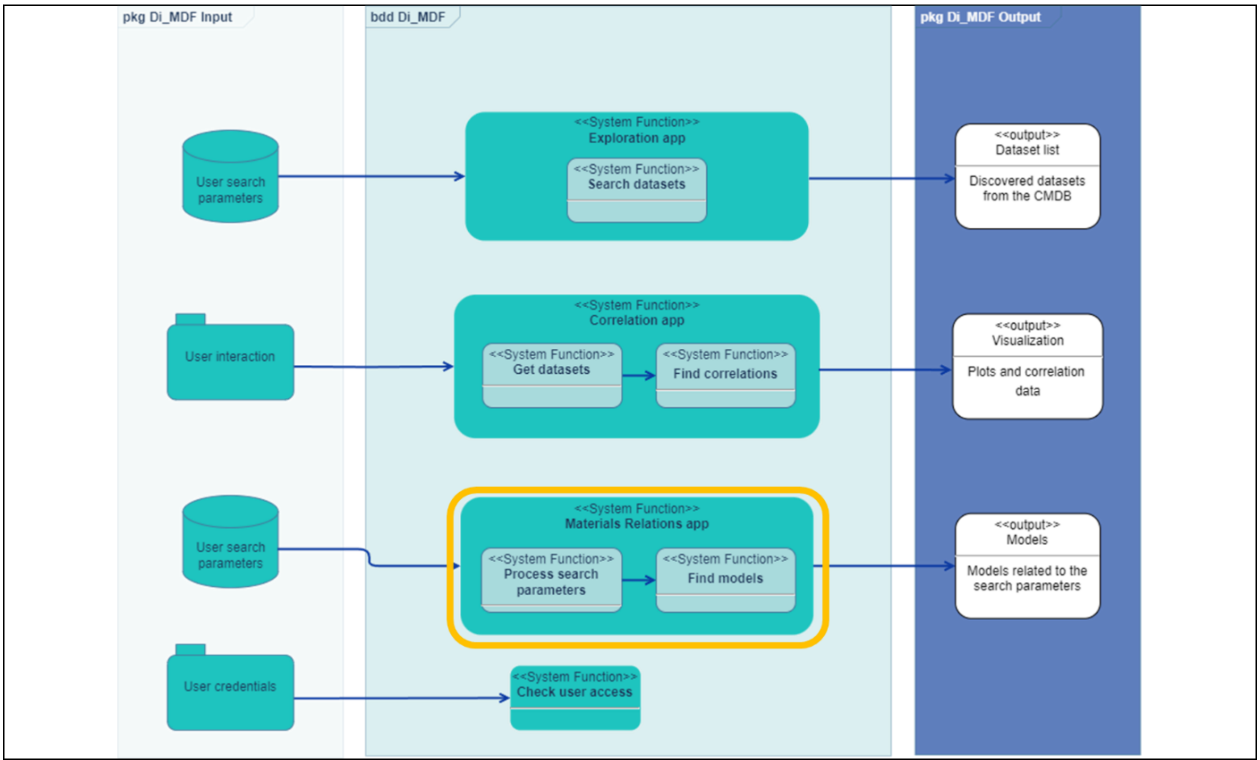
Describe the connection between the functional component and the requirements

- **PC1r1,2,3,12**: The correlation app identified and quantified correlations between available data. It can be used to optimize product performance (mechanical, rheological) based on material and process parameters. - **PC2r2**: Once correlations are quantified, these can be approximated and used for the prediction of mechanical and physical properties of a material based on their structure.

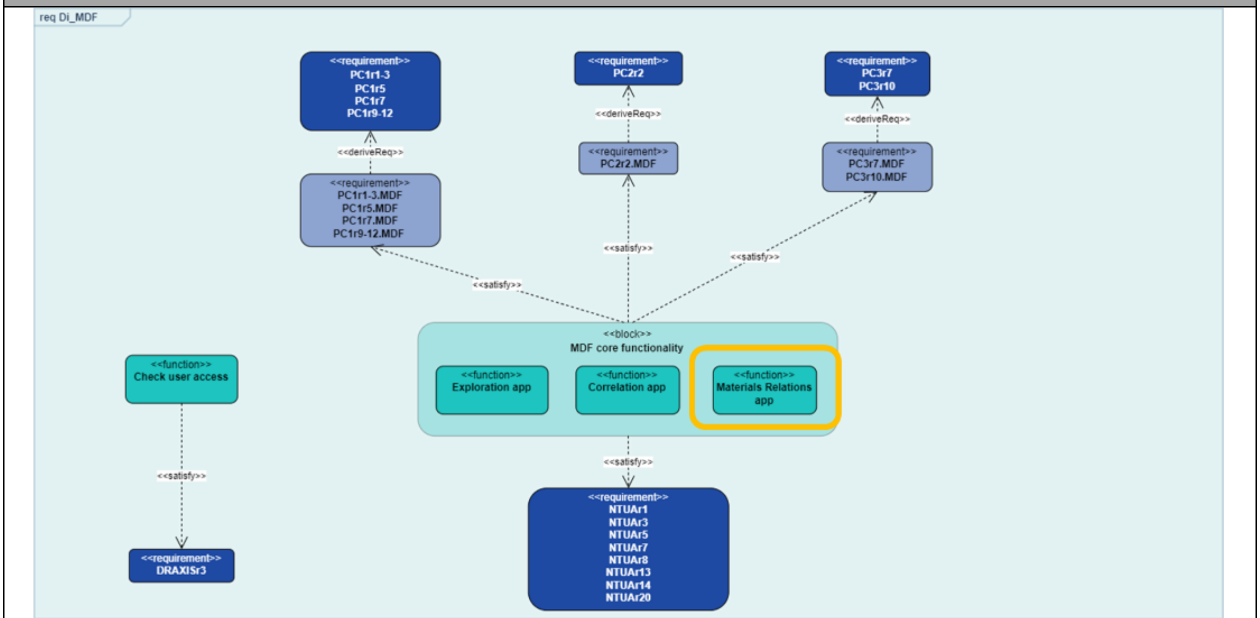
Table 76. Correlation App functional component diagram, structure and requirements

5.5.2.4 Materials Relation App

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

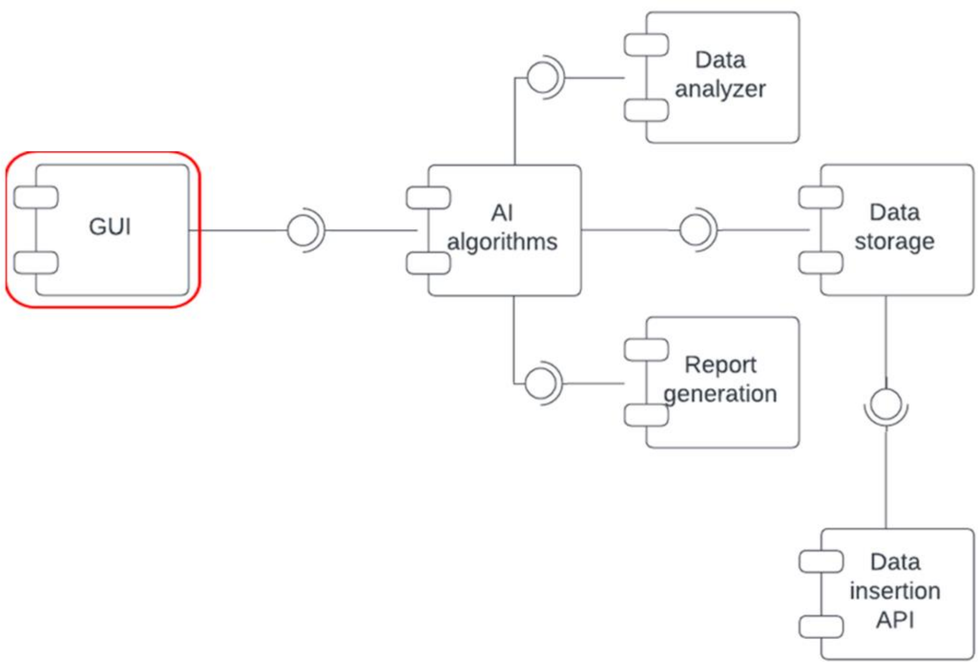
- **PC1r1,5,7,12**: The materials relation app provides information about/ allows the discovery of material behavior, material models and corresponding modeling parameters. - **NTUAr14**: An ontology is underlying, which is built upon existing ontologies.

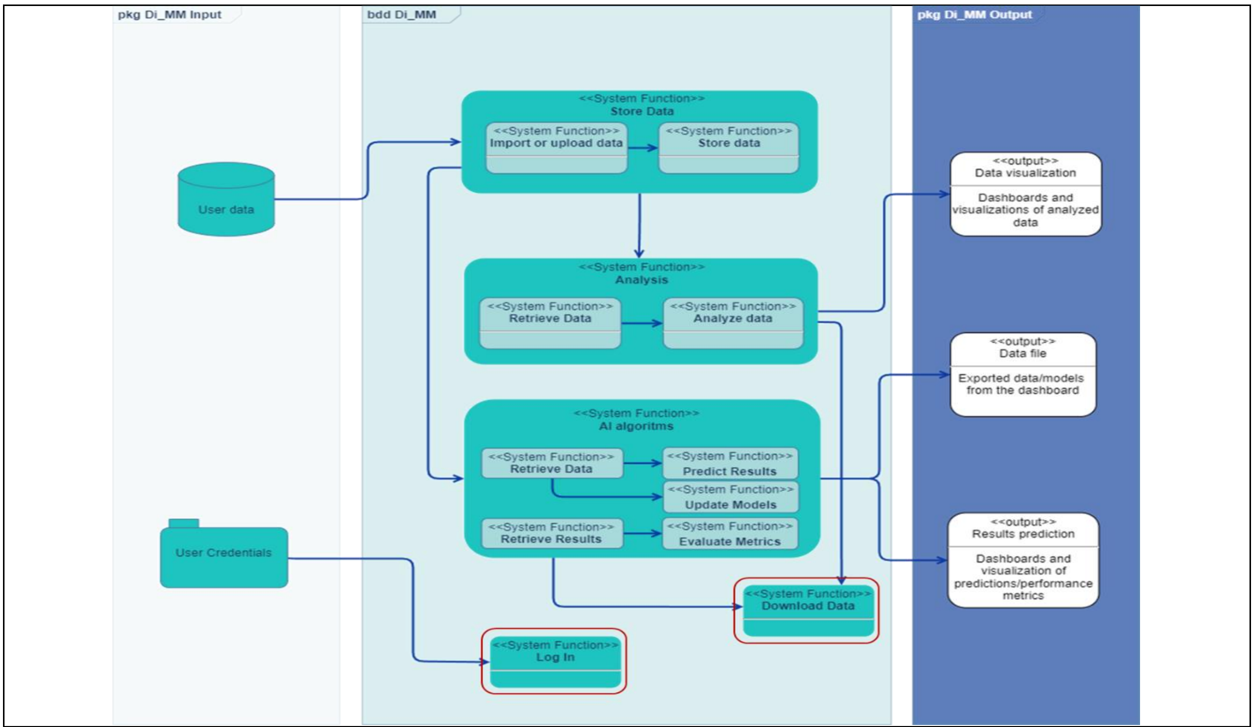
Table 77. Materials Relation App functional component diagram, structure and requirements

5.6 DIMAT MATERIALS MODELER (DI^{MM})

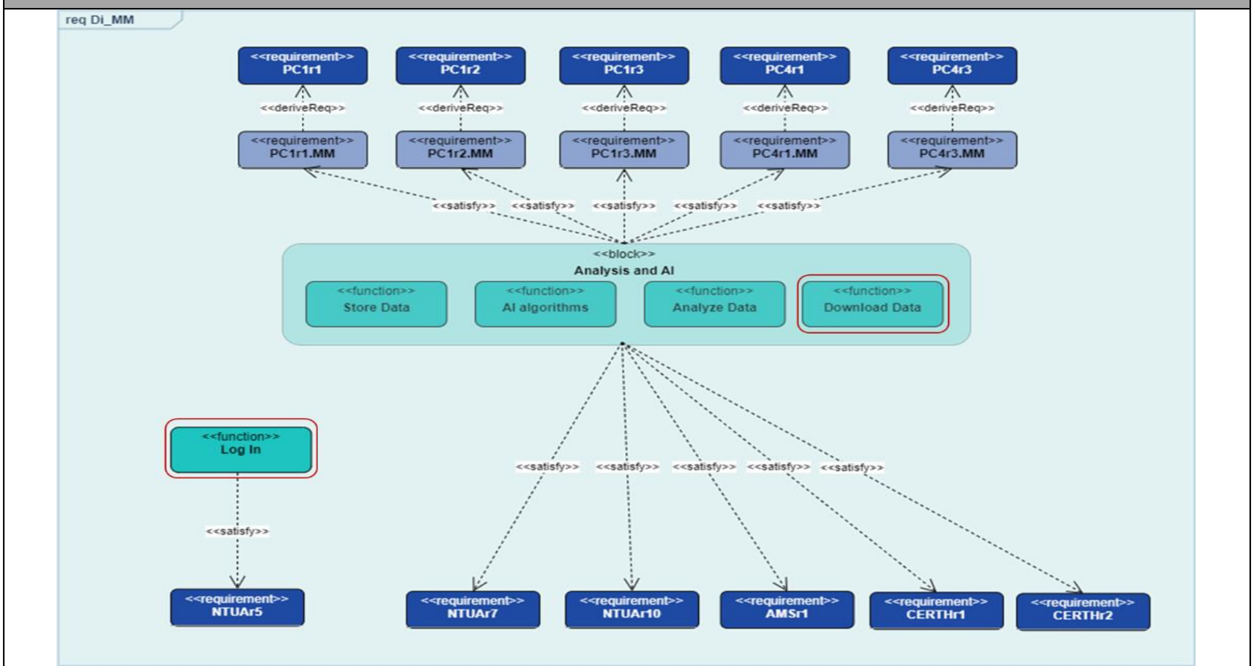
5.6.1 Functional components, functional structure and requirements interactions

5.6.1.1 GUI

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

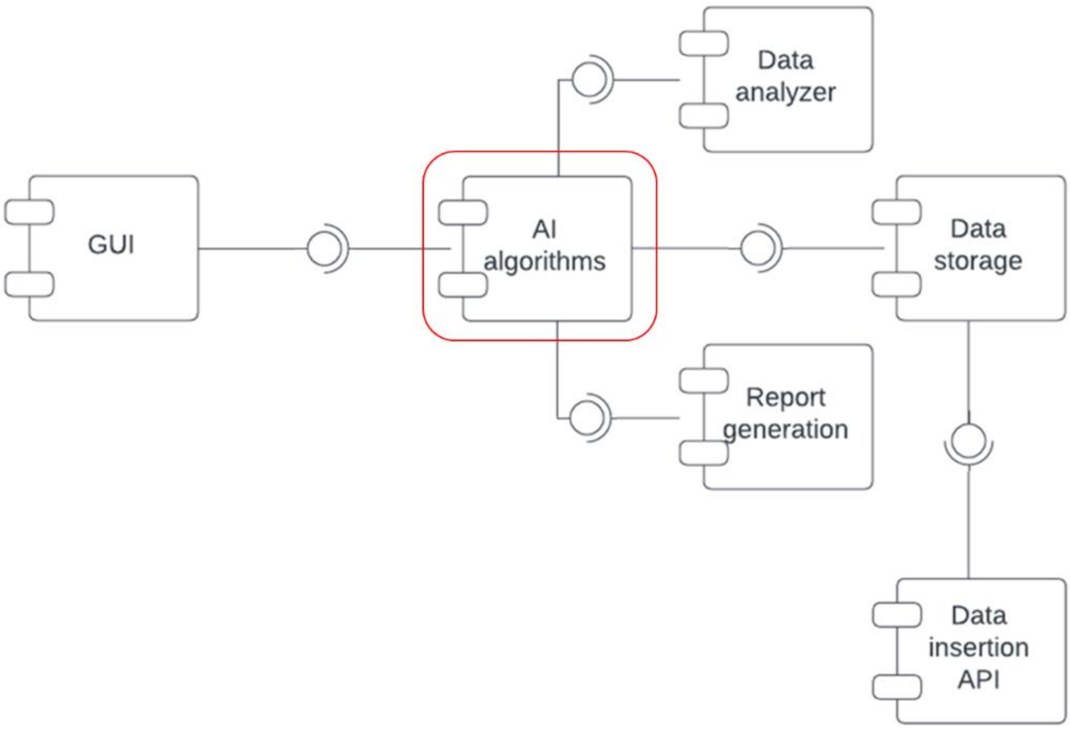


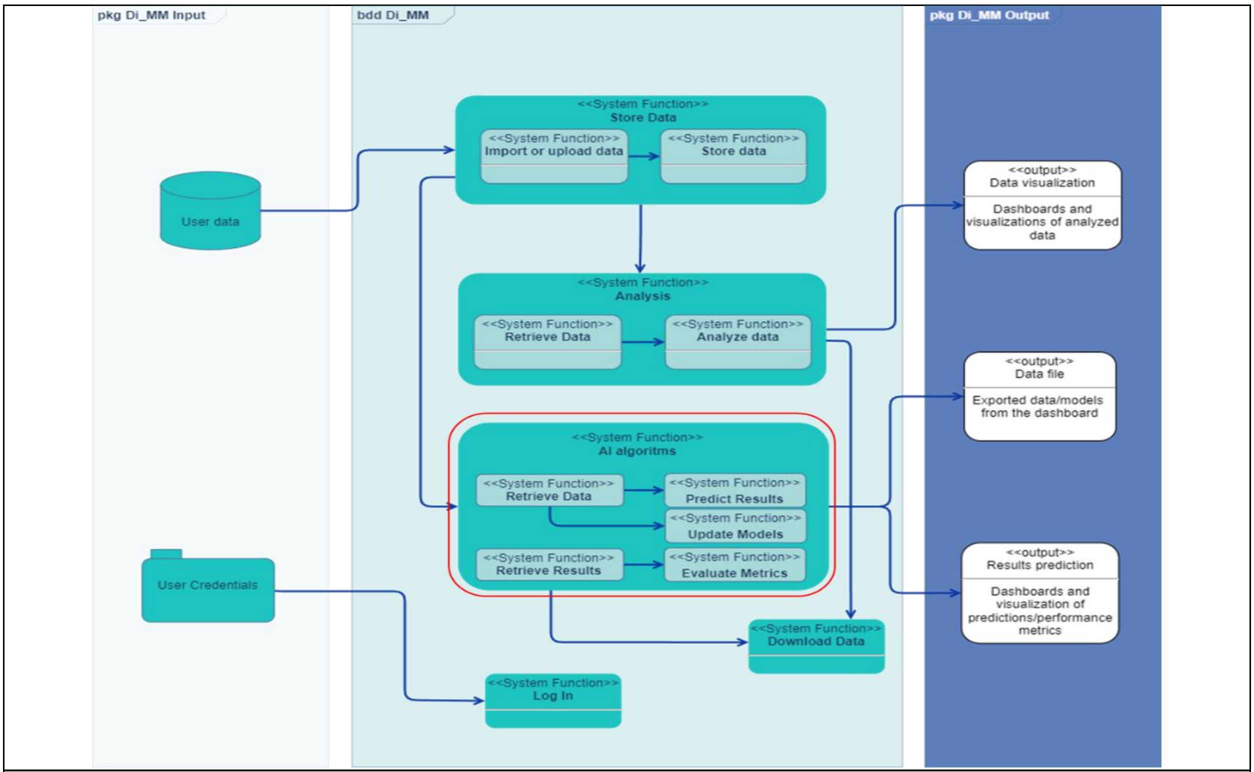
Describe the connection between the functional component and the requirements

- **NTUAr5** (*The toolkit shall be accessible only by authorised users):* Only authorized users shall be able to log in, view and edit the data available through the tool.

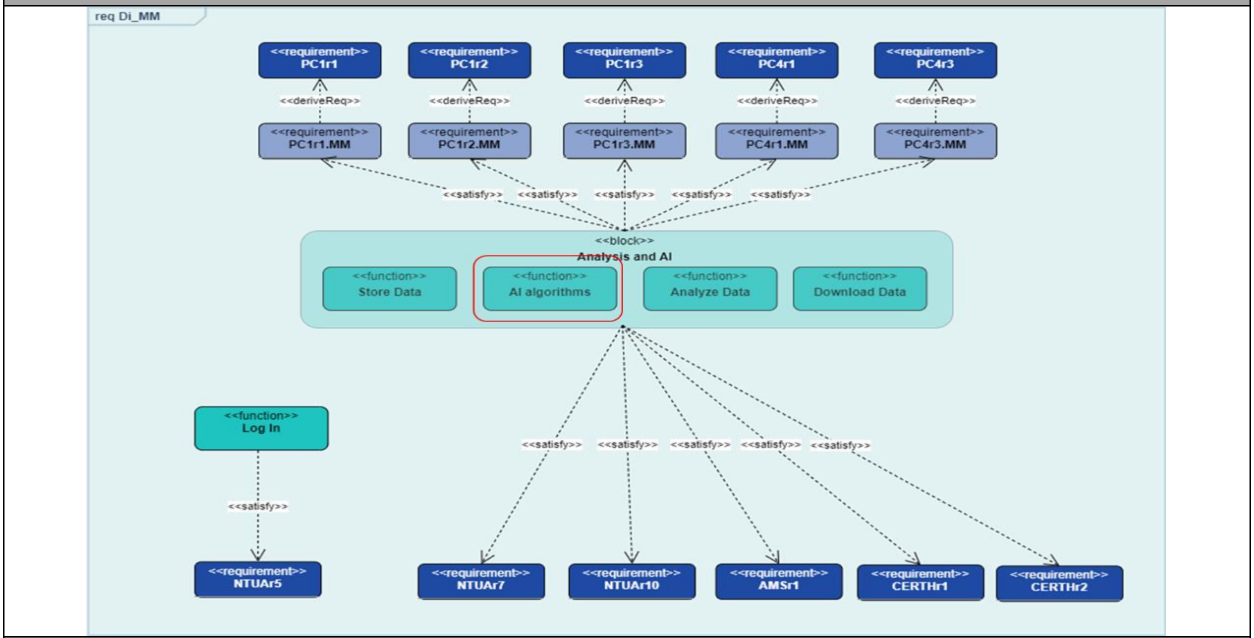
Table 78. GUI functional component diagram, structure and requirements

5.6.1.2 AI Algorithms

Functional component diagram

Functional structure diagram



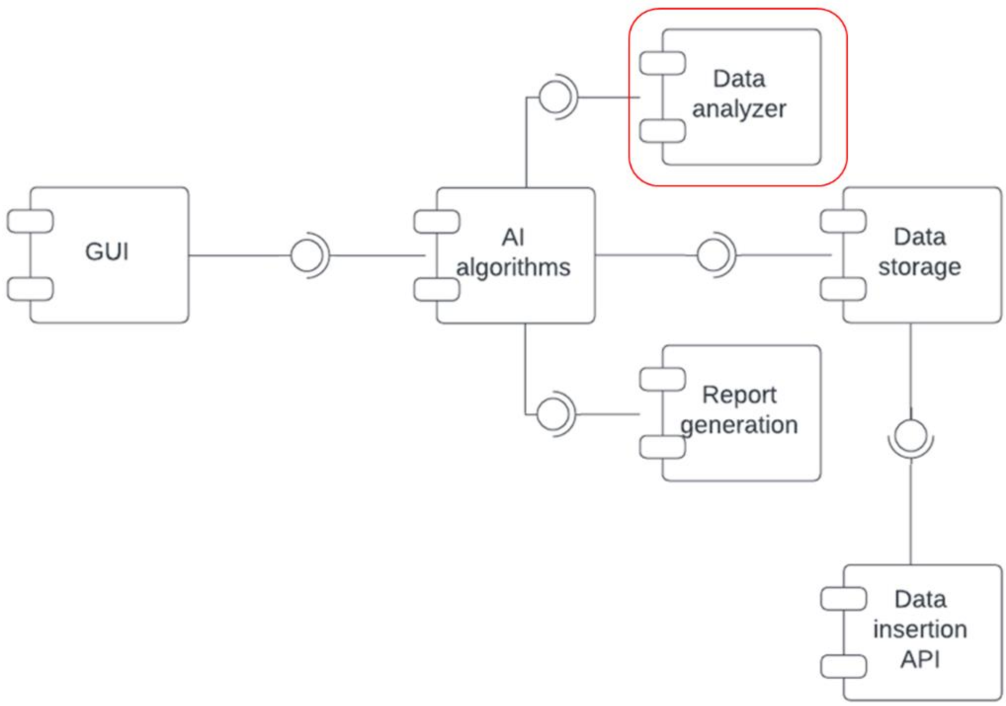
Requirements linked to the functional component

