Event Handler

A flexible solution for planning event spaces

Alex Bozeat, Anthony Fappiano, Alex He, Rex Hu, Ronna Min

Summary

Problem

Event organizers usually have to create a floor plan for their event. With many attendees, locations, or time slots to sign up for, it can become very complicated to coordinate. The organizer must ensure that every attendee is accounted for and given the resources they need. While document sharing is easy nowadays, it's still difficult to collaborate on an image or diagram, such as a map for an event. This usually leads to organizers sharing the floor plan and registration options with suboptimal tools, such as email images and Google Forms and Sheets. Alternatively, they may resort to pen and paper signups.

Solution

We aim to create an application for event organizers to create the layout of an event, and allow attendees to select their desired locations. The first step would require the organizer to designate a floor plan in one of two ways: they will be able to upload a floor plan, or draw up their own in the application. Next, the event organizer can visually designate available spots in the room, e.g. tables or seats which attendees can then sign up for; the layout can be modified at any time by the organizer. The application will ensure that there are no conflicts during spot selection, similar to the seat selection process for an airline. Finally, attendees will be able to see relevant information about the event, and the event organizer will be able to view the entire layout, as well as a full list of attendees with registration details.

Description of Stakeholders

Event Handler is an application meant for organizing any event. It will help flexibly arrange the physical space and layout of the event, whether it is occupied by tables, desks, open space, or anything else. This requires only two user groups: the event organizers and attendees. These user groups can modify and sign up for events, respectively.

Group 1: Event organizers

Event organizers are responsible for coordinating attendee signups, and making sure that that attendees are happy with the spot they signed up for. Event Handler would streamline the process and make it easier to share online. Organizers have the option to use an existing floor plan or image, as well as the option to create one from scratch. They can easily invite attendees to select their locations. They can also see an aggregate view of the map and attendee details.

Group 2: Event attendees

Event attendees want to sign up for the best location possible. This means that they want to see a layout of the room or location. Event Handler offers a visual view, so they can figure out which available spot best accommodates them. Once an event organizer has invited them to an event, they can conveniently select their desired location, and the app will ensure that no two attendees can select the same spot. They will also be able to access any event details that the organizer provides.

Similar Existing Applications

There are various existing options for table planners:

- Social tables (https://www.socialtables.com/)
    - Specifically for event planners; guest listing, organize seating
- Table Planner (https://tableplanner.app)
    - Create multiple events
    - Guest listing and seating; allows for collaboration on the entire event
- Allseated (https://www.allseated.com/)
    - 3D layout viewing; mainly intended for curating the physical space; tabletop and buffet displays
    - Guest listing and seating, with meal types, RSVP's, notes
- Magic Table Planner (https://www.magictableplanner.com/)
    - Guest listing and seating, RSVP form and tracking
    - Designs can be exported/shared
- WeddingWire (https://www.weddingwire.com/)
    - Geared toward weddings, organizes seating at tables

Based on this list, there is a lot of competition. However, we have not found a FOSS application that helps with the communication between organizers and attendees for less casual events. Career fairs, farmer's markets, and other community events require more coordination. We would like our application to cover most general cases, and become the go-to for users who do not want to pay for a solution, or would like more flexibility.

Functional/Non-functional requirements

Functional requirements

The main technologies that will be used come from the MEAN stack: MongoDB, Express.js, Angular, and Node.js. Event Handler needs to be able to design the floor plan of the event or upload an image if the user made one beforehand. Simple shapes and labels will be used to accomplish this and it will be saved as some sort of picture that will be attached to the event. We can enable the user to design a floor plan with technologies such as HTML5's Canvas API or jQuery draggable elements with snapping functionality. It should then be able to match people, companies, etc. to the created labels from the floor plan. There should be a sign up feature before being able to create an event. To prevent random people from joining the event, there should be an option to only join using a link or by whitelisting certain accounts. The event information is stored in the database. Specifically, information such as name of event, time, host, maximum people, connection between attendee and label, usernames, passwords, and emails.

Non-functional requirements

The website design should be simple and easy to understand what is on the page on both desktop and mobile. The floor plan designer should be as intuitive and simple as possible because it should be able to be used for many different events. If the user decides to upload their own image, it should let the user know what is accepted as images. Signing up

should be as pain free as possible because it is required for many of the features of the website.

