SOFTWARE ENGINEERING MIDTERM PROJECT

1 Objective

Design and implement an original/unique software system utilizing the skills acquired in this course

2 Project phases, deliverables, and due dates

This project must be completed by teams of three or four students using an agile process model.

Each deliverable must be submitted on or before the indicated due date. Teams will not be allowed to proceed to a subsequent project phase without submitting a satisfactory deliverable for the previous phase.

Each deliverable document must have a title page indicating the project title, deliverable name, team name and membership, and submission date.

The following will always be considered when grading your work in this course:

- **Presentation**: the report is well formatted, easy to read, and easy to navigate;
- **Quality of writing**: language, grammar, clarity, professionalism;
- **Consistency**: all required points are addressed in the same order as listed in each section below.

3 Project team info sheet (20 pts, due 5/11/2020)

Get to know your team and make decisions about your own rules. Submit a word document containing the following:

- Introduce your team: give your team a unique name and provide a group picture of the entire team. Given that this is a 100% online class, the group picture should be a screenshot of video conference session (via Google/Zoom/WebEx/whatever) showing the camera feed of each team member. What are the strengths of your team as a whole? **(2 pt)**
- Introduce each team member: include a picture and a 100-word biography of each team member. A biography should be professional as if you were introducing yourself to a prospective employer. Identify one team member who will serve as the main contact for the instructor **(6 pts)**
- Hold your first virtual meeting: include a picture showing your camera feeds in a video conferencing software of your choice **(2 pt)**
- Discuss every aspect of the *team agreement* in detail and describe all your decisions in the writeup **(8 pts)** Your team agreement must include the guidelines for the following (but feel free to add any other items you believe are important):
  - methods of communication (email, phone, messenger, text, ...)
  - communication response times (email, phone, messenger, text, ...)
  - meeting attendance (when to meet, whether all meetings are mandatory, ...)
  - running meetings (when, where, face-to-face vs. online, who takes minutes/notes, ...)
  - meeting preparation (whether preparation is needed, what to prepare, ...)
  - version control (what to/not to commit, content of log messages, ...)
  - division of work (how to divide work, who will decide who does what, ...)
  - submitting assignments (when to submit, who will submit, who will review the submission, ...)
  - contingency planning (what if a team member drops out, what if a team member consistently misses meetings, what if a team member is academically dishonest, ...)
- Each team member needs to create GitHub account. Include each member's GitHub account name in the writeup **(2 pts)**

4 Project proposal (20 pts, due 9/11/2020)

Submit an informal essay (word, 2 pages max) describing the following:

- Intended use of the system: who and how will use the system **(8 pts)**
- Its overall functionality: what will the system do, how will the system help its users accomplish their tasks **(6 pts)**
- Main components of the system: break down the system into logical or architectural components and provide the rationale for this breakdown **(6 pts)**

5 System requirements (20 pts, due 13/11/2020)

Submit a system requirements document (Word/PDF with a UML use case diagram, 5 pages max limit) as described below:

- Begin with a narrative to give a general overview of the system's *functional* requirements. Provide one big use case diagram illustrating the overall functionality of the system. Describe each use case in an easy to understand natural language **(2 pts)**
- Using the use case diagram as a starting point, convert each use case into a user story. Organize your user stories as a numbered list **(4 pts)**
  A typical format of the user story is as follows: As a *user role*, I want *goal* so that *reason*:
  - The *user role* represents an actor or a developer who benefits from this story;
  - The *goal* of the story is a feature or function in the system, a tool, or a part of a production pipeline;
  - The *reason* describes the benefit to the customer or user when this feature or function is used.
- For each user story, provide a set of pre- and post-conditions (refer to each corresponding user story by their numbers). Note: list pre- and post-conditions under the corresponding user story. Note: there should be a single list of user stories in your document **(4 pts)**
- Are there any user stories that are too big or complex (these user stories are usually referred to as 'epics')? Can some of them be decomposed into smaller user stories? For each user story, mention whether or not it may need to be broken down (this will be done later) **(2 pt)**
- Provide a separate list of any relevant *nonfunctional* requirements **(2 pt)**
- Include a *glossary* that defines all relevant terms that may have a special meaning in the context of the system **(4 pts)**

6 Product backlog (20 pts, due 20/17/2020)

Submit a detailed prioritized product backlog document (word/PDF, no page limit) as described below:

- Refine your user stories taking into account the instructor's feedback. Break down previously identified large user stories (epics). Indicate which epics resulted in what new user stories **(4 pts)**
- Estimate the size of your user stories. Use Fibonacci numbers within the range of 1 to 8 to represent a relative size of each user story. Label each user story as high, medium, or low priority. Note the cumulative size of all user stories in your product backlog **(4 pts)**
- Provide an updated numbered list of all user stories; indicate pre- and post-conditions **(2 pt)**
- Taking into account the pre- and post-conditions, identify a subset of user stories to be implemented during the first sprint (there will be a total of three sprints). Be sure that the cumulative size of the selected user stories is about 1/3 of the size of the full backlog. Describe the functionality that your partially implemented system will have at the end of this sprint **(6 pts)**
- Design key features of the user interface; provide sketches of your designs **(4 pts)**

7 Midterm presentation (20 pts, due 20-24/11/2020)

Each team will make an in-class presentation reflecting on the following:

- The project, its goals, its domain, project users and their needs **(4 pts)**
- Main functional and non-functional requirements of the project **(4 pts)**
- Highlights of the product backlog and sprint planning strategies **(4 pts)**
- Any lessons learned about interacting with the customer, teamwork, and other non-technical aspects of the project **(4 pts)**

Submit a PPT used for the in-class presentation. The presentation must be free of typos, readable and understandable by the audience, and neatly formatted **(4 pts)**