---



**GitHub Username**: dorren

# EventHub

## Description

**Write a brief summary of what your app does. What problem does your app solve?**
It's an app to help people to find and organize events. It let users to find events they're interested, and let organizer to reach a broader audience of common interests.

## Intended User

**Who is your intended user? (For example, is this an app for dog owners? Families? Students? Travelers?)**
Anyone with common interests. Like soccer, basketball, opera, foodie, etc.

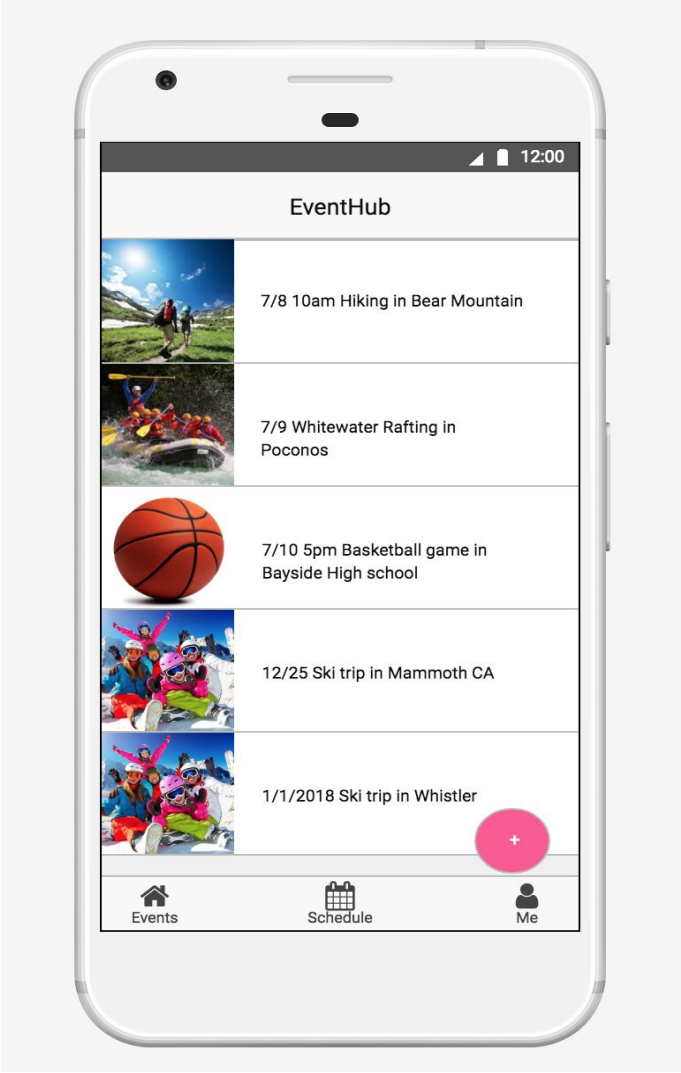
# Features

List the main features of your app. For example:

User's stories

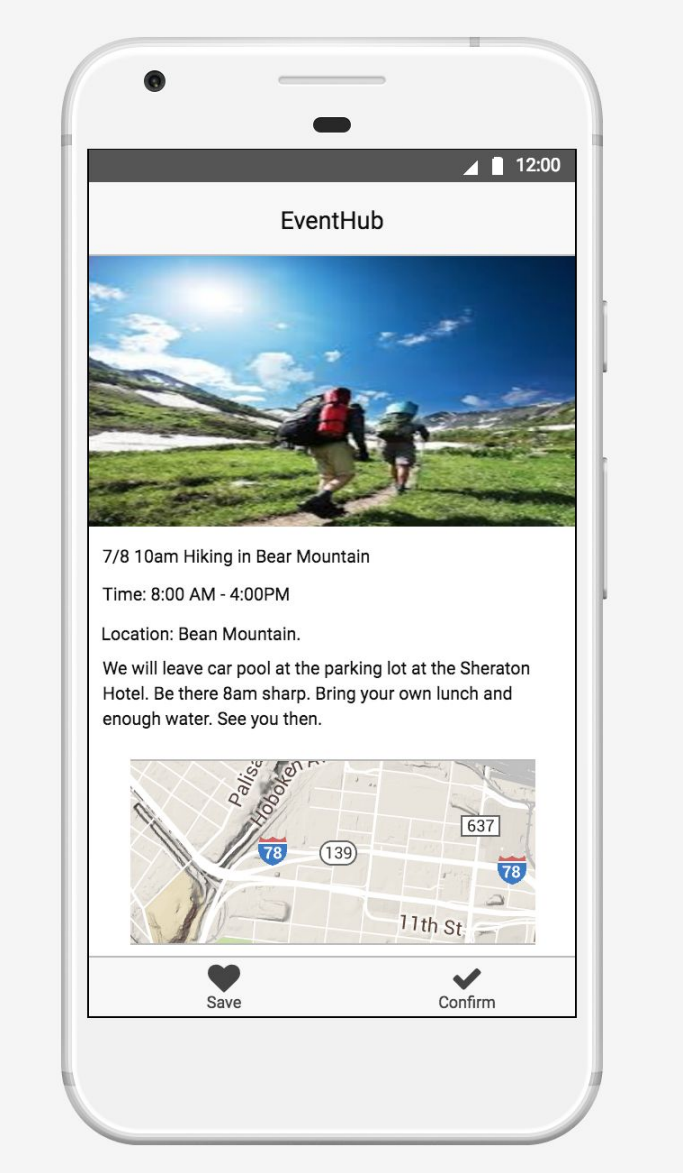
- List events
- Save/bookmark events
- Confirm event
- List saved/bookmarked events
- Create event

