

Team software project – one of the steps students take towards career in software industry

Sanja Čandrić, sanjac@inf.uniri.hr

Martina Ašenbrener Katić, [masenbrener@inf.uniri.hr](mailto:massenbrener@inf.uniri.hr)

Presentation outline

- Introduction
- Industry expectations and requirements
- Software engineering course
- Software development project
- Evaluation process
- Students questionnaire
- Conclusion
- Future work



Introduction

- Job offers – employers require/prefer previous development experience or at least teamwork experience
- 10 years od Department of informatics – discussions with software industry representatives (round tables)
- Team collaboration in large software development
- Students prefer practical assignments
- Based on feedback during and after their study



Motivation for team software project

- To simulate real-world situation in educational environment
- Create opportunities for students to experience real life software development
- Software engineering course, Department of Informatics, UNIRI
- Students actively participate in team assignments: share knowledge, learn, discuss, explore
- Students get feedback from their peers



Software engineering course

- 1st year of graduate study of informatics
- 6 ECTS
- Mandatory for BI students (14), Elective for ICS students (2)
- 16 students in winter semester 2017./2018.

- In this course:
 - To define concepts from the field of software engineering and software systems development
 - To experience professional team software development within given time and budget
 - To develop software based on the project



Grading points

- Software project with presentation 60 points (threshold 40%)
- Theory exam 30 points (threshold 40%)
- Quizzes 5 points
- Active participation 5 points



Software engineering course - topics

- Ethics in software engineering
- Management of user requirements
- Team organization
- Planning and project control
- Risk management
- Cost estimation
- Software tools, CASE tools
- UI/UE
- Tools for team software development

Students already have previous knowledge about tools and methods for software development (Introduction to SE, Data modelling, Process modelling, Programming, ...)

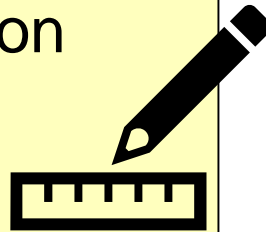
Assignments & software artefacts

- Software specification
- Process model
- Data model
- ToDo list
- Effort estimation
- Roles and responsibilities
- Software
- User manual and instructions
- Presentation for the peers
- Analysis/comparison of the plan and reality



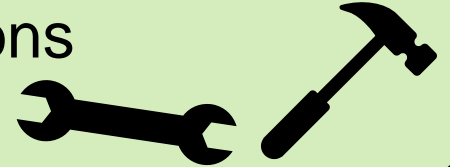
1

- Software specification
- Process model
- Data model



3

- Software
- User manual and instructions



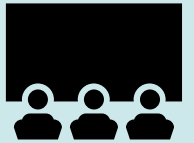
2

- ToDo list
- Effort estimation
- Roles and responsibilities



4

- Presentation for the peers
- Analysis/comparison plan vs. reality



5

- Evaluation – 4 levels



Specification

- General instructions (general requirements) on 2 applications
- 2 teams – dorm management (students applications, rent for students, rent for visiting professors)
- 2 teams – tire shop (sale and replacement with reservations)
- Students add some features (analysis of the market, creative approach, ...)
- Promise to the customer
- Different specifications from 2 teams!
- Process model
- Data model



General instructions - example

- Student dorm has several buildings. Rooms are 2- 3- and 4-bed. There are some single use bedrooms available only for visiting teachers, these rent for the selected number of days based on a daily fee. Some single bed rooms are apartments (with kitchen and private bathroom). Some multi-bed rooms have private bathroom as well.
- For students rooms are available only during the entire year, except for finishing students who can rent it for a shorter period.
- Payment is monthly based. Receipt is issued to each student, except for students which have a free stay – for all of them a single receipt is issued and payed by Student Support Centre.
- During the application process for the dorm, students register in the application, enter their data and average grade. In the end of application period a list is formed and rooms are assigned by their preferences.



ToDo list

- Activities extracted from the specification
- Roles and responsibilities
- Effort estimation
- Workload evenly shared, complexity too
- Execution plan
- Dependencies
- Deadline for each activity (set of activities)



Software development, presentation & test

- Students choose development environment
- Specification & models → development
- Testing
- User manual

- Presentation to the peers / customers

- Comparison of estimated effort and deadline for the activities from the plan and their realization – did not affect the grade



Evaluation

- Grading the team is OK
- How to grade team members individually?
- 4 levels of assessment: teacher, team, self, peers

Teacher

- Specification compared with the final software
- Responsibilities are taken into account
- Workload share
- Correctness and complexity
- User experience and interface design



Evaluation

Team

- A variant of Michaelsen method
 - assign other members of their team a score. A number of points are divided among teammates (without evaluating herself or himself)
- Precise contribution of each team member
- What could he/she do to contribute more

Self

- My contribution to the team
- How could I contribute more
- How would I grade myself

Peers

- Test software to the specification (teams switch in testing)
- General impression and details, comments

Questionnaire

- 33 questions, Likert scale
- 5 more open questions
- 11 students filled in the questionnaire

- Software development as main assignment 4,1
- You knew what was expected from you 4,2
- Motivation for software development 4
- Team chooses members 3,7
- Team chose development environment 4,5
- 4th year student able to use any environment 4,1
- Team approach 4,5



Questionnaire

- Lack of time 3,6
- Responsibilities from ToDo didn't change 3,3
- Responsible person was really in charge 3,7
- Easy to assess team member contribution 4,1
- This course was useful for me 4,1
- I invested a lot of effort 4
- ECTS points are correctly set 4,2
- Project should be kept for the next years 4



Conclusion

- Students' opinion received by questionnaire and informal discussion
 - Students participated very actively
 - Students were satisfied with the project
 - Insight to problems and challenges of team development
 - Feedback from other students (team members and peers)
 - Learning by doing
- Students are aware that team software project brings them closer to their future career in software industry
- Students' opinion regarding assignments and course organization is positive
- Interesting:
 - Teacher: not easy to assess individual student's contribution
 - Student: easy to evaluate team member's contribution



Conclusion

- Main goal – to simulate team software development environment
- Challenging for students and teachers
- Students built a reference for their future job applications and interviews
- To enforce the connection with the industry – future work



Future work

- To emphasize the connection with software industry
- If possible, to include industry representatives in the development process
 - Definition of the field of application
 - Suggesting / choosing development tools
 - Team mentors – mentoring team activities
 - Not as team members!
- To improve the assessment process – participation in the jury of experts
- To improve the self-assessment process – recognize their mistakes and how to improve
- Students will become more familiar with business environment and requirements which will be set for them in team software industry development
- Additional assignments – not necessary

- Thank you for your attention!
- Questions?

