Ruhr Master School
of Applied Sciences

Dieses Wahlpflichtmodul ist ein Angebot der:

Fachhochschule Dortmund
University of Applied Sciences and Arts

Masterstudiengang Digital Transformation

Innovation Driven Software Engineering

Hochschule Bochum
Bochum University
of Applied Sciences

Fachhochschule Dortmund
University of Applied Sciences and Arts

Westfälische Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

STIFTUNG MERCATOR

Number						
MOD1-01	**Innovation Driven Software Engineering**					
Language	**Duration**	**Semester**	**Frequency of offer**		**Type of module**	**ECTS**
english	1 Semester	1	Winter semester only		Compulsory	6

1	Events	Eventtype	Planned group size	Workload		HPW
				Contact-hours	Self-study	
			25	60	120	4

2 Learning Outcomes / Competencies

Knowledge:
- Understanding the role of innovation in the software development lifecycle, particularly in the context of digital transformation
- Mastering the stages and techniques of Design Thinking and in user-centered software design
- Gaining a deep understanding of Agile methodologies, including SCRUM and Kanban, and their relevance to iterative software development.
- Familiarity with essential tools for software development, including version control (Git), bug tracking systems, UML modeling, and Agile project management tools (e.g., Jira).
- Understanding DevOps principles, continuous integration/continuous delivery practices, and tools for automating software deployment and monitoring.
- Knowledge of emerging trends such as AI-driven development, cloud-native architectures, microservices, IoT, and blockchain technologies in the context of innovative software solutions.
- Understanding the key elements of successful pitches, including audience analysis, value propositions, and clear communication of technical solutions to non-technical stakeholders.

Skills:
- Apply Design Thinking techniques to create and prototype software solutions that effectively address user needs and challenges
- Setup and manage a team based on agile principles
- Utilize software tools to manage collaborative development in team environments
- Create and interpret UML diagrams (use case, sequence, class diagrams, etc.) and other modeling tools for designing and communicating software architecture.
- Prototype cloud-native solutions using containers, microservices, and serverless architectures
- Adapt and integrate new and emerging technologies into software projects
- Develop and deliver compelling pitches for software innovations, tailored to various audiences, including investors, stakeholders, and users.

Competence – attitude:
- Ability to think critically and apply innovative methodologies to design, develop, and refine software solutions that address complex, real-world problems
- Lead cross-functional teams in the development of software products that prioritize user needs
- Ability to work in interdisciplinary teams, integrating perspectives from design, business, and technology to drive software innovation and translate between different domains
- Apply Lean startup principles to turn software prototypes into viable products or business ventures, validating ideas, and scaling solutions in the market
- Competence in considering ethical, social, and environmental impacts of software solutions
- Practice communication strategies to clearly explain technical concepts, business value, and user benefits.

3 Course Description and Course Structure

Innovation-Driven Software Engineering explores the intersection of modern software development practices and innovation, focusing on creating cutting-edge, user-centered solutions. In today's digital landscape, software is not just about functionality; it must also emphasize novelty, usability, and user

	delight. Modern software development processes are highly creative, iterative, and dynamic, requiring collaboration with multiple stakeholders, particularly end users. Design Thinking plays a pivotal role in this process, serving as a human-centered methodology that integrates users throughout the development journey to ensure that the final product addresses their needs, challenges, and pain points. This process fosters continuous innovation by refining ideas through empathy-driven exploration, often resulting in prototypes that can serve as the foundation for startup ventures or new business models. In parallel, Agile Software Development methodologies—such as SCRUM and Kanban—complement these innovation processes by emphasizing short, iterative development cycles, frequent user feedback, and the ability to quickly pivot in response to changing requirements. Agile's adaptability, when combined with innovative thinking, fosters environments that are not only reactive but proactive, driving continual improvements in both product quality and user satisfaction. To support this dynamic and iterative approach, a robust DevOps pipeline is introduced, focusing on integrating development and operations teams to ensure continuous integration, deployment, and monitoring. This infrastructure, aided by tools like Version Control Systems (e.g., Git), Bug Tracking and Ticket Management Systems (e.g., Jira), enables efficient team collaboration, rapid troubleshooting, and effective project management. Furthermore, modeling techniques such as Unified Modeling Language (UML) Diagrams, Entity-Relationship Diagrams (ERD), and Business Process Model and Notation (BPMN) are essential for visualizing software architecture, workflows, and system interactions. These tools provide clarity in the design and development phases, ensuring alignment between stakeholder vision and the technical implementation. Students will also explore emerging trends in software engineering, such as AI-driven development, cloud-native architectures, and microservices, all of which play a critical role in digital transformation and large-scale innovation. Topics such as Test-Driven Development (TDD) and Continuous Delivery/Continuous Deployment (CI/CD) will further solidify their understanding of modern software engineering practices. By the end of this course, students will gain not only technical expertise but also the innovation mindset required to lead digital transformation projects, creating software solutions that are both innovative and highly responsive to evolving user needs. Course Structure • Introduction to Innovation in Software Engineering • Design Thinking and Prototyping • Agile Methodologies in Software Development • Tools and Techniques for Modern Software Development • DevOps and Continuous Delivery • Advanced Topics in Software Innovation
4	**Teaching Methods** • Interactive lectures: Traditional lecture format enhanced with real-time discussion and interactive elements. If applicable, industry professionals, startup founders, or tech innovators deliver guest lectures with additional industry insights • Groupwork: Collaborative projects where students work in interdisciplinary teams to prototype innovative software solutions • Hands-on Workshops: Practical sessions where students apply tools and techniques discussed in class • Self-Directed Learning and Research: Students explore specific areas of interest related to the course content through independent study and research • Peer Reviews and Critique: Students provide constructive feedback on each other's work during project development and pitch presentations
5	**Participation Requirements** none

6	**Examination Forms** Assessment of the course: Theoretical knowledge (50%): Written or Oral Exam at the end of the course, Practical Skills and Scientific Competences (50%): Development of an innovative software prototype or concept for a given real-world challenge using design thinking and agile methodologies, supported by documentation and a presentation including pitching.
7	**Requirements for the award of credit points** Passed exam and passed semester assignments
8	**Usability of the module (in other study programs)** • MOD2-01 – Usability Engineering • MOD-E03 – Human Centered Digitalization
9	**Significance of the grade for the final grade** 5,00 %
10	**Module Representative** Prof. Dr. Sabine Sachweh **Lecturer** see current course catalog or individual study plan in the Portal of the University of Applied Sciences and Arts Dortmund
11	**Literature** Plattner, H., Meinel, C., & Weinberg, U. (2009). Design thinking (p. 64f). Landsberg am Lech: Mi-Fachverlag. Uebernickel, F., Jiang, L., Brenner, W., Pukall, B., Naef, T., & Schindlholzer, B. (2020). Design thinking: The handbook. World Scientific. Przybilla, L., Klinker, K., Lang, M., Schreieck, M., Wiesche, M., & Krcmar, H. (2020). Design thinking in digital innovation projects—Exploring the effects of intangibility. IEEE Transactions on Engineering Management, 69(4), 1635-1649. Belling, S. (2020). Design Thinking with Agile: Shared concepts and applications. Succeeding with Agile Hybrids: Project Delivery Using Hybrid Methodologies, 109-117. Corral, L., & Fronza, I. (2018, September). Design thinking and agile practices for software engineering: an opportunity for innovation. In Proceedings of the 19th Annual SIG Conference on Information Technology Education (pp. 26-31). Tsui, F., Karam, O., & Bernal, B. (2022). Essentials of software engineering. Jones & Bartlett Learning.