Project Timeline

The first stage of the project after coming up with an idea is to setup a GitHub repository and assign roles. We will also need to plan out the application by creating wireframes. This will all be completed with the proposal due on 1/24. After that, we can start creating the frontend, of which the majority should be finished by spring break so that we can demo a prototype. While the frontend is being developed, the backend team can create the database structure and begin implementing API endpoints. This should be done alongside the frontend and also be completed by around spring break. After the frontend and backends are completed, they can be merged together by implementing the API calls. The goal is to have a working application by the beginning of April. This gives us the month of April to test the app, fix any bugs, and work on the final presentation.

Basic Site Structure

Login & Signup

- Creation of user accounts
- Login and edit user profile

Event Management

- View hosted events
- View attended events
- View event invitations

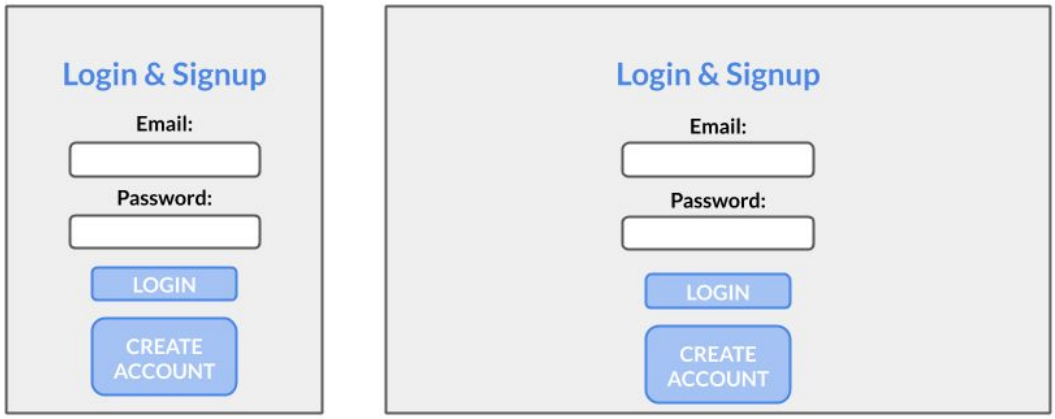
Event Creation

- Edit event details
- Add/create map
- Invite/remove attendees
- Add/Edit form for attendee registration

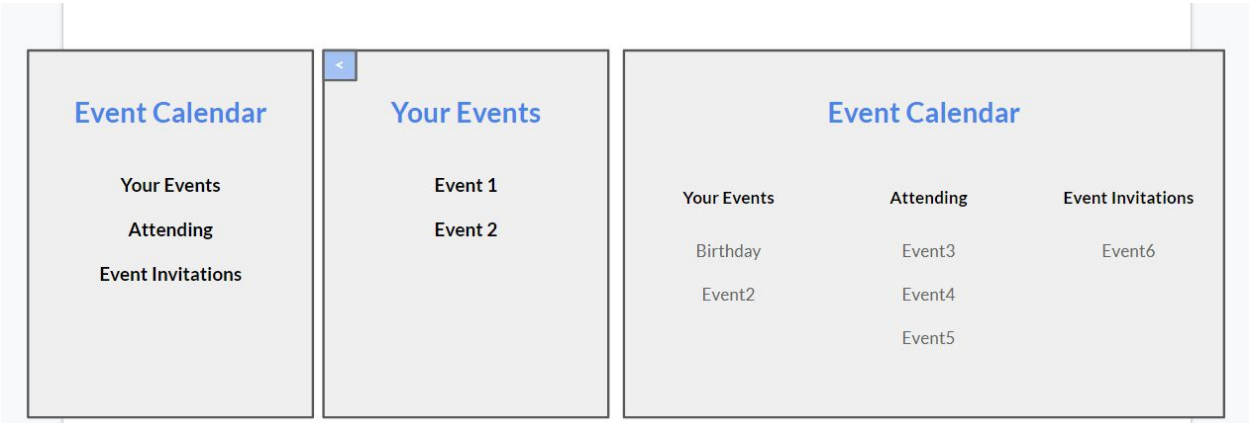
Attendee View

- View event details, map, other attendees
- Select/register for a spot on the map

