RUNTIME WARRIORS

System Design Document

Anabelle Hsiao · Jeremy La · Ricky Su · Mohamad El Kadri · Nevin Wong · Ivy Wills

Table of Contents

Frontend CRC

Events Page	
Parent/Subclasses: n/a	
Responsibilities: ● Get an instance of all events from the backend ● Display said events to the user in a compact view	Collaborators: ● Search ● EventsDisplay

EventsDisplay	
Parent/Subclasses: n/a	
Responsibilities: ● Given any collection of events, display them all to the user in a compact view	Collaborators: ● EventCard

EventCard	
Parent/Subclasses: n/a	
Responsibilities: ● Given an instance of a singular event, display it to the user in a compact view	Collaborators: ● n/a

Add Events	
Parent/Subclasses: n/a	
Responsibilities ● For a user to add an event to the database. ● Allows user to specify title, description, location, time, date, category, and whether or not the event is intended for women only	Collaborators ● Side bar ● After an event is added it will appear on the event page of all users. ● Can be added to the liked events page by any user. ● If a user attends this event it will

	appear in the history page

Liked	
Parent/Subclasses: n/a	
Responsibilities: ● Provides users the ability to interact with the like and dislike button for each event	Collaborators: ● Sidebar ● LikesDAO

Login Page	
Parent/Subclasses: n/a	
Responsibilities: ● If registered 1. the page provides the user the ability to log into their account 2. The user is able to access user-only features once logged in, such as join events,get notifications,etc ● If not registered 1. The page is able to redirect to the register page, and the user can register	Collaborators: ● Sidebar ● User ● register

Register	
Parent/Subclasses: n/a	
Responsibilities: ● Updates global state when user clicks outside of Register. ● Updates user info state (username, password, email, firstname, lastname) per keystroke. ● Posts user info when submit button	Collaborators: ● RegisterContext ● User

is clicked.	

RegisterContext	
Parent/Subclasses: Context	
Responsibilities: ● Stores global state to be used in other classes.	Collaborators: ● n/a

Sidebar	
Parent/Subclasses: n/a	
Responsibilities: ● Act as a router between the different pages of the application.	Collaborators: ● Events Page ● Liked ● Login ● Register

Backend CRC

Search	
Parent/Subclasses: n/a	
Responsibilities: ● Return an instance of every event in the database ● Return an instance of a user from the database given their id	Collaborators: ● MongoDB eventual Database

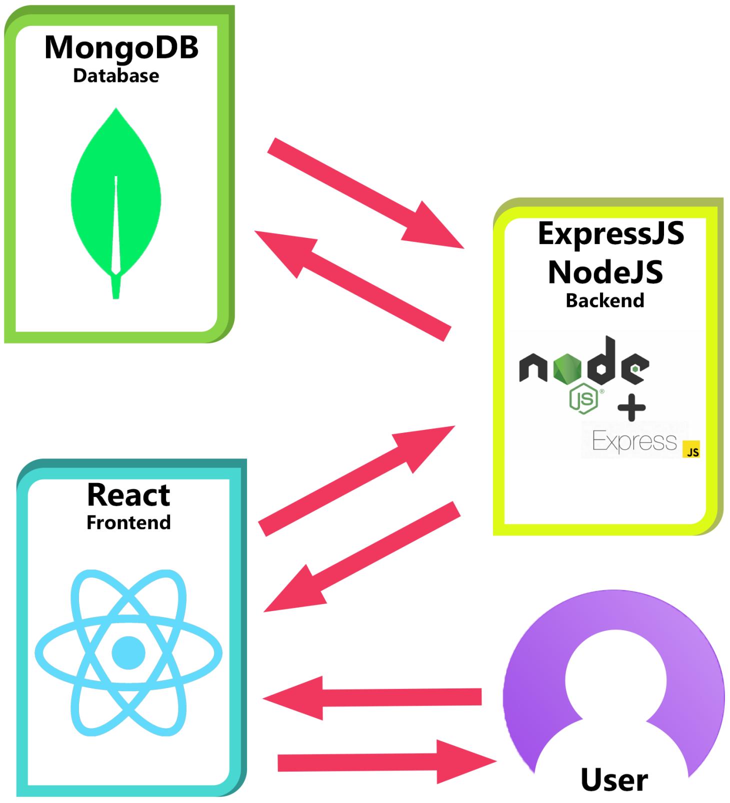
MongoDB eventual Database	
Parent/Subclasses: n/a	
Responsibilities: ● Provides collections of data containing all the information users interact with while using the application.	Collaborators: ● n/a

Event	
Parent/Subclasses: n/a	
Responsibilities: ● Provides a model for events that users can create, like and sign up for. ● Defines how the event will be formatted in the MongoDB database *eventual* ● Provides the format that events are returned when called from the database to be displayed/interacted with in the frontend.	Collaborators: ● MongoDB eventual Database

User	
Parent/Subclasses: n/a	
Responsibilities: ● Provides a model for users that contains their username, encrypted password, email, and name object with first and last name. ● Defines CRUD operations for users.	Collaborators: ● MongoDB eventual Database

LikesDAO	
Parent/Subclasses: n/a	
Responsibilities: ● Provides functionality to create and delete a like object from testLikes collection in MongoDB. ● Provides functionality to update the num_likes field in each object of testEvents collection in MongoDB.	Collaborators: ● MongoDB eventual Database

Software Architecture



We will be building our project on a 3-tier architecture using the MERN Stack. Having as a 1st tier React which is responsible of the front end web application, 2nd tier ExpressJS and NodeJS acting as a middleware responsible of being the runtime environment and server application framework and at 3rd tier MongoDB which is our No-SQL database.
Reference to the 3-tier architecture: https://www.mongodb.com/mern-stack