DEGREE CURRICULUM

ENTERPRISE SOFTWARE ARCHITECTURE

Coordination: GARCIA GONZALEZ, ROBERTO

Academic year 2015-16

Subject's general information

Subject name	ENTERPRISE SOFTWARE ARCHITECTURE
Code	102029
Semester	2nd
Typology	Mandatory
ECTS credits	9
Groups	1
Theoretical credits	4.5
Practical credits	4.5
Coordination	GARCIA GONZALEZ, ROBERTO
Office and hour of attention	To be agreed, contact rgil@diei.udl.cat, rgarcia@diei.udl.cat or jvirgili@diei.udl.cat
Department	Informàtica i Enginyeria Industrial
Teaching load distribution between lectures and independent student work	Total load: 225h - Lectures (40%) = 90h - Independent student work (60%) = 135h
Modality	Presencial
Important information on data processing	Consult this link for more information.
Language	English
Degree	Degree in Computer Engineering
Distribution of credits	GARCIA GONZALEZ, ROBERTO (2,1) GIL IRANZO, ROSA MARIA (2,7) VIRGILI GOMÀ, JORDI (6)
Office and hour of attention	To be agreed, contact rgil@diei.udl.cat, rgarcia@diei.udl.cat or jvirgili@diei.udl.cat
E-mail addresses	rgarcia@diei.udl.cat rgil@diei.udl.cat jvirgili@diei.udl.cat

GARCIA GONZALEZ, ROBERTO
GIL IRANZO, ROSA MARIA
VIRGILI GOMÀ, JORDI

Subject's extra information

To properly follow this subject, it is recommended to have consolidated software engineering and Web applications development skills like those presented in the Software Engineering and Web Systems and Technologies courses.

Learning objectives

- To present Web engineering and its fundamental applications patterns and architectures in the context of an agile methodology and Behaviour Driven Development.
- To have a global vision of the existent technologies to implement enterprise applications using the previous patterns and architectures, especially Java, Javascript and the Web.
- To put into practice the previous concepts and technologies through the development of a Web application project using Java (Spring) and Javascript (AngularJS), following an agile methodology and behaviour driven development (BDD).

Significant competences

Strategic Competences

CT2. Mastering a foreign language, especially English.

CT3. Training Experience in the use of the new technologies and the information and communication technologies.

Cross-disciplinary Competences

EPS11. Capacity to understand the needs of the user expressed in a no technical language.

Specific Competences

GII-TI1. Capacity to understand the environment and needs of an organisation in the field of the information and communication technologies.

GII-TI2. Capacity to choose, design, deploy, integrate, evaluate, build, manage, explode and keep the hardware, software and network technologies inside the cost and quality requirements.

GII-TI5. Capacity to select, deploy, integrate and manage systems of information that satisfy the needs of the organisation, within the cost and quality requirements.

GII-TI6. Capacity to conceive systems, applications and services based in network technologies, including Internet, web, e-commerce, multimedia, interactive services and mobile computation.

Subject contents

1. Agile Web Applications Development Process
    1. Behaviour Driven Development (BDD) with Cucumber Java
    2. Scrum using ZenHub
    3. Teamwork coordination using Version Control
    4. Continuous Integration using TravisCI
    5. Automated deployment using Heroku

2. Application Architecture Patterns
    1. Introduction and layered architecture
    2. Patterns in the domain layer
    3. Patterns in the integration layer
    4. Patterns in the presentation layer
    5. Technologies for pattern application
3. Web Application Implementation
    1. Server side: Java (Spring)
    2. Client side: Javascript (AngularJS)

Methodology

The methodology is based on a Project Based Learning approach where an enterprise software application is developed, focusing on Web-based applications. The course starts with a review of Web development frameworks from an industrial point of view analysing the number of job offers, StackOverflow questions, LinkedIn mentions, etc. Based on these dimensions, Spring is selected for the server side and AngularJS for the client one.

The first weeks of the course focus on fundamental aspects of software development, including project management ideas and development process concepts. Software architecture, specification, analysis and design are presented from a client/server and Web perspective, guided by enterprise application patterns. These concepts are the starting point for the students project development.

The students select a project, under professors guidance, and complete a first deliverable where they apply these fundamental concepts to specify, analyse and design their project.

The rest of the course focuses on the implementation of the defined project and the required concepts related to agile software development, Spring or Angular are introduced as needed to complete the development of the project.

Development plan

Weeks 1 - 2: Frameworks and Methodologies (Spring, AngularJS, SCRUM, BDD,...). Project definition.

