

Course guide

270017 - PROP - Programming Projects

Last modified: 30/01/2024

Unit in charge: Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science.
Degree: BACHELOR'S DEGREE IN INFORMATICS ENGINEERING (Syllabus 2010). (Compulsory subject).
Academic year: 2023 **ECTS Credits:** 6.0 **Languages:** Catalan, Spanish

LECTURER

Coordinating lecturer: ALICIA MARIA AGENO PULIDO

Others:

Primer quadrimestre:

ALICIA MARIA AGENO PULIDO - 11, 12, 13, 14, 21, 22, 23, 31, 32, 33, 41, 42, 43
SERGIO ÁLVAREZ NAPAGAO - 23
CARLES ARNAL CASTELLO - 12, 31, 33
PABLO BLANCO DEL PRADO - 13, 21
RICARDO FERNÁNDEZ DOMENECH - 41, 43
SALVADOR MEDINA HERRERA - 42
ALEJANDRO IVÁN PAZ ORTIZ - 22, 32
MIQUEL SANCHEZ MARRE - 11

Segon quadrimestre:

ALICIA MARIA AGENO PULIDO - 11, 12, 13, 14, 41, 42, 43
CARLES ARNAL CASTELLO - 12, 13, 41
PABLO BLANCO DEL PRADO - 11, 14
RICARDO FERNÁNDEZ DOMENECH - 42, 43

PRIOR SKILLS

Students are expected to be able to:

- Solve medium-difficulty algorithmic problems from clear specifications and implement their solutions in an imperative programming language.
- Understand basic program structuring mechanisms (modularisation, encapsulation, abstract data types, classes) and apply them to small and medium-sized tasks (a small number of modules).
- Understand object-oriented programming elements (classes, objects, execution mechanisms).
- Use an imperative object-oriented language.
- Use and program data structured in this language.
- Use libraries in this language.
- Master basic error location and correction strategies for simple modules.

REQUIREMENTS

- Prerequisite EDA
- Pre-Corequisite IDI
- Pre-Corequisite IES

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:

CT2.3. To design, develop, select and evaluate computer applications, systems and services and, at the same time, ensure its reliability, security and quality in function of ethical principles and the current legislation and normative.

CT2.4. To demonstrate knowledge and capacity to apply the needed tools for storage, processing and access to the information system, even if they are web-based systems.

CT2.5. To design and evaluate person-computer interfaces which guarantee the accessibility and usability of computer systems, services and applications.

CT3.6. To demonstrate knowledge about the ethical dimension of the company: in general, the social and corporate responsibility and, concretely, the civil and professional responsibilities of the informatics engineer.

CT4.1. To identify the most adequate algorithmic solutions to solve medium difficulty problems.

CT4.2. To reason about the correction and efficiency of an algorithmic solution.

CT4.3. To demonstrate knowledge and capacity to apply the fundamental principles and the basic techniques of the intelligent systems and its practical application.

CT5.1. To choose, combine and exploit different programming paradigms, at the moment of building software, taking into account criteria like ease of development, efficiency, portability and maintainability.

CT5.2. To know, design and use efficiently the most adequate data types and data structures to solve a problem.

CT5.3. To design, write, test, refine, document and maintain code in an high level programming language to solve programming problems applying algorithmic schemas and using data structures.

CT5.4. To design the programs' architecture using techniques of object orientation, modularization and specification and implementation of abstract data types.

CT5.5. To use the tools of a software development environment to create and develop applications.

CT5.6. To demonstrate knowledge and capacity to apply the fundamental principles and basic techniques of parallel, concurrent, distributed and real-time programming.

CT6.1. To demonstrate knowledge and capacity to manage and maintain computer systems, services and applications.

CT8.6. To demonstrate the comprehension of the importance of the negotiation, effective working habits, leadership and communication skills in all the software development environments.

CT8.7. To control project versions and configurations.

Generical:

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

TEACHING METHODOLOGY

Topics will be explained in a practical way through the use of numerous examples.

The theory lessons will introduce the necessary knowledge, techniques and concepts for the course project.

Some of the laboratory classes (probably towards the beginning of the course) will be used to briefly present notations, languages, libraries and tools. In most laboratory classes, however, students take the initiative. A number of hours are available for project groups to work together and to discuss any doubts they have with the lecturer.

The two-hour theory classes and two-hour laboratory classes will take place once a week, except for the last two weeks (laboratory classes only). First week there are no laboratory classes (and there are two theory sessions).

The bulk of work on the subject will be on the project, which will have a strong algorithmic and data structure component.

The approach to implementing this project will be as follows:

- Students (15-20) from a laboratory group will form teams of four.
- Each laboratory group will be assigned a tutor.
- Within each team, a single student will be responsible for class programming and testing.

