

Requirements Specification Document

PlanIt

Group Calendar App

Table of Contents

1 Scope	3
1.1 Identification	3
1.2 System Overview	3
2 CSCI Component Breakdown	4
3 Functional Requirements	5
3.1 GUI Client CSC	5
3.1.1 Login Page CSU	5
3.1.2 Main Page CSU	5
3.1.3 Settings Page CSU	5
3.2 Database CSC	5
3.2.1 User Data	5
3.2.2 Group Data	6
3.3 Back-end CSC	6

1 Scope

1.1 Identification

This Software Requirements Specification (SRS) documents the requirements for the group scheduling app, called PlanIt.

1.2 System Overview

PlanIt will be a mobile application built with React Native and NodeJS. This application will be used for group scheduling and will draw from certain APIs such as the Google Calendar API and certain databases including Firebase for user authentication and security. PlanIt will feature many different components, consisting of a Login Page, a profile page displaying the user's private calendar, a sidebar used for viewing the user's current groups, a settings page, a popup to create new events, and a page to create a new group.

2 CSCI Component Breakdown

CSCI PlanIt is composed of the following CSCs:

2.1 GUI Client CSC:

2.1.1 Login Page CSU: Page that allows users to login/create a new account with username and password.

2.1.2 Main Page CSU: Page that displays the user's private calendar, and different buttons that allow for new events, new groups, etc.

2.1.3 Setting Page CSU: Page that allows users to edit profile, log out, change aesthetic features, link Google Calendar, etc.

2.2 Database CSC: Stores user's profile information and group information.

2.2.1 User data CSU: User information shall include username, password, calendar information, friends, linked accounts, and events with their corresponding visible groups.

2.2.2 Group data CSU: Group information shall include group name, unique ID, password, group members.

2.3 Back-end CSC: Stores offline data and updates every time a change happens.

3 Functional Requirements

3.1 GUI Client CSC

3.1.1 Login Page CSU

- 3.1.1.1 The login page shall log users into the app.
- 3.1.1.2 The login page shall allow users to create an account through a username.
- 3.1.1.3 The login page shall allow users to create an account through Google.
- 3.1.1.4 The login page shall allow users to create an account through Outlook.
- 3.1.1.5 The login page shall allow users to create an account through Apple.
- 3.1.1.6 The login page shall allow existing users to log in using their preferred method.
- 3.1.1.7 The login page shall bring users to the main page.

3.1.2 Main Page CSU

- 3.1.2.1 The main page shall hold the user's private calendar.
- 3.1.2.2 The main page shall contain a button that allows users to add events to that calendar.
- 3.1.2.3 The main page shall contain a menu bar with three buttons.
 - 3.1.2.3.1 The menu bar shall feature a settings page button, navigating to the settings page.
 - 3.1.2.3.2 The menu bar shall feature a new group button.
 - 3.1.2.3.3 The menu bar shall feature an existing group button, which will display the user's existing groups in a sidebar.
- 3.1.2.4 The main page shall contain a button that allows users to switch between monthly, daily, yearly calendar views.
- 3.1.2.5 Clicking on events shall open an event settings popup.
 - 3.1.2.5.1 The event setting shall contain a changeable name.
 - 3.1.2.5.2 The event setting shall contain a description.
 - 3.1.2.5.3 The event setting shall allow users to either select an event (fixed amount of time) or a task (unknown amount of time).
 - 3.1.2.5.4 The event setting shall contain a list of groups the event is visible to.
- 3.1.2.6 The main page shall display the name of whatever group the user is in (displays username if not in group).
- 3.1.2.7 The main page shall contain a button where you can see everyone that is in your group.
- 3.1.2.8 The main page shall contain a zoom-in button that shows the calendar on the page only.
- 3.1.2.9 The zoom-in button shall turn into a zoom-out button during zoom-in status.
- 3.1.2.10 The main page shall feature a button that allows users to invite new members to the group.

3.1.3 Settings Page CSU

- 3.1.3.1 The settings page shall have an "edit profile" button.
- 3.1.3.2 The settings page shall have a "log out" button.

- 3.1.3.3 The settings page shall have an option to change calendar themes.
- 3.1.3.4 The settings page shall have an option to change the default size of the calendar with three positions “day, week, month”.

3.2 Database CSC

3.2.1 User Data

- 3.2.1.1 The database shall store the user's username.
- 3.2.1.2 The database shall store the user's password.
- 3.2.1.3 The database shall store the user's current timezone.
- 3.2.1.4 The database shall store the user's events.
 - 3.2.1.4.1 The event shall include its accessibility to groups.
 - 3.2.1.4.2 The event shall include its starting and ending time.
 - 3.2.1.4.3 The event shall include its name.
 - 3.2.1.4.4 The event shall include its description.
 - 3.2.1.4.5 The event shall include a list of users that have accessibility in the rate from “description only/edit only/edit, delete and grant permission to others”.
 - 3.2.1.4.6 The event shall include its repeat pattern like “daily, weekly, monthly, yearly”.
 - 3.2.1.4.7 The event shall include its location.
- 3.2.1.4 The database shall store the user's linked third-party accounts.
- 3.2.1.5 The user can change their username and password at any time.

3.2.2 Group Data

- 3.2.2.1 The database shall store the group name.
- 3.2.2.2 The database shall store the group id.
- 3.2.2.3 The database shall store a password.
- 3.2.2.4 The database shall store the list of members.

3.3 Back-end CSC

- 3.3.1 The backend shall store a copy of all current calendars for offline access.
- 3.3.2 The backend shall connect to Google Calendar API.
- 3.3.3 The backend shall connect to Apple Calendar API.
- 3.3.4 The backend shall connect to Outlook Calendar API.
- 3.3.5 The backend shall store users' account information.
- 3.3.6 The backend shall store users' accounts data (Google and Apple's username and password) which is used to connect to APIs.
- 3.3.7 The backend shall update all calendars to and from the server when it is opened and connected to the internet.
- 3.3.8 The backend shall update the calendar when the “update” button is clicked.
- 3.3.9 The backend shall submit the data to the server when a new event is added on the user's account.

3.3.10 The backend shall fetch data from the database when a group is created/added/changed.

3.3.11 The backend shall support different levels of access permissions for shared calendars (view-only, edit/ manage members, etc.).

3.3.12 The backend shall automatically detect scheduling conflicts for events within the user's calendar or across group members.

3.3.13 The backend shall display event times in the user's local time zone.