COURSE SYLLABUS AND COURSE REQUIREMENTS
ACADEMIC YEAR 2025/26 SEMESTER 2

Course title	*Project 2*
Course Code	IVB068AN
Hours/Week: le/pr/lab	0 / 0 / 6
Credits	11
Degree Programme	Computer Science Engineering BsC
Study Mode	*Full-time*
Requirements	-
Teaching Period	2025/26-2
Prerequisites	Project 1
Department(s)	Systems and Software Technologies
Course Director	Dr. STORCZ, Tamas
Teaching Staff	*Dr STORCZ, Tamas; LABORCI, Gergely; LÉNÁRT, Anett; NOVÁK, Péter*

COURSE DESCRIPTION

A short description of the course (max. 10 sentences).
Neptun: Instruction/Subjects/Subject Details/Basic data/Subject description

The Project 2 course is the direct practical continuation of the theoretical and design foundations established in Project 1. This semester, the emphasis is on the transition from "paper to reality": students implement their projects based on the functional and technical specifications they created themselves. All phases of the work are carried out individually, so each student bears full responsibility for every step of the development process. During the semester, the refinement of the plans is followed by an intensive implementation phase, and the process concludes with rigorous testing and the preparation of documentation. The aim of the course is for students to experience the implementation stage of the full software (or system) development life cycle. By the end of the course, students present a working, validated, and documented final product. This process requires not only technical knowledge but also disciplined time management.

SYLLABUS

Neptun: Instruction/Subjects/Subject Details/Syllabus

1. GOALS AND OBJECTIVES

Goals, student learning outcome.
Neptun: Instruction/Subjects/Subject Details/Syllabus/Goal of Instruction

The primary objective of the course is to enable students to independently implement and assure the quality of a previously designed complex system. The intended learning outcomes include confident, proficiency-level use of the selected technology stack (programming languages, frameworks, hardware devices) and the ability to resolve unexpected technical obstacles that arise during development independently. Students will learn system-level testing methodologies, master structured debugging processes, and gain routine in managing technical debt. Another key goal is to develop precise technical writing skills, so that students can produce clear, standards-compliant documentation that is understandable to both professional and end-user audiences. By the end of the course, students will demonstrate that they can carry a project through on their own from concept to finished product.

2. COURSE CONTENT

Neptun: Instruction/Subjects/Subject Details/Syllabus/Subject content

	TOPICS
LABORATORY PRACTICE	*1. Finalizing the P1 specifications* *2. Foundations: project setup and structure* *3. Core development and testing* *4. POC phase and functional testing* *5. Development* *6. Testing and bug fixing* *7. Documentation and presentation*

DETAILED SYLLABUS AND COURSE SCHEDULE

ACADEMIC HOLIDAYS INCLUDED

PRACTICE, LABORATORY PRACTICE

week	**Topic**	**Compulsory reading; page number (from … to …)**	**Required tasks (assignments, tests, etc.)**	**Completion date, due date**
1.	Finalizing the P1 specifications			
2.	Foundations: project setup and structure		git repo URL	end of week #2
3.	Core development		development schedule	end of week #3
4.	Core development			
5.	Core development, unit tests		presentation URLs	end of week #5
6.	Core development, integration tests			
7.	POC phase, functional testing		POC presentation	On practice class
8.	Development, POC extension		POC presentation	On practice class
9.	Spring break			
10.	Development, testing/bug fixing			
11.	Development, testing/bug fixing			
12.	Documentation, presentation		MVP presentation	On practice class
13.	Documentation, presentation		MVP presentation	On practice class
14.	Presentation retake			On practice class

3. ASSESSMENT AND EVALUATION

(Neptun: Instruction/Subjects/Subject Details/Syllabus/Examination and Evaluation System)

ATTENDANCE

In accordance with the Code of Studies and Examinations of the University of Pécs, Article 45 (2) and Annex 9. (Article 3) a student may be refused a grade or qualification in the given full-time course if the number of class absences exceeds 30% of the contact hours stipulated in the course description.

Method for monitoring attendance *(e.g.: attendance sheet / online test/ register, etc.)*

participant list / on-line test / progress report submission

ASSESSMENT

Cells of the appropriate type of requirement is to be filled out (course-units resulting in mid-term grade or examination). Cells of the other type can be deleted.

Course resulting in mid-term grade *(PTE TVSz 40§(3))*

Mid-term assessments, performance evaluation and their ratio in the final grade *(The samples in the table to be deleted.)*

Type	Assessment	Ratio in the final grade
Implementation progress/git activity	*100%*	*30 %*
Technical specification	*100%*	*20 %*
MV system developed, up and running	*100%*	*50 %*

Opportunity and procedure for re-takes (PTE TVSz 47§(4))

The specific regulations for improving grades and resitting tests must be read and applied according to the general Code of Studies and Examinations. E.g.: all tests and assessment tasks can be repeated/improved at least once every semester, and the tests and home assignments can be repeated/improved at least once in the first two weeks of the examination period.

Resubmission after first evaluation is possible before week #13

Grade calculation as a percentage

based on the aggregate performance according to the following table

Course grade	Performance in %
excellent (5)	85 % …
good (4)	70 % ... 85 %
satisfactory (3)	55 % ... 70 %
pass (2)	40 % ... 55 %
fail (1)	below 40 %

The lower limit given at each grade belongs to that grade.

4. SPECIFIED LITERATURE

In order of relevance. (In Neptun ES: Instruction/Subject/Subject details/Syllabus/Literature)

COMPULSORY READING AND AVAILABILITY

[1] Published document templates

RECOMMENDED LITERATURE AND AVAILABILITY