|  |
| --- |
| Designer drug database |
| Final Document |
| Senior Project, CIS 4911- U01 |
| **Professor: Seyedmasoud Sadjadi Mentor: Dr. Luis Arroyo** |
| **Team Member: Carlos Dominguez** |
| **2/2/2015** |

|  |
| --- |
|  |

Copyrights and Trademarks Notices

Nothing yet

Executive Summary

One or two paragraph

Table of Contents

ss

1. Introduction

Introduce the introduction (one or two paragraphs)

* 1. Problem definition.
  2. Scope of system.
  3. Over all development methodology.
  4. Definitions, acronyms, and abbreviations (at most one page).
  5. Overview of document

1. Feasibility Study
   1. Description of current system. Identify limitations and constraints

In this case there is no current system, the system is to be created from the beginning

* 1. Description of alternative solutions considered.

The current options are platform independent framework, hybrid apps crators, and responsive web design application.

For platform independent framework there are several different frameworks that can be used. Some of these are:

**Sencha Touch 2**

**Knowledge required:** HTML, CSS, JavaScript, general web development

**Platform support:** iOS, Android, Blackberry

**Cost:** Free [under commercial and open source licenses](http://www.sencha.com/products/touch/license/) (paid OEM license available)

**Documentation:** [Examples, screencasts, and tutorials](http://docs.sencha.com/touch/2-0/#!/guide/getting_started)

**jQuery Mobile Summary**

**Knowledge required:** HTML, CSS, jQuery

**Platform support:** Most mobile browsers

**Cost:** Free (license: [MIT and GPL](http://jquery.org/license/))

**Documentation:** There are [many demos on the site](http://jquerymobile.com/demos/1.1.0/) and [books on jQuery Mobile](http://www.amazon.com/s/ref=nb_sb_noss?url=search-alias%3Daps&field-keywords=jquery+mobile)

The other option is hybrid app creators (HTML/JAVASCRIPT AND HTML/RUBY BASED). The best choices based on support for hybrids are:

[**Appcelerator**](http://www.appcelerator.com/):  This is a solution that allows you to develop native apps with HTML/Javascript (run through a UIWebView on iPhone) . (Free)

[**Phonegap**](http://phonegap.com/): Similar to Appcelerator, I mentioned these two as they seem to have the most vibrant communities, and most extensive support. (Free)

[**Rhomobile**](http://rhomobile.com/) : This is a solution that uses Ruby, especially loved by Ruby on Rails developers. (Free only for noncommercial applications, prices vary)

The last option to create an app like this is to use a responsive web design. Some responsive web design frameworks are:

**Bootstrap:** Sleek, intuitive, and powerful front-end framework for faster and easier web development.   
**Webpage**: getboostrap.com

**Foundation 3:** An advanced responsive front-end framework. Foundation 3 is built with Sass, a powerful CSS preprocessor, which allows us to much more quickly develop Foundation itself — and gives you new tools to quickly customize and build on top of Foundation.  
**WebPage**: foundation.zurb.com

* 1. Recommendation with explanation of why the solution was selected.

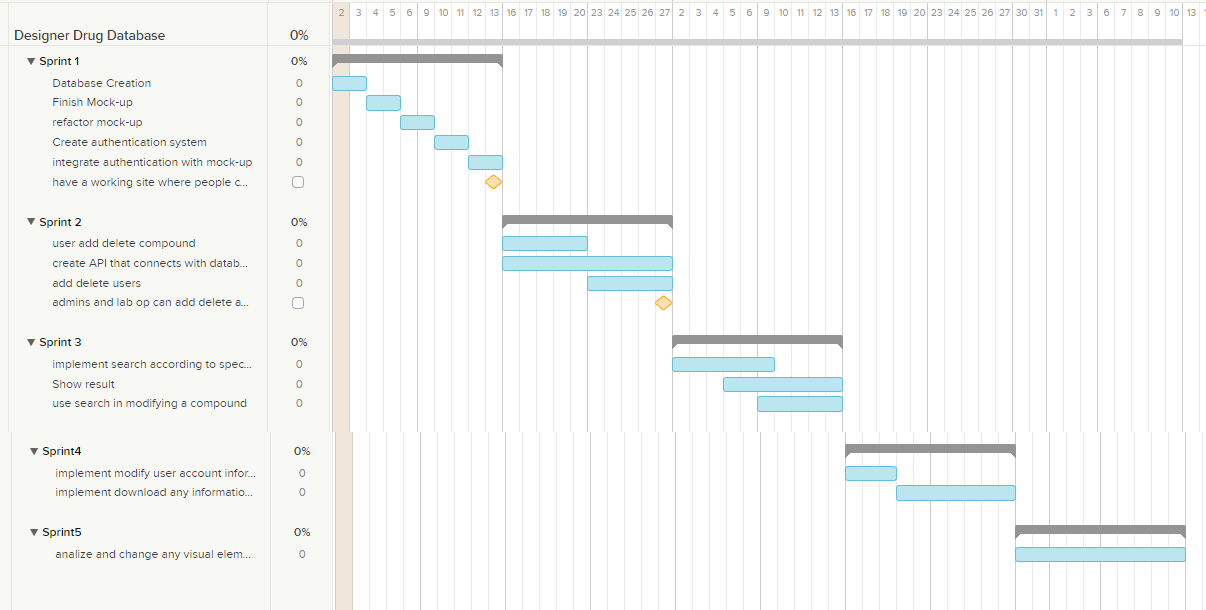
All this choices have pros and cons but every choice that is not platform dependent and uses the web as their main platform has more less the same vision and goal. However, between them the responsive web design is the more appealing choice given it responsiveness, where a web app will look good in any device and adjust to the resolution.

