

SYLLABUS

1. Course title:

Software Development

2. Code:

RI302

3. Cycle of study:

1

4. ECTS credits:

6

5. Type of course: Mandatory Elective**6. Prerequisites:**

[RI202] Object-oriented programming, [RI207] Databases

7. Class restrictions:**8. Duration / semester:**

1

6

9. Weekly contact hours:

9.1. Lectures:

3

9.2. Seminars:

1

9.3. Laboratory/Practice classes:

1

10. Faculty:

Faculty of Electrical Engineering

11. Department/study program:

Electrical Engineering and Computer Science

12. Lecturer:

PhD. Emir Mešković, assistant prof.

13. Lecturer's e-mail:

emir.meskovic@untz.ba

14. Web site:

--

15. Course aims:

The aim of the course is to learn students to develop programs for the Java platform, be familiar with the UML class and sequence diagrams notation, understand the concepts of design patterns, be familiar with the methodologies used in software testing, design and development , and be able to apply object-relational mapping to communicate with a database management system utilizing the JPA2 specification.

16. Learning outcomes:

After completing the course, the students will: be able to develop programs for the Java platform, be familiar with the UML class and sequence diagrams notation, understand the concepts of design patterns, be familiar with the methodologies used in software testing, design and development , and be able to apply object-relational mapping to communicate with a database management system utilizing the JPA2 specification.

17. Course content:

Java for C++ programmers: JVM introduction, primitive types, class types, interfaces, packages, exceptions handling, generic classes. Software testing. Unit testing. UML notation: class, sequence, object, package, activity, component diagrams. Classical and agile software development methods . Design patterns: singleton, template, adapter, factory, composite, visitor, decorator. Object relational mapping: JPA2. Code refactoring.

18. Learning methods:

Lectures and auditive exercises - a PowerPoint presentation is displayed, panels and chalkboards are used for additional explanations, typical examples of Java code are presented.

Laboratory exercises - student's attendance is compulsory and active participation in teaching. By using a selected IDE, the solutions of the assignments that cover topics previously done within lectures and auditive exercises are practically tested in laboratory exercises.

19. Assessment methods:

The final grade is based on the continuous assessments, which are performed throughout the semester with quizzes, homework assignment, and the final exam. The final exam is done practically in the form of a project that implies the development of a software system using the Java programming language and the JPA2 specification for communication with the database.

20. Assessment components:

The examination grade is based on the total number of points the student has obtained by fulfilling the pre-exam examinations and passing the final exam. Students can achieve a maximum of 100 points according to the following scale:

Homeworks	25
First midterm exam	20
Second midterm exam	20
Total pre - exam	65
Final exam	35
Total	100

21. Required reading list:

C. Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Addison Wesley, 2004.

M. Fowler, UML Distilled: A Brief Guide to the Standard Object Modeling Language, Addison Wesley, 2003

22. Web sources:**23. Applicable starting from the academic year:**

2016/2017

24. Adopted in the Faculty/Academy session:

04.04.2016