CS 425 Software Engineering – Project Part 3: Design

Released Nov 6, 2018; due MONDAY Nov 19, 2018 at 11:59 pm; points: 100

Weight 10% of course grade

Deliverables

Note: In the following **<T3>** denotes a team of three students and **<T4>** denotes a team of four. Also, the notation **<X/Y>** means **X** applies to **<T3>** and **Y** to **<T4>**.

For this part of the project you should provide a *Design Document (DD)* with the following structure:

0. **Cover Page:** Should contain the following items:

    - Department, university
    - Project title
    - Team number [also, optional, Team name]
    - Team members
    - Instructors
    - External advisor(s)
    - Date

1. **Table of contents**
2. **Abstract:** a revised, shorter version of your project's abstract (75 to 100 words).
3. **Introduction**: a shorter general description (between 250 to 500 words) that briefly re-states the goals of your project and gives a concise account of progress made since the previous report (specification). Indicate also significant changes/updates made to your project's requirements.
4. **High-level and medium-level design**: present the project in terms of high level architecture, subsystems, and program units (modules). Given the diversity of projects, there is significant flexibility here. In any case, you should include, with accompanying textual descriptions, the following:

    - At least one *system-level diagram*, e.g., your system's context model (see Chapter 5 of the CS 425 textbook), your system's architectural pattern (see Chapter 6), or your component-based high level design (see lecture of 11/08/2018).
    - The structuring of your software in *program units*.
        - In the case of object-oriented solutions, the classes are examples of such program units, hence a design class diagram with details of attributes, operations, relationships, and multiplicity constraints should be provided (at least **<9/12>**classes are expected). Briefly describe the role of each class and indicate its main attributes and methods (in total, at least **<20/25>** methods should be described).
        - In non-object oriented solutions, program units can be modules, functions, procedures, subroutines, etc. Show the organization

(hierarchical or not) of these units (at least **<15/20>**units are expected) and provide for each of them: name, description, the higher level unit (e.g., subsystem) to which the program unit belongs, its input, its output, program units called by this unit, its exceptions or interrupts, and any additional comments that could enhance the description of the unit.

- Describe the main *data structures* that will be used in the project. If *database tables* are used, for each table indicate its fields (columns) and its primary key(s). For instance, a table containing information on employees may look like the following one (note that the primary key, shown in bold, is ID – the employee's ID #) (see below)

Example database table

ID	Last Name	First Name	Position	Department	Office	Telephone	Email

5. **Detailed design:** At least 3 **<T3>** or at least 4 **<T4>** state charts, activity diagrams, and/or pseudocode describing in details non-trivial components of your system's behavior (operations/functions). Use at least 2 types of detailed design notations (of the above three: state charts, flowcharts, pseudocode).

6. **Initial hardware design – as applicable**

   - A high-level diagram showing the organization of the hardware components of your system (that is, main components and their connections).
   - A list of potential components with brief descriptions of their roles. Include several snapshots (photos, figures, or diagrams) of components likely to be used. Indicate the sources used for snapshots/figures.

7. **User interface design:** provide at least 9 **<T3>** or at least 12 **<T4>** snapshots of the user interface, with accompanying descriptions. In these snapshots, the main user interface components with details (e.g., panels, toolbars, menus, menu items, buttons, data entry boxes, etc.) should be presented, and the format used in outputting results, messages, reports, and/or statistics should be shown.

8. **Glossary updates:** revise your project glossary from P-2, and provide at least 20 terms **<T3>** or 25 terms **<T4>** related to your project's problem domain (you can reuse those included in P-2).

9. **Contributions of team members**. Provide an estimate on how much time each team member worked on this project part (P-3), and indicate on what specific activities.

Grading Note that both the technical content and the presentation style (including quality of writing and document formatting) of your design document will be graded.

Project Assignment 3: Design

Criteria	Ratings		Pts
Overall Presentation, sections 1, 2, 8, 9	20 pts Full Marks	0 pts No Marks	20 pts
Section 3	15 pts Full Marks	0 pts No Marks	15 pts
Sections 4 and 5	40 pts Full Marks	0 pts No Marks	40 pts
Sections 6 and 7	25 pts Full Marks	0 pts No Marks	25 pts
			Total Points: 100