

Software Requirements Specification

for

Web Planner

**Version <1.0>**

**Prepared by**

**Group Name: Untitled**

|  |  |  |
| --- | --- | --- |
| **Conner Batson** | **011668692** | **conner.batson@wsu.edu** |
| **Josh Giurculete** | **011580119** | **jousua.giurculete@wsu.edu** |
| **Gema Hernandez** | **011562367** | **gema.hernandez@wsu.edu** |
| **Michael Wheeler** | **011526058** | **michael.j.wheeler@wsu.edu** |
|  |  |  |

|  |  |
| --- | --- |
|  |  |
| **Date: 10/25/2019** |  |
|  |  |
|  |  |
|  |  |

**Contents**

**Revisions iii**

**1** **Introduction 1**

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 2

**2** **Overall Description 3**

2.1 Product Perspective 3

2.2 Product Functionality 3

2.3 Users and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

**3** **Specific Requirements 5**

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Behaviour Requirements 6

**4** **Other Non-functional Requirements 7**

4.1 Performance Requirements 7

4.2 Safety and Security Requirements 7

4.3 Software Quality Attributes 7

**5** **Other Requirements 8**

**Appendix A – Data Dictionary 9**

**Appendix B - Group Log 10**

**Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft 1 | Conner Batson  Josh Giurculete  Gema Hernandez  Michael Wheeler | Initial drafting of SRS | 10/22/1029 |

# 

# 

# Introduction

## Document Purpose

The product being described by SRS version 1.1 is a schedule planner. This SRS specifically covers a general description of the product’s initial features.

## Product Scope

*This web application is a calendar for college students. The main purpose is to help students organize their academic schedules, bills deadlines, workout sessions and miscellaneous activities.*

## Intended Audience and Document Overview

*The intended audience for this document are college students and the incredible professor, Dr. Xinghui Zhao. This SRS contains information regarding product functionality, software and hardware components, databases and specific details about user-product interactions. College students should read this SRS beginning with the overview and moving on to the sections 3.2( Functional Requirements), 4.1 (Performance Requirements), and 4.2(Safety and Security Requirements). The Incredible Professor Dr. Xinghui Zhao should read this SRS starting from the overview sections and moving through sections 2, 3, 4 and 5.*

## Definitions, Acronyms and Abbreviations

SRS: Software Requirements Specification

## Document Conventions

## *In general this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1” margins found in this template. For Section and Subsection titles please follow the template.*

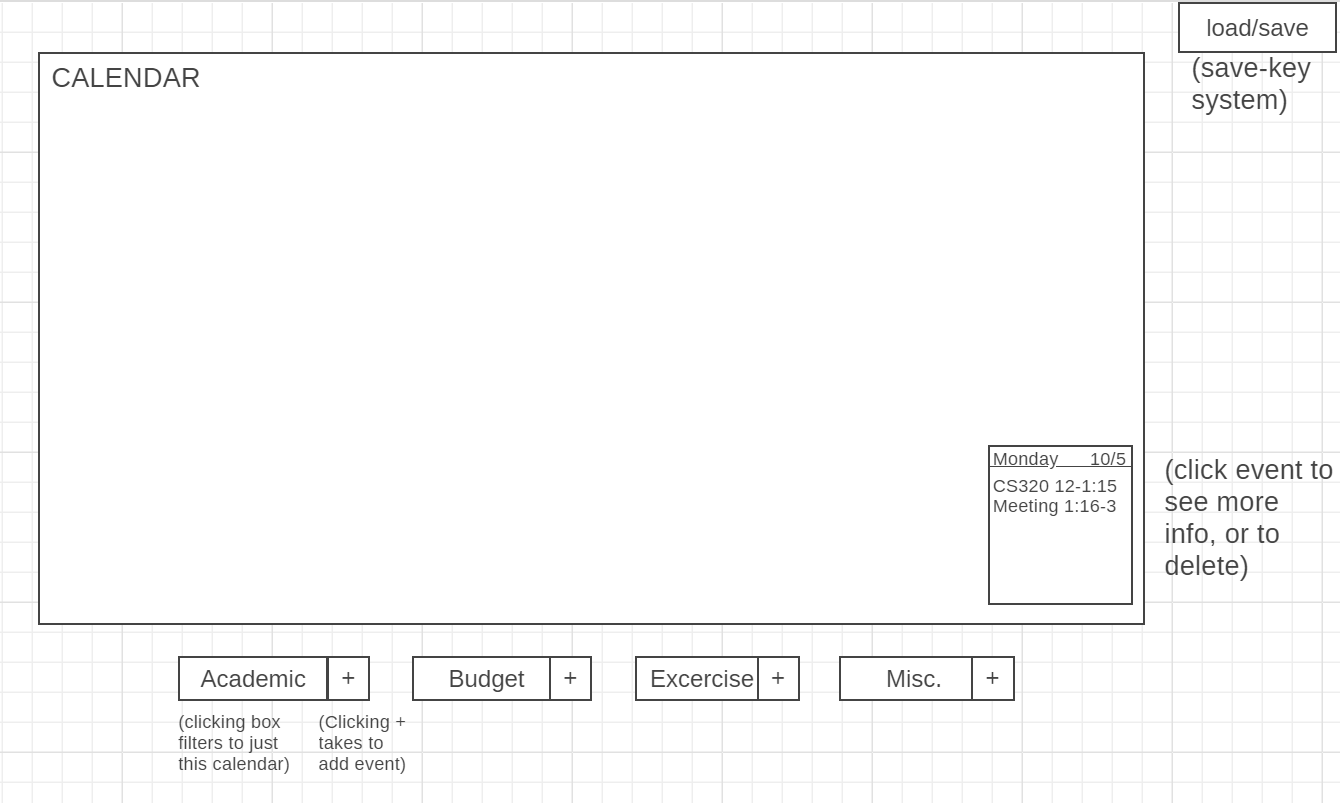
## References and Acknowledgments

Our main reference will be slides covered in class and the textbook. We will add the sources we reference as we work on the project.