The programming language used will be Java.

LEARNING OBJECTIVES OF THE SUBJECT

1. Use previously acquired programming skills to develop a medium-sized programming project, select appropriate algorithms and data structures and build a program correctly and efficiently.
2. Organize the design and implementation work of a medium-sized team (three to four people) according to a predetermined schedule. This organization includes both the overall planning and the allocation of tasks between group members.
3. Identify program specification, design and implementation components that can be factorised and immediately resolved and use object-orientation (OO) mechanisms for effective factorisation.
4. Understand object-oriented design and programming principles and the advantages and disadvantages of adopting this programming paradigm.
5. Use class and inheritance mechanisms to avoid redundancy and capture specification, design and implementation abstractions. They should also understand the advantages of this use of object orientation in terms of reliability, modifiability, portability, reusability and productivity.
6. Deal effectively with anomalous program situations using, if necessary, the exception mechanism provided by the programming language. Test the program systematically and exhaustively.
7. Design a reasonably usable and effective user interface for a program using a GUI library.
8. Develop (in groups) a medium-sized programming project, according to a pre-defined development plan and architecture.

STUDY LOAD

Type	Hours	Percentage
Hours large group	30,0	20.00
Hours small group	30,0	20.00
Guided activities	6,0	4.00
Self study	84,0	56.00

Total learning time: 150 h

CONTENTS

ISE review: Specification, class diagrams and use cases.

Description:

Life cycles. Project definition. Requirements. Functional specification. Initial domain modelling. UML class (domain) diagram. Use case diagram. Textual explanation. Case study.

Introduction to Java and OO concepts in Java.

Description:

Object-oriented (OO) specification, design and programming. Classes, objects, attributes, methods and relationships. Encapsulation and hiding, abstraction and classification, inheritance. Polymorphism, static and dynamic binding. Other relationships: association, aggregation, dependency.

Design and implementation in Java of the Three-tier architecture

Description:

The three-tier architecture design paradigm and its Java implementation



Program testing

Description:

Verification. Program testing and debugging. Component testing. Testing and integration strategies. Drivers and stubs/mocks. Test planning. Scaffolding and regression tests. Testing help tools (JUnit). Handling errors and exceptions.

Design patterns in Java

Description:

Some of the classical design patterns will be explained and implemented in Java (decorator, state, singleton, etc)

Basic Concepts of Interface design in Java

Description:

How to design a simple graphical user interface using Java standard libraries and applying the principles of 3-layer architecture

Software development toolchain

Description:

There will be an overview of the typical toolchain to develop software: version control (Git), testing integration (JUnit)

ACTIVITIES

Software life cycle review

Description:

Review of software engineering concepts and UML.
Practical example of (part of) a first delivery.

Specific objectives:

3, 4, 5

Full-or-part-time: 7h

Theory classes: 4h

Self study: 3h



Project description and team creation

Description:

Description of project deliveries. Team creation for practical sessions.

Specific objectives:

2, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 2h

Laboratory classes: 2h

Introduction to Java and Basic OO concepts in Java

Description:

Topic 2 development.

Specific objectives:

4, 5, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 16h

Theory classes: 8h

Self study: 8h

Design and implementation in Java of the Three-tier architecture

Description:

Topic 3 development.

Specific objectives:

3, 4, 5, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 6h

Theory classes: 4h

Self study: 2h

Program debugging

Description:

Topic 4 development.

Specific objectives:

6, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 12h

Theory classes: 6h

Self study: 6h

Design patterns in Java

Description:

Some classic design patterns will be introduced (or reviewed if already known from the ISE course) and implemented in Java

Specific objectives:

4, 5

Full-or-part-time: 8h

Theory classes: 4h

Self study: 4h

Java: implementing interfaces

Description:

Topic 6 development.

Specific objectives:

7, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 8h

Theory classes: 4h

Self study: 4h

First delivery of the project

Description:

Implementation principles and design. Students will fully implement the domain model and the basic algorithmic kernel for the project and provide test suites for the implemented part.

Specific objectives:

1, 2, 3, 4, 5, 6, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Second delivery of the project

Description:

Intermediate project delivery. The documentation should include the complete diagram of classes (UML) of the project.

Specific objectives:

1, 2, 3, 4, 5, 7, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Third delivery of the project

Description:

Final project delivery. The documentation should include the full implementation of the project and the test suites as appropriate, and the user manual. Each team will also give a presentation of their project.

Specific objectives:

1, 2, 3, 4, 5, 6, 7, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 2h

Guided activities: 2h

Supervision of first assignment

Description:

Laboratory classes related to the first assignment.

