EGE UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT
BIM447 SOFTWARE ENGINEERING

Fall 2021
TERM PROJECT ASSIGNMENT

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 project topic that is chosen **must be** submitted under the heading **Project Topic Selection** on the **EGEDERS** page of the course by **20.10.2021** by each team. The project will be undertaken in teams consisting of **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 the quality of your prototype. 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 (09.11.2021)

1) Introduction
2) Gantt Chart of the Project Plan
3) Network Diagram of the Project Plan
4) Division of labor within the team (Who is going to do which tasks?)
5) Risk analysis document (Risk Identification and Risk Analysis steps)
6) Conclusion

* It should be clear in the project plan (in gantt chart and network diagram) that which tasks are available, when does each task start and end, what are the dependencies between tasks, who is going to do which tasks? If your gantt chart or network diagram are capable of showing division of labor, you do not need to put step 4 in your report. Otherwise, you need to define division of labor in step 4 as an extra item.

REQUIREMENTS (ANALYSIS) REPORT (30.11.2021)

1) Introduction
2) Identification of Viewpoints
   - Principal Viewpoints of the System
   - Description of each Viewpoint
   - Viewpoint Hierarchy Diagram
3) Requirements Definition
   - Definition of requirements of each viewpoint
4) Requirements Classification (considering functionality)
   - Functional Requirements
   - Non-functional Requirements
   - Domain Requirements
5) Requirements Classification (considering lifetime)
   - Volatile Requirements
   - Enduring Requirements
6) [*OPTIONAL*] Requirements interview (online) with stakeholders (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)
7) Requirements Prioritization and Negotiation
8) Requirements Traceability Matrix
9) Fully Dressed Use Cases of Main Scenarios
10) Domain Model as a UML diagram (Note that domain model and class diagram are different UML diagrams, only the domain model is asked for this report)
11) Conclusion

REQUIREMENTS PRESENTATION (07.12.2021)

The teams are required to present the results of their analysis efforts as a short (no more than 15 min.) PowerPoint slideshow. The presentation must refer to **all the works** included in the Requirements Report. However, the content of the works can be reduced in order not to exceed the time limit. For example, only a few requirements for each requirement category and a few use cases which you consider to be the most important ones can be included in the presentation.

ARCHITECTURAL MODEL (28.12.2021)

1. Introduction
2. Define the architecture of your system. Which model did you use (one of the generic architectural models or a new hybrid model you developed)? What are the main components of your architecture? What sub-systems are placed in these components? How are the relationships established between components and subsystems? Give brief definitions for each of these questions.
3. Show your architecture as a block diagram. (Draw your architecture including components, subsystems and relationships.
4. Explain why you did not choose other possible candidate architectures for your system.
5. Prepare a small legend or dictionary section in your report to clarify all your entities with.
6. Conclusion

* 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.

PROTOTYPE PRESENTATION (11.01.2022 & 18.01.2022)

The teams are required to present the results of their all design efforts and prototype implementation (demo) as a short (no more than 15 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. Your presentation should also include the test cases you have taken into consideration.

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. In your presentations, you need to explain what criteria are taken into consideration during the design of the UI. The prototype must give a message for all other use cases that are not implemented.

Team Work

- The project will be undertaken in teams consisting of **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 four, all contributions would be labeled 25%. 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.