Pro Calendar

Software Requirements Specification (SRS)

*Members:*

# **Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary****Author(s)** | **Description of Version** | **Date Completed** |
|  |  |  |  |

# **Review and Approval**

|  |  |  |  |
| --- | --- | --- | --- |
| **Approving Party** | **Version Approved** | **Signature** | **Date** |
|  |  |  |  |

# 

# **Requirements Document Review History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Reviewer** | **Version Reviewed** | **Signature** | **Date** |
|  |  |  |  |

# 

# Table of Contents:

## Introduction

* 1. Purpose
  2. Scope
  3. Definitions of Acronyms and Abbreviations
  4. Overview
  5. Business Context

## General Description

* 1. Features/Functions

## Business Requirements

## User Requirements

## Functional Requirements

* 1. Database and Identification
  2. Customization
  3. Events

## Interface Requirements

* 1. User Interfaces
  2. Hardware Interface
  3. Software Interface

## API

* 1. Kivy API
  2. Sqlite3

## Performance Requirements

* 1. User Identification
  2. User Account
  3. Account Management
  4. Event Management

## Other Non-Functional Attributes

* 1. Security
  2. Availability
  3. Performance
  4. Maintainability

## Operational Scenarios

* 1. Account Creation
  2. Event Creation
  3. Reminders

## Use Case And User Stories

* 1. User Stories
  2. Use Case Diagram

## Sequence Diagrams

* 1. Account Creation
  2. Event Creation

## Collaboration Tools

* 1. Communication
  2. Document Storage and Control
  3. File Sharing and Management
  4. Project Management
  5. Version Control

## Group Collaboration Information

* 1. Communication
  2. Document Storage and Management
  3. File Sharing and Management
  4. Project Management
  5. Version Control
  6. Group Strengths
  7. Group Weaknesses

## Appendices

* 1. MOSCOW
  2. Procedures for Project Requirement Changes
  3. Schedule
  4. Requirements Traceability Matrix
  5. Verification and Validation Testing Document

## 1. Introduction

* This document is a software requirements specification for our calendar application, Pro Calendar. This SRS will state requirements as well as describe the purpose and functionality of the application.

### 1.1 Purpose

* This SRS will provide a description of the goals and requirements of Pro Calendar. This allows the development team to start planning with the requirements in mind, as well as allowing the team to test whether or not the project satisfies all the requirements.

### 1.2 Scope

* Pro Calendar will be used as a time management tool. It will allow the user to:
* View their plans for upcoming or previous weeks
* Create reminders and events
* Integrate reminders/events with their Google account (if desired) for cross-platform functionality

### 1.3 Definitions of Acronyms and Abbreviations

* SRS: Software Requirements Specification

### 1.4 Overview

This document is divided into four sections. These sections are

* Introductory Section
  + This section holds the introduction and description of the application.
* Design and Requirements Section
  + This section holds the functions, requirements and API of choice for the application
* Operational Section
  + This section encompasses our operational scenarios and user cases
* Conclusion Section
  + This section includes our collaboration tools and appendix for the document

### 1.5 Business Context

* This application can be used by businesses, as well as individuals. Anyone who needs a calendar application to make plans and design schedules can make use of this app.

## 2. Overall Description

* The Pro Calendar is meant to be used for the sake of organization, and planning. Providing users with an easy to work with interactive calendar. Maximizing the efficiency, and organization, of those using this platform.

2.1 Features/Functions

* The calendar will contain functions such as:
  + Creating new events/dates/reminders
  + Adding information based around the event/date/reminder
  + Adding personal information to personalize the Calendar

## 3. Business Requirements

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| BR1.1 | We must use a Database in order to store the login and other information of users. | M |
| BR1.2 | We must allow users to customize the appearance of the calendar | S |
| BR1.3 | We must allow users to use this application for managing events | M |

## 4. User Requirements

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| UR1.1 | Application will identify users by unique e-mail address or username | M |
| UR1.2 | Users are required to create an account or connect their Google account | M |
| UR1.3 | This Application will work offline based | M |

## 5. Functional Requirements

5.1 Database and Identification

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| FR1.1 | Application will store user information in a database | M |
| FR1.2 | Application will identify users by unique e-mail address or username | M |
| FR1.3 | Application will identify users by Google login | M |

5.2 Customization

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| FR2.1 | Application will allow users to customize the appearance of the calendar | S |
| FR2.2 | Application will allow users to change how many days the calendar displays | M |

5.3 Events

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| FR3.1 | Application will allow the user to create events | M |
| FR3.2 | Application will allow the user to add checkboxes to events | M |
| FR3.3 | Application will allow the user to add text field to events | M |
| FR3.4 | Application will allow the user to enable or disable notifications per event | M |

