



Universitat de Lleida

DEGREE CURRICULUM  
**GESTIÓ I MILLORA DE LA  
QUALITAT**

Academic year 2013-14

## Subject's general information

|   |  |
|---|--|
| <b>Subject name</b>   | GESTIÓ I MILLORA DE LA QUALITAT  |
| <b>Code</b>   | 102053   |
| <b>Semester</b>   | 2n Q Avaluació Continuada  |
| <b>Typology</b>   | Obligatòria  |
| <b>ECTS credits</b>   | 6  |
| <b>Theoretical credits</b>  | 3  |
| <b>Practical credits</b>  | 3  |
| <b>Department</b>   | Informàtica i Enginyeria Industrial  |
| <b>Teaching load distribution between lectures and independent student work</b> | <ul style="list-style-type: none"> <li>- Autonomous work (60%) = 90h</li> <li>- Classroom work (40%) = 60h</li> <li>* Training activities (90%) = 54h</li> <li>* Assessment activities (10%) = 6h</li> </ul> |
| <b>Important information on data processing</b>                                 | Consult <a href="#">this link</a> for more information.  |
| <b>Language</b>   | Spanish 80%<br>English 20%   |
| <b>Distribution of credits</b>  | 6 ECTS = 150 h   |
| <b>Office and hour of attention</b>   | <p>Despaxt 3.18.</p> <p>The hour of attention is to be determined and will be communicated to students in the presentation of the subject.</p>   |

Esteve Brugulat Cuñé  
Roberto García González

## Learning objectives

Not specified

## Competences

University of Lleida strategic competences

- Master Information and Communication Technologies.
- Master a foreign language.

Degree-specific competences

- Ability to develop, maintain and evaluate software services and systems which satisfy all the user's requirements and perform in a reliable and efficient way, are cost effective to develop and maintain, and comply with quality regulations specified in the theories, principles, methods and practices of Software Engineering.
- Ability to identify, evaluate and manage potential associated risks which could appear.

## Subject contents

Theory:

- Quality: definition and standards
  1. Introduction to the concept of quality
  2. Standards and quality models
- Software analysis and improvement
  1. Static analysis
    1. Software Metrics
    2. Codereview
      1. Pair-Programming
      2. Tool-assisted code review
      3. Style checkers
    3. Bugs analysis
      1. Techniques and patterns
      2. Bugcheckers
  2. Dynamic analysis
    1. Software testing
      1. Continuous Integration
      2. Analysis of coverage
    2. Performance analysis, benchmarking, profiling

Tools and practices:

- Error handling
- Static analysis tools and debugging
- Dynamic analysis tools
- Automatic Test

## Development plan

Week 1: Introduction and concepts  
Week 2: Introduction to models and quality standards  
Week 3: Product Quality  
Week 4: Process Quality  
Week 5: Software metrics for project and process  
Week 6: Software metrics for product  
Week 7: Code review and static analysis  
Week 8: Defense for the first midterm practices  
Week 9: Midterm exam 10th (12:00 to 14:00 h) P 2.04  
Week 10: Working with metrics and static analysis software  
Week 11: Working with metrics and static analysis software  
Week 12: Debugging and Error Handling  
Week 13: Debugging and Error Handling  
Week 14: Quality and Software testing  
Week 15: Defense for the second midterm practices  
Week 16: Exam 11th (9:00 to 11:00 h) P 2.04  
Week 20: Examination Day 30 (12:00 to 14:00 h) P 2.01

## Evaluation

- Theory:
  - Midterm 1 = 30%
  - Midterm 2 = 30%
- Laboratory:
  - Practical assignment 1 = 20%
  - Practical assignment 2 = 20%

## Bibliography

- Galin, D. (2003), *Software Quality Assurance: From Theory to Implementation*, Prentice Hall.
- García Rubio, F. O.; Garzías Parra, J.; Genero Bocco, M. F. & Piattini Velthuis, M. G. (2008), *Medición y estimación del software. Técnicas y métodos para mejorar la calidad y la productividad*, RA-MA EDITORIAL.
- Piattini Velthuis, M. G.; García Rubio, F. O. & Caballero Muñoz-Reja, I. (2006), *Calidad de Sistemas Informáticos*, Ra-Ma Editorial, S.A.