Specific objectives:

1, 2, 3, 4, 5, 6, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 45h

Laboratory classes: 16h

Self study: 29h

Supervision of second and third assignments

Description:

Laboratory classes related to the second and third assignments.

Specific objectives:

1, 2, 3, 4, 5, 6, 7, 8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 41h 48m

Laboratory classes: 12h

Self study: 29h 48m

Third assignment presentation

Description:

Presentation of the third assignment and interview.

Specific objectives:

8

Related competencies :

G1. ENTREPRENEURSHIP AND INNOVATION: to know and understand the organization of a company and the sciences which govern its activity; capacity to understand the labour rules and the relation between planning, industrial and business strategies, quality and benefit. To develop creativity, entrepreneur spirit and innovation tendency.

G5. TEAMWORK: to be capable to work as a team member, being just one more member or performing management tasks, with the finality of contributing to develop projects in a pragmatic way and with responsibility sense; to assume compromises taking into account the available resources.

Full-or-part-time: 2h 12m

Guided activities: 2h 12m

GRADING SYSTEM

Final mark: Project_Mark

Project mark: $(0.40 * \text{Delivery1} + 0.15 * \text{Delivery2} + 0.45 * \text{Delivery3}) * \text{FT}$

Given that the project is team work in which all team members (and only team members) must participate, the final project mark will be weighted according to a work factor (FT). This work factor is a score (0

Transversal competency in teamwork: Assessment will be based on a simple formula, available to the students, with the group tutor scoring competency aspects for each team member.

Transversal competency in entrepreneurship and innovation: Assessment is performed considering the initiative of the team in the two moments in which they have to make their own decisions, not predetermined in the subject: the extra functionalities of the project and the algorithms and data structures with which the main functionality is going to be solved. Therefore, the evaluation is a combination of two criteria:

- 1) Depending on the optional functionalities proposed versus the optional functionalities implemented
- 2) Depending on the initiative of the team when choosing algorithms and DSs to implement the main functionalities

BIBLIOGRAPHY

Basic:

- Budd, T. An introduction to object-oriented programming. 3rd ed. Addison-Wesley, 2002. ISBN 0201760312.
- Binder, R.V. Testing object-oriented systems: models, patterns and tools. Addison-Wesley, 2000. ISBN 9780321700674.
- Arnold, K.; Gosling, J.; Holmes, D. The Java programming language. 4th ed. Upper Saddle River: Addison-Wesley, 2006. ISBN 0321349806.
- Peralta, A.J.; Rodríguez, H. Enginyeria del software: programació orientada a objectes. Edicions UPC, 1994. ISBN 8476533748.
- Tahcheiv, P. [et al.]. JUnit in action. 2nd ed. Manning, 2011. ISBN 9781935182023.
- Weisfeld, M. The object-oriented thought process. Fifth edition. Boston: Addison-Wesley Professional, 2019. ISBN 9780135181966.
- Gamma, Erich. Design patterns : elements of reusable object-oriented software. Reading, MA [etc.]: Addison-Wesley, cop. 1995. ISBN 0201633612.
- Sarcar, V. Java design patterns: a hands-on experience with real-world examples. 3rd ed. Berkeley, California: Apress L.P., 2022. ISBN 9781484279717.

Complementary:

- Meyer, B. Object-oriented software construction. 2nd ed. Prentice Hall, 1997. ISBN 0136291554.
- Booch, G.; Rumbaugh, J.; Jacobson, I. The unified modeling language user guide. 2nd ed. Addison-Wesley, 2005. ISBN 0321267974.
- Rumbaugh, J.; Jacobson, I.; Booch, G. The unified modeling language reference manual. 2nd ed. Addison-Wesley, 2005. ISBN 0321245628.
- Lee, R.C.; Tepfenhart, W.M. UML and C++: a practical guide to object-oriented development. 2nd ed. Prentice Hall, 2001. ISBN 0130290408.
- Pressman, R.S.; Maxim, B.R. Software engineering: a practitioner's approach. 8th ed. New York: McGraw Hill Higher Education, 2020. ISBN 9781260548006.
- Cooper, A.; Reimann, R.; Cronin, D.; Noessel, C. About face: the essentials of interaction design. 4th ed. Wiley, 2014. ISBN 9781118766576.
- Thomas, David. Pragmatic Programmer: your journey to mastery. 20th ed. Addison Wesley, 2020. ISBN 9780135957059.
- Martin, Robert C. Clean architecture : a craftsman's guide to software structure and design. Prentice Hall, [2018]. ISBN 9780134494166.

RESOURCES

Hyperlink:

- <https://www.uml.org/>- <https://www.oracle.com/java/index.html>