CalPal Requirements

CJ Guttormsson, Ben Romano, and Bryan Anderson

What Our Product Is

- Problem:
  - Other calendar apps exist, but are limited
  - Coordinating with others can be difficult
  - No way to find new events easily
- Focuses of our calendar
  - Let users share their events with others easily, and with granular control
  - Allow friends or colleagues to find common time easily
  - Make discovery of nearby events seamless

Glossary

Friend: Someone with whom you share calendar access

Event Visibility:
- Hidden: Not shown to anyone
- Friends: Only shown to friends
- Partial: Show to friends, but hide details
- Locations: Anyone within a radius can see

OAuthToken: Token generated by Google to allow logins

Event info: Attributes of an event, including start and stop time, location, description, and Friends who are invited

Database: the remote backend that stores user's events, friends, etc.

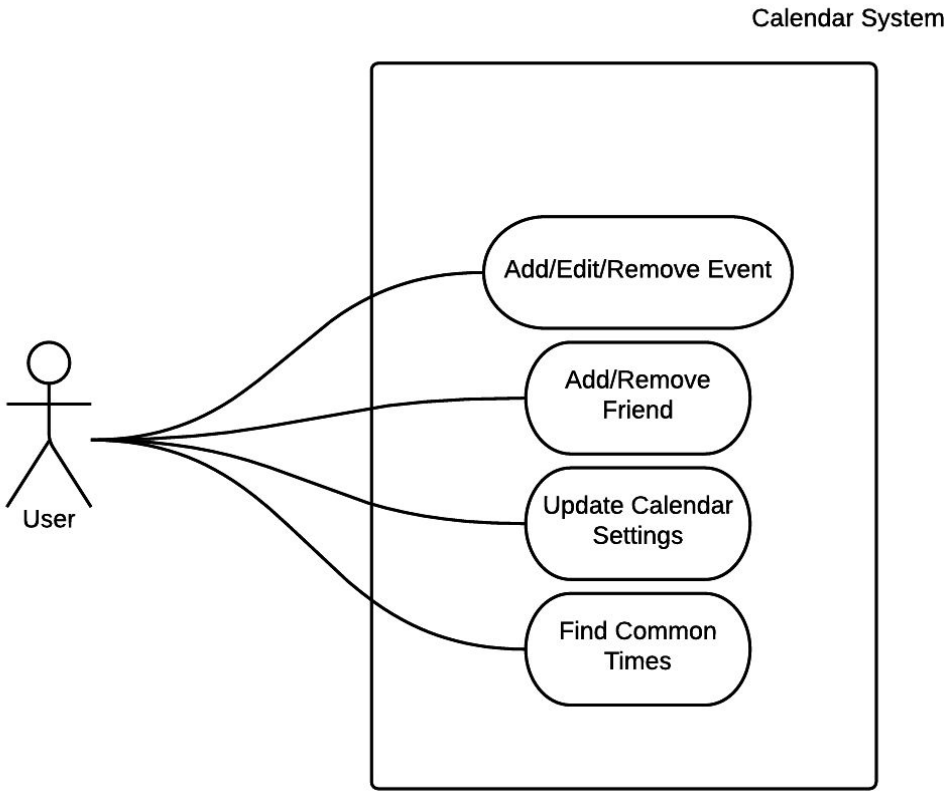
Functional Requirements

- User must have a Google+ account
- User able to login using Google+ and give our app permission to all their calendars; although, only one will be used
- User is able to find friends by email address
- User needs GPS enabled to use location-based features
- User is able to find upcoming events
- User is able to find common times with other users
- User is able to create events

Non-Functional/Performance Requirements

- Calendars must sync within 10 minutes of making a change
  - Includes visibility to those it's shared with
- The app must run without crashing on device with Lollipop or greater
- The app must not drain more battery life than necessary
  - Syncing should be kept to minimum
  - Not use location services too much
- Startup should take no longer than three seconds
- The user should be able to view the calendar by day, week, or month