## Homepage




- Homepage of the app, listing all available events.
- User can click each event to go to the detail page.

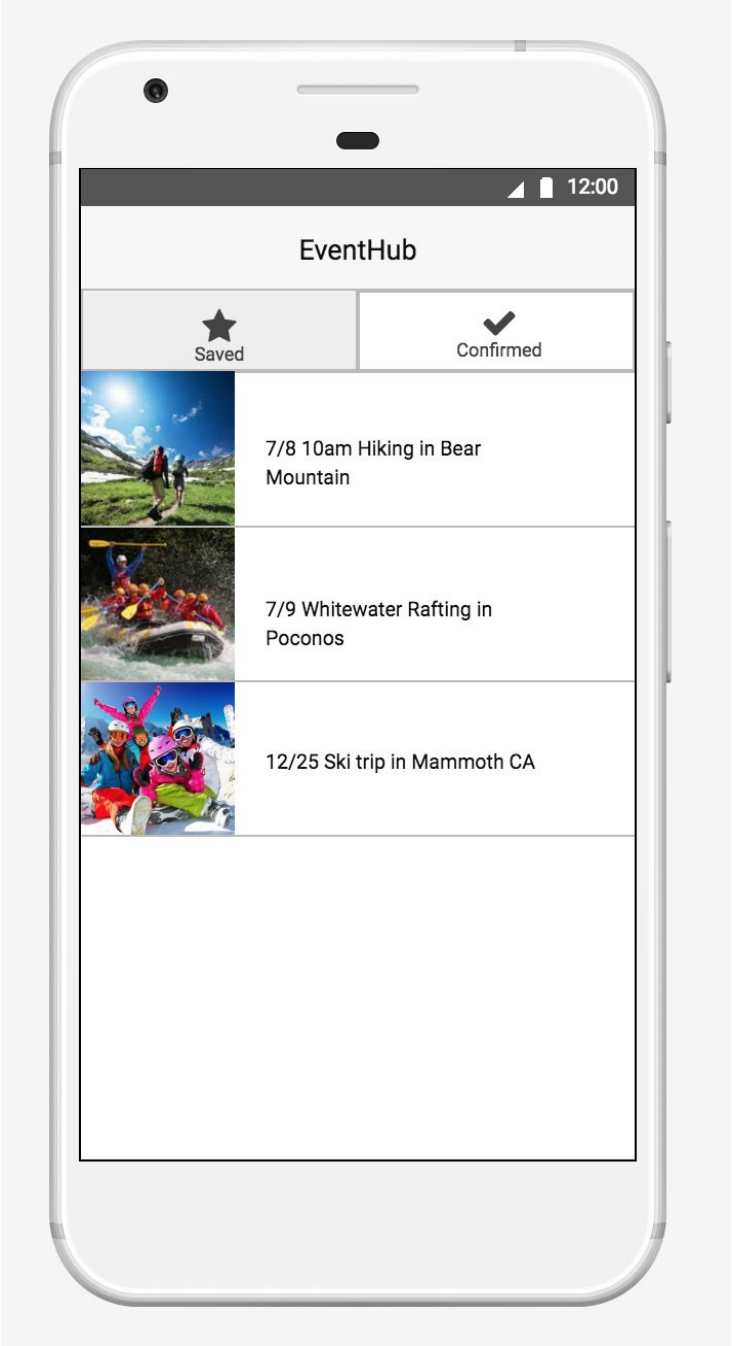
## Event Detail




Detail event page.

