



**Politecnico
di Torino**

Politecnico di Torino

Master's Degree in Digital Skills for Sustainable Societal Transitions

**A Digital Tool to Support Project-Based Learning:
The Case of an Interactive Dashboard for Decision-
Making Workshops**

Master's Thesis

Candidates

Alireza Soleiman Bejarpasi

Setareh Ghorbani

Supervisors

Prof. Sara Torabi Moghadam - DIST

Prof. Gianvito Urgese - DIST

Virginia Pellerey - DIST

Academic Year

2025-2026

Abstract

Digital technologies are increasingly embedded in higher education, especially in courses based on active learning, project work, and collaborative workshop processes. However, when these courses rely on multiple disconnected tools, course materials, visual collaboration, spreadsheets, communication channels, external decision-support tools, and final outputs can become distributed across separate platforms. This fragmentation can weaken workflow continuity, reduce output traceability, increase cognitive load, and make it more difficult for professors and teaching assistants to follow the development of group reasoning.

This thesis addresses this problem through the design, development, validation, and refinement of an interactive educational platform for a Decision-Making course at Politecnico di Torino. The research adopts Design Thinking as its methodological framework, moving from educational needs analysis and problem definition to technological pathway exploration, prototype development, user validation, and fine-tuning. The proposed strategy is based on a layered Moodle–Django platform logic: Moodle is understood as the institutional Learning Management System (LMS) layer, while Django provides a custom interactive workshop layer for project-based activities that require specialised interfaces, data continuity, visual interaction, and reviewable outputs.

The developed prototype organises student work around persistent project workspaces and translates the course methodology into a connected sequence of workshop and support modules. These include project framing, stakeholder analysis, problem and objective structuring, indicator selection and ranking, indicator analysis, SWOT analysis, scenario building, scenario analysis, external decision-support tools, cost–benefit analysis, and final project review. Together, these modules are designed to reduce tool-switching, preserve intermediate outputs, and support a more coherent workshop progression.

The prototype was validated through an in-person formative evaluation session with six former students of the target course and the course professor as an expert academic observer. The validation combined direct observation, Likert-scale survey responses, open-ended feedback, and criterion-based thematic analysis. The findings showed a positive reception of the integrated project workspace and confirmed the educational relevance of workflow continuity, data continuity, and output traceability. At the same time, participants identified areas for refinement, including navigation consistency, clearer validation messages, stronger

save feedback, improved guidance in complex modules, collaboration support, and more complete reporting functions.

The thesis contributes a refined proof-of-concept prototype and a platform strategy for reducing digital fragmentation in workshop-based higher education. It does not claim full institutional deployment or completed Moodle integration through Learning Tools Interoperability 1.3 (LTI 1.3). Instead, it demonstrates the educational relevance and technical plausibility of a Django-based custom interactive workshop layer that can complement an institutional LMS and support more coherent, traceable, and pedagogically aligned project-based learning workflows.

Keywords

Digital Fragmentation; Project-Based Learning; Workshop-Based Learning; Learning Management System; Moodle–Django Platform Strategy; Design Thinking; Workflow Continuity

Acknowledgements

We would like to express our sincere gratitude to our supervisors, Prof. Sara Torabi Moghadam, Prof. Gianvito Urgese, and Virginia Pellerey, for their guidance, feedback, and support throughout the development of this thesis. Their academic advice and constructive comments were essential in shaping the direction of the research and improving the quality of the final work.

We are also grateful to Prof. Sara Torabi Moghadam, as the professor of the Decision-Making course, and to Virginia Pellerey for her teaching support during the course and validation process. We would also like to thank the former students who participated in the validation session. Their observations, comments, and feedback provided valuable insights for evaluating and refining the prototype.

Finally, we would like to thank our families and friends for their encouragement, patience, and support during this academic journey.

Table of Contents

Abstract	2
Keywords.....	4
Acknowledgements	5
Table of Contents	6
List of Figures	8
List of Tables.....	10
List of Abbreviations.....	11
Chapter 1. Introduction.....	12
1.1 Research Context and Problem Statement.....	12
1.2 Institutional Context and Research Case	14
1.3 Research Questions and Objectives.....	15
1.4 Scope, Delimitations, and Thesis Structure.....	17
Chapter 2. Methodology.....	19
2.1 Methodological Framework	19
2.2 Case Study, Materials, and Data.....	23
2.3 Methodological Outputs	25
Chapter 3. Theoretical and Conceptual Foundation	29
3.1 Institutional and Educational Context	29
3.2 Workshop-Based Learning and Multi-Stage Decision-Making Workflows	31
3.3 User Roles and Current Digital Practices	35
3.4 Digital Fragmentation and Platform Requirements.....	39
3.5 Maturity and Implementation Readiness Analysis.....	43
3.6 Result of the Maturity Analysis and Platform Strategy.....	48
3.7 Technical Objectives of the Platform	50
3.8 Outcome of the Theoretical and Conceptual Phase.....	52
Chapter 4. Prototype Development	55

4.1 From Educational Requirements to Platform Design	55
4.2 System Architecture	58
4.3 Technology Stack and Development Environment	61
4.4 Django Project Structure	63
4.5 Data Model and Persistence Strategy	65
4.6 User Workflow and Project Dashboard.....	69
4.7 Workshop Module Implementation.....	71
4.8 Interface Design and Interaction Logic	88
4.9 Export and Reporting Functions.....	90
4.10 Technical Limitations and Future Development	93
Chapter 5. Validation and Fine-Tuning.....	97
5.1 Validation and Fine-Tuning within the Design Thinking Process	97
5.2 Validation Design and Setup	98
5.3 Validation Session and Real-Time Observations	103
5.4 Survey Responses and Thematic Analysis	107
5.5 Evaluation Against the Validation Criteria	114
5.6 Fine-Tuning and Prototype Refinement	116
Chapter 6. Discussion.....	122
6.1 Discussion of Findings	123
6.2 Theoretical Contribution	127
6.3 Practical Implications	130
Chapter 7. Conclusion	133
7.1 Conclusive Remarks and Key Findings.....	133
7.2 Limitations.....	135
7.3 Future Development Directions.....	136
References	139

List of Figures

Figure 1. Fragmented digital tool usage in workshop-based higher education.	13
Figure 2. Main consequences of digital fragmentation in workshop workflows.	14
Figure 3. Thesis structure and research progression across the seven chapters..	18
Figure 4. Research questions, Design Thinking stages, methods, and outputs.	20
Figure 5. Design Thinking-based research design of the thesis.	21
Figure 6. Educational context of the target course at Politecnico di Torino.	30
Figure 7. Current multi-stage workshop workflow and associated tool transitions.	35
Figure 8. interaction needs and fragmentation gaps in the current course environment.	37
Figure 9. Mapping from fragmentation problems to platform requirements.	41
Figure 10. Maturity and implementation readiness matrix of standalone technological pathways.	47
Figure 11. Derivation of the Moodle–Django platform strategy from the maturity analysis.	49
Figure 12. Requirement-to-design translation in the prototype.....	57
Figure 13. Hybrid Moodle–Django architecture. Source: Authors.	60
Figure 14. Django platform architecture. Source: Authors.	61
Figure 15. Django project structure and application logic.	65
Figure 16. Simplified data model of the platform.	67
Figure 17. Main project dashboard and workshop cards.....	71
Figure 18. Workshop 1 Idea Canvas interface.	73
Figure 19. Workshop 2.1 Stakeholder Analysis interface.	74
Figure 20. Workshop 2.2 Problem Tree interface.	75
Figure 21. Workshop 2.3 Objective Tree interface.	76
Figure 22. Workshop 3.1 Indicator Selection interface.....	77
Figure 23. Workshop 3.2 Indicator Ranking / SRF interface.....	78

Figure 24. Workshop 4.1 Indicator Analysis interface.....	79
Figure 25. Workshop 4.2 SWOT Analysis interface.....	80
Figure 26. Workshop 5 Q-Methodology scenario-building interface.	81
Figure 27. Workshop 5 Start Park scenario-building interface.	83
Figure 28. Workshop 6 Scenario Analysis interface.	84
Figure 29. Workshop 7 External Decision-Support Tools interface.	85
Figure 30. Workshop 8 Cost–Benefit Analysis / CBA interface.....	86
Figure 32. Participants navigating the workshop sequence on the localhost prototype at validation session.....	101
Figure 33. Mean Likert scores per survey statement, based on seven survey responses .	109
Figure 34. Open-ended survey responses coded by validation criterion.....	110

List of Tables

Table 1. Overview of research aim, objectives, and research questions with corresponding thesis locations.....	16
Table 2. Adapted Design Thinking framework across the thesis phases	21
Table 3. Case study, materials, and data sources used in the research.....	25
Table 4. Digital tools currently used in the course and their contribution to workflow fragmentation.....	38
Table 5. Prioritised requirements for the proposed integrated platform.....	43
Table 6. Technical objectives derived from literature, requirements analysis, and maturity analysis.	52
Table 7. Translation of platform requirements into prototype design responses	57
Table 8. Technology Stack of the Prototype	62
Table 9. Main Data Models and Their Role in the Platform	68
Table 10. Workshop modules and their main prototype functions.....	87
Table 11. Export Functions Across the Platform	92
Table 12. Technical limitations and future development directions.	95
Table 13. Validation criteria and methodological alignment	99
Table 14. Survey Instrument: Questions, Type, and Criterion Alignment.....	102
Table 15. Summary of real-time usability observations by prototype area.....	106
Table 16. Descriptive statistics for Likert-scale survey statements, n = 7	109
Table 17. Criterion-by-criterion evaluation of the prototype	115
Table 18. Fine-tuning interventions: issue, revision, and validation trigger	119
Table 19. Fine-tuning outcomes mapped to validation criteria	121

List of Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
API	Application Programming Interface
CBA	Cost-Benefit Analysis
CSV	Comma-Separated Values
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
LMS	Learning Management System
LTI	Learning Tools Interoperability
NPV	Net Present Value
PDF	Portable Document Format
PNG	Portable Network Graphics
SRF	Simos-Roy-Figueira
SWOT	Strengths, Weaknesses, Opportunities, and Threats
UI	User Interface
UX	User Experience
WCAG	Web Content Accessibility Guidelines

Chapter 1. Introduction

1.1 Research Context and Problem Statement

Digital technologies increasingly shape how higher education is organised, delivered, and experienced. This transformation is particularly visible in courses built around active learning, project work, and iterative processes, where digital environments directly affect the organisation of work, the continuity of activities, and the quality of interaction among participants. Over the last two decades, higher education has moved from the simple use of digital repositories and online communication channels toward a broader rethinking of teaching and learning practices. Universities increasingly adopt digital tools not only to extend traditional teaching, but also to create more flexible, interactive, and student-centred learning environments (Bond et al., 2018; Petchamé et al., 2023).

This transformation has been shaped by technological change, evolving labour-market expectations, and growing attention to digital competences as a core dimension of advanced education. These developments are often framed through the concept of Education 4.0, which connects pedagogical innovation with the increasing integration of digital infrastructures into institutional life (Miranda et al., 2021; Salmon, 2019). However, the existence of digital tools does not automatically produce coherent or meaningful learning environments. In many institutions, digital solutions have been accumulated over time in response to immediate needs rather than through a unified pedagogical design. As a result, universities may appear technologically equipped while still operating through fragmented digital ecosystems (Henderson, Selwyn & Aston, 2017).

This issue becomes particularly important in project-based and workshop-based learning. These pedagogical approaches require students to work through connected stages of analysis, collaboration, decision-making, and iterative production (Thomas, 2000; Kokotsaki, Menzies and Wiggins, 2016). Unlike transmissive teaching models, project-based and workshop-based learning depend on continuity: each phase builds directly on the outputs of the previous one. The educational value of these approaches is therefore inseparable from their cumulative structure. Students cannot produce a well-reasoned final deliverable unless the reasoning developed at earlier stages has been preserved, made visible, and carried forward coherently.

The central problem addressed by this thesis is the fragmentation of the digital ecosystem that supports collaborative academic work. In many university settings, students and instructors rely on a patchwork of separate tools, each serving a limited function: one platform hosts course materials, another supports file exchange, another manages communication, and others are adopted informally for group coordination or visual work (Henderson, Selwyn & Aston, 2017; Petchamé et al., 2023). This arrangement may appear practical in the short term, but it generates structural weaknesses when activities are interdependent and extend across multiple stages.

The fragmented structure described above is summarised in Figure 1, which presents a general view of how learning activities can become distributed across separate digital tools without a central workflow hub.

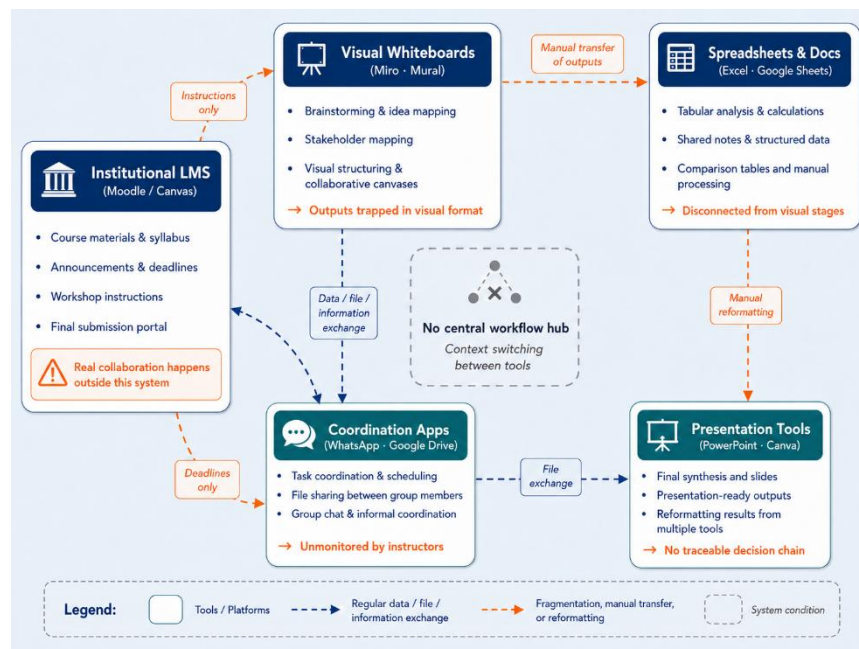


Figure 1. Fragmented digital tool usage in workshop-based higher education.

Source: Authors.

Fragmentation affects learning in four interconnected ways. First, it disrupts workflow continuity, as users must repeatedly move between systems to reconstruct the state of the work. Second, it reduces traceability, making it difficult to understand how decisions were reached and how outputs evolved across stages. Third, it weakens collaboration, since communication, task management, and feedback are separated in ways that can make group coordination uneven and partially invisible to instructors. Finally, it creates accessibility barriers, as managing multiple interfaces increases cognitive demand, particularly for users

with limited technical experience. Figure 2 summarises the main consequences of digital fragmentation for workflow continuity, traceability, collaboration, and accessibility.

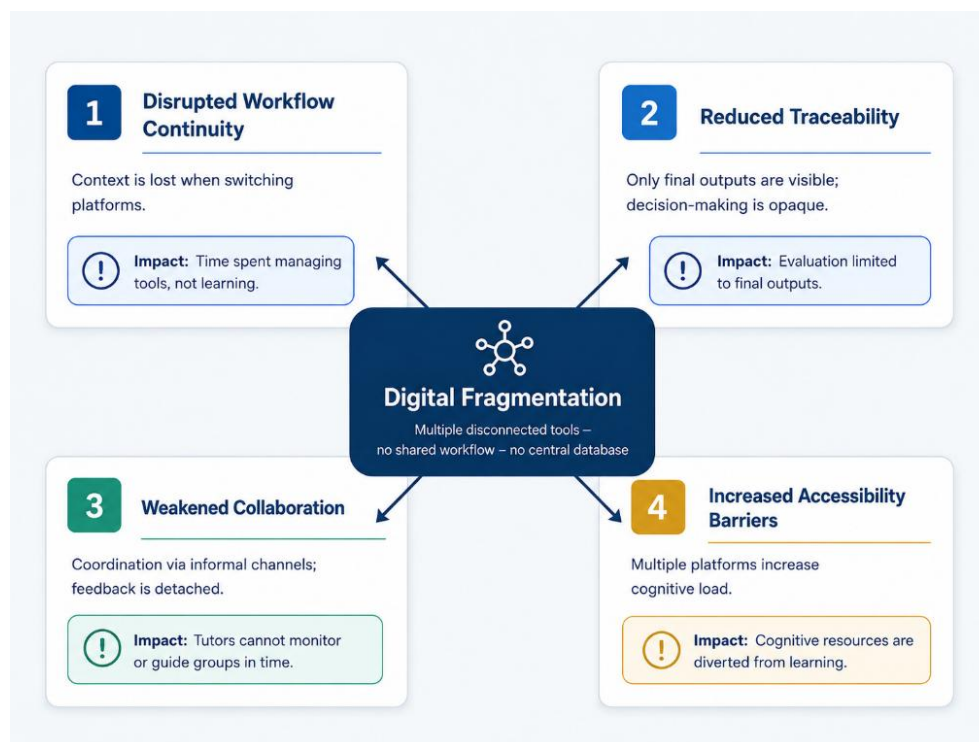


Figure 2. Main consequences of digital fragmentation in workshop workflows.

Source: Authors.

1.2 Institutional Context and Research Case

The relevance of this problem is particularly evident at Politecnico di Torino, a large technical university that operates within a structured digital ecosystem combining a formal institutional Learning Management System (LMS) with a range of complementary tools. Within this setting, many courses increasingly involve applied, collaborative activities in which students work in groups, produce outputs across multiple phases, and depend on continuity between stages.

This thesis focuses on the graduate-level course Evaluation Methods and Decision-Making Approaches at Politecnico di Torino. The course is organised around a collaborative decision-making project in which student groups analyse a complex urban or campus-related problem, identify stakeholders, structure problems and objectives, select and rank indicators, build alternative scenarios, evaluate their performance, and synthesise the results into a final project review. The course is therefore not based on isolated assignments, but on a

cumulative workshop sequence in which each activity depends on outputs produced in earlier stages.

This structure makes the course a suitable case for investigating fragmented digital support and developing a more integrated platform response. In the prototype, the original workshop logic is translated into a more granular set of digital workshop modules and support modules, allowing each methodological step to be supported through a dedicated interface while remaining part of the same project-based workflow.

Rather than proposing a solution detached from the institutional context, the thesis works within the existing university infrastructure. This makes the research practically relevant beyond the prototype stage, because the proposed platform is conceived not as a replacement for the institutional LMS, but as a complementary interactive workshop layer able to support activities that conventional LMSs do not fully accommodate.

The platform considered in this thesis is intended to serve three user groups: students, professors, and teaching assistants. Students require a coherent workspace in which they can develop, save, revise, and connect workshop outputs across the full project sequence. Professors require visibility over the development of group reasoning, not only access to final submitted outputs. Teaching assistants require tools that allow them to monitor progress, provide guidance, and support students during intermediate stages of the workflow. A useful platform must therefore address these roles within a shared and coherent digital environment.

In the current proof-of-concept prototype, the implemented workflow focuses primarily on the student-side project workspace, while more advanced professor and teaching-assistant dashboards are treated as future development directions.

1.3 Research Questions and Objectives

The thesis is guided by two research questions:

RQ1. How can an interactive educational platform conceptually support a complete, multi-stage collaborative workshop workflow in higher education?

RQ2. How should an interactive educational platform be technically designed and developed within an institutional university environment?

These questions reflect the dual nature of the thesis. The first question concerns the conceptual and pedagogical problem: how an interactive platform can respond to the

fragmentation of workshop-based learning by supporting continuity, collaboration, accessibility, and traceability. The second question concerns the technical and design problem: how such a platform can be implemented within a realistic institutional setting, using an architecture that connects university infrastructure with custom interactive workshop functionality.

Methodologically, the thesis adopts Design Thinking as the main framework for moving from user needs and problem definition to technological exploration, prototype development, validation, and refinement. This methodological structure is presented in detail in Chapter 2.

The main objective of the thesis is to design, develop, and evaluate an integrated interactive platform supporting a complete collaborative workshop-based workflow, with the aim of reducing digital fragmentation and improving continuity, accessibility, collaboration, and traceability at Politecnico di Torino.

This objective is articulated into three sub-objectives. The first is to identify the educational and technical requirements of the target course context and determine the most appropriate technology strategy and platform architecture for an integrated institutional solution. The second is to design and develop a functional platform prototype implementing the identified requirements. The third is to evaluate the prototype through qualitative user validation and targeted refinement.

Table 1. Overview of research aim, objectives, and research questions with corresponding thesis locations.

Category	Content	Chapters
Main Objective	Design, develop, and evaluate an integrated interactive platform supporting a complete collaborative workshop-based workflow, reducing digital fragmentation and improving continuity, accessibility, collaboration, and traceability at Politecnico di Torino.	Ch. 1–7
Sub-objective 1	Identify the educational and technical requirements of the target course context and determine the most appropriate technology strategy and platform architecture for an integrated institutional solution.	Ch. 3
Sub-objective 2	Design and develop a functional platform prototype implementing the identified requirements.	Ch. 4

Sub-objective 3	Evaluate the prototype through qualitative user validation and targeted refinement.	Ch. 5
RQ 1	How can an interactive educational platform conceptually support a complete, multi-stage collaborative workshop workflow in higher education?	Ch. 3,6
RQ 2	How should an interactive educational platform be technically designed and developed within an institutional university environment?	Ch. 4,5,6

Source: Authors.

1.4 Scope, Delimitations, and Thesis Structure

The thesis does not aim to deliver a fully institutionalised production system or a universally applicable platform for all teaching contexts. The prototype functions as a proof of concept and does not include large-scale infrastructure rollout, long-term organisational adoption, or production-level deployment within the university infrastructure. The study is grounded in a single institutional context, Politecnico di Torino, and in one specific target course. For this reason, the findings are context-specific and are not intended to be statistically generalisable.

The scope of the thesis is nevertheless broader than the development of a technical artefact alone. The research examines the educational problem of digital fragmentation, identifies the requirements of a workshop-based course, evaluates alternative technological pathways through a maturity and implementation readiness analysis, develops a prototype response, and validates the prototype through user feedback. The platform is therefore treated as both a technical output and a research artefact through which assumptions about workflow continuity, collaboration, accessibility, and traceability can be explored and assessed.

The thesis follows a seven-chapter structure. Chapter 1 introduces the research problem, institutional context, research questions, objectives, scope, and thesis structure. Chapter 2 presents the methodological approach, including the Design Thinking framework, the case-study context, the research materials and data sources, and the activities conducted across the three research phases. Chapter 3 provides the theoretical and conceptual foundation of the thesis by analysing the educational context, user needs, digital fragmentation, existing tool categories, maturity and implementation readiness, and the rationale for the selected platform strategy. Chapter 4 documents the prototype development process, including the

platform architecture, technology stack, data model, user workflow, workshop modules, interface design, and export functions. Chapter 5 presents the validation and refinement process, including the validation design, user session, survey results, thematic analysis, evaluation findings, and prototype improvements implemented in response to user feedback. Chapter 6 discusses the findings in relation to existing literature and examines their theoretical contribution and practical implications. Chapter 7 concludes the thesis by summarising the main findings, outlining the principal limitations of the study, and identifying future development directions.



Figure 3. Thesis structure and research progression across the seven chapters.

Source: Authors.

Chapter 2. Methodology

This chapter presents the methodological framework adopted in the thesis. The research is structured around Design Thinking, used as a human-centred and iterative approach for moving from problem understanding to solution development, validation, and refinement. This methodological choice reflects the nature of the research problem: the issue addressed in the thesis is not only technical, but also pedagogical, organisational, and user-centred. The central problem concerns the fragmentation of digital tools used to support a collaborative, multi-stage workshop-based course. In response, the thesis develops and evaluates an integrated educational platform designed to improve workflow continuity, accessibility, collaboration, and traceability.

The chapter is organised into three sections. Section 2.1 presents the methodological framework and explains how Design Thinking is adapted into three research phases. Section 2.2 introduces the case study, materials, and data sources used in the research. Section 2.3 summarises the methodological outputs produced by the three phases and explains how they structure the following chapters of the thesis.

2.1 Methodological Framework

This thesis adopts Design Thinking as its main methodological framework. In this research, Design Thinking is understood not only as a general creative approach, but as a structured process for developing and refining a solution in response to a complex, user-centred problem. It is particularly relevant when the problem is not fully defined at the beginning and when the final artefact must respond to user needs, organisational constraints, technical feasibility, and iterative feedback. For this reason, Design Thinking has been widely used in design-oriented and innovation contexts where the aim is to move from problem understanding to a tested and refined solution (Brown, 2008; Brown and Wyatt, 2010; Dorst, 2011; Liedtka, 2015; Plattner, Meinel and Leifer, 2011).

Design Thinking is appropriate for this thesis because the research does not begin from a predefined technological solution. Instead, it begins from an educational problem observed in the target course: the use of multiple disconnected tools to support a collaborative decision-making workflow. The methodology therefore starts by understanding the course context, the needs of students, professors, and teaching assistants, and the difficulties created

by the existing fragmented workflow. It then moves toward problem definition, exploration of possible technological pathways, prototype development, validation, and fine-tuning.

The relationship between the research questions, the Design Thinking stages, the methodological activities, and the outputs produced by the thesis is summarised in Figure 4.

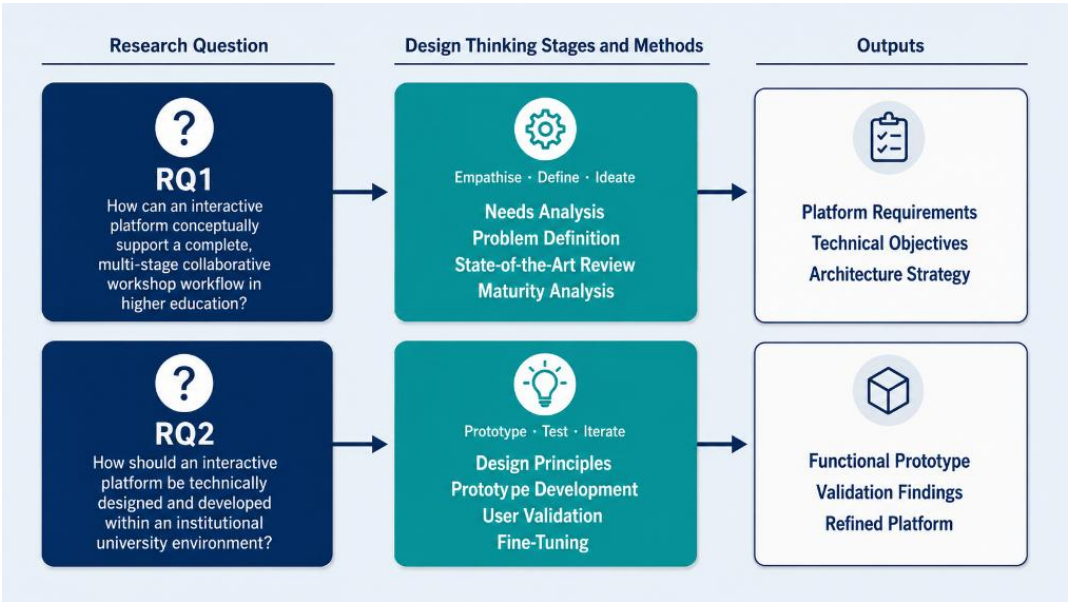


Figure 4. Research questions, Design Thinking stages, methods, and outputs.

Source: Authors.

As shown in Figure 4, the methodology links each research question to a specific phase of the research design. The first research question is mainly addressed through the conceptual and analytical work of the Empathise, Define, and Ideate stages, while the second research question is addressed through prototype development, validation, and refinement. The figure also clarifies that the methodology is not limited to technical implementation, but includes problem framing, technological pathway selection, user validation, and iterative improvement.

The three-phase adaptation of Design Thinking is summarised in Table 2. The table shows how each phase is connected to specific Design Thinking stages, methodological focus, and thesis chapters.

Table 2. Adapted Design Thinking framework across the thesis phases

Research phase	Design thinking stage(s)	Main methodological focus	Chapter
Theoretical and conceptual foundation	Empathise, Define, Ideate	Analysis of the course context, user needs, fragmentation problem, existing tools, and candidate technological pathways	Chapter 3
Prototype development	Prototype	Translation of requirements into design principles, system architecture, data model, interface logic, and workshop modules	Chapter 4
Validation and fine-tuning	Test, iterate	User validation session, survey feedback, thematic analysis, and targeted refinement	Chapter 5

Source: Authors.

Table 2 clarifies the role of each phase in the overall research design. Phase 1 produces the conceptual and technical basis for the platform, Phase 2 translates this basis into a functional Django prototype, and Phase 3 evaluates and refines the prototype through user validation. In this way, the thesis moves from educational problem analysis to platform development and from prototype implementation to evidence-based refinement.

The overall Design Thinking-based research design of the thesis is presented in Figure 5.

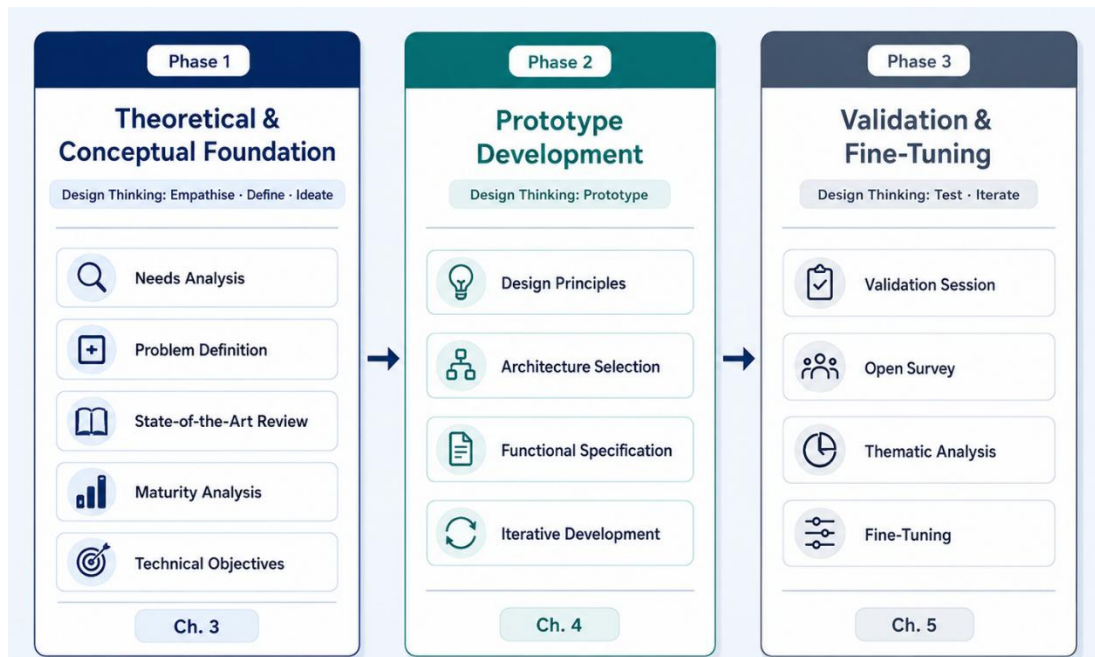


Figure 5. Design Thinking-based research design of the thesis.

Source: Authors.

Figure 5 shows how the thesis is organised as a progressive but iterative research process structured into three methodological phases. Although the thesis chapters are presented in a linear order, the methodological logic is iterative: the understanding of the educational problem informs the prototype design, the development process reveals technical and usability issues, and the validation findings guide further refinement.

Phase 1 - Theoretical and Conceptual Foundation corresponds to the Empathise, Define, and Ideate stages of Design Thinking. In this phase, the research focuses on understanding the educational context, defining the fragmentation problem, and identifying the most suitable technological direction for the platform. This phase establishes the conceptual and technical basis for the prototype.

Phase 2 - Prototype Development corresponds to the Prototype stage. In this phase, the requirements and technical objectives derived from the first phase are translated into a functional Django-based platform. The purpose of this phase is to transform the conceptual framework into a testable research artefact.

Phase 3 - Validation and Fine-Tuning corresponds to the Test and Iterate stages. In this phase, the prototype is evaluated through formative validation and refined according to user feedback. The purpose of this phase is to assess whether the prototype responds to the intended educational workflow and to identify targeted improvements.

This three-phase structure allows the thesis to move from educational problem analysis to platform development and from prototype implementation to evidence-based refinement. In this sense, Design Thinking provides both the methodological logic of the thesis and the structure through which the research artefact is developed and evaluated.

The methodology also has a design-oriented and mixed-methods character. Qualitative and interpretive elements are used to analyse the educational context, user needs, validation observations, and open-ended feedback. At the same time, structured comparative analysis is used to assess technological pathways through the maturity and implementation readiness analysis. This combination allows the study to address both research questions: how an interactive platform can conceptually support a complete multi-stage collaborative workshop workflow, and how such a platform can be technically designed and developed within an institutional university environment.

2.2 Case Study, Materials, and Data

The empirical case study of this thesis is the graduate-level course Evaluation Methods and Decision-Making Approaches at Politecnico di Torino. The course was selected because it is organised around a collaborative, multi-stage workshop workflow in which student groups progressively develop a decision-making project. The course requires students to move from project framing to stakeholder analysis, problem and objective structuring, indicator selection and ranking, scenario building, scenario evaluation, cost–benefit analysis, and final project synthesis. This structure makes the course particularly suitable for investigating digital fragmentation, because each workshop stage depends on outputs produced in previous stages.

The course is not based on isolated assignments. Instead, it follows a cumulative project-based logic in which intermediate outputs must remain available, understandable, and reusable across the full workflow. For example, stakeholders identified in the early stages may influence the interpretation of objectives and scenarios; indicators selected and ranked during the middle stages inform later scenario analysis; and scenario-building outputs provide the basis for final evaluation and project review. When these activities are distributed across disconnected tools, students and instructors must manually reconstruct the relationship between stages. For this reason, the course provides a relevant case for examining how a digital platform can support workflow continuity, data continuity, collaboration, accessibility, and output traceability.

The case study involves three main user groups: students, the course professor, and teaching assistants. Students are the main active users of the prototype because they create and develop project workspaces, complete workshop modules, save outputs, and prepare final synthesis materials. The professor represents the course-design and evaluation perspective, requiring visibility over the development of group reasoning and the relationship between intermediate outputs and final results. Teaching assistants represent the support and monitoring perspective, requiring access to group progress and opportunities to provide guidance during the workflow. Although the current proof-of-concept prototype focuses mainly on the student-side workflow, the needs of professors and teaching assistants are considered in the requirements analysis and future development directions.

The materials used in the research can be grouped into four categories: course-related materials, conceptual and theoretical materials, prototype-development materials, and

validation materials. Course-related materials include the structure of the Decision-Making course, workshop descriptions, existing exercises, methodological steps, and discussions with the professor and tutors. These materials supported the understanding of the educational context and the identification of the fragmentation problem.