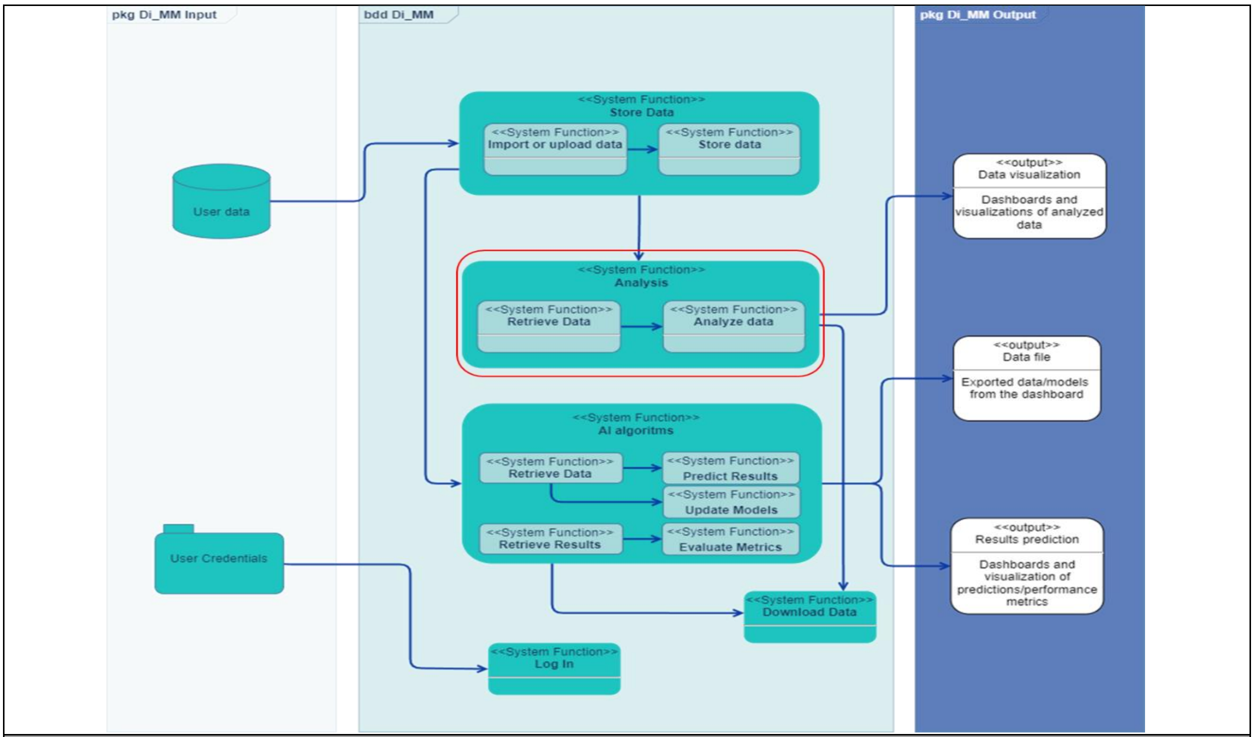


Describe the connection between the functional component and the requirements

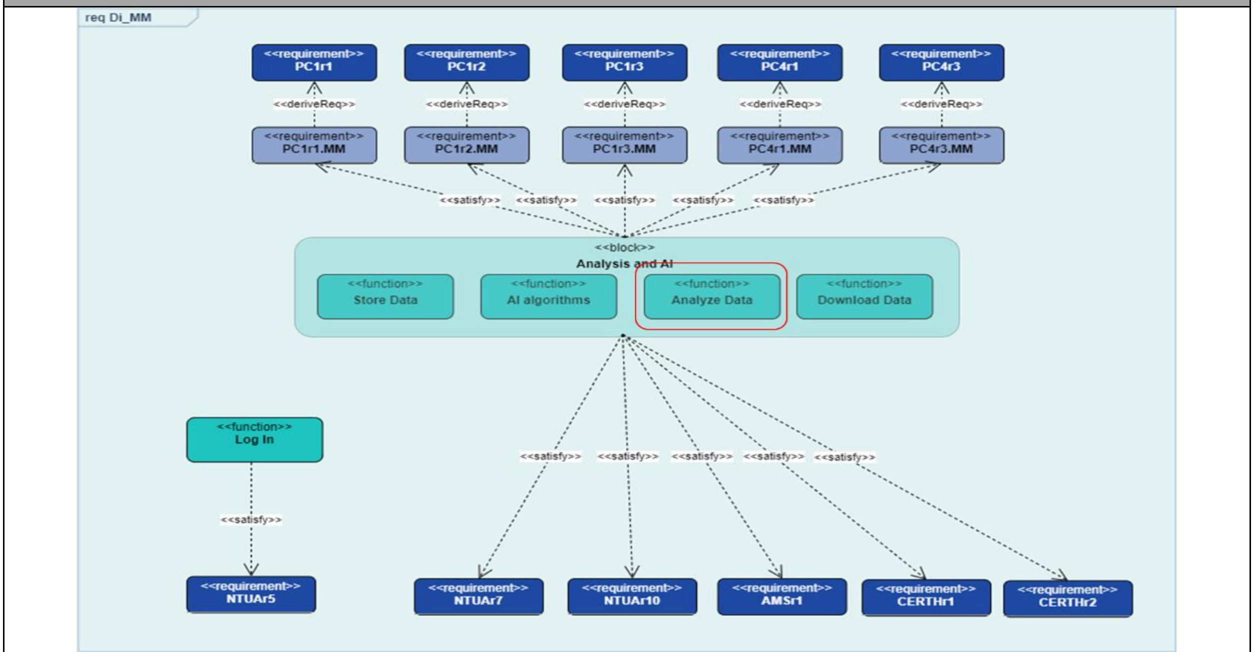
- **NTUAr7:** The toolkit shall provide sufficient documentation and the user will be able to access examples and documentation on how to use the AI Algorithms. - **CERTHr1**: The toolkit shall be able to leverage GPU acceleration for training and inference of machine learning models - **CERTHr2**: The toolkit shall be able to incorporate techniques to avoid overfitting under normal conditions (sufficient data quality and quantity)

Table 79. AI Algorithms functional component diagram, structure and requirements

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

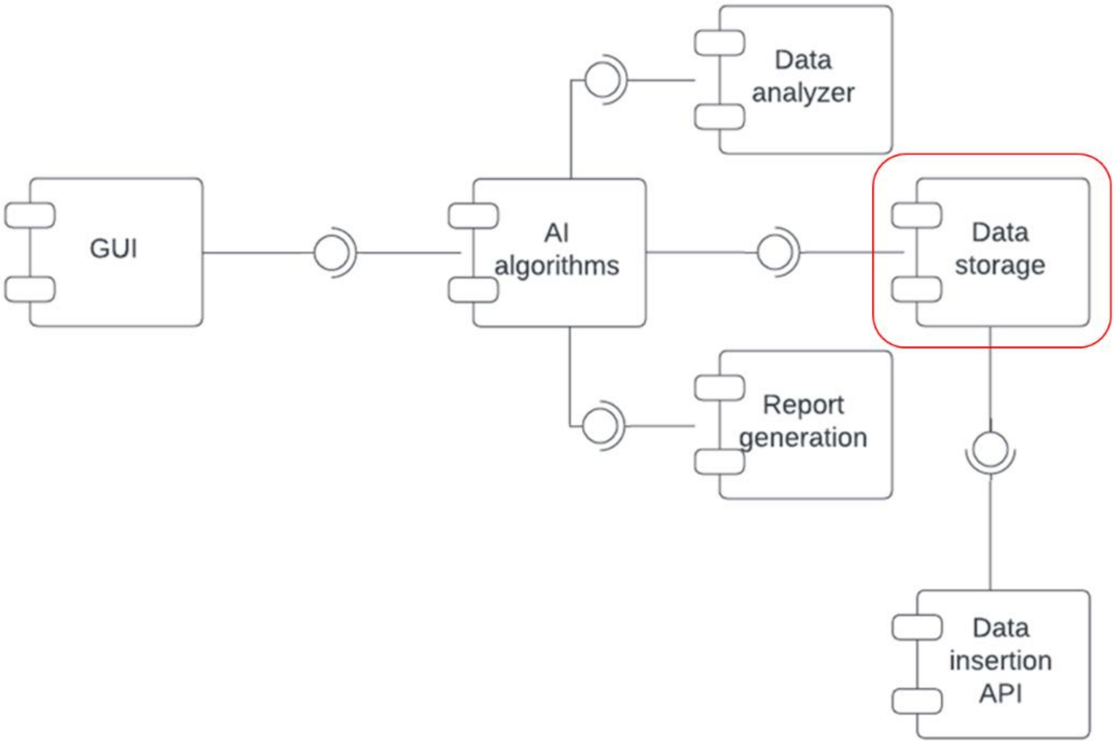


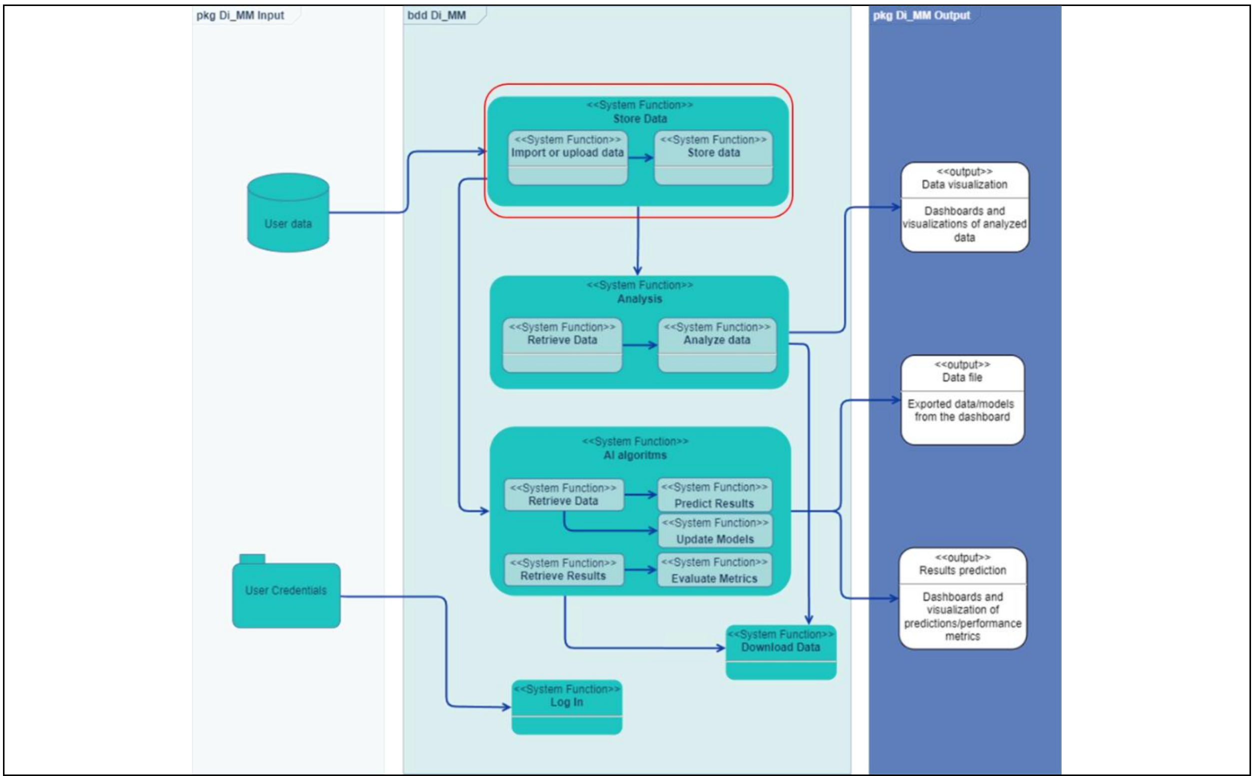
Describe the connection between the functional component and the requirements

NTUAr10: The toolkit shall be able to suggest optimal parameters concerning material processes

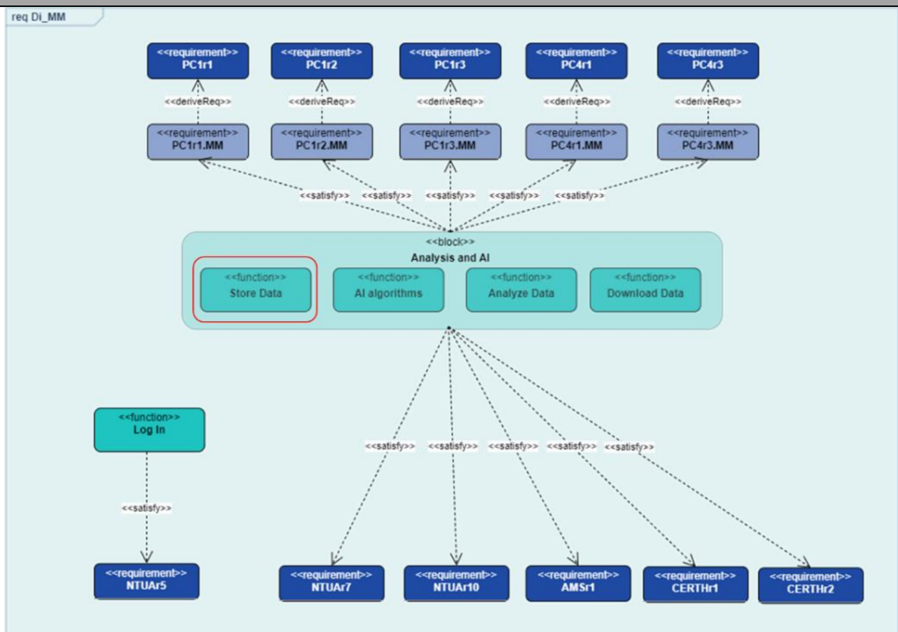
Table 80. Data Analyzer functional component diagram, structure and requirements

5.6.1.3 Data Storage

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

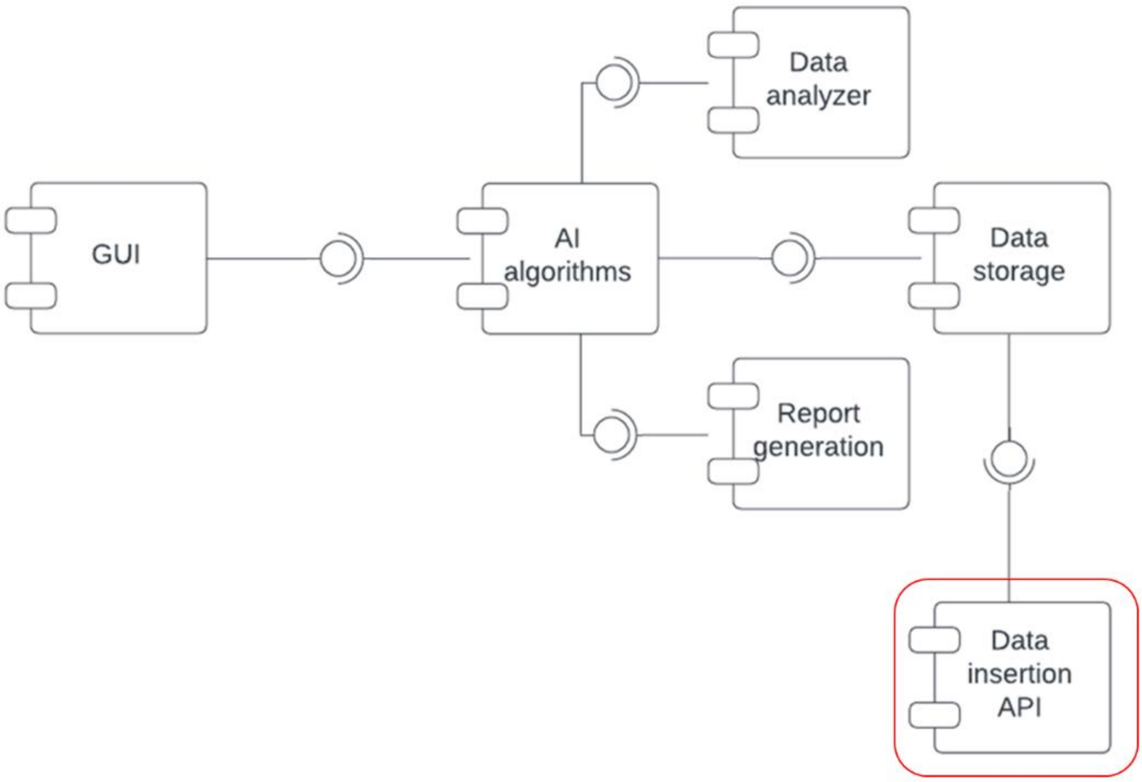


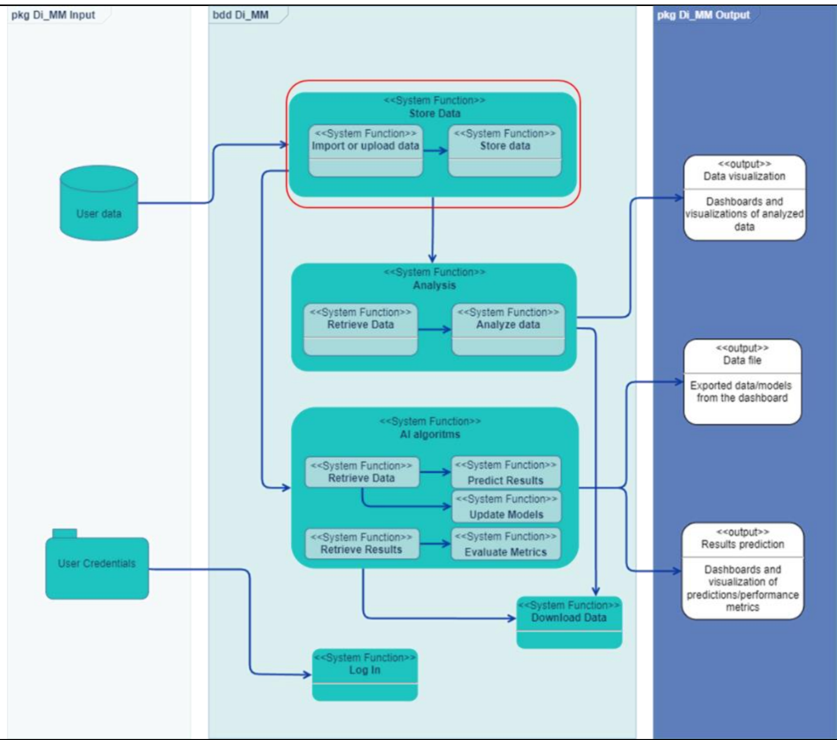
Describe the connection between the functional component and the requirements

- **AMSr1**: (*The toolkit shall check the input data analyzing the format and the range to avoid extrapolation of predictions*): The user will be able to store model output data.

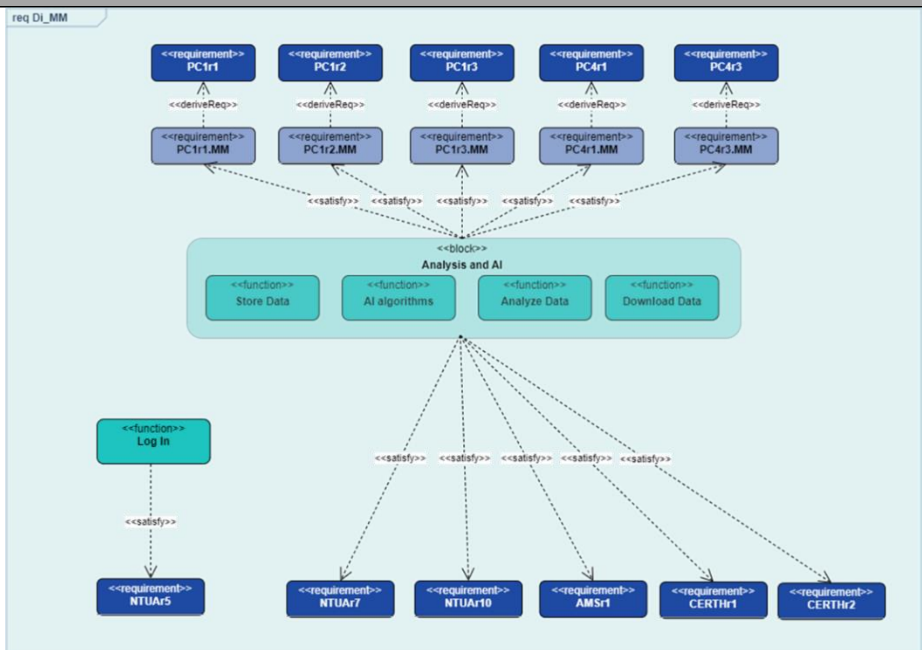
Table 81. Data Storage functional component diagram, structure and requirements

5.6.1.4 Data Insertion

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

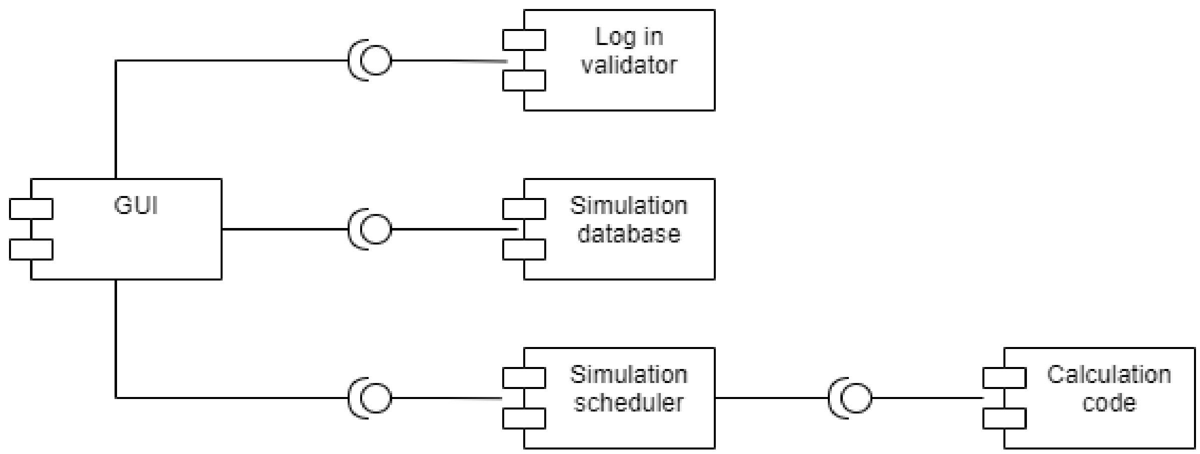
<Describe the interactions, connections and relations between the functional component and each requirement> - **AMSr1**: (*The toolkit shall check the input data analyzing the format and the range to avoid extrapolation of predictions*): The user will be able to store model output data.

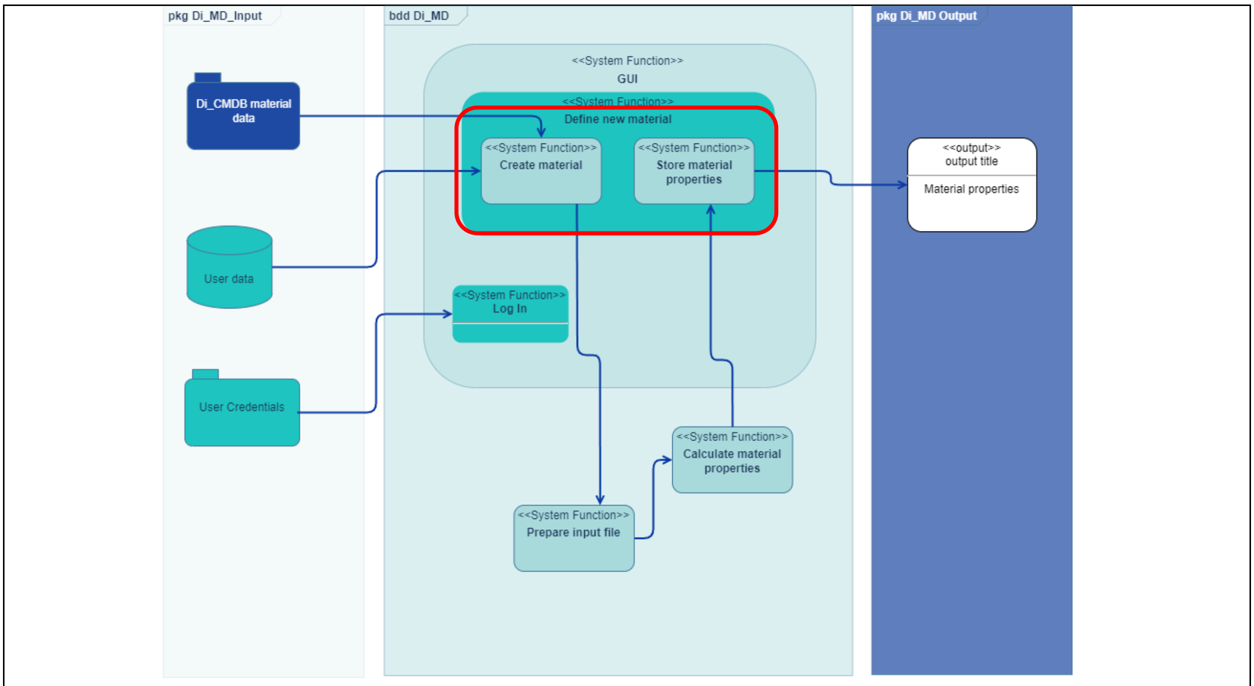
Table 82. Data Insertion functional component diagram, structure and requirements

5.7 DIMAT MATERIALS DESIGNER (DI^{MD})

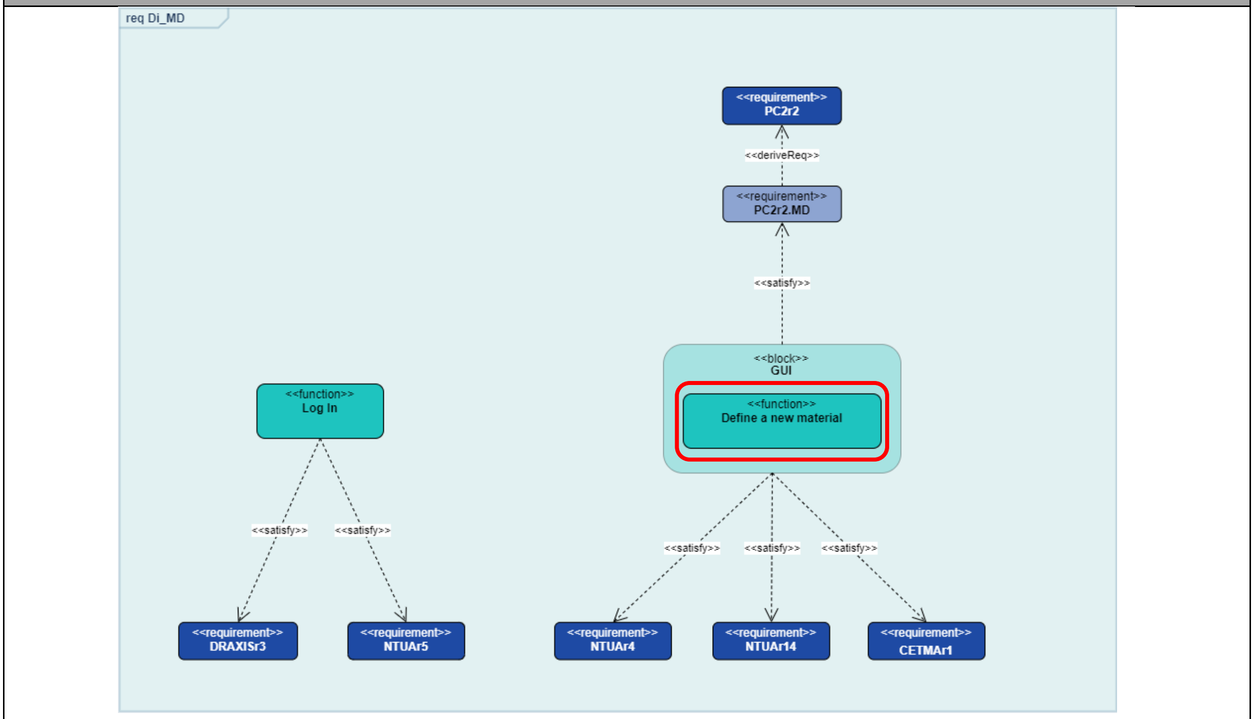
5.7.1 Functional components, functional structure and requirements interactions

5.7.1.1 Define a new material

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

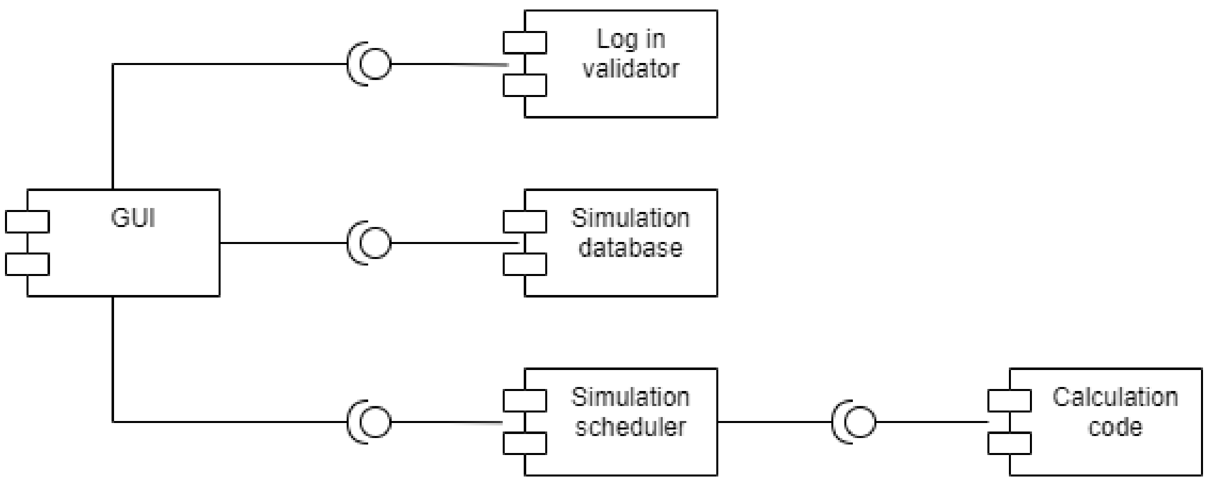


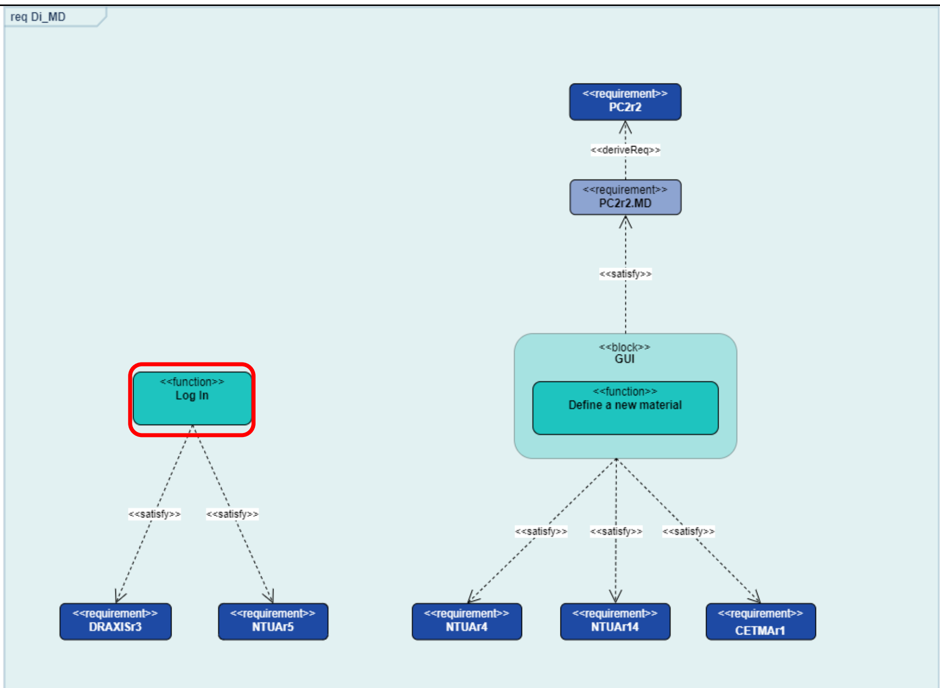
Describe the connection between the functional component and the requirements

- **NTUAr4** (*The toolkit shall offer an intuitive UI*): The user interface allows the user to easily insert, request and visualize data. - **NTUAr14** (*The toolkit shall incorporate at least one (e.g., EMMO) of the most popular material ontologies*): The toolkit is able to exchange data in a standardized way with other toolkits or platforms. - **CETMAr1** (The toolkit shall be able to communicate and exchange data with the calculation code, even if it installed on a different server): The toolkit implements adequate mechanisms to communicate with the calculation code.

