Syllabus: Design & Testing

Goals

The goal of this subject is to learn how to produce a typical small- to mid-scale web information system using industrial-strength components, tools, and methods.

Prospective students

We expect our students to be interested in working as developers of web information systems, to be active (that is, problem-solving, dynamic, curious, and enquiring), to have a strong sense of ethics (in particular, we expect them not to tolerate cheating, including, obviously, plagiarism), and to know how to behave in a formal setting (that is, addressing people, wearing appropriate attires, and observing appropriate hygiene and etiquette customs). They are also expected to have good abstraction skills, good modelling skills, a good command of the Java platform, and a good command of the Eclipse integrated development environment. They are also expected to have a little experience at developing simple web information systems; at least, they must have passed the following subjects: “Fundamentos de la Programación”, “Análisis y Diseño de Datos y Algoritmos”, and “Introducción a la Ingeniería del Software y los Sistemas de Información”.

Competences

This subject helps the students develop the following general (GXX) and specific (EXX) competences, as required by the Official Verification Sheet:

- G05. Ability to design, develop, and maintain systems, services, and applications using the methods of Software Engineering as a tool for quality assurance.
- G09. Ability to solve problems and make decisions with initiative, autonomy, and creativity. Ability to learn to communicate and get the knowledge across, skills, and abilities of the Technical Computing Engineer profession.
- E25. Ability to develop, maintain, and evaluate software systems and services that satisfy the user requirements and behave reliably and efficiently, are affordable to develop and maintain, and meet quality standards, applying the theories, principles, methods, and practices of Software Engineering.
- E28. Ability to identify and analyse problems and design, develop, implement, verify, and document software solutions based on adequate theories, models, and techniques.
- E30. Ability to design appropriate solutions in one or more application domains using Software Engineering methods that integrate ethical, social, legal, and economic issues.

Workload

This subject requires the following workload to our students:

- Lectures: four hours per week, 60 hours per semester, 120 hours per academic year.
- Homework: six hours per week, 90 hours per semester, 180 hours per academic year.
- Workweeks: fifteen per semester, two semesters per academic year.

Academic activities

The academic activities include, but are not limited to the following:

- Theory lectures: these lectures consist of slide-based presentations in which the lecturers present the knowledge required to achieve the goals of this subject.
- Problem lectures: in these lectures, the students are expected to present their solutions to a number of problems that were previously assigned by the lecturers.
- Project work lectures: in these lectures, the students are expected to work on several small projects that are expected to help them put their knowledge into practice.
- Homework: it consists of reviewing the material delivered in the lectures, working on problems, and working on projects to achieve the goals of the subject.
- Tutorials: the students can book tutorials to clarify some issues. Please, read document "On your tutorials" before booking one.

Learning materials

The learning materials include, but are not limited to the following:

- Annotated slides: the slides include a written transcript of what the lecturers comment on them. Please, note that the lecturers might provide additional insights into some slides if requested by the students. Such additional insights are not documented.
- Project statements: these documents describe the requirements and the evaluation procedures that the students must meet regarding each deliverable.
- Recommended bibliography: unless otherwise stated, the learning materials that are provided by the lecturers are enough to achieve the goals of this subject; the lectures, however, include additional references to selected bibliography that might help our students learn more.
- Resources: the resources include workspace templates, project templates, components, and configurations that are used throughout the course.

Most learning materials are available at the USE's e-learning platform (http://ev.us.es) prior to each lesson. Bibliography references are introduced at the end of each lesson and they are typically available at the USE's e-library (http://bib.us.es). Resources are available at the subject's repository (ftpes://practica:practica@pimpom.lsi.us.es/DT) until the first deliverable deadline.

Evaluation and grading

According the current regulations, there are three evaluation summons, namely: June, September, and December. The evaluation and the grading are based on deliverables and control checks, but the procedures are slightly different depending on the summon.

Key ideas regarding deliverables

- A deliverable is a package that students must produce in groups of up to six people.
- A deliverable is described by means of a statement that sets the minimum requirements to earn a particular grade, describe the evaluation procedure, and provide delivery instructions.
- Deliverables must be accompanied by a responsibility statement in which the students state that they have not cheated. No deliverable shall be evaluated unless it is accompanied by a checked, signed, and dated copy of the corresponding responsibility statement.
- The grades that can be awarded to deliverables are A^+ ("Sobresaliente 10 con opción a Matrícula de Honor"), A ("Sobresaliente 9"), B ("Notable 8"), C ("Aprobado 6"). A deliverable that does not meet the C-Level requirements is awarded an F ("Suspenso 4" or "Suspenso 0" depending on how bad the results are).
- The number of Honours that can be awarded is limited according to the Spanish regulations. In cases in which there are more students with an A^+ than Honours can be awarded, then the Honours shall be raffled amongst those students unless they make a formal agreement and inform the lecturers.