Conceptual and theoretical materials include academic literature on digital transformation in higher education, project-based learning, workshop-based learning, collaborative learning, Learning Management Systems, digital fragmentation, usability, accessibility, technology acceptance, software quality, and maturity analysis. These materials were used to frame the problem theoretically and to justify the criteria used for platform requirements and technological pathway selection.

Prototype-development materials include the artefacts produced during the design and implementation of the Django-based platform. These include architecture decisions, data models, Django views and templates, interface structures, workshop-module definitions, export logic, and implementation notes. These materials document how educational needs and platform requirements were translated into a functional prototype.

Validation materials include observations from the in-person validation session, survey responses collected through Google Forms, Likert-scale ratings, open-ended comments, thematic grouping notes, and fine-tuning records. These materials were used to evaluate the prototype according to six criteria: usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. The validation data were not intended to support statistical generalisation. Instead, they were used for formative prototype evaluation and targeted refinement.

The validation involved former students of the target course and the course professor as an expert academic observer. Former students were suitable participants because they had already experienced the original workshop workflow and could compare the prototype with the previous use of separate tools such as Moodle, visual boards, spreadsheets, shared files, and external decision-support tools. The professor's participation added an instructional and methodological perspective, especially in relation to course coherence, evaluation needs, and possible institutional usefulness.

The validation process was treated as low-risk educational research. Participants were not current students being assessed in the course, reducing the risk of academic pressure. Participation was voluntary, and the feedback was used for research and prototype-

improvement purposes. The analysis focused on aggregated responses and recurring patterns rather than individual performance. Since the prototype could in the future handle student data, academic outputs, and institutional authentication, the thesis also recognises the importance of secure data handling, role-based access, and GDPR-compliant deployment in any future institutional implementation.

The case study, materials, and data sources used in the thesis are summarised in Table 3.

Table 3. Case study, materials, and data sources used in the research.

Methodological phase	Data sources and materials	Role in the research
Theoretical and Conceptual Foundation	Course structure, workshop materials, professor and tutor input, direct course experience, academic literature	Supports the identification of user needs, fragmentation problems, and platform requirements.
Prototype Development	Architecture decisions, data models, interface specifications, workshop-module definitions, implementation artefacts	Documents how the identified requirements are translated into a functional platform.
Validation and Fine-Tuning	User testing observations, survey responses, thematic analysis notes, fine-tuning records	Evaluates the prototype and guides targeted improvements.

Source: Authors.

Table 3 shows how each group of materials supports a specific part of the research process. Course-related and conceptual materials support the first phase of the methodology by helping to identify the educational needs and platform requirements. Prototype-development materials support the second phase by documenting the transformation of requirements into a functional Django platform. Validation materials support the third phase by providing evidence for evaluating and refining the prototype.

2.3 Methodological Outputs

The three methodological phases described in Section 2.1 produce the main outputs that structure the rest of the thesis. These outputs should not be understood as final research findings in isolation, because the detailed analysis, prototype development, and validation results are presented in Chapters 3, 4, and 5. Instead, they represent the methodological results of each phase: the concrete analytical, design, technical, and evaluative outputs generated through the Design Thinking process.

Phase 1 - Theoretical and Conceptual Foundation

The first phase produces the theoretical and conceptual foundation of the thesis. It begins with the analysis of the target course, its workshop structure, user roles, and existing digital practices. The main output of the Empathise stage is an understanding of the educational context and of the needs of students, professors, and teaching assistants. This understanding shows that the target course depends on a cumulative decision-making workflow in which early outputs must remain available and meaningful for later stages.

The Define stage translates this contextual understanding into the central research problem of the thesis: digital fragmentation in a multi-stage collaborative workshop workflow. The main output of this stage is the formulation of the fragmentation problem in relation to workflow continuity, data continuity, collaboration, accessibility, and output traceability. This problem framing clarifies why the existing use of disconnected tools is not only an organisational issue, but also a pedagogical and methodological challenge.

The Ideate stage produces the platform requirements, maturity analysis, and technological strategy. During this stage, existing tool categories and possible technological pathways are examined in relation to the needs of the target course. The exploratory maturity and implementation readiness analysis identifies Moodle and Django as the two strongest and most complementary pathways. Moodle provides the institutional learning-management layer, while Django provides the custom interactive workshop layer. The main output of Phase 1 is therefore an operational design basis for the prototype: a Moodle–Django platform strategy supported by technical objectives related to workflow continuity, accessibility and inclusivity, collaboration support, traceability, integration readiness, scalability, and long-term feasibility. These outputs are developed in Chapter 3.

Phase 2 - Prototype Development

The second phase produces the functional prototype described in Chapter 4. This phase translates the requirements and technical objectives derived from Phase 1 into a Django-based educational platform. The main output is a working proof-of-concept prototype organised around project-based workspaces and connected workshop modules.

The prototype phase produces several concrete design and technical outputs. These include the system architecture, the Moodle–Django platform logic, the Django project structure, the data model, the persistence strategy, the user workflow, the project dashboard, the workshop modules, the interface design, and the export and reporting functions. The platform

implements course-specific modules such as Idea Canvas, Stakeholder Analysis, Problem Tree, Objective Tree, Indicator Selection, SRF Ranking, Indicator Analysis, SWOT Analysis, Scenario Building, Scenario Analysis, External Decision-Support Tools, Cost–Benefit Analysis / CBA, and Project Review.

The methodological importance of this phase is that it transforms the conceptual problem into a testable artefact. The prototype makes it possible to examine whether workflow continuity, data persistence, output traceability, and low-friction interaction can be supported through a custom interactive workshop layer. In this sense, the prototype is not only a technical product, but also a research artefact through which the assumptions developed in the theoretical and conceptual phase can be tested.

Phase 3 - Validation and Fine-Tuning

The third phase produces the validation findings and fine-tuning interventions presented in Chapter 5. The Test stage evaluates the prototype through an in-person formative validation session with former students of the target course and the course professor as an expert academic observer. The validation uses direct observation, Likert-scale survey responses, open-ended comments, and criterion-based thematic analysis.

The main output of the Test stage is an evaluation of the prototype according to six validation criteria: usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. These criteria allow the prototype to be assessed both as a digital interface and as an educational workflow environment. The validation findings show which aspects of the platform were perceived as useful and coherent, and which aspects required refinement.

The Iterate stage produces the final group of methodological outputs: the fine-tuning interventions. These refinements respond to issues identified during validation, including navigation reliability, form validation, save feedback, interface consistency, guidance in complex modules, reporting functions, and later-stage workflow completeness. The fine-tuning phase does not redesign the prototype from the beginning. Instead, it improves the existing proof of concept so that it more effectively supports the educational workflow identified in the earlier phases.

Together, the three phases create a coherent methodological progression. Phase 1 defines the problem and derives the platform strategy. Phase 2 translates this strategy into a functional prototype. Phase 3 evaluates and refines the prototype based on user feedback. This

progression ensures that the thesis does not move directly from a general problem to a technical solution. Instead, the proposed platform is grounded in educational needs, justified through conceptual and technological analysis, implemented as a functional artefact, and evaluated through formative user validation.

This methodological structure also clarifies how the research questions are addressed. The first research question, concerning how an interactive educational platform can conceptually support a complete multi-stage collaborative workshop workflow, is mainly addressed through the outputs of Phase 1 and further interpreted in Chapter 6. The second research question, concerning how such a platform should be technically designed and developed within an institutional university environment, is mainly addressed through the outputs of Phases 2 and 3 and further discussed in Chapter 6.

In summary, the methodology produces three connected outputs: a conceptual and technical foundation, a functional Django-based prototype, and a validated set of refinement priorities. These outputs provide the bridge between the research problem introduced in Chapter 1 and the detailed analysis, development, validation, discussion, and conclusion presented in the following chapters.

Chapter 3. Theoretical and Conceptual Foundation

This chapter establishes the theoretical and conceptual foundation of the thesis by connecting the educational context of the target course with the broader literature on digital transformation, project-based learning, workshop-based workflows, and fragmented digital ecosystems. Within the Design Thinking framework introduced in Chapter 2, this chapter corresponds to the Empathise, Define, and Ideate stages. It first examines the educational context and user needs, then defines the fragmentation problem affecting the current workshop workflow, and finally explores possible technological pathways through an exploratory maturity and implementation readiness analysis. The chapter concludes by deriving the technical objectives that guide the prototype development in Chapter 4.

3.1 Institutional and Educational Context

The empirical context of the thesis is Politecnico di Torino, a technical university that operates an institutional LMS supporting course administration, material distribution, communication, and formal submission management. This infrastructure provides an important institutional backbone. However, as documented in comparable higher education contexts, general-purpose LMS platforms are not always designed to accommodate the interactive demands of workshop-based and project-based courses, which require custom activity flows, collaborative input, and continuous traceability of group reasoning across multiple stages (Lonn & Teasley, 2009; Petchamé et al., 2023).

The course examined in this thesis is a graduate-level offering situated within a programme that prepares students for work in complex, interdisciplinary, and sustainability-oriented professional environments. The course is structured around a sequence of collaborative workshops in which students, organised into small working groups, analyse a real or realistic problem scenario, develop analytical frameworks, identify priorities, and produce structured outputs through progressive collective decision-making. This pedagogical structure distinguishes the course from conventional lecture-based formats because the learning process depends not only on content delivery but on the continuity of reasoning, data, decisions, and outputs across multiple stages.

The LMS serves as the official course hub: it provides access to session materials and instructions, manages formal submission points, and communicates deadlines and organisational information. However, because the LMS does not natively support the

workshop-specific interaction logic required by the course , including collaborative real-time input, structured sequential progression through analytical stages, and dynamic group-level traceability , participants routinely supplement it with other platforms. This supplementation produces a fragmented digital ecosystem in which different parts of the educational workflow are managed through disconnected systems. Students may use one platform for visual collaboration, another for file sharing and communication, and another for producing and exchanging documents. Tutors monitor group progress through informal channels that are not integrated with the LMS, while the professor receives outputs in various formats and from multiple sources. As a result, it becomes difficult to observe the full progression of group work coherently.

Figure 6 summarises the educational context of the target course and the relationship between the institutional LMS, course actors, workshop clusters, and external tools.

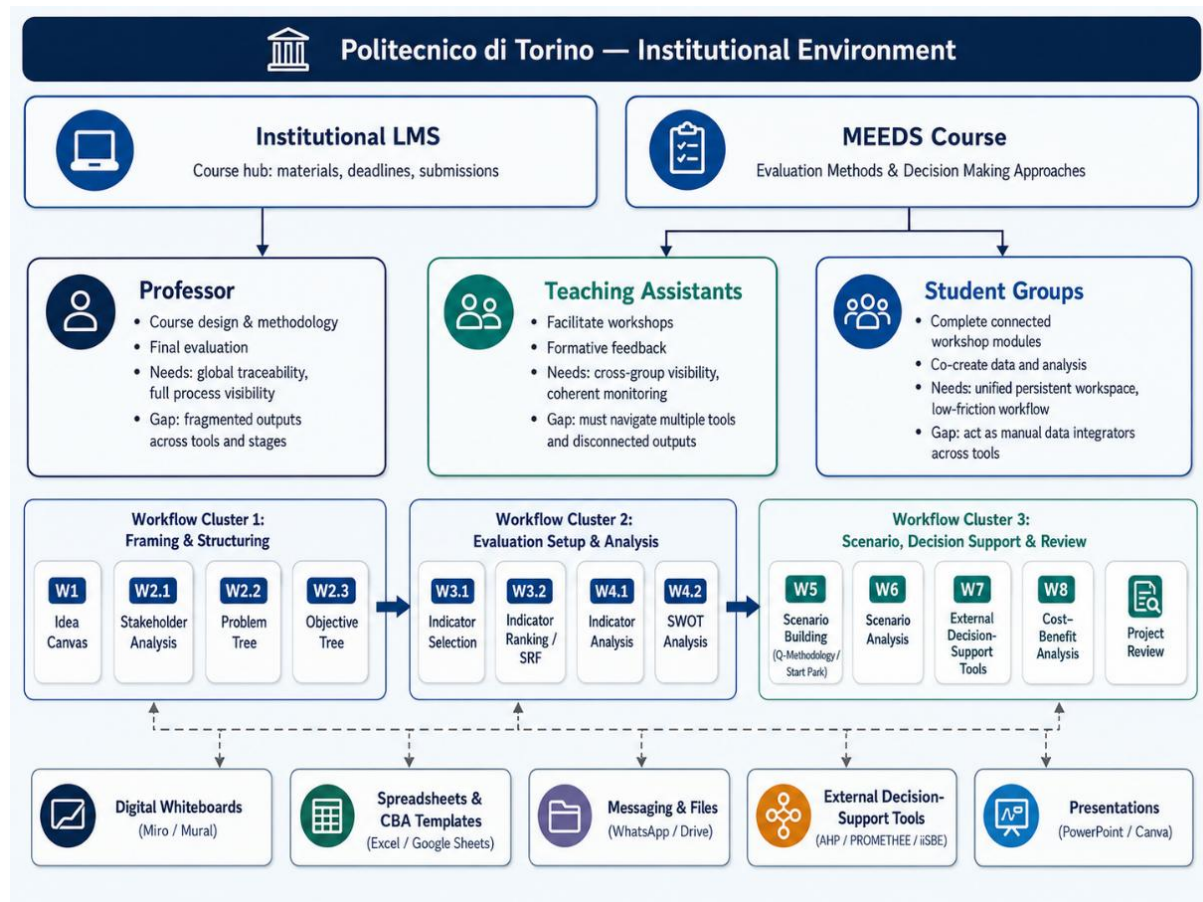


Figure 6. Educational context of the target course at Politecnico di Torino.

Source: Authors.

3.2 Workshop-Based Learning and Multi-Stage Decision-Making

Workflows

The course Evaluation Methods and Decision-Making Approaches, is organised around a structured structured decision-making process applied to real-life case studies. The specific empirical application may change from one semester to another, depending on the teaching focus, available materials, and selected project context. For example, one case study may concern the sustainable regeneration of campus structures at Politecnico di Torino, while other editions of the course may address different urban, environmental, infrastructural, or sustainability-oriented decision problems.

Within this structure, students are asked to navigate a multi-dimensional decision-making process: defining the problem context, identifying and classifying stakeholders, structuring problems and objectives, selecting indicators, generating alternative scenarios, evaluating their performance, and synthesising results into a final project review.

Because the course addresses complex public-sector and urban planning decisions, it places strong emphasis on methodological rigour. Students apply formal evaluation methods and structured decision-making frameworks, translating qualitative context into quantifiable indicators, strategic priorities, scenario alternatives, and evaluation results. The course structure therefore demands strong continuity: outputs generated in early analytical stages form the data foundation for the evaluative models applied in later stages. Any disruption in information flow or loss of data fidelity between tasks can compromise the group's ability to reach scientifically sound and traceable conclusions.

This logic is closely aligned with the literature on project-based and collaborative learning. Project-based learning is widely recognised as a student-centred pedagogical approach in which learners engage with authentic tasks, produce artefacts, and develop knowledge and skills through sustained activity over time (Zhang and Ma, 2023; Guo et al., 2020). Reviews and meta-analyses indicate that project-based learning can positively influence academic achievement, affective outcomes, higher-order thinking, and collaboration, although the effects depend on context and implementation quality (Zhang and Ma, 2023; Guo et al., 2020). A key feature of this approach is that learning does not occur through a single isolated act. Instead, students move through stages of problem framing, inquiry, analysis, ideation, revision, evaluation, and presentation, with each stage contributing to the quality of the final result.

Collaborative learning adds a further dimension. Research on collaborative online learning shows that the quality of group interaction is shaped by the tools and environments used by learners, not only by group composition or task design (Stahl, Koschmann and Suthers, 2006). Digital collaboration can be effective, but its success depends on the visibility of shared work, the coordination of contributions, and support for meaningful interaction rather than simple co-presence. Stahl, Koschmann and Suthers (2006) argue that technology must actively scaffold joint knowledge construction rather than merely provide a shared space.

Workshop-based courses represent a specific form of project-based learning in which students move through structured stages of inquiry, ideation, comparison, negotiation, evaluation, and decision-making. Such courses are workflow-intensive because intermediate artefacts are created, discussed, refined, and carried forward into later stages. The literature on online and hybrid learning also indicates that educational activities often combine synchronous interaction with asynchronous preparation and follow-up (Petchamé et al., 2023). This is particularly relevant in workshop-based formats, where live sessions are used for discussion and collective action, while asynchronous phases are used for review, reflection, consolidation, and preparation for later tasks.

In the target course, this pedagogical logic is operationalised through a connected sequence of workshop modules and support modules. The platform does not treat these activities as isolated tasks. Instead, it translates the course workflow into a continuous digital structure in which outputs from earlier modules can inform later analytical and evaluative activities.

Workshop 1 - Idea Canvas. Students define the initial project idea, describe the problem context, identify early intervention directions, and begin to frame the decision-making challenge. This stage establishes the conceptual starting point of the project and supports early shared understanding within the group.

Workshop 2.1 - Stakeholder Analysis. Students identify and classify the actors involved in or affected by the project. Stakeholders are analysed according to dimensions such as role, typology, resources, power, and interest. This module supports the transition from a general project idea to a structured understanding of the social and institutional environment.

Workshop 2.2 - Problem Tree. Students organise the central problem, its causes, and its effects through a hierarchical visual structure. This activity helps transform initial observations into an explicit causal model, making the logic of the problem more visible and discussable.

Workshop 2.3 - Objective Tree. Students translate the problem structure into a corresponding objective structure. Negative problem statements are reformulated into desired conditions, allowing the group to move from diagnosis to strategic orientation.

Workshop 3.1 - Indicator Selection. Students select relevant indicators for evaluating the project and its possible scenarios. This stage connects the qualitative understanding developed in earlier modules with measurable criteria that can support later assessment.

Workshop 3.2 - Indicator Ranking / SRF. Students rank and weight selected indicators using the Simos-Roy-Figueira method. The use of structured ranking supports transparent prioritisation and helps make group preferences explicit before scenario analysis.

Workshop 4.1 - Indicator Analysis. Students analyse selected indicators by collecting, entering, and interpreting supporting data. This stage strengthens the evidence base of the project and prepares the group for more structured comparison between alternatives.

Workshop 4.2 - SWOT Analysis. Students organise strengths, weaknesses, opportunities, and threats related to the project. This module supports strategic reflection by connecting internal project characteristics with external contextual conditions.

Workshop 5 - Scenario Building through Q-Methodology or Start Park. Scenario generation is supported through two alternative pathways. In the Q-Methodology pathway, students work with stakeholder-based Q-sorting and scenario extraction in order to model different perspectives and derive scenario interpretations. In the Start Park pathway, students generate spatial scenarios through stakeholder cards, solution cards, equipment cards, activity cards, site-plan placement, and scenario summary. These two pathways reflect different but complementary ways of constructing scenarios: one centred on stakeholder perspectives and factor interpretation, and the other centred on spatial configuration and design-based scenario composition.

Workshop 6 - Scenario Analysis. Students analyse and compare scenarios generated through either Q-Methodology or Start Park. This module supports scenario evaluation by allowing groups to select the scenario source and compare alternatives using the indicators and analytical structures developed in earlier stages.

Workshop 7 - External Decision-Support Tools. This module provides structured access to external decision-support tools selected by the professor, including AHP / Super Decisions, PROMETHEE, and the iiSBE / Sustainable MED Cities dashboard. It is not primarily an interactive data-entry workshop. Its purpose is to reduce navigation

fragmentation by centralising access to external tools within the platform workflow and making their role in the decision-making process clearer for students.

Workshop 8 - Cost–Benefit Analysis / CBA. This module transforms the course’s Excel-based cost–benefit analysis exercise into a guided digital workflow. Students can select scenarios from Q-Methodology or Start Park, enter cost items, benefit items, and assumptions such as time horizon, discount rate, and maintenance rate, calculate Net Present Value and Cost–Benefit Ratio, perform sensitivity analysis, and compare completed CBA analyses across scenarios.

Final synthesis and presentation. At the end of the decision-making process, students consolidate the outputs produced across the previous stages into a final project synthesis. This final output connects the initial problem framing, stakeholder analysis, objective structure, indicator selection, scenario development, scenario evaluation, and economic assessment into a coherent explanation of the group’s reasoning and recommendations.

Although the course is pedagogically organised around workshop-based decision-making, the prototype implements this logic through a more granular set of digital modules. This allows each analytical activity, such as stakeholder analysis, problem-tree construction, objective-tree development, indicator ranking, SWOT analysis, scenario building, external tool consultation, cost–benefit analysis, and final review, to be supported through a dedicated interaction logic while remaining part of the same overall workflow.

Because each module relies on outputs generated in previous stages, students carry a high coordination load when these activities are distributed across disconnected tools. They must translate qualitative brainstorming into structured taxonomies, convert stakeholder profiles into weighted criteria, connect indicators to scenario evaluation, and relate scenario-building outputs to cost–benefit analysis and final synthesis. For tutors and professors, fragmentation obscures the traceability of student reasoning, making it difficult to track how an initial idea evolved into indicators, scenarios, evaluation results, and final recommendations.

Figure 7 presents the current multi-stage workshop workflow and shows how different workshop activities are connected to separate external tools.

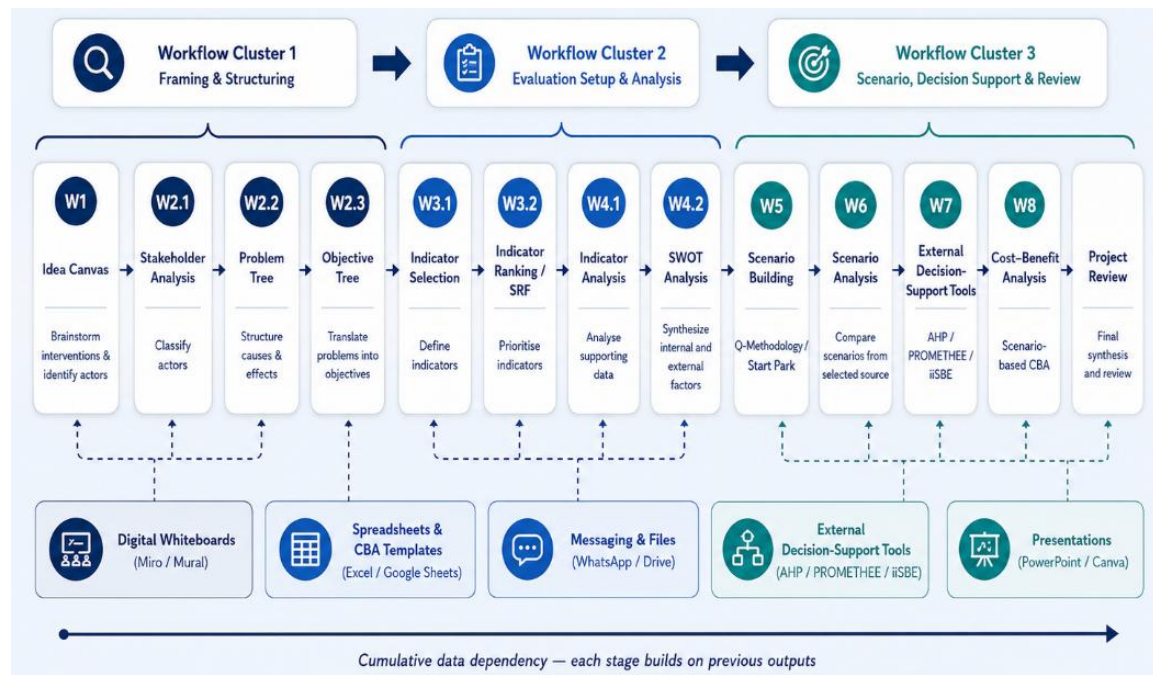


Figure 7. Current multi-stage workshop workflow and associated tool transitions.

Source: Authors.

3.3 User Roles and Current Digital Practices

The platform proposed in this thesis must respond to the needs of three main stakeholder groups in the course workflow: students, professors, and teaching assistants. These groups interact with the decision-making process differently, are affected by digital fragmentation in different ways, and require different levels of visibility, interaction, and access. However, the current proof-of-concept prototype is mainly implemented from the student-side workflow. Professors and teaching assistants are included in the requirements analysis because their monitoring, feedback, and evaluation needs are essential to the course, while dedicated dashboards and advanced role-based access remain future development directions.

User roles and access needs

Students are the primary active users of the current prototype. Their experience depends heavily on co-creation, cumulative data management, and the ability to revise outputs across stages. During the course, students engage in brainstorming, stakeholder mapping, problem and objective structuring, indicator selection and ranking, scenario building, scenario evaluation, and final synthesis. In the current fragmented ecosystem, students are often forced to act as system integrators: they manually transfer data from visual boards to spreadsheets, reformat outputs for later stages, and manage different file versions across

several tools. This creates a high risk of data loss, transcription error, version confusion, and unnecessary cognitive load. In platform terms, students therefore need a coherent project workspace where they can create and edit workshop outputs, save intermediate results, reuse data across modules, export visual or tabular outputs, and review the development of their project across the full workflow.

The professor serves as the principal architect and final evaluator of the course. This role includes defining the pedagogical structure, selecting evaluation methodologies, guiding the overall decision-making process, and assessing final group outputs. Unlike students, who work at the micro-level of individual workshop activities, the professor requires a macro-level view of the complete educational process. In a fragmented tool environment, the institutional LMS may capture the final submitted artefact, but it does not reveal the process through which that artefact was produced. If a student group reaches a final scenario recommendation, the professor should be able to understand how that conclusion relates to earlier stages such as stakeholder analysis, problem and objective structuring, indicator selection, and scenario evaluation. In the current prototype, this need is addressed mainly through reviewable project outputs and the preservation of intermediate artefacts, while full professor dashboards, assessment views, and advanced evaluation tools remain future development directions.

Teaching assistants act as the intermediary layer between the professor's course design and the students' practical execution. Their role is to facilitate workshop sessions, guide groups through complex methodological steps, and provide formative support before errors propagate into later stages. Fragmentation limits this monitoring capacity because teaching assistants may need to move between separate visual boards, spreadsheets, shared documents, and communication channels simply to understand the progress of one group. Feedback is also weakened when comments are separated from the specific artefacts or workflow stages to which they refer. In platform terms, teaching assistants require visibility over group workspaces, access to intermediate outputs, and, in future versions, tools for contextual feedback, progress monitoring, and stage-specific guidance.

The current proof-of-concept prototype addresses these role differences mainly through its workflow design rather than through a complete multi-role permission system. Students are the main implemented users who create and edit project workspaces, complete workshop modules, save intermediate outputs, and export results. Professors and teaching assistants are considered as review-oriented stakeholders whose needs inform the platform

requirements, especially in relation to traceability, feedback, and supervision. More advanced access capabilities, such as dedicated professor dashboards, teaching-assistant monitoring panels, contextual comments, stage locking, action logs, and detailed permission management, remain future development directions.

Figure 8 summarises the interaction needs of students, teaching assistants, and professors, together with the fragmentation gaps affecting each user group.

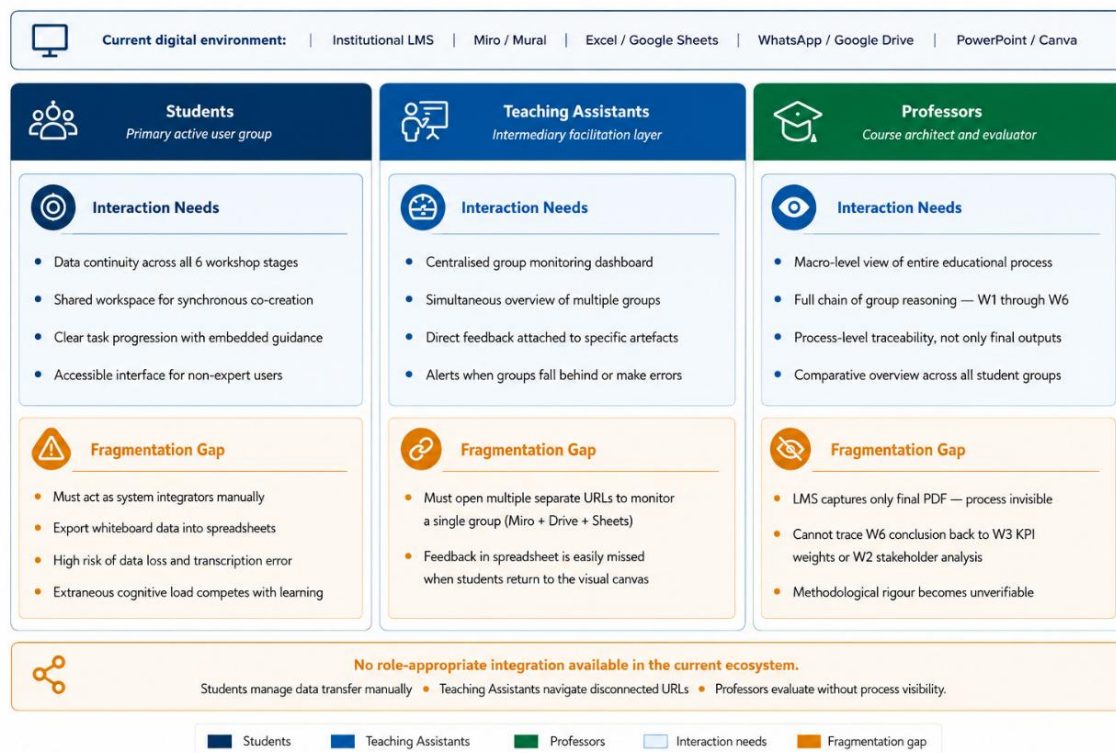


Figure 8. interaction needs and fragmentation gaps in the current course environment.

Source: Authors.

Current digital practices and fragmentation bottlenecks

The current digital environment supporting the course is not a single ecosystem but a patchwork of independent platforms. The institutional LMS provides the official course space for materials, announcements, deadlines, and formal submissions. However, because the LMS does not fully support the workshop-specific interaction logic required by the course, students and instructors rely on additional tools for visual collaboration, shared documents, quantitative analysis, scenario-related calculations, external decision-support tools, presentation preparation, and informal coordination. These tools are useful within their own domains, but their combined use creates fragmentation when outputs must move across multiple workshop modules and evaluative stages.

Table 4. Digital tools currently used in the course and their contribution to workflow fragmentation

Tool category	Example platforms	Primary uses in course workflow	Key limitations and fragmentation impact
Institutional LMS	Moodle / institutional LMS	Course materials, announcements, deadlines, formal submissions	Provides course governance but does not support bespoke interactive workshop modules, stage-to-stage data continuity, or the full decision-making workflow.
Visual whiteboards and canvases	Miro, Mural, Canva, shared visual boards	Idea Canvas, stakeholder mapping, problem/objective structuring, SWOT visual work, early-stage ideation	Outputs often remain trapped in visual formats and must be manually transferred into later analytical tools.
Spreadsheets and documents	Excel, Google Sheets, Google Docs, CBA Excel templates	KPI selection, Delphi/SRF weighting, scenario-related calculations, Cost–Benefit Analysis, shared notes	Disconnected from earlier visual and scenario-building work; requires manual re-entry and creates version-control risks.
Presentation tools	Powerpoint, Canva	Final deliverables, synthesis outputs, communication of results	Static outputs make it difficult to trace the reasoning and intermediate decisions behind final submissions.
External decision-support tools	AHP / Super Decisions, PROMETHEE, iisbe / Sustainable MED Cities dashboard	Multi-criteria decision support, sustainability assessment, external scenario evaluation support	Valuable for specialised analysis, but usually accessed outside the main course workflow, increasing navigation fragmentation.
Coordination apps	Google Drive, Email, whatsapp	File sharing, task coordination, informal group communication	Informal and unmonitored; tutors cannot easily track participation, feedback, or group progress.

Source: Authors

When the course progresses through its connected workshop modules, the burden of integrating these tools falls mainly on students. In the early framing and structuring modules, outputs produced through visual or informal tools may remain separated from the structured data needed in later analytical stages. Stakeholder classifications, problem-tree elements,

objective-tree logic, and early project assumptions often need to be transferred manually into spreadsheets or other documents before they can be reused for indicator selection, ranking, scenario analysis, or cost–benefit evaluation.

This manual transfer creates several risks. First, workflow continuity is weakened because students must repeatedly reconstruct the connection between earlier qualitative outputs and later quantitative analyses. Second, version-control problems can emerge when the same information is updated in one tool but not reflected in another. For example, if a missing stakeholder, indicator, or scenario assumption is identified later in the process, the group may need to revise multiple disconnected files or boards. Third, the logical connection between initial problem framing, scenario generation, external decision-support tools, cost–benefit analysis, and final synthesis may become distributed across several software environments, with no automated link between them.

The principal bottlenecks are therefore workflow discontinuity, loss of pedagogical traceability, increased cognitive load, and fragmented feedback. Workflow continuity is broken when outputs from one module must be manually re-entered into another environment. Pedagogical traceability is reduced because professors and teaching assistants may only see final submitted artefacts rather than the intermediate reasoning process that produced them. Cognitive load increases because students must manage several platforms, file versions, formats, and accounts while also completing the methodological tasks of the course, which competes directly with learning itself (Kirschner, Paas & Kirschner, 2009).

3.4 Digital Fragmentation and Platform Requirements

The course-specific problems described in the previous sections reflect a broader issue in digitally mediated higher education: the fragmentation of learning environments across multiple disconnected tools. Digital transformation in higher education is not limited to the digitisation of learning materials. It involves organisational, pedagogical, and infrastructural change across the university environment (Petchamé et al., 2023). In this broader sense, digital transformation refers to the strategic use of information, communication, and connectivity technologies to reshape institutional processes, learning experiences, and modes of participation.

LMSs have become central infrastructures in this landscape because they host materials, assignments, administrative structures, and formal communication across many programmes and institutions (Petchamé et al., 2023). At the same time, digital transformation has

encouraged the spread of additional tools for whiteboarding, teamwork, data exploration, analytics, and communication, especially in blended and hybrid learning contexts (Petchamé et al., 2023; Ahmad et al., 2021). However, the educational value of digital transformation depends not only on the presence of technologies, but on how well those technologies support the logic of the learning process.

The fragmentation gap describes a situation in which teaching and learning activities are distributed across tools that are not integrated at the level of user experience, data flow, or pedagogical logic. The problem is not that these tools are individually weak. On the contrary, many are strong within their own domains. The issue is that they are usually optimised for local functions rather than for the continuity of a complete educational workflow. A learner may read instructions in an LMS, brainstorm in a whiteboard platform, manage tasks in another application, submit results somewhere else, and receive feedback through a different channel. Even if each step is technically possible, the learning environment becomes cognitively and operationally fragmented.

This condition has direct consequences for workshop-based and project-based learning. Project-based learning depends on continuity, documentation, collaboration, and clear transitions between stages (Guo et al., 2020; Zhang and Ma, 2023). Digital collaborative learning also depends on the visibility of shared work, the coordination of contributions, and support for meaningful interaction rather than simple co-presence (Stahl, Koschmann and Suthers, 2006). When workshop activities are distributed across disconnected platforms, the burden of integration shifts from the digital environment to students, tutors, and professors.