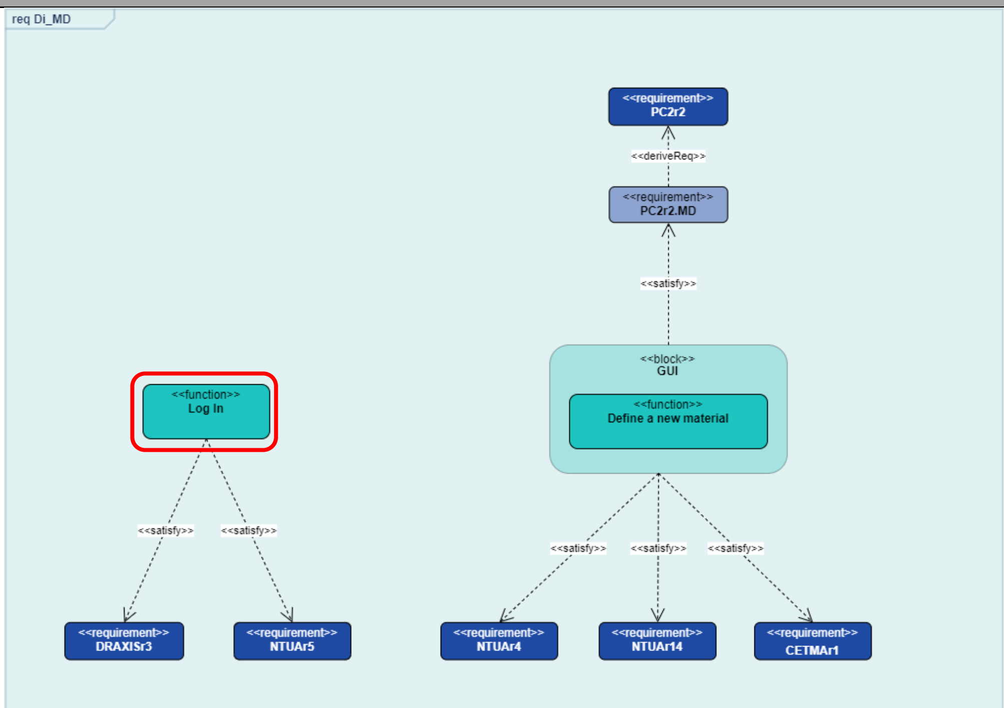
Table 83. Functional component diagram, structure and requirements

5.7.1.2 Log in

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

- DRAXISr3 (The toolkits shall manage user access for every pilot partner): The toolkit is able to guarantee access to the users and give access to data based on their identity. - NTUAr5 (The toolkit shall be accessible only by authorised users): Access to the toolkit functionality and data is granted only to authorized users.

Table 84. Log in functional component diagram, structure and requirements

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

5.8.1 General Architecture

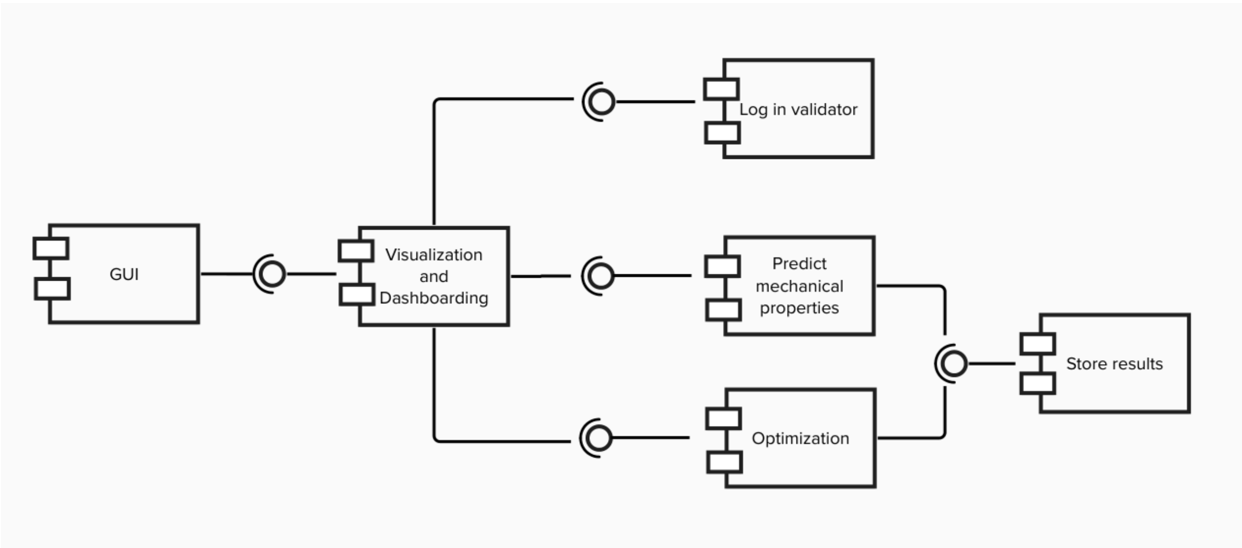


Figure 40: Materials Mechanical Properties Simulator diagram

5.8.2 Functional components

The tables below describe updates to the main functional components identified in the General Architecture of the MMS toolkit. The functional components “Log in Validator”, “Optimization API”, and “Data Storage” remain the same as in version 1.

5.8.2.1 GUI

Name:	*GUI*	
Description:	*The way in which users interact with the Visualization and Dashboarding component.*	
	Interface	
Operation	***Inputs***	***Outputs***
Predict MMS properties	- Microstructure parameters, geometry, and compound properties	- *Mechanical properties* - Stress, strains and displacements fields
Optimize MMS properties	- Target mechanical properties	- Optimized parameter
Log in	- Credentials	- Access is granted to specific resources (according to user's access permissions)
Modify visualization	- Modifications selected	- Visualization modified
Store simulation results	- Data to store	- Data file

Table 85. GUI of the functional component of the DiMMS

5.8.2.2 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***
Predict MMS properties	- *Microstructure parameters and compound properties*	- *Mechanical properties* - *Stress, strains and displacements fields*
Optimize MMS properties	- *Target mechanical properties*	- *Optimized parameters*
Log in	- *Credentials*	- *Access is granted to specific resources (according to user's access permissions)*
Modify visualization	- *Modifications selected*	- *Visualization modified*
Store simulation results	- *Data to store*	- *Data file*

Table 86. Visualization and dashboarding of the functional component of the DiMMS

Co-funded by the European Union

5.8.2.3 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* - Stress, strains and displacements fields
Modify visualization	- *Modifications selected*	- *Visualization modified*
Store simulation results	- Data to store	- Data file

Table 87. Predict Mechanical Properties API of the functional component of the DiMMS toolkit

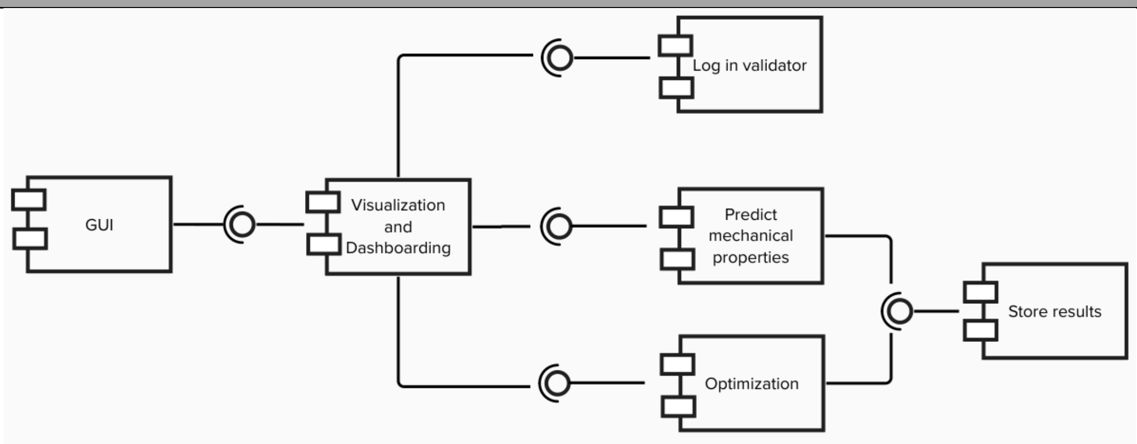
5.8.3 Interactions

The interactions "Log-in", "Predict Mechanical Properties", "Optimization", and "Store results" remain the same as in version 1.

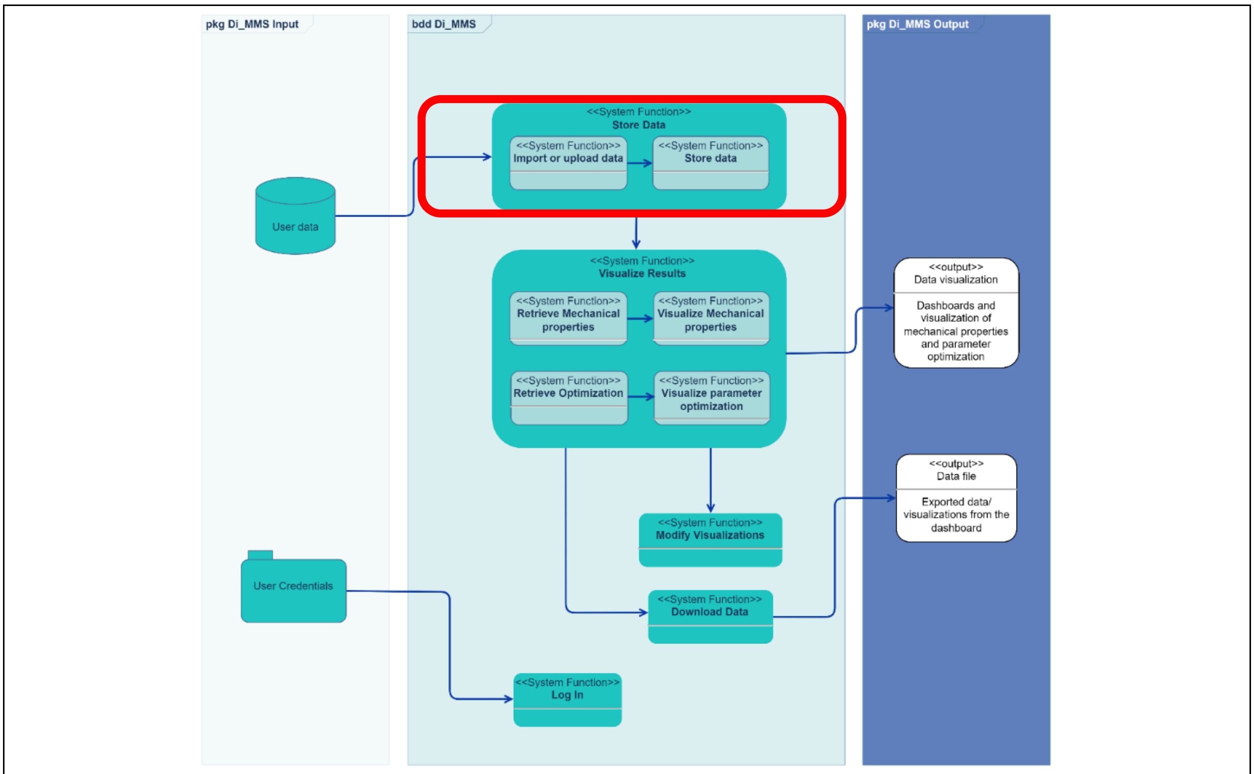
5.8.4 Functional components, functional structure and requirements interactions

5.8.4.1 Store data

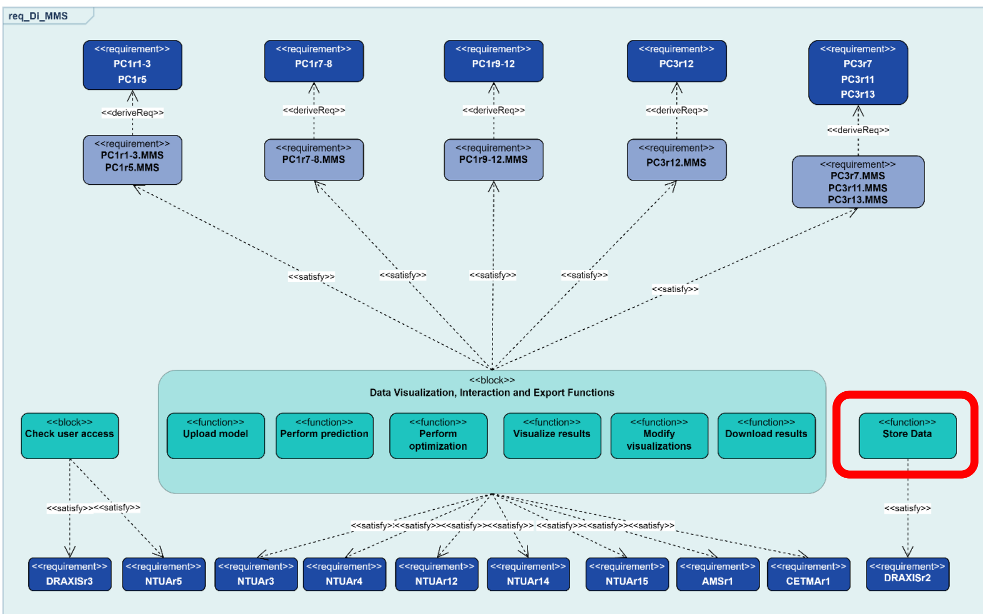
Functional component diagram



Functional structure diagram



Requirements linked to the functional component

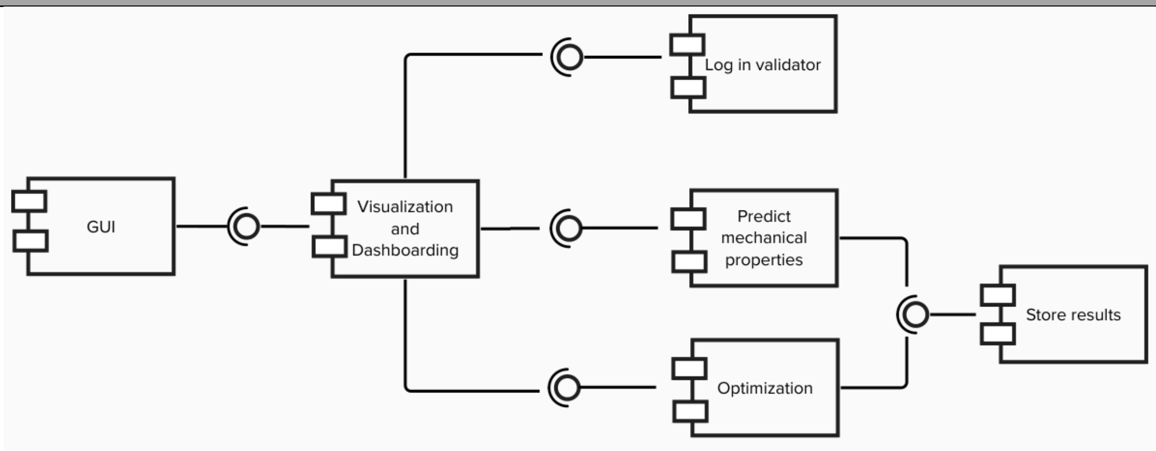


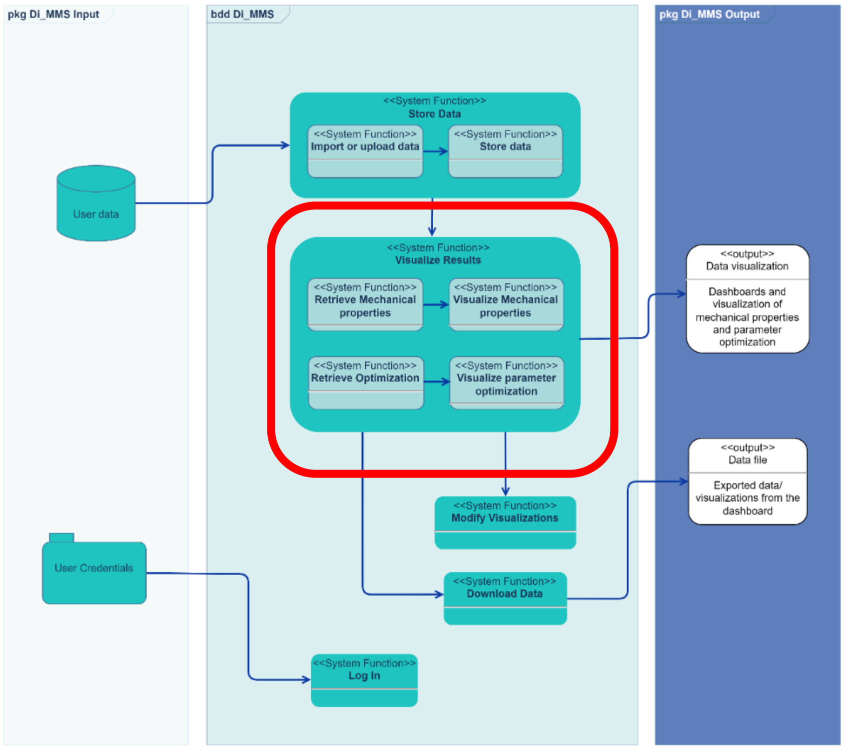
Describe the connection between the functional component and the requirements

- **DRAXISr2** (*Data shall be collected from the pilots and maintained to provide the required outputs*): Data collected in modelling construction, T6.1, and model validation, T7.1 and T7.3, will be appropriately organized and maintained.

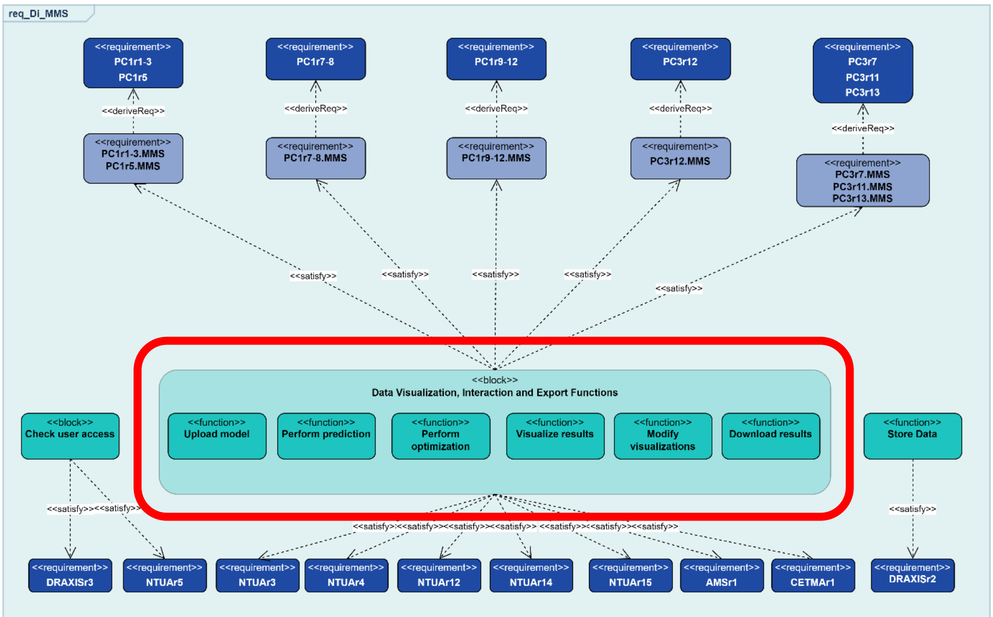
Table 88. Store data functional component diagram, structure and requirements

5.8.4.2 Visualization and dashboarding

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

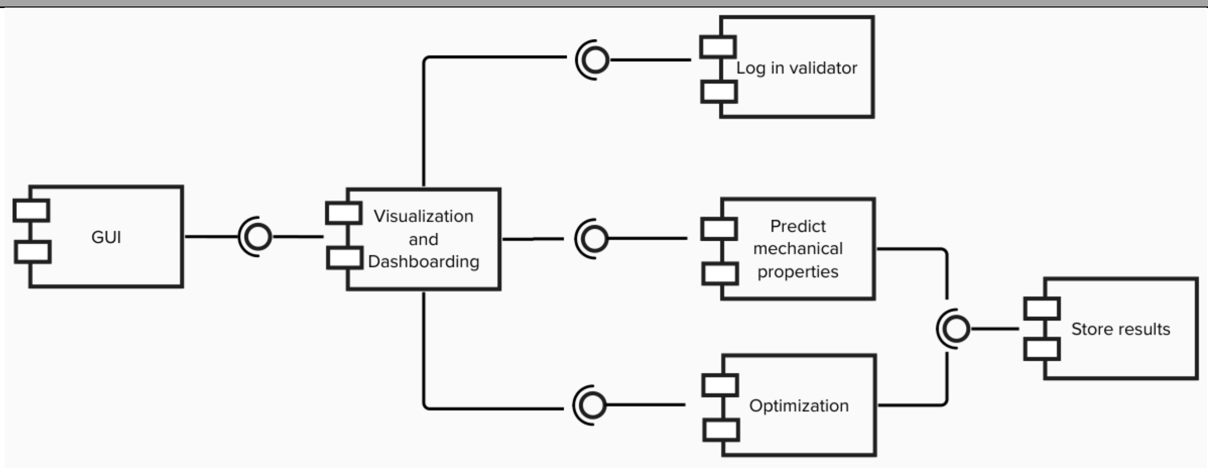


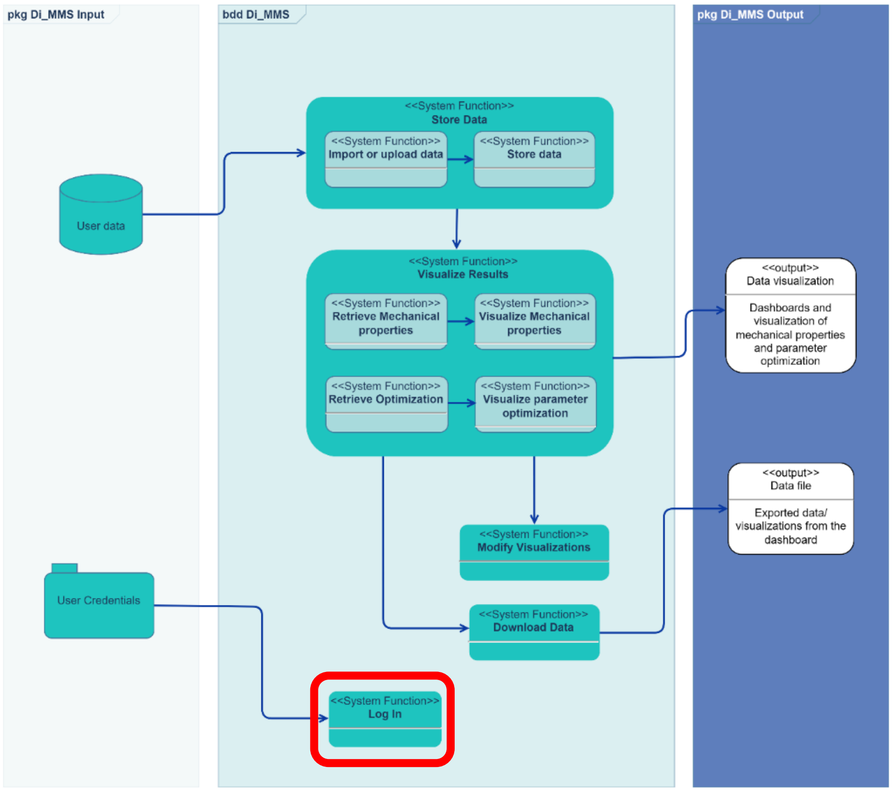
Describe the connection between the functional component and the requirements

- **NTUAr3** (*The toolkit shall be able to connect with at least the 2 other toolkits of the corresponding DiMAT suite*): MMS output can be used in MPS, as mechanical properties in process modelling, and DTPC, as part of the digital twin. - **NTUAr4** (*The toolkit shall offer an intuitive UI*): MMS will visualize stress, strain or displacement field interactively. The user can access, visualize, and export data. - **NTUAr12** *(The toolkit shall support data formatted in at least 3 different formats):* MMS toolkit combines molecular dynamic and finite element output (frd, txt, csv) with reduced models (csv, json). - **NTUAr13** *(The toolkit shall be appropriately updated when necessary):* Physical and reduced models will be appropriately updated when necessary. - **NTUAr15** (*The digital twin shall be able to communicate with the physical counterpart*): The MMS output could be used as a virtual physical model in a digital twin. - **AMSr1** *(The toolkit shall check the input data analysing the format and the range to avoid extrapolation of predictions)*: The toolkit also examines the range of input data values. It verifies whether the input data falls within the permissible range defined for each parameter. - **CETMAr1** *(The toolkit shall be able to communicate and exchange data with the calculation code, even if it installed on a different server)*: The format output of MMS toolkit will be the same than other DiMAT toolkits as MPS or DTPC to ensure the interoperability.