# Overall Description

## Product Perspective

This product is a calendar that helps college students organize deadlines for school, work, bills and miscellaneous activities. This product is intended to be used as the main organizational tool. It will help students focus on activities that are more relevant for college students. There will be four buttons at the bottom of the calendar. Each button will allow the user specify a category to be shown. There will be a load/save button that will create a link to that specific user information.

**

## Product Functionality

* View Calendar of events
* Filter by Academic, Budget, Exercise, and Miscellaneous.
* Add event for each category of events.
* Click on event within day on calendar, to view info or delete.
* Save/Load personal calendar with personal save key.

## Users and Characteristics

The primary user of this web application will be college students who have planning needs for their day to day life. The characteristics of the typical student who uses this planner application would be students who keep up with their academic deadlines, keep a basic budget, and exercise. This student would be interested in keeping a centralized planner to keep track of these in one place.

## Operating Environment

The hardware platform intended for this web application are laptops or PCs. This web application is not intended for smaller screens. The interface for the software will be Google Chrome, which will run on Windows, Linux, and OSX.

## Design and Implementation Constraints

A time constraint is a constraint that limits the project completion on **SOME DATE**. Developers must develop in JavaScript and HTML, where the project is limited to the Google Chrome interface. Additionally, this software will be developed on Intellij, using Airbnb coding style. Github will be our version control method for development.

## User Documentation

The user documentation for this web application will be a FAQ page which describes general use scenarios and answers questions in a way that describes general use.

## Assumptions and Dependencies

The product assumes that users will want to add and filter events onto a calendar planner that are specifically related to budget, exercise, academic schedules, and miscellaneous activities.

# Specific Requirements

## External Interface Requirements

### User Interfaces

The user interface for the web planner will show an interactive calendar, where each day on the calendar can display the events planned by the user. For each event, the user can click on it to show the description, or to delete the event. Beneath the calendar will be 4 buttons, these buttons will lead to pages that show calendars for just the specific category. (Academic, Budget, Exercise, Miscellaneous). For each button there will also be an attached (+) button, which is used to add an event for the given category.

At the top right of the interface there will be a save/load button for the user to save/load their calendar with a create save-key.

### Hardware Interfaces

This calendar is intended to be used on a laptop or desktop. The data used will be added by the user. The software will handle the four categories: Academic, Exercise, Budget, and miscellaneous.

### Software Interfaces

This html/javascript interface will be displayed on Google Chrome, which will run on Windows, Linux, and OSX.

### Communications Interfaces

The communications interface requirements for using this web planner would be a web browser with internet connectivity.

## Functional Requirements

Schedule Planner Functions:

1. If the user clicks the button to add an event, the system shall provide the user with the ability to add an event to the calendar
2. If the user clicks the button to remove an event, the system shall provide the user with the ability to remove an event currently in the calendar
3. If the user clicks the filter button(s), the system shall provide the user with the ability to filter specific events based on which button is clicked (budget, exercise, etc.)
4. If the user clicks a specific day on the calendar, the system shall provide the user with the ability to view all events occurring on that day.
5. When the user is looking at the calendar, the system shall display a brief description of events happening on each day in the order they occur.
6. When the user is looking at a specific day, the system shall display a description of events happening on that day in chronological order.

Possible Functions:

1. If the user is able to save their calendar, the system should provide the user with the ability to access their saved calendar when the user requests it.
2. If the user is able to view events happening on a specific day, the system should provide the user with the ability to return to the calendar view of all registered events.
3. If the user is able to filter events by clicking a button, the system should provide the user with the ability to clear all filters on the calendar of events.
4. When the user is editing the calendar, the system should be able to save its calendar so that the user can access it at another time.

## Behaviour Requirements

### Use Case View

# Other Non-functional Requirements

## Performance Requirements

Any interaction should occur within a few seconds, nearly no loading times.

## Safety and Security Requirements

* *The user might lose information regarding assignment, bills, exercice and other deadlines.*
* *The user should have a back-up storage for this information.*
* *The software is not responsible for updating deadline changes and alterations*
* *The data entered by the user can be accessed by anyone with the access-string to log onto the user’s information. so the string should be secured by the user and only shared with people who the user allows.*

## Software Quality Attributes

This software will be easy to maintain, the variables will be well named to give a clear understanding of what it data it represents. The functions will be well commented in a clear and concise manner. Such that any maintainer, or curious user, will be able to clearly understand what is going on.

# Other Requirements

For this project we will be implementing the database with Mongo.

**Appendix A – Data Dictionary**

*We have not worked on the project yet, so we don’t know exactly what variables are going to be used.*

*<Data dictionary is used to track all the different variables, states and functional requirements that you described in your document. Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.>*

**Appendix B - Group Log**

*October 15th:*

*1:15 - 2:00 Picking product for project, and looking over requirements for SRS*

*October 22nd:*

*1:20 - 5:00 Working on SRS document as a group, with final bits left for each’s own time.*