**COURSE DESCRIPTION – ACADEMIC YEAR 2022/2023**

<table>
<thead>
<tr>
<th>Course title</th>
<th>Advanced Software Design Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course code</td>
<td>76087</td>
</tr>
<tr>
<td>Scientific sector</td>
<td>INF/01</td>
</tr>
<tr>
<td>Degree</td>
<td>Master in Software Engineering for Information Systems (LM-18)</td>
</tr>
<tr>
<td>Semester</td>
<td>1</td>
</tr>
<tr>
<td>Year</td>
<td>1</td>
</tr>
<tr>
<td>Credits</td>
<td>6</td>
</tr>
<tr>
<td>Modular</td>
<td>No</td>
</tr>
<tr>
<td><strong>Total lecturing hours</strong></td>
<td>40</td>
</tr>
<tr>
<td><strong>Total exercise hours</strong></td>
<td>20</td>
</tr>
</tbody>
</table>

**Attendance**

Not compulsory. Non-attending students must contact the lecturer at the start of the course to agree on the modalities of the independent study.

**Prerequisites**

Basic courses in Programming and Software Engineering. Familiarity with UML and software modelling. Familiarity with the basics of object-orientation and automated testing.

**Course page**

https://ole.unibz.it/

**Specific educational objectives**

The course belongs to the type caratterizzanti – discipline informatiche and is part of the Foundations in Software Engineering.

The course objective is to familiarize students with advanced tools and techniques to design a software-based system, understanding in theory and practice their applicability to achieve software requirements and the expected consequences in the implementation of each one. Students will acquire skills and competencies resulting from identifying requirements and implementing them through various design techniques, focusing on improving code maintenance and introducing extensibility in the suitable spots. Design evaluation is approached through the analysis of metrics and visualization techniques. The students are exposed to problem-solving techniques that allow the synthesis of software design solutions satisfying the system's requirements.

**Lecturer**

Martins Guerra Eduardo

**Contact LA**

SER-I 1.08, 21 Via Cassa di Risparmio, eduardo.guerra@unibz.it

**Scientific sector of lecturer**

ING-INF/05

**Teaching language**

English

**Office hours**

Tuesdays from 10h to 12h.

**Lecturing Assistant (if any)**

--

**Contact LA**

--

**Office hours LA**

--

**List of topics**

- Design Patterns Application and Interaction
- Evolutionary Design Techniques (TDD, BDD, Refactoring)
- Domain Modeling (DDD)
Learning outcomes

**Knowledge and understanding**
- D1.2 To be able to analyze and solve even complex problems in the area of Software Engineering for Information Systems with particular emphasis on the use of studies, methods, techniques and technologies of empirical evaluation;
- D1.4 To know in depth the principles, structures and use of computer systems for the automation of information systems;
- D1.5 To know the fundamentals, techniques and methods of design, customization and implementation of software to support the automation of new generation information systems for industrial production and business;

**Applying knowledge and understanding**
- D2.4 To be able to define an innovative technical solution to an application problem that meets technical, functional and organisational constraints and requirements;
- D2.5 To be able to extend and modify in an original way an existing technical solution or a formal model taking into account changed conditions, requirements and evolution of the technology;

**Making judgments**
- D3.4 To be able to reconcile the objectives of the project that are in conflict, to trade-off cost, resources, time, knowledge or risk;

**Communication skills**
- D4.5 To be able to prepare and conduct technical presentations in English;
- D4.6 To be able to interact and collaborate during the implementation of a project or research with peers and experts;

**Learning skills:**
- D5.2 To be able to keep up to date independently with developments in the most important areas of information technology;
- D5.3 In the context of a problem solving activity, to be able to extend knowledge, even if incomplete, taking into account the final objective of the project;

**Assessment**
The assessment is based on the lab assessment and the final exam. The lab assessment is composed practical activities that can be performed by the students during the course. The final exam evaluates the students' understanding of the theoretical backgrounds and the ability of solving problems.
<table>
<thead>
<tr>
<th>Assessment language</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assessment typology</td>
<td>Monocratic</td>
</tr>
<tr>
<td>Evaluation criteria and criteria for awarding marks</td>
<td>For attending students, the grade is calculated based on (i) the lab assessment (50% weight) and (ii) the final exam (50% weight). For non-attending students, if they can follow the delivery schedule for the lab assessments, the grade is calculated the same way. In exceptional cases, the grade can be calculated based only on a final exam that includes questions related to the labs’ content.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Required readings</th>
<th>The course will be based on lecture notes.</th>
</tr>
</thead>
</table>
• Open educational resources, representing alternative or supplementary materials, shall be linked to the course website. |

| Software used | Software Modelling (e.g. Argo UML, Papyrus, StarUML), Java JDK, Java Programming IDE (e.g. Eclipse, IntelliJ) |