Fragmentation reduces transparency because participants cannot easily understand what has been done, what remains to be done, or how group activity has evolved over time. It also weakens feedback. Effective feedback in workshop-based learning should be timely, contextual, and connected to the relevant stage of work. When an instructor must move between several systems to observe progress, review intermediate outputs, and communicate guidance, feedback can become delayed or detached from the activity to which it refers.

Figure 9 maps the main fragmentation problems to the corresponding platform requirements.

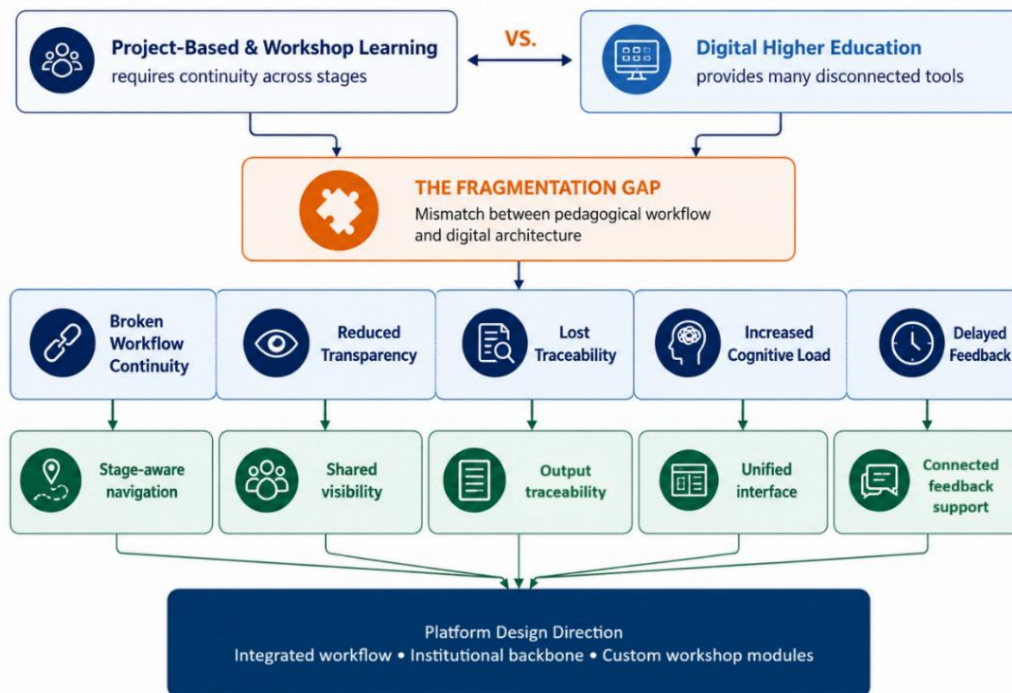


Figure 9. Mapping from fragmentation problems to platform requirements.

Source: Authors.

The operational difficulties observed in the target course show that its pedagogical design cannot function optimally without dedicated digital support. The platform must support not only content access, but also sequential data continuity, custom interaction, role-based access, traceability, accessibility, and collaboration. These needs can be grouped into four categories.

Functional requirements.

The platform must preserve sequential data persistence, so that outputs from early workshops automatically carry forward to later workshops. For example, stakeholders entered in Workshop 2.1, indicators selected and ranked in Workshops 3.1 and 3.2, and scenarios generated in Workshop 5 should remain available for scenario analysis and cost–benefit evaluation without repeated manual re-entry. The platform must also provide custom interactive modules for course-specific methods, including SRF playing cards, Problem and Objective Trees, and Q-Methodology grids. In addition, it must support role-based access control, allowing students to edit their group workspace, tutors to read and comment across supervised groups, and professors to review and evaluate globally. In the longer term, institutional integration through LTI or SSO is also necessary to connect the platform with the PoliTo LMS.

Non-functional requirements.

The platform should be web-based and accessible through a browser without local installation. It should rely on a database structure capable of isolating data by course, group, and workshop stage. It should also handle student data, grading-related information, and intellectual outputs in accordance with EU data protection requirements.

Accessibility and inclusivity requirements.

The platform must remain usable by participants with different levels of digital literacy and across different device types. This includes low-friction navigation, clear instructions, backend handling of complex calculations, embedded pedagogical scaffolding, and responsive layouts.

Collaboration and traceability requirements.

The platform must make collective decision-making visible and evaluable across the full workshop sequence. The ideal platform should support action logging, versioning, contextual feedback, and future synchronous state management. Although not all of these features need to be fully implemented in the first prototype, they define the long-term development direction.

Requirements were prioritised using the MoSCoW framework, while distinguishing between core prototype requirements and future development directions. Data continuity, custom workshop modules, role-sensitive access, output traceability, and low-friction interaction are treated as Must-Have requirements. External tool centralisation and pedagogical scaffolding are treated as Should-Have requirements. More advanced functions, such as contextual tutor feedback, detailed action logging, version history, and real-time multi-user synchronisation, are recognised as future development priorities rather than fully implemented features in the current prototype.

Table 5. Prioritised requirements for the proposed integrated platform

Requirement	Meaning for the platform	Priority
Data continuity	Outputs from earlier modules should remain available for later modules without repeated manual re-entry.	Must have
Custom workshop modules	The platform should support course-specific methods such as stakeholder analysis, problem/objectives trees, SRF, Q-methodology, start park, scenario analysis, and CBA.	Must have
Role-sensitive access	Students work within group projects, while professors and teaching assistants review outputs according to their responsibilities.	Must have
Output traceability	Intermediate outputs should be preserved so final results can be understood in relation to earlier decisions.	Must have
Low-friction and accessible interface	The platform should reduce navigation between tools and remain usable for participants with different levels of technical confidence.	Must have
External tool centralisation	Access to tools such as AHP / super decisions, promethee, and IISBE / sustainable med cities should be organised inside the platform workflow.	Should have
Pedagogical scaffolding	Instructions, labels, and guidance should be embedded inside modules to reduce cognitive load.	Should have
Contextual feedback	Tutors should be able to attach comments or guidance to specific project outputs or workflow stages.	Should have / future development
Advanced collaboration history	Detailed action logs, version history, live editing, or real-time synchronisation may be added in future versions.	Could have / future development

Source: Authors

3.5 Maturity and Implementation Readiness Analysis

The selection of an implementation pathway for the proposed platform requires more than a general comparison of software features. Because the platform is intended to support collaborative, workshop-based, project-oriented learning in higher education, each possible pathway must be assessed in relation to institutional robustness, pedagogical suitability, technical feasibility, and long-term implementability. For this reason, the thesis adopts an exploratory maturity and implementation readiness analysis.

Maturity models are commonly used in information systems, IT management, software quality, and e-learning research to assess the development, readiness, or capability of systems, organisations, or technological processes across defined dimensions and maturity levels. Their purpose is not only to describe existing tools, but to provide a structured basis for comparison, diagnosis, and improvement. Prior work on maturity-model development emphasises that such models should make their dimensions, scoring logic, and intended scope explicit, especially when they are used as exploratory decision-support instruments (de Bruin et al., 2005; Becker, Knackstedt and Pöppelbuß, 2009; Mettler, 2011). In educational technology, maturity-based approaches have also been used to assess institutional e-learning capability and continuous improvement in digitally supported learning environments (Marshall, 2010). For this reason, the maturity analysis in this thesis is used as an exploratory and transparent comparison tool to support technological pathway selection, not as a universal or statistically validated ranking of educational technologies.

The purpose of this analysis is not to identify a single existing tool that can solve the entire problem alone. Instead, it compares the main standalone technological pathways available for supporting the course workflow and evaluates their relative suitability for the specific educational and institutional context of the thesis. In order to avoid treating the final architecture as a predetermined solution, the Moodle–Django combination is not included as a scored alternative in the maturity matrix. Instead, it is treated as the design strategy that emerges from the comparative analysis of the strongest standalone pathways.

The analysis compares eight standalone alternatives: Moodle, Canvas LMS, Django custom web platform, Miro, Mural, Trello, Tableau, and a native desktop application. These alternatives are not functionally equivalent. Moodle and Canvas represent institutional LMS pathways. Django represents a custom web development pathway. Miro and Mural represent collaborative whiteboard environments. Trello represents project and task management. Tableau represents data visualisation. The native desktop application pathway represents a locally installed custom software option.

Each alternative has clear strengths and limitations. LMS platforms such as Moodle and Canvas provide institutional stability, role management, content organisation, and formal assessment support, but they are limited in relation to bespoke workshop interaction and visual co-creation. Whiteboard platforms such as Miro and Mural support ideation, visual mapping, and real-time collaboration, but they are weaker in course governance, assessment structure, and long-term traceability. Task-management tools such as Trello support

responsibility assignment and progress tracking, but their pedagogical scope is too narrow for a complete learning environment. Data-visualisation tools such as Tableau can support analysis and communication of evidence, but they do not provide the broader course workflow, collaboration, or traceability structure required by the thesis. Django offers high flexibility, workflow integration potential, scalability, and tailored development capacity, but as a standalone pathway it does not automatically provide the institutional familiarity and governance of an LMS. Native desktop applications are the least suitable because they raise problems of installation, accessibility, integration, maintenance, and distributed collaboration.

The seven dimensions used in the analysis were selected by combining three sources of justification: maturity-model literature, software-quality criteria, and the educational requirements derived from the target course. General maturity-model literature supports the use of explicit dimensions and staged scoring to compare system capability and implementation readiness (de Bruin et al., 2005; Becker, Knackstedt and Pöppelbuß, 2009; Mettler, 2011). Software-quality models such as ISO/IEC 25010 provide a reference for assessing properties such as reliability, compatibility, usability, maintainability, portability, and overall product quality, which are relevant when comparing technological pathways for an educational platform (ISO/IEC, 2023). Educational technology literature supports the inclusion of institutional e-learning capability, workflow support, accessibility, collaboration, and learning-process continuity as relevant criteria for digitally mediated higher education (Marshall, 2010; Petchamé et al., 2023; Guo et al., 2020; Stahl, Koschmann and Suthers, 2006).

The first four dimensions address technological maturity and implementation readiness. D1 Market Presence and Adoption considers whether the pathway is established and trusted in higher education or comparable institutional settings. D2 Ecosystem and Community Support evaluates the availability of documentation, developer communities, integrations, plugins, libraries, and long-term support. D3 Platform Stability and Roadmap assesses reliability, maintenance continuity, update frequency, and long-term sustainability. D4 Scalability and Integration Readiness considers whether the pathway can support larger numbers of users, structured data growth, and connection with external systems through APIs or standards such as Learning Tools Interoperability (LTI) 1.3 (1EdTech Consortium, 2019).

The remaining three dimensions address pedagogical and functional suitability for the target course. D5 Workflow Integration measures whether the pathway can support a connected, multi-stage educational process rather than only isolated tasks. This is important because project-based learning depends on continuity between stages and the preservation of intermediate outputs (Guo et al., 2020; Zhang and Ma, 2023). D6 Inclusivity and Accessibility assesses whether the pathway can support users with different levels of digital confidence, different devices, and accessible interaction patterns, in line with the broader importance of accessibility in web-based environments (W3C, 2024). D7 Collaboration Support evaluates whether the pathway can support shared work, visible group artefacts, coordination, and meaningful interaction, which are central conditions for collaborative learning environments (Stahl, Koschmann and Suthers, 2006; Ahmad et al., 2021; Vuopala et al., 2016).

Each alternative was scored on a five-level maturity scale: 1 = Initial, 2 = Emerging, 3 = Functional, 4 = Mature, and 5 = Advanced. Higher values indicate stronger readiness and suitability. The scoring logic is exploratory and comparative. Equal importance is assigned to all seven dimensions because no formal stakeholder-weighting exercise was conducted at this stage, and because the purpose of the analysis is to compare alternatives transparently rather than impose a predetermined hierarchy of priorities. The overall score for each pathway is calculated as the mean of its seven dimension scores.

The resulting maturity and implementation readiness matrix is presented in Figure 10.

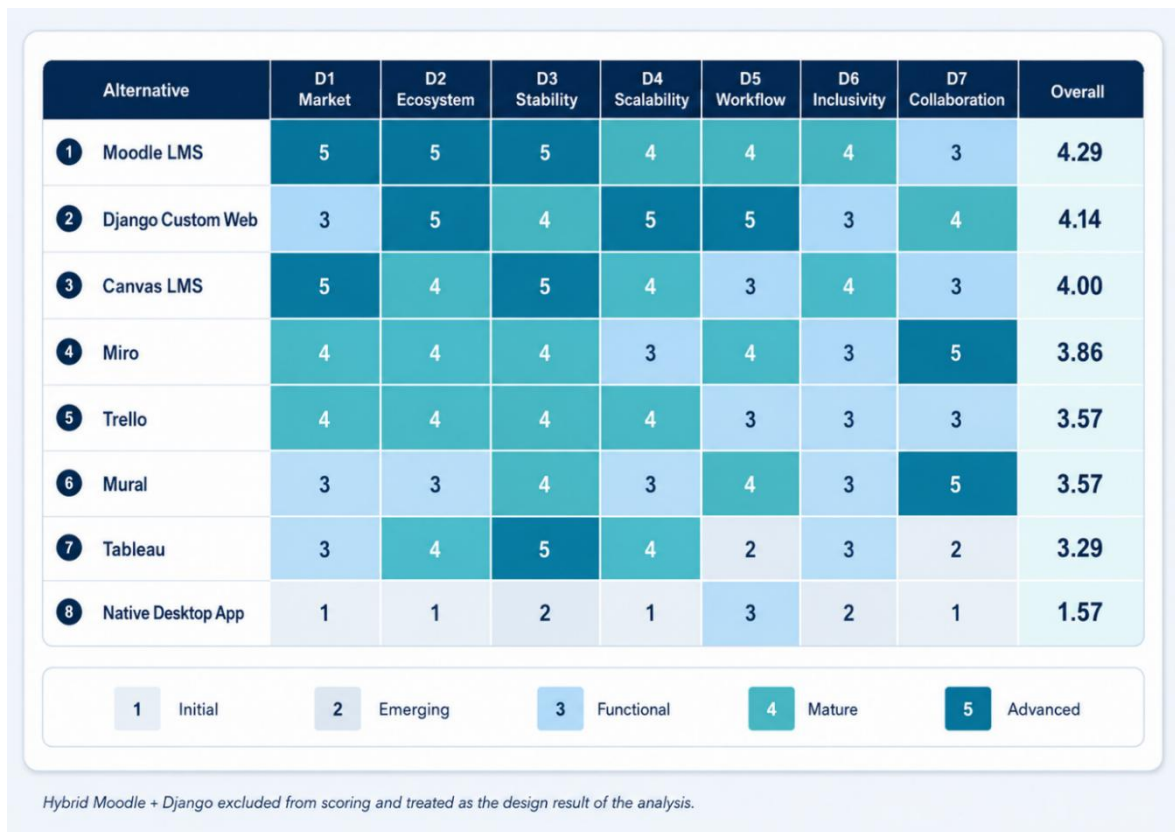


Figure 10. Maturity and implementation readiness matrix of standalone technological pathways.

Source: Authors

The scores in Figure 10 should be interpreted as an exploratory assessment of pathway suitability for the specific requirements of this thesis, rather than as an absolute ranking of software quality. Moodle receives the highest overall score because it is strong in market adoption, ecosystem maturity, platform stability, inclusivity, and institutional integration, which are central qualities for an official university learning environment. However, its lower collaboration score reflects the fact that a standard LMS is less suited to highly bespoke, real-time, workshop-specific interaction. Django receives a slightly lower overall score because, as a custom web framework, it has less direct institutional adoption as an educational platform, but it scores highly in ecosystem support, scalability, integration readiness, and workflow customisation, making it suitable for building the interactive workshop layer. Canvas follows a similar LMS logic to Moodle, with strong institutional maturity but weaker support for bespoke workflow integration. Miro and Mural score highly for collaboration and visual co-creation, but their lower scores in institutional governance, traceability, and integration readiness limit their suitability as complete course platforms. Trello and Tableau are mature within narrower functional domains, but they do not cover the full pedagogical workflow required by the thesis. The native desktop application pathway

receives the lowest score because it creates disadvantages in accessibility, installation, maintenance, integration, and collaborative use. Overall, the matrix shows that no standalone pathway fully satisfies all requirements; instead, the result supports a layered strategy in which Moodle provides institutional stability and Django provides custom workflow integration.

3.6 Result of the Maturity Analysis and Platform Strategy

The maturity analysis produces a clear ranking among the eight standalone alternatives. Moodle achieves the highest overall score at 4.29, followed closely by Django custom web platform at 4.14. Canvas LMS follows with 4.00, Miro with 3.86, Trello and Mural with 3.57, Tableau with 3.29, and the native desktop application with 1.57.

The result shows that Moodle and Django are the two strongest pathways in the analysis, but they are strong for different reasons. Moodle ranks first because it provides institutional familiarity, stable educational infrastructure, accessibility support, role management, and course-level governance. These qualities are essential in a university context, where any proposed platform must fit within existing institutional processes and not operate as an isolated external system.

Django ranks second overall, but it performs particularly strongly in the dimensions most closely related to the custom requirements of the thesis: workflow integration, scalability, ecosystem support, and tailored development. This is significant because the target course requires interactive workshop modules, persistent project-level data, visual analytical tools, and stage-to-stage continuity. These are precisely the areas where a custom web framework can provide greater flexibility than a standard LMS.

Canvas also performs strongly as an institutional LMS pathway, but it is less directly aligned with the thesis context because the target institutional ecosystem is already centred on Moodle. Miro and Mural confirm the importance of visual collaboration and real-time co-creation, but their weaker performance in institutional governance, assessment structure, and long-term traceability makes them insufficient as complete platform foundations. Trello and Tableau are useful in narrower roles, respectively for task coordination and data visualisation, but neither can support the full pedagogical workflow. The native desktop application pathway is clearly unsuitable because it performs weakly in accessibility, integration readiness, ecosystem support, and distributed collaboration.

Figure 11 summarises how the Moodle–Django platform strategy is derived from the complementary strengths identified in the maturity analysis.

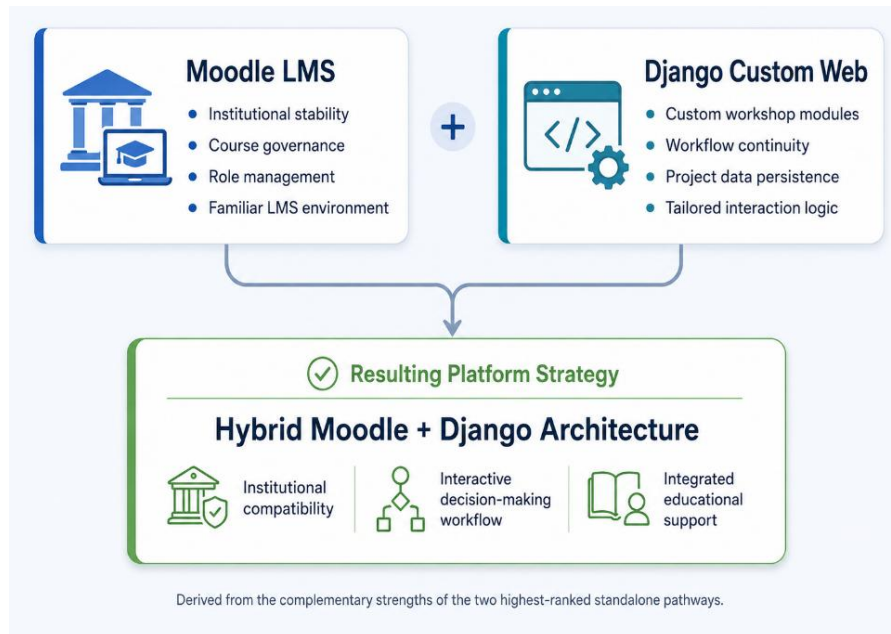


Figure 11. Derivation of the Moodle–Django platform strategy from the maturity analysis.

Source: Authors.

The most important conclusion is therefore not that one existing tool can solve the problem alone. Instead, the maturity analysis identifies two complementary foundations. Moodle provides the institutional learning-management layer required for course organisation, role management, and future integration with the university environment. Django provides the custom interactive workshop layer required to implement bespoke modules, structured data persistence, visual interaction, and workflow continuity. On this basis, the platform strategy adopted in the thesis combines Moodle and Django: Moodle remains the institutional LMS layer, while Django becomes the custom web-based workshop layer.

It is important to recognise that this strategy does not completely remove the presence of multiple digital systems. Moodle and Django remain two distinct technological environments, and in this sense the solution is not a single monolithic platform. However, the fragmentation addressed by the thesis is not simply the existence of more than one tool, but the absence of a coherent relationship between tools, data, activities, and user roles. The proposed Moodle–Django strategy therefore aims to transform uncontrolled fragmentation into a structured layered architecture. Moodle continues to provide the institutional course environment, while Django concentrates the interactive workshop workflow, data persistence, and project-based outputs in one connected space. The expected improvement

is therefore not the elimination of all tool boundaries, but the reduction of manual transfer, duplicated work, disconnected outputs, and loss of traceability across the workshop process.

This combined Moodle–Django strategy is therefore not assumed before the analysis. It is derived from the comparative maturity results. The prototype developed in Chapter 4 applies this conclusion by designing a Django-based interactive platform intended to operate in connection with Moodle’s institutional learning environment.

The maturity analysis also has methodological limitations. First, scoring necessarily involves analytical judgement, even when dimensions and maturity levels are predefined. Second, equal weighting across dimensions is appropriate for an exploratory comparison, but it may not fully reflect how different stakeholder groups would prioritise factors such as accessibility, governance, customisation, or collaboration in practice. Third, the analysis focuses on platform-level maturity and implementation readiness rather than live user behaviour. For this reason, it should be understood as a conceptual selection instrument that informs prototype development, not as a substitute for the validation phase presented in Chapter 5.

3.7 Technical Objectives of the Platform

The contextual analysis, literature review, requirements analysis, and maturity analysis together make it possible to derive the technical objectives of the proposed platform. These objectives do not arise from technical preference alone. They emerge from the mismatch between project-based pedagogical needs and fragmented digital ecosystems, and from the comparative evidence showing that Moodle and Django address complementary dimensions of the problem.

Workflow continuity. The first technical objective is workflow continuity. The literature on project-based learning and workshop-based educational processes shows that learning unfolds across connected stages and depends on the preservation of links between artefacts, interactions, and outcomes (Guo et al., 2020; Zhang and Ma, 2023). The platform should therefore support a complete process rather than separate tasks. Students should be able to move from course preparation to workshop interaction, from intermediate outputs to subsequent decisions, and from activity to review within one coherent environment.

Accessibility and inclusivity. The second technical objective is accessibility and inclusivity. Digitally mediated higher education can increase flexibility and participation, but it can also

create barriers related to interaction mode, technical conditions, and unequal support if the environment is not carefully designed (Petchamé et al., 2023). Because the platform is intended for students, professors, and teaching assistants, and because workshop-based learning depends on active participation, the environment must minimise avoidable barriers across stages and devices.

Collaboration support. The third technical objective is collaboration support. The literature on digital collaborative learning and online whiteboards shows that interaction quality depends strongly on how the platform supports shared work, visible group outputs, and meaningful coordination (Ahmad et al., 2021; Vuopala et al., 2016; Stahl, Koschmann and Suthers, 2006). In the context of this thesis, collaboration is supported primarily through persistent shared project workspaces, structured workshop outputs, and a common workflow in which group work remains available across modules. The platform therefore strengthens collaboration by making collective outputs easier to organise, review, and carry forward. Full real-time multi-user synchronisation, such as live cursors or simultaneous editing across all group members, remains a future development direction rather than a core claim of the current prototype.

Traceability of learning and evaluation logic. The fourth technical objective is traceability. Project-based learning depends not only on final deliverables, but also on intermediate work, process quality, and the relation between contributions and outcomes (Guo et al., 2020). In fragmented digital ecosystems, traces of learning become dispersed across systems, making it harder to reconstruct progression, evaluate participation, and provide meaningful process-oriented feedback. The platform must therefore preserve how activities, decisions, and outputs evolve over time.

Integration readiness, scalability, and long-term feasibility. The final technical objective combines integration readiness, scalability, and long-term feasibility. Digital transformation in higher education requires not only innovation, but also the capacity to fit within institutional systems and remain sustainable over time (Petchamé et al., 2023). Moodle contributes institutional robustness and educational continuity, while Django contributes extensibility and support for specialised modules. Their combination supports both immediate prototype implementation and future institutional usefulness.

Table 6. Technical objectives derived from literature, requirements analysis, and maturity analysis.

Technical Objective	Definition	Literature Sources	Supporting Maturity Dimensions
Workflow Continuity	The platform must support a complete, stage-aware process allowing students to move seamlessly from preparation through workshop activity to evaluation within one coherent environment.	Guo et al. (2020); Zhang and Ma (2023); Petchamé et al. (2023)	D5 Workflow Integration
Accessibility and Inclusivity	The platform must minimise barriers across devices, interaction modes, and participation contexts, ensuring equitable access for students, professors, and teaching assistants.	Petchamé et al. (2023); Vuopala et al. (2021)	D6 Inclusivity and Accessibility
Collaboration Support	The platform must support collaboration through persistent shared group workspaces, structured workshop outputs, and visible group artefacts, while advanced real-time synchronisation remains a future development direction.	Vuopala et al. (2021); Stahl, Koschmann and Suthers (2006)	D7 Collaboration Support
Traceability of Learning and Evaluation	The platform must preserve traces of activities, artefacts, decisions, and contributions across stages to support process-oriented evaluation and meaningful feedback.	Guo et al. (2020); Zhang and Ma (2023)	D5 Workflow Integration; D7 Collaboration Support
Integration Readiness, Scalability, and Long-Term Feasibility	The platform must be technically integrable into institutional environments, scalable beyond the prototype, and sustainable in terms of long-term maintenance and institutional adoption.	Petchamé et al. (2023); Wang (2021)	D1 Market Presence; D2 Ecosystem; D3 Stability; D4 Scalability

Source: Authors

3.8 Outcome of the Theoretical and Conceptual Phase

The theoretical and conceptual phase produces a clear operational basis for the prototype developed in Chapter 4. The analysis begins from the specific educational context of the target course at Politecnico di Torino and shows that the course depends on a continuous

sequence of workshop-based and decision-support activities. Each stage generates outputs that must remain available, traceable, and usable in later stages. The examination of students, professors, and teaching assistants further shows that all user groups are affected by fragmentation, although in different ways.

The literature review confirms that this problem is not only a local organisational issue, but part of a broader challenge in digitally mediated higher education. Project-based and workshop-based learning require continuity, collaboration, documentation, transparency, and feedback across time. When learning activities are distributed across disconnected tools, the burden of integration is transferred to students and instructors. This weakens workflow continuity, reduces traceability, increases cognitive load, and makes feedback less contextual.

The requirements analysis translates these educational and operational problems into platform requirements. The most important requirements are data continuity, custom workshop modules, role-sensitive access, output traceability, low-friction interaction, pedagogical scaffolding, and future support for more advanced collaboration features. The analysis also recognises that not all desirable features need to be fully implemented in the proof-of-concept prototype. Detailed action logging, version history, contextual tutor feedback, and real-time synchronous collaboration are treated as future development directions rather than core claims of the current implementation.

The maturity and implementation readiness analysis provides the technical basis for the platform strategy. Moodle achieves the highest overall score among the standalone alternatives, while Django achieves the second-highest score and performs particularly strongly in workflow integration and tailored development. The result of the analysis is therefore the decision to combine the two most suitable and complementary pathways: Moodle as the institutional learning-management layer and Django as the custom interactive workshop layer.

The outcome of this phase is an operational design brief for the prototype. The platform developed in Chapter 4 should not be limited to digitising isolated workshop activities. Instead, it should organise the complete decision-making workflow, including early project framing, stakeholder analysis, problem and objective structuring, indicator selection and ranking, indicator analysis, SWOT analysis, scenario generation through Q-Methodology and Start Park, scenario analysis, structured access to external decision-support tools, cost–benefit evaluation, and final project review.

This result directly informs the prototype development phase. Chapter 4 therefore does not begin from an abstract technical preference, but from a structured conceptual outcome: the need for a web-based, institutionally compatible, Django-driven workshop platform designed around workflow continuity, accessibility and inclusivity, collaboration support, output traceability, external tool centralisation, scenario evaluation, cost–benefit analysis, and long-term feasibility.

Chapter 4. Prototype Development

This chapter documents the development of the prototype platform designed in response to the theoretical and conceptual foundation established in Chapter 3. Within the Design Thinking framework adopted in this thesis, Chapter 4 corresponds to the Prototype stage. Its purpose is to show how the educational requirements, technical objectives, and Moodle–Django platform strategy derived in the previous chapter were translated into a functional web-based prototype.

The prototype was developed as a Django-based custom interactive workshop layer intended to complement, rather than replace, the institutional LMS environment. In this strategy, Moodle is understood as the institutional LMS layer responsible for course governance, materials, roles, deadlines, and formal learning-management functions, while Django provides the custom interactive workshop layer needed to support the specific decision-making workflow of the course. The implementation presented in this chapter focuses primarily on the Django layer. Future institutional integration with Moodle through standards such as LTI 1.3 remains part of the long-term deployment direction.

The platform is organised around project-based workspaces. Each project functions as the central container for the group’s workshop outputs, allowing data produced in one module to remain available and reusable in later modules. This structure directly responds to the fragmentation problem identified in Chapter 3 by reducing repeated manual transfer between external tools and by supporting data continuity, output traceability, and a coherent workshop progression.

The implemented prototype translates the course methodology into a sequence of connected workshop modules and support modules. These include project creation and dashboard access, Idea Canvas, Stakeholder Analysis, Problem Tree, Objective Tree, Indicator Selection, Indicator Ranking through the Simos-Roy-Figueira method, Indicator Analysis, SWOT Analysis, Scenario Building through Q-Methodology and Start Park, Scenario Analysis, External Decision-Support Tools, Cost–Benefit Analysis / CBA, and Project Review.

4.1 From Educational Requirements to Platform Design

The prototype development process begins from the requirements and technical objectives derived in Chapter 3. The platform was not designed as a generic learning website, but as a

response to a specific educational problem: the fragmentation of a multi-stage, workshop-based decision-making workflow across disconnected tools. For this reason, the prototype design is guided by the core requirements derived in Chapter 3: workflow continuity, data continuity, custom workshop modules, structured collaboration, accessibility and low-friction interaction, output traceability, role-sensitive access, and integration readiness.

Workflow continuity is addressed by organising the platform around a single project workspace. Instead of treating each workshop as an isolated page or independent file, the prototype links workshop outputs to the same project record. This allows data produced in earlier modules, such as stakeholders, indicators, scenario definitions, and performance values, to remain available for later analytical and evaluative stages.

Structured collaboration is supported through interactive workshop interfaces that require students to work actively with the course methods. The prototype includes visual and form-based components such as stakeholder matrices, problem and objective trees, SRF ranking boards, Q-sort interfaces, scenario dashboards, SWOT boards, and cost–benefit analysis forms. These components are designed to support learning-by-doing rather than passive content consumption.

Accessibility and low-friction interaction are addressed through a consistent web-based interface, repeated navigation patterns, clear instructions, structured cards, visual feedback, and export options. The aim is to make the platform usable by students with different levels of technical confidence while reducing the need to move between unrelated tools.

Output traceability is addressed by storing the main outputs of each module inside the project workspace. The prototype preserves intermediate artefacts, such as selected indicators, stakeholder classifications, ranking results, scenario data, and final review outputs, so that the development of the project can be followed across the workflow. Detailed action logging, version history, and advanced tutor annotation are treated as future development directions rather than core claims of the current prototype.

Role-sensitive access is addressed at a foundational level through authenticated project access and the distinction between student workspaces and review-oriented use by instructors or teaching assistants. A more complete role model, including dedicated professor dashboards, teaching-assistant monitoring views, stage locking, and contextual feedback tools, remains a future development direction.

Figure 12 summarises how the main platform requirements are translated into prototype design responses and implemented features.

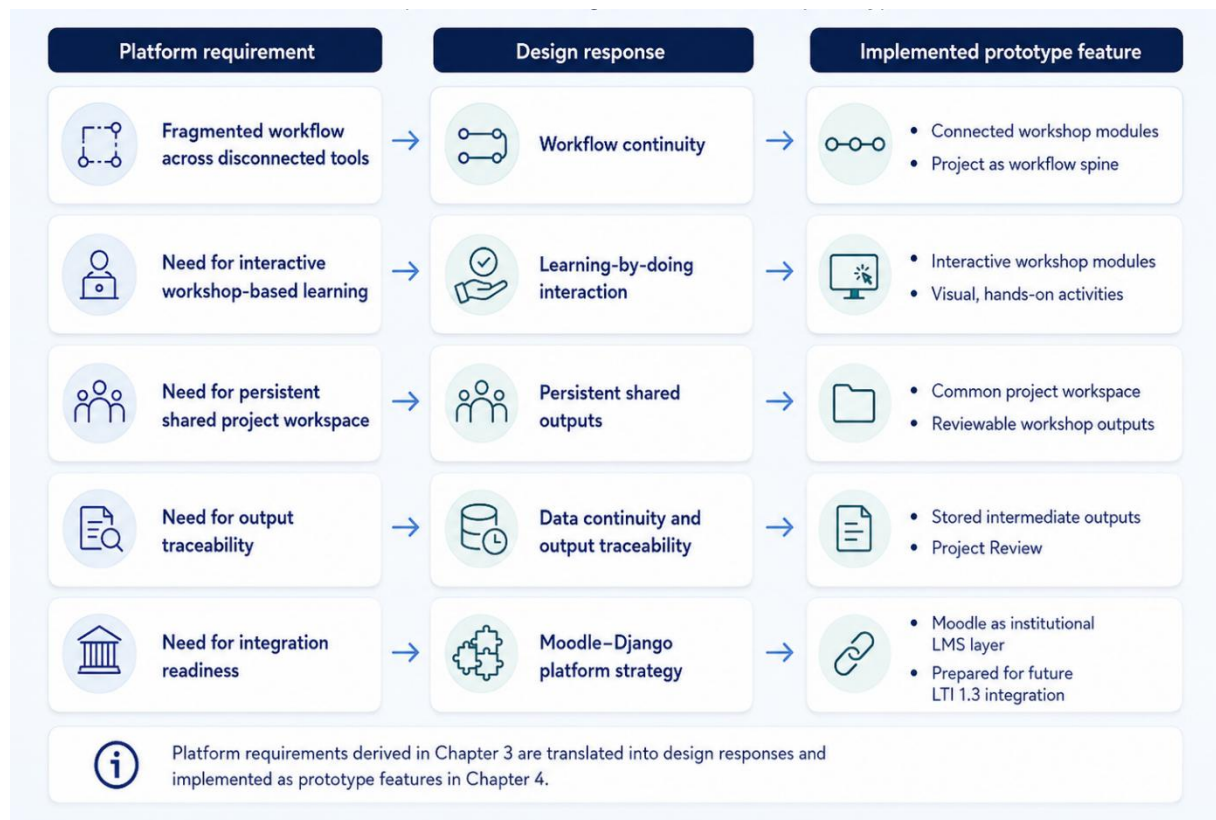


Figure 12. Requirement-to-design translation in the prototype

Source: Authors.