Key ideas regarding control checks

- A control check is a test that helps a student prove that he or she is the legitimate author of his or her deliverables and that he or she has actually learnt from working on them.
- Control checks are described by means of a statement that sets a few additional requirements on a deliverable and some delivery instructions.
- Control checks are organised in laboratories in which the students can work with their computers and use the materials that they wish as long as they do not interact with anybody.
- Control checks are not expected to take the students for more than four hours.
- Students are requested to produce a deliverable by the end of every control check.
- The grade that is awarded to such deliverables is either "FAILED" or "PASSED".

June summon

- Every lesson has an associated deliverable. Hereinafter we refer to the lessons as L01-L12 and to the corresponding deliverables as D01-D12.
- Deliverable D01 is just intended to help the students get familiar with the subject and to help them recruit and set up their workgroups. Deliverables D02-D07 deal with a realistic project that must be developed in several shifts. Deliverables D08 and D10 introduce new projects and deliverables D09 and D11 are maintenance projects. Deliverable D12 is a hackathon that the students must negotiate with the lecturers.

- Students are requested to set up their work groups for deliverables D01, D02-D07, and D08-D12. Simply put: a student may work with just one group to produce deliverables D01-D12, one group to produce deliverables D01-D07 and another to produce deliverables D08-D12, one group to produce deliverable D01 and another group to produce deliverables D02-D12, or three different groups to produce deliverables D01, D02-D07, and D08-D12, respectively. No other group re-organisations are allowed.
- There are two control checks, one by the end of the autumn semester and another by the end of the spring semester.
- To pass the subject, a student must pass deliverables D02-D12 and both control checks, in which case he or she shall be awarded the rounded average of the grades awarded to the previous deliverables.
- Exceptionally, a student whose rounded average in deliverables D02-D07 is at least B, whose rounded average in deliverables D08-D12 is at least A, and has passed the spring control check, will be awarded the rounded average grade of his or her deliverables even if he or she has failed the autumn control check and/or some deliverables.
- Exceptionally, a student whose rounded average in deliverables D02-D07 is at least B, whose rounded average in deliverables D08-D12 is at least A, but has not passed the spring control check, can sit for an exceptional control check in September, even if he or she has failed the autumn control check and/or some deliverables. If he or she passes the exceptional control check, then his or her grade shall be the rounded average grade of the deliverables that he or she has delivered for the June summon; otherwise, his or her grade shall be F ("Suspenso 4" or "Suspenso 0" depending on how bad the results are).
- There is a critique after each evaluation. In cases in which a deliverable or a control check has a flaw and it can be mended during the critique in a sensible time, the grade awarded will be "C" or "PASSED", respectively.

September and December summons

- There are six deliverables in which the students must implement six projects that shall be similar in requirements and effort to the projects developed for the June summon.
- The six deliverables must be produced by the same group of students; no re-organisations of the groups are allowed.
- To pass the subject, a student must pass every deliverable and the control check, in which case he or she shall be awarded the rounded average of the grades awarded to his or her deliverables.
- There is no exception to the previous grading rule in September or December, except for the exception that allows some students to sit for an exceptional control check in September.
- There is a critique after the evaluation. In cases in which a deliverable or a control check has a flaw and it can be mended during the critique in a sensible time, the grade awarded will be "C" or "PASSED", respectively.

Lessons

The subject consists of the following lessons:

L01. Introduction
L02. Domain models
L03. Persistence models
L04. Repositories and services
L05. Views
L06. Controllers
L07. Deployment
L08. Lessons learnt
L09. Functional testing
L10. Performance testing
L11. User testing
L12. Hackathon

The lessons are taught according to the work programme that accompanies this syllabus. Please, consult it and bear in mind that it shall not change unless there is a major unexpected event that prevents the work programme to be carried out normally, e.g., a strike or a plenary meeting of a student union.

Context

Next, we describe the context of this subject in terms of degree, subjects that must have been passed before enrolling it, subjects that must be taken concurrently, and subjects that are expected to be taken later.

Degree

- Software Engineering

Previous subjects

- Foundations of Computer Programming (Freshmen's year) (*)
- Analysis and Design of Data Structures and Algorithms (Sophomore's year) (*)
- Introduction to Software Engineering and Information Systems (Sophomore's year) (*)
- Architecture and Integration of Software Systems (Sophomore's year)
- Operating Systems (Sophomore's year)

The subjects with a (*) are quintessential. No student should enrol Design and Testing unless he or she's passed them.

Co-subjects

- Software Process and Management (Junior's year)
- Requirements Engineering (Junior's year)

It’s strongly recommended that the students enrol the previous co-subjects together with Design and Testing.

Succeeding subjects

- Managing and Evolving Software Configurations (Senior’s year)
- Planning and Managing Software Projects (Senior’s year)
- Software Engineering and Professional Practice (Senior’s year) (*)
- Database Complements (Senior’s year, optional)
- Security in Computing Systems (Senior’s year, optional)

It’s strongly recommended that the students should not enrol the subjects with a (*) unless they have already passed Design and Testing.