## 6. Interface Requirements

### 6.1 User Interface

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| IR1.1 | Clear and concise GUI that is easy to understand and utilize but not ugly to look at | S |
| IR1.2 | Simplistic UI interaction, created around the idea of quickly being able to read and use | M |
| IR1.3 | Color-coded events for easy identification. | M |

### 6.2 Hardware Interface

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| IR2.1 | Compatible with Windows | M |
| IR2.2 | Compatible with Linux | M |
| IR2.3 | Compatible with Mac | M |
| IR2.4 | Compatible with Android/IoS | M |

### 6.3 Software Interface

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| IR3.1 | Program will run on latest edition of popular browsers, including Chrome, Edge, Safari, Firefox | M |
| IR3.2 | Compatible with mobile and desktop\* | M |

\* Currently researching different methods for responsiveness such as Google Material Design or Bootstrap (those being two examples) that have responsiveness built into elements. Having these will enable an easier build, as well as enable the team to understand responsive design better.

## 7. API

### 7.1 Kivy API

* Works universally with Windows, Linux and Mac products

### 7.2. Sqlite3

* Works best with offline data storage

## 8. Performance Requirements

#### 8.1. User Identification

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| PR1.1 | Users will be identified via their Google accounts or by accounts established for the Calendar itself. | M |
| PR1.2 | Verification of their account will be required should they incorrectly enter their information too many times. Verification will be required in the form of a confirmation email the first time someone logs on. | M |

#### 8.2. User Profile

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| PR2.1 | Users will have access to their profile information, where they can change the color palette of their calendar, change their password, sign out, and there’s another method of adding events there as well. | M |
| PR2.2 | The user will have access to see their contacts should they add some to the application | M |

8.3. Account Management

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| PR3.1 | .Users will have access to all features. Period updates that include new features will be open to users. | M |
| PR3.2 | Users will be able to add or remove contacts. | M |

8.4. Event Management

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| PR4.1 | Users will have the ability to create as many events per month as they choose. | M |
| PR4.2 | Users will be able to create events during the entire year, such as birthdays and save those so they join the calendar the next year as well. | M |
| PR4.3 | Users will be able to remove events whenever they desire. | M |

## 9. Other Non-Functional Requirements

### 9.1. Security

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| NFR1.1 | Security features will be in place via the email used. For example if you signed in with a Google account, you will receive an email that goes along with new logins. | M |
| NFR1.2 | If logins are incorrect up to a certain point, the prompts for a new password will be asked if not attached to a Google account. | M |

### 9.2. Availability

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| NFR2.1 | Users will be asked to save during their creation of events, especially if they’re making several at certain times. | M |

### 9.3. Performance

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| NFR3.1 | Updates to events, or event creation, once saved will be locked in, but can be updated at any time, with or without Internet connection. | M |

### 9.4. Maintainability

|  |  |  |
| --- | --- | --- |
| ID | Requirement | MoSCoW |
| NFR4.1 | Maintenance will be done periodically, to update or fix issues. | M |
| NFR4.2 | Updates will periodically carry new implemented features. | M |

## 10. Operational Scenarios

### 10.1. Account Creation

* When a user enters into the creation field, they have the option to use their Google accounts, or create an entirely new account. Those who have a Google account can easily just log in, while those who do not will have to create a username, enter a password, enter an email, and their numbers in order to create an account.
* After creating their account, they will be sent a verification email using the email they entered earlier to finish the creation process and have access to the Calendar itself.

### 10.2. Event Creation

* Upon entering the Pro Calendar, users will be prompted to create a new ‘Event’ on the current date, the one they entered the Calendar on, so if the date is 10/14/2020, the prompt will ask if they want to create a new Event for the current date.
* When you toggle over the other days on display in the Calendar, they will also show the same prompt allowing you to create Events for those days too.

### 10.3 Reminders

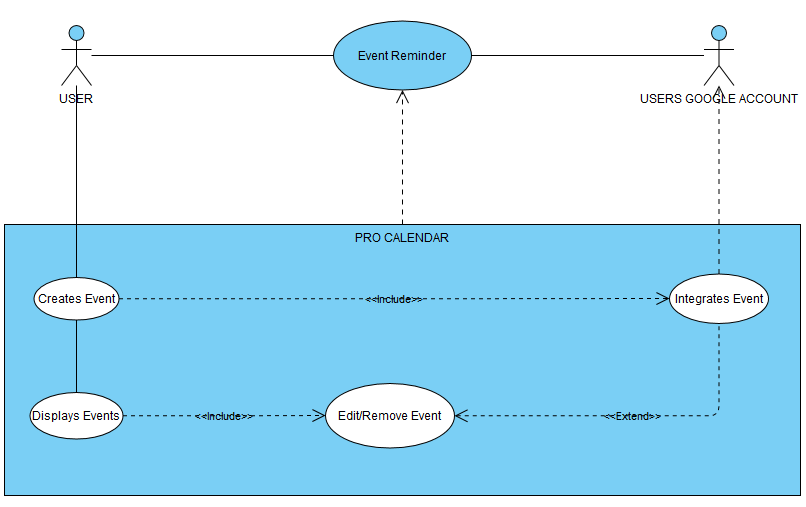
* When you’re done creating an event, you can set up Reminders. These will be notifications that are sent via your email.