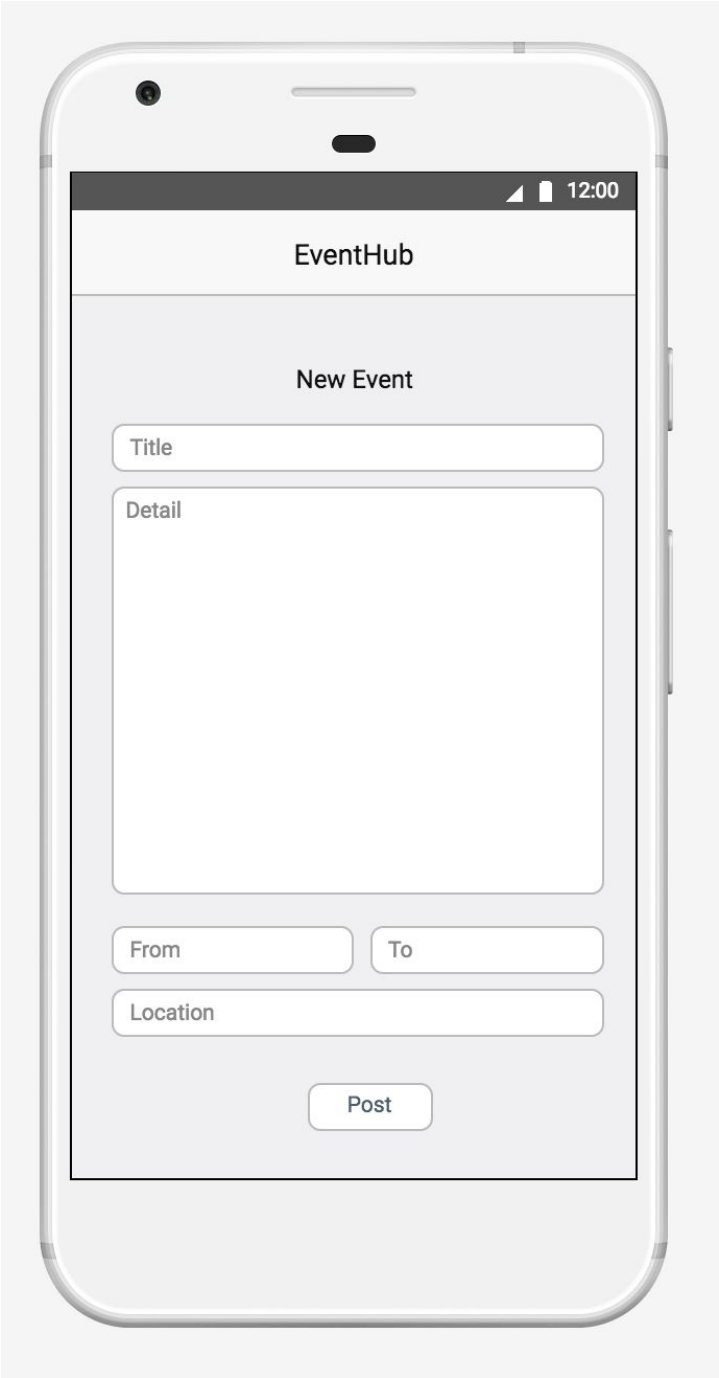
- User can save the event.
- User can confirm participating the event.
- If user is the organizer, it will not have save/confirm button, but will list participants below the map.

