Code Workstation

Status Report 5

11/18/2015 - 12/9/2015

https://github.com/lrickard/CodeWorkstation/wiki

Lukas Rickard - Project Leader - l_rickard@u.pacific.edu

Jake MacMillan - Integrations Lead - j_macmillan1@u.pacific.edu

Project Schedule

Current Schedule

Below is our rough development plan for next semester. We plan to start by writing classes and manually testing each of our features as we create them if possible. Later in the semester we plan to do usability testing and hopefully we will be using our program to develop itself later in the semester.

Code Workstation Gantt Chart	2/5	2/19	3/4	3/18	4/1	4/15	4/29	5/11
Create Button Buttons	██	██						
Create Basic Command Box for output	██	██						
Test Button Buttons	██	██						
Create Command Box Auto Run Functionality	██	██						
Test Command Boxes	██	██						
Create and Test Drop-Down Menus		██	██	██				
Create Dock and Dock creation system		██	██	██				
Test Dock Creation system and scalability of existing features		██	██	██				
Implement dock resizing mechanics				██	██	██		
Implement window resizing mechanics				██	██	██		
Test and debug resizing features				██	██	██		
Implement keyboard shortcuts						██	██	
Usability testing						██	██	
Debugging and Interface polishing						██	██	██

Schedule Changes

Our schedule for this semester is irrelevant. We are planning for next semester and taking a cautious approach to it at that. Observing the Gantt chart above one can see that at least two independent tasks at any given time. This will hopefully allow each member of the team to be working on something separate that can be brought together each month. We have allotted specific time at the end of next semester in order to work on polishing our software, doing usability testing, and implementing stretch features.

Current Status

We have honestly not spent very much time on this project during this status report period. We have edited the Design document a small amount and made a basic skeleton of a test plan. We added notes on how we plan to implement each feature in words because we think it will help us get started next semester. We also took time to predict ways that features in our program could be abused so we can keep them in mind while developing; We call them our "What could go wrong" sections. Overall, we are confident we will be able to start developement smoothly next semester.

Plan for 1/22/2016 - 2/5/2016

Start on:

1. Basic Command Box for Integrations
2. Implementing Button Button
3. Implementing Command Box features

Work on most:

1. Buttons and Button Button
2. Getting source code and integration plan organized

Finish:

1. The Basic Command Box for Integrations

Challenges and Concerns

We want to make sure we start the semester out correctly and stay organized. For us this means using version control to its full power and staying in sync with each other about how we will integrate each features every few weeks.

Updated Artifacts

Test Plan - We made this, It's got practical implementation and testing plans.

Included with Update

Project Proposal (Unchanged)

Requirements Analysis Document (Unchanged)

Wiki and Repository (Linked in documents - not included in zip)

Status Report 1, 2, 3, 4, and 5 (These will not change)

System Design Document

Test Plan