**Black Box Test Plan (3%)**

A test plan is a detailed document that outlines test cases for a given software system. Test cases are the functionalities and/or features to test in a software system. A good test plan has the following properties:

Repeatable: able to be replicated by other developers on the team
Specific: descriptive steps to follow
Focused: Each test should only analyze one particular function of the system
Black box test plans are documents that use plain language for stakeholders to follow to verify the program. You must create a black box test plan with at least ten unique black box est cases with the required test case information (see a template in the Lecture Slides). If you are not implementing your project, you may leave the Actual Results column blank (but it must be present in the plan to receive full credit).

Link:
https://docs.google.com/document/d/17aWjN6UDS7pKrAJ-na8HBd04idwWKtsWrVSMj7Di2qY/edit?usp=sharing

**Final Presentation (15%)**

Instead of a final exam, your group must give a 10 or 12 minute oral presentation about your project. The presentation must include the original problem statement, an explanation and rationale for the proposed solution, updated use cases with diagrams, a visual representation of your project (i.e. mock user interface), a discussion on limitations and future work, processes and tools used, and things you learned. All project team members must participate in the final presentation for the project. The slides for your presentation are due before PM4 on December 4th before class. The final presentations will take place in class December 4, 6, and 11. The order of lightning talks will be randomly selected immediately before class on December 4.

Link:
https://docs.google.com/presentation/d/1Shbx7RxgMk4qOZn1PGxvSyBzmV4oEIPVe_sOminJ3hE/edit?usp=sharing

**Final Report (15%)**

The final report must be no more than 10 pages (not including references) and follow the same ICSE formatting guidelines as the Proposal Document. The final project report must contain the following: * a relevant title and all group members listed as authors; * the abstract with feedback addressed; * an updated introduction incorporating proposal feedback to define the problem and motivate the project solution; * a motivating example to provide a scenario of how your project

would be used and why it is relevant to software engineers working together; * a background to define any key terms or concepts related to your work (if applicable); * an updated related work citing relevant work and tools, including the novelty of your project; * a description of the implementation design decisions, processes, and testing approach * a deployment plan section explaining how you would deploy and maintain your project for users if you were to release it; * a discussion explaining possible opportunities for future extensions and limitations of your project; * a conclusion revisiting the problem statement and project work completed; * and references for any works that you cite.

Link:
https://docs.google.com/document/d/1Vrq1NrGZvbwwiC6yFbaM-e8Livu93E52hMozertOaXM/edit?usp=sharing

**Retrospective (2%)**

At the conclusion of the project, you will asked to complete a brief group and individual retrospective about the team project. This will allow you to reflect on what went well, what didn't go well, and what you learned in addition to determining your contributions and the effort of your teammates towards the project. The overall grade will be impacted by reported group dynamics that cannot be determined through analyzing contributions to your repository.

Your repository should also include the previous project submissions, project management (i.e. issues, project, etc.), and code (if applicable). A README file should provide an overview of your project and contain each of the group member first and last names and PID.

Link:
https://github.com/alexandersg-vt/CS3704-Project