EGE UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT
BIM447 SOFTWARE ENGINEERING

Fall 2016
TERM PROJECT ASSIGNMENT

Content of this document:

Objectives

The assignment's objective is to gain experience in software engineering practices.

Project

- The project will be chosen by each group from one of the project descriptions in the attached “Project Proposals” document. The names and numbers of group members along with the project topic that is chosen **must be** submitted under the heading **Projects Groups and Topics Assignment** on the **moodle** page of the course by **06.10.2016** by each team. The project will be undertaken in teams consisting of **three or four** members. **All the members must belong to the same lab group.**

Expected Work

The teams should apply waterfall process model. They are free to use any tool or report format that may be preferred for its being very suitable for the teams’ specific task. For example, teams may add a *data flow diagram* or even a *flowchart* to their analysis or design if they think they are necessary to get the job done. “*Getting the job done*” is however a strict requirement and it is expected from the team as a pay-back of the freedom that they have been granted. You will frequently find the instructors act as customers who even do not know what exactly they should expect from the project, or in general from software or computers. On the other hand, the team should be motivated to discover and demonstrate what the project may do for the end users. The teams must perform all kinds of research including moving to the field and performing interviews with the experts and potential users in the analysis phase of the project. Teams should submit either videos of the interviews with domain experts and potential users in CDs, or the telephone numbers and email addresses of the domain experts along with the requirements taken from them. Competition and *confidentiality between the teams* is encouraged as well as cooperation *and participation within the teams*.

Assessment of Project

Your work will be graded mainly based on your domain analysis and design performance. Therefore, you are strongly advised to explore the domain deeply and come up with a competitive design that provides the best possible automation. *Getting the job done,* although crucial, is not enough for good grades. A team with a superior design will receive better grades. On the other hand, if you cannot come up with profound solutions to the key problems, you are not likely to get very good grades, no matter how beautiful (or long) your reports may be.

- Every report must include a proper introduction and conclusion, *relevant to the report content.*
- Report format and neatness must be acceptable as a *professional artifact*.

The first work expected from the teams are:

PROJECT PLAN (31.10.2016)

1) Gantt Chart of the Project Plan
2) Network Diagram of the Project Plan
3) Division of labor within the team (who is going to do what)
4) Risk analysis document (including security and reliability risks where applicable)
5) Any other schema or other forms of description that you think is necessary

REQUIREMENTS (ANALYSIS) REPORT (21.11.2016)

1) Introduction
2) Identification of Viewpoints
    - Principal Viewpoints of the System
    - Viewpoint Hierarchy Diagram
    - Requirements of each Viewpoint

3) Requirements Definition (considering functionality)
    - Functional Requirements
    - Non-functional Requirements
    - Domain Requirements

** If any of #4 is applicable to your project:

4) Requirements Definition (considering lifetime)
    - Volatile Requirements
    - Enduring Requirements
5) Requirements interview with domain experts (you should either record the interview on CD or put a section for the list of requirements taken from each expert along with his/her email and telephone number into your report)
6) Requirements Prioritization and Negotiation
7) Requirements Traceability Matrix
8) Fully Dressed Use Cases of Main Scenarios
9) Domain Model as a UML class diagram
10) Any other schema or other forms of description that you think is necessary

REQUIREMENTS PRESENTATION (28.11.2016)

The teams are required to present the results of their analysis efforts as a short Power-Point slide-show.

ARCHITECTURAL MODEL (05.12.2016)

- Define the architecture of your system in order to indicate the major components of your system (subsystems) and the relations between them.
- Explain why you did not choose other possible candidate architectures for your system.
- Show your architecture as a block diagram and a class diagram.

- Indicate if you have employed any generic architecture, architectural style or a reference model.
- Be realistic in determining your subsystems. Note that you will need to refer to this architectural partitioning in the project plan while arranging the division of labor between your developers.
- Clarify all your entities with a legend or dictionary where necessary.

DESIGN DOCUMENT (19.12.2016)

1. Introduction (purpose of the design document, organization of the document)
2. System Sequence Diagram of each use case
3. Sequence Diagram of main use cases
4. Design Class Diagram
5. Glossary (Explain your terms in a dictionary)
6. Any other schema or other forms of description that you think is necessary
7. Implementation of a main use case
8. Tests of the implemented use case
9. Requirements validation with the domain expert (you should either record the interview on CD, or put a section for the list of requirements validated and not validated by each expert along with his/her email and telephone number into your report)
10. Updated requirements document
11. The criteria taken into consideration during the design of the UI
12. Conclusion

DESIGN PRESENTATION (26.12.2016)

The teams are required to present the results of their design efforts and prototype implementation (demo) as a short (20 min.) presentation and demonstration. We expect to be able to understand your system design clearly from your presentation, so you should include anything you think is necessary to achieve this. You are recommended to present your design, referencing your system architecture. You may change any part of your previous choices as long as you explain in your report what has been changed and why.

The prototype should cover the main use cases of your design and indicate how the user interface will function. You are expected to implement all of the UI and only the use case that is indicated in the project topics document. The UI implemented must give a message for all other use cases that are not implemented.

Team Work

- The project will be undertaken in teams consisting of **three (3) or four (4)** members formed by the students themselves. All the members must belong to the same lab group. Group and/or team changes are not tolerated.

- Teams are encouraged to decide on their *division of labor* and *decision taking mechanism* as soon as possible. They may elect a team leader or may choose to employ more democratic ways of taking decisions.

- When the reports are handed in, they should be accompanied by a statement, signed by the team members, stating the percentage contribution of each team member. For example, in an equally contributing team of five, all contributions would be labeled 20%. Unequal contributions should be clearly stated and agreed so that marks can be adjusted. If there are any problems on reaching agreement, please discuss this with your lab instructor as soon as possible.

- Group work should be the result of collaboration only within the group.

Late Submissions

Late work will receive the following penalty:

- 1 day late – 10% penalty; more than 1 day late – 10% per day penalty.