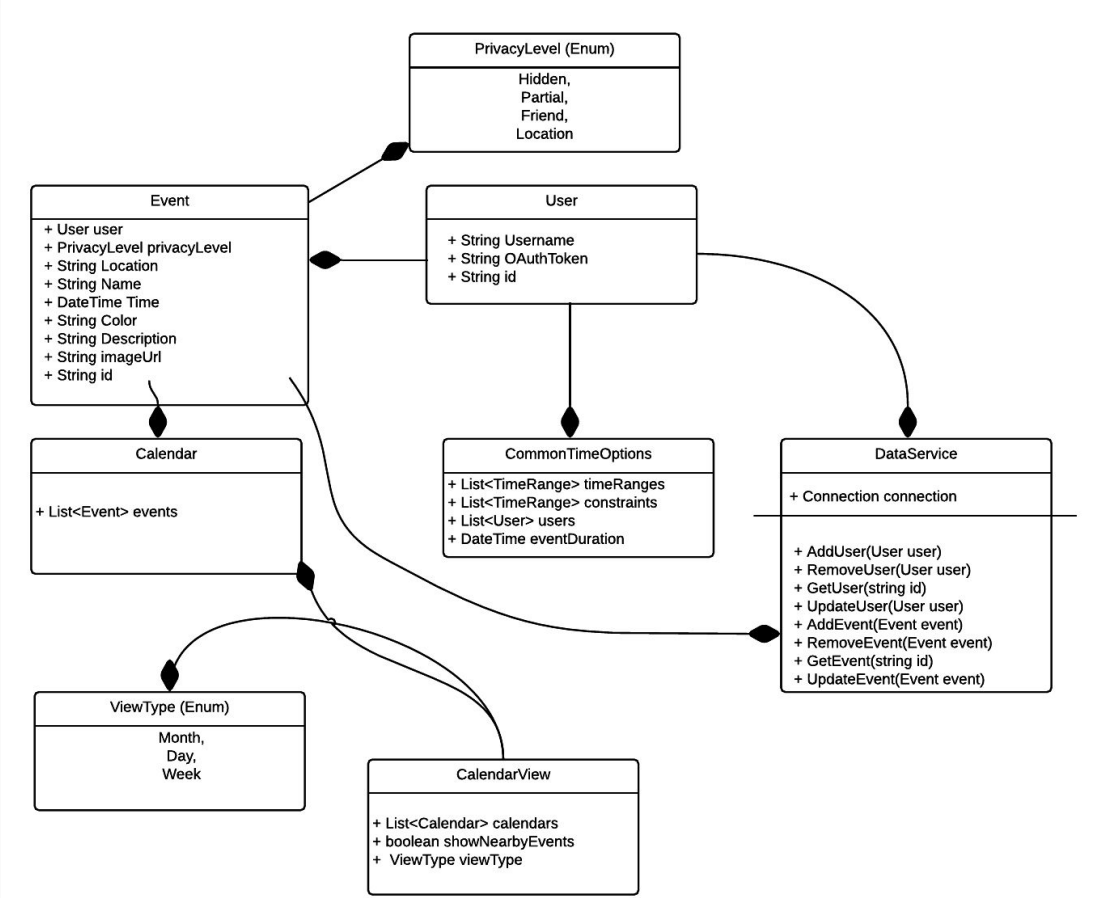
Use Case Diagram



Classes

- Calendar
  - Contains event and user who the calendar belongs to if associated with a user
- CalendarView
  - Stores calendars to display
  - Has display settings (month,day, view)
  - Stores which one is the main calendar
- Event
  - Contains event info (description, times, etc.)
- CommonTimeOptions
  - Used to find common time between friends
- DataService
  - Interacts with database

High-Level Class Diagram



System Architecture

Login Activity

Calendar Activity

Create/Edit Event Activity

Common Time Finder Activity

Friend List Activity

Add Friend Popup

Calendar options popup

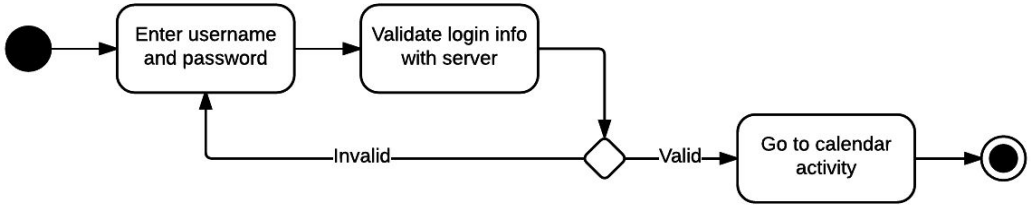
Navbar

Server

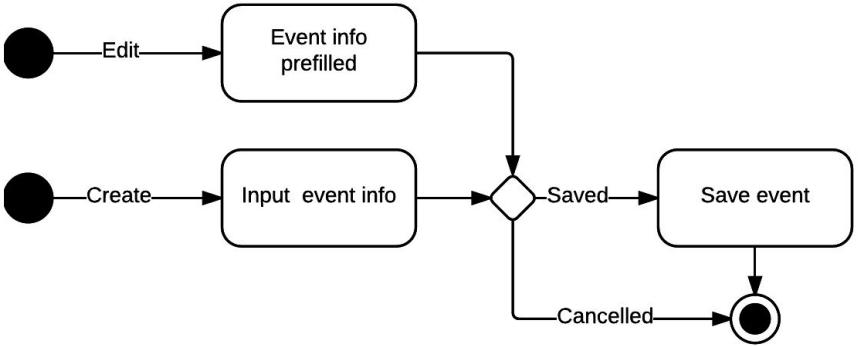
Database

Selected Activity Diagrams (1)

