esade

TEACHING GUIDE - 2025-2026

Software Development for Business

UGRA_016269

Departments	Data, Analytics, Technology and Artificial Intelligence (DATA), Dept. of Operations, Innovation & Data Sciences
Teaching Languages	English
ECTS	6
Teacher responsible	Maider Abad Vázquez - maider.abad@esade.edu

Course Goals

• Mastery of Software Development Fundamentals: The primary objective is to provide students with a comprehensive understanding of software development. This includes learning the basics of software design principles, patterns, and best practices to ensure the creation of effective and maintainable code.

• Proficiency in Essential Tools and Technologies: The course aims to familiarize students with fundamental tools and technologies in software development. This includes terminal usage, Git and version control, Docker and containerization, as well as server access and usage.

• Development of Practical Skills: Students are expected to gain solid practical skills throughout the course. This is achieved through hands-on exercises and projects that reinforce learning and enable them to apply their knowledge in real-world situations.

• Preparation for Efficient Collaboration: Another important objective is to prepare students for effective collaboration in software development projects. This is accomplished by learning robust version control practices using Git and GitHub, as well as by understanding how to work in server and container environments.

Previous knowledge

This course provides an introduction to the applicability, deployment, and design of software. It is necessary to know how to program or have a general understanding of programming. Additionally, it is important to understand basic concepts such as what a script is, how to use a terminal and other fundamental elements. Familiarity with creating and running scripts, using terminal commands, and having a general understanding of Unix systems and

virtual environments can be particularly useful.

Prerequisits

This course will be managed through a dedicated eCampus website. Students will find there all the necessary materials, including assigned readings and pre-class work, class materials, and further references. Students should familiarize themselves with this environment before the start of the course and check for updates regularly. Additionally, to participate in this course, students must download and install Docker and Git for coding, as well as create a GitHub profile. Detailed instructions on how to install the required software will be provided, along with future guidelines for library installation and server access. It is the students' responsibility to ensure that they can write and execute code on their computers. This step is crucial for completing coursework and engaging in coding exercises.

Teaching methodology

The teaching methodology integrates theoretical content, Q&A sessions, hands-on exercises, and at-home practice. Each lesson will begin with theoretical concepts followed by a dedicated time for questions and clarifications. Students will then engage in hands-on exercises to apply the theoretical knowledge in practical scenarios. Additionally, at-home practice assignments will reinforce learning and provide further hands-on experience. Some sessions will also be reserved exclusively for the development of a group project, which will be delivered in stages throughout the course until its completion.

Description

Course contribution to program

This course lays the groundwork for the System Architecture module in the BBA & BAIB program, equipping students with essential knowledge in software applicability, deployment, and design. The skills acquired here will pave the way for upcoming courses in Database Management and Design, Cloud Solutions and DevOps management. Following this course, students will delve into bridging scientific and specialized subjects like programming and mathematics with business studies. While emphasizing tools like Docker and Git, the problem-solving skills developed will be adaptable across various software environments and technologies.

Short description

Students are introduced to modern software development practices through a combination of collaborative coding, version control workflows, and guided technical challenges. Students will engage in hands-on programming, code reviews, and iterative project work to develop robust coding habits and problem-solving skills. The course lays a strong foundation for advanced courses in system design, DevOps, and scalable application development.

Content

#	Topic
1	Software Design Principles and Best Practices: Students will explore foundational principles, patterns, and best practices in software design, laying the groundwork for writing effective and maintainable code. Additionally, they will learn how to organize and manage a software project using the Scrum methodology, gaining practical insights into agile planning, sprint execution, and team collaboration.
2	Terminal Usage and Scripting: Students will learn essential terminal commands, Bash scripting, and file management techniques to support efficient development workflows and automation tasks.
3	Git and Version Control: This section focuses on setting up repositories, managing branches, and collaboration workflows using Git and GitHub, ensuring robust version control practices within software projects.
4	Introduction to Docker and Containerization: Students will gain an understanding of Docker, including setting up and managing containers and creating Docker images, enabling them to streamline application deployment processes.
5	Server Usage and Access: This section covers server environments, teaching students how to set up and access remote servers to deploy applications effectively, and preparing them for real-world deployment scenarios.

Assessment

Tool	Assessment tool	Category	Weight %
Individual or team exercises	As part of their autonomous work, students will be required to complete a series of assignments during their independent study hours. Assignments may include individual tasks, group projects and oral presentations.	Ordinary round	30.00%
Written and/or oral exams	Students will be given the opportunity to sit for a mid-term exam. This exam is designed to prepare you for the final exam and to evaluate your knowledge and understanding of the topics covered in the first half of the course.	Ordinary round	30.00%
Written and/or oral exams	Students will be required to sit for a final exam, where they will demonstrate their understanding of the topics covered throughout the course. The final exam will consist of a variety of	Ordinary round	40.00%

Tool	Assessment tool	Category	Weight %
	assessment methods, including short exercises, quizzes, and coding		
Written and/or oral exams	Students who fail to pass the course will be required to take a retake exam, which will be a comprehensive assessment of all course material. This exam will account for 70% of their final course grade, 30% corresponds to the grade for assignments.	Retake	70.00%
Attendance and punctuality	Attendance. In accordance with ESADE regulations, attendance is mandatory for this course. Students who fail to attend 80% of the course will not be allowed to pass and will be required to sit the retake exam.	Ordinary round	0.00%

PROGRAMS

DBAI21-Double Degree in Business Administration and Artificial Intelligence for Business (Undergraduates: Business)

DBAI21 Year 3 (Mandatory)

DBAI21 Year 1 (Mandatory)

DBAI23-Double Degree in Business Administration and Artificial Intelligence for Business (Undergraduates: Business)

DBAI23 Year 1 (Mandatory)

DBAI23 Year 3 (Mandatory)