Weeks 3 - 4: 1st Development Sprint

1st Spring Review

Weeks 5 - 6: 2nd Development Sprint

2nd Spring Review and Deliverable

Weeks 7 - 8: 3rd Development Sprint

3rd Spring Review

Week 9: 1st Midterm Exam

Weeks 10 - 11: 4th Development Sprint

4th Spring Review and Deliverable

Weeks 12 – 13: 5th Development Sprint

5th Spring Review

Weeks 14 - 15: 6th Development Sprint

6th Spring Review and Deliverable

Weeks 16 - 17: 2nd Midterm Exam

Evaluation

The evaluation is fundamentally based on the development of a project following an agile methodology following 6 sprints and 3 deliverables:

- Sprints 1 and 2, 1st Deliverable: 20% grade
- Sprints 3 and 4, 2nd Deliverable: 20% grade
- Sprints 5 and 6, 3rd Deliverable: 20% grade

For each sprint (review), the performance in developing the assigned tasks will be evaluated, which will be contextualised for each corresponding deliverable that takes into account aslo the developed product. The grade will combine the outcomes for each deliverable together with its associated sprints.

The evaluation is complemented with two exams performed individually:

- 1st Midterm Exam: 20% grade, about fundamental aspects of the frameworks and methodologies used so far.
- 2nd Midterm Exam: 20% grade, about the development experience of the project as a whole.

Bibliography

Main References

- Fowler, M.; Rice, D. (2003). Patterns of Enterprise Application Architecture. Addison-Wesley.
- Amuthan, G. (2014). Spring MVC: Beginner's guide Birmingham. Packt Publishing.
    - Electronic Version[1]: https://www.dawsonera.com/abstract/9781783284887
- Walls, C. (2014). Spring in Action, 4th Edition. Manning.
- Walmsley, P. (2007). XQuery. Sebastopol, CA: O'Reilly Media.
- McLaughlin, B.; Edelson, J. (2006). Java and XML (3rd edition). O'Reilly.
    - Electronic Version[1]: http://proquest.safaribooksonline.com/059610149X

Additional References

- XML Quick Reference, http://www.mulberrytech.com/quickref/XMLquickref.pdf
- Larman, C. (2002). UML y Patrones. Prentice-Hall (segunda edición).
- Larman, C. (2005). Applying UML and Patterns. Prentice-Hall (third edition).
- Cummins, F. (2002). Enterprise Integration: An Architecture for Enterprise Application and System Integration. Wiley Publishing.
- Sommerlad, P.; Stal, M. (1996). Pattern-Oriented Software Architecture, Volume 1: A System of Patterns. John Wiley & Sons.
- Gamma, E.; Helm, R.; Jonson, R.; Vlissides, J. (2003). Patrones de Diseño. Elementos de software orientado a objetos reutilizable. Addison-Wesley.
- Gutiérrez Gallardo, Juan Diego. (2005). Manual imprescindible de XML. Anaya Multimedia.
- Hunter, D., Rafter, J., Fawcett, J., Vlist, E. van der, Ayers, D., Duckett, J., Watt, A., et al. (2007). Beginning XML, 4th Edition. Indianapolis, IN: Wrox.
- Ray, E.T. (2003) Learning XML, 2nd Edition. Sebastopol, CA: O'Reilly Media.
- Harold, E. R.; Jeans, W. S. (2004). XML in a Nutshell, 3rd Edition. Sebastopol, CA: O'Reilly Media.
- Walmsley, P. (2002). Definitive XML Schema. Upper Saddle River, NJ: Prentice Hall.
- Vlist, E. van der. (2002). XML Schema: The W3C's Object-Oriented Descriptions for XML. Sebastopol, CA: O'Reilly Media.
- Brundage, M. (2004). XQuery: the XML query language. Boston, MA: Addison-Wesley Professional.
- Martín Quetglás, Gregorio. (2005). Curso de XML : introducción al lenguaje de la Web. Pearson educación.
- Harold, Elliotte Rusty. (2005). XML. Anaya Multimedia.
- Esposito, Dino. (2003). Programación en XML para Microsoft .NET. McGraw-Hill.
- Keogh, Jim. (2003). J2EE : manual de referencia. McGraw-Hill.
- Cauldwell, P.; Charla, R.; Chopra, V. (2002). Servicios Web XML. Anaya Multimedia.
- McGovern, J.; Tyagi, S.; Stevens, M. E.; Mathew, S. (2003). Java Web Services Architecture. Morgan

Kaufmann.
- Monson-Haefel, R. (2004). J2EE Web services. Adisson-Wesley.
- Newcomer, E.; Lomow, G. (2004). Understanding SOA with web services. Addison-Wesley.
    - Electronic Version[1]: http://proquest.safaribooksonline.com/0321180860

[1] This book is accessible from the Universitat de Lleida network using this link