## My Schedule



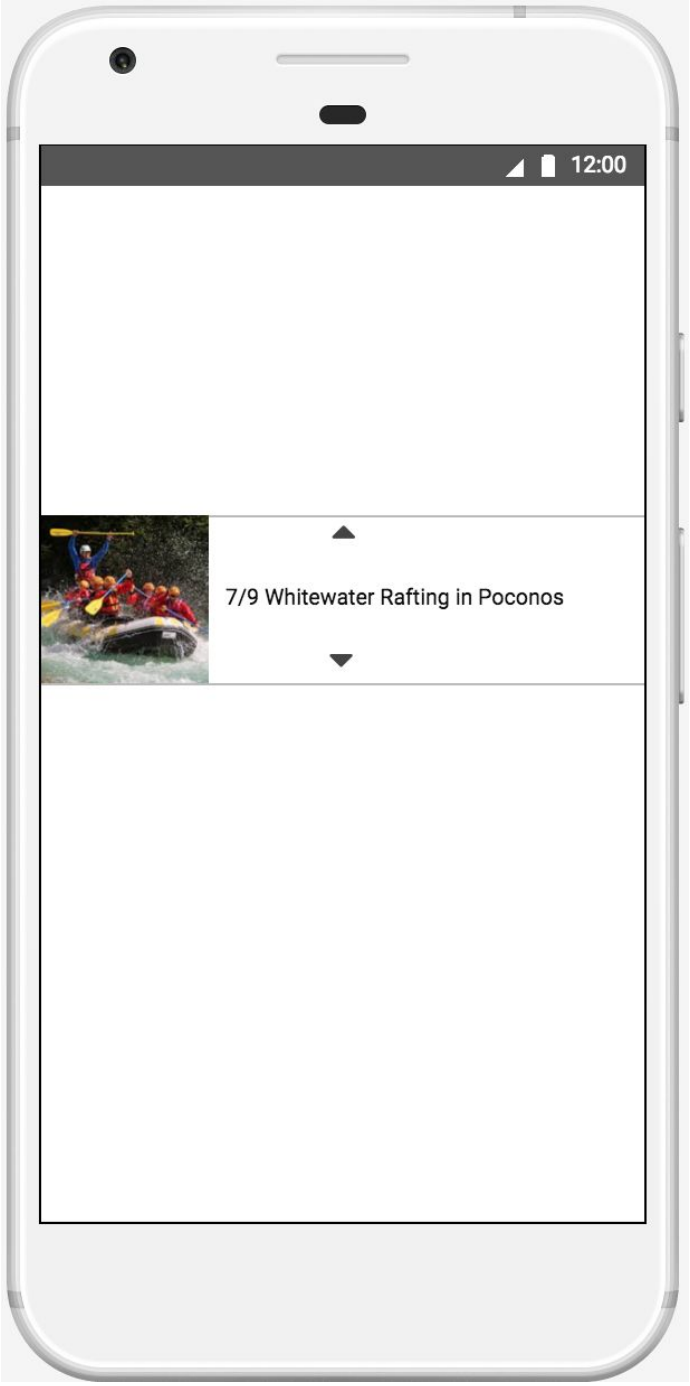

My schedule page. List saved and confirm events.

## Create Event Form




Form used to create event.

## Widget




Widget list all user's upcoming confirmed events.
User can slide up/down to view each event.

## Key Considerations

**How will your app handle data persistence?**

- Backend will be using RethinkDb as data storage.
- Coupled with node.js app to provide the API data access.
- RethinkDb can do push notification, which will call node.js app, which will call Google Cloud Messaging (GCM), which delivers the message to android app.

**Describe any edge or corner cases in the UX.**

If user doesn't have any bookmarked or confirmed events. App will provide a link to send them to the event listing page.

**Describe any libraries you'll be using and share your reasoning for including them.**

- App will use Picasso to render image.
- If organizer provides a location, app will display a google map.

**Describe how you will implement Google Play Services or other external services.**

App will use GCM to deliver real time notification.

## Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

### Task 1: Project Setup

- Setup new AWS instance for backend API server.
- Install RethinkDb, nodejs, nginx webserver.
- Setup sample data

### Task 2:

- Build Content Provider to access API server.

### Task 3:

- Build UI for homepage.
- Build Event list fragment.

### Task 4

- Build UI for Event Detail Page.
- Research and incorporate Google Map.

### Task 5:

- Build UI for My schedule Page.

### Task 6:

- Build UI for creating events.
- Research uploading image.

## Implementation Requirements

Core Platform Development

1. App will validate user input when creating new event.
2. App will provide text information for non-text UI elements for accessibility requirement.

Google Play Service

1. App will use GCM to provide push notification.
2. App may use Admob to fulfill Capstone requirement.
3. App will use Google map to show event location.

Material Design

1. App will use transition between listing and detail page.

Data Persistence

1. App will implement content provider to pull data from custom API server.

2. App will use loader or asyncTask to fetch the API data.