

Software Engineering Lab

Week 3 Task 3.1: Wikis and Design Documents

Overview

Understanding and sharing the requirements for a project are a key influence in the success of any software solution. Team development teams often use a project design document to help capture some of these details. Design documents also need to change over time to be current and useful. In a team we will create a project flow design document using GitHub wiki pages.

- Purpose:** To create a project flow design document using a collaborative format of a wiki.
- Tasks:** As a team, create a project flow design document using GitHub wiki pages.
- Time:** This task should be starting in your lab class and demonstrated via classroom when done.
- Resources:** Your course teacher(s), your team members and the lab class.
See:
- <https://help.github.com/articles/about-github-wikis/>
 - <https://guides.github.com/features/wikis/>
- Feedback:** Your teacher(s) will give you feedback on the document your team creates and your activity. Your individual contribution evidence will be documented and submitted to classroom.

Task 3.1 Submission Details and Assessment Criteria

You should create a personal (unique) document with an image of your individual contribution to the work, as well as the wiki change log showing your changes. Also include the location of your GitHub wiki.

Instructions

Your team can be a new one for this task, or your team from the last task.

The exact document structure is up to your team but it is recommended that you create a single page with sections, not a number of different pages.

You may only get the chance to start this document in the class, and not finish it, so make sure you know what each person needs to do to finish the document before you leave the class.

Your document may contain text, lists (bullet points), images and diagrams, but keep the document to a basic level. You should have at least one image so you know how to.

Here is a basic table of contents for sections your document can have.

1. Team and Title Details

Note who is in your team and the title of the game.

2. Overview of the project.

A very short description of the story (a few sentences), target platform, visual style and audio style.

3. Project flow screens and flow.

Show the basic screens of the project flow and how a player can get between each. (See the Notes at the end of this document.) For example, show the *homepage* screen, main menu, options, login, breaking-news, closing credits, etc.

4. User Interface

For any of the screens you have noted, describe the details. Note visual changes and sounds that would be played.

5. Usage

Describe the mechanics or flows of the project. For example, what is the result of a user login, what is a legal checkout, and so on. Think about what events occur during each action, and what would be needed (such as rules, images or sounds).

6. Assets:

List what assets are needed. This includes images and sounds. There is no need for actual images, just a clear list of what is needed.

Steps:

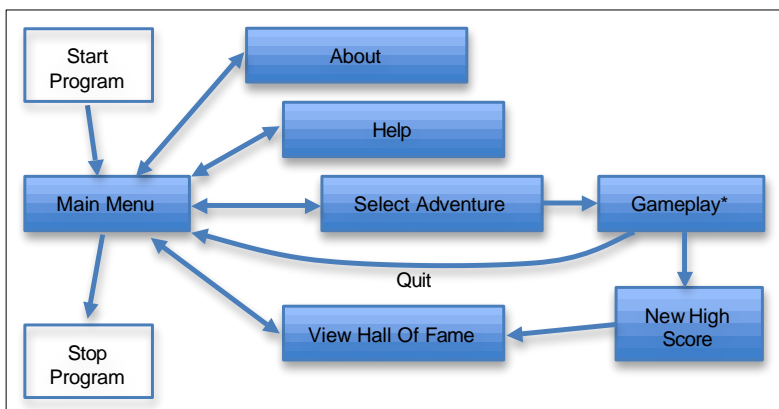
1. As a team, **review** the project flow design document template details above and understand what you need in each section.
2. Discuss the **details** needed for the project flow. (Don't take too long on this as it's not the point of the task!)
3. Your team will then need to plan what document sections are needed, and **divide** the content between team members to complete.
4. Create a GitHub **project** and add in all the team members.
5. Create a starting **wiki** page. See the online help for details.
6. Each team member will then help to create the design document **content** using a GitHub wiki.
7. When you have the overall sections done and everyone has contributed some content, **stop**. The document does not need to be perfect for this task!

Extra Notes

A sample (Game Screen Flow)

The idea of game screens or flow is simply to capture each “stage” of the game, and the different types of user interaction, at each part of the game. There are several terms used such as game “states”, “screens” or “stages” but it doesn't matter which one you use, as long as your team is consistent!

Below is a simple screen flow example. Note that there should be some indication what is the trigger or event that caused the change from one screen to another, such as pressing the “q” key or “clicked exit button”.



Adding Images

Adding images to GitHub wikis takes a few steps, keeping in mind that wikis on github are stored in a git repository.

1. Clone the github wiki (not the project wiki) to your local machine
2. Add the image to the wiki repository
3. Stage and push the wiki back to the remote website repository
4. Link to the image using a relative URL.

<http://mikehadlow.blogspot.com.au/2014/03/how-to-add-images-to-github-wiki.html>