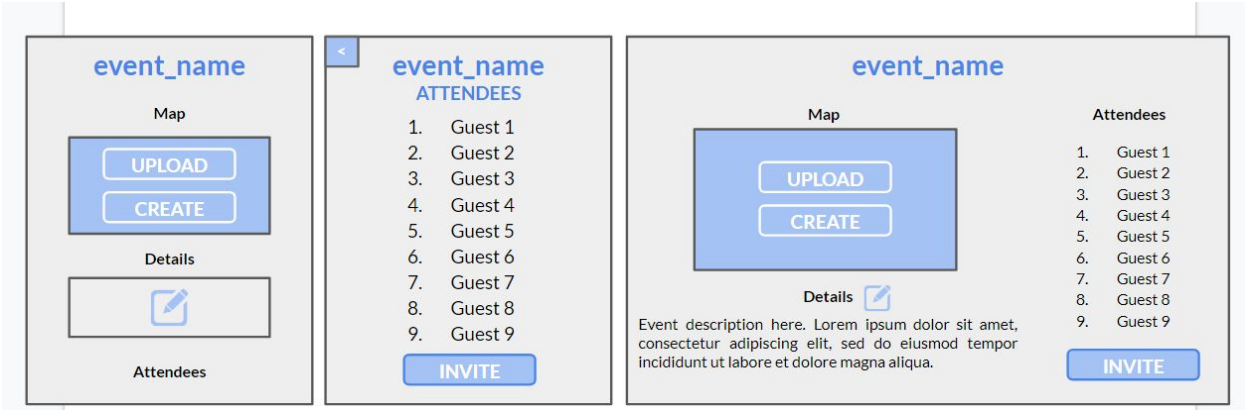
Wireframes



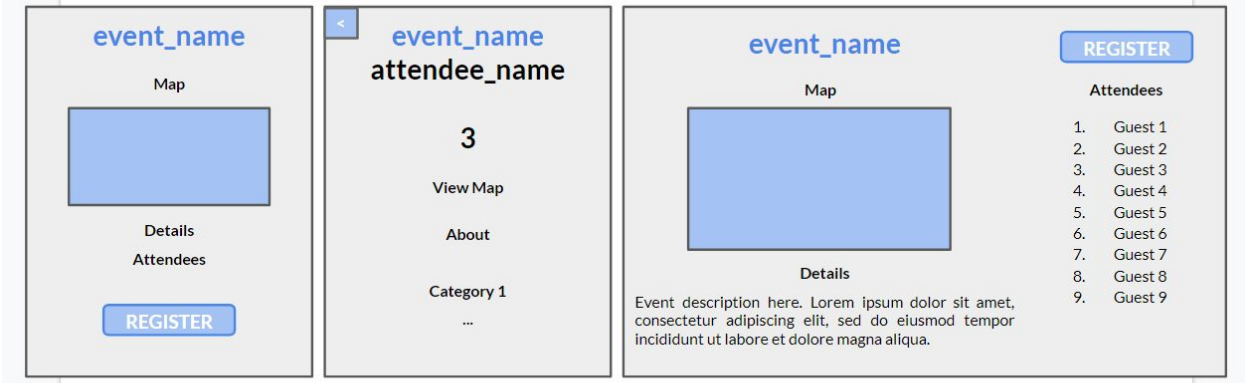
Login & Signup



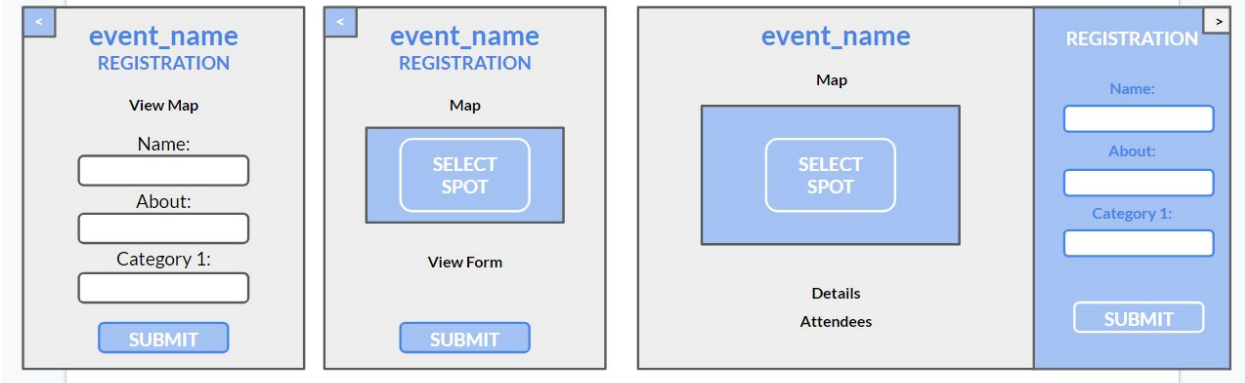
Event Management



Event Creation



Attendee View



Attendee Registration

Future Plan

If our project timeline allows, we would like to implement the following stretch goals:

- Include amenities for different locations, e.g. windows, power outlets
- Options for exporting floor plans, e.g. posters, pamphlets
- Drawing a large rectangle to insert multiple tables