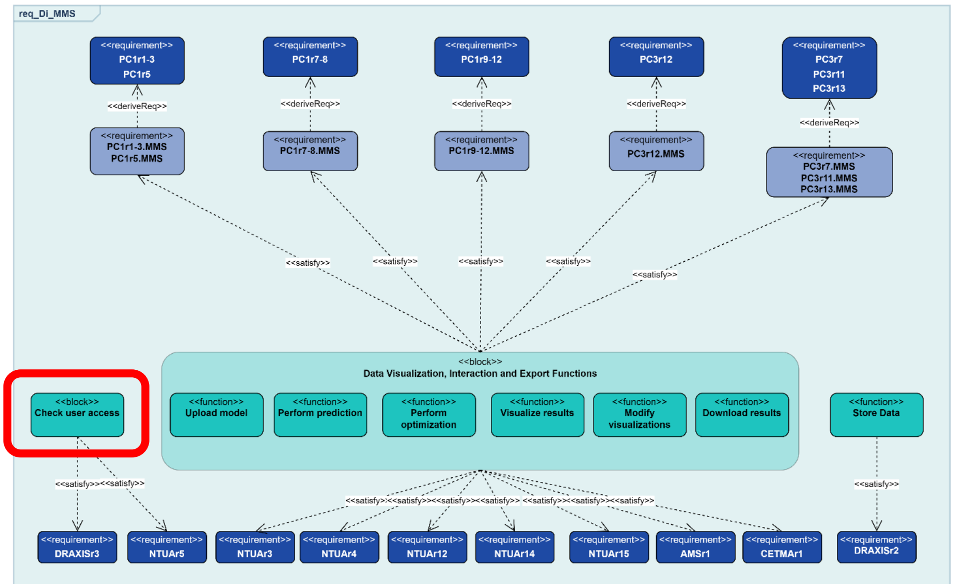
Table 89. Visualization and dashboarding functional component diagram, structure and requirements

5.8.4.3 Log in

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

- **DRAXISr3** *(The toolkits shall manage user access for every pilot partner)*: Upon logging in, users will gain access to specific information based on their respective permissions. - **NTUAr5** (*The toolkit shall be accessible only by authorised users*): Access to the toolkit will be restricted solely to authorized users, who will have the ability to log in, view, and edit the available data.

Table 90. Log in functional component diagram, structure and requirements

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

5.9.1 General Architecture

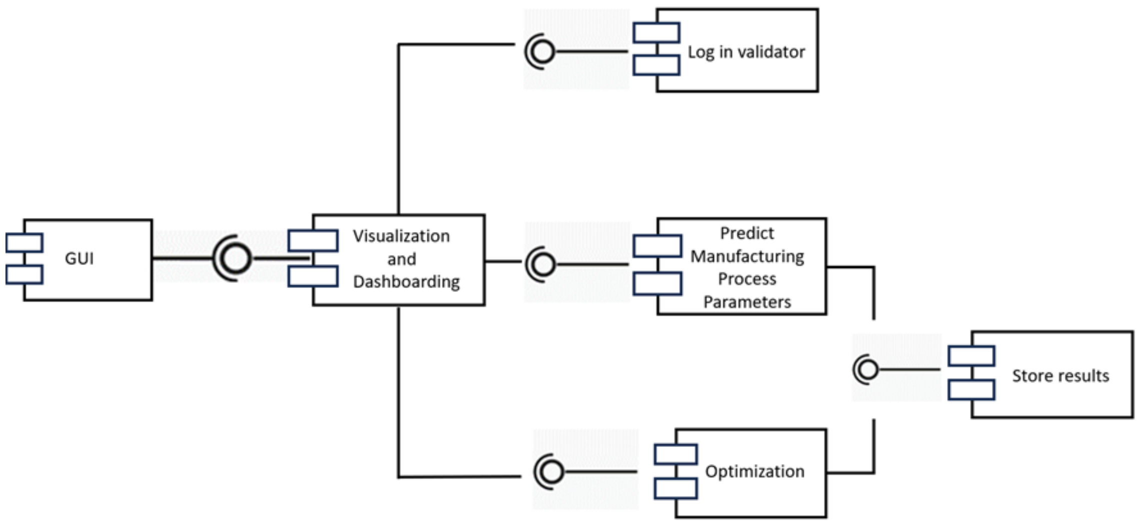


Figure 41. Materials Processing Simulator Diagram

5.9.2 Functional components

The tables below describe updates to the main functional components identified in the General Architecture of the MPS toolkit. The functional components “Log in Validator”, “Predict and Optimization Process” and “Data Storage” remain the same as in version 1.

5.9.2.1 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 91. GUI of the functional component of the DiMpS

5.9.2.2 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 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 92. Visualization and dashboarding of the functional component of the DiMpS

5.9.2.3 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 93. Log in validator of the functional component of the DiMpS

5.9.2.4 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 MPS parameters	- Materials properties and boundary conditions of manufacturing process	- Manufacturing process parameters and materials behaviour
Store simulation results	- Data to store	- Data file

Table 94. Predict Manufacturing Process Parameters of the functional component of the DiMpS

5.9.2.5 Optimization results

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

Table 95. Optimization results of the functional component of the DiMpS

5.9.2.6 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 96. Data Storage of the functional component of the DiMpS

5.9.3 Interactions

The sequence diagrams below describe updates to the main interactions identified for the MPS toolkit. The interactions “Log in”, “Predict and Optimization Process” and “Data Storage” remain the same as in version 1.

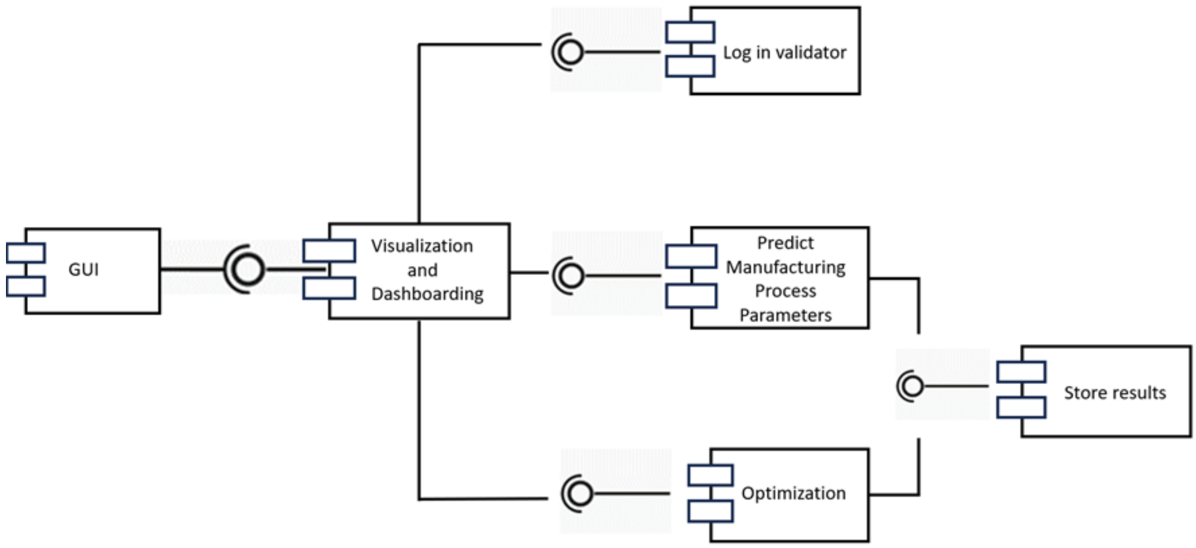
5.9.4 Mapping of functional components to IIRA functional domain

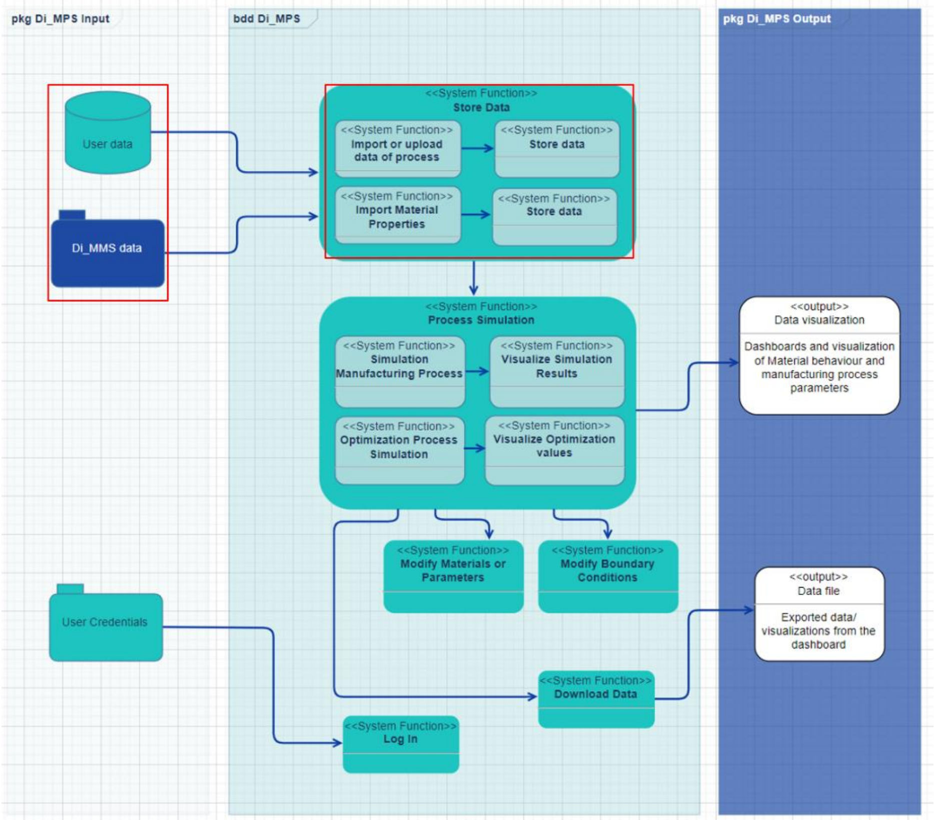
Component	***Domain***	***Subdomain***
GUI	*Application*	*UI*
Visualization and Dashboarding	*Application*	*Logic & Rules*
Log in validator	*Application*	*Logic & Rules*
Predict Manufacturing Process Parameters	*Application*	*API + Logic & Rules*
Optimization	*Application*	*API + Logic & Rules*
Store results	*Information*	*Data*

Table 97. Mapping of functional components to IIRA DiMPS

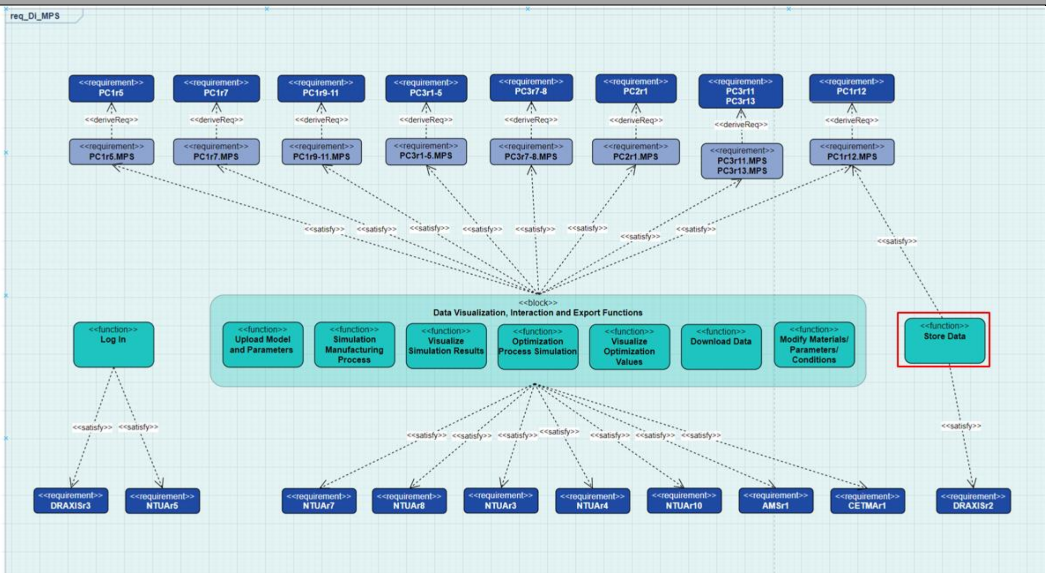
5.9.5 Functional components, functional structure and requirements interactions

5.9.5.1 Store data

Functional component diagram

Functional structure diagram



Requirements linked to the functional component

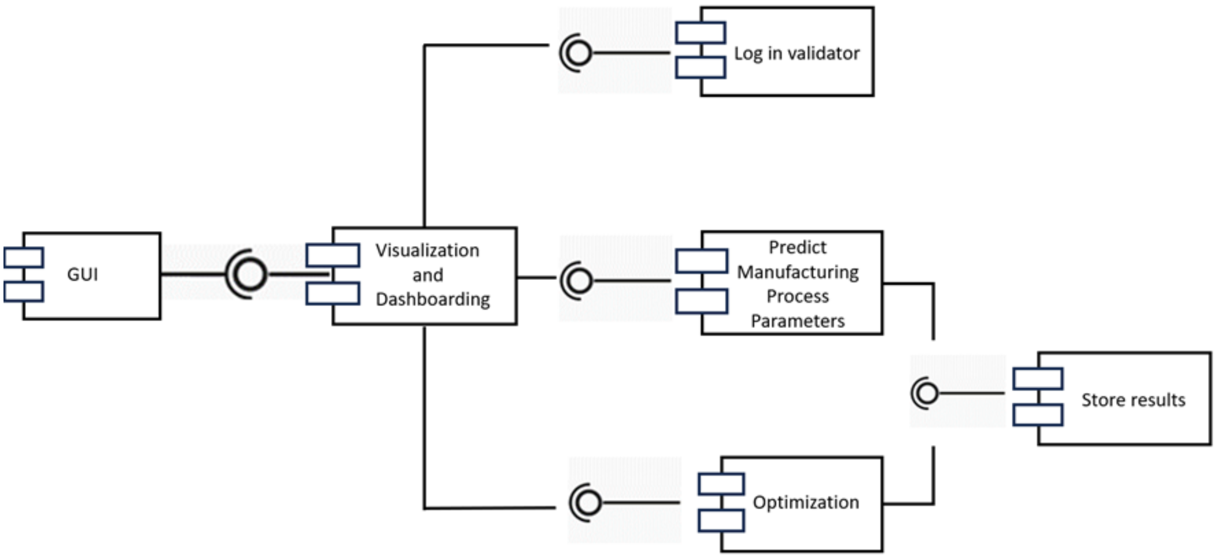


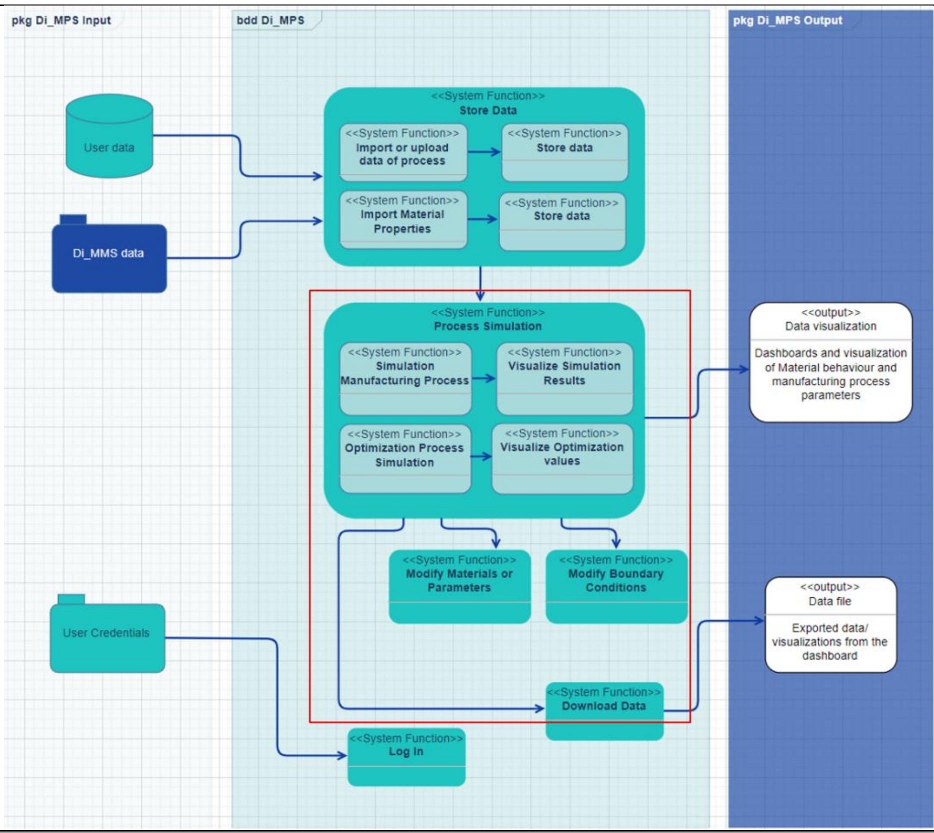
Describe the connection between the functional component and the requirements

- DRAXISr2 (Data shall be collected from the pilots and maintained to provide the required outputs, as well as properties materials collected from DiMAT toolkit MMS for develop properly simulation material process).

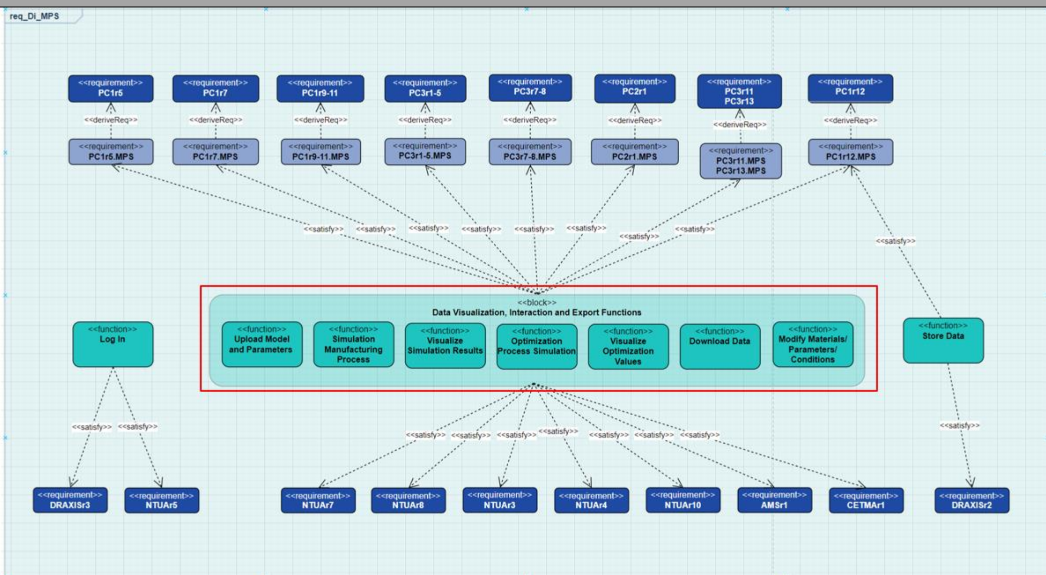
Table 98. Store data functional component diagram, structure and requirements

5.9.5.2 Visualize Results & Modify Visualizations & Download Data

Functional component diagram

Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

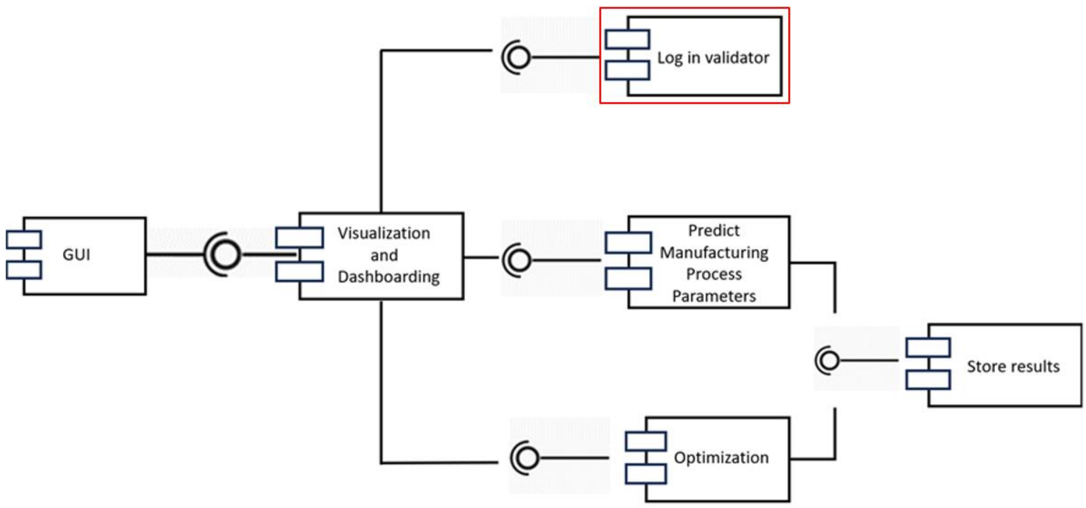
- NTUAr7: (The toolkit shall provide sufficient documentation): Visual representation of defined process geometry, interact with inputs data and selects the output needs shall be effectively communicated and documented. Supplementary materials, such as demos, tutorials and project deliverables will provide further guidance for users. - NTUAr8: (The toolkits shall be able to connect with at least the other toolkits of DiMAT suite) The toolkit will provide secure communication with at least the other toolkits of the suite over channels supporting widely used protocols. - NTUAr3: (The toolkits shall be able to connect with at least the other toolkits of DiMAT suite) MMS output can be used like input for MPS to develop more accurate simulation parameters, and DTPC to interact with manufacturing processes and obtaining more realistic simulation. - NTUAr4: (The toolkit shall offer an intuitive UI): MPS will visualize the geometry which defines the manufacturing process, also the different kind of results demands depending on the material transformation process. The user can access, visualize, import, and export data. - NTUAr10: (The toolkit shall be able to suggest optimal parameters concerning materials processes) MPS will suggest to users an optimization of inputs parameters to obtain more accurate data, with the aim of optimizing manufacturing processes. - AMSr1: (The toolkit shall check the input data analysing the format and the range to avoid extrapolation of predictions): The users will be able to store data and create a database to suggest in future simulation to verify properly input. - CETMAr1: (The toolkit shall be able to communicate and exchange data with the calculation code, even if it installed on a different server): The toolkit implements adequate mechanisms to communicate with the rest of the toolkits, especially with MMS and DTPC.

Table 99. Visualiza Results & Modify Visualizations & Download Data functional component diagram, structure and requirements

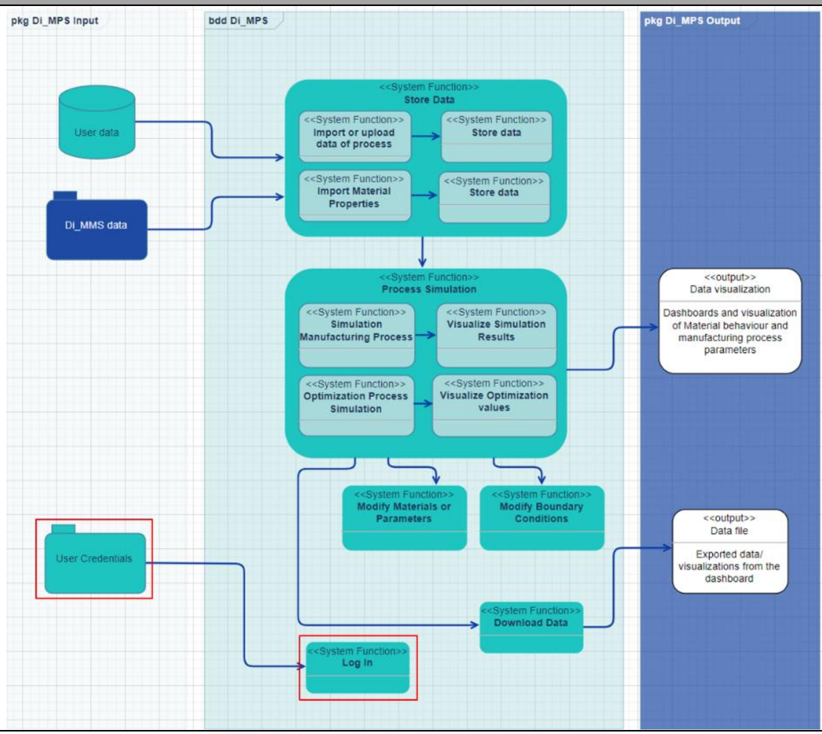
Co-funded by the European Union

5.9.5.3 Log in

Functional component diagram



Functional structure diagram



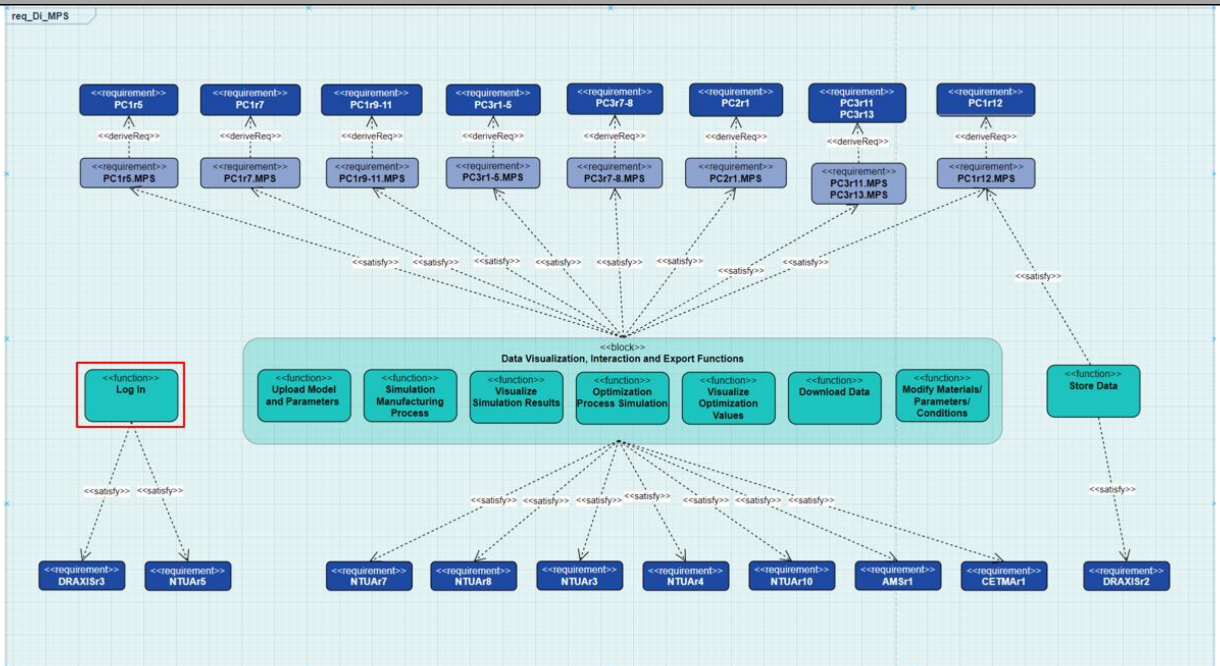
Requirements linked to the functional component

Describe the connection between the functional component and the requirements
- DRAXISr3 (The toolkits shall manage user access for every pilot partner): Upon log in, users shall be able to access specific information, according to their rights. - NTUAr5 (The toolkits shall be accessible only by authorised users): Only authorized users shall be able to log in, view and edit the data available through the tool.

