UCLouvain

linfo1002

2026

IT projects 2

5.00 credits	30.0 h + 30.0 h	Q2

Language :	French
Place of the course	Louvain-la-Neuve
Prerequisites	These projects assume the parallel acquisition of basic notions in programming as targeted by the course LSINC1101.
Main themes	This teaching unit revolves around programming projects. The objectives are: • to model simple situations using computer systems; • to design a simple interactive service accessible via a web interface and using a relational database; • understand the basic principles of how interactive websites work; • to confront professional constraints: group work, meeting deadlines, sense of responsibility; • to acquire transversal skills taking notes, writing reports.
Learning outcomes	**At the end of this learning unit, the student is able to :** With regard to the AA reference system of the "Bachelor in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • S1.I2 • S2.1, S2.2, S2.4 • S4.1, S4.2, S4.3 • S5.2, S5.3, S5.4, S5.5 Students who successfully complete this course will be able to: • analyze a concrete situational problem requiring the development of an interactive website and perceive the role that it will have to play; • design the website corresponding to the identified needs and justify the design choices; • implement the website wisely using the Python language and a database; • document the web application and implement unit tests to validate its proper functioning; • use a programming environment comprising integrated programming tools such as an editor, compiler, debugger, and tools for handling files, tests, documentation. Students will have developed methodological and operational skills. In particular, they will have developed their ability to: • contribute to group functioning within the framework of cooperative active learning devices of the project type, explain the issues (advantages, disadvantages) of group work and give some operational leads to promote effective group work; • lead a process of developing a computer application; • understand a situational problem described via written documents, an oral presentation and extract what makes its essence and reformulate it in order to define the expected result; • establish the specifications and a roadmap for a project; • outline the architecture of the interactive website to give a high-level description allowing any IT specialist to quickly perceive its structure; • document the application so that it can easily be adapted later by another IT specialist; • design and perform tests to validate the developed application; • collaborate effectively on application development; • write a coherent and structured project report in order to convince of the success of the project;

Evaluation methods	• Evaluation of individual and group participation during the year, based on the follow-up by the person from the teaching staff delegated to follow up the group (15%). • Evaluation of the different phases of the project on the basis of written reports and documentation (70%). • Evaluation of the final phase of the project based on the oral presentation (10%). • Peer review assessment (5%). Although participation counts for 15%, a student who does not actively participate in the work of his group may see his other grades (such as the mark of a phase of a project) reduced based on, for instance (non-exhaustive), code tracking via Git, group evaluation, monitoring by tutors, assistants and the professor. Participation in all teaching activities is compulsory. The non-submission of a project will result in an absence rating. There is no exam. Students who fail in June will be able to redo an individual project during the summer which will replace the 70% of the evaluation of the phases of the project. An oral presentation of the project will also be organized (10%). The other activities are not organized during the summer, so the marks of participation (15%) and peer review (5%) will be kept for the second session. Although AI is one of modern digital tools, it is essential to understand the basics of programming yourself in order to use it effectively. This course aims to teach the fundamentals of computer science through a relatively simple project for an expert. Therefore, the use of generative AI (such as ChatGPT, Consensus, Perplexity, etc.), other than for writing assistance (i.e., rephrasing a sentence you wrote yourself, correcting it, or translating it), is strictly prohibited. The instructor reserves the right to give a 0 to some or all projects containing AI-generated code. The use of AI may also be considered a case of cheating, which will be addressed by the student's judging panel. The instructor also reserves the right to summon a student in case of doubt about the authenticity of the code in order to verify that they are capable of explaining the code they have submitted. Students taking the Human-Machine Interface course in parallel will have to apply their knowledge in the framework of Project 2. The evaluation of the interface will be done through a joint final report between the two courses, but this part does not count in the evaluation of the Project 2 course. Students are invited to refer to the HMI course sheet concerning the evaluation of the interface itself.
Teaching methods	Project learning in groups of multiple students. Each group will be followed by a member of the teaching staff. 4 or 5 lectures are organized (depending on progress) to give the basics of web technologies, the python Flask framework, SQL, unit tests or Graph.js. Students are encouraged to ask questions during the lecture, and of course the follow-up sessions. This teaching unit addresses issues related to sustainable development and transition through a module dedicated to exploring the performance and power consumption of web servers according to the technological choice among various web technologies.
Content	The course assumes basic knowledge of the Python programming language, as covered in LINFO1101. The course also assumes the student is learning databases and SQL in parallel, as covered in LINFO1006. Students work in groups to solve more complex problems than those covered in the programming course and apply their database knowledge at the end of the year. Each group will work on a unique project, but with several deadlines and intermediate developments. Generally speaking, students should learn to: • work effectively in a group • write correct programs • document their codes During the different phases, the following skills will be assessed: • write tests that validate the proper functioning of their programs • document their programs and associated testing • take a critical look at the work of other groups of students to help them improve (constructive peer-review) • evaluate the performance of their programs • find python libraries and modules that solve similar problems • compare the features and performance of different solutions to the same problem • document and analyze these differences in functionality and performance • build simple SQL queries • quantitative analysis of the energetic impact of selecting a particular web technology
Inline resources	Moodle (https://moodle.uclouvain.be/enrol/index.php?id=3990) Syllabus (https://sites.uclouvain.be/linfo1002/)

Other infos	The course presupposes knowledge of Python. If the student has missed the Computer Science 1 course, it is his/her duty to participate in the remedial sessions. The course presupposes knowledge of SQL language for selection queries, projections and joins and database concepts from the 8th week, as organized in the LINFO1006 course. A student who does not have LINFO1006 in parallel will have to follow online course modules, without however being able to claim additional credits or a difference in treatment. Project-based learning means that **a significant part of the skills must be acquired by the students themselves** via the resources listed in the syllabus, such as online tutorials.
Faculty or entity in charge	INFO

Programmes containing this learning unit (UE)				
Program title	Acronym	Credits	Prerequisite	Learning outcomes
Bachelor in Computer Science	SINF1BA	5		🔍
Minor in Computer Sciences	MINSINF	5		🔍