Table 7. Translation of platform requirements into prototype design responses

Requirement	Prototype design response	Implemented scope in the current prototype
Workflow continuity	The platform is organised around project-based workspaces, where outputs from each module remain connected to the same project record.	Implemented through the Project model and module-specific data persistence across the workshop sequence.
Data continuity	Outputs such as stakeholders, selected indicators, ranking results, scenario data, and performance values are stored for reuse in later modules.	Implemented for the main workflow modules; further automation can be expanded in future versions.

Custom workshop modules	Course-specific methods are translated into dedicated Django interfaces rather than being handled through external generic tools.	Implemented across Idea Canvas, Stakeholder Analysis, Problem Tree, Objective Tree, Indicator Selection, SRF Ranking, Indicator Analysis, SWOT Analysis, Scenario Building, Scenario Analysis, CBA, and Project Review.
Structured collaboration	Group work is supported through shared project workspaces and interactive analytical interfaces.	Implemented as persistent shared outputs; real-time synchronous multi-user editing remains future work.
Accessibility and low-friction interaction	The interface uses repeated navigation patterns, structured cards, embedded instructions, and clear action buttons.	Implemented through a consistent web-based academic dashboard interface.
Output traceability	Intermediate outputs are preserved so that final results can be interpreted in relation to earlier project decisions.	Implemented through stored module outputs and the Project Review section; detailed action logging and version history remain future work.
Role-sensitive access	The platform distinguishes between project workspaces and review-oriented use by instructors or teaching assistants.	Implemented at a foundational level; advanced professor and teaching-assistant dashboards remain future development.
Integration readiness	The architecture is designed to complement Moodle as the institutional LMS layer and Django as the custom interactive workshop layer.	Conceptually and architecturally supported; full Moodle integration through LTI 1.3 remains a future institutional deployment direction.

Source: Authors

4.2 System Architecture

The prototype follows a Moodle–Django platform strategy composed of two complementary layers: the institutional LMS layer and the custom interactive workshop layer. This architecture directly reflects the outcome of the maturity and implementation readiness analysis presented in Chapter 3, where Moodle and Django emerged as the two strongest and most complementary technological pathways. Moodle provides the institutional structure required for course governance, access to materials, role management, deadlines, and formal learning-management functions, while Django provides the flexibility needed to implement bespoke workshop modules, visual analytical tools, project-level data persistence, and stage-to-stage workflow continuity.

In the current prototype, the development work focuses primarily on the Django-based custom interactive workshop layer. Moodle is treated as the institutional environment with which the platform is designed to integrate in a future deployment scenario. The architecture therefore remains compatible with future institutional connection through standards such as Learning Tools Interoperability (LTI) 1.3 (1EdTech Consortium, 2019), but full production-level Moodle integration is treated as a future development direction rather than as a core claim of the current proof-of-concept prototype. This distinction is important because the prototype was developed and tested mainly in a local development environment, while institutional deployment would require additional configuration, security validation, user-role mapping, and production infrastructure.

Within the Django layer, the application is organised around projects. Each project represents a group's decision-making case and functions as the central workspace for the complete workshop sequence. When a user creates or accesses a project, the system provides a project dashboard through which the connected workshop modules and support modules can be entered, saved, revised, and reviewed. Each module reads from and writes to project-related data, either through relational database models or through structured JSON fields within the Project model. This project-centred structure allows the platform to preserve data continuity and output traceability across the full workflow.

The architecture can be understood through three internal Django layers. The presentation layer is built with Django templates, HTML, CSS, Bootstrap, and JavaScript. It provides the user interface, workshop pages, visual dashboards, forms, modals, instructions, and export controls. The application layer is implemented through Django views, forms, URL routing, utility functions, validation logic, and data-processing functions. It is responsible for processing user actions, saving module data, performing calculations, preparing template context, and generating exports. The data layer is managed through Django's Object-Relational Mapper and uses SQLite in the prototype environment. This development configuration supports rapid iteration and functional testing, while remaining compatible with future migration to a production database such as PostgreSQL.

The resulting architecture is therefore not a replacement for the institutional LMS. It is a complementary structure in which Moodle maintains the institutional learning-management role, while Django provides the specialised interactive environment required by the workshop-based decision-making workflow. This separation of responsibilities allows the

prototype to address the fragmentation problem without proposing a disconnected external platform.

Figures 13 and 14 present the overall Moodle–Django architecture and the internal Django platform architecture.

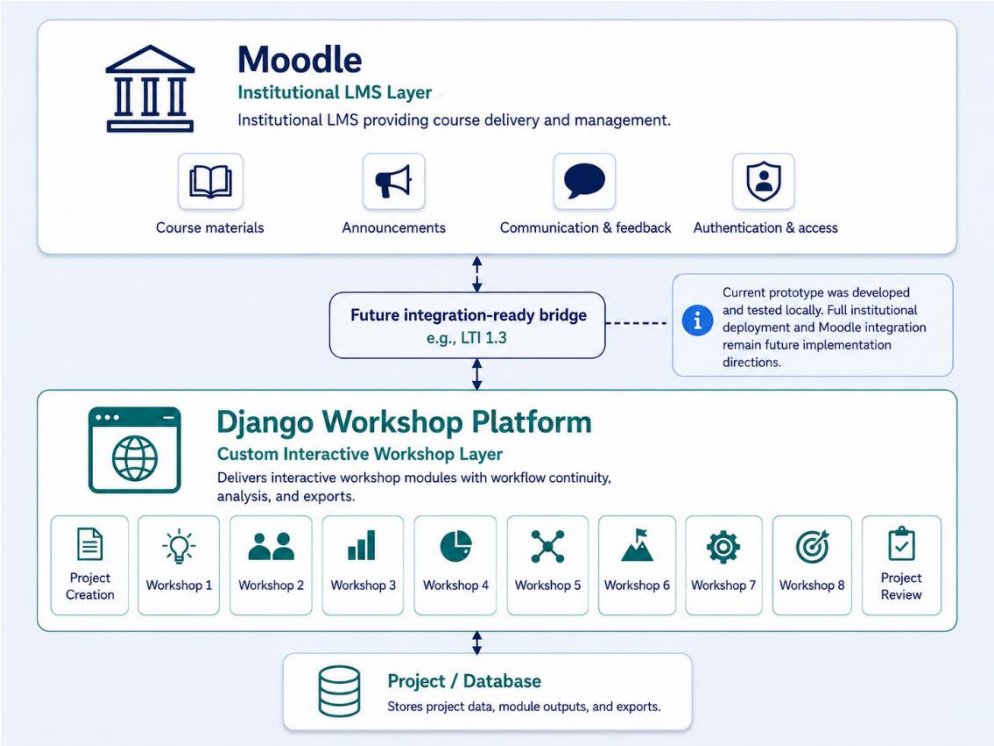


Figure 13. Hybrid Moodle–Django architecture.

Source: Authors.

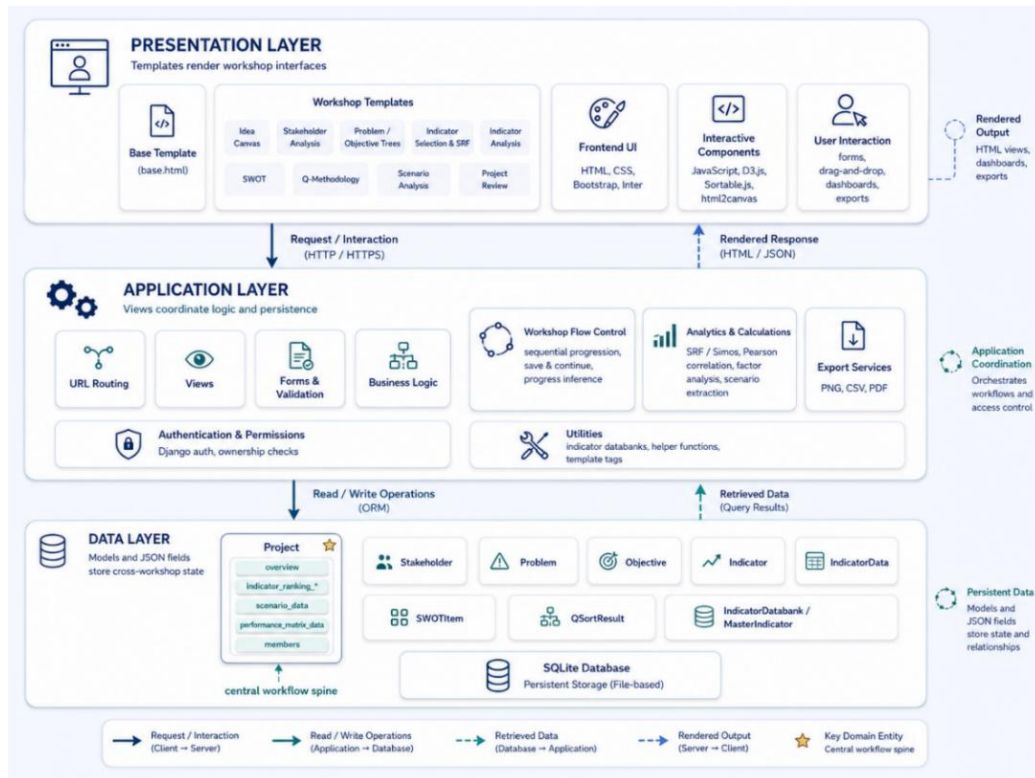


Figure 14. Django platform architecture.

Source: Authors.

4.3 Technology Stack and Development Environment

The prototype was developed using Python and Django. Django was selected because it provides a stable and well-documented web framework with built-in support for URL routing, template rendering, relational database modelling, user authentication, form handling, and administrative functionality. These capabilities make it suitable for the development of a structured educational prototype requiring multiple connected workshop modules, project-level data persistence, and authenticated user access.

The backend was implemented with Django 5.2.6 and Python (Django Software Foundation, 2026). The prototype uses SQLite as its database during development and local testing. SQLite was selected for the prototype phase because it is lightweight, simple to configure, and adequate for iterative local development and functional testing. In a future production deployment, migration to PostgreSQL would be more appropriate to support concurrent database access, stronger scalability, institutional backup procedures, and long-term data-management requirements.

The frontend uses Django templates, HTML5, CSS3, Bootstrap 5.3.3, and vanilla JavaScript. Bootstrap provides the responsive layout structures, button components, modal dialogs, card grids, and badge systems that support the platform interface. Custom CSS is applied to produce the academic dashboard aesthetic used across the prototype, including structured page headers, rounded cards, soft shadows, status indicators, and consistent section layouts. The interface is designed to remain clear, readable, and accessible for users with different levels of technical confidence.

Several JavaScript libraries are used to support interactive workshop functions. Sortable.js supports drag-and-drop interactions, especially in ranking and card-based activities. D3.js, together with browser-based SVG and canvas logic, supports visual and analytical components such as trees, matrices, and charts. html2canvas enables client-side image export, allowing visual workshop outputs to be captured and downloaded as PNG files directly from the browser.

The prototype also includes server-side and data-processing tools. ReportLab is used for structured PDF generation in reporting functions. CSV exports are implemented through Django HTTP responses. Data-processing operations, including indicator databank import, numerical calculations, and tabular data handling, are supported by NumPy, pandas, and openpyxl. These libraries are important because the platform must handle both qualitative workshop artefacts and quantitative decision-support data.

The development environment also includes deployment preparation files, including a requirements specification and containerisation-related files. These elements support environment reproducibility and future deployment preparation. However, the current prototype remains a proof-of-concept implementation developed and tested in a local environment. Full institutional deployment, production database migration, and Moodle integration through LTI 1.3 remain future development directions.

Table 8. Technology Stack of the Prototype

Component	Technologies	Purpose
Backend framework	Python, Django 5.2.6	Application logic, routing, authentication, models, views

Development database	SQLite	Local data persistence during prototype development
Production database (planned)	PostgreSQL	Concurrent access, scalability, institutional deployment
Frontend	Django templates, HTML, CSS, JavaScript	Workshop pages and user interaction
UI framework	Bootstrap 5.3.3	Responsive layout, buttons, cards, modals
Interaction libraries	Sortable.js, D3.js	Drag-and-drop ranking and data visualisation
Export tools	html2canvas, ReportLab, CSV responses	PNG, PDF, and CSV outputs
Data-processing tools	NumPy, pandas, openpyxl	Indicator import, calculations, tabular processing
Deployment preparation	Dockerfile, docker-compose.yml, requirements.txt	Future deployment and environment reproducibility
Integration readiness	LTI 1.3 / PyLTI1p3, future direction	Planned institutional Moodle connection and role transmission in a future deployment scenario

Source: Authors

4.4 Django Project Structure

The Django implementation is organised around one root project and one primary application. The root project, `workshop_service`, contains the global configuration, environment settings, installed applications, middleware configuration, database settings, and root URL routing. The main application, `workshops`, contains the models, views, URL patterns, forms, utility modules, custom template tags, and HTML templates that collectively implement the custom interactive workshop layer.

The central backend files are `models.py`, `views.py`, `urls.py`, and `forms.py`. The `models.py` file defines the main data structures used across the platform, including projects, stakeholders, problem-tree nodes, objective-tree nodes, indicators, indicator-analysis records, SWOT items, Q-sort results, scenario-related data, and cost-benefit analysis data. The `views.py` file contains the main application logic, including workshop page rendering, data saving,

calculation handling, export generation, dashboard navigation, and redirects between modules. The `urls.py` file maps user-facing routes to the corresponding view functions, while `forms.py` supports project creation and structured input validation.

The template structure is organised under `templates/workshops/`. Each workshop module or support module is represented through a dedicated HTML template or template group. These templates include pages for project creation and project overview, stakeholder analysis, problem tree, objective tree, indicator selection, SRF indicator ranking, indicator analysis, SWOT analysis, scenario building through Q-Methodology and Start Park, scenario analysis, external decision-support tools, Cost–Benefit Analysis / CBA, and Project Review. This template organisation reflects the educational structure of the platform while allowing each module to contain interaction logic specific to its method.

A shared base template provides the global layout structure of the platform. It defines the common page frame, navigation behaviour, user authentication status, login and logout controls, shared visual styling, and reusable interface patterns. This ensures that users encounter a consistent environment across the prototype, even though the workshop modules differ substantially in their methodological content and interaction logic.

The platform also includes dedicated utility modules. For example, the SRF calculation logic is separated into a utility module so that the indicator-ranking method can be processed consistently and reused across the relevant views. Indicator databank import and processing are also handled through dedicated utilities, supporting the transformation of spreadsheet-based indicator sources into structured platform data. Custom template tags and filters are stored in the `templatetags` directory and support operations such as dictionary access, string formatting, and the rendering of complex analytical dashboard data inside Django templates.

One technical characteristic of the current prototype is that part of the CSS styling and JavaScript interaction logic remains embedded directly within individual templates. This approach supported rapid iterative development because each workshop module required different interface behaviours, visual structures, and validation patterns. However, it also introduces a maintainability limitation: as the platform grows, repeated style definitions and script blocks should be progressively refactored into external static files and reusable components. This issue is treated as a future development direction in Section 4.10.

Figure 15 shows the Django project structure and the organisation of the main application files.

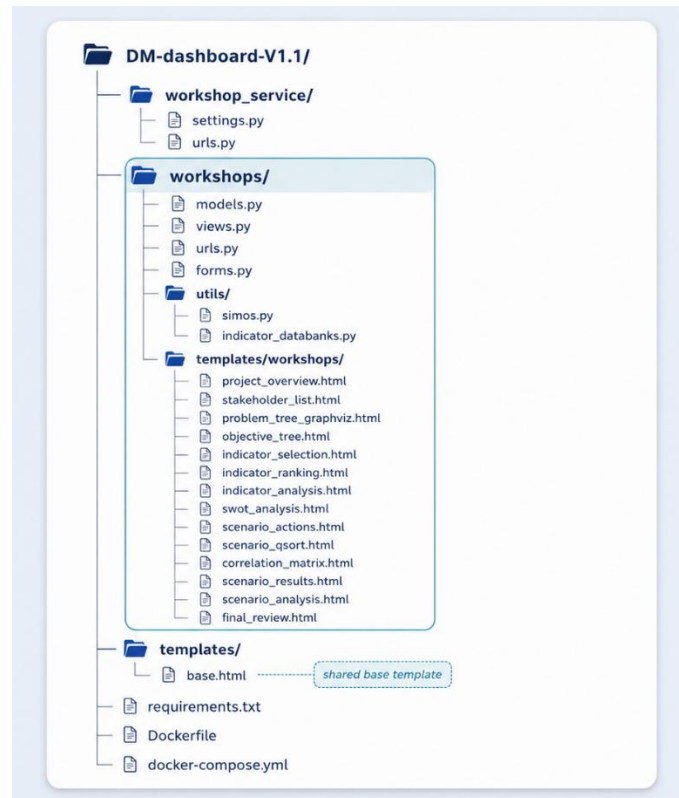


Figure 15. Django project structure and application logic.

Source: Authors.

4.5 Data Model and Persistence Strategy

The data model is organised around the central Project model. Each project instance represents a complete group workspace associated with a specific case study. It stores the project title, subtitle, ownership information, selected indicator databank, and structured workflow data. The Project model therefore functions as the persistent anchor of the platform. Rather than treating each workshop module as an isolated page or independent file, the platform links module outputs to the same project record, preserving data continuity across the full analytical sequence.

The platform uses a mixed persistence strategy based on relational models and structured project-level data fields. Relational models are used where the data structure is stable, repeatable, and benefits from database integrity and querying. For example, stakeholders are stored as project-related records with attributes such as name, power, interest, level, typology, and resources. Problem-tree and objective-tree data are represented through node-based structures that preserve hierarchical relationships. Indicators are stored as project-related records containing category, criterion, unit, acceptance status, ordering, ranking

information, and calculated weights. SWOT entries are stored according to their quadrant category, while indicator analysis records preserve supporting images, calculated values, and bibliographic references.

Structured JSON fields are used where the data is more dynamic, nested, or methodologically complex. This is particularly useful for modules such as Idea Canvas, scenario building, Q-sorting, scenario analysis, and performance matrices, where outputs may include flexible arrays, nested objects, labels, scores, scenario definitions, and analytical metadata. JSON storage allows the prototype to support complex interaction patterns without requiring a new database table for every variation of the method.

This combination of relational and JSON-based persistence supports both stability and flexibility. Relational models provide consistency for entities that are repeatedly used across the workflow, such as stakeholders, indicators, SWOT items, and indicator analysis records. JSON fields provide flexibility for complex educational artefacts, such as canvas sections, Q-Methodology data, Start Park scenario structures, scenario-analysis values, CBA assumptions, and review outputs. In this way, the data model reflects the pedagogical logic of the course: some elements must be structured and reusable, while others must remain adaptable to the evolving nature of project-based work.

Data persistence is implemented through a combination of standard Django form submissions and asynchronous JavaScript requests. Conventional actions, such as project creation, structured form saving, and file uploads, are handled through standard POST submissions. More interactive modules, such as Idea Canvas editing, stakeholder updates, ranking activities, Q-sort saving, scenario dashboards, and visual analytical components, use JavaScript-based requests and JSON endpoints. This hybrid approach allows the platform to support both form-based workflows and dynamic interactive activities within the same Django application.

The current prototype infers module progress mainly from the presence of saved data rather than from a separate central progress-tracking model. For example, the existence of stakeholder records indicates progress in Stakeholder Analysis, accepted and ranked indicators indicate progress in Indicator Selection and SRF Ranking, scenario data indicates progress in Scenario Building, and saved CBA records indicate progress in Cost–Benefit Analysis. This approach is sufficient for the proof-of-concept prototype, but a future version could introduce a dedicated progress model to support more detailed professor and teaching-assistant dashboards, stage locking, completion analytics, and version history.

The data model therefore supports the core objectives identified in Chapter 3. It improves workflow continuity by linking all outputs to one project workspace. It supports data continuity by making outputs from earlier modules available to later modules. It improves output traceability by preserving intermediate artefacts instead of reducing the course process to a final submitted file. Finally, it prepares the platform for future institutional deployment by keeping the project structure compatible with more advanced role management, reporting, and integration mechanisms.

Figure 16 summarises the simplified data model of the platform and the central role of the Project model.

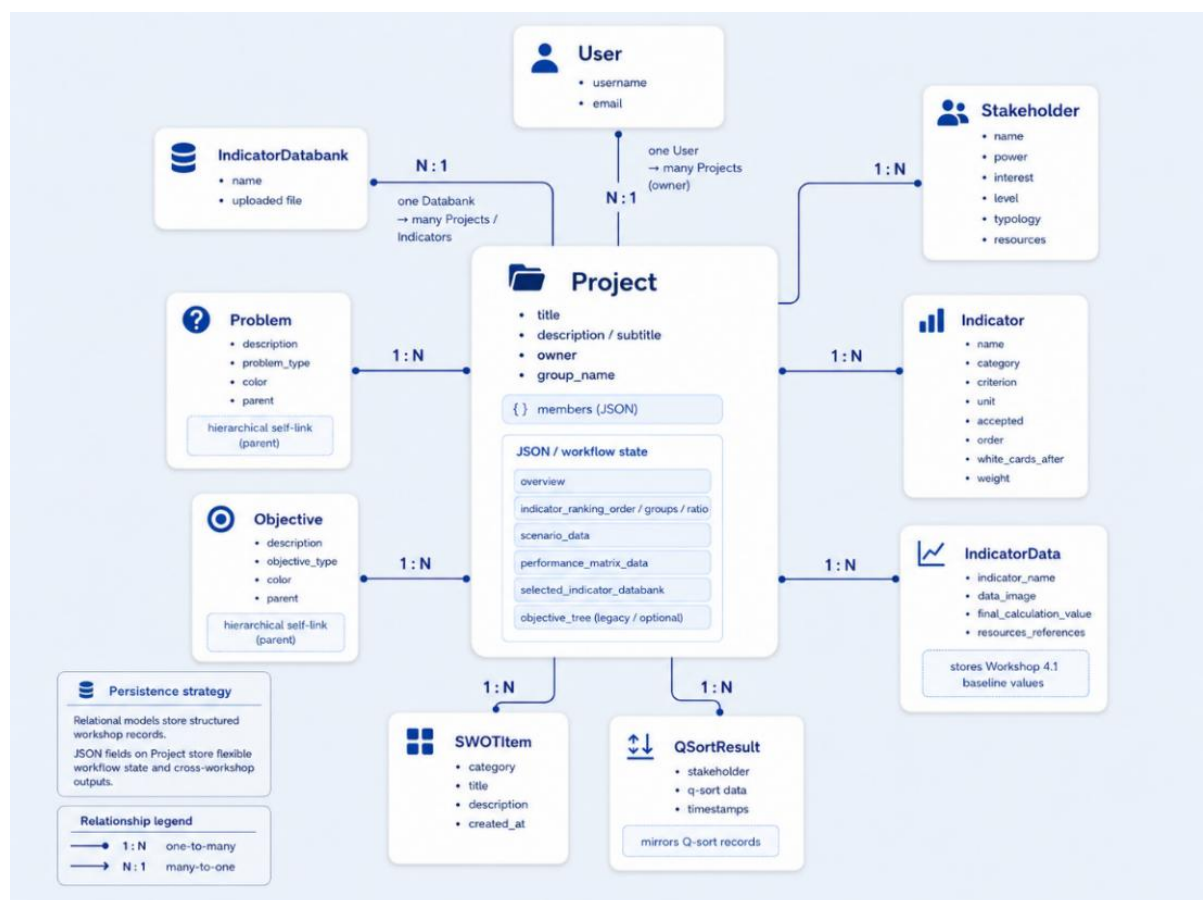


Figure 16. Simplified data model of the platform.

Source: Authors.

Table 9. Main Data Models and Their Role in the Platform

Model / field	Main role	Workshop connection
Project	Central container for project title, subtitle, owner, selected databank, and structured workflow data	Full platform
Project overview	Stores idea canvas sections and sdgs notes	Workshop 1
Stakeholder	Stores stakeholder name, power, interest, level, typology, and resources	Workshop 2.1
Problem	Stores problem-tree nodes and hierarchical relationships	Workshop 2.2
Objective	Stores objective-tree nodes and relationships	Workshop 2.3
Indicator	Stores selected indicators, accepted status, order, white cards, and calculated weights	Workshops 3.1–3.2
Indicator data	Stores uploaded chart/image, final calculated value, and references	Workshop 4.1
Swot item	Stores swot entries by quadrant category	Workshop 4.2
Project scenario data	Stores scenario-building data, including q-methodology actions, q-sorts, scenario labels, and related analytical outputs	Workshop 5
Start park scenario data	Stores spatial scenario-building elements such as cards, placements, and scenario summaries	Workshop 5
Performance matrix	Stores scenario-analysis values, scenario labels, and comparison metadata	Workshop 6
External decision-support tool	Stores or displays structured references to external tools selected by the professor	Workshop 7
Cba data	Stores selected scenario, cost items, benefit items, assumptions, npv, cost–benefit ratio, and sensitivity-analysis outputs	Workshop 8
Project review data	Aggregates saved outputs from previous modules into a final synthesis and review structure	Project review

Source: Authors

4.6 User Workflow and Project Dashboard

The user workflow is organised around the project workspace. After accessing the platform, users enter the dashboard environment where they can either create a new project or continue working on an existing project. Each project represents one group case study and acts as the central container for all workshop outputs. This structure ensures that the platform is not experienced as a collection of disconnected tools, but as a coherent sequence of linked activities.

When a project is created, the user provides basic project information such as the project title and subtitle. These elements help identify the case study and appear across the dashboard and workshop pages. The project information can be edited later, allowing the group to refine the framing of the project as its understanding develops during the workshop sequence. This is important because project-based learning is iterative: the initial framing of a case may evolve as students analyse stakeholders, define problems, select indicators, and compare scenarios.

The project dashboard functions as the main navigational hub of the prototype. It displays the project identity, the sequence of workshop modules, access buttons, and progress-related indicators. From this page, users can enter the different modules of the platform, including Idea Canvas, Stakeholder Analysis, Problem Tree, Objective Tree, Indicator Selection, SRF Ranking, Indicator Analysis, SWOT Analysis, Scenario Building, Scenario Analysis, External Decision-Support Tools, Cost–Benefit Analysis / CBA, and Project Review. The dashboard therefore translates the educational workflow into a visible digital pathway.

The dashboard is designed to support workflow continuity. Instead of asking students to remember where each output is stored or which external file must be opened next, the platform presents the workshop modules as parts of one connected process. This reduces context switching and helps users understand how earlier activities contribute to later analytical stages. For example, stakeholders, indicators, scenario data, and performance values are not treated as isolated outputs, but as project-level data that can be reused across the workflow.

The dashboard also supports reviewability. As students move through the platform, saved outputs remain associated with the project workspace and can be revisited. This allows

students to revise previous stages when necessary and allows instructors or teaching assistants to understand the development of the project from early framing to final synthesis. In the current prototype, this reviewability is primarily supported through persistent saved outputs and the Project Review section. More advanced instructor dashboards, completion analytics, version history, and contextual feedback tools remain future development directions.

The workflow follows a stage-aware logic but does not strictly prevent users from moving between modules. This decision reflects the iterative character of project-based learning. Although the modules are presented in a recommended sequence, users may need to return to earlier stages to correct, refine, or extend their work. For instance, a group may update the stakeholder analysis after identifying a new actor during SWOT Analysis, or revise indicator choices after reviewing scenario-analysis requirements. The prototype therefore balances structured progression with flexible navigation.

Overall, the user workflow and dashboard design support the main technical objectives derived in Chapter 3. They improve workflow continuity by centralising access to all modules, support accessibility through a consistent navigation structure, encourage structured collaboration through shared project workspaces, and contribute to output traceability by preserving intermediate artefacts within the same project environment.

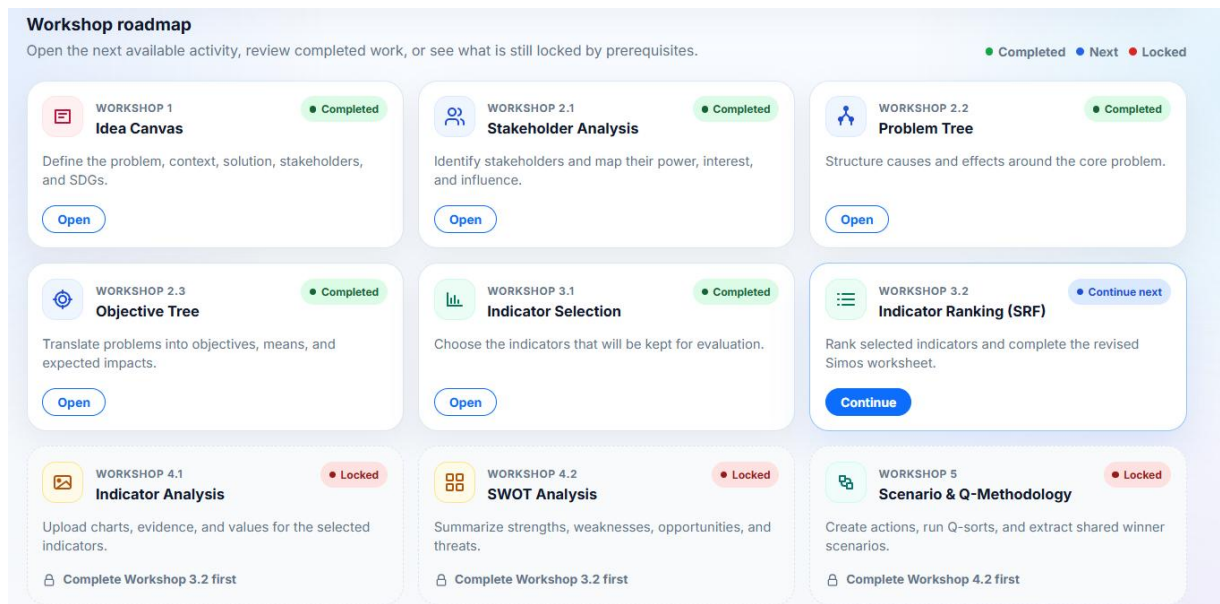
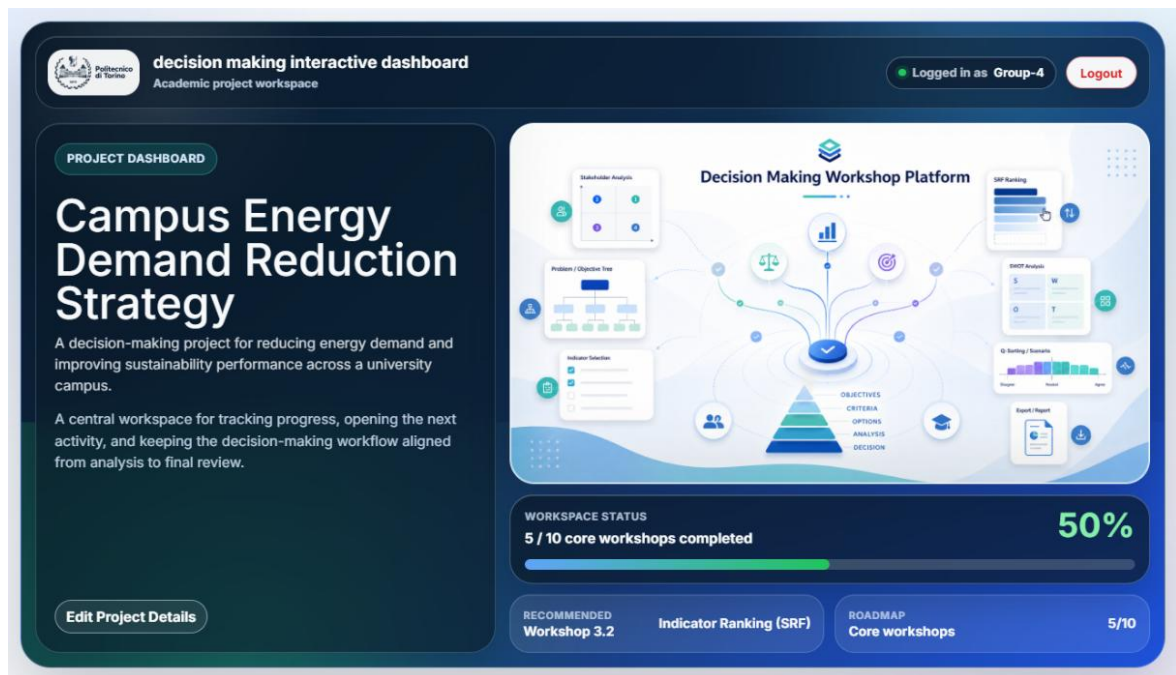


Figure 17. Main project dashboard and workshop cards.

Source: Authors.

4.7 Workshop Module Implementation

The prototype translates the course workflow into a sequence of connected workshop modules and support modules. Each module corresponds to a specific methodological step in the decision-making process and is designed to produce outputs that can be stored, reviewed, and reused within the same project workspace. The implementation therefore does

not treat the workshops as independent pages, but as parts of a cumulative project-based workflow.

The modules are accessed through the project dashboard and are connected through the central project data structure described in Section 4.5. This organisation supports workflow continuity by allowing outputs from earlier modules to inform later stages. It also supports output traceability because the intermediate artefacts of the project remain available within the platform rather than being dispersed across external tools.

Workshop 1 - Idea Canvas

Workshop 1 supports the initial framing of the project. The module allows students to describe the project context, identify possible intervention ideas, define initial actors, and structure early reflections around the case study. It functions as the starting point of the group's project workspace and provides the first structured layer of information for the following modules.

The interface is organised into structured input areas rather than a free-form document. This design helps students move from open brainstorming toward a more readable and reviewable project description. The saved content becomes part of the project overview and remains available for later review. In this way, the Idea Canvas replaces an isolated brainstorming board with a persistent project-framing module.

An important function of this module is that some early data introduced during project framing can be stored as structured project data and connected to later modules. For example, actors or stakeholders identified during the Idea Canvas stage can be preserved in the project database and reused or refined during Stakeholder Analysis. This reduces repeated manual re-entry, supports data continuity, and reinforces the platform's role as a connected workflow rather than a sequence of disconnected pages.

By storing both narrative project framing and structured early inputs, Workshop 1 establishes the first layer of output traceability. Later stages can therefore be interpreted in relation to the initial project definition, intervention ideas, and actor identification developed at the beginning of the workflow.

