

# CSCC01 - Just on Time

## System Design Documentation

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

## Table of Contents

### CRC Cards Description

- Backend
  - userStrategies
  - passportController
  - userController
  - adminController
  - eventController
  - eventRoutes
  - adminRoutes
  - userRoutes
  - userVerification
  - authentication
  - eventValidation
  - userValidation
  - httpResponse
- Frontend
  - Header
  - UserButton
  - Login
  - Signup
  - customerInfo
  - settingSidebar
  - Spinner
  - AdminDashboard

### System Architecture Design

## CRC Cards Description

### Backend

userStrategies

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

#### passportController

Collaborators	Responsibilities	Parent
userModel userRegisterStrategy userLoginStrategy	combines all the passport strategies and the serializing together so that it can be easily accessed in the backend.	userStratergies

#### UserController

Collaborators	Responsibilities
passport expressValidator httpResponse verificationToken passportController	Has functions for login user, register user, logout user, verifyEmail, resendCode, changing password, updateInformation and returns appropriate status code and messages.

#### AdminController

Collaborators	Responsibilities
passport userModel eventModel httpResponse	Updates unverified event and organizer status and returns appropriate status code and messages.

#### eventController

Collaborators	Responsibilities
ImageService path express validator eventModel httpResponse	Creates, updates and returns(GET) events with appropriate status code and messages.

## eventRoutes

Collaborators	Responsibilities
eventController authentication eventValidationSchema ImageService	Handles post and get request for event

## adminRoutes

Collaborators	Responsibilities
authentication adminController	Handles post and get request for admin

## userRoutes

Collaborators	Responsibilities
authentication userController	Handles post and get request for admin

## userVerification

Collaborators	Responsibilities
verificationToken crypto	Util for sending verification email to users

## authentication

Collaborators	Responsibilities
httpResponse	Middleware that checks that users are authenticated.

## eventValidation

Collaborators	Responsibilities
express validator	Validates the info being added to event based on event schema

## userValidation

Collaborators	Responsibilities
express validator	Validates the info being added to user based on user schema

## httpResponse

Collaborators	Responsibilities
	A util for sending standard status code and error messages throughout the backend

## Frontend

### Header

Collaborators	Responsibilities
react bootstrap userButton logo header.css	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
--------------------	---

## Index

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.

## customerInfo

Collaborators	Responsibilities
react bootstrap userButton	allows a registered user to view and update their personal information

## settingSidebar

Collaborators	Responsibilities
react bootstrap customerInfo	creates a display that allows a logged in user to navigate between the account settings pages

## Spinner

Collaborators	Responsibilities
react bootstrap signup login	renders a loading animation whenever the pages are loading

## AdminDashboard

Collaborators	Responsibilities
---------------	------------------

react bootstrap	renders the unverified events and organizers for the admin to verify
--------------------	--

#### OrganizerSignup

Collaborators	Responsibilities
react bootstrap OrganizerSignupForms	renders the signup page for user to register as an organizer

#### OrganizerSignupForms

Collaborators	Responsibilities
react bootstrap Spinner	Creates the multiple signup forms needed for organizer signup

## 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.

