CptS422 Software Engineering Principles II (Fall 2019)

Course Project

1. General Description

The project is the central part of the grade for this course. You will apply and exercise what you learn from this course in the process of this project so that the concepts and abstract understandings on the software engineering principles can be transformed into practical skills.

The goals of the project include the following:

1. Provide an environment for software engineering practice in a smaller scale, enabling you to practice and experience varied software testing activities within a single semester, and to encounter and learn how to solve common software testing problems.
2. Facilitate basic understanding of various concepts about software testing, such as testing strategies and testing techniques.
3. Foster and/or enhance the ability to conduct self-motivated demand-driven research---you will need to pick up a few other things that are not to be covered in this course in order to get the project well done.
4. Help harness basic software development skills such as project planning/management, communication, and team collaboration as well as software version control.

2. Project Topics

The objective is to learn by practicing software testing strategies and techniques along with associated, representative tools. At the beginning of the course, you will form project teams with 3 to 5 members. During the semester, the project team will work together throughout all milestones of this project. Some potential projects and clients will be suggested (refer to the document on suggested project topics) but you are encouraged to identify your own.

3. Team

Each team should have 3-5 members. One member should serve as the contact – not necessarily the leader but will be responsible for the regular communications of your team with your instructor and TA, such as scheduling meetings for milestones check. By the deadline of the *Milestone 0*, submit your team information in a single PDF to the TA and the instructor, including the following elements:

- Project topic (if you come up with your own project, you need to inform and convince the instructor that your topic meets the expectations --- see the document on Course Project Topics)
- Team name
- Contact: (Name and email address)
- Members: (Names and your EECS server login usernames, including the contact’s)
- Project name (should reflect the project topic chosen, and incorporate the team name for differentiation purposes)

- GitHub/bitbucket repository URL for this course project, where the TA's account has been added as a collaborator (observer). TA's account will be posted soon on Blackboard Discussion Forums.

Students who couldn't find teammates by the deadline should see the instructor to work around this as early as possible.

4. Technical Milestones/Deliverables

There will be three major technical milestones, preceded by Milestone 0 on team creation, as described above.

Milestone/Deliverable 1

In this project deliverable, each team will be asked to implement a software system followed by requirements analysis and design. Students will produce a report explaining the role of each team member and major project activities that took place during the development process.

Assessment criteria for this deliverable include (1) the clarity of the report, (2) the choice and justification of followed practices and process, and (3) the quality of the produced software (including the tests). Particular emphasis will be given to (1) the functionalities provided by the software and whether the functionalities fulfil the user requirements, (2) the history of the project activities in the version control tool, and (3) the code readability and documentation.

Milestone/Deliverable 2

For the deliverable, each team will test the system using black box and white box testing techniques. The focus will be unit testing.

Submissions will be assessed on the choice of techniques and the justification, and on the quality of the test cases (e.g., completeness, coverage, discovered bugs). In addition to the test related artifacts, each team will submit a report explaining the degree of confidence in the correctness of the software, the tools and techniques used, and they will reflect on the suitability of their choices and provide alternative solutions if the conclusion shows that a technique/tool that was used did not show to be efficient.

Milestone/Deliverable 3

For this deliverable, each team will continue to test the system developed in Milestone 1 with a focus on integration testing, system testing, and other testing techniques (e.g., object-oriented testing, mutation testing, etc.). The goal is to supplement the (unit) testing done in Milestone 2, but improvements of previously developed unit tests should be part of this milestone as well. Similar to the other two milestones, a milestone report will be submitted along with the developed tests. Each team will have the options to target either the system developed by a different team, or the system developed by the team itself.

The evaluation criteria for this deliverable include (1) the choice and justification of techniques used, (2) improvements made for the previously developed unit tests, (3) quality of the test cases for integration testing and system testing (in terms of completeness, coverage, discovered bugs), and (4) clarity and thoroughness of the report.

5. Project Evaluation

The grade of this project consists of the grade per milestone received as a team and the peer evaluation score per team member. First, each member will receive the same grade as the team grade. Second, the project grade will be adjusted for each team member based on how teammates peer-evaluate the member. Details on peer evaluation will be given later.

All team members are expected to contribute equally to all project deliverables and to all components of each project deliverable. As part of all deliverables, the team will list the tasks assigned to each team member, the tasks completion percentages, and the tasks completion dates. The instructor will use the history of the version control system (e.g., submitted artifacts such as code and documentation, time stamps, commit messages) to verify the task completion percentages and the individual contribution of each team member.

The group grade will be assigned by evaluating the following components:

Deliverables	**Components**	**Percentage of the final grade**
Deliverable 1		**15%**
	Delivered software code	5%
	Software documentation	5%
	Report	5%
Deliverable 2		**15%**
	Test cases	10%
	Quality measurement of test cases	
	Report	5%
Deliverable 3		**15%**
	The improved and added test cases	5%
	Quality of the improved and added test cases	5%
	Report	5%