Politecnico di Torino
Academic project workspace

decision making interactive dashboard

Logged in as: Group-4 [Logout](#)

WORKSHOP 1

Idea Canvas

Capture the core logic of the project by defining the problem, context, solution, activities, supporting conditions, and expected contribution to the Sustainable Development Goals.

[Back to Project Home](#) [Download as One Image](#) [Download by Sections](#) [Save & Continue](#)

2. Solution Design

Clarify what the project proposes and the main actions that will make it happen.

B1 Solution

- Develop an integrated decision-making framework for prioritising campus energy demand reduction strategies.
- Combine stakeholder analysis, indicator weighting, baseline assessment, SWOT analysis, Q-methodology, and scenario comparison.
- Compare alternative energy transition scenarios using weighted sustainability indicators.
- Support transparent, evidence-based decisions for future campus energy planning.

B2 Activities

- Identify key stakeholders involved in campus energy use and decision-making.
- Build problem and objective trees to structure the energy demand challenge.
- Select and rank sustainability indicators using Delphi and SRF methods.
- Collect baseline values for selected indicators.
- Define possible strategic actions and extract winner scenarios through Q-methodology.
- Compare scenarios against baseline indicator values.

3. Support System

Capture who is involved, what resources are needed, and how the project will be communicated.

C1 Stakeholders

- University Energy Manager
- Sustainability Office
- Facility Management Department
- Student Representatives
- Academic Staff
- Municipal Energy Authority
- Technology Provider
- Campus Users

C2 Resources

- Campus energy consumption data
- Building-level performance reports
- Sustainability strategy documents
- Technical expertise from facility managers
- Student and staff feedback
- Indicator databank and workshop outputs

C3 Dissemination

- Final project review dashboard
- Presentation to course tutors and professor
- Visual scenario comparison report
- Exported workshop outputs for academic submission
- Recommendations for campus sustainability planning

4. Impact on SDGs

Select Sustainable Development Goals and the project contribution notes.

SDGs

Selected SDGs

SDG 7
Affordable and Clean Energy

SDG 9
Industry, Innovation and Infrastructure

SDG 11
Sustainable Cities and Communities

SDG 12
Responsible Consumption and Production

SDG 13
Climate Action

The project contributes to sustainable energy transition by reducing campus energy demand, improving building performance, supporting evidence-based infrastructure decisions, and promoting climate-conscious behaviour among campus users.

Figure 18. Workshop 1 Idea Canvas interface.

Source: Authors.

Workshop 2.1 - Stakeholder Analysis

Workshop 2.1 supports the identification, classification, and interpretation of stakeholders involved in the project. Students can enter stakeholder names and classify them according to relevant attributes such as power, interest, level, typology, and available resources. These structured fields transform stakeholder mapping from a static visual exercise into reusable project data.

The module also supports the transition from initial actor identification to more formal stakeholder analysis. Actors introduced during the Idea Canvas can be refined, expanded, or

reorganised in this stage. This creates a direct connection between early project framing and the analytical classification of actors.

Stakeholder data are stored as project-related records, which allows them to remain available for later stages of the workflow. This is important because stakeholders influence the interpretation of problems, the selection of indicators, and the construction of scenarios. By preserving these records inside the platform, the prototype reduces the need to manually copy stakeholder information from whiteboards to spreadsheets or presentation files.

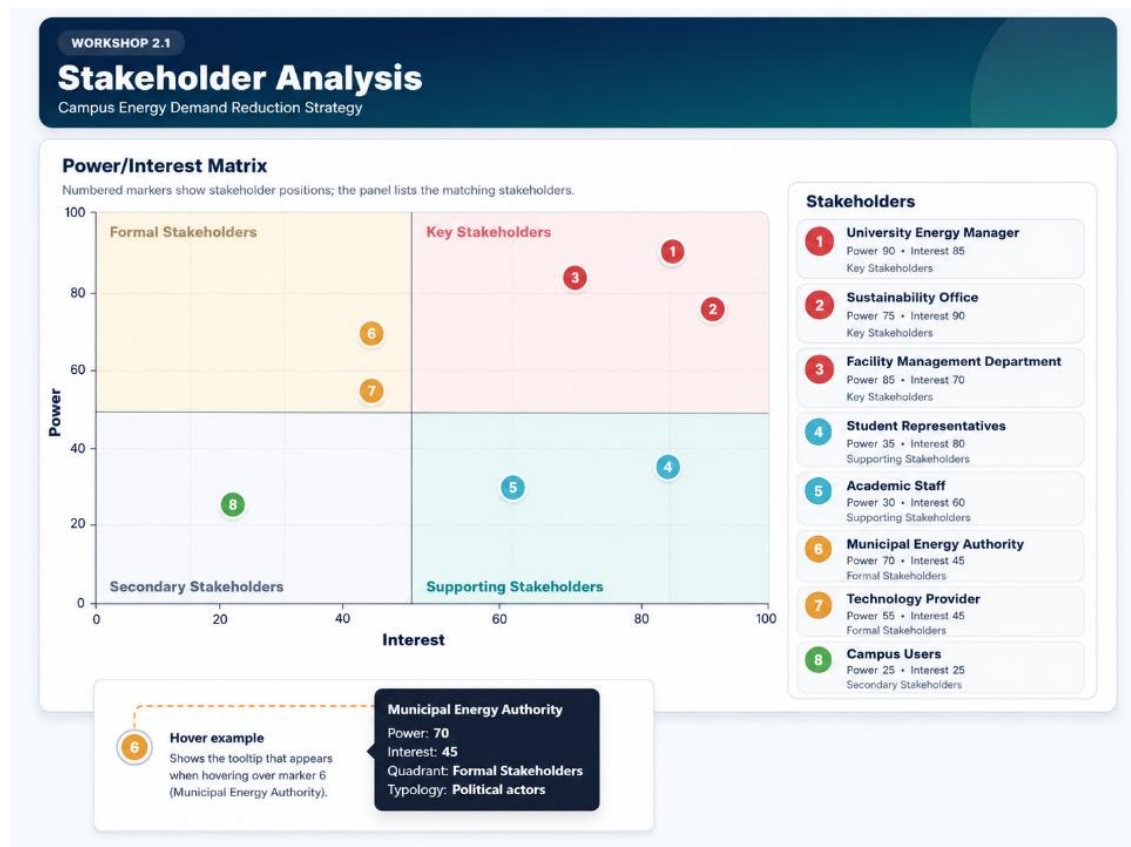


Figure 19. Workshop 2.1 Stakeholder Analysis interface.

Source: Authors.

Workshop 2.2 - Problem Tree

Workshop 2.2 implements the Problem Tree method. Students identify a central problem and organise related causes and effects into a structured hierarchy. The module helps students understand the logic of the problem situation by requiring them to distinguish between the main issue, its underlying causes, and its consequences.

The Problem Tree module is designed to support both visual reasoning and structured storage. Students can create and organise problem nodes, while the platform preserves the relationships between them. This means that the problem analysis does not remain only as a drawing or diagram; it becomes a stored analytical artefact within the project workspace.

This module is important for output traceability because later design decisions should be understandable in relation to the problem definition. When students select objectives, indicators, or scenarios, the Problem Tree provides a record of the reasoning that shaped the project's analytical direction.

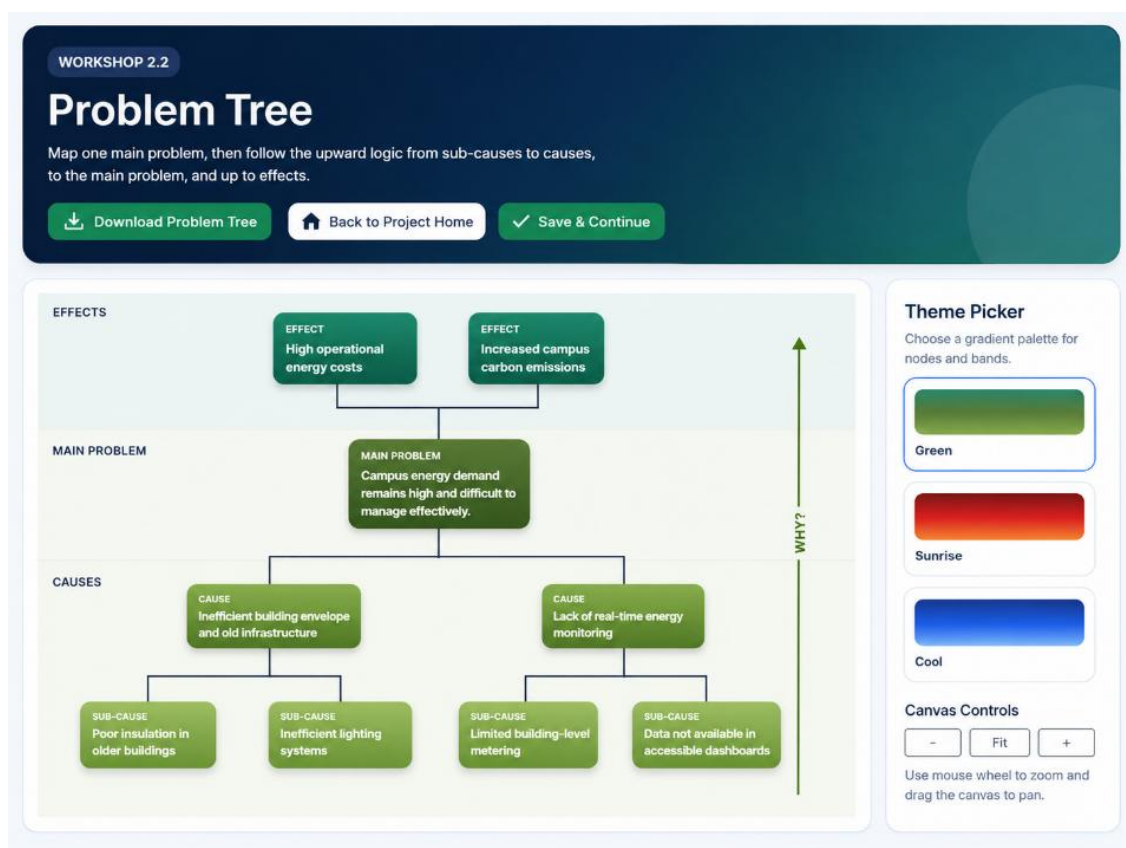


Figure 20. Workshop 2.2 Problem Tree interface.

Source: Authors.

Workshop 2.3 - Objective Tree

Workshop 2.3 develops the Objective Tree as a positive reformulation of the Problem Tree. Students transform identified problems into objectives and organise them into a coherent hierarchy. This supports the methodological transition from problem diagnosis to solution framing.

The Objective Tree module is separated from the Problem Tree because it represents a distinct stage of reasoning. While the Problem Tree explains what is wrong and why, the Objective Tree defines what should be improved and how the desired situation can be structured. This distinction helps students move from analysis to planning.

By storing objective-related data within the project workspace, the platform preserves the connection between problems and objectives. This makes the group's reasoning more reviewable and allows instructors to understand how the project evolved from an initial diagnosis into a structured set of goals.

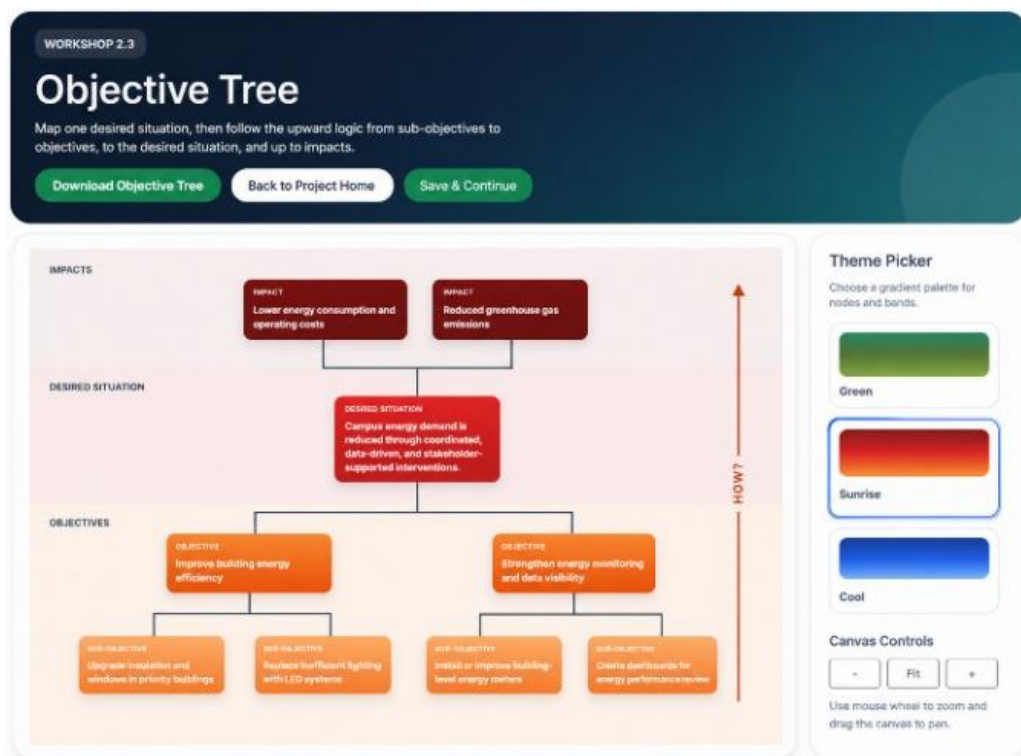


Figure 21. Workshop 2.3 Objective Tree interface.

Source: Authors.

Workshop 3.1 - Indicator Selection

Workshop 3.1 supports the selection of indicators from a structured indicator databank. Students can browse and select indicators relevant to their project, using information such as category, criterion, description, and unit of measure. This module marks the transition from qualitative framing to quantitative evaluation.

The indicator selection module is designed to reduce dependence on external spreadsheets. Instead of manually searching, copying, and reorganising indicator data, students work

inside a structured interface connected to the project workspace. Selected indicators become part of the project data and can be reused in ranking, analysis, scenario comparison, and later review.

This module is important because indicators function as the bridge between conceptual project objectives and measurable evaluation criteria. By storing selected indicators in the database, the platform helps maintain methodological continuity between the early analytical stages and the later decision-support stages.

WORKSHOP 3.1 ← Back to Project Home Save & Continue

Indicator Selection (Delphi)

Choose the project databank, review the available categories, and build a focused shortlist of up to 15 indicators for the next SRF ranking step.

Databank-driven selection workflow Maximum shortlist: 15 indicators Active databank: databank1

STEP 1 Select Indicator Databank Current: databank1

This choice is saved to the project and determines which workbook powers Workshop 3.1.

Select Indicator Databank

databank1 Load Databank

Choose source
Select the workbook that contains the indicator structure for this project.

Review categories
Open the accordion sections to inspect candidate indicators and descriptions.

Save shortlist
Save your accepted indicators before moving on to Workshop 3.2.

STEP 3 Explore Categories and Build the Shortlist Choose up to 15

Expand the main domains and subcategories to review the databank structure, then accept the indicators that belong in your final set.

Selections update the summary immediately, but the shortlist must still be saved before you continue.

B. Energy

C. Water

D. Solid Waste

E. Environmental quality

F. Transportation and mobility

F1. Performance and mobility services

F2. Green mobility

F3. Safety in mobility

F. Transportation and mobility > F3. Safety in mobility

ACCEPT	INDICATOR	CRITERION	DESCRIPTION	UNIT
☐	F3.1 - Percentage of the neighborhood designated as a pedestrian/car free zone	Pedestrian infrastructure	This indicator measures the proportion of the neighborhood that is designated as a pedestrian or car-free zone. It assesses the prioritization of pedestrian safety and ...	%
☐	F3.2 - Percentage of roads' length that has dedicated sidewalks	Availability of sidewalks	This indicator measures the proportion of road length that has dedicated sidewalks. It assesses the accessibility of the neighborhood for pedestrians, promoting connectivity and walkability.	%

Figure 22. Workshop 3.1 Indicator Selection interface.

Source: Authors.

Workshop 3.2 - Indicator Ranking / SRF

Workshop 3.2 implements indicator ranking through the Simos-Roy-Figueira method. The module allows students to organise selected indicators according to their relative importance and to represent priority gaps through white cards or equivalent ranking structures.

This module is one of the strongest examples of custom interaction in the prototype. The SRF method is difficult to support through a standard LMS because it requires specific

ranking logic, visual organisation, and calculation of weights. The Django implementation makes this method available through a dedicated interactive interface.

The ranking results are saved as project data, including the ordering of indicators, priority groups, white-card gaps, and calculated weights. These outputs are then available for later indicator analysis and scenario evaluation. This prevents the ranking process from becoming a disconnected spreadsheet activity and supports output traceability across the decision-making workflow.

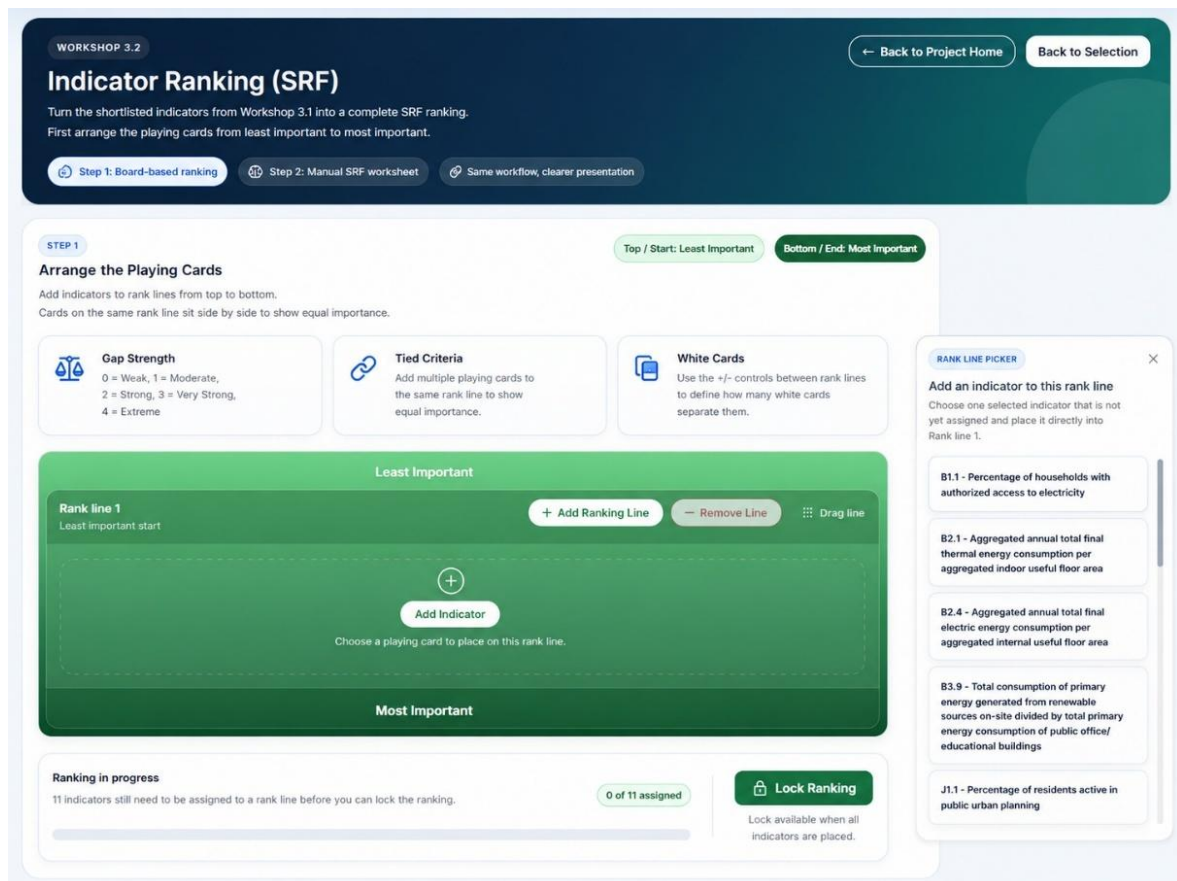


Figure 23. Workshop 3.2 Indicator Ranking / SRF interface.

Source: Authors.

Workshop 4.1 - Indicator Analysis

Workshop 4.1 supports the detailed analysis of selected indicators. Students can add values, supporting images, calculations, explanatory notes, and references related to each indicator. This module helps transform selected indicators into evidence-based analytical components. The module is designed to strengthen the connection between indicator selection and later evaluation. An indicator is not treated only as a label or criterion; it becomes associated with data, interpretation, and source material. This is especially important in a decision-making

course, where final scenario evaluation depends on the quality and reliability of the underlying indicator information.

By attaching supporting data and references to each project indicator, the platform improves reviewability. Instructors can better understand not only which indicators were selected, but also how students interpreted and supported them.

WORKSHOP 4.1 [Back to Project Home](#) [Download PDF Report](#) [Save & Continue](#)

Indicator Analysis

Upload the supporting charts or evidence for each accepted indicator, record the final calculation value, and store the references that justify each result for the next workshop steps.

In Progress

B3.9 - Total consumption of primary energy generated from renewable sources on-site divided by total primary energy consumption of public office/educational buildings

Share of renewable energy on-site, on total primary energy consumptions for public office/ educational building operations

%

Upload the chart, graph, or screenshot used for this indicator.

Click anywhere in this area to choose an image file.

FINAL CALCULATION VALUE

Store the result that will be reused in downstream scenario comparison.

Enter the final result used for this indicator

In Progress

J1.1 - Percentage of residents active in public urban planning

Community involvement in urban planning activities

Level

Upload the chart, graph, or screenshot used for this indicator.

Click anywhere in this area to choose an image file.

FINAL CALCULATION VALUE

Store the result that will be reused in downstream scenario comparison.

Enter the final result used for this indicator

RESOURCES / REFERENCES

Capture sources, formulas, links, notes, or short methodological justification.

In Progress

B1.1 - Percentage of households with authorized access to electricity

Access to electrical service %

Upload the chart, graph, or screenshot used for this indicator.

Click anywhere in this area to choose an image file.

FINAL CALCULATION VALUE

Store the result that will be reused in downstream scenario comparison.

Enter the final result used for this indicator

RESOURCES / REFERENCES

Capture sources, formulas, links, notes, or short methodological justification.

Add sources, formulas, notes, or justification

Figure 24. Workshop 4.1 Indicator Analysis interface.

Source: Authors.

Workshop 4.2 - SWOT Analysis

Workshop 4.2 implements SWOT Analysis through a structured board divided into strengths, weaknesses, opportunities, and threats. Students can add, edit, and organise SWOT items within the project workspace.

The SWOT module helps students synthesise earlier analytical work into a strategic interpretation of the project context. Internal factors, such as institutional capabilities or limitations, can be considered together with external opportunities and threats. This makes SWOT Analysis a bridge between detailed analysis and scenario-building.

By storing SWOT items as project-related outputs, the platform prevents the analysis from becoming a separate presentation slide disconnected from the rest of the workflow. The SWOT board remains part of the project record and can be revisited during scenario definition and final review.

SWOT Analysis
Campus Energy Demand Reduction Strategy

Strengths Capabilities, resources, and positive conditions already in place.	Weaknesses Internal limitations or gaps that reduce effectiveness.
Existing institutional commitment to sustainability	Some buildings have outdated infrastructure
Availability of technical expertise within the university	Energy data is not always centralised or easy to interpret
Presence of energy consumption data and facility records	Renovation budgets are limited
Strong potential for student engagement	Behavioural change among users is difficult to maintain

Opportunities External trends or openings the project could benefit from.	Threats External risks or adverse conditions that could affect outcomes.
Access to national and European sustainability funding	Rising energy prices
Growing demand for net-zero campus strategies	Delays in infrastructure renovation
Possibility to use the campus as a living laboratory	Resistance to behavioural or operational changes
New technologies for monitoring and energy optimisation	Uncertainty in funding availability

Figure 25. Workshop 4.2 SWOT Analysis interface.

Source: Authors.

Workshop 5 - Scenario Building

Workshop 5 supports scenario building through two alternative but complementary pathways: Q-Methodology and Start Park. This module is one of the most important synthesis stages of the platform because it transforms the analytical outputs produced in earlier workshops into structured scenario alternatives. While previous modules focus on framing the project, identifying actors, structuring problems and objectives, selecting indicators, ranking priorities, analysing data, and developing SWOT interpretations, Workshop 5 uses these outputs to support the construction of possible future scenarios.

The module is designed to avoid treating scenario building as an isolated creative exercise. Instead, scenario construction is positioned as a continuation of the project workflow. Stakeholders, objectives, indicators, and SWOT elements provide the background logic for

defining strategic options, interpreting different perspectives, and developing alternative scenario directions. By storing scenario-building data inside the project workspace, the platform supports data continuity and output traceability between earlier analytical stages and later scenario evaluation.

In the Q-Methodology pathway, students work with a structured sorting logic to represent different preferences, priorities, or strategic orientations. The platform supports the organisation of statements, actions, or strategic options through a Q-sort interface. This allows students to move beyond a simple list of ideas and arrange scenario-relevant elements according to a meaningful distribution. The Q-sort process helps make stakeholder perspectives and strategic tensions more explicit, because students must decide which elements are more central, less central, or more contested in the scenario-building process.

Figure 26 shows the Q-Methodology interface used for sorting and organising scenario-related statements or actions.



Figure 26. Workshop 5 Q-Methodology scenario-building interface.

Source: Authors.

The Q-Methodology path also supports the interpretation of scenario results. Once sorting data are saved, the platform can preserve Q-sort outputs, scenario labels, and related analytical information as part of the project record. This is important because the scenario results should not appear as final choices without explanation. They should remain connected to the reasoning process that produced them. In this way, the Q-Methodology pathway supports the traceability of scenario formation and prepares the data for later comparison in Scenario Analysis.

The Start Park pathway supports a different but equally important form of scenario construction. Instead of focusing mainly on sorting statements or strategic options, Start Park uses a more spatial, card-based, and design-oriented logic. Students can work with different scenario elements, such as stakeholder cards, solution cards, equipment cards, activity cards, spatial placements, and scenario summaries. This pathway is particularly useful when scenarios need to be understood not only as abstract strategic positions, but also as spatial configurations or design-based alternatives.

Figure 27 shows the Start Park interface used to construct spatial and card-based scenario alternatives.

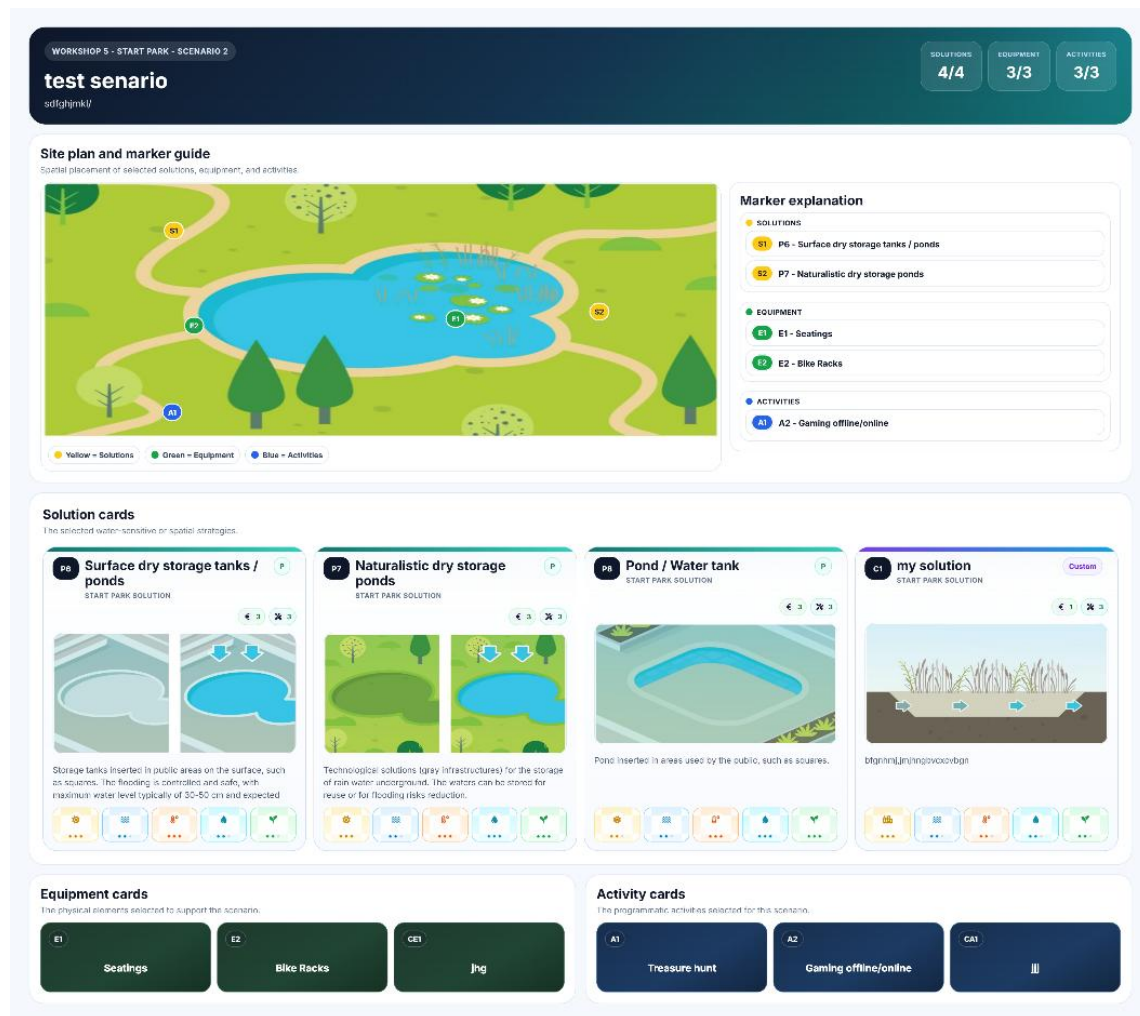


Figure 27. Workshop 5 Start Park scenario-building interface.

Source: Authors.

The inclusion of Start Park strengthens the flexibility of the prototype because it gives students another way to construct and compare alternative futures. Some decision-making activities are better supported through analytical sorting, while others require visual and spatial organisation. By including both Q-Methodology and Start Park, the platform recognises that scenario building can involve multiple forms of reasoning: interpretive, strategic, spatial, and design-based.

Both pathways share the same educational purpose: they help students move from analysis to synthesis. The Q-Methodology pathway supports the modelling and interpretation of different perspectives, while the Start Park pathway supports the construction of more spatial and tangible scenario alternatives. Together, they allow Workshop 5 to function as a bridge

between earlier analytical modules and later evaluation modules, especially Scenario Analysis, External Decision-Support Tools, Cost–Benefit Analysis, and Project Review.

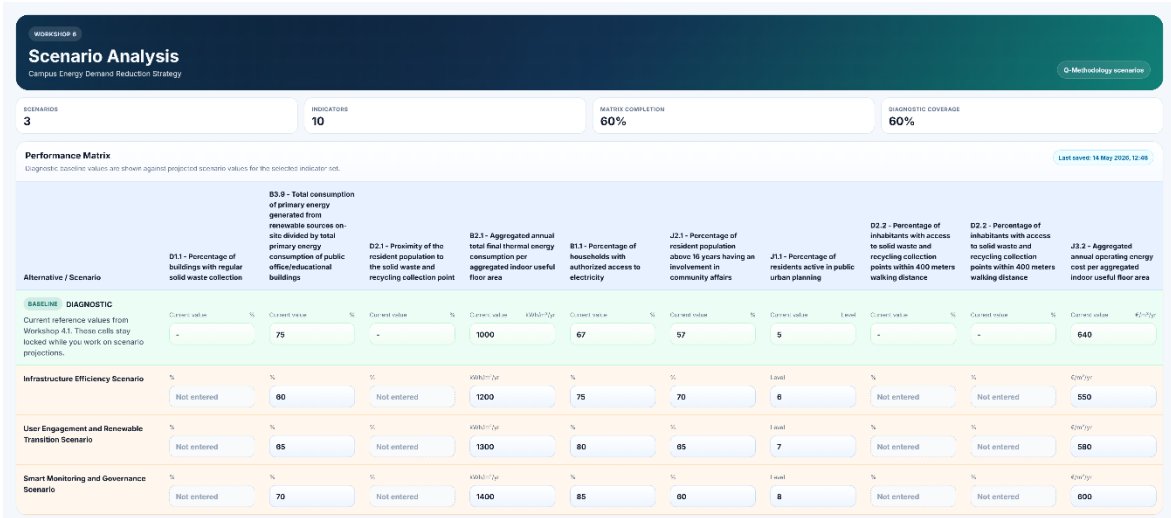
From a technical perspective, the scenario-building module relies on structured project-level data. Q-Methodology outputs, Q-sort data, scenario labels, Start Park cards, placements, and scenario summaries are stored as part of the project workspace. This ensures that scenario alternatives remain available for later evaluation and review. The module therefore contributes directly to the main objectives of the prototype: reducing tool fragmentation, supporting workflow continuity, preserving intermediate outputs, and making the development of scenarios more visible and reviewable.

Workshop 6 - Scenario Analysis

Workshop 6 supports the comparison and evaluation of scenarios. Students assess alternative scenarios against selected indicators and performance values, allowing different options to be compared in a more structured way.

This module depends directly on earlier modules, especially Indicator Selection, Indicator Ranking, Indicator Analysis, and Scenario Building. The value of the project-centred data model becomes clear here: scenario analysis can draw on accumulated project data rather than requiring students to reconstruct the evaluation framework manually.

The scenario-analysis outputs are stored as performance matrix data. This allows the results to be reviewed later and connected to the final project synthesis. The module therefore supports both quantitative comparison and output traceability.



Workshop 6 Scenario Analysis
Campus Energy Demand Reduction Strategy

SCENARIOS: 3 INDICATORS: 10 MATRIX COMPLETION: 60% DIAGNOSTIC COVERAGE: 60%

Performance Matrix
Diagnostic scenario values are shown against projected scenario values for the selected indicator set.

Alternative / Scenario	D1.1 - Percentage of buildings with regular solid waste collection	B3.9 - Total consumption of primary energy generated from renewable sources on-site divided by total primary energy consumption of public office/educational buildings	D2.1 - Proximity of the resident population to the solid waste and recycling collection point	B2.1 - Aggregated annual total fuel thermal energy consumption per aggregated indoor useful floor area	B1.1 - Percentage of households with authorized access to electricity	J2.1 - Percentage of resident population above 16 years having an involvement in community affairs	J1.1 - Percentage of residents active in public urban planning	D2.2 - Percentage of inhabitants with access to solid waste and recycling collection points within 400 meters walking distance	D2.2 - Percentage of inhabitants with access to solid waste and recycling collection points within 400 meters walking distance	J3.2 - Aggregated annual operating energy cost per aggregated indoor useful floor area
BASELINE	Current value: 75	Current value: 1000	Current value: 67	Current value: 57	Current value: 5	Current value: 6	Current value: 5	Current value: 640		
DIAGNOSTIC	Current reference values from Workshop 4.3. These cells stay locked while you work on scenario projections.									
Infrastructure Efficiency Scenario	Not entered	90	Not entered	1200	75	70	6	Not entered	Not entered	550
User Engagement and Renewable Transition Scenario	Not entered	85	Not entered	1300	80	65	7	Not entered	Not entered	580
Smart Monitoring and Governance Scenario	Not entered	70	Not entered	1400	85	60	8	Not entered	Not entered	600

Figure 28. Workshop 6 Scenario Analysis interface.