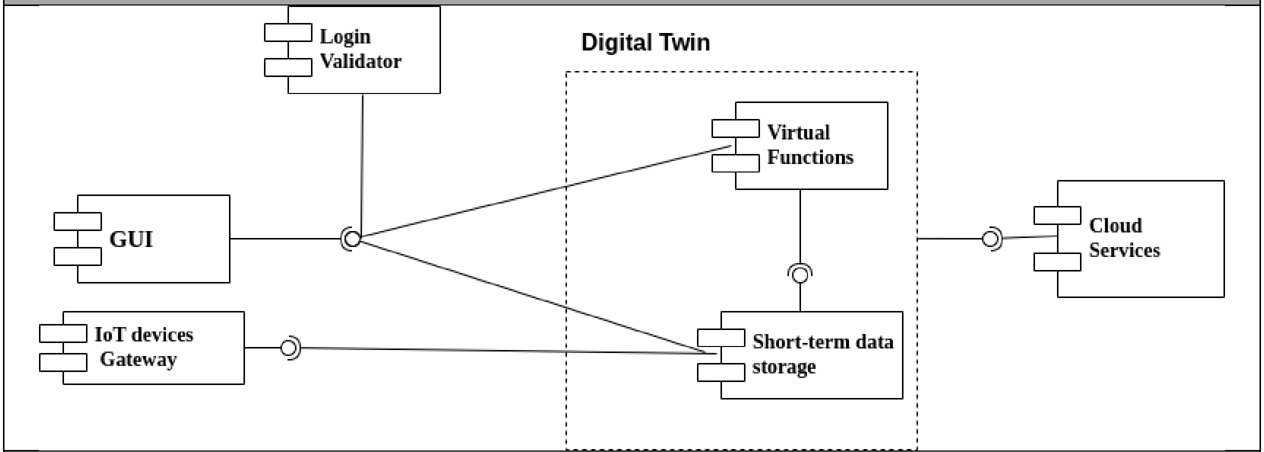
Table 100. Log in functional component diagram, structure and requirements

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

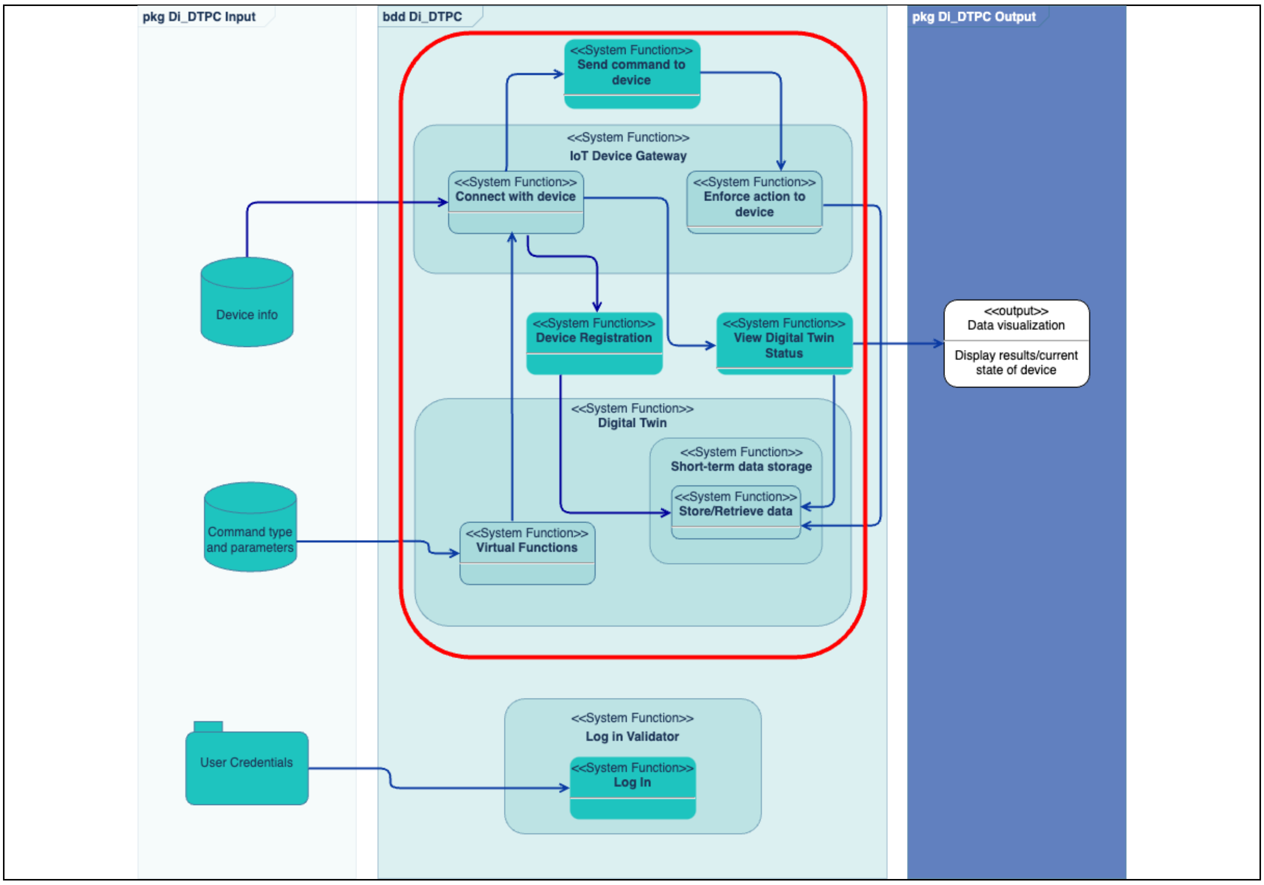
Concerning the DTPC toolkit, the General Architecture and Functional components remain the same, as described in the first version of the Viewpoints' Deliverable. The only update pertains to the Log in Validator, which permits access according to users and roles defined through Keycloack. Users are granted access to specific resources and capabilities according to their access permissions.

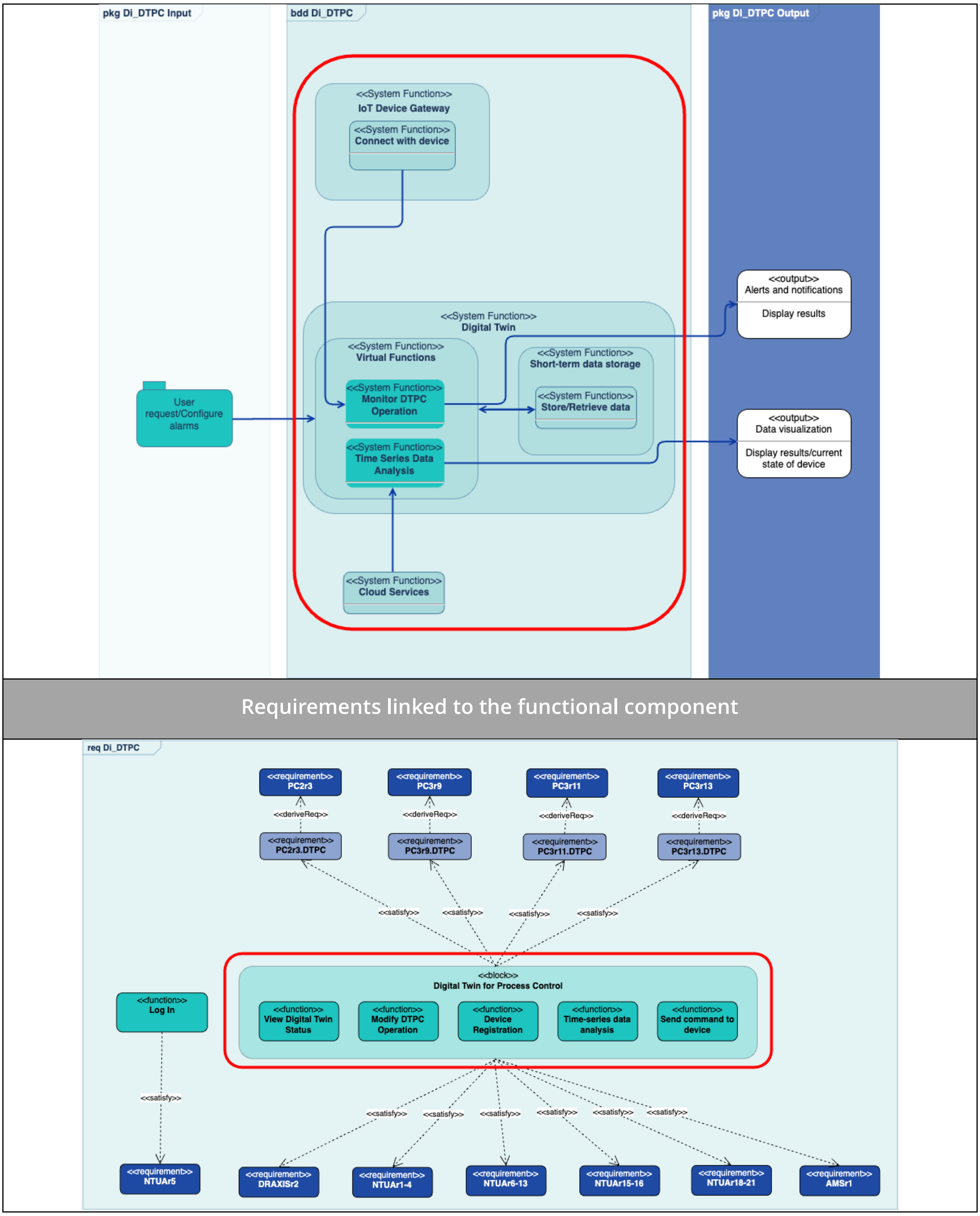
5.10.1.1 Digital Twin for Process Control

Functional component diagram



Functional structure diagram


pkg Di_DTPC Input
bdd Di_DTPC
pkg Di_DTPC Output
<<System Function>>
Send command to device
<<System Function>>
IoT Device Gateway
<<System Function>>
Connect with device
<<System Function>>
Enforce action to device
Device info
<<System Function>>
Device Registration
<<System Function>>
View Digital Twin Status
<<output>>
Data visualization
Display results/current state of device
<<System Function>>
Digital Twin
<<System Function>>
Short-term data storage
<<System Function>>
Store/Retrieve data
Command type and parameters
<<System Function>>
Virtual Functions
<<System Function>>
Log in Validator
User Credentials
<<System Function>>
Log In


pkg Di_DTPC Input
bdd Di_DTPC
pkg Di_DTPC Output
<<System Function>>
IoT Device Gateway
<<System Function>>
Connect with device
<<output>>
Alerts and notifications
Display results
<<System Function>>
Digital Twin
<<System Function>>
Virtual Functions
<<System Function>>
Short-term data storage
User request/Configure alarms
<<System Function>>
Monitor DTPC Operation
<<System Function>>
Store/Retrieve data
<<output>>
Data visualization
Display results/current state of device
<<System Function>>
Time Series Data Analysis
<<System Function>>
Cloud Services
Requirements linked to the functional component
req Di_DTPC
<<requirement>>
PC2r3
<<requirement>>
PC3r9
<<requirement>>
PC3r11
<<requirement>>
PC3r13
<<deriveReq>>
<<requirement>>
PC2r3.DTPC
<<requirement>>
PC3r9.DTPC
<<requirement>>
PC3r11.DTPC
<<requirement>>
PC3r13.DTPC
<<satisfy>>
<<block>>
Digital Twin for Process Control
<<function>>
Log In
<<function>>
View Digital Twin Status
<<function>>
Modify DTPC Operation
<<function>>
Device Registration
<<function>>
Time-series data analysis
<<function>>
Send command to device
<<requirement>>
NTUAr5
<<requirement>>
DRAXISr2
<<requirement>>
NTUAr1-4
<<requirement>>
NTUAr6-13
<<requirement>>
NTUAr15-16
<<requirement>>
NTUAr18-21
<<requirement>>
AMSr1

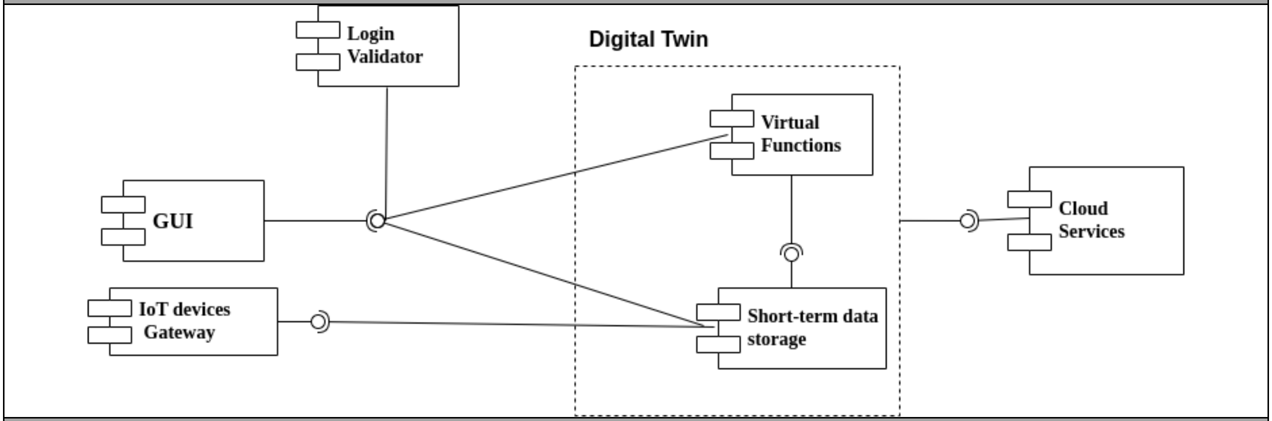
Describe the connection between the functional component and the requirements
• **DRAXISr2, NTUAr6, NTUAr12**: Data collected from pilots regarding the DTPC toolkit will be appropriately organized and maintained. These data should be stored long-term to cloud databases and be supported in multiple data formats. • **NTUAr3, NTUAr8, NTUAr9**: The toolkit will provide secure communication with at least the other 2 toolkits of the suite over channels supporting widely used protocols. • **NTUAr15, NTUAr13, NTUAr19, NTUAr20**: The DTPC components should be updated with new information coming from the installed equipment along with any new configurations of the devices. The toolkit's operation should be checked and controlled regularly. • **NTUAr1, NTUAr2, NTUAr10**: Real time data accessibility should be ensured and time-series analysis through Virtual functions of the DTPC toolkit will be performed so that predictions about material's and devices states are provided. This analysis will provide users with optimal parameters concerning the process execution. • **NTUAr16, NTUAr18, AMSr1**: The toolkit will notify users of defects concerning the physical counterpart and prevent them from causing undesirable effects to the equipment. The input data shall also be checked regarding the provided info. • **NTUAr7, NTUAr11, NTUAr21:** The Digital Twin's architecture, equipment and devices' configuration, visual representation and user's interaction with the toolkit should be documented in detail. Dashboards should provide visualization, modification and data analysis capabilities. • **NTUAr4:** DTPC UI ensures that users can visualize, modify and interact with results effectively and efficiently. • **PC2r3, PC3r9, PC3r11, PC3r13** (grouped): Data concerning devices and equipment's state transmitted by the installed sensors and IoT devices, should be accessible by the UI.

Table 101. Digital Twin functional component diagram, structure and requirements

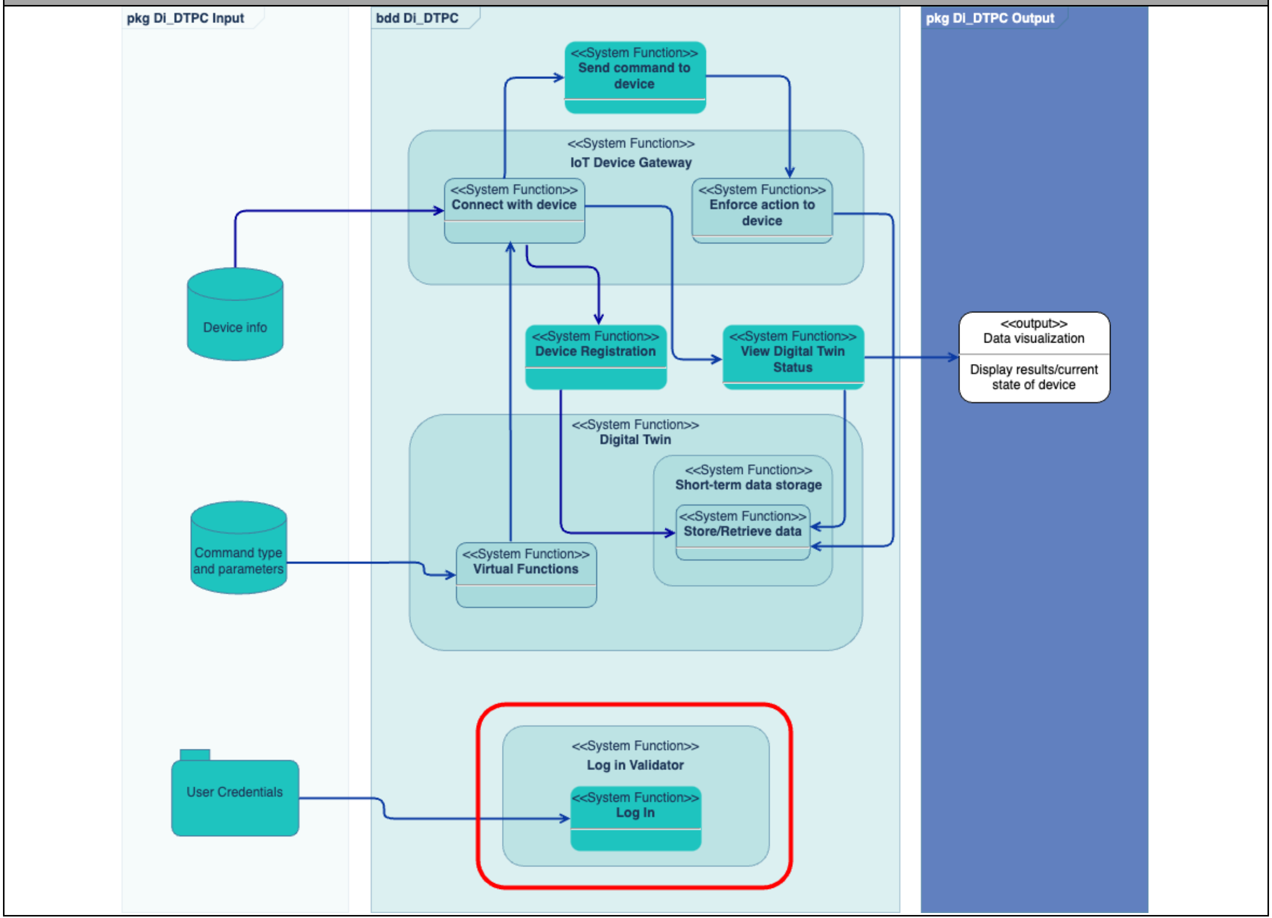
Co-funded by the European Union

5.10.1.2 Log in

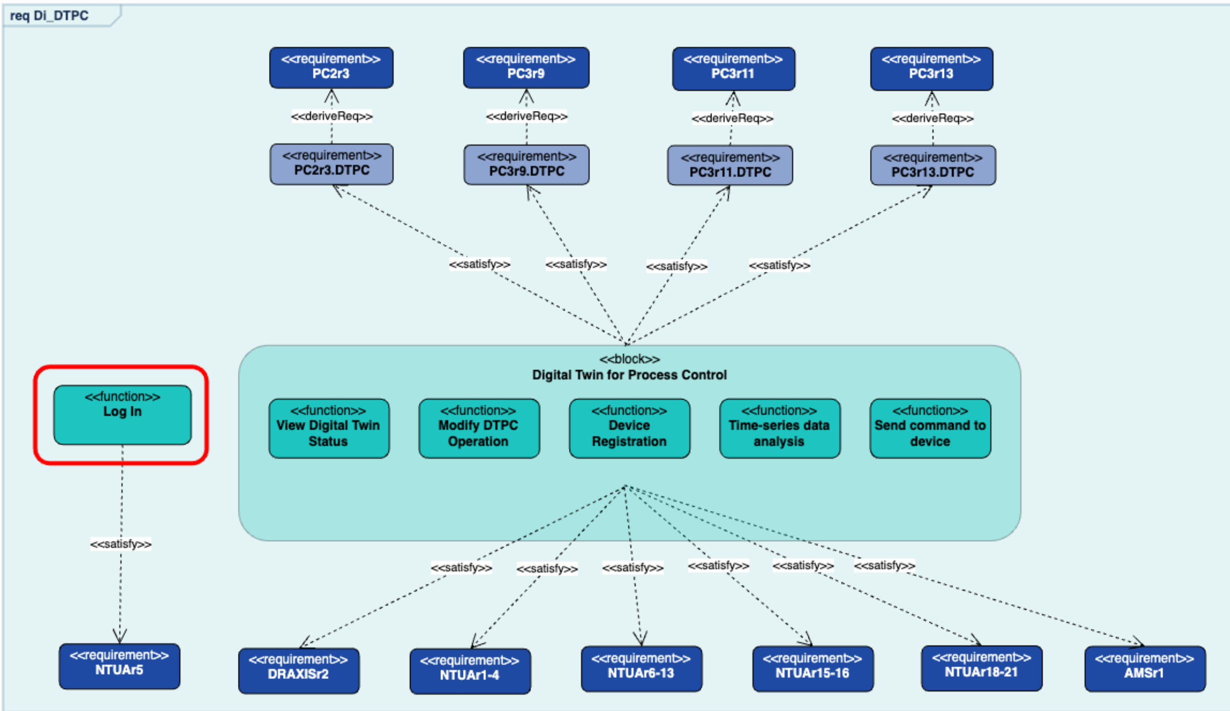
Functional component diagram



Functional structure diagram



Requirements linked to the functional component



Describe the connection between the functional component and the requirements

- **NTUAr5** (*The toolkit shall be accessible only by authorised users):* Only authorized users shall be able to log in, view and edit the data available through the tool. This is ensured by the use of Keycloak.

Table 102. Log in functional component diagram, structure and requirements

Co-funded by the European Union

6 IMPLEMENTATION VIEWPOINT

This task will technically describe the different components of the DiMAT Architecture. To describe the interconnection between the solutions' different components and define the technologies for their development. The Application Programming Interfaces (API) will be specified to define the interconnection and communication of the solutions. 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 implementing the DiMAT toolsets described in WP4, WP5 and WP6.

6.1 IMPLEMENTATION VIEWPOINT APPROACH FOR DIMAT

6.1.1 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 implementing the previously defined components and functionalities and 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, behavior, 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.

6.1.2 Implementation Viewpoint Methodology

In the Implementation Viewpoint of the DiMAT architecture, a detailed technical description of how each toolkit is designed will be presented. The methodology for completing this document is the following:

- **Internal Architecture Diagram**: Initially, a diagram is developed to provide an updated and more detailed internal view of the solution's architecture than the previous version.
- **Component Interaction Analysis**: A comprehensive analysis of how the architectural components of each solution interact and exchange data is presented, alongside indications on how data are pre-processed or cleaned.
- **Technologies and Software Used**: The technologies and software used for developing the toolkit and implementing its functionalities are provided. The selection of software components adheres to the principles mentioned in the benchmarking sections of Deliverable 2.2. One of the crucial requirements when selecting a tool regards its open-source nature. In some cases, preference is also given to tool emerging or being provided from research projects where the consortium partners are involved, considering their offerings and added value provided for DiMAT. Access to documentation and popularity of a software are also considered. Finally, the adoption of a tool by industries is a strong indicator for choosing it. Many of the toolkits mentioned have been already evaluated as part of the respective analysis in D2.2. Upon the submission of D2.2, some further tools were considered by the DiMAT toolkit developers based on their progress in terms of implementation and maturity.Required Hardware and Infrastructure: The necessary hardware and infrastructure for deploying the solution are detailed (e.g., IoT devices, servers).
- **Mapping of Activities and Devices**: A mapping is performed using the activities described in the Usage Viewpoint, describing the necessary devices for implementing each activity.

Co-funded by
the European Union

- **Implementation Maps**: Implementation maps explain how the components interact to deliver the necessary functionalities for implementing the desired activities.
- **Deployment Pattern**: The pattern used for deploying each toolkit is provided, mentioning how the users can install it.
- **List of Developed APIs**: A list of the developed APIs used for connecting the components is provided, detailing the endpoints, their inputs, and outputs.

6.1.2.1 TOOLKIT NAME (PARTNER)

6.1.2.1.1 Internal Architecture of Toolkit

- **General Description**: A diagram of the toolkit's internal architecture detailing its main components and their interactions.
- **Key Components**: A list of key components with a brief description of their function within the toolkit.

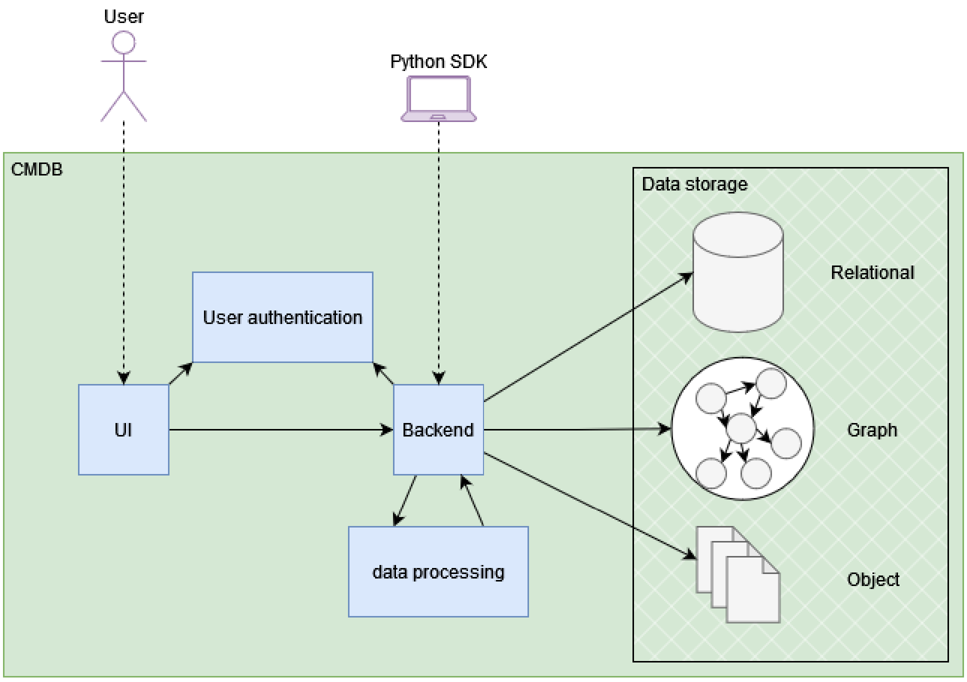


Figure 42. Internal Architecture Example

6.1.2.1.2 Data handling

- **Data Storage**: Details on the methods and technologies used for data storage, including types of databases (SQL, NoSQL), storage structures, etc.
- **Data Analysis:** Methods and tools used for data analysis, including statistical techniques, machine learning models, etc.

