CSCC01 - Just on Time
System Design Documentation

Arya Sharma, Bhoomi Patel, Sanan Rao, Virthiya Uthayanan, Yuto Omachi

Table of Contents

CRC Cards Description

Backend

userRegisterStratergy

Collaborators	Responsibilities
passport-local bcrypt customerModel	Validate and authenticates request body Add user to the customer model (register user)

organizerRegisterStratergy

Collaborators	Responsibilities
passport-local bcrypt eventOrganizerModel	Validate and authenticates request body Add user to the eventOrganizer model (register organizer)

userController

Collaborators	Responsibilities
passport customerRegisterStratergy	return status code return appropriate message

eventController

Collaborators	Responsibilities
passport event model Mongoose fuse	return status code return appropriate message Has different methods for the event responsibilities like search, getOrganiser events etc.

adminController

Collaborators	Responsibilities
passport	return status code

organizerRegisterStratergy	return appropriate message

userRoutes

Collaborators	Responsibilities
express customerController	Handles post and get request for customer

eventRoutes

Collaborators	Responsibilities
express eventController	Handles post and get request for events

adminRoutes

Collaborators	Responsibilities
express adminController	Handles post and get request for admin

Frontend

Header

Collaborators	Responsibilities
react bootstrap userButton logo header.css search	Creates a navbar(header) with the just on time logo and a button to go between login and signup.

Userbutton

Collaborators	Responsibilities
react bootstrap	Creates a button with a drop down to navigate between login and signup.

Login

Collaborators	Responsibilities
react bootstrap	Creates a login page with necessary fields required for login and handles error sent by backend

Signup

Collaborators	Responsibilities
react bootstrap	Creates a signup page with necessary fields required for signup and handles error sent by backend

CustomerHome

Collaborators	Responsibilities
react reactDom login signup header	Render the home page with the header and adds routes so that other pages can be rendered when user navigates from one page to another.

Search

Collaborators	Responsibilities
react reactDom search RenderSearchCards spinner	Render the search page with the events based on the search term

CreateEvent

Collaborators	Responsibilities
react reactDom spinner uploadImage alert creatEventform	Render the create event page with a form to be filled to create a new event

CustomerInfo

Collaborators	Responsibilities
react reactDom inputValidator settingSidebar react-icons	Render the customer info page which displays all the info about the user and it can updated

dashboard

Collaborators	Responsibilities
react reactDom Modal verifyEventForm	Render the admin dashboard which shows the unverified events and organizers that are to be verified

resetPassword

Collaborators	Responsibilities
react reactDom spinner axios InputField	Render the restPassword page which lets the user reset their password.

organize/main

Collaborators	Responsibilities
react reactDom carousel eventCard	Render the main page for the organizer it shows all the info regarding the organizer's events

eventInfo

Collaborators	Responsibilities
react reactDom spinner ImageSlider react-icons	Render the event info page, this page renders all the info about the events and lets the user place a bid as well.

System Architecture Design

We used MERN stack to make a web application. In the backend we decided to use node and express and in the frontend we have used React, a front end JavaScript library for building user interfaces based on UI components.

We have used MongoDb since we wanted a relational database, MongoDb is a NoSQL database.