Source: Authors.

Workshop 7 - External Decision-Support Tools

Workshop 7 provides structured access to external decision-support tools selected by the professor or relevant to the course methodology. This module recognises that not every specialised analytical function needs to be rebuilt inside the prototype.

The purpose of this module is to reduce fragmentation while keeping methodological flexibility. Instead of leaving students to search for external tools independently or access them through informal links, the platform places them inside the structured workshop sequence. This helps preserve the logic of the workflow even when some specialised analysis remains outside the Django environment.

This module also supports integration readiness at the pedagogical level. It shows that the platform can act as a central workshop environment while still connecting students to external resources when appropriate.

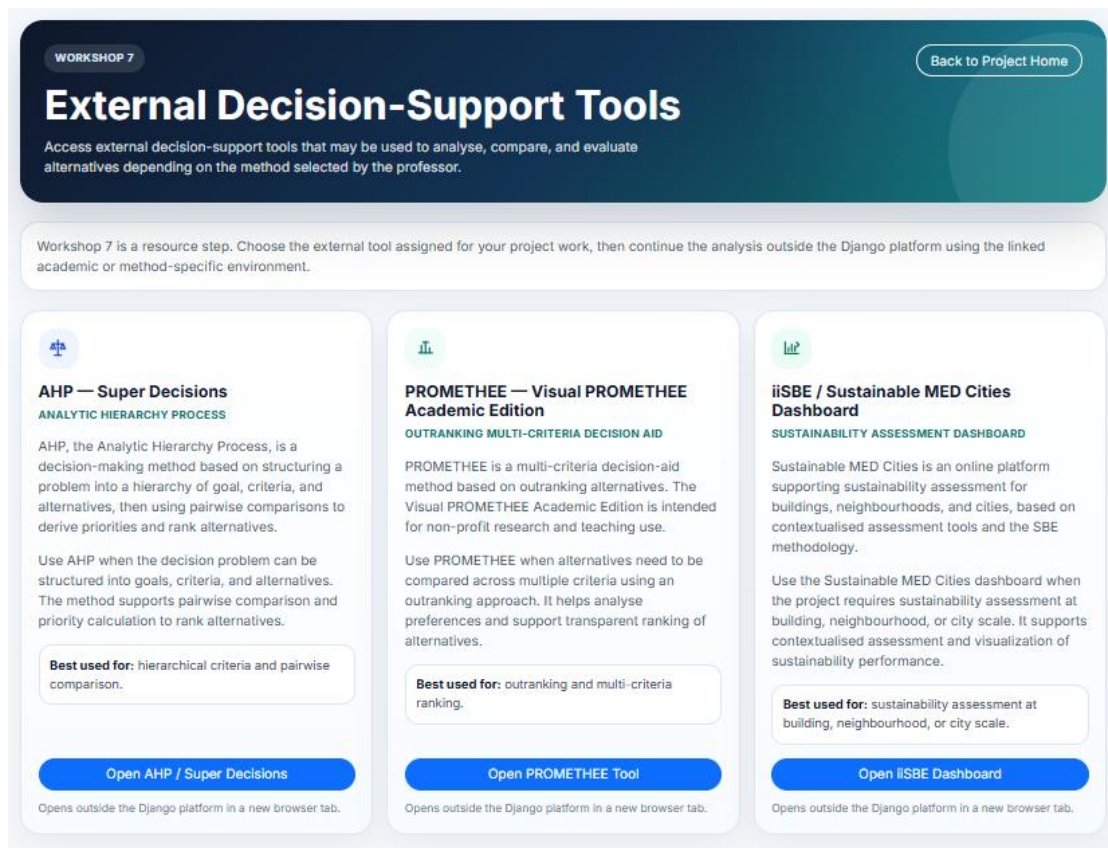


Figure 29. Workshop 7 External Decision-Support Tools interface.

Source: Authors.

Workshop 8 - Cost–Benefit Analysis / CBA

Workshop 8 supports Cost–Benefit Analysis / CBA. This module allows students to organise cost items, benefit items, assumptions, and evaluation values related to a selected scenario or intervention.

The CBA module extends the platform from qualitative, strategic, and multi-criteria analysis toward economic evaluation. It gives students a dedicated space to structure financial reasoning and compare the expected costs and benefits of project alternatives.

By storing CBA data inside the project workspace, the platform connects economic evaluation to the rest of the decision-making workflow. This makes the final assessment more coherent because cost–benefit reasoning can be interpreted alongside stakeholder analysis, indicator weighting, scenario analysis, and previous project decisions.

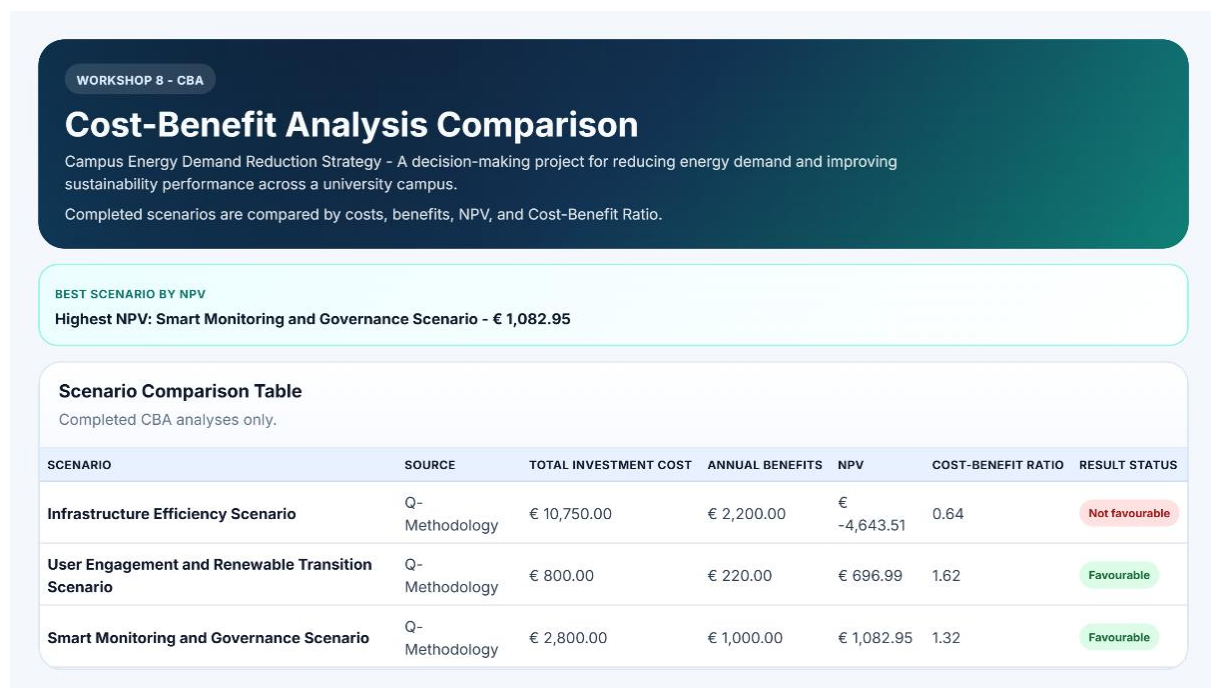


Figure 30. Workshop 8 Cost–Benefit Analysis / CBA interface.

Source: Authors.

Project Review

The Project Review module provides a final synthesis space where saved outputs from the previous modules can be reviewed together. It is not treated as another workshop, but as a review-oriented section that supports reflection, checking, and final preparation.

This module brings together the main outputs of the project workflow, including project framing, stakeholders, problem and objective structures, selected indicators, ranking results, indicator analysis, SWOT items, scenario-building data, scenario-analysis outputs, and CBA results. In this way, Project Review helps students understand the project as a complete process rather than as a collection of disconnected tasks.

For instructors and teaching assistants, Project Review improves reviewability because it provides a clearer view of how the project developed across stages. It supports output traceability by connecting the final synthesis to the intermediate work that produced it.

Overall, the workshop module implementation demonstrates how the prototype operationalises the technical objectives derived in Chapter 3. Workflow continuity is supported by the project-centred module sequence. Accessibility is supported through consistent interface structures and embedded guidance. Collaboration is supported through shared project workspaces and persistent group outputs. Output traceability is supported by saving intermediate artefacts across the workflow. Integration readiness is supported by designing the Django platform as a custom interactive workshop layer that can complement the institutional LMS layer in future deployment.

Table 10. Workshop modules and their main prototype functions.

Module	Main function	Main stored output
Workshop 1 Idea Canvas	Frames the project context and initial ideas	Project overview data
Workshop 2.1 Stakeholder Analysis	Identifies and classifies actors	Stakeholder records
Workshop 2.2 Problem Tree	Structures central problem, causes, and effects	Problem-tree nodes
Workshop 2.3 Objective Tree	Converts problems into objective structures	Objective-tree data
Workshop 3.1 Indicator Selection	Selects relevant indicators from the databank	Accepted project indicators
Workshop 3.2 Indicator Ranking / SRF	Ranks and weights selected indicators	Ranking order, groups, weights
Workshop 4.1 Indicator Analysis	Adds values, evidence, images, and references	Indicator-analysis records
Workshop 4.2 SWOT Analysis	Synthesises strengths, weaknesses, opportunities, and threats	SWOT items

Workshop 5 Scenario Building	Builds scenarios through Q-Methodology and Start Park	Scenario-building data
Workshop 6 Scenario Analysis	Compares scenarios against selected indicators	Performance matrix data
Workshop 7 External Decision-Support Tools	Provides structured access to professor-selected external tools	Tool references / links
Workshop 8 Cost–Benefit Analysis / CBA	Structures economic evaluation of selected scenarios	CBA costs, benefits, assumptions, and results
Project Review	Aggregates saved outputs for final synthesis and review	Review-ready project summary

Source: Authors

4.8 Interface Design and Interaction Logic

The interface design of the prototype was developed to support two connected objectives: usability and methodological understanding. Because the platform contains several different workshop modules, the interface needed to remain consistent enough for users to recognise it as one coherent environment, while also flexible enough to support different types of interaction, such as forms, visual matrices, drag-and-drop ranking, hierarchical trees, scenario dashboards, and export functions.

The general visual language follows a modern academic dashboard style. Workshop pages use structured headers, rounded content cards, soft shadows, status indicators, action buttons, modals, and section-based layouts. This consistency is important because the platform supports a complex multi-stage workflow. A stable interface structure helps students understand where they are, what they are expected to do, and how each module relates to the wider project sequence.

A repeated page structure is used across the prototype. Each workshop page normally includes a header area with the workshop number, module title, short explanation, navigation controls, and main action buttons. These elements help users orient themselves immediately when entering a module. Buttons such as “Back to Dashboard”, “Save”, “Save and Continue”, and export controls are positioned consistently where possible, reducing the need for users to relearn navigation patterns across modules.

Content cards are used to organise inputs and outputs into smaller, readable units. This is especially useful because many workshop activities involve several pieces of information at the same time. In the Idea Canvas, cards help separate different parts of the project framing.

In Stakeholder Analysis, they support data entry and summary views. In Indicator Selection and Indicator Analysis, they help organise technical indicator information. In Scenario Building and Project Review, cards make complex analytical outputs easier to scan and compare.

The interaction logic is based on learning-by-doing. Instead of asking students only to read instructions or upload final files, the platform requires them to interact directly with the methods of the course. For example, students classify stakeholders, build problem and objective structures, select indicators, organise rankings, analyse scenarios, and review project outputs inside the platform. This supports the pedagogical aim of transforming workshop methods into active digital processes rather than static documentation.

Several modules use visual and interactive components to make methodological tasks more understandable. The stakeholder module uses visual classification logic to support the interpretation of power and interest relationships. The Problem Tree and Objective Tree modules use hierarchical structures to represent causal and goal-oriented reasoning. The SRF ranking module uses drag-and-drop interaction to make priority ordering more concrete. Scenario-related modules use structured dashboards and matrices to support comparison, interpretation, and review.

Visual scaffolding is also used to reduce cognitive load. Complex methods are supported through short instructions, labels, legends, explanatory notes, and validation messages. These elements are important because the platform is designed for students who may not be experts in either the digital tool or the decision-making methods. The interface therefore does not only collect data; it also guides users through the logic of the task.

Modals are used where additional explanation is needed without interrupting the workflow. This allows the platform to provide contextual help while keeping the main page clean. For example, explanatory modals can clarify methodological concepts, calculation logic, or interpretation rules. This supports accessibility and low-friction interaction because students can access guidance at the point where they need it instead of searching through external documents.

The prototype also uses immediate feedback to support user confidence. Save confirmations, progress indicators, completion signals, and visible output updates help users understand whether their actions have been recorded correctly. This is particularly important in a multi-

stage platform, where uncertainty about whether data has been saved could weaken trust in the system and encourage students to return to external files.

From a technical perspective, the interface is implemented through Django templates, HTML, CSS, Bootstrap, and JavaScript. Bootstrap supports the responsive layout, cards, buttons, modals, and grid structures, while custom CSS defines the academic dashboard aesthetic. Interactive behaviour is supported by JavaScript libraries such as Sortable.js for drag-and-drop activities and D3.js for visual analytical components.

The interface design also supports output traceability. Since workshop outputs are saved within the project workspace and remain accessible through the platform, users can return to previous stages and review how the project developed. The Project Review section strengthens this logic by bringing together the main outputs of the workflow in a final synthesis space.

Overall, the interface design and interaction logic translate the technical objectives of Chapter 3 into practical user experience decisions. Workflow continuity is supported through consistent navigation and project-based access. Accessibility is supported through clear layouts, repeated patterns, instructions, and visual feedback. Collaboration is supported through shared project workspaces and persistent outputs. Output traceability is supported through stored artefacts and reviewable module results. In this way, the interface contributes directly to the thesis aim of reducing digital fragmentation in workshop-based learning.

4.9 Export and Reporting Functions

Export and reporting functions form an important part of the prototype because workshop-based courses require students to document, review, and communicate the outputs produced during the learning process. In the previous fragmented workflow, students often had to manually copy data from whiteboards, spreadsheets, and presentation files in order to prepare final submissions. The prototype reduces this burden by embedding export and reporting functions directly within the platform environment.

The export system supports the broader objective of output traceability. By allowing students to export intermediate outputs from different modules, the platform makes the development of the project easier to document and review. Exports are therefore not treated only as final deliverables, but also as evidence of the analytical steps followed throughout the workflow.

The prototype supports three main export formats: CSV, PNG, and PDF. CSV exports are used when the main output is structured tabular data, such as stakeholder lists, selected indicators, ranking results, Q-sort records, scenario matrices, or performance values. These exports allow structured data to be reviewed, verified, and reused in external reporting workflows when necessary.

PNG exports are used for visual workshop artefacts. This is important because several modules produce outputs that are best understood visually, such as the Idea Canvas, stakeholder matrices, problem and objective structures, SRF ranking boards, Q-sort distributions, scenario dashboards, and performance matrices. Image exports allow these visual artefacts to be included in reports, presentations, or academic documentation without requiring students to manually recreate them outside the platform.

PDF exports are used for more formal reporting outputs. In particular, modules such as Indicator Analysis and Project Review benefit from structured PDF generation because they combine textual descriptions, values, images, references, and synthesis content. PDF reports support a more complete and readable form of academic documentation than isolated screenshots or spreadsheet files.

From a technical perspective, export functions are implemented through a combination of client-side and server-side mechanisms. PNG exports are generated through browser-based capture logic such as `html2canvas` or `SVG/canvas` export methods, depending on the visual structure of the module. CSV exports are generated through Django HTTP responses, allowing structured records to be downloaded directly from the relevant module. PDF reports are generated server-side, using reporting tools such as ReportLab where formal document structure is required.

The export functions also support accessibility and low-friction interaction. Instead of asking users to move between multiple tools to prepare outputs, the platform provides export controls close to the relevant workshop content. This keeps the reporting process connected to the activity itself and reduces the risk of losing information during manual transfer.

In the current prototype, export functions are implemented selectively according to the type of output produced by each module. Modules that generate structured data use CSV exports, modules that generate visual artefacts use PNG exports, and modules that require formal documentation use PDF reports. Future versions of the platform could extend this logic by

introducing unified portfolio export, automated report compilation, and professor-oriented export dashboards for comparing outputs across groups.

Overall, the export and reporting functions strengthen the prototype's role as a connected workshop environment. They support workflow continuity by keeping outputs inside the platform until they are needed externally, improve output traceability by preserving intermediate artefacts, and help students prepare reviewable academic documentation without returning to a fragmented set of external tools.

Table 11. Export Functions Across the Platform

Module	Export Formats	Purpose
Workshop 1 - Idea Canvas	PNG	Visual documentation of project framing and early project ideas
Workshop 2.1- Stakeholder Analysis	PNG, CSV	Stakeholder matrix, stakeholder list, and structured actor data
Workshop 2.2 - Problem Tree	PNG	Visual documentation of central problem, causes, and effects
Workshop 2.3 - Objective Tree	PNG	Visual documentation of objectives and hierarchical relationships
Workshop 3.1 - Indicator Selection	PNG, CSV	Indicator selection summary and accepted indicator records
Workshop 3.2 - Indicator Ranking / SRF	PNG, CSV	Ranking board, priority groups, white-card logic, and calculated weights
Workshop 4.1 - Indicator Analysis	PDF, PNG where relevant	Indicator report with values, supporting images, and references
Workshop 4.2 - SWOT Analysis	PNG	Visual documentation of strengths, weaknesses, opportunities, and threats
Workshop 5 - Scenario Building	PNG, CSV	Q-sort outputs, scenario structures, factor-loading visuals, and scenario data
Workshop 6 - Scenario Analysis	PNG, CSV	Performance matrix and scenario-comparison values
Workshop 7 - External Decision-Support Tools	Link/reference export or documentation note	Structured access to external tools and related references
Workshop 8 - Cost–Benefit Analysis / CBA	CSV, PDF where relevant	Cost items, benefit items, assumptions, and economic evaluation results

Project Review	PDF	Consolidated project synthesis and review-ready portfolio summary
----------------	-----	---

Source: Authors

4.10 Technical Limitations and Future Development

The prototype demonstrates the feasibility of a Django-based custom interactive workshop layer for supporting a connected decision-making workflow. However, it remains a proof-of-concept implementation developed within the scope of a master's thesis. For this reason, several limitations must be recognised. These limitations do not invalidate the prototype; rather, they clarify the boundary between the implemented platform and the requirements of a production-level institutional system.

The first limitation concerns deployment. The prototype was developed and tested mainly in a local development environment. This was appropriate for iterative development and functional validation, but it does not reproduce the technical conditions of a full institutional deployment. A production version would require server configuration, domain management, HTTPS, environment-variable management, institutional backup procedures, monitoring, and long-term maintenance planning.

The second limitation concerns database infrastructure. The current prototype uses SQLite during development and local testing. SQLite is suitable for lightweight development, but it is not the most appropriate solution for a production platform with simultaneous users, multiple groups, and long-term institutional data storage. A future version should migrate to PostgreSQL or another production-ready relational database system in order to support concurrency, scalability, stronger backup procedures, and more robust data-management practices.

The third limitation concerns Moodle integration. The architecture is designed according to a Moodle–Django platform strategy, where Moodle functions as the institutional LMS layer and Django functions as the custom interactive workshop layer. However, full institutional Moodle integration through LTI 1.3 is treated as a future deployment direction rather than as a fully validated production feature of the current prototype. Future work should implement and test LTI-based launch, authentication, role transmission, course linking, and user provisioning within the actual institutional Moodle environment.

The fourth limitation concerns role management and instructor-facing functionality. The current prototype focuses primarily on the student-side workflow and on the implementation

of workshop modules. Although the platform structure supports future role-sensitive access, more advanced professor and teaching-assistant dashboards remain to be developed. Future versions should provide monitoring views, group comparison tools, contextual feedback functions, progress analytics, and review interfaces that allow instructors to follow group development more effectively.

The fifth limitation concerns collaboration. The prototype supports collaboration mainly through shared project workspaces and persistent outputs. This means that group members can work around the same project structure and preserve their results across modules. However, full real-time synchronous multi-user editing, live cursors, simultaneous state updates, and conflict-resolution mechanisms are not implemented as core features. These functions would require additional technologies such as WebSockets, asynchronous task handling, and more advanced client-side state management.

The sixth limitation concerns output traceability. The prototype preserves key intermediate outputs by storing module data inside the project workspace. This supports reviewability and allows the project to be followed across stages. However, detailed action logging, version history, user-level change tracking, and rollback mechanisms remain future development directions. These features would be necessary if the platform were used for formal assessment of group contribution or detailed process evaluation.

The seventh limitation concerns interface maintainability. During development, some CSS and JavaScript logic was embedded directly within individual templates to support rapid iteration and module-specific interaction. This approach was useful during prototype development, but it should be refactored in future versions. Reusable CSS files, modular JavaScript components, shared interface patterns, and component-based templates would make the platform easier to maintain and extend.

The eighth limitation concerns accessibility and responsive behaviour. The interface was designed to be clear, readable, and web-based, but some complex visual modules, such as trees, matrices, ranking boards, and scenario dashboards, are naturally more suitable for desktop use than for small mobile screens. Future development should include more systematic accessibility testing, keyboard navigation checks, screen-reader compatibility improvements, colour-contrast validation, and responsive optimisation for tablets and smaller devices.

The ninth limitation concerns export and reporting functions. The prototype provides selective export functions such as CSV, PNG, and PDF depending on the module. However, export logic could be expanded into a more unified reporting system. Future versions could generate a complete project portfolio automatically, combining selected outputs from all workshop modules into one structured report for students, teaching assistants, and professors.

The final limitation concerns scalability and validation under real course conditions. The prototype has not yet been tested with multiple groups working concurrently in a live classroom or institutional server environment. Future testing should examine performance under simultaneous use, data consistency during collaborative sessions, user-role behaviour, and the ability of the platform to support a complete course cycle from project creation to final review.

Overall, these limitations define a clear future development path. The current prototype establishes the core logic of the platform: project-based workspaces, connected workshop modules, persistent outputs, interactive methodological tools, and export functions. Future development should move from proof-of-concept toward institutional deployment by strengthening Moodle integration, database scalability, real-time collaboration, instructor dashboards, detailed output traceability, accessibility compliance, and automated reporting.

Table 12. Technical limitations and future development directions.

Area	Current limitation	Future development direction
Deployment	Prototype tested mainly in local development environment	Production server, HTTPS, monitoring, backups, and maintenance plan
Database	Sqlite used for development and local testing	Migration to postgresql or another production-ready database
Moodle integration	Moodle–Django strategy defined architecturally, but full institutional integration remains future work	LTI 1.3 launch, role transmission, course linking, and user provisioning
Role management	Student-side workflow prioritised	Professor and teaching-assistant dashboards, group monitoring, and review tools
Collaboration	Persistent shared project outputs implemented	Real-time multi-user editing, live updates, conflict management

Output traceability	Main module outputs are stored and reviewable	Detailed action logging, version history, user-level change tracking
Interface maintainability	Some CSS and javascript embedded in individual templates	Refactoring into reusable static files and modular components
Accessibility	General readable interface implemented; complex modules remain desktop-oriented	Accessibility testing, keyboard support, screen-reader checks, responsive optimisation
Export and reporting	Selective CSV, PNG, and PDF exports implemented	Unified project portfolio export and professor-oriented reporting
Scalability validation	Not tested under full live course conditions	Concurrent user testing, institutional pilot, and performance evaluation

Source: Authors

Chapter 5. Validation and Fine-Tuning

5.1 Validation and Fine-Tuning within the Design Thinking Process

This chapter presents the validation and fine-tuning phase of the prototype developed in Chapter 4. Within the Design Thinking framework adopted in this thesis, Chapter 5 corresponds to the **Test** and **Iterate** stages. Its purpose is to examine whether the Django-based custom interactive workshop layer supports the intended educational workflow in a clear, usable, accessible, and pedagogically meaningful way, and to explain how feedback from the validation process informed targeted prototype refinements.

The validation phase was designed as a formative evaluation rather than as a large-scale institutional deployment test. This distinction is important because the prototype remains a proof-of-concept implementation developed and tested mainly in a local environment. The platform is intended to complement the institutional LMS layer rather than replace it, while full Moodle integration through LTI 1.3 remains a future institutional deployment direction. Therefore, the validation focuses on the prototype's usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness within its defined research scope.

The validation process combined direct observation, structured survey responses, open-ended comments, and thematic analysis. Participants interacted with the prototype, explored the workshop sequence, identified usability issues, and evaluated the extent to which the platform supported workflow continuity, data continuity, output traceability, low-friction interaction, and educational usefulness. The findings were then used to identify strengths, weaknesses, and priority areas for refinement.

The fine-tuning phase did not aim to redesign the platform from the beginning. Instead, it focused on targeted revisions to issues that emerged during validation, including navigation reliability, form validation feedback, interface consistency, progression logic, workshop structure, and clearer user guidance. In this way, Chapter 5 closes the design loop between prototype development and user-based evaluation by showing how the prototype was tested, what was learned, and how the platform was refined in response to the findings.

5.2 Validation Design and Setup

The validation phase was designed to assess the prototype as an educational workflow environment rather than as a set of isolated software features. The main objective was to understand whether the platform could support the workshop-based decision-making process in a way that was usable, clear, accessible, coherent, and educationally useful. The validation also aimed to identify practical problems requiring refinement before the prototype could be considered a stable proof-of-concept.

The validation design followed the methodological framework established in Chapter 2. In that chapter, the validation and fine-tuning phase is defined as the Test and Iterate stage of the Design Thinking process, based on user validation, survey feedback, thematic analysis, and targeted refinement. The evaluation was therefore not intended to produce statistically generalisable results. Instead, it was used as a formative evaluation to identify meaningful strengths, weaknesses, usability problems, and improvement priorities in relation to the thesis objectives.

Validation Goals and Evaluation Criteria

The validation focused on six criteria: usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. These criteria were selected by combining the educational needs identified in the previous chapters with established evaluation concepts from usability engineering, human-system interaction, accessibility standards, technology acceptance, educational technology, and software-quality literature. Usability and clarity concern whether users can understand the interface, navigate the workflow, and complete tasks without unnecessary friction; these concerns are central to usability evaluation and human-system interaction (Nielsen, 1994; ISO, 2018). Accessibility is linked to the need for web-based systems to support users with different abilities, devices, and interaction conditions (W3C, 2024). Workflow coherence reflects the thesis-specific requirement that project-based and workshop-based learning depends on continuity across stages and preservation of intermediate outputs (Guo et al., 2020; Petchamé et al., 2023). Perceived usefulness draws on technology acceptance literature, where usefulness concerns the extent to which a system is perceived as supporting task performance (Davis, 1989). Completeness is used to assess whether the proof-of-concept prototype covers the functional scope required

to support the intended workflow, in line with software-quality concepts such as functional suitability and usability (ISO/IEC, 2023).

Usability refers to whether users can navigate the platform and complete the main tasks without unnecessary friction. Clarity refers to whether the instructions, page structure, and progression logic are understandable. Accessibility refers to whether users with different levels of technical confidence can interact with the prototype without excessive difficulty. Workflow coherence refers to whether the platform is perceived as a connected project-based workspace rather than a sequence of disconnected pages. Perceived usefulness refers to whether users consider the platform valuable for the course context. Completeness refers to whether the prototype appears sufficiently developed within the defined proof-of-concept scope of the thesis.

Table 13. Validation criteria and methodological alignment

Validation criterion	Meaning in this thesis	Connection to thesis requirements
Usability	Users can navigate the platform and complete tasks without unnecessary friction.	Accessibility and low-friction interaction
Clarity	Instructions, labels, buttons, and progression logic are understandable.	Workflow continuity and user guidance
Accessibility	Users with different levels of technical confidence can use the platform effectively.	Inclusive and low-friction interaction
Workflow coherence	The platform feels like a connected workflow rather than disconnected pages.	Workflow continuity and data continuity
Perceived usefulness	Users see the platform as valuable for the Decision-Making course.	Structured collaboration and educational relevance
Completeness	The prototype appears sufficiently developed within its proof-of-concept scope.	Output traceability and prototype maturity

Source: Authors.

Participant Profile and Sample Justification

The validation session was conducted in person with six former students of the target Decision-Making course at Politecnico di Torino. The course professor was also present during the session and contributed as an expert academic observer. This participant profile was appropriate because the former students had already experienced the original workshop workflow and could compare the prototype with the fragmented use of separate tools such as spreadsheets, visual boards, shared files, and external decision-support platforms.

The professor's presence added an expert instructional perspective. While the students evaluated the platform mainly from the perspective of usability, navigation, collaboration, and learning experience, the professor was able to assess the prototype in relation to the pedagogical structure of the course, methodological coherence, and potential institutional usefulness.

After the session, seven Google Form responses were collected: six from former student participants and one from the professor. For this reason, the survey results are interpreted as a combined formative evaluation, including both informed student-user feedback and expert instructor feedback. The sample was not intended to support statistical generalisation. Instead, it was suitable for qualitative prototype validation, where the objective is to identify recurrent usability issues, educational strengths, and practical directions for improvement.

Testing Environment and Prototype Status

The validation was conducted using the proof-of-concept version of the Django-based prototype. The prototype was tested in a local development environment, consistent with the scope of the thesis. Therefore, the validation did not assess production-level deployment, institutional server performance, long-term scalability, or full Moodle integration through LTI 1.3. These aspects remain future development directions.

This distinction is important because the aim of the session was not to validate the platform as a finished institutional system. Instead, the session evaluated whether the prototype's core design logic was understandable and useful: a project-based workspace, connected workshop modules, clearer navigation between stages, data continuity, output traceability, and low-friction interaction. The testing environment therefore reflected the prototype's current research status as a functional proof of concept.

Session Procedure

The validation session took place on 31 March 2026 and lasted approximately one hour and thirty minutes. At the beginning of the session, the platform was introduced to the participants, including its purpose, general structure, project-based workflow, and relationship to the existing course process. The participants were then invited to interact with the prototype and explore the main workshop sequence.

During the hands-on phase, participants navigated through the platform, tested key workshop pages, interacted with visual and form-based components, and noted difficulties, unclear elements, technical issues, or suggestions for improvement. The session focused especially on the user experience of moving through the platform as a connected educational workflow. The researcher observed how participants interacted with the interface, where they hesitated, which buttons or page transitions created confusion, and which parts of the platform appeared immediately understandable.

The course professor also observed the session and completed the same feedback form after interacting with the prototype. This made it possible to combine student-user observations with an expert view of the platform's pedagogical and methodological relevance.

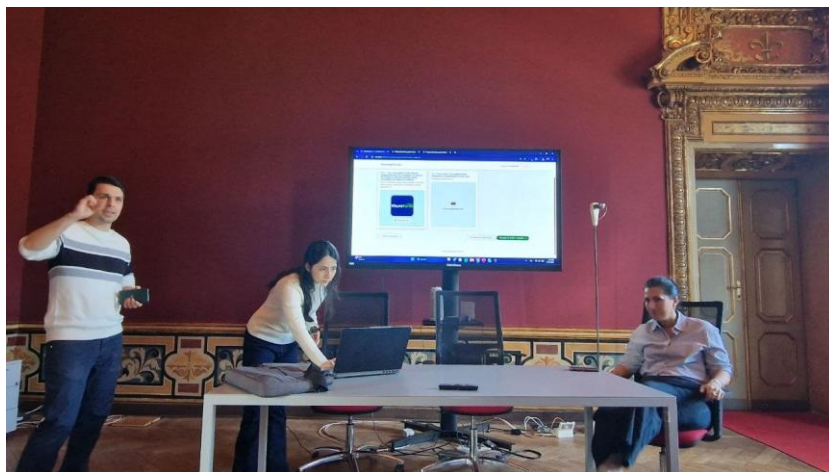


Figure 31. Overview of the validation session at Politecnico di Torino.

Source: Authors.



Figure 32. Participants navigating the workshop sequence on the localhost prototype at validation session

Source: Authors.

Survey Instrument and Analysis Method

At the end of the session, participants completed a Google Form survey. The survey included both Likert-scale statements and open-ended questions. The Likert-scale statements were used to collect structured feedback on the platform's workflow, usability, data continuity, collaboration support, interactive features, and perceived institutional fit. The open-ended questions were used to collect more detailed comments about advantages, disadvantages, bugs, confusing screens, missing features, and overall educational value.

The use of open-ended questions was important because many relevant prototype issues cannot be fully captured through numerical ratings alone. For example, a participant may give a positive score to the general idea of workflow continuity while still identifying a specific navigation button, validation message, or workshop page that requires improvement. The combination of rating questions and written comments therefore allowed the validation to capture both overall perception and specific implementation issues.

The survey responses were analysed through criterion-based thematic grouping. Comments were reviewed and organised according to the same six validation criteria used in the survey instrument: usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. This ensured that the interpretation of open-ended responses remained aligned with the validation framework rather than introducing a separate set of themes.

Table 14. Survey Instrument: Questions, Type, and Criterion Alignment

Statement / Question	Type	Criterion
Step-by-step locking mechanism guided understanding of methodology's logical sequence	Likert 1–5	Usability; Clarity
Having all tools in a single platform reduced cognitive load of managing the project	Likert 1–5	Usability; Workflow Coherence
Platform made it easy to track how decisions made in early workshops affected final scenario results	Likert 1–5	Workflow Coherence
Using this single platform would make collaboration easier compared to separate tools (Excel, Miro)	Likert 1–5	Perceived Usefulness
Visual feedback elements (progress badges, Next Recommended Step) supported workflow continuity	Likert 1–5	Clarity; Workflow Coherence

Data flowed smoothly from one workshop to the next without loss of information	Likert 1–5	Workflow Coherence; Completeness
User interface made complex tasks (building trees, ranking cards) feel intuitive and easy	Likert 1–5	Usability; Clarity
I was able to use the platform independently without much technical help	Likert 1–5	Accessibility
Interactive features (drag-and-drop, chart loading) worked smoothly without lagging or crashing	Likert 1–5	Usability
I can easily see this platform fitting into a university's standard learning system (Moodle/Canvas)	Likert 1–5	Perceived Usefulness; Completeness
Biggest advantage and disadvantage of all-in-one platform vs separate tools	Open-ended	Perceived Usefulness
Bugs, confusing screens, or data transfer errors , specify which workshop	Open-ended	Usability; Completeness
Suggested features to improve collaboration	Open-ended	Perceived Usefulness; Accessibility
Final thoughts on educational value, technical design, or overall user experience	Open-ended	All criteria

Source: Authors.