6.1.2.1.3 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
Comp1	Comp2	Txt, csv, RDF, etc…	MB, KB, GB…	API/Other

Table 103. Data Flow Example

- **Source Component**: The component where the data flow originates.
- **Target Component/Toolkit**: The component or toolkit that receives the data flow.
- **Data Type**: Format of the exchanged data (TXT, CSV, RDF, etc.).
- **Data Size Estimation**: An estimation of the size of the exchanged data (MB, KB, GB).
- **Transfer Method**: Protocol or method used for data transfer (API, others).

6.1.2.1.4 Necessary Equipment to implement defined activities

Activity	Necessary devices	Functionality
Ring an alarm/Send notification	Buzzer	For the alarm to sound
	Sensors	Capture environmental conditions

Table 104. Necessary Equipment Example

- Activities and Necessary Devices:
  - **Activity:** Name of the activity.
  - **Necessary Devices:** A list of devices needed to carry out the activity.

- **Functionality:** Description of the functionality provided by each device.
- **IoT Devices**: A list of necessary IoT devices, including technical specifications.
- **Servers**: Details on the necessary servers, including recommended configurations.
- **Network Equipment**: Required network equipment, such as gateways and switches.

6.1.2.2 Deployment PATTERN

6.1.2.2.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
ExampleAPI	HTTP	http://example.com/api1:5000	POST	JSON, PNG	JSON

Table 105. API Example

API Details:

- **API:** Name of the API.
- **Protocol:** Protocol used (HTTP, MQTT, etc.).
- **Endpoint/Topic:** URL or topic of the API.
- **Method:** API method (GET, POST, PUT, etc.).
- **Request Data Format:** The data sent in the request (JSON, XML, PNG, etc.).
- **Response Data Format:** Format of data received in the response (JSON, XML, etc.).

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

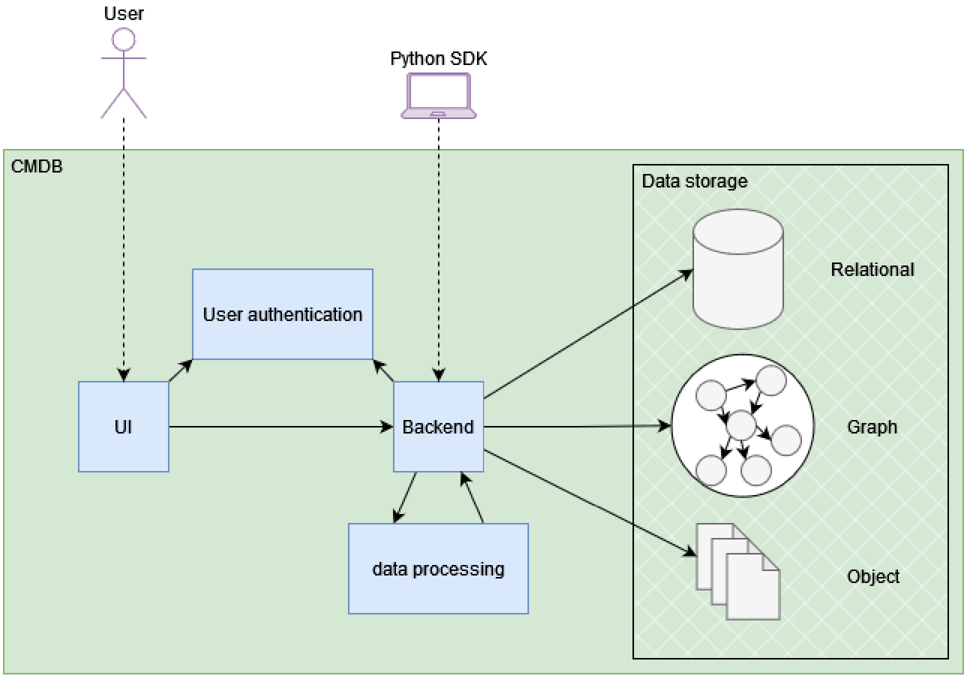


Figure 43. Internal Architecture DICMDB

The Cloud Materials Database (CMDB) is an advanced database solution designed to provide comprehensive data storage, integration, exploration, and the generation of new knowledge and recommendations. This section outlines the key architectural components and technologies that underpin the CMDB system.

6.2.1 Data handling

6.2.1.1 Data Storage

The backend of CMDB employs a combination of SQL and graph databases to manage structured and semi-structured data efficiently. These databases facilitate complex queries and relationships, essential for comprehensive data exploration and knowledge generation. Additionally, CMDB supports file storage solutions, including HDF5 files, providing robust capabilities for handling large datasets and complex data formats.

6.2.1.2 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
UI	Backend	Txt, csv, RDF, JSON	KBs	HTTP
Backend	Data storage	HDF5, RDF, JSON	KBs to GBs	Python
UI/Backend	User authentication	JSON	KBs	HTTP
Backend	Data processing	JSON	KBs to MBs	Python/ HTTP
Data processing	Backend	JSON	KBs to MBs	Python/ HTTP
SDK	Backend	JSON	KBs to MBs	HTTP

Table 106. Data Flows DICMDB

6.2.1.3 Data Preprocessing

Another crucial component within the CMDB ecosystem is the data2rdf tool, which plays a vital role in data integration. The data2rdf tool takes raw files, such as those generated by measurement machines (e.g., tensile test machines), and converts them into RDF files. This transformation ensures that the terms from the raw files are mapped to a controlled vocabulary, enhancing data consistency and interoperability. The data2rdf tool processes the raw input files and outputs RDF files where terms are represented by standardized vocabulary terms. Additionally, it references HDF5 files that store the time-series data from the raw files. This approach ensures that the RDF files remain efficient and manageable by excluding the bulkier time-series data, which can be accessed separately as needed.

6.2.1.4 Data Analytics

CMDB is shipped with an integrated Jupyter environment, empowering users to directly process and analyze data within the platform. This feature allows users to develop, test, and publish integration and post-processing scripts, fostering collaboration and the sharing of analytical methods and results among users.

To further enhance usability, CMDB includes a Python Software Development Kit (SDK). The SDK simplifies interaction with the CMDB system, providing intuitive and programmatic access to its functionalities. This enables users to automate workflows, perform complex data manipulations, and integrate CMDB capabilities into their applications seamlessly.

6.2.2 Technologies employed

CMDB leverages a microservice architecture deployed on top of Docker containers. This approach ensures scalability, flexibility, and ease of deployment, allowing individual services to be developed, deployed, and scaled independently. Each microservice handles specific functionalities within the CMDB ecosystem, promoting a modular and maintainable system architecture.

For user authentication and authorization, CMDB integrates with Keycloak, an open-source identity and access management solution. Keycloak provides a centralized mechanism for managing users, roles, and permissions, ensuring secure and streamlined access to the CMDB platform.

Regarding the benchmarked tools in Deliverable 2.2, PostgreSQL was used for storing data in CMDB on a lower level of semantic integration due to its scalability and as it is actively maintained (cf. benchmark in deliverable D2.2). PostgreSQL handles large datasets and high workloads efficiently and it supports complex queries, joins, and subqueries and is, therefore, ideal for the usage in CMDB. Additionally, Elasticsearch is a potential candidate for enabling the analysis of large datasets stored in CMDB. Elasticsearch also is actively maintained and scalable. Besides PostgreSQL none of the tools were used for data storage in CMDB, as CMDB generally is a novel approach for data storage on higher levels of semantic integration. The applicability of Elasticsearch is still under consideration.

6.2.3 Implementation maps

To implement the CMDB system, the first step is to establish a microservice architecture using Docker containers. Each service, such as user management with Keycloak, SQL and graph databases for data storage, and file storage solutions for HDF5 files, should be containerized and deployed independently. This setup ensures scalability and maintainability. Next, integrate the user management system with Keycloak to handle authentication and authorization, securing access to the CMDB platform.

Following the core infrastructure setup, deploy the integrated Jupyter environment to enable data processing and script development directly within the CMDB system. This environment will facilitate collaboration and ease of use for end-users. Additionally, implement the Python SDK to provide programmatic access to CMDB functionalities, enabling users to automate workflows and integrate CMDB capabilities into their applications seamlessly. Finally, incorporate the data2rdf tool to handle data integration tasks, converting raw measurement files into RDF files with references to HDF5 file-stored time-series data. This comprehensive implementation ensures a robust, scalable, and user-friendly CMDB system.

6.2.4 User Manuals

A user manual will be provided in the future and will be accessible to all users.

6.2.4.1 Necessary Equipment to implement defined activities

Requires an internet connection for updates and authentication via Keycloak.

6.2.4.2 Recommended computing resources for solution

- 2-core CPU, 16GB RAM.

6.2.5 Deployment PATTERN

CMDB leverages a microservice architecture deployed on top of Docker containers. This approach ensures scalability, flexibility, and ease of deployment, allowing individual services to be developed, deployed, and scaled independently. Each microservice handles specific functionalities within the CMDB ecosystem, promoting a modular and maintainable system architecture. CMDB runs in Docker containers, ensuring compatibility across different computing environments.

6.2.5.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Get knowledgeitem (e.g., dataset)	HTTP	/api/knowledge/kitems/{kitem_id}	GET	N/A	JSON
Create knowledgeitem (e.g., dataset)	HTTP	/api/knowledge/kitems	POST	JSON	JSON

Search knowledgeitem (e.g., dataset)	HTTP	/api/knowledge/kitems/search	POST	JSON	JSON

Table 107. API DICMDB

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

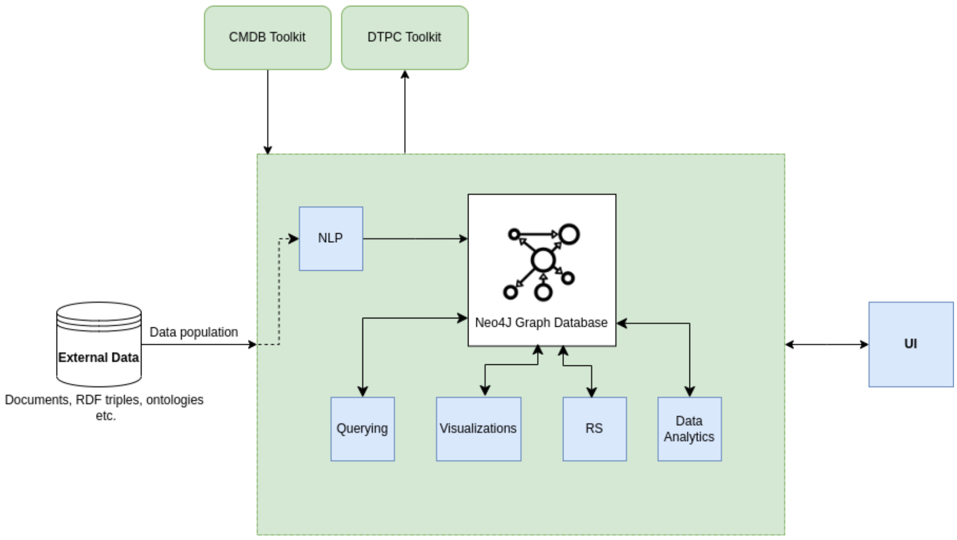


Figure 44. Internal Architecture DIKAF

The most central component of the KAF toolkit is the Graph Database. The graph database used in KAF is the Neo4J graph database, an open-source and popular solution. This database is populated with data coming from external sources that are documents describing material processes, RDF triples or through appropriately designed queries by the users of the toolkit if they are familiar with Cypher. 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.

6.3.1 Data handling

6.3.1.1 Data Storage

The data for this toolkit are stored in a knowledge graph database implemented in Neo4J.

6.3.1.2 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
GUI	Neo4J database	Txt	MB	Neo4J Python API
External data	Neo4J database	CSV	MBs	HTTP
KAF	User	PNG	MBs	HTTP
KAF	DTPC	TXT	MBs	HTTP API

Table 108. Data Flows DIKAF

6.3.1.3 Data Preprocessing

Same as v1 of Implementation Viewpoint.

6.3.1.4 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
Data population	Data can be inserted in the graph through appropriately designed CSV/XLSX files for which the template is provided.
Visualization	Parts of the graph are visualized to give the user a brief overview of the relationships between subsets of the graph's entities. These visualizations can either be generated through queries by the user if they are familiar with Cypher or by selecting from appropriately designed menus (filters) aiming to cover various use-cases for the less experienced users.

NLP	Necessary techniques to extract entities and relationships from text documents concerning materials
Recommender System	Recommender systems used for suggesting similar materials or appropriate configurations for the business processes stored in the graph.
Data analytics	Functionalities developed in Python aiming to address specific use cases.
Querying	Preselected filters applied to the graph to aid the users in querying and retrieving information and visualizations.

Table 109. Data Analytics DIKAF

6.3.2 Technologies employed

- The tools/libraries selected for developing aspects of KAF are selected so that they adhere to the metrics examined in the benchmarking mentioned in Deliverable 2.2. These are:
- Neo4J is employed as the graph database. It is selected for its strong points compared to other solutions as explained also in the first version of the Viewpoint,
- Visualizations are generated with Neovis, a software implemented in Javascript that allows for the creation of interactive visualizations of (parts of) the knowledge graph. Neovis is an open-source Javascript library that can be used to display (sub)graphs from Neo4J. It's GitHub statistics indicate that it is a popular choice, having 1.6K stars and more than 300 forks. Using Neovis significantly reduces the effort for visualing graphs with information on them' by being able to operate as a script called from the front-end. Open-source LLMs are employed for Natural language processing tasks (NLP) such as Named Entity Recognition (NER). Open-source LLMs have been in the spotlight of research during the recent period. One of their main usages can be for NLP tasks such as NER or for aiding in understanding user requests. For example, one of the most popular open-source LLMs, Llama3 is downloaded more than 900K times during the past months. Also, quantized models of open-source LLMs allow for deploying in scenarios where limited amounts of GPU is available. Integration with Python scripts is very straightforward.
- Python libraries such as spacy for NLP tasks. As also explained in the first version, spaCy is a popular and well documented library.
- Ampligraph, a Python library, for embedding of the KG in geometric space which can be used for the purposes of similarity extraction, recommendations, etc. Ampligraph was among the technologies examined in D2.2 and is a valid option since it contains

implementations of the most popular graph embedding algorithms, it is well-documented and open-source.

- Front-end (User Interface) created with Vue.js. Vue is an open-source framework for developing web user interfaces. It has more than 48K stars and more than 8K forks indicating its popularity. It is also supported by major technological industries. Being developed in Javascript it allows easy integration with other useful Javascript libraries such as Neovis for visualization and node.js for connecting with the backend scripts.
- HTTP APIs are created with Flask (Python) and node.js (Javascript), both of which are very popular technologies for implementing backend APIs, allowing enough flexibility for the developer. Both have large and active supporting communities, are easy to use, with small learning curve, ideal for designing RESTful APIs.

6.3.3 Implementation maps

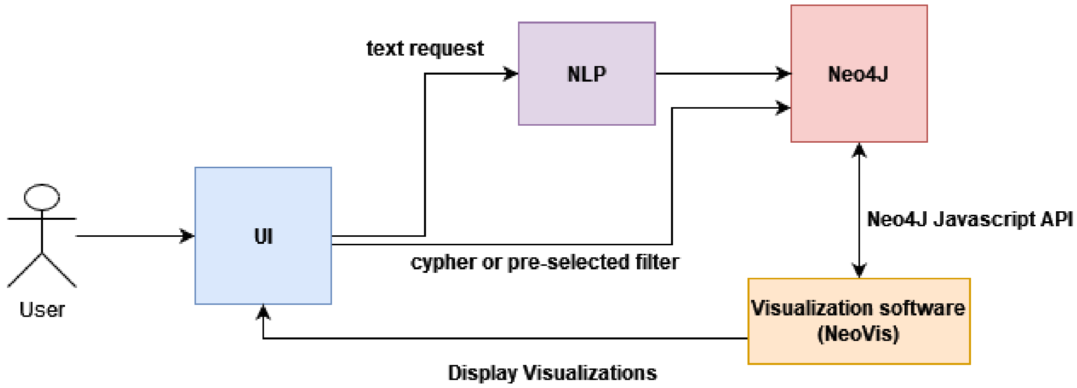


Figure 45. Implementation Maps DIKAF

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 backend. If the request is in text format an NLP component further processes the input in case data is not in a cypher format, and the processed input is passed to the Neo4j Database to perform the suitable queries. By employing Neo4J Javascript APIs that communicate with the NeoVis visualization library, the results are returned and displayed to the user through the UI for further inspection and analysis.

6.3.4 User Manuals

6.3.4.1 Necessary Equipment to implement defined activities

There is no specific equipment required to deploy the KAF toolkit besides a server or PC. No IoT devices are necessary or any specific network equipment. All activities can be implemented if the users have access to a PC and Internet Connection (if they do not wish to have a local installation of the toolkit).

6.3.4.2 Recommended computing resources for solution

The necessary computing resources vary depending on the size of the graph. Considering a medium size knowledge graph (less than a million nodes and edges), 16 GB of RAM would suffice. This amount of RAM is also enough for executing the machine learning models of the ampligraph library concerning knowledge graph embedding.

Regarding the LLM for NLP tasks, if the user wants to use such capabilities, at least 4GBs of GPU is suggested. The visualization libraries employed (such as Neovis) do not require significant computing resources.

6.3.5 Deployment PATTERN

The solution is provided as a dockerized application using Docker containers. It is hosted in DiMAT cloud infrastructure but can also run locally in a standard PC.

6.3.5.1 APIs description

The APIs developed are developed either with Python or Node.js. They run at the backend and get the input from the User Interface and pass it to the developed algorithms for querying, visualizations, data uploading, data analytics, etc.

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Execute query	HTTP	/query	GET	JSON	JSON
Get product info	HTTP	/product/:name	GET	JSON	JSON

Delete product	HTTP	/product/:name	DELETE	-	-
Find similar materials	HTTP	/similarities	POST	JSON	JSON
Embed KG	HTTP	/embed	POST	JSON	TEXT
Upload data to KG	HTTP	/upload	POST	JSON	TEXT
Download template for data upload	HTTP	/download	GET	-	File
Retrieve data	HTTP	/fetch-data	POST	JSON	JSON

Table 110. API DIKAF

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

The figure below represents the internal architecture of the MEC-LCA toolkit, updated since V1 of the Viewpoints with the addition of the Keycloak component.

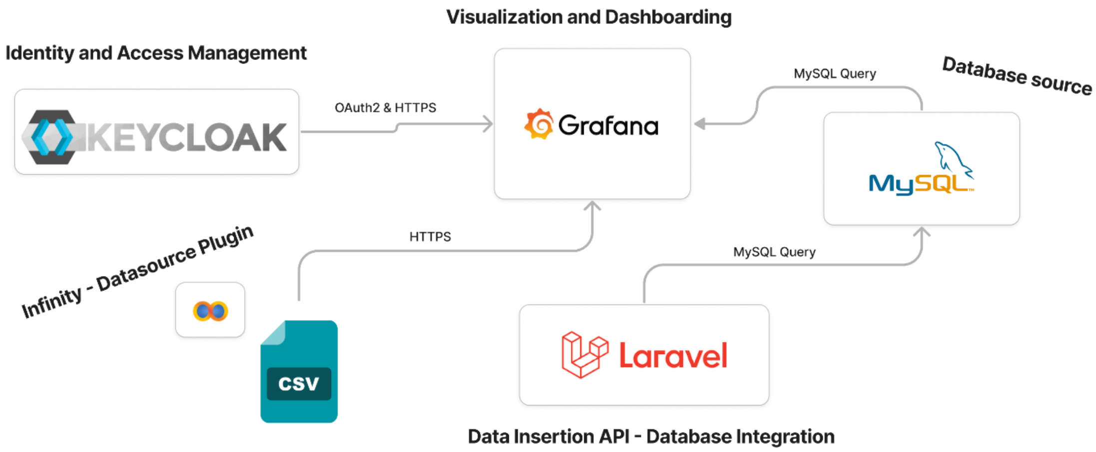


Figure 46. Internal Architecture DIMECLCA

6.4.1 Data handling

6.4.1.1 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
GUI	Grafana	CSV	500 MB	HTTP/S
External data	Grafana	CSV	100 MB	Datasource plugin
Laravel	MySQL database	MySQL Query	500 MB	MySQL Query
MySQL database	Grafana	MySQL Query	500 MB	HTTP/S
Keycloak	Grafana	OAuth 2/ HTTP/S	10 KB	HTTP/S

Table 111. Data Flows DIMECLCA

6.4.2 Technologies employed

The technologies utilized in MEC-LCA are the same as presented in D3.3 with the following additions:

- Keycloak, applied to all toolkits in DiMAT, is an open-source identity and access management solution with significant community support, allowing for Single-Sign On and fine-grained authorization options which are ideal for cases where precise control over user permissions and access levels is important. Grafana which is employed in MEC-LCA is easily integrated with Keycloak.
- Infinity – Datasource Plugin, simplifies data integration, allowing users to upload various file types such as CSV, JSON and XML into the MEC-LCA toolkit directly through its interface. It is offered by Grafana and its code is regularly updated on GitHub.

6.4.3 Implementation maps

The diagram below has been updated since V1 with the additions of the data source plugin (infinity) and more elaboration on the login and access.

Once the user specifies the desired data through the GUI, the system loads this data either from the data storage or the data source plugin. Before any content is displayed, the user's credentials are validated through a login process. After successful authentication, the system checks the user's access rights to determine what resources they are authorized to view or manipulate. This could involve checking permissions, roles, or groups associated with the user's account. Once access is granted, the user can view or manipulate the requested data according to their permissions and the capabilities of the system.

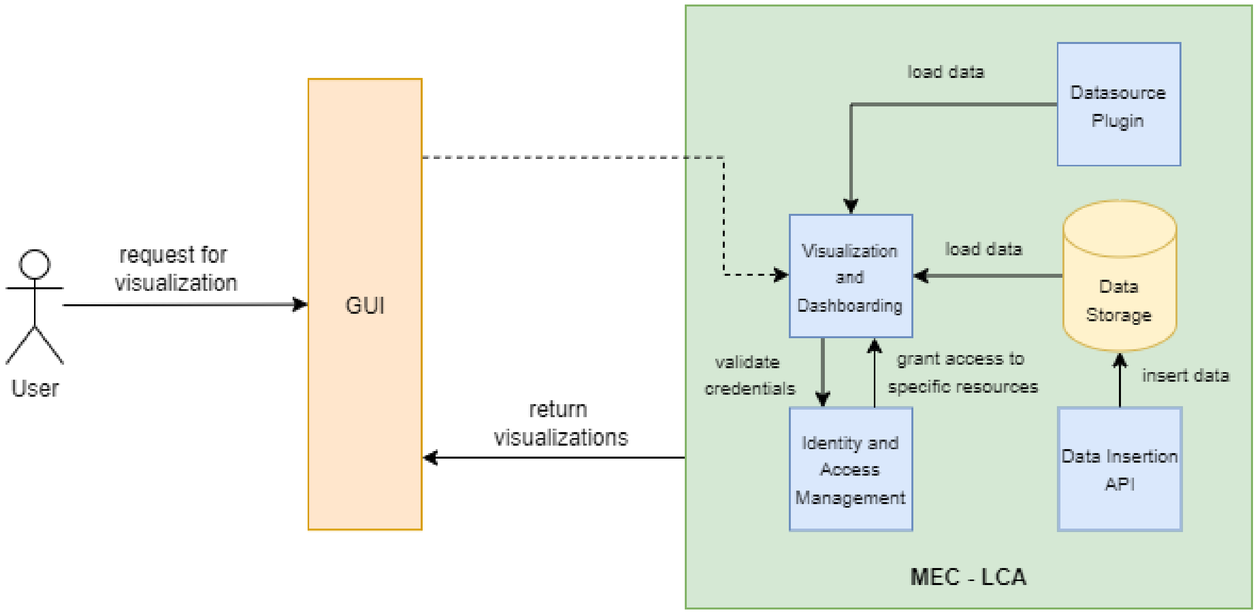


Figure 47. Implementation Maps DIMECLCA

6.4.4 User Manuals

6.4.4.1 Servers

The application is deployed on a DRAXIS server using Docker. This server, equipped with the DiMAT stack, handles the execution of the necessary components, such as Laravel for backend processing, MySQL for database management, and Grafana for data visualization.

6.4.4.2 Network Equipment (e.g., Gateways, etc.)

Internet connectivity is crucial for accessing the services hosted on the server. While no dedicated network equipment is required in this case, MEC-LCA including Grafana, Laravel, and MySQL, rely on stable internet connectivity to ensure accessibility as users access the toolkit through the application's URL.

6.4.4.3 Recommended computing resources for solution

The toolkit relies on Docker to host essential services such as Grafana for data visualization, Laravel for backend processing, and MySQL for database management, requiring a robust infrastructure to support its functionalities. For standard usage scenarios, a minimum configuration of 2 CPUs, 4 GB of RAM, and 120 GB of storage is suggested. This setup ensures smooth operation under typical workloads. However, in anticipation of higher levels of traffic and data processing, an upgraded configuration is recommended. This includes at least 4 CPUs, 8 GB of RAM, and 240 GB of storage. With this enhanced setup, the toolkit can effectively handle heavier workloads, support additional services, and maintain optimal performance during peak usage periods.

6.4.5 Deployment PATTERN

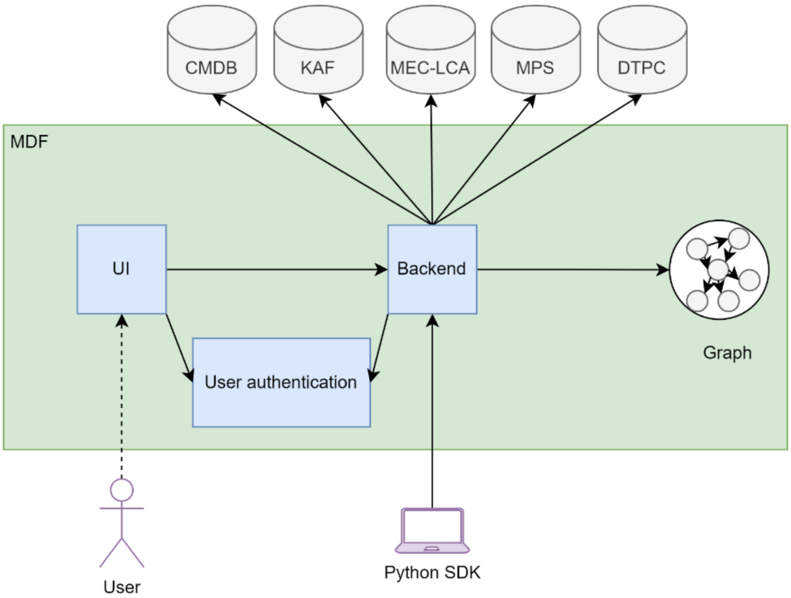
The toolkit is deployed using Docker containers and hosted on a DRAXIS server. It is designed as a standalone installation but can support both monolithic and microservices deployment modes.

6.4.5.1.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Data Insertion API	HTTP	http://api.dimat-meclca.draxis.gr/api/store	GET		JSON

Table 112. API DIMECLCA

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



The Materials Design Framework (MDF) is a specialized tool designed to support engineers in the materials design process by offering comprehensive services for discovering connections between models, materials, and materials properties. A significant feature of MDF is the "correlation app," designed to assist engineers in exploring attributes related to their specific use cases. This application provides tools for analyzing and visualizing correlations between different data attributes, enabling users to gain deeper insights and make data-driven decisions. The correlation app enhances the overall functionality of MDF by offering specialized analytical capabilities tailored to the needs of materials engineers.