Given that it have been established that a responsive web design is the best choice, now we are going to analyze the different choices. It is important to say that if we compare bootstrap with Foundation 3 we find that bootstrap has a lot more documentation and example on the web. Moreover, there is support because it is in large use nowadays. Looking deeper, If we look at the applications and framework that integrate with the listed responsive web design frameworks, it is important to say that bootstrap integrates very well with another framework call angularjs, which is developed and maintained by google. Thus it is my recommendation to implement the system using the bootstrap responsive web design framework.

1. Project Plan
   1. Project Organization

The project currently doesn’t have many user stories but there is a lot to be done.

* + 1. Project Personnel
    2. Hardware and Software Resources
  1. Identification of Tasks, Milestones and Deliverables



* 1. Cost of the Project

1. System Requirements

Introduce the proposed system (one or two paragraphs).

* 1. Functional and Nonfunctional Requirements – similar to RD
  2. Requirements Analysis - similar to RD

1. System Design (i.e., overall system design)

Introduce the system design chapter (one or two paragraphs). State the uses cases you are implementing.

* 1. Overview – overview of system decomposition. Identify the architectural patterns used and state why they were selected.
  2. Subsystem Decomposition - describe each of the major subsystems. Identify the use cases (or parts of use cases) associated with each subsystem. Refer to use cases in appendix B.
  3. Hardware and Software Mapping – map subsystems to h/w and s/w.
  4. Persistent Data Management – identify data that needs to be stored e.g., attributes of objects, and primary attributes (may use a table format).
  5. Security/Privacy – describe user authentication processes, encryption of data, and use of firewalls or security servers.

1. Detailed Design

Introduce the detailed design chapter (one or two paragraphs).

* 1. Overview – briefly describe the behavior and structure of each subsystem. Describe the design patterns used and why they were selected.
  2. Static model – detailed description of the structure for each subsystem. May include detailed class diagrams. Appendix C.
  3. Dynamic model – state machine diagram for the main control object in each subsystem. Include the design of the algorithms used in the problem solution. Refinement of the sequence diagram from the analysis model. Appendix D.
  4. Code Specification - describe the class interfaces (attributes and method signatures) and constraint (invariants, pre-condition and post-conditions). Code should be in Appendix E.

1. System Validation

Introduce the system validation chapter (one or two paragraphs).

* 1. Subsystem Tests – test each of the subsystems. This will involve the creation of a test drivers and stubs. Include the code for the test drivers and stubs in Appendix G.
  2. System Tests - For each use case create at least 3 test cases, 2 sunny day and one rainy day, should include security test cases. Each test case should include: test case id, purpose, test setup environment, test inputs, and expected outputs.
  3. Evaluation of Tests – evaluate how successful the tests were. Use a tabular form.

1. Glossary - define terms used in document, especially domain specific terms.
2. Appendix
   1. Appendix A - Project schedule (Gantt chart or PERT chart).
   2. Appendix B – All use cases with nonfunctional requirements.
   3. Appendix C – User Interface designs.
   4. Appendix D – Analysis models (static and dynamic)
   5. Appendix E – Design models (static and dynamic)
   6. Appendix F – Documented Class interfaces (code) and constraints.
   7. Appendix G – Documented code for test drivers and stubs.
   8. Appendix H – Diary of meeting and tasks for the **entire semester**.
3. References

**User’s Guide:**

1. Cover Page
2. Copyright and trademark information.
3. Introduction – brief description of application.
4. Hardware and Software Requirements.
5. Installation and setup
6. Getting Started - How to run application. Include user screens where appropriate and the description to run the two most important use cases.
7. Quick reference
8. Accessing online help
9. References