5.3 Validation Session and Real-Time Observations

The validation session provided an opportunity to observe how users interacted with the prototype in a realistic testing situation. While the survey responses collected after the session provided structured and reflective feedback, the live session made it possible to identify immediate usability problems, hesitation points, navigation issues, and interface behaviours that affected the user experience during direct interaction with the platform.

The session was particularly useful because participants were not evaluating the prototype only as a visual interface. They were interacting with it in relation to a course workflow that they already knew from previous experience. This allowed them to assess whether the prototype reduced the fragmentation of the original workshop process, whether the sequence of modules was understandable, and whether the platform supported a clearer connection between early-stage project framing and later analytical outputs.

The testing focused on the prototype as a connected project-based workspace. Participants explored the main available workshop flow, including project creation, dashboard navigation, Idea Canvas, stakeholder analysis, problem and objective structuring, indicator-related activities, SWOT analysis, and scenario-building functions. Some comments also referred to modules that were absent, incomplete, or still under development at the time of validation, especially scenario analysis, external decision-support tool links, and final synthesis/reporting functions. These comments became important inputs for the later fine-tuning and completion of the prototype scope.

During the hands-on phase, participants engaged actively with the prototype. They moved between modules, tested form inputs, interacted with visual components, and examined how the platform organised the workshop sequence. The researcher observed where participants proceeded confidently, where they hesitated, and where interface behaviour interrupted the workflow. Informal comments made during the session were also useful for identifying immediate sources of confusion, especially around navigation buttons, required fields, and the progression from one workshop stage to the next.

Overall, participants understood the general purpose of the platform and recognised the value of having workshop activities organised within a single environment. The strongest positive reaction concerned the possibility of keeping exercises, outputs, and project information in one place instead of distributing them across separate tools. At the same time, participants identified several points where the prototype required improvement, particularly in relation to interface consistency, validation messages, button behaviour, clearer guidance, and the completeness of some later-stage functions.

Observed Navigation and Usability Patterns

Several recurring usability patterns emerged during the validation session. First, the project-based structure was generally understandable. Participants were able to recognise the dashboard as the central access point for the workshop sequence and understood that each workshop belonged to the same project workspace. This confirmed the value of organising the platform around a shared project container rather than separate independent pages.

Second, participants responded positively to the idea of an all-in-one platform. The possibility of moving between exercises, collecting outputs, and maintaining project continuity within one environment was repeatedly identified as a major advantage. This

observation directly supports the thesis objective of reducing digital fragmentation in the course workflow.

Third, difficulties appeared when participants reached more complex or methodologically dense modules. Workshop 2, especially the Problem Tree and Objective Tree sections, generated several comments about progression logic, required steps, button behaviour, and the need for more explicit guidance. This suggests that complex analytical activities require stronger scaffolding than simpler dashboard or overview pages.

Fourth, participants identified several technical and interface issues that did not undermine the overall platform concept but affected smooth use. These included non-functional navigation buttons, unclear validation behaviour when required fields were empty, inconsistent navigation controls, uneven layout behaviour, and missing or incomplete export/reporting functions.

Finally, some comments pointed beyond immediate bug fixing toward future feature development. Participants suggested stronger collaboration support, live auto-save or synchronisation, clearer tutorial guidance, side notes or comments, and more complete reporting outputs. These suggestions are useful because they show how users imagined the platform evolving from a proof-of-concept prototype toward a more complete collaborative learning environment.

Summary of Real-Time Observations

The real-time observations collected during the session were organised into four categories: navigation failures, form validation gaps, interface inconsistencies, and completeness issues. Navigation failures refer to buttons or routing elements that did not execute the expected action. Form validation gaps refer to cases where the platform did not clearly guide the user when mandatory information was missing. Interface inconsistencies refer to elements that behaved differently across modules or created visual imbalance. Completeness issues refer to missing or incomplete outputs, export functions, or workflow elements that participants expected to find in a complete platform.

Table 15. Summary of real-time usability observations by prototype area

Prototype area	Observation category	Observed issue
Workshop 1 Idea Canvas	Form validation gap	Saving without required fields filled produced no clear error message or confirmation.
Workshop 2.1 Stakeholder Analysis	Interface inconsistency	Dropdown menus for stakeholder attributes did not always close automatically after selection.
Workshop 2.2 Problem Tree	Navigation / validation issue	The “Add to Analyse” action did not respond clearly when no main problem was selected, and no explanatory error message was shown.
Workshop 2.2 Problem Tree	Interface / output inconsistency	Causes and effects were not always displayed clearly in the generated diagram, and the export/download function required improvement.
Workshop 2.3 Objective Tree	Navigation failure	The “Next Workshop” button did not execute the intended routing action.
Workshop 2.3 Objective Tree	Progression clarity	Participants needed clearer guidance on the step-by-step progression from problem statements to objective structures.
Workshop 3.1 Indicator Selection	Interface clarity	Some criterion/category labels required clearer visual differentiation.
Workshop 3.2 Indicator Ranking / SRF	Form validation gap	Errors on the ranking page were shown mainly through red field outlines, without sufficient explanatory text.
Workshop 3.2 Indicator Ranking / SRF	Method support	Participants suggested that the ranking logic and card-positioning process should remain more visible during data entry.
Workshop 4.1 Indicator Analysis	Interface inconsistency	Image upload boxes could resize unevenly depending on adjacent text length.
Workshop 4.2 SWOT Analysis	Form validation / progression issue	Empty SWOT entries could be saved without clear error feedback, and continuation after completion needed clearer guidance.
Workshop 5 Scenario Building / Q-Methodology	Form validation gap	Empty-field errors appeared too late in the process rather than directly on the relevant page.
Workshop 5 Scenario Building / Q-Methodology	Interface inconsistency	Some action buttons in the correlation or scenario-result areas displayed with overlapping or inconsistent layout.

Workshop 5 Scenario Building / Q- Methodology	Output completeness	The scenario results area required clearer navigation back to the dashboard and more complete export/reporting templates.
General platform	Navigation failure	Session timeout could produce a “Permission denied” message, creating a risk of confusion or unsaved data loss.
General platform	Interface consistency	Navigation controls such as Next, Previous, Save, and Dashboard were not always visually or behaviourally consistent across modules.
General platform	Collaboration support	Participants suggested additional collaborative features, such as notes, comments, task-management support, or live saving.

Source: Authors.

The observations show that the main concept of the platform was understandable, but that the prototype still required targeted refinement at the interaction level. The most important issues were not related to the overall Moodle–Django platform strategy or to the idea of a project-based workspace. Instead, they concerned the practical reliability of navigation, the clarity of validation messages, the consistency of interface components, and the completeness of reporting and continuation functions.

These findings are important because they connect validation directly to fine-tuning. The session did not indicate that the platform needed to be redesigned from the beginning. Rather, it showed that the prototype required targeted corrections to support the same educational logic more reliably: smoother transitions between modules, clearer user feedback, more consistent controls, and better guidance in complex workshop stages.

5.4 Survey Responses and Thematic Analysis

After the hands-on validation session, participants completed the Google Form survey described in Section 5.2. The survey produced seven responses in total: six from former students of the Decision-Making course and one from the course professor, who contributed as an expert academic observer. For this reason, the results are interpreted as a combined formative evaluation rather than as a statistically generalisable user study. The purpose of the survey was to identify how the prototype was perceived in relation to the validation criteria and to collect concrete suggestions for fine-tuning.

The survey combined ten Likert-scale statements with four open-ended questions. The Likert-scale responses provided a structured overview of perceived performance across

usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. The open-ended responses provided more detailed qualitative material, including comments on advantages, disadvantages, bugs, confusing screens, missing features, and possible improvements.

Overall, the numerical results indicate a positive reception of the prototype. All mean scores were equal to or above 4.00 on a five-point scale. This suggests that participants generally perceived the platform as useful, understandable, and aligned with the needs of the course. However, the results also show that the prototype required refinement in specific areas, especially visual feedback, data-flow confidence, and independent use without additional support.

The highest-scoring statements concerned the platform's ability to make the connection between early workshop decisions and final scenario results visible, and the smoothness of interactive features such as drag-and-drop and chart loading. Both received a mean score of 4.71. These results are important because they directly support two central goals of the thesis: improving output traceability and making interactive workshop activities easier to manage within a single project-based environment.

The lowest-scoring statements concerned visual feedback and workflow continuity, data flow between workshops, and the ability to use the platform independently without much technical help. Each received a mean score of 4.00. Although these scores remain positive, they identify areas where the prototype required fine-tuning. In particular, they suggest the need for clearer progress feedback, stronger confirmation messages, more visible save behaviour, and more explicit guidance in complex workshop stages.

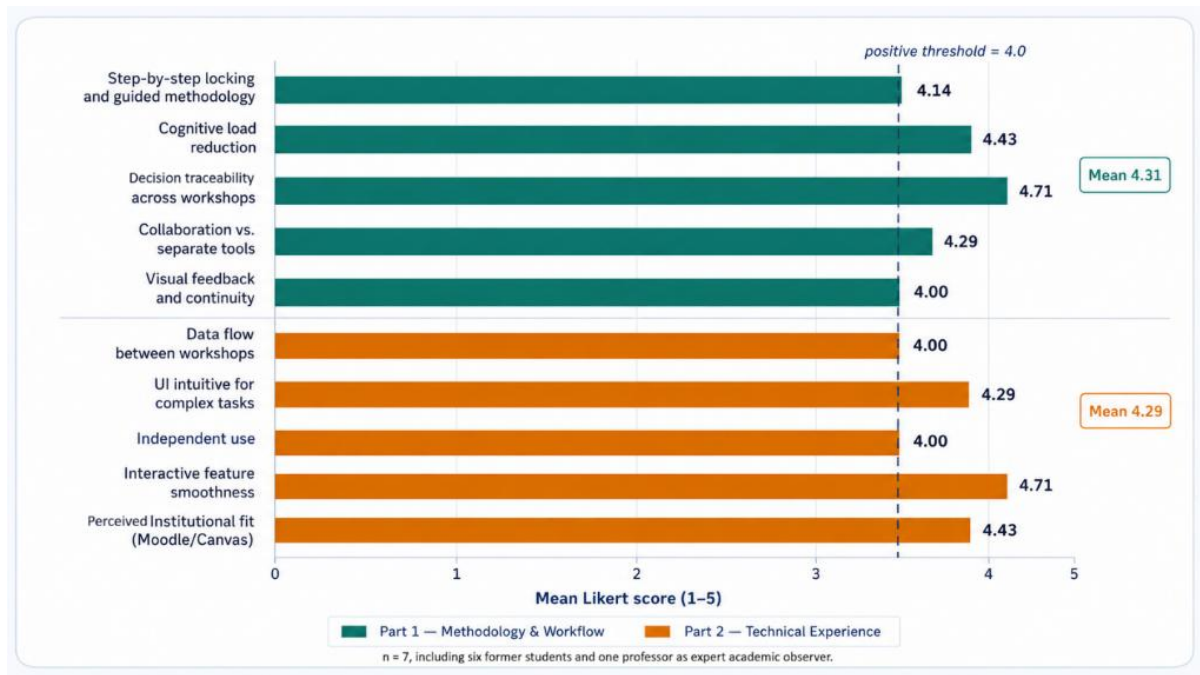


Figure 33. Mean Likert scores per survey statement, based on seven survey responses
Source: Authors.

Table 16. Descriptive statistics for Likert-scale survey statements, $n = 7$

Statement	Min	Max	Mean	Std. Dev.	Criterion
Step-by-step locking mechanism	3	5	4.14	0.69	Usability / Clarity
Cognitive load reduction	4	5	4.43	0.53	Usability / Workflow coherence
Decision tracking across workshops	4	5	4.71	0.49	Workflow coherence
Collaboration compared with separate tools	2	5	4.29	1.25	Perceived usefulness
Visual feedback and continuity	2	5	4.00	1.15	Clarity / Workflow coherence
Data flow between workshops	3	5	4.00	0.82	Workflow coherence / Completeness
UI intuitiveness for complex tasks	3	5	4.29	0.76	Usability / Clarity
Independent use	2	5	4.00	1.00	Accessibility
Interactive feature smoothness	4	5	4.71	0.49	Usability
Perceived institutional fit with Moodle/Canvas	3	5	4.43	0.79	Perceived usefulness / Completeness

Source: Authors.

The open-ended responses were analysed using the same six validation criteria introduced in Section 5.2: usability, clarity, accessibility, workflow coherence, perceived usefulness, and completeness. This avoids introducing a separate classification system and keeps the qualitative interpretation aligned with the survey instrument. Comments on navigation problems, button behaviour, and task completion were coded under usability. Comments on instructions, validation messages, and progression cues were coded under clarity. Comments on legibility and independent use were coded under accessibility. Comments on data flow, stage continuity, and traceability were coded under workflow coherence. Comments on the educational value of the all-in-one workspace, including its value for group work, were coded under perceived usefulness. Comments on missing features, reporting functions, advanced collaboration tools, and future development needs were coded under completeness.

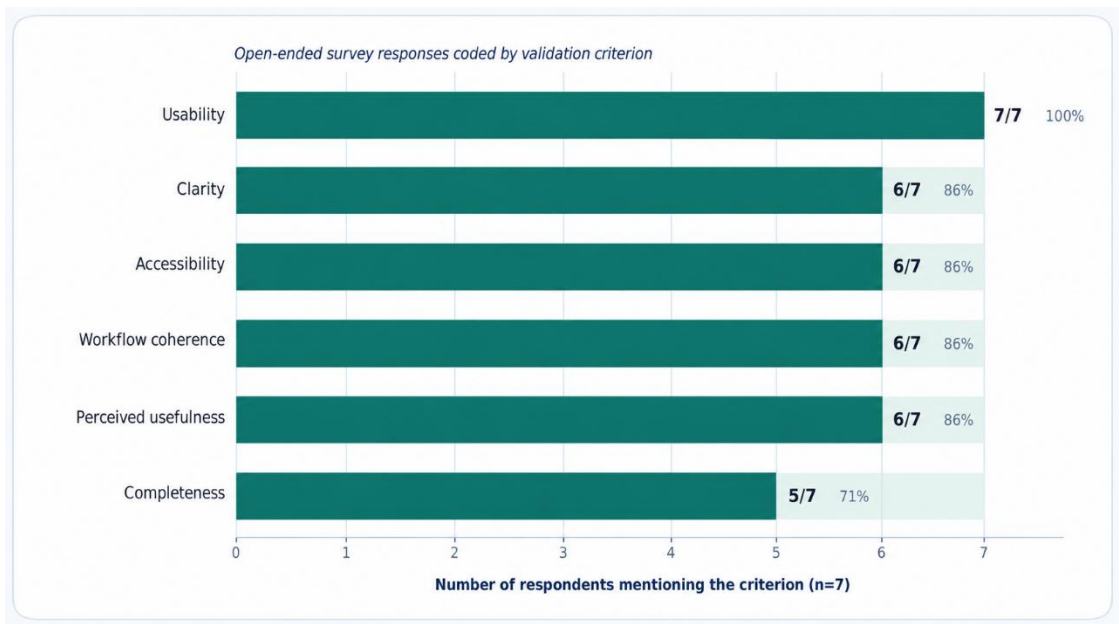


Figure 34. Open-ended survey responses coded by validation criterion.

Source: Authors.

Usability

Usability was one of the most frequently discussed criteria in the open-ended responses. The issues identified by participants concerned non-functional navigation buttons, unclear save behaviour, insufficient form validation feedback, and inconsistent navigation elements across modules. The most recurrent individual issue concerned the failure of the “Next

Workshop” button in some Workshop 2 sections, especially around the Objective Tree stage. This issue was important because participants generally understood what the button was supposed to do, but the interface did not always execute the expected action.

Participants also reported that navigation controls such as Next, Previous, Save, and Dashboard should behave consistently across all workshop modules. These observations show that the platform’s overall structure was understandable, but that its interaction grammar needed refinement. Usability was therefore supported at the level of the general platform concept and main workflow structure, but required improvement at the level of navigation reliability, button behaviour, save confirmation, and task completion feedback.

Clarity

Clarity concerned whether participants understood what to do, how to proceed, and how each workshop activity related to the overall decision-making methodology. The validation showed that participants generally understood the project-based structure of the platform, but needed stronger guidance in methodologically complex modules. This was especially relevant in the Problem Tree, Objective Tree, SRF Ranking, and Q-Methodology sections, where users needed to understand both the interface action and the underlying methodological logic.

Participants reported that some required fields could be left empty without receiving a clear explanatory message. In these cases, the platform either redirected the user or failed silently, creating uncertainty about whether the action had been completed correctly. Participants also suggested clearer instructions, examples before complex tasks, more visible progression cues, and explanatory validation messages. These findings show that clarity in the prototype depends not only on visual layout, but also on pedagogical scaffolding. The platform therefore required clearer guidance, more explicit error messages, and stronger step-by-step support in complex workshop stages.

Accessibility

Accessibility was evaluated through both numerical scores and open-ended comments. The statement concerning independent use received a positive but relatively lower mean score compared with other items. This suggests that most participants could use the platform, but that some users still needed additional support, especially in more complex workshop modules.

The open-ended comments show that accessibility was not only a question of visual design, but also of instructional support. Participants suggested adding examples in numerical input fields, tutorial demos before each workshop step, clearer explanations of complex methods, and stronger progression guidance. They also noted that long text sometimes did not fit well inside cards, boxes, or diagram shapes. This affected readability in modules such as tree-building and ranking, where users may enter longer labels or descriptions. These findings indicate that the platform should continue improving its pedagogical scaffolding, responsive text handling, and visual hierarchy in order to support users with different levels of technical confidence.

Workflow coherence

Workflow coherence was one of the strongest dimensions of the validation. The Likert-scale statement concerning decision traceability across workshops received one of the highest mean scores, indicating that participants perceived the platform as effective in showing how early decisions could influence later outputs. This finding directly supports the thesis objective of improving workflow continuity and output traceability in a workshop-based course.

The qualitative responses reinforced this result. Participants described the platform as useful because it created a single procedure for the project, made it easier to move back and forth between exercises, and helped collect outputs in one place. This is significant because the participants had previous experience with the course's fragmented workflow and were therefore able to compare the prototype with the earlier use of separate tools. However, the results also show that data continuity still required clearer interface feedback. Some participants wanted stronger confirmation that information had been saved correctly and transferred between stages. Therefore, the validation confirms the value of the project-based workspace, while also identifying the need for clearer data-flow signals and save confirmations.

Perceived usefulness

Perceived usefulness was strongly confirmed by both the Likert-scale results and the open-ended responses. Participants recognised the platform's value in reducing cognitive load, making the workflow more interactive, and keeping project outputs accessible in one place. Several comments described the platform positively because it made the course process easier to follow and reduced the burden of managing multiple disconnected tools.

Participants also recognised the collaborative value of having the workshop sequence inside one platform. The all-in-one structure was perceived as useful because it could reduce the need for groups to coordinate work across separate files, visual boards, spreadsheets, and communication channels. In this sense, collaboration is interpreted here as part of the platform's perceived usefulness: participants valued the possibility of organising shared project work within one coherent environment. This finding is important because it shows that participants did not evaluate the prototype only as a technical interface. They understood it as an educational environment connected to the structure of the Decision-Making course.

The perceived usefulness of the platform therefore lies in its ability to make the workshop process more coherent, visible, collaborative, and manageable. At the same time, one possible limitation of integrated platforms is that they may reduce some flexibility or personalisation compared with separate specialised tools. This point does not weaken the thesis argument, but it should be acknowledged in the discussion as a trade-off between integration and tool-specific flexibility.

Completeness

Completeness was the criterion that most clearly connected validation findings to later fine-tuning. Participants identified missing or incomplete elements in the tested version of the prototype, especially in relation to later-stage scenario analysis, external decision-support tools, export formats, and final reporting. Suggestions included adding or improving the performance matrix for scenario analysis, including links to external tools such as AHP / Super Decisions, PROMETHEE, and iiSBE / Sustainable MED Cities, improving PDF templates, and offering raw export formats such as CSV or Word.

Participants also suggested additional features for group coordination, such as side notes, comments, task-management support, live auto-save, and real-time synchronisation. These suggestions should not be treated as a separate validation theme, but as completeness and future-development issues. They show that participants understood the platform as a shared project workspace, but expected stronger tools for communication and coordination within that workspace. In this sense, the validation confirms the relevance of structured collaboration as a design goal, while also showing that the current prototype supports collaboration mainly through shared project continuity rather than through advanced synchronous or asynchronous collaboration features.

The correct interpretation is therefore that validation feedback helped clarify which later-stage functions, reporting outputs, and collaboration-related features needed to be completed, strengthened, or better represented in the final prototype. In this way, completeness became a direct input for the fine-tuning phase. It helped transform the prototype from a mainly early-to-mid workflow demonstrator into a more complete proof-of-concept covering the full sequence from project creation to Project Review.

Synthesis of the validation criteria

The criterion-based analysis shows a coherent pattern. Participants broadly validated the main design logic of the platform: an integrated project-based workspace can reduce fragmentation, improve workflow coherence, support decision traceability, and make the course process easier to follow. The strongest findings relate to the educational value of integration, the visibility of connections between workshop stages, and the usefulness of having outputs collected in one environment.

At the same time, the validation identified several implementation-level weaknesses. These concerned navigation reliability, validation messages, interface consistency, guidance in complex modules, collaboration-related future features, and the completeness of later-stage outputs and reporting functions. These weaknesses do not challenge the overall Moodle–Django platform strategy. Rather, they show that the prototype required targeted refinement in order to better realise that strategy in practice.

The cross-criterion synthesis therefore supports the role of Chapter 5 within the Design Thinking process. The Test stage confirmed the relevance of the platform concept, while the Iterate stage translated user feedback into specific fine-tuning priorities. The next section evaluates these findings against the six validation criteria before Section 5.6 explains the prototype refinements implemented in response.

5.5 Evaluation Against the Validation Criteria

Because Section 5.4 has already discussed the survey findings according to the six validation criteria, this section does not introduce a new set of themes or repeat the detailed qualitative analysis. Instead, it synthesises the same evidence into a compact criterion-by-criterion evaluation of the prototype. The purpose is to clarify, for each criterion, what was confirmed as a strength and what required refinement before the prototype could be considered a stable proof of concept.

Overall, the validation indicates that the platform was positively received and that its core design logic was understood by participants. The main strengths concerned the integrated project-based workspace, the visibility of connections between workshop stages, the perceived usefulness of reducing work across disconnected tools, and the possibility of making group outputs more reviewable within one environment. The main weaknesses concerned implementation-level issues, including inconsistent navigation controls, insufficient validation messages, incomplete or unclear reporting functions, the need for stronger guidance in complex workshop modules, and the absence of more advanced collaboration features.

Table 17. Criterion-by-criterion evaluation of the prototype

Validation criterion	Main validation finding	Confirmed strength	Required refinement
Usability	Users understood the general navigation logic and interacted with the main modules.	Dashboard-based access and interactive components were broadly usable.	Fix inconsistent buttons, unclear save behaviour, and validation gaps.
Clarity	The overall workflow was understandable, but complex methods required more guidance.	The platform made the workshop sequence more legible than separate tools.	Add clearer instructions, examples, progression cues, and explanatory error messages.
Accessibility	Most users could interact with the prototype, but independent use varied.	Web-based access and centralised structure reduced tool-switching friction.	Improve text fitting, examples, visual hierarchy, and tutorial-style guidance.
Workflow coherence	Participants valued the connected project-based workspace.	The platform supported continuity between early decisions and later outputs.	Strengthen save confirmation, progress indicators, and data-continuity signals.
Perceived usefulness	Participants recognised the educational value of the all-in-one platform.	The platform reduced perceived fragmentation and cognitive management overhead.	Maintain flexibility and avoid over-rigid workflows in future versions.
Completeness	The prototype was meaningful as a proof of concept, but some later-stage functions needed completion.	The platform demonstrated the feasibility of the proposed workflow structure.	Improve scenario analysis, external tool access, CBA, Project Review, reporting, and export options.

Source: Authors.

The criterion-by-criterion evaluation confirms that the prototype's main contribution is not limited to individual features. Its value lies in the reorganisation of the course workflow into a more coherent project-based digital environment. The validation supports the thesis argument that a custom interactive workshop layer can complement the institutional LMS layer by addressing workshop-specific needs that are difficult to manage through general-purpose tools alone.

At the same time, the evaluation shows that the prototype requires refinement before it can be considered ready for broader use. The most urgent refinements concern navigation reliability, clearer validation feedback, stronger instructional scaffolding, improved interface consistency, later-stage reporting and export functions, and more advanced collaboration support. These findings directly inform the fine-tuning process discussed in the next section.

5.6 Fine-Tuning and Prototype Refinement

The fine-tuning phase corresponds to the Iterate stage of the Design Thinking process. Its purpose was not to redesign the platform from the beginning, but to improve the prototype in response to the issues identified during validation. The validation findings showed that the main platform concept was understood and positively received, while several implementation-level aspects required refinement. The fine-tuning process therefore focused on navigation reliability, form validation, interface consistency, clearer guidance, workshop progression, reporting functions, and the completion of later-stage modules within the proof-of-concept scope.

The refinements were prioritised according to their impact on the educational workflow. Issues that interrupted progression between modules, created uncertainty about saved data, or reduced the clarity of complex workshop tasks were treated as higher priority. Suggestions concerning broader collaboration features, such as live synchronisation, task management, or advanced commenting, were considered important future development directions but were not all treated as core refinements within the current thesis scope.

Technical and Interface Refinements

The first group of refinements concerned technical reliability and interface consistency. Several navigation buttons and routing elements were reviewed to reduce the risk of users becoming blocked between workshop stages. Particular attention was given to transitions between connected workshop pages, where participants had reported unclear or non-

functional “Next Workshop” behaviour. Navigation controls such as Save, Next, Previous, and Dashboard were also reviewed to make their placement and function more consistent across modules.

Form validation was another important area of refinement. During validation, participants reported that some pages allowed empty or incomplete submissions without sufficient explanation. The fine-tuning process therefore focused on making required fields clearer, improving error messages, and providing more explicit feedback when data was missing or incorrectly entered. This was especially relevant in methodologically structured modules such as Idea Canvas, Problem Tree, Objective Tree, SRF Ranking, SWOT Analysis, and Q-Methodology.

Interface consistency was also improved through adjustments to layout, visual hierarchy, and component behaviour. The validation showed that some text labels, cards, upload boxes, and action buttons did not always adapt well to different content lengths. Fine-tuning therefore included adjustments to visual balance, text fitting, card readability, and the behaviour of interactive components. These changes were intended to improve accessibility and reduce unnecessary confusion during task completion.

Workshop-Specific Refinements Across the Prototype

In addition to general technical and interface corrections, the fine-tuning phase also addressed workshop-specific issues that emerged during validation. These refinements were not limited to one module. They concerned the clarity, usability, progression logic, and completeness of several workshops across the platform.

Workshop 1 Idea Canvas required improvements in form validation, save feedback, and visual consistency. Participants needed clearer confirmation when information was saved and stronger guidance when required fields were incomplete. These refinements were important because the Idea Canvas functions as the entry point for the project workflow and establishes the first structured description of the group’s case.

Workshop 2 required more substantial structural refinement because it includes several connected analytical steps: Stakeholder Analysis, Problem Tree, and Objective Tree. Validation feedback showed that participants understood the educational importance of these activities, but needed clearer guidance through their sequence. The Problem Tree module was refined to make the progression from core problem to causes and effects more explicit. The Objective Tree module was also clarified so that users could more easily understand the

transition from problem statements to desired objectives. These refinements were important because Workshop 2 acts as a bridge between initial project framing and later analytical stages.

Workshop 3 required refinements related to indicator selection and SRF ranking. In the Indicator Selection module, some labels and categories needed clearer visual differentiation. In the SRF Ranking module, participants needed more explicit error messages, clearer guidance for card positioning, and better visual handling of long indicator names. These changes supported the transition from qualitative project understanding to structured indicator prioritisation.

Workshop 4 required interface and progression improvements. In the Indicator Analysis module, layout issues affected the consistency of image upload areas. In the SWOT Analysis module, participants expected clearer validation feedback when empty entries were saved and a clearer continuation path after completing the activity. These refinements improved the usability of the strategic analysis stage.

Workshop 5 required refinements in the Q-Methodology and scenario-building workflow. Participants identified issues related to validation timing, action-button layout, colour logic, and reporting outputs. The fine-tuning process therefore focused on making the Q-sorting and scenario-result stages clearer, more consistent, and easier to navigate.

Later-stage modules were also strengthened in response to validation feedback. Workshop 6 Scenario Analysis was refined as the module responsible for comparing scenarios using selected indicators and performance values. Workshop 7 External Decision-Support Tools was positioned as a structured access point for tools selected by the professor, such as AHP / Super Decisions, PROMETHEE, and iiSBE / Sustainable MED Cities. Workshop 8 Cost–Benefit Analysis / CBA was included to translate the spreadsheet-based CBA exercise into a guided digital workflow. Finally, the Project Review page was refined as a synthesis space that aggregates outputs from the previous modules and supports final review of the project development.

These workshop-specific refinements show that validation did not lead to isolated bug fixing only. It supported a broader improvement of the prototype across the full project workflow, from the first project-framing activity to later scenario analysis, cost–benefit evaluation, and Project Review.

Reporting and Export Refinements

Participants also identified reporting and export functions as areas requiring improvement. In a fragmented workflow, final outputs are often disconnected from the intermediate reasoning that produced them. For this reason, export and reporting functions are important not only as technical conveniences, but as mechanisms for output traceability. The fine-tuning process therefore paid attention to improving the clarity and usefulness of exported outputs, including visual downloads, structured summaries, and final project review materials.

The Project Review function is especially relevant in this respect. It does not replace the analytical modules, but gathers their main outputs into a review-oriented structure. This supports students in preparing their final synthesis and supports professors or teaching assistants in understanding the development of the group's reasoning across the project workflow.

Table 18. Fine-tuning interventions: issue, revision, and validation trigger

Prototype area	Validation issue	Fine-tuning intervention	Triggering feedback or observation
General navigation	Inconsistent or unclear navigation between modules	Review and standardisation of Save, Next, Previous, and Dashboard controls	Participants reported unclear or non-functional navigation buttons.
General data entry	Missing or unclear validation messages	Addition or improvement of required-field feedback and explanatory error messages	Participants reported that empty fields could be submitted without clear feedback.
General save behaviour	Users were sometimes unsure whether data had been saved correctly	Stronger confirmation and clearer response after saving data	Participants requested clearer save feedback and data-flow confirmation.
General interface	Visual imbalance, text overflow, and inconsistent component behaviour	Layout, card, and text-fitting adjustments	Participants reported that long text did not always fit inside shapes, cards, or visual elements.
Workshop 1 Idea Canvas	Required fields and save behaviour were not always clear	Improved form validation, save feedback, and visual consistency	Participants needed clearer confirmation and guidance in the first project-framing stage.
Workshop 2.1 Stakeholder Analysis	Dropdowns and stakeholder interface elements required smoother behaviour	Refinement of stakeholder input behaviour and visual feedback	Participants observed interface inconsistencies in stakeholder attribute selection.

Workshop 2.2 Problem Tree	Progression from core problem to causes and effects was not sufficiently guided	Clearer step-by-step logic for defining the main problem before adding causes and effects	Participants needed clearer guidance before adding analytical elements.
Workshop 2.3 Objective Tree	Transition from problem statements to objectives required clearer guidance	Clearer relationship between problem structures and objective formulation	Participants needed stronger support in moving from diagnosis to objectives.
Workshop 3.1 Indicator Selection	Labels and categories needed clearer differentiation	Improvement of category/criterion visibility and interface clarity	Participants reported that some labels required stronger visual distinction.
Workshop 3.2 Indicator Ranking / SRF	Ranking errors and card-positioning logic were not always clear	More explicit error messages and clearer ranking guidance	Participants requested stronger support during SRF card ranking.
Workshop 4.1 Indicator Analysis	Image upload and layout areas were visually inconsistent	Refinement of upload-area layout and visual balance	Participants observed uneven resizing of upload boxes.
Workshop 4.2 SWOT Analysis	Empty entries and continuation after completion required clearer feedback	Improved validation feedback and clearer continuation path	Participants expected clearer error messages and next-step guidance.
Workshop 5 Scenario Building	Q-sorting, correlation outputs, and scenario results needed clearer layout and feedback	Refinement of validation timing, button layout, colour logic, and reporting outputs	Participants identified layout and output issues in the scenario-building workflow.
Workshop 6 Scenario Analysis	Later-stage scenario comparison needed to be represented more clearly	Strengthening of performance-matrix and scenario-comparison logic	Participants identified scenario analysis as a necessary later-stage function.
Workshop 7 External Decision-Support Tools	External tools were expected within the platform workflow	Structured access to AHP / Super Decisions, PROMETHEE, and iiSBE / Sustainable MED Cities	Participants suggested adding relevant external decision-support tool links.
Workshop 8 Cost–Benefit Analysis / CBA	Spreadsheet-based CBA needed a clearer digital workflow	Inclusion of guided CBA structure for costs, benefits, assumptions, and results	Participants expected later-stage evaluation functions to be included in the platform.
Project Review	Final synthesis and reporting needed improvement	Refinement of Project Review as a synthesis page aggregating previous outputs	Participants requested clearer final reporting and reviewable outputs.

Source: Authors.

Table 19. Fine-tuning outcomes mapped to validation criteria

Validation criterion	Fine-tuning outcome
Usability	Navigation and interaction issues were prioritised to reduce friction during task completion.
Clarity	Instructions, progression cues, and error messages were improved to make complex methods easier to follow.
Accessibility	Text readability, layout balance, and guidance were refined to support users with different levels of technical confidence.
Workflow coherence	Stage-to-stage progression and the connection between modules were strengthened.
Perceived usefulness	The all-in-one project workspace was preserved and extended to better support the full course workflow.
Completeness	Later-stage modules, reporting functions, and Project Review were strengthened within the proof-of-concept scope.

Source: Authors.

The fine-tuning process confirmed the iterative nature of the thesis methodology. Validation did not function only as a final assessment of the prototype; it produced practical knowledge about how the platform should be improved. The feedback showed that the Moodle–Django platform strategy was meaningful for the course context, while also revealing the practical conditions required for such a platform to work effectively: reliable navigation, explicit feedback, readable interfaces, guided complex methods, and complete reviewable outputs.

The resulting prototype should therefore be understood as a refined proof of concept. It does not represent a fully deployed institutional system, and it does not claim full Moodle/LTI 1.3 integration. However, it demonstrates how a Django-based custom interactive workshop layer can complement the institutional LMS layer by supporting a connected, project-based, traceable, and more coherent workshop workflow.

Chapter 6. Discussion

This chapter discusses the findings of the thesis in relation to the research questions, the theoretical and conceptual foundation, the prototype development process, and the validation results. While Chapter 5 presented the validation and fine-tuning phase in detail, the purpose of this chapter is interpretive. It explains what the design decisions, implementation outcomes, validation findings, and prototype refinements reveal about the broader problem addressed in the thesis: digital fragmentation in collaborative, multi-stage, workshop-based higher education.