## 11. User Case and User Stories

11.1. User Stories

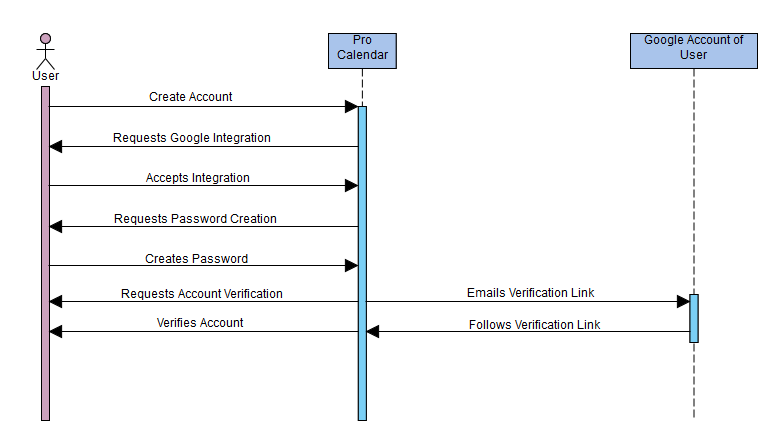
* As a user, I want ease of access during the login process.
* As a user, I want to easily create new events.
* As a user, I want to easily maneuver through the Calendar.
* As a user, I want to easily edit events that I’ve created should I need to.
* As a user, I want to receive notifications when events are coming up so I don’t forget.