A central component of MDF is the "materials relation" service, which enables users to discover and explore connections between different models, materials, and their properties. This service facilitates the identification of relevant relationships, driving innovation and informed decision-making in materials design. MDF extends the search capabilities of other toolkits by integrating a materials-relation component. This enhanced search function allows users to perform more targeted and comprehensive searches, improving the discoverability of relevant materials data and relationships.

6.5.1 Technologies employed

The custom frontend, built using Angular, provides an intuitive and user-friendly interface for accessing MDF's capabilities. For user management, MDF connects to the DiMAT ´s central instance of Keycloak, ensuring secure and streamlined authentication and authorization. This integration provides a consistent user management experience across different tools within the ecosystem. MDF is deployed using Docker containers, like the CMDB system, ensuring scalability, flexibility, and ease of deployment. Each component, including the materials-relation service, search functionality, and correlation app, is containerized, promoting modularity and maintainability.

Technologies employed based on the benchmarks in Deliverable 2.2:

- Scikit-learn is used for training machine learning models and identifying correlations inside data. It was decided to use scikit-learn as it offers a simplistic way to setting up machine learning models in a sufficient depth necessary to fulfil the purpose of the correlation component of the MDF. Scikit-learn also has an active community and the code is publicly available.
- For storing materials relations, none of the benchmarked software is used. Instead, the CMDB toolkit is used. This is done to not introduce another storage module and, instead, rely on already developed toolkits in DiMAT.
- For the advanced search to be provided in the MDF, possibly an LLM will be used, such as Llama, as it is open source and actively maintained.

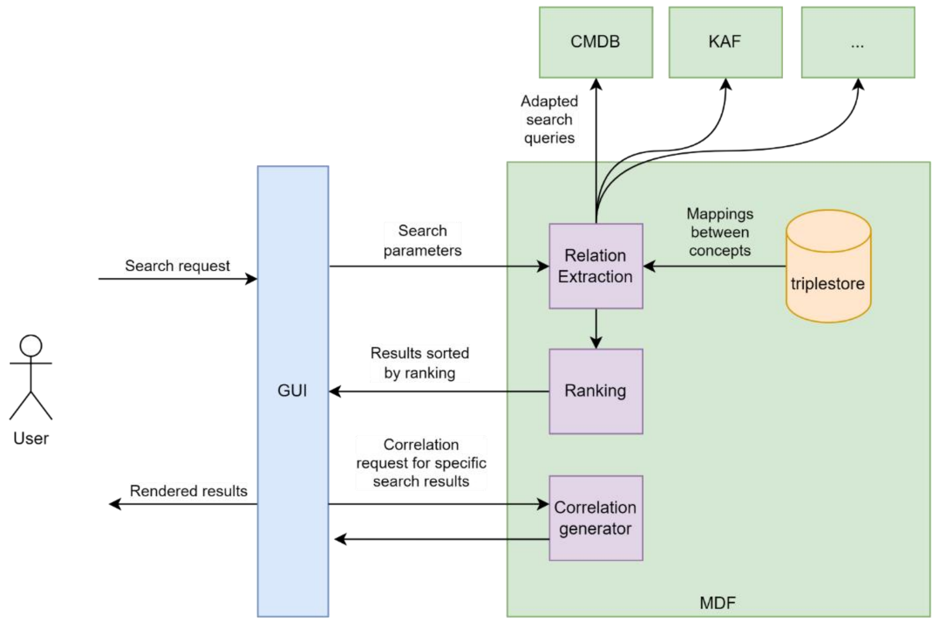
6.5.2 Implementation maps

A significant feature of MDF is the "correlation app," designed to assist engineers in exploring attributes related to their specific use cases. This application provides tools for analyzing and visualizing correlations between different data attributes, enabling users to gain deeper insights and make data-driven decisions. The correlation app enhances the overall functionality of MDF by offering specialized analytical capabilities tailored to the needs of materials engineers.

A central component of MDF is the "materials relation" service, which enables users to discover and explore connections between different models, materials, and their properties. This service facilitates the identification of relevant relationships, driving innovation and informed decision-making in materials design.

MDF extends the search capabilities of other toolkits by integrating a materials-relation component. This enhanced search function allows users to perform more targeted and comprehensive searches, improving the discoverability of relevant materials data and relationships.

We start by developing the correlation app that will help engineers explore attributes related to the data of their use case. We then proceed with the materials relation service to enable the discovery of connections between models, materials, and properties. Next, extend the search functionality to capture the other DiMAT's toolkits as well as to incorporate the materials-relation component, enhancing the search capabilities for users.



6.5.3 User Manuals

6.5.3.1 Necessary Equipment to implement defined activities

Activity	Necessary devices	Functionality
Ring an alarm/Send notification	Buzzer	For the alarm to sound
	Sensors	Capture environmental conditions

Table 113. Necessary Equipment DIMDF

6.5.3.2 Network Equipment

Requires an internet connection for updates and authentication via Keycloak.

6.5.3.3 Recommended computing resources for solution

Minimum hardware Requirements: 1-core CPU, 8GB RAM.

6.5.4 Deployment PATTERN

Runs in Docker containers, ensuring compatibility across different computing environments.

6.6 DIMAT MATERIALS MODELLING AND BEHAVIOUR PREDICTION (DI^{MM})

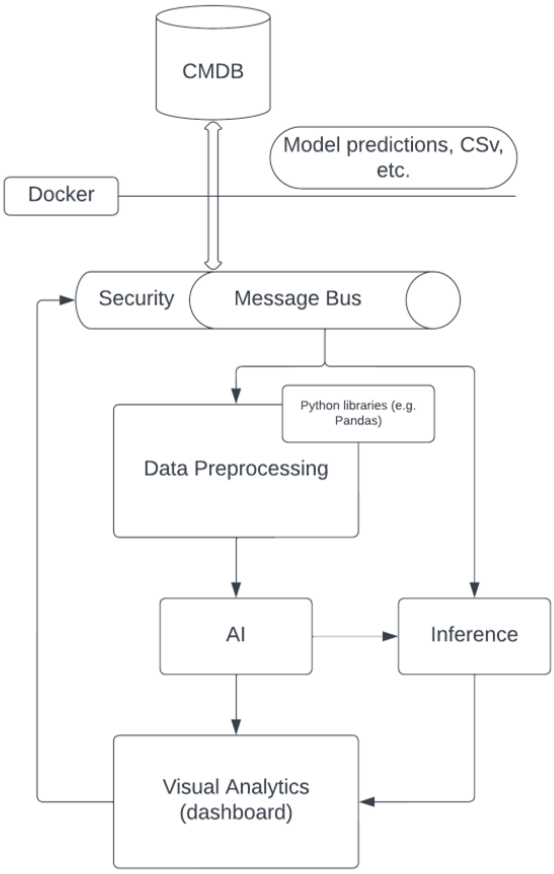


Figure 48. Internal Architecture DIMM

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 the database for efficient data retrieval. It also enables bi-directional information flow. Data is collected via the partners' submissions which involve the material properties and behaviors under different manufacturing conditions. This data is then preprocessed and analyzed 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.

6.6.1 Data handling

6.6.1.1 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.

6.6.1.2 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
CMDB	MM	CSV, json	MBs	API/Other
MM	User	CSV, json	MBs	API/Other

Table 114. Data Flows DIMM

6.6.1.3 Data Preprocessing

Data preprocessing encompasses an array of procedures, including data cleaning, 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 adapt as new data becomes available, enhancing their performance and accuracy.

6.6.1.4 Data Analytics

Utilising a range of AI algorithms focusing on robustness, 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. 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.

6.6.2 Technologies employed

In the development of DiMAT MM, a core set of advanced software technologies and libraries are strategically employed to optimize the system's capabilities in machine learning and data analysis for enhanced material property predictions. Scikit-learn is utilized extensively for its robust machine learning algorithms, along with algorithms like XGBoost. Pandas supports this by enabling sophisticated data manipulation and analysis, particularly adept at handling CSV files and performing essential metrics calculations. Complementing these, NumPy offers powerful tools for data manipulation through its array and matrix functions, crucial for feature importance calculations and other quantitative analyses.

Visualization of data and results is achieved through a combination of Matplotlib, Seaborn, and Altair. Matplotlib and Seaborn are chosen for their ability to produce detailed static graphs, essential for initial data exploration and analysis, while Altair provides interactive visual capabilities that enhance user engagement and understanding of complex data relationships. HuggingFace was also employed for LLM support. Streamlit is integrated to create a user-friendly interface, allowing users to interact with the system, perform data analysis, and view predictions efficiently. The entire solution is encapsulated within Docker, which ensures that the application is easily deployable across different environments, maintaining consistency and facilitating scalability.

These technologies represent the primary tools in the DiMAT MM toolkit, each selected for their specific strengths in handling different aspects of the project from data handling and model building to user interaction and deployment. This focused selection of technologies ensures that DiMAT MM is both powerful in functionality and efficient in operation, tailored to meet the demanding requirements of predicting material properties in industrial applications.

Technologies employed based on the benchmarks in Deliverable 2.2:

- Scikit-learn is used for training machine learning models and identifying correlations inside data. It was decided to use scikit-learn as it offers a straight forward way to set up agile classic machine learning models in order to fulfill the requirements of the MM (since the amount of data, at least at the initial stages did not warrant the use of deep learning frameworks like pytorch and tensorflow). Scalability and accuracy were the two metrics this tool exceled above any other alternative.
- HuggingFace was used for the LLM support for the "Ask the Assistant" feature of the tool as it is a vast open-source platform that hosts wide range of tools in the LLM field. It was selected because it is the most popular platform and very highly maintained. Due to that, the ease of use of this tool was where it clearly exceled.

6.6.3 Implementation maps

The implementation map, serving as the data flow diagram for DiMAT MM, begins with the data collection phase, where initial data is gathered from various sources. This data then enters an in-depth preprocessing stage, where it undergoes cleaning, normalization, and preparation—setting the stage for a more refined analysis. Following this, the data advances into the AI analysis layer, where it is processed through advanced machine learning algorithms to enhance the predictive accuracy and reliability of the model.

After analysis, the processed data moves to the prediction and inference layer. Here, robust models based on the data are developed, leveraging mathematical and statistical tools. The AI analysis includes feature importance calculations, which are pivotal in determining which parameters most significantly affect material properties. This step not only aids in refining the predictive models but also enhances the interpretability of the AI system, offering insights into the underlying factors driving material behavior. The refined data then progresses to the prediction and inference layer, where precise material properties predictions are made.

The outcomes from this stage are then directed towards a user-friendly interface developed using Streamlit, which visualizes the results in an interactive and accessible manner. This visualization aids users in comprehending complex data relationships and predictions clearly and thoroughly. The final stage in the implementation map captures the integration of these insights back into the Materials Design Framework. This integration ensures that the valuable findings from the current analyses are available for future iterations, promoting continuous improvement and refinement of the system.

6.6.4 User Manuals

A user manual and sufficient documentation will be available and accessible in the future to all users.

6.6.4.1 Servers

A server will be needed to host the dockerized solution.

6.6.4.2 Recommended computing resources for solution

For the MM solution which seeks to leverage machine learning (ML) for insights and predictions from raw material data, a well-equipped computing setup is essential. A server with 16 RAM and a state-of-the-art CPU with the processing power to handle ML algorithms efficiently would be needed. Utilizing GPUs for accelerated model training and inference tasks would be desirable. Local storage solutions are crucial for managing datasets and model outputs effectively.

6.6.4.3 Deployment PATTERN

The MM toolkit will be deployed as a dockerized application. The server handles the execution of all the components, including the calculation code, which is the most resource demanding one.

6.6.4.4 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Read Data	HTTP	/api/v1/upload	POST	CSV	CSV
Perform EDA	HTTP	/api/v1/eda	GET	N/A	JSON

Feature Importance	HTTP	/api/v1/feature-importance	GET	N/A	JSON
Predict Properties	HTTP	/api/v1/predict	POST	JSON	JSON

Table 115. API DIMM

6.7 DIMAT MATERIALS DESIGN, CHARACTERIZATION AND SYNTHESIS (DI^{MD})

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 principal components of the toolkits are:

1. The frontend, a graphical user interface where the user ask for material properties and the results are presented.
2. The backend, a webserver which collects inputs from the user and outputs from the calculations, saving and retrieving them into and from the internal database.
3. The internal material database, which stores the input of the user and the results obtained.
4. The simulation scheduler, which manage the interaction between the webserver and the calculation code, being a webserver itself.
5. The calculation code, which executes the virtual experiments.

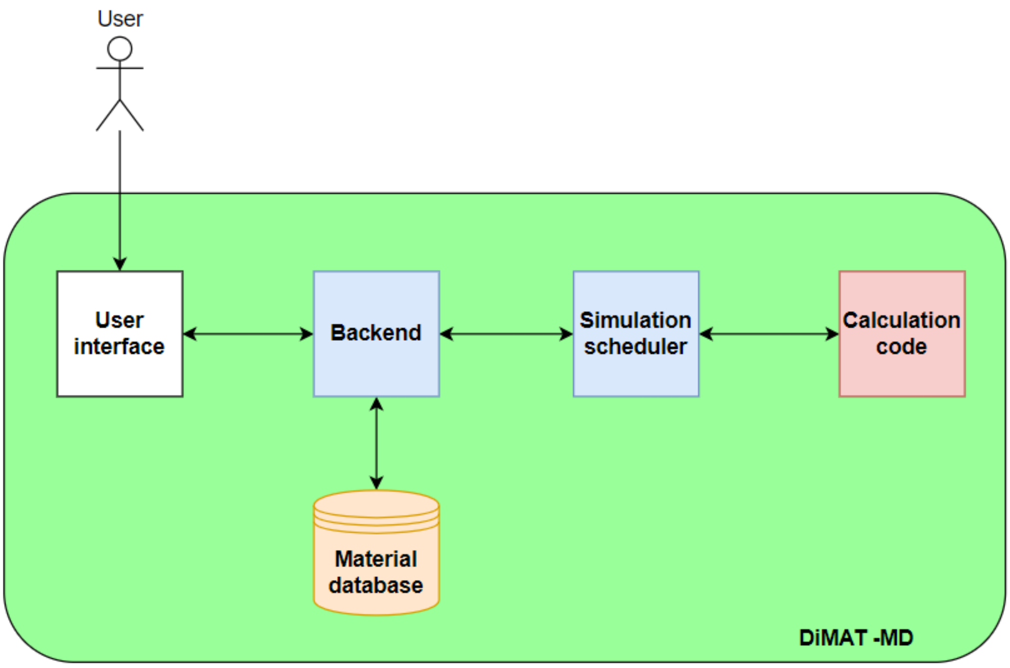


Figure 49. Internal Architecture DIMD

The backend and the simulation scheduler are being split in two different components, from an architectural point of view, because the scheduler needs a direct access to the machine where the calculation code is installed, while the backend could be deployed anywhere, provided that a connection is made available. This allows greater flexibility in choosing a suitable deployment configuration.

6.7.1 Data handling

6.7.1.1 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.
- **Internal material database**: this database stores the defined materials and their properties once they are calculated and is located on the same machine of the backend.
- **Scheduler database**: this database stores the calculations scheduled, the inputs and

the outputs once available before this data are transmitted back to the webserver to be stored into the internal material database. It is located on the calculation machine.

- **CMDB**: the properties of the material components or the final properties of the composite can be retrieved or uploaded from a CMDB instance.

6.7.1.2 Data Flows

There are three data flows in this toolkit:

1. Inputs from the user coming from the frontend are sent to the backend via a REST API exposed by the backend. These inputs consist of the definition of the composite material in terms of base components and their properties, microstructures and their properties, composite layup. This data, once received from the Backend, are saved into the internal material database. All the CRUD operations are implemented, which means that the data flow is bi-directional.
2. Complete description of the defined composite materials is sent from the backend to the simulation scheduler via a REST API exposed by the scheduler. When results are available, data are transferred back via the same API.
3. Optionally, material definitions and properties can be uploaded or retrieved from a CMDB instance.

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
Frontend	Backend	Json	KB	REST API
Backend	Scheduler	Json	KB	REST API
Backend	CMDB	Json	KB	REST API

Table 116. Data Flows DIMD

6.7.1.3 Data Preprocessing

Apart from a basic validation occurring inside the frontend, the only preprocessing needed is translating the data to a valid Digimat® input file. A dedicated Python function takes care of the input validation and the translation process.

6.7.1.4 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.

6.7.2 Technologies employed

Different frameworks and technologies are used to implement the different components of the toolkit.

The backend and the simulation scheduler are two webservers implemented via FastAPI [4], a Python web framework conceived for building REST API. It generates OpenAPI documentation for the API built in an automated way. 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.

The internal material database is implemented using PostgreSQL [5], a well-known open-source object-relational database system. It has been chosen over Sqlite3 python package given its superior performances in terms of concurrent access and scalability (see section 6.4.1 of the D2.2 deliverable). Interaction between the webserver and the database is managed via the SQLAlchemy [6] package, a Python toolkit developed for database access, manipulation and management.

The scheduler database is implemented via SQLite, a small SQL database engine. In this case, given the low performances requirements, the ease of implementation has been a factor to choose this specific database engine over PostgreSQL. Also, in this case the interaction is managed via the SQLAlchemy toolkit.

The calculation code is Hexagon Digimat®. 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 and will be employed to execute virtual experiments on complex materials and calculate their mechanical and thermal properties.

The frontend is implemented via NiceGUI [7], a Python UI framework that allows to build complex interfaces. It has been chosen mainly because it is built on top on FastAPI and can serve UI via browser or in native mode.

Finally, Docker containers are used to deploy the implemented components as better described below.

6.7.3 Implementation maps

The next figure shows how the different components interact starting from the initial input of the user. Once the user asks for a specific material configuration to be analysed, the user interface send an appropriate request to the backend containing all inputs. The configuration is stored into the material database and sent to the simulation scheduler via an appropriate request.

All data are internally packaged into json files, sent back and forth via API requests. The simulation scheduler pre-processes the received data and produces a valid Digimat® input file which is then passed to the code whose execution is triggered. The scheduler can manage a queue of different simulations, so while waiting for other inputs from the user, can retrieve the output data generated by the simulation code and produce a structured json file with the requested output data.

The output json data is then processed and sent back to the backend, which provides to save the requested material properties into the material database in the specific record related to the simulation. The results can now be requested by the user via the graphical interface. Apart from insert data and viewing output locally, the user can also request data from the CMDB and import them into the local material database. In this case, once the specific command is made via the user interface, the backend provides to ask data from the CMDB via an appropriate request and save the data locally. The opposite dataflow is possible, from the local database to the CMDB, and is managed in the same way by the backend after the specific request is made by the user in the user interface.

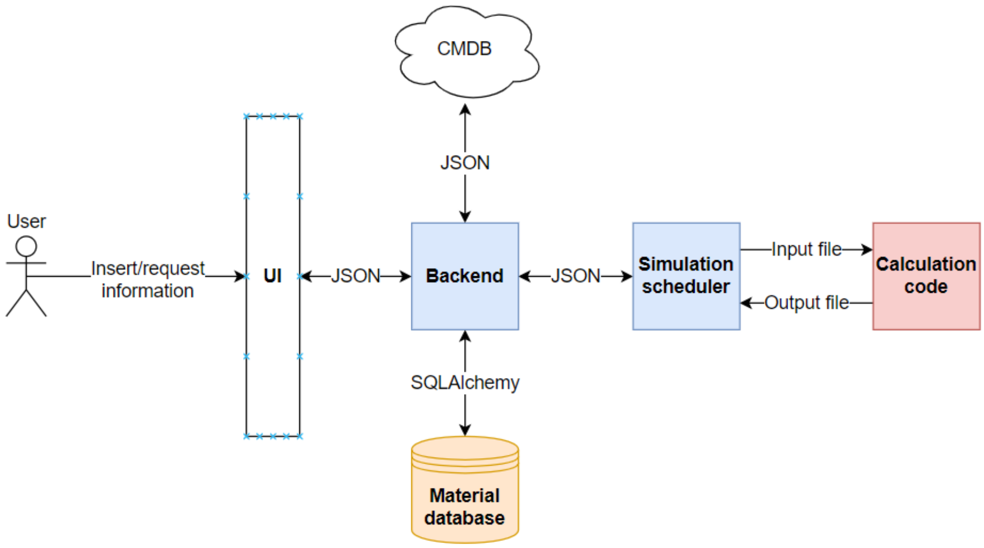


Figure 50. Implementation Map DIMD

6.7.4 User Manuals

6.7.4.1 Servers

The application is deployed on a single server using Docker. The server handles the execution of all the components, including the calculation code, which is the most resource demanding one.

6.7.4.2 Recommended computing resources for solution

To run the finite element simulations, the calculation code should be installed on a machine with at least 8 cores and 32 GB RAM. The available space disk should be of at least 50 GB.

6.7.4.3 Deployment PATTERN

The toolkit is deployed using Docker containers hosted on a Cetma server, one container for each of the toolkit components, except for the calculation code. The calculation code is installed on the same machine. The toolkit can support a monolithic deployment, as the default configuration, or can be split into two microservices, one related to the data management (UI, backend and database) and the other related to the calculations (simulation scheduler and calculation code).

6.7.4.4 APIs description

All endpoints are accessible only from the frontend. The /api/jobs endpoint is exposed by the simulation scheduler, while all the other by the backend webserver.

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Create material	HTTP	/api/materials	POST	JSON	JSON
Read materials	HTTP	/api/materials	GET	-	JSON
Update material	HTTP	/api/materials/{id}	PATCH	JSON	JSON
Delete material	HTTP	/api/materials/{id}	DELETE	-	-
Create microstructure	HTTP	/api/microstructures	POST	JSON	JSON
Read microstructures	HTTP	/api/microstructures	GET	-	JSON

Update microstructure	HTTP	/api/microstructures/{id}	PATCH	JSON	JSON
Delete microstructure	HTTP	/api/microstructures/{id}	DELETE	-	-
Create composite	HTTP	/api/composites	POST	JSON	JSON
Read composites	HTTP	/api/ composites	GET	-	JSON
Update composite	HTTP	/api/ composites/{id}	PATCH	JSON	JSON
Delete composite	HTTP	/api/ composites/{id}	DELETE	-	-
Create simulation	HTTP	/api/jobs	POST	JSON	-
Get simulation results	HTTP	/api/jobs/{id}	GET	-	JSON

Table 117. API DIMD

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

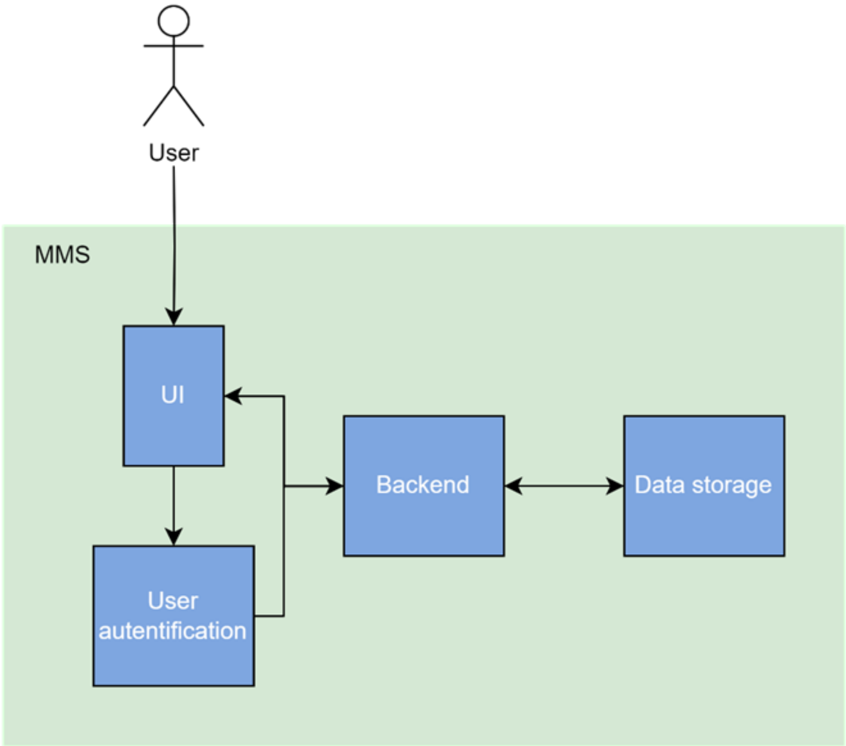


Figure 51. Internal Architecture DIMMS

The diagram illustrates the internal architecture of the MMS toolkit. Upon user authentication, input data including compounds, fractions, microstructure, properties are transmitted to the backend, consisting of a physical reduced model that correlates inputs with predicted mechanical properties. The surrogate model (SM) serves as a simplified and computationally efficient representation of the physical models, capturing the relationships between input variables and output responses in real time. Data results will be visualized and stored accordingly.

- 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

6.8.1 Data handling

6.8.1.1 Data Storage

- **Simulation data:** the input, the output, and configuration data as the name and version of the subrogated model will be stored at the database.

6.8.1.2 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
CMBD	MMS	CSV, JSON	MBs	Custom developed APIs
MMS	CMBD	CSV, JSON	MBs	Custom developed APIs
MM	MMS	CSV, JSON	MBs	Custom developed APIs
MMS	MPS	CSV, JSON	MBs	Custom developed APIs
MMS	DTPC	CSV, JSON	MBs	Custom developed APIs
UI	Data storage	CSV, JSON	MBs	Backend

Table 118. Data Flows DIMMS

6.8.1.3 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.

6.8.1.4 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.

6.8.2 Technologies employed

Python libraries will be used for the following functionalities:

- Machine-learning tasks: e.g., scikit-learn will be used for developing real time and surrogate models. This component was evaluated in D2.2 as Python SVMs library and dimensionality reduction. The open-source solution LIME will complement scikit learn for giving interpretability and explainability.
- Pandas will be used for efficient data manipulation handling modelling inputs, outputs and results.
- Matplotlib and Plotly will generate detailed visualizations of the modelling results.
- Numerical packages as numpy, scipy will provide robust numerical computing capabilities required for performing scientific calculations
- The atomistic models will be developed with LAMMPS, a high-precision molecular dynamics simulations software for modeling material behavior at atomic scales. This open-source code supports parallel computing and GPU acceleration, crucial for large-scale simulations in material analysis, as pointed in D2.2. Custom Python scripts will be used to pre-process molecular dynamics simulations, ensuring setup and parameterization. OVITO will complement LAMMPS by visualizing and analyzing simulation data, enabling post-processing of atomic-scale simulations.
- Finite element models will be developed with open-source software such us CALCULIX, selected for its ability to handle complex analyses. Its open-source nature, compatibility with large-scale models, and parallel computing capabilities make it an ideal solver. As stated in D2.2, it can be used for linear and nonlinear calculations, static, dynamic and thermal analysis. FREECAD and Gmsh, also evaluated in D2.2, will be utilized. FREECAD and Gmsh, also evaluated in D2.2, will be utilized. FREECAD will be used for generating geometries, and Gmsh for meshing the bodies. Both tools offer Python libraries for automating the process, enabling the creation of complex geometries and meshes for accurate structural or mechanical simulations. Paraview, along with custom Python scripts, will be used for visualizing and interpreting simulation results, aiding analysis and communication of findings within the project.
- Docker for dockerizing the application.

