

Campus Event Map Application

Project Features List

Group 4 - Team Murphy's Law

Project Features changes and Priority:

Campus Event Map

Conveniently see all events that are occurring on the CU Boulder Campus with a map view with pins above event location. The view of events will change based on the current date.

1 . Event Details

Clicking on an event will display its location, time, and event organizer. Will also give user option to follow the event by adding it to their "Favorites", RSVP, and have easy access to event details and updates. A database has been created.

2 . Event List Page

See a list of all events based on date, location, and department. These will include events with the engineering center, music department, Program Council, Arts and Theatre Department etc. Not started

3 . Event Search Function

Be able to search and print out a list of events by either event name, organization name, location, or department. Not Started

4 . Event Application Process

The app will provide the application process to properly and orderly create events for any user on the service. They can customize their event, including description, when and where it occurs, how long it lasts, and how many times it will occur. When the event is approved, the creator is able to set exclusivity, send messages to participants, and cancel the event if they desire. HTML has been created.

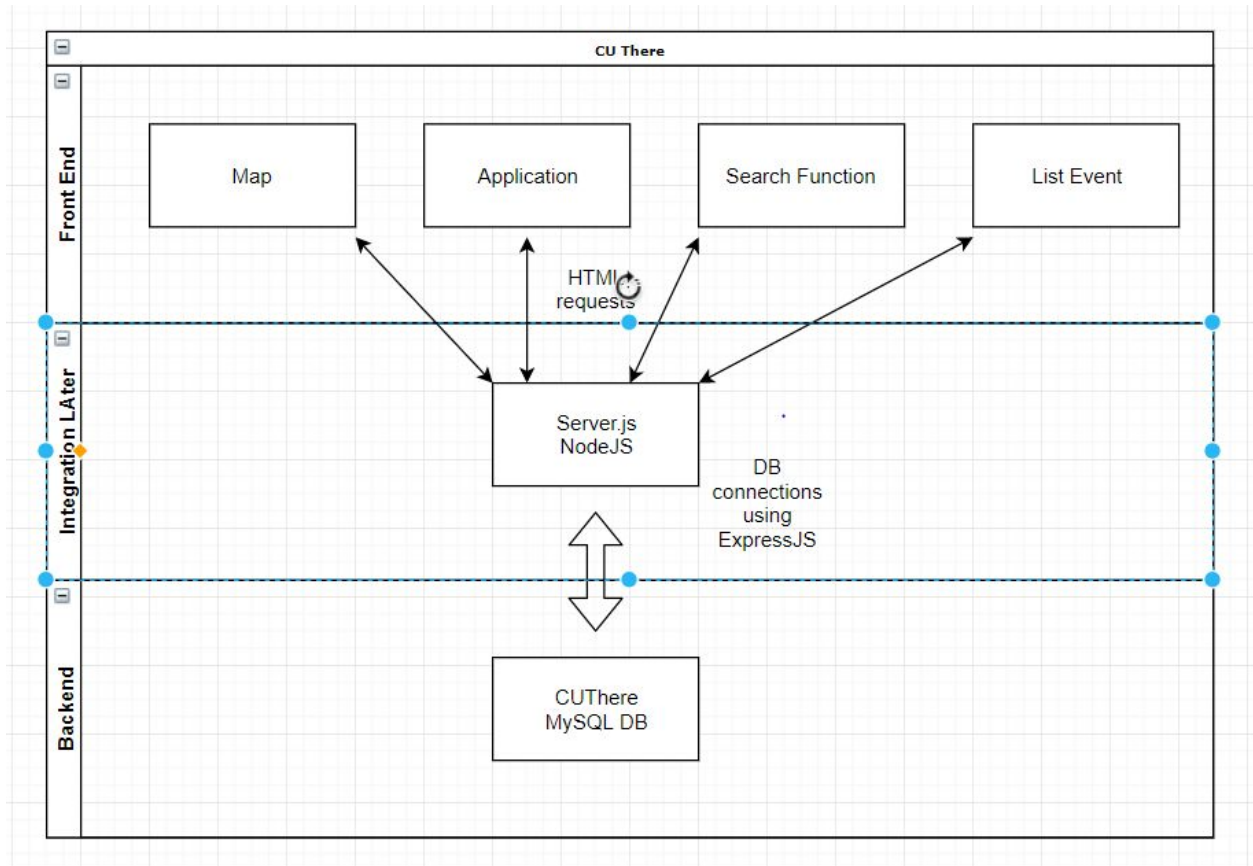
5 . Side Navigation Bar

A panel that can be accessed by pressing a button in the corner of the application. The panel will have navigation buttons to different pages of the application. Functionality has been completed; the final step is to decorate.

6. Notifications

The app will provide notifications which: tell the user upcoming events, notify the user when events to which they've RSVP'd that have been canceled, and alert the user of any event from a favorited org or department. Not Started

Architecture Diagram



Web Service Design

We will be using the Google Maps API to implement a functional map into our website. Events will be queried from our database and exist as pin on our map. A heat map will be used to display event density of specified area. We may also implement a location search function through the Google Maps API.

Database Design

We will be holding all user information and event details within the database. This will allow the HTML pages to directly query from the database whenever there is an html request. Event will also be tied to specific event organizers

Database:

eventDetails table

- eventID: Primary KEy
- eventName [VARCHAR(50)]: Name of the event
- dateOfEvent [DATE]: Date of when event occurs
- timeStart [TIME]: The time in which the event will begin
- timeEnd [TIME]: Time when event will end
- eventDescription [VARCHAR(500)]: Description of event
- organizerID [INT]: Foreign Key
- buildingID: Foreign Key
- rsvp [BOOL]: Whether or not a reservation is necessary before attending the event.

organizers table

- organizerID [INT] : Primary key for the organizer table
- organizeName [VARCHAR(50)]: Name of the organizer
- publicEmail [VARCHAR(100)]: The organizer Email, not required
- publicPhone [VARCHAR(13)]: The organizer phone number, not required
- pictureLink [VARCHAR(150)]: Profile picture of the organizer

locations table

- buildingID [INT]: Primary key
- buildingName [VARCHAR(100)]: Name of the building
- roomNumber [INT]: Room number of the building
- buildingNameAbbv [VARCHAR(4)]: Abbreviation of the name of the building

users table

- userID [INT] : Primary key
- userName [VARCHAR(35) : Username of user
- password [VARCHAR(16)] : Password of users
- email [VARCHAR(50)] : email of user
- DOB [DATE] : Date of birth of user
- firstName [VARCHAR(10)] : First name of user
- lastName [VARCHAR(10)] : Last name of user