The discussion is grounded in the full research progression developed across the previous chapters. Chapter 1 defined the central research problem as the fragmentation of digital tools supporting collaborative academic work and introduced the aim of designing, developing, and evaluating an integrated interactive platform for a workshop-based course at Politecnico di Torino. Chapter 2 established Design Thinking as the methodological framework, organising the research into the Empathise, Define, Ideate, Prototype, Test, and Iterate stages. Chapter 3 developed the theoretical and conceptual foundation by analysing the target course, user roles, current digital practices, digital fragmentation, platform requirements, technological pathways, and technical objectives. Chapter 4 documented the development of the Django-based prototype as a custom interactive workshop layer designed to complement the institutional LMS layer. Chapter 5 then evaluated the prototype through formative validation and explained how participant feedback informed targeted fine-tuning.

The discussion therefore does not repeat the descriptive content of the previous chapters. Instead, it interprets the results in relation to existing literature and uses them to clarify the contribution of the thesis. The chapter addresses the two research questions within the discussion rather than treating them as separate descriptive sections. The first research question concerns how an interactive educational platform can conceptually support a complete, multi-stage collaborative workshop workflow in higher education. The second research question concerns how such a platform should be technically designed and developed within an institutional university environment.

A central argument of this chapter is that the value of the proposed platform does not lie only in digitising individual workshop activities. Its main contribution is the reorganisation of the workshop process into a coherent project-based workspace, where outputs from earlier stages can remain visible, reusable, and connected to later analytical and evaluative tasks.

This responds directly to the fragmentation problem identified in the thesis, where students and instructors previously had to coordinate work across disconnected tools, files, visual boards, spreadsheets, communication channels, and external decision-support environments.

At the same time, the discussion remains careful about the limits of the work. The prototype should be understood as a refined proof of concept rather than a fully deployed institutional system. It demonstrates the educational and technical plausibility of the Moodle–Django platform strategy, but it does not claim production-level deployment, institutional scalability testing, or full Moodle integration through LTI 1.3. These remain future development directions. For this reason, the discussion interprets the findings as evidence of conceptual validity, prototype feasibility, and formative educational usefulness, rather than as proof of complete institutional readiness.

The chapter is organised into three main parts. Section 6.1 discusses the findings in relation to the research questions and compares them with existing literature on learning management systems, project-based learning, usability, accessibility, and educational technology design. Section 6.2 presents the theoretical contribution of the thesis. Section 6.3 discusses the practical implications of the work for the target course, Politecnico di Torino, and similar workshop-based higher education contexts.

6.1 Discussion of Findings

The findings of this thesis can be positioned in relation to three main areas of literature: learning management systems, project-based and workshop-based learning, and educational technology design. Existing literature recognises the importance of Learning Management Systems as institutional infrastructures for organising materials, assignments, communication, and formal course administration (Petchamé et al., 2023). However, the findings of this thesis show that the institutional strength of an LMS does not automatically solve the needs of complex workshop-based learning. In the target Decision-Making course, the main difficulty was not the absence of a digital course space, but the absence of a connected environment able to preserve intermediate outputs, support interactive methods, and maintain continuity across several decision-making stages.

This finding is important because it reframes digital fragmentation as more than a technical inconvenience. In project-based and workshop-based learning, fragmented digital support can affect the educational process itself. These learning formats depend on the progressive development of a project through connected stages of problem framing, stakeholder analysis,

objective formulation, indicator selection, scenario development, evaluation, and final synthesis. If these stages are distributed across disconnected tools, students must repeatedly reconstruct the logic of the project, manually transfer information, and manage different file formats and versions. This increases cognitive load and weakens the continuity of the learning process, which is consistent with literature emphasising the relationship between instructional design, cognitive load, and learning efficiency (Kirschner, Paas and Kirschner, 2009).

In relation to the first research question, the findings show that an interactive educational platform can conceptually support a multi-stage collaborative workshop workflow when it is designed as a connected project-based environment rather than as a collection of isolated pages. The prototype supports workflow continuity by placing the workshop modules inside a shared project workspace. This structure allows students to move through the course methodology as a cumulative process, rather than treating each workshop as a separate exercise. In this sense, the platform contributes to project-based learning by making the relationship between stages more visible and easier to follow.

The findings also show that data continuity and output traceability are central to the educational value of the platform. The prototype does not only provide access to different workshop pages; it also preserves key outputs produced during the workflow. Stakeholder information, problem and objective structures, selected indicators, ranking results, scenario data, performance values, CBA inputs, and final review outputs can remain connected to the same project. This helps students reuse earlier work and helps professors or teaching assistants understand how later conclusions relate to earlier decisions. Compared with fragmented workflows, where final deliverables may become separated from the reasoning process that produced them, the prototype demonstrates how project-based data persistence can strengthen traceability.

This contribution differs from the use of separate tools such as visual boards, spreadsheets, shared files, or external decision-support applications. Such tools can be valuable within their own functional domains, but they do not automatically preserve the continuity between stages. A spreadsheet may support calculations, a visual board may support brainstorming, and an LMS may support submission, but the connection between these artefacts often remains manual. The innovation of the prototype is therefore not the creation of another isolated tool, but the design of a connected workshop layer in which methodological steps, data, visual outputs, and review functions are organised around the same project workspace.

The validation results support this interpretation. Participants recognised the value of having exercises, outputs, and project information in one place. The positive response to decision traceability across workshops indicates that users perceived the platform as a connected workflow rather than as a set of disconnected screens. This finding is consistent with the thesis objective of reducing fragmentation, but it also extends the literature by showing how the problem of fragmentation can be addressed through workflow design, not only through tool selection.

The findings also relate to usability and technology acceptance literature. Usability evaluation focuses on whether users can understand an interface, navigate it, and complete tasks effectively and efficiently (Nielsen, 1994; ISO, 2018). Technology acceptance literature also highlights perceived usefulness as a central factor in whether users consider a system valuable for task performance (Davis, 1989). In the validation, participants did not evaluate the platform only as a technical interface. They evaluated it as an educational workflow environment. This means that perceived usefulness in this case depended on whether the platform helped them follow the course process, reduce tool-switching, preserve outputs, and understand the connection between stages. The thesis therefore shows that usefulness in workshop-based educational technology is not only a matter of convenience, but also a matter of pedagogical continuity.

Accessibility and clarity also emerged as important dimensions of the findings. Accessibility is often associated with web standards, interface legibility, and inclusive access conditions (W3C, 2024). In this thesis, accessibility also has a pedagogical dimension. The validation showed that participants could generally use the platform, but that complex methods required stronger guidance, clearer validation messages, examples, and progression cues. This suggests that an educational platform must support not only access to the interface, but also access to the logic of the method. In modules such as Problem Tree, Objective Tree, SRF Ranking, and Q-Methodology, usability and accessibility depend on how clearly the platform communicates the methodological sequence.

Collaboration was supported by the prototype, but in a limited and realistic way. The current platform supports collaboration mainly through shared project continuity and preserved group outputs. This is important because collaborative learning requires shared artefacts around which students can discuss, negotiate, and make decisions. However, the validation also showed that users expected richer collaboration mechanisms, such as comments, side notes, task-management support, live auto-save, and real-time synchronisation. Therefore,

the platform conceptually improves collaborative organisation, but it does not yet provide a complete collaborative work environment. This distinction is important because it prevents overclaiming and clarifies the difference between the current proof of concept and a future production-level platform.

In relation to the second research question, the findings show that the most appropriate technical strategy is a layered platform architecture. The institutional LMS should remain responsible for course governance, access to materials, announcements, deadlines, formal learning-management functions, and institutional continuity. A custom web framework should provide the specialised interactive workshop layer required by the course. In this thesis, Moodle represents the institutional LMS layer, while Django provides the custom interactive workshop layer.

This layered strategy responds to limitations found in both standalone LMS use and fully independent custom development. A standard LMS provides institutional stability, but it is not designed to implement highly specific workshop modules such as stakeholder matrices, problem and objective trees, SRF ranking boards, Q-sorting interfaces, scenario dashboards, performance matrices, and cost–benefit analysis workflows. A fully standalone custom platform could implement these functions, but it would risk becoming disconnected from the university’s official digital ecosystem. The Moodle–Django strategy addresses this tension by combining institutional stability with custom pedagogical flexibility.

The technical findings also show the importance of separation of responsibilities. Moodle and Django should not duplicate one another. Instead, each layer should perform the functions for which it is most suitable. Moodle should remain the institutional course environment, while Django should manage interactive workshop functionality and project-specific data. This is the main difference between the proposed strategy and a simple accumulation of additional tools. The thesis does not propose more fragmentation. It proposes a structured relationship between layers, where the institutional platform and the custom workshop layer have distinct but complementary roles.

Django proved suitable for the proof-of-concept because the platform required a structured backend, reusable views, database models, URL routing, template rendering, form handling, authentication support, interactive front-end behaviour, and export functions. The project-based data structure is central to this technical design. Each project functions as the persistent container for the group’s work, and workshop outputs remain connected to the same project

record or related data structures. This supports the conceptual goals of workflow continuity, data continuity, and output traceability through technical implementation.

The role of LTI 1.3 should be interpreted carefully. LTI 1.3 is relevant because it provides a standard way for an external learning tool to connect with an LMS while supporting secure launch, identity transmission, and role information (1EdTech Consortium, 2019). In a future institutional deployment, Moodle could remain the official course environment while launching the Django workshop layer as an integrated external tool. However, the current prototype does not validate LTI 1.3 in a production Moodle environment. It demonstrates integration readiness at the architectural level, not completed institutional integration.

Overall, the findings show that the thesis contributes a specific form of educational technology design: a custom interactive workshop layer that complements the institutional LMS and supports a cumulative decision-making workflow. Compared with existing material on LMS use, usability, accessibility, and educational technology adoption, the innovation of this work lies in connecting these concepts to a real multi-stage workshop process and implementing them through a functional Django prototype. The prototype remains context-specific and proof-of-concept in scope, but it demonstrates how digital fragmentation can be addressed through a layered architecture, project-based persistence, method-specific interaction, and reviewable outputs.

6.2 Theoretical Contribution

The theoretical contribution of this thesis lies in the way it connects educational fragmentation, workshop-based learning, and platform architecture into a single design argument. The thesis does not only describe a technical prototype. It develops a conceptual model for understanding how digital platforms can support cumulative, collaborative, and multi-stage learning processes in higher education.

The first theoretical contribution is the reframing of digital fragmentation as a pedagogical problem. In many discussions of educational technology, fragmentation may appear mainly as a practical inconvenience: students must open several tools, manage files, and move between platforms. This thesis shows that, in workshop-based and project-based courses, fragmentation has a deeper educational effect. It weakens the continuity of reasoning, separates intermediate outputs from final deliverables, reduces visibility for professors and

teaching assistants, and increases the cognitive burden placed on students. In this sense, fragmentation is not only a technical problem of disconnected systems. It is a structural problem that affects how learning processes are organised, followed, and evaluated.

This contribution is especially relevant for project-based and workshop-based learning. These pedagogical formats depend on the connection between stages. Students do not simply complete independent tasks; they progressively build a project through problem framing, stakeholder analysis, objective formulation, indicator selection, ranking, scenario building, evaluation, and final synthesis. If the digital environment does not preserve the relationship between these stages, the educational logic of the course becomes harder to sustain. The thesis therefore contributes to the literature by showing that digital platform design must be evaluated not only by usability or feature availability, but also by its ability to preserve pedagogical continuity.

The second theoretical contribution is the concept of the interactive workshop layer. Conventional LMSs are important institutional infrastructures, but they are often designed around content delivery, course administration, submission management, and communication. They are not always suited to bespoke workshop methods requiring visual interaction, structured data reuse, sequential progression, and project-level traceability. This thesis proposes the interactive workshop layer as a conceptual extension of the LMS: a specialised digital layer that supports active project work while remaining complementary to the institutional learning-management environment.

This concept helps clarify the role of custom platforms in higher education. A custom platform should not necessarily replace the LMS. Instead, it can extend the LMS by supporting activities that require more flexible, method-specific interaction. In this thesis, Django performs this role by supporting stakeholder matrices, problem and objective trees, indicator selection, SRF ranking, Q-Methodology, Start Park scenario building, scenario analysis, external decision-support tool access, Cost–Benefit Analysis, and Project Review. The theoretical implication is that educational platform design can be understood as a layered system: institutional stability and governance can remain in the LMS layer, while pedagogical specificity and interactive workflow support can be developed in a custom layer.

The third contribution is the separation-of-responsibilities model for educational platform architecture. The thesis shows that a layered platform strategy can respond to the limitations of both standalone LMS use and standalone custom development. The LMS layer provides institutional legitimacy, familiar access, role structure, and long-term course governance.

The custom web layer provides the flexibility required for interactive workshop modules and project-based data persistence. The theoretical value of this model is that it avoids treating educational technology choices as mutually exclusive. Instead of choosing between institutional stability and pedagogical flexibility, the thesis shows how the two can be combined through a layered architecture.

The fourth contribution concerns traceability as a design principle. In fragmented workflows, traceability is often reduced because the final submission becomes separated from the process that produced it. This thesis argues that traceability should be designed into the platform structure from the beginning. Project-based data persistence, stored intermediate outputs, stage-to-stage continuity, and final Project Review all contribute to making the development of group reasoning more visible. Although the current prototype does not yet implement full action logging or version history, it demonstrates how traceability can be strengthened by preserving the main analytical artefacts inside one project workspace.

The fifth contribution is methodological. By using Design Thinking as the organising framework, the thesis demonstrates how a platform can be developed as a research artefact. The prototype is not only an implementation outcome; it is also a way of testing assumptions about workflow continuity, accessibility, collaboration, traceability, and institutional fit. The validation and fine-tuning phase shows that user feedback can transform the prototype from a technical solution into an evaluated and refined educational design. This supports a design-oriented understanding of educational technology research, where knowledge is produced through the combined process of problem analysis, artefact development, validation, and iteration.

Taken together, these contributions position the thesis between educational theory and technical design. On the educational side, the thesis contributes a clearer understanding of why fragmented digital ecosystems are problematic for workshop-based learning. On the technical-design side, it contributes a platform strategy based on the combination of an institutional LMS layer and a custom interactive workshop layer. On the methodological side, it demonstrates how Design Thinking can guide the transition from educational needs to prototype development and formative validation.

The theoretical contribution of the thesis is therefore not limited to the specific prototype developed for one course at Politecnico di Torino. The prototype is context-specific, but the underlying design logic is more transferable. Other higher education contexts involving multi-stage collaborative workflows may face similar problems of tool-switching, data

discontinuity, reduced traceability, and fragmented feedback. For such contexts, the thesis offers a conceptual model: preserve the institutional LMS as the stable course layer, add a custom interactive layer for workflow-specific activities, organise work around persistent project spaces, and treat traceability and continuity as core design principles rather than secondary features.

6.3 Practical Implications

The practical implications of this thesis concern how the proposed platform strategy could inform the future digital support of workshop-based and project-based courses. The prototype was developed for a specific Decision-Making course at Politecnico di Torino, but the issues it addresses are not unique to that course. Many higher education activities depend on connected stages of analysis, collaboration, evaluation, and synthesis, while still being supported by disconnected digital tools. For this reason, the practical value of the thesis lies both in the specific prototype and in the broader implementation logic it demonstrates.

For Politecnico di Torino, the prototype offers a concrete example of how the existing institutional LMS environment could be complemented by a custom interactive workshop layer. The platform does not require the university to replace its LMS or abandon the existing course-management structure. Instead, it suggests a way to extend institutional digital infrastructure by adding specialised functionality for courses that require more interactive, cumulative, and method-specific workflows.

This is important because many university platforms are effective for distributing materials, managing deadlines, communicating with students, and collecting final submissions, but they are less effective at supporting the development of intermediate workshop outputs. The prototype shows that a Django-based layer can address this gap by preserving project data across modules, supporting interactive methods, and making final outputs more traceable to earlier decisions.

For the target Decision-Making course, the platform provides a more coherent digital structure for managing the workshop sequence. The course requires students to move from project framing to stakeholder analysis, problem and objective structuring, indicator selection, ranking, scenario building, scenario analysis, cost–benefit analysis, and final review. In the fragmented workflow, these activities risk becoming separated across different tools and formats. The prototype addresses this by placing the full sequence inside a shared project workspace.

This has practical benefits for students. It reduces the need to search for files, transfer data manually, or reconstruct earlier decisions when moving to later stages. It also helps students understand that each workshop is not an isolated exercise but part of a cumulative project. This can support clearer learning because students can see how early choices influence later evaluation and synthesis.

The platform also has practical implications for professors and teaching assistants. By preserving intermediate outputs, it can make the development of group reasoning more visible. Professors and tutors would not need to rely only on final submitted files to understand the quality of the work. They could review the sequence of project outputs and identify where methodological problems, missing data, or weak reasoning emerged. In a future version, this could support more timely formative feedback and better monitoring of group progress.

The validation findings show that participants valued the all-in-one structure but also expected stronger collaborative and feedback features. This has an important practical implication: an integrated platform should not only store outputs, but also support guidance around those outputs. Future development should therefore prioritise features such as comments, notes, task-management support, tutor feedback fields, clearer completion indicators, and possibly version history. These features would be especially useful for teaching assistants, who often act as the bridge between the professor's methodology and the students' practical work.

The thesis also has practical implications for future institutional integration. The current prototype does not implement full Moodle integration, but it is designed around a strategy that could support future connection through LTI 1.3. This means that the platform can be understood as integration-ready rather than institutionally deployed. The practical implication is that future work should focus on the institutional steps required to connect the Django layer to the official LMS environment. These steps would include Moodle-side configuration, secure authentication, role mapping, server deployment, production database migration, data-protection review, and testing with real course users.

Beyond Politecnico di Torino, the platform strategy may be relevant for other courses organised around multi-stage collaborative workflows. Similar needs can appear in architecture studios, urban planning workshops, engineering design courses, sustainability assessment courses, public policy labs, and other project-based learning environments. In these contexts, students often need to move from qualitative analysis to structured

evaluation, from group discussion to evidence-based decision-making, and from intermediate outputs to final synthesis.

The specific workshop modules developed in this thesis are tailored to the Decision-Making course, but the underlying design logic is transferable. Other courses could adapt the same principles: organise work around a persistent project workspace, connect modules through shared data, preserve intermediate outputs, provide interactive method-specific interfaces, and support final review through an aggregated synthesis page. The technical implementation would vary depending on the course, but the platform logic could remain useful.

The findings also suggest several practical priorities for future implementation. First, navigation and validation feedback must be reliable before any broader use. Users should clearly understand when data has been saved, when required fields are missing, and how to move to the next stage. Second, complex methods require stronger pedagogical scaffolding, including examples, short instructions, and step-by-step guidance. Third, collaboration features should be expanded beyond shared outputs toward comments, notes, task assignment, and live synchronisation where feasible. Fourth, reporting and export functions should be improved so that final outputs are useful for both students and instructors.

Finally, institutional adoption should be gradual. A small-scale pilot would be more appropriate than immediate full deployment. Such a pilot could test the platform with real student groups, collect tutor and professor feedback, evaluate technical performance, and identify institutional requirements for long-term maintenance. This staged approach would preserve the strengths of the prototype while acknowledging its current limits as a master's thesis proof of concept.

In practical terms, the thesis therefore offers both an artefact and a strategy. The artefact is the Django-based prototype developed for the target course. The strategy is the broader approach of complementing the institutional LMS with a custom interactive workshop layer. This approach can help universities support complex project-based learning without abandoning existing infrastructure or forcing students to manage fragmented digital ecosystems on their own.

Chapter 7. Conclusion

This final chapter concludes the thesis by bringing together the main outcomes of the research. It first presents the conclusive remarks and key findings of the work, then clarifies the main limitations of the study, and finally outlines future development directions for the prototype and its possible institutional evolution. The chapter does not repeat the full discussion presented in Chapter 6, but summarises the overall contribution of the thesis and identifies the conditions under which the proposed Moodle–Django platform strategy could be further developed.

7.1 Conclusive Remarks and Key Findings

This thesis addressed the problem of digital fragmentation in collaborative, multi-stage, workshop-based higher education. The starting point of the research was the observation that complex educational workflows are often supported through several disconnected digital tools. In the target Decision-Making course at Politecnico di Torino, students and instructors relied on different environments for course materials, visual collaboration, spreadsheets, file exchange, communication, external decision-support tools, and final presentation outputs. Although each tool may be useful in isolation, their combined use created a fragmented workflow in which data, decisions, feedback, and intermediate outputs were distributed across separate platforms.

The thesis showed that this fragmentation is not only an organisational inconvenience. In workshop-based and project-based learning, fragmentation can affect the continuity of the learning process itself. Students are expected to move progressively from project framing to stakeholder analysis, problem and objective structuring, indicator selection, ranking, scenario building, scenario analysis, cost–benefit evaluation, and final review. When these stages are managed across disconnected tools, students must manually transfer information, reconstruct earlier decisions, manage file versions, and maintain consistency between outputs. This increases cognitive load, weakens data continuity, reduces output traceability, and makes it more difficult for professors and teaching assistants to follow the development of group reasoning.

In response, the thesis developed and evaluated a Django-based interactive workshop prototype designed to complement the institutional LMS environment. The proposed solution was not based on replacing Moodle or creating a fully independent platform.

Instead, the thesis proposed a layered Moodle–Django strategy: Moodle remains the institutional LMS layer for course governance and formal learning-management functions, while Django provides the custom interactive workshop layer for method-specific activities, project data, visual interaction, exports, and final review.

The main finding of the thesis is that a custom interactive workshop layer can reduce digital fragmentation by reorganising the course workflow around persistent project workspaces. The value of the prototype does not lie only in digitising individual workshop activities. Its main contribution is the connection between stages. By preserving intermediate outputs, supporting data reuse, and making project development more reviewable, the prototype demonstrates how a fragmented workshop process can be transformed into a more coherent and traceable digital workflow.

The validation confirmed the educational relevance of this approach. Participants recognised the value of having workshop activities, project outputs, and progression logic organised within one environment. The strongest findings concerned workflow coherence, decision traceability, perceived usefulness, and the reduction of tool-switching. At the same time, the validation showed that a platform of this kind must be reliable, clear, guided, and carefully refined before it can support broader use. Navigation consistency, validation messages, save feedback, guidance in complex modules, export functions, and collaboration features all emerged as important areas for improvement.

The thesis therefore contributes both an artefact and a design strategy. The artefact is the refined Django-based proof-of-concept prototype developed for the target course. The strategy is the broader idea of complementing the institutional LMS with a custom interactive layer for workshop-specific learning activities. This approach shows that higher education platforms should not only provide access to materials and submissions. In courses based on collaboration, decision-making, and iterative project development, digital platforms should also support the continuity of reasoning, the persistence of intermediate outputs, and the reviewability of the learning process.

Within its defined scope, the thesis achieved its aim. It analysed a real educational problem, translated that problem into platform requirements, developed a functional prototype, validated it with informed users, and refined it based on feedback. The final outcome is not a production-ready institutional system, but a refined proof of concept that demonstrates the educational relevance and technical plausibility of the Moodle–Django platform strategy.

7.2 Limitations

The findings of this thesis should be interpreted within the limits of its research design, prototype scope, and validation context. These limitations do not reduce the value of the work, but they clarify the conditions under which the findings are valid and identify the aspects that require further development.

The first limitation concerns the research context. The thesis is grounded in one specific course at Politecnico di Torino. This focus allowed the prototype to respond closely to the real structure, methods, and workflow of the target Decision-Making course. However, it also means that the findings are context-specific. The platform strategy may be transferable to other project-based and workshop-based courses, but such transfer would require adaptation to each course's methods, institutional setting, user roles, assessment structure, and technical infrastructure.

The second limitation concerns the maturity of the prototype. The platform was developed as a proof of concept. It demonstrates the feasibility of a Django-based custom interactive workshop layer, but it does not represent a finished production system. Several advanced functions remain outside the completed scope of the current prototype, including full professor and teaching-assistant dashboards, detailed role management, action logging, version history, advanced assessment tools, richer collaboration features, and more systematic accessibility refinement.

The third limitation concerns deployment. The prototype was developed and validated mainly in a local development environment. It was not tested as a production service under institutional server conditions. Therefore, the thesis cannot claim production-level scalability, long-term availability, security hardening, stable performance under concurrent course use, or institutional maintenance readiness. A live deployment would require server configuration, production database migration, authentication and role-management configuration, security review, data-protection review, backup procedures, and performance testing.

The fourth limitation concerns Moodle integration. The thesis proposes a Moodle–Django platform strategy in which Moodle remains the institutional LMS layer and Django provides the custom interactive workshop layer. However, full Moodle integration through LTI 1.3

was not implemented or validated in a production environment. LTI 1.3 is therefore treated as a future institutional integration direction rather than as a completed component of the current prototype.

The fifth limitation concerns validation. The validation involved six former students of the target course and the course professor as an expert academic observer, producing seven survey responses. This was appropriate for formative prototype evaluation, especially because participants were familiar with the original workshop workflow. However, the findings should not be interpreted as statistically generalisable. They provide evidence of perceived usefulness, usability issues, and refinement priorities, but not large-scale evidence of learning outcomes, institutional adoption, or long-term use.

Finally, the validation was not conducted during a live course deployment with active student groups over a full semester. A real course setting would introduce additional factors, including deadlines, repeated use, group dynamics, tutor intervention, concurrent access, and assessment pressure. These conditions could reveal additional usability, collaboration, and performance issues that were not visible in a single validation session.

In summary, the thesis demonstrates the educational relevance and technical plausibility of the proposed platform strategy within a bounded proof-of-concept scope. Its main limitations concern contextual generalisability, production deployment, full institutional integration, advanced collaboration and feedback features, and long-term validation in a live course environment.

7.3 Future Development Directions

The thesis opens several directions for future development and research. These directions extend the current proof-of-concept prototype toward a more complete, institutionally deployable, and pedagogically richer platform. They also respond directly to the limitations identified in the previous section.

The first future direction is full institutional integration between the Django workshop layer and the official LMS environment. In the current thesis, Moodle is treated as the institutional LMS layer and Django as the custom interactive workshop layer, but full Moodle integration through LTI 1.3 remains a future development direction. A next step would be to configure the platform as an external tool launched from Moodle, supporting secure access, identity transmission, role mapping, and a more seamless student experience. This would require

collaboration with institutional IT services, Moodle-side configuration, authentication setup, security review, and data-protection assessment.

The second direction concerns production deployment and technical hardening. The current prototype uses SQLite in the development environment, which is appropriate for local testing and proof-of-concept validation. A production version should migrate to a database such as PostgreSQL in order to support concurrent access, stronger reliability, backup procedures, and long-term data management. Future development should also include server deployment, environment configuration, error handling, logging, backup strategies, security testing, and performance testing under realistic classroom conditions.

The third direction is a controlled pilot implementation in the target course. The validation conducted in this thesis was formative and involved former students and the course professor. A live pilot would test the platform under real teaching conditions, including active student groups, deadlines, repeated use across several weeks, tutor intervention, assessment expectations, and concurrent access. Such a pilot would provide stronger evidence about educational usefulness, usability over time, group collaboration, instructor workload, and technical reliability.

A fourth direction is the development of more advanced professor and teaching-assistant support tools. The current prototype focuses primarily on the student-side workflow and project review. Future versions should strengthen the instructor side through dashboards, progress monitoring, feedback tools, and review functions. Useful features could include a professor dashboard showing the progress of all groups, teaching-assistant monitoring views, completion indicators, alerts for incomplete sections, and the ability to attach comments directly to specific workshop artefacts.

A fifth direction concerns collaboration and accessibility refinement. The current prototype supports collaboration mainly through shared project workspaces and preserved group outputs. Future development could extend this through contextual comments, shared notes, task assignment, improved save confirmation, live auto-save, and possibly real-time synchronisation where technically feasible. Accessibility should also be developed more systematically through responsive testing across devices, keyboard navigation review, contrast checks, clearer text hierarchy, and more consistent guidance for complex methodological tasks.

Another important direction is the improvement of reporting, export, and traceability functions. The current prototype includes several export and review features, but future versions could provide more complete final reports, structured Word or PDF outputs, downloadable datasets, versioned exports, and clearer links between intermediate decisions and final conclusions. A stronger reporting system would support both students and instructors by making the development of group reasoning easier to review.

A final, longer-term direction is the possible integration of AI-supported guidance. This should not be understood as a replacement for professors, teaching assistants, or course methodology. Instead, an AI component could provide contextual support within the platform, helping students understand instructions, check whether required fields are complete, receive reminders about methodological steps, or ask questions about workshop procedures. Any AI-supported function would need to be carefully controlled, transparent, and supervised by the teaching team, especially where assessment-related activities are involved.

Together, these future directions show how the prototype could evolve from a refined proof of concept into a more complete digital environment for workshop-based higher education. The most immediate priorities are institutional integration, production deployment, live-course piloting, and stronger professor/teaching-assistant support. Collaboration, accessibility, reporting, and AI-supported guidance represent further directions for increasing the platform's educational value.

References

- 1EdTech Consortium. (2019). Learning Tools Interoperability (LTI) Core Specification, version 1.3. <https://www.imsglobal.org/spec/lti/v1p3>
- Ahmad, A., Moughal, I., Alexandru, C. A., & Constantin, A. (2021). Systematic review of online collaborative whiteboard platforms (OCWP) for higher education. Abstract from University of Edinburgh Learning and Teaching Conference 2021.
- Bond, M., Marín, V. I., Dolch, C., Bedenlier, S., & Zawacki-Richter, O. (2018). Digital transformation in German higher education: Student and teacher perceptions and usage of digital media. *International Journal of Educational Technology in Higher Education*, 15, Article 48. <https://doi.org/10.1186/s41239-018-0130-1>
- Brown, T. (2008). Design thinking. *Harvard Business Review*, 86(6), 84-92.
- Brown, T., & Wyatt, J. (2010). Design thinking for social innovation. *Stanford Social Innovation Review*, 8(1), 31-35.
- Django Software Foundation. (2026). Django documentation. <https://docs.djangoproject.com/>
- Dorst, K. (2011). The core of 'design thinking' and its application. *Design Studies*, 32(6), 521-532. <https://doi.org/10.1016/j.destud.2011.07.006>
- Guo, P., Saab, N., Post, L. S., & Admiraal, W. (2020). A review of project-based learning in higher education: Student outcomes and measures. *International Journal of Educational Research*, 102, Article 101586. <https://doi.org/10.1016/j.ijer.2020.101586>
- Henderson, M., Selwyn, N., & Aston, R. (2017). What works and why? Student perceptions of useful digital technology in university teaching and learning. *Studies in Higher Education*, 42(8), 1567-1579. <https://doi.org/10.1080/03075079.2015.1007946>
- Kirschner, F., Paas, F., & Kirschner, P. A. (2009). A cognitive-load approach to collaborative learning: United brains for complex tasks. *Educational Psychology Review*, 21, 31-42. <https://doi.org/10.1007/s10648-008-9095-2>
- Kokotsaki, D., Menzies, V., & Wiggins, A. (2016). Project-based learning: A review of the literature. *Improving Schools*, 19(3), 267-277. <https://doi.org/10.1177/1365480216659733>

- Liedtka, J. (2015). Perspective: Linking design thinking with innovation outcomes through cognitive bias reduction. *Journal of Product Innovation Management*, 32(6), 925-938. <https://doi.org/10.1111/jpim.12163>
- Lonn, S., & Teasley, S. D. (2009). Saving time or innovating practice: Investigating perceptions and uses of learning management systems. *Computers & Education*, 53(3), 686-694. <https://doi.org/10.1016/j.compedu.2009.04.008>
- Miranda, J., Navarrete, C. D., Noguez, J., Molina-Espinosa, J. M., Ramirez-Montoya, M. S., Navarro-Tuch, S. A., Bustamante-Bello, M. R., Rosas-Fernandez, J. B., & Molina, A. (2021). The core components of Education 4.0 in higher education: Three case studies in engineering education. *Computers & Electrical Engineering*, 93, Article 107278. <https://doi.org/10.1016/j.compeleceng.2021.107278>
- Petchamé, J., Iriondo, I., Korres, O., & Paños-Castro, J. (2023). Digital transformation in higher education: A qualitative evaluative study of a hybrid virtual format using a smart classroom system. *Heliyon*, 9(6), Article e16675. <https://doi.org/10.1016/j.heliyon.2023.e16675>
- Plattner, H., Meinel, C., & Leifer, L. (Eds.). (2011). *Design thinking: Understand - improve - apply*. Springer. <https://doi.org/10.1007/978-3-642-13757-0>
- Salmon, G. (2019). May the Fourth be with you: Creating Education 4.0. *Journal of Learning for Development*, 6(2), 95-115. <https://jl4d.org/index.php/ejl4d/article/view/352>
- Stahl, G., Koschmann, T., & Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 409-426). Cambridge University Press.
- Thomas, J. W. (2000). A review of research on project-based learning. The Autodesk Foundation.
- Vuopala, E., Hyvönen, P., & Järvelä, S. (2016). Interaction forms in successful collaborative learning in virtual learning environments. *Active Learning in Higher Education*, 17(1), 25-38. <https://doi.org/10.1177/1469787415616730>
- Wang, H. (2021). Design and implementation of web online education platform based on user collaborative filtering algorithm. In *Proceedings of the 2021 5th International Conference on Electronic Information Technology and Computer Engineering* (pp.

- 2911-2918). Association for Computing Machinery.
<https://doi.org/10.1145/3482632.3487539>
- Zhang, L., & Ma, Y. (2023). A study of the impact of project-based learning on student learning effects: A meta-analysis study. *Frontiers in Psychology*, 14, Article 1202728. <https://doi.org/10.3389/fpsyg.2023.1202728>
- Becker, J., Knackstedt, R., & Pöppelbuß, J. (2009). Developing maturity models for IT management. *Business & Information Systems Engineering*, 1, 213–222.
<https://doi.org/10.1007/s12599-009-0044-5>
- de Bruin, T., Freeze, R., Kulkarni, U., & Rosemann, M. (2005). Understanding the main phases of developing a maturity assessment model. In *Proceedings of the 16th Australasian Conference on Information Systems*.
- Marshall, S. (2010). A quality framework for continuous improvement of e-learning: The e-learning maturity model. *Journal of Distance Education*, 24(1), 143–166.
- Mettler, T. (2011). Maturity assessment models: A design science research approach. *International Journal of Society Systems Science*, 3(1/2), 81–98.
<https://doi.org/10.1504/IJSSS.2011.038934>
- ISO/IEC. (2023). ISO/IEC 25010:2023: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model. International Organization for Standardization.
- W3C. (2024). Web Content Accessibility Guidelines (WCAG) 2.2. World Wide Web Consortium. <https://www.w3.org/TR/WCAG22/>
- Nielsen, J. (1994). *Usability engineering*. Morgan Kaufmann.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319–340.
<https://doi.org/10.2307/249008>