6.8.3 Implementation maps

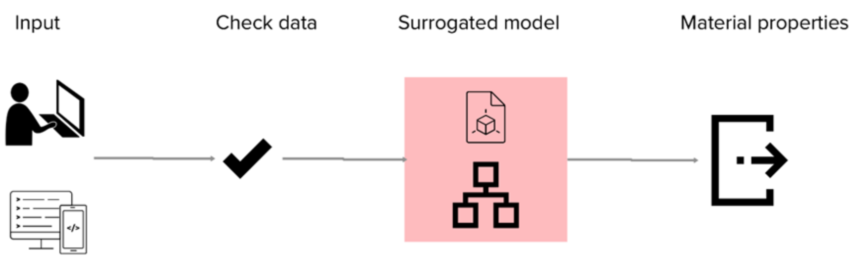


Figure 52. Implementation Map of Di^{MMS}

The following figure offers a comprehensive overview of the implementation of the mechanical surrogate model. It begins with user input or input from an application providing the model variables. The data format and variable values are validated to prevent inputs beyond the trained range. Subsequently, the model maps these variables to material properties, which become the output of the process.

6.8.4 User Manuals

In the following table, the devices required for the MMS toolkit are presented.

Activity	Necessary devices	Functionality
Molecular dynamic simulations	Server	Perform numerical simulations using molecular dynamic software as LAMMPS.
Finite element model development	Server	Perform numerical simulations using freeware finite element software.
Results visualization	Server	3D visualizations of numerical simulations
Subrogated model training	Server	Machine learning training of the reduced model based on a ANN methodology or equivalent.

Table 119. User Manuals of DiMMS

6.8.4.1 Servers

Servers play a central role in the construction and deployment of the MMS toolkit, serving as the backbone for various functions. These functions encompass the execution of physical models such as finite element models and molecular dynamic models, facilitating both preprocessing and postprocessing numerical activities, the automation of processes with Python scripts, and the visualization of numerical results. They also serve as platforms for the training of reduced models, enabling the optimization of computational resources. Ultimately, servers will be used for the deployment of the MMS toolkit.

6.8.4.2 Recommended computing resources for solution

The minimum and optimum characteristics of a server to run molecular dynamic simulations with LAMMPS can vary depending on the size and complexity of the simulation. The next values can be an initial guideline:

- **Minimum**: CPU dual-core or quad-core processor with 8GB RAM and 100 GB hard drive.
- **Optimum**: CPU multi-core processor with 32 GB RAM or more and 500 GB SSD. GPU with CUDA support can significantly speed simulations.

Similarly, for running finite element simulations with CALCULIX the next values can be used as reference:

- **Minimum**: Quad-core processor (Intel Core i5 or equivalent) with 8GB RAM and 100 GB hard drive. The operating system should be compatible with CALCULIX as Windows or Linux
- **Optimum**: CPU multi-core processor with 32 GB RAM or more and 500 GB SSD. GPU with CUDA support can significantly speed simulations. Linux-based OS are recommended for scientific computing

For training a reduced model the characteristics of a server vary depending on the size of the dataset and complexity of the model. The next values are an initial reference.

- **Minimum**: Multi-core processor (Intel Core i5 or equivalent) with 8GB RAM and 100 GB hard drive, and basic GPU with CUDA support
- **Optimum**: Multi-core processor with 32 GB RAM or more and 500 GB SSD, and high-end NVIDIA GPU with CUDA support.

6.8.5 Deployment PATTERN

The solution will be delivered as a Dockerized application consisting of Docker containers and can be hosted in the cloud or installed locally. It is important to consider that for specific real-time 3D visualization features, local installation might be required.

6.8.5.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Add new project	HTTP	/api/	POST	JSON	JSON
Run new prediction	HTTP	/api/prediction	POST	JSON	JSON
Delete prediction	HTTP	/api/prediction/{id}	DELETE	-	
View output	HTTP	/api/prediction/view	PATCH	JSON	JSON
Delete results	HTTP	/api/prediction/view/{id}	DELETE	-	-
Download results	HTTP	/api/prediction/view/{id}	GET	-	JSON

Table 120. API DIMMS

6.9 DIMAT MATERIALS PROCESSING SIMULATION (DI^{MPS})

The diagram below illustrates the internal architecture of the MPS toolkit. User will log in using the credentials, and the interaction with UI. The integration of input such as materials properties, geometry parametrization, boundary conditions, and principal processes parameters will be update for user or provided by other toolkits primary for DiCMDB and DiMMS. In this point, surrogate model servers will give support to the user with a recommendation relationship between upload data and output selection. The output obtained for the simulation will be visualized for the validation of the user and stored for feed the data base and increase the prediction of the toolkits. Data storage should be introduced to DiCMDB to increase the functionality of the rest of toolkit and continue developing new accurate predictions.

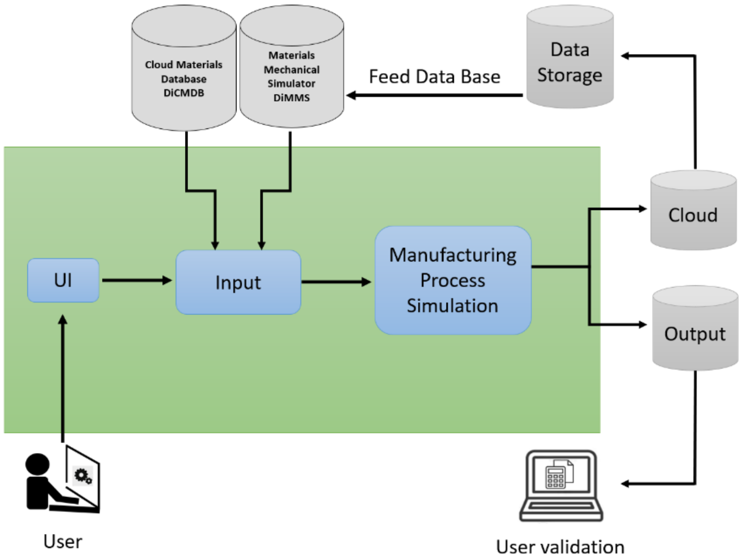


Figure 53. Internal Architecture DiMPS

6.9.1 Data handling

6.9.1.1 Data Storage

For a better understanding of the data, a cloud-based, collaborative approach with the other toolkits would be desirable. Providing the user with a complete and high-value material study. The simulation data input, such as parameterized geometries used in simulations, recommended mesh type, processing conditions parameters, as well as the output data of previous simulations, will be stored in the database, which is considered necessary to facilitate the use of the toolkit. Moreover, keeping the database updated for feeding the other toolkits.

6.9.1.2 Data Analysis

Depending on the type of material and transformation process, the user will be asked to enter a specific input required for the process simulation. Computational fluid dynamics calculations will be selected depending on the type of fluid, resulting in output with the principal parameter of the process and the behaviour of the material during processing. The outputs will be generated by an interpolation algorithm developed from the results of previous simulations.

Co-funded by the European Union

6.9.1.3 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
CMBD	MMS	CSV, JSON	MBs	Custom developed APIs
MMS	CMBD	CSV, JSON	MBs	Custom developed APIs
MM	MMS	CSV, JSON	MBs	Custom developed APIs
MMS	MPS	CSV, JSON	MBs	Custom developed APIs
MMS	DTPC	CSV, JSON	MBs	Custom developed APIs
UI	Data storage	CSV, JSON	MBs	Backend

Table 121. Data Flow DiMPS

6.9.1.4 Data Preprocessing

For data processing, a multivariate interpolation algorithm has been developed, based on previously executed simulation results.

6.9.1.5 Data Analytics

The output data are based on an interpolation algorithm that uses a previous database from simulated processing conditions.

6.9.2 Technologies employed

- Due to its multiple advantages, such as the capacity to integrate different code languages (enabling cross-platform development), widely applicable, a large number of users that provide fast responses to any requirements, and its easy memory management, Python libraries are suitable choices for implementing the toolkit within the DiMAT project, in which the integration and interoperability of vast functionalities must be united. Therefore, the following functionalities will be used:
- Data management: Pandas, it offers useful features that promote an easy process of writing and reading data. In addition, it´s applicable to diverse domains.

- Visualizations generation: Matplotlib, plotly. These libraries allow an easy way to access large amounts of data, offering the possibility of creating accurate plots from it. In addition, they include different types of configuration settings that make them extensive and customizable. Numerical packages: numpy, scipy. Among the advantages to be implemented in the development of the toolkits, is the efficient array manipulation with high performance of computation. Moreover, SciPy can handle advanced mathematical computations like signal processing or statistical modelling.
- Machine-learning tasks: TensorFlow, Pytorch. As stated in the first version of the Viewpoint's deliverable, these are open-source libraries among the most popular, which have a high level of documentation, high-performance computing with self-differentiation capability, and the ability to be customizable and extensible.
- Docker for dockerizing the application.
- Fluid dynamic solver: OpenFoam. As it was described in the D2.2 (parametric evaluation) and the first version of the Viewpoint's deliverable, OpenFoam constitutes a powerful toolbox for numerical problem solutions, including thermal complex equations, which makes it suitable for analysis of materials behavior when temperature parameter is involved.
- Finite element preprocessor: CALCULIX, PrePoMax, LS-Prepost, TexGen, FreeCad, Gmesh, python scripts. All these preprocessors (freeware and compatible with different platforms) offer a wide range of functions, such as the capacity to import and export models and geometry in different formats. In addition, they allow the creation of 2D and 3D sketches, as well as the automatically generated meshes.
- Finite element postprocessor: Paraview. This software and technologies adhere to what was described in D2.2 for benchmarking. It is used in the visualization of simulation results, providing parallel processing and handling large amounts of data. to this purpose, it is suitable due to its high adaptability and process efficiency. Moreover, it is a license-free software.

6.9.3 Implementation maps

The following figure shows a comprehensive overview of the Di^{MPS} implementation. Here, the user selects, through the User Interface (front-end), the appropriate parameters according to the simulation model. The data format and variable values are filtered/validated to prevent inputs other than the trained range. Subsequently, simulation calculation is based on a multivariate interpolation algorithm developed from previous simulation data. Then the forecasted values are passed to the UI and are displayed to the users in a friendly format.

Figure 54. Implementation Map of DiMPS

6.9.4 User Manuals

Activity	Necessary devices	Functionality
Molecular dynamic simulations	Server	Perform numerical simulations using molecular dynamic software such as LAMMPS.
Finite element model development	Server	Perform numerical simulations using freeware finite element software.
Results visualization	Server	3D visualizations of numerical simulations
Subrogated model training	Server	Machine learning training of the reduced model based on an ANN methodology or equivalent.

Table 122. User Manuals DiMPS

6.9.4.1 Servers

The MPS toolkit is deployed on a single server using Docker. The server will handle the execution of all component models, such as finite element models facilitating both preprocessing and postprocessing numerical activities, the automation of processes with Python scripts, and the visualization of numerical results.

6.9.4.2 Recommended computing resources for solution

The minimum and optimum characteristics of a server to run Computational Fluid Dynamics with **OpenFoam** depending on the size and complexity of the simulation. The next requirements are an initial guideline:

- **Minimum**: CPU dual-core or quad-core processor with 8GB RAM and 100 GB hard drive.
- **Optimum**: CPU multi-core processor with 32 GB RAM or more and 500 GB SSD. GPU with CUDA support can significantly speed simulations.

The minimum and optimum characteristics for running finite element **simulations** with CALCULIX the next requirements are a guideline:

- **Minimum**: Quad-core processor (Intel Core i5 or equivalent) with 8GB RAM and 100 GB hard drive. The operating system should be compatible with CALCULIX as Windows or Linux
- **Optimum**: CPU multi-core processor with 32 GB RAM or more and 500 GB SSD. GPU with CUDA support can significantly speed simulations. Linux-based OS are recommended for scientific computing

For training a reduced model requirement of server depending on the size of dataset and **complexity** of the simulation model.

- **Minimum**: Multi-core processor (Intel Core i5 or equivalent) with 8GB RAM and 100 GB hard drive, and basic GPU with CUDA support
- **Optimum**: Multi-core processor with 32 GB RAM or more and 500 GB SSD, and high-end NVIDIA GPU with CUDA support.

6.9.5 Deployment PATTERN

The Di^{MPS} toolkit will be deployed as a Dockized app using a Docker container as an executable package, including the calculation code. As Docker allows the application to be deployed in any environment due to compatible features and enhance computing power, it could be installed on a local server or hosted in the cloud repository. This concept is the same as the one implemented in the Di^{MMS} toolkit.

6.9.5.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
ExampleAPI	HTTP	https://mps.dimat-tools.eu/	POST	JSON, PNG	JSON

Add new project	HTTP	/api/	POST	JSON	JSON
Run new prediction	HTTP	/api/prediction	POST	JSON	JSON
Select prediction options	HTTP	/api/prediction	POST	JSON	JSON
Delete prediction	HTTP	/api/prediction/{id}	DELETE	-	
View output	HTTP	/api/prediction/view	PATCH	JSON	JSON
Delete results	HTTP	/api/prediction/view/{id}	DELETE	-	-
Download results	HTTP	/api/prediction/view/{id}	GET	-	JSON

Table 123. API DiMPS

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

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 an InfluxDB time-series database. 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.

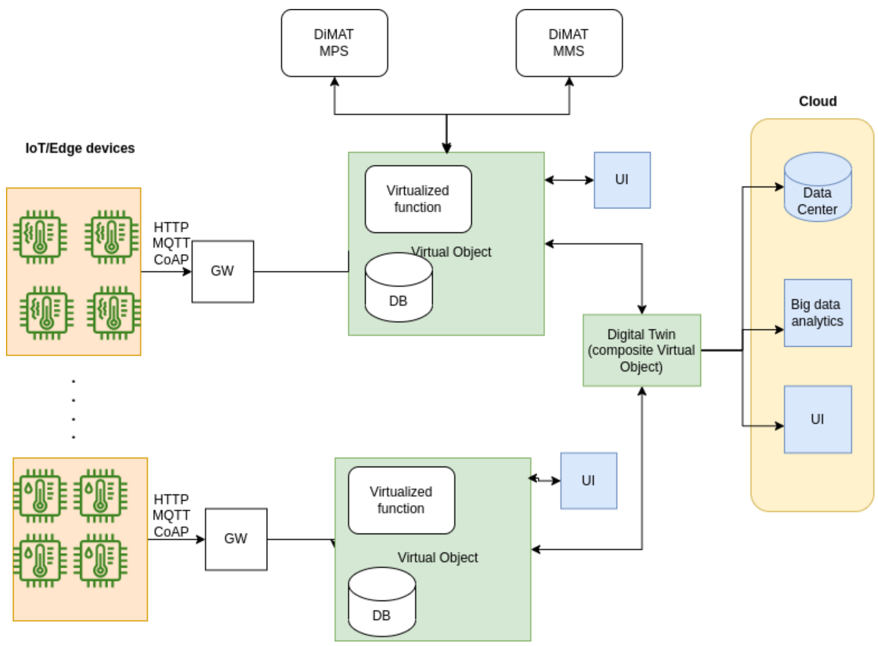


Figure 55. Internal Architecture DIDTPC

6.10.1 Data handling

6.10.1.1 Data Storage

- **Data from the equipment (real-time/historical):** The Digital Twin relies heavily in time-series analysis. To this end, the InfluxDB database is employed for data storage. InfluxDB offers dashboards for visualizing time-series as well as Python libraries that allow communication with the Virtual Functions developed in this language.
- **Configuration data:** Data pertaining to the physical counterpart's properties and the connection with the Digital Twin (e.g., sampling rate, IP address, APIs) are stored in special yaml files that can be accessed by the DT (Virtual Objects).

6.10.1.2 Data Flows

Source component	Target Component/Toolkit	Data type	Data size estimation	Method
IoT devices	Influx DB	CSV, JSON	MBs	HTTP API, MQTT

InfluxDB	Virtual Functions	Python data structures (dictionaries)	MBs	Influx DB Python client
Virtual Object	Virtual Object/Digital Twin	CSV, JSON	MBs	HTTP API
Digital Twin	Other toolkits	CSV, JSON, RDF	MBs	Custom developed APIs

Table 124. Data Flows DIDTPC

6.10.1.3 Data Analytics

The Digital Twin for Process Control toolkit can support by default some data analytics virtual functionalities that are essential for every type of digital twin. Due to the nature of a Digital Twin being very specific to the use cases of the end-user (Consortium pilot, enterprise, etc.) and the corresponding equipment available, more detailed functionalities can be developed by using the toolkits mentioned in the following section (10.3). Regarding the basic analytics supported out of the box, these can be the following:

Virtual Function	Details
Alerting functionalities	Based on real-time data and rules specified by the user, the DTPC can perform analysis on the current state of the equipment and alert when the rules are violated either through actions on the user's side (e.g., turn on a LED, flip a switch) or via specified communication channels (e.g., e-mail to an admin account).
Historical data analysis	Analysis of past data to discover correlations, extract information about optimal parameter selection, etc.
Visualization functions	Ability to create figures based on specified parameters provided by the user. Dashboard capabilities for more general overview of the measured resources.

Table 125. Data Analytics DIDTPC

6.10.2 Technologies employed

- NEPHELE VO software stack. As also mentioned in D2.2 and the first version of the Viewpoints deliverable, it is a lightweight software stack developed as part of the NEPHELE project, by NTUA which are also partners of DiMAT,

- Open-source popular Python libraries can be used for developing algorithms for the following functionalities:
    - Forecasting: e.g., Prophet, statsmodels, etc.
    - Visualizations generation: e.g., Matplotlib, seaborn, plotly, etc.
    - Machine-learning tasks: e.g., TensorFlow
- Interactive 3D visualizations: Mayavi. Mayavi has the benefit of providing 3D visualizations and due to its Python implementation, it is easy to incorporate it w'InfluxDB time-series databases for storing the current state of the Physical Twin. InfluxDB is one of the most popular time-series databases. It is open-source, provides a friendly user interface that allows for basic data handling. Moreover, it is supported out of the box with the Nephele VO software stack.
- NiceGUI for simple User Interface design. NiceGUI is an open-source Python framework used to design interfaces. Its strong point is the ease with which it can interact with Python scripts.
- Vue.js for front-end design. The reasons behind Vue.js selection are also mentioned in the KAF toolkit.'Grafana for dashboard export to webpage. Grafana is a leading provider for dashboard creation as indicated by Github statistics with more than 64K stars and 12K forks. It has an open-source version and a very active surrounding community. Moreover, its minimum requirements of 500MB RAM and 1 CPU core are ideal even for deployment near the network edge.
- Docker for dockerizing the application. Docker allows for deploying the application without ease accross any device, regardless of the environment in which they were developed.

6.10.3 Implementation maps

In addition to the implementation map presented in the first version (sending a command to a device) we also present how the different components interact to provide two more functionalities. The first one is forecasting, where the user requests to see the evolution of time-series stored in the InfluxDB of a Virtual Object. The user selects through the User Interface (front-end) the appropriate parameters that are passed through the backend to the appropriate forecasting algorithm that retrieves data from InfluxDB. Then the forecasted values are passed to the UI and are displayed to the users in a friendly format.

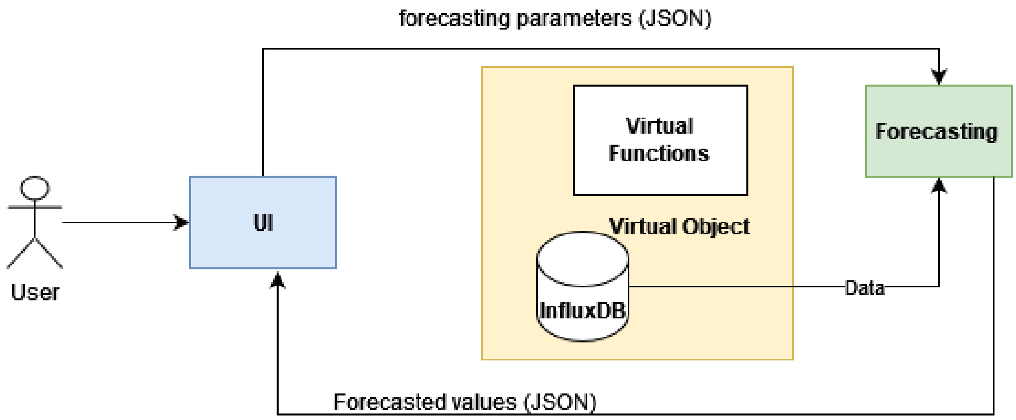


Figure 56. Implementation Map DIDTPC

In the next map the capability of the users to see the evolution of the metrics captured by one (or more) VOs is displayed. In a similar manner as before the Grafana service retrieves data from one or more instances of the InfluxDB database, produces the dashboard and exports it as a webpage that the users are transferred to in the front-end.

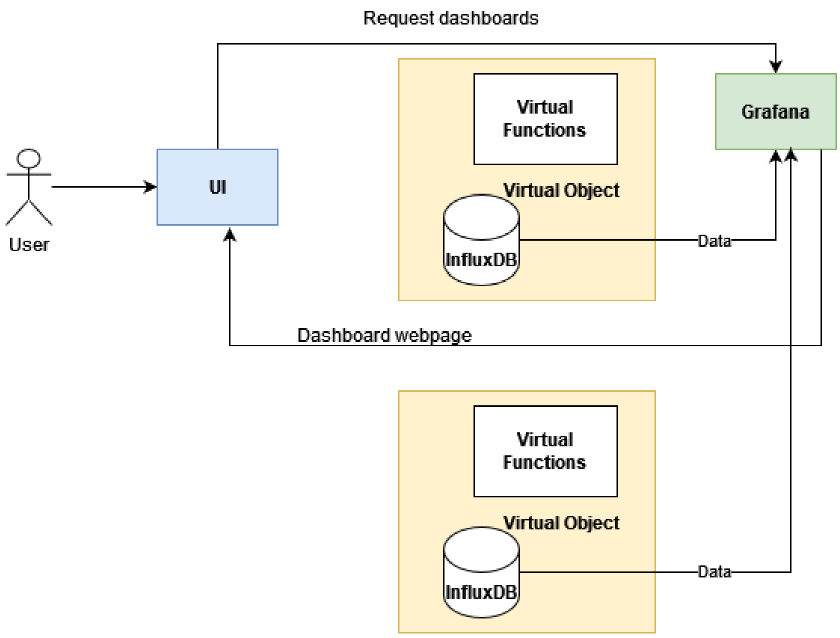


Figure 57. Implementation Map 2 DIDTPC

6.10.4 User Manuals

In the following table, specific devices required for the DTPC toolkit to be able to fulfil its functionalities are presented. It is important to keep in mind that not necessarily all the devices should be installed per activity, but at least a sufficient subset depending on the users' specific use cases and requirements.

Activity	Necessary devices	Functionality
Get current state of equipment View Digital Twin Status	Temperature sensors	Capture environmental conditions
	Humidity	
	Pressure sensor	Capture pressure applied by a tube on a bladder
	Power meter	Energy consumption
	Other internet enabled sensors/IoT devices	Monitor user-specified characteristics
Send command to a device	Actuator devices	Perform actions on the physical equipment
Device registration	IoT device with internet access	Needed in order to register the new device's address and establish communication
Ring an alarm (as part of other activities)	Iot device with internet access	In case the notifications are sent as an alarm sound on the user's side, a buzzer connected to an IoT device is necessary to be installed.
Perform time-series data analysis	Sensors that produce measurements Indicatively: Thermocouple, pressure sensors, power meters	Real data (either real-time or at least historical data) are necessary to implement meaningful data analysis

Table 126. User Manual DIDTPC

Co-funded by the European Union

6.10.4.1 Servers

Setting up a server is necessary for deploying the DTPC toolkit locally. The server executes the Nephele VO stack and the necessary virtual functions.

6.10.4.2 Network Equipment

Internet connectivity is crucial for the communication between the actual equipment and the Digital Twin. To this end, the users wanting to deploy this toolkit should have any of the following to establish an Internet connection: Wireless Access Points (WaP), Routers or access to LTE network. Moreover, in case that multiple measurements are generated and there is need to retrieve it through a single point, then a Gateway would be an efficient solution.

6.10.4.3 Recommended computing resources for solution

When deploying the DTPC toolkit, the user should consider that the main elements that require computing resources, since the NEPHELE VO stack is very lightweight to implement, are the Virtual Functions that demand resources depending on the needs of the user. For simple monitoring purposes, even a device such as a Raspberry Pi 4 is sufficient. For the majority of the possible use cases, a server with 2vCPUs and 4GBs of RAM should be sufficient for implementing the relevant functions and storing/retrieving and visualizing this data from an InfluxDB instance. This setup is adequate for running the basic functionalities offered from the toolkit out of the box.

However, for deploying more complex functionalities such as 3D visualizations, predictive maintenance, etc. more resources are needed. To be able to cover most cases we recommend at least 4vCPUs and 16GBs of RAM. The end-user should also consider if the functionalities require GPU.

6.10.5 Deployment PATTERN

The solution will be provided as a dockerized application consisting of docker containers. The solution can be hosted in the cloud or installed locally. It is important to notice though that for some real-time 3D visualization functionalities local installation could be necessary.

6.10.5.1 APIs description

API	Protocol	Endpoint/Topic	Method	Request data format	Response data format
Nephele VO API measured resources	HTTP	:port/measuredResources	GET	-	JSON
Nephele VO API other metrics of VOs	HTTP	:port/metricName	GET	-	JSON
Metrics stored in InfluxDB	HTTP	/metrics	GET	-	JSON
Forecasting	HTTP	/forecasting	GET	JSON	JSON
Set ranges for time-series alerts	HTTP	/alerts	GET	-	-

Table 127. API DIDTPC

7 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 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.

From a business perspective, stakeholder expectations and needs have been identified and addressed to ensure that the solutions developed reflect business and regulatory objectives. Usability has been a major focus, structuring the solutions in a way that makes them accessible and efficient for end users, facilitating their adoption and optimizing the user experience. In terms of functionality, the document has described in detail how each component of the DiMAT solutions interacts and contributes to a technological ecosystem, enabling seamless integration and operability. Finally, the implementation of these solutions has been planned to ensure technical integration and efficient deployment.

8 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] Industrial Internet Reference Architecture (IIRA) - https://hub.iiconsortium.org/iira

[3] Functional Viewpoint - https://hub.iiconsortium.org/portal/FunctionalViewpoint/593c45ba74e589000f715381

[4] Fastapi website

[5] PostgreSQL website

[6] SQLAlchemy website

[7] NiceGUI website

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 and LIST OF TABLES for pages.
1.2	Figure 2 Change	
1.3	Figure 4 Change	
1.4	Figure 7 Change	
1.5	Figure 10 Change	
1.6	Figure 13 Change	
1.7	Figure 17 Change	
1.8	Figure 19 Change	
1.9	Figure 21 Chage	
1.10	Figure Table 11 Change	
1.11	Figure Table 14 Change	
1.12	Figure Table 28 Change	
1.13	Figure Table 29 Change	
1.14	Figure Table 30 Change	
1.15	Figure Table 31 Change	
1.16	Figure Table 33 Change	
1.17	Figure Table 34 Change	
1.18	Figure Table 44 Change	
1.19	Figure Table 70 Change	
1.20	Figure 44 Change	
1.21	Figure 47 Change	
1.22	Figure 49 Change	
1.23	Figure 50 Change	
2	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 5.4.1,5.5.1), the necessary updates have been made to reflect the interactions with the components of the reference architecture.
3	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 6.1.2, 6.2.2, 6.3.2, 6.4.2, 6.5.1,6.6.2, 6.7.2, 6.8.2, 6.9.2, 6.10.2. 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 128. Changes in the deliverable based on the reviewers' comments

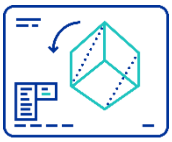
APPENDIX B

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.

Co-funded by
the European Union

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					

Co-funded by the European Union

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	

Co-funded by
the European Union