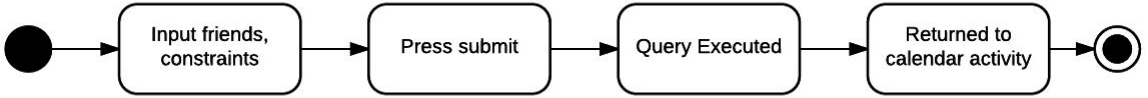
Login



Create/Edit Event

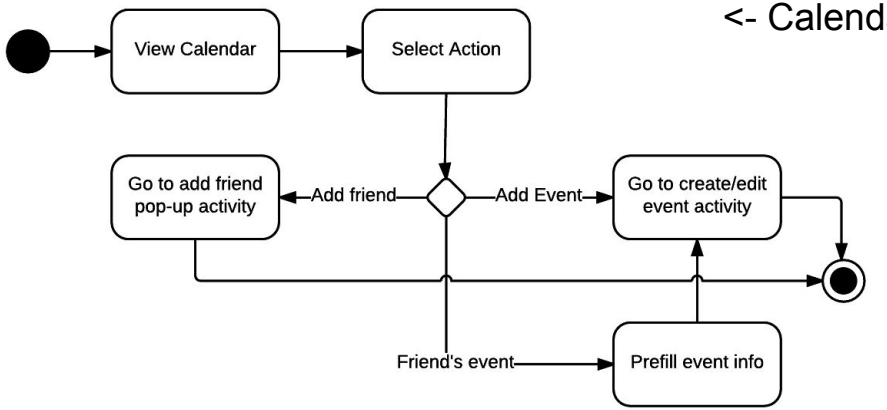


Common time finder

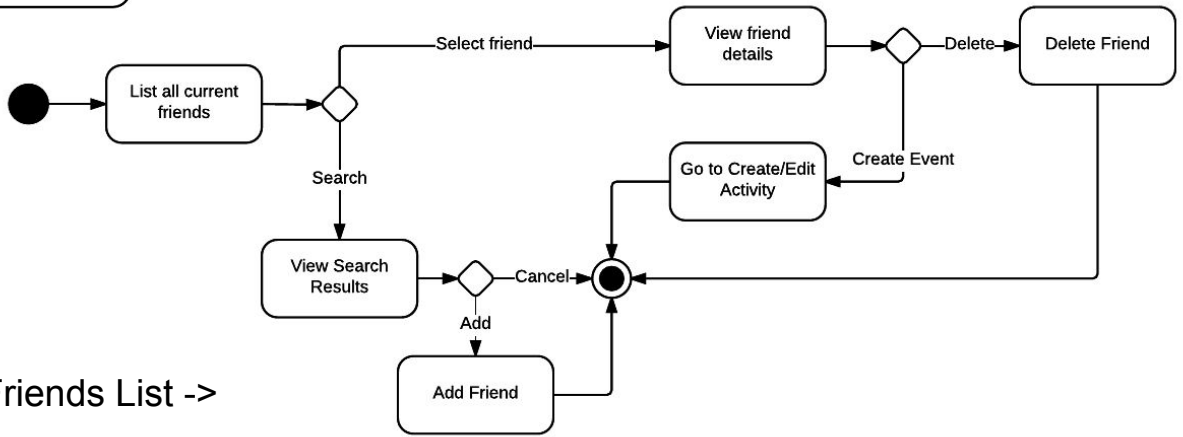


Selected Activity Diagrams (2)

<- Calendar



Friends List ->



Future Features

- Syncing from other services
  - Facebook, school account
- Allow an user to maintain multiple calendars
  - Work, personal, etc.
- Notifications for upcoming events
- Adding pictures to events

Risks

- Difficulties in development
  - Unable to learn Android APIs, Android Studio
  - Unable to create a backend that can
- Difficulties from outside sources
  - Problems with Google OAuth and Calendar APIs
  - Unable to find way to host backend
- Difficulty syncing calendar with our custom events

Summary and Conclusion