R1: Design Draft

Overview

Students will develop a project design plan draft with their team members in this assignment. Each team should submit one plan together. The plan should provide the reader with a high-level description of the project, its major technical tasks, support platforms / frameworks, and expected deliverables upon completion. It should include figures to help the reader understand the design and no more than 1500 words, excluding figures, headers, captions, and references. Any outside sources used to build the report, and any outside tools the team will be using, should be properly cited in ACM or IEEE format.

Report Elements

The plan should include the following sections and cover the entire project (***through the end of the semester***).

Introduction

In the introduction, your team should describe the value of the project via a brief overview of its sector and the identification of features that make your project unique. The introduction should clearly address these topics:

Purpose / Need

Describe the purpose in society for the project you are designing. Specifically, explain why it is needed, by whom, and in what context. Identify how the project is useful and goes beyond a mere demonstration.

Domain & Prior Art

Explain the field / area of the project and describe existing work in this area. Provide examples from existing commercial products, web searches, or articles appearing in the literature. All examples should be cited. Identify what features will differentiate you solution from others and why your project will make a difference.

Statement of Work

Identify what work is to be done during the semester. The statement of work should include details and/or estimates wherever possible. Quantify the performance specifications and/or requirements of your design. For example, software elements should note dependencies (libraries / frameworks) and special tools.

This section should also incorporate a list of core and secondary features / tasks / milestones, target dates for completion, and who on the team will complete the task. Students may use a table or other visualization for this information.

Deliverable Artifacts

State artifact(s) to be delivered by project's end (software, critical documents, etc.). For each, include…

- A short description of the artifact – how does it fit into the project as a whole?
- Accessibility / usability / maintenance plan – how will the project remain viable / useful in the long term?

Mockups

In the design process, developing mockups can help guide the project and give a visceral feel for how it will function. A *mockup* in engineering usually consists of a set of visual models that demonstrate how an artifact with work and that provide a mechanism to evaluate the design. In this assignment, student teams should develop mockups for their design project. Mockups can include wireframes, draft schematics, and/or other diagrams. The mockup submissions should be professional in nature (e.g., should NOT merely be scanned sheets of paper) and should be clearly labeled.

Interfaces

Any aspect of a project that a human or outside system will interact with should include an interface model mockup. For physical and graphical interfaces, this should include clear sketches of all controls, the function of such controls, and how the user will interact with them. Programming and software interfaces should include a draft specification for calls and/or protocols, and ideally diagrams of the communications.

Systems

Projects with multiple systems or subsystems should include visualizations of the systems. These should include how the (sub)systems communicate with one another, how they are connected, their dependencies, and how they function in general and together.

Networking

Models for all telecommunications – including personal or near-area networks – must be documented explicitly in draft form. This a includes clear protocol draft if applicable. If specific application layer protocols are used (e.g., HTTP), the specific use case and implementation should be included within the models.

Storyboards

All scenes from software (dialog sequences, control paths, and screens) must be modeled in the mockup in storyboard or via other visualization. The storyboard should include a mockup of each software screen and how it can be reached. For games, this will include all game screens as well as level sequences.

Submissions

Students will submit a PDF document on Canvas.