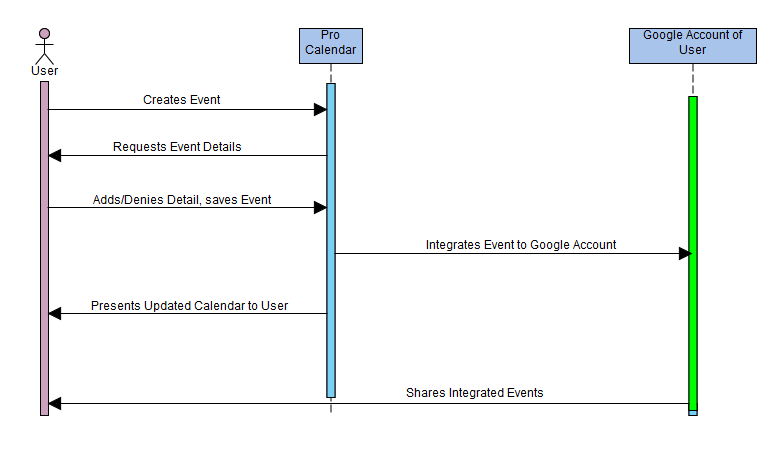
### 11.2 Use Case Diagram



12. Sequence Diagrams

12.1. Account Creation



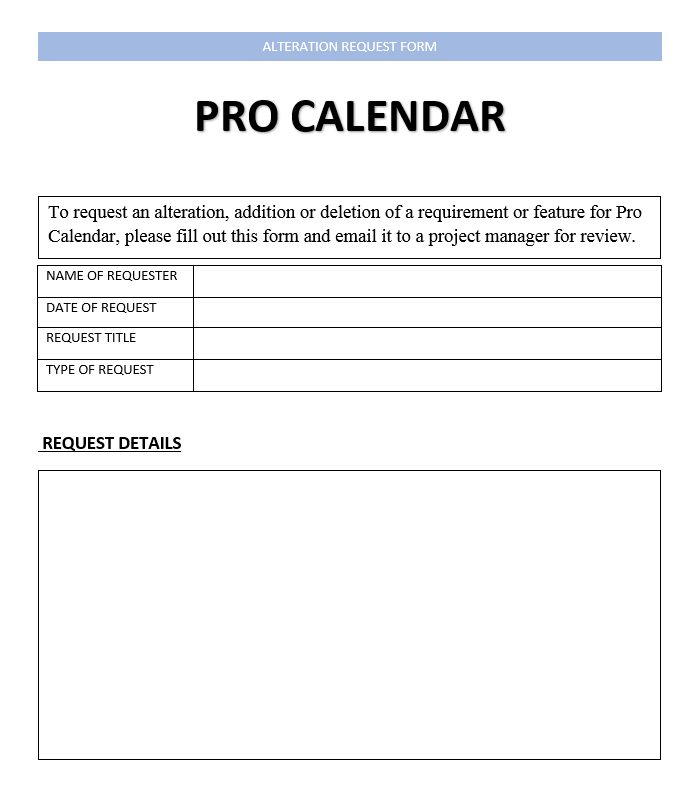
12.2. Event Creation

## 15. Appendices

### 15.1 MOSCOW

* Requirements organized according to MoSCoW prioritization levels (Must Have, Should Have, Could Have, Won’t Have) in order to maximize functionality

### 15.2 Procedures for Project Requirement Changes

* Requesting an alteration, addition, or deletion of Pro Calendar's requirements and features is done through submission of our Alteration Request Form. The form must include the requesters name, the date of the request, a title for the request (i.e. Integration with Outlook) a request type (Alteration, Addition or Deletion) and a short paragraph about the specifics or key notes of the request. This form is then to be emailed to a project manager for review. The project managers will discuss the feasibility of such a request and then decide whether to approve or deny it. If approved, the request is moved to the development team for action.

### 15.3 Schedule

* Software Requirement Specification (draft)
* High Level Design (draft and final)

### 15.4 Requirements Traceability Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Pro Calendar | | | | | |
| Requirement ID | Business Requirement/Use Case | Requirement ID | Functional Requirement/Use Case | Priority | Test Case |
| BR1.1 | We must use a Database in order to store the login and other information of users. | FR1.1 | Application will store user information in a database | 1 | TC001 |
|  |  | FR1.2 | Application will identify users by unique e-mail address or username | 1 | TC002 |
|  |  | FR1.3 | Application will identify users by Google login | 1 | TC003 |
| BR1.2 | We must allow users to customize the appearance of the calendar | FR2.1 | Application will allow users to customize the appearance of the calendar | 2 | TC004 |
|  |  | FR2.2 | Application will allow users to change how many days the calendar displays | 1 | TC005 |
| BR1.3 | We must allow users to use this application for managing event | FR3.1 | Application will allow the user to create events | 1 | TC006 |
|  |  | FR3.2 | Application will allow the user to add checkboxes to events | 1 | TC007 |
|  |  | FR3.3 | Application will allow the user to add text field to events | 1 | TC008 |
|  |  | FR3.4 | Application will allow the user to enable or disable notifications per event | 1 | TC009 |

### 15.5 Verification and Validation Testing Document

**User Identification**

|  |  |
| --- | --- |
| Test Case ID | Test Case Description |
| TC- | Verify if a user can enter their username |
| TC- | Verify if a user can enter their password |
| TC- | Verify if a user is logged in when valid email and password is entered |
| TC- | Verify a user email cannot make multiple accounts |
| TC- | Verify a new user can choose a username |
| TC- | Verify a new user can choose a password |
| TC- | Verify a new user can create a new account |

**User Interface**

|  |  |
| --- | --- |
| Test Case ID | Test Case Description |
| TC- | Verify a user can visit the application home page |
| TC- | Verify a user can see the home page |
| TC- | Verify a user can utilize the ‘sign-in’ button |
| TC- | Verify a user can utilize the ‘create an account’ button |
| TC- | Verify a user can utilize the ‘forgot password’ button |
| TC- | Verify a user can utilize the ‘create event’ button |
| TC- | Verify a user can utilize the buttons for editing events |
| TC- | Verify a user is able to enter details for an event |
| TC- | Verify a user is able to save their events |

**User Documentation**

|  |  |
| --- | --- |
| Test Case ID | Test Case Description |
| TC- | Verify that the Support Contact link is working and displayed in the top left corner of every page |
| TC- | Verify that the FAQ is available under the drop-down menu |
| TC- | Verify that the tutorial is available under the drop-down menu |

**Communication Interface**

|  |  |
| --- | --- |
| Test Case ID | Test Case Description |
| TC- | Verify that users are getting email notifications for their events |
| TC- | Verify that users are getting email notifications about their account status |
| TC- | Verify that users are getting push notifications about their events |
| TC- | Verify that users are getting email notifications about their account payment |

**Account Management**

|  |  |
| --- | --- |
| Test Case ID | Test Case Description |
| TC- | Verify that users can pay to elevate their account privileges |
| TC- | Verify that every user has an ID number tied to their account |
| TC- | Verify that users receive a temporary login code is password is forgotten and for logging in with 2-factor authentication (2FA) |