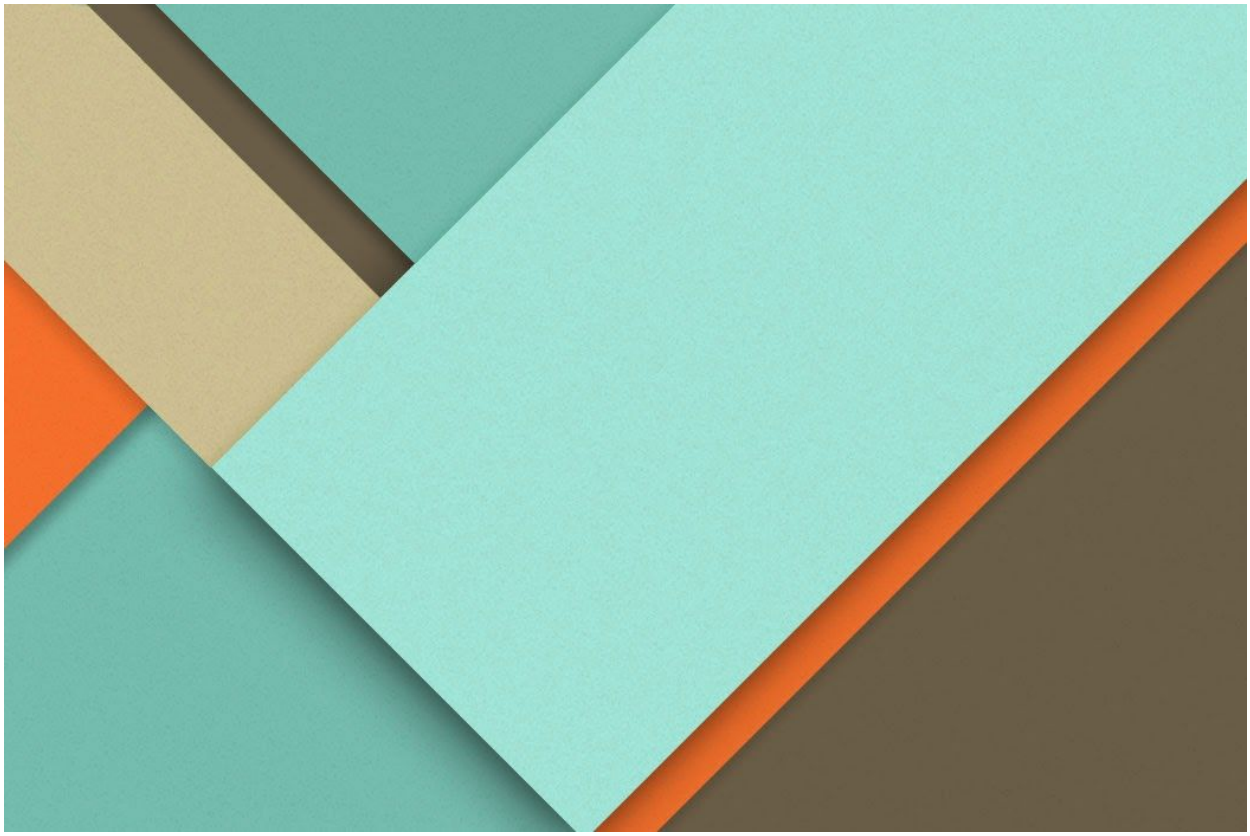


System Design

19/10/2020

Mohammad Amr Khan
Tony Xu
Sofia Rahul
Rui Wu
Winston Ge
Mohammad Sajjad
Sahil Hakimi

Team TBD
CSCC01

Table of Contents

CRC Cards

Front-End

Back-End

System Architecture

For this project as a group, we chose to use the MERN stack as the technology stack and the MVC model as the architectural model.

System Interaction with Environment

The system is a web application that is built using the MERN stack. MongoDB Atlas is the NoSQL database that is going to be used to store data. We are using the Atlas version which is a cloud-based deployment, offered on three major cloud providers from Mongo. Express.js is being used to help build APIs to the database. Firebase is used for user authentication.

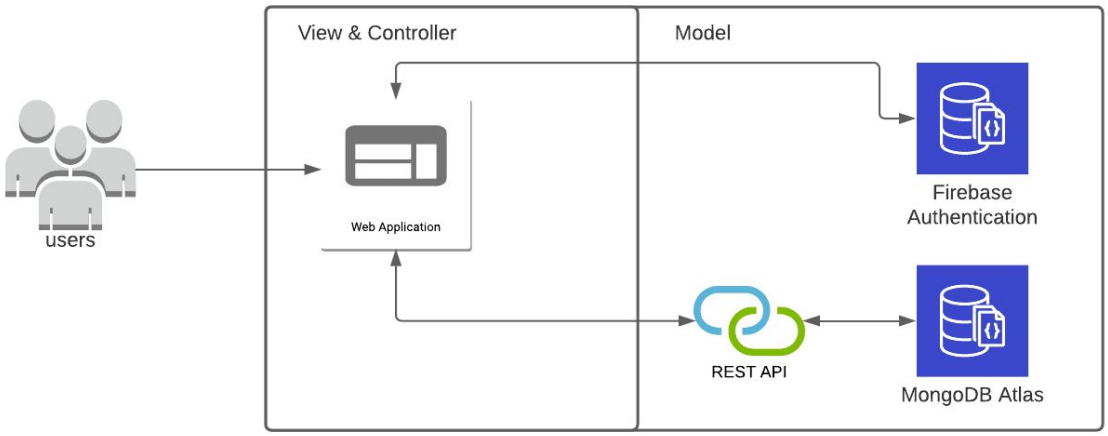
Accounts have already been set up for Firebase and MongoDB Atlas and are being used to send and get data from those sources.

All Node.js dependencies have been added to the package.json, so running **npm install** will install them.

How to Run Application

1. Ensure that you have React installed on your system before proceeding
2. Clone the project from the Github repository
3. Using CLI launch the project
    a. To launch the back-end:
        i. Navigate to <Repo_Home>/u-impactify/backend
        ii. Run the command - **npm install** (only do this if on the first time)
        iii. Run the command - **nodemon server**
    b. To launch the front-end:
        i. Navigate to <Repo_Home>/u-impactify/backend
        ii. Run the command - **npm install** (only do this if on the first time)
        iii. Run the command - **npm start**
        iv. The web app will be available using a browser at localhost:3000

System Architecture



The diagram above shows system interaction and flows with the application. An explanation of the components below.

Component	Description
User	The user is the person who will be interacting with the application
Web App	The web application is what the user will be interacting with and where they will see information based on what they select
Firebase Authentication	Authentication utility of firebase is leveraged, and the credentials are stored in Firebase
REST API	Used to send data from the front-end to MongoDB using API calls
MongoDB Atlas	A cloud version of MongoDB is being used. This will be used to store data.

System Decomposition

MongoDB

MongoDB will be used to store information about the users (social initiatives, learners, consultants) that will be displayed on their profiles. MongoDB will also be used to store

information about courses and jobs that are available. This will be part of the model in the MVC architecture.

Web Application

The web application that is being developed will serve as the viewer and controller in the MVC architecture. Uses will view everything using the website and interact with it and any changes will be reflected on the web pages. Express.js and Axios will be used to get or put information from MongoDB (model). In the case of an error what went wrong will be displayed on the page.

Firebase Authentication

Firebase Authentication will be used to authenticate users for the website. Firebase will allow users to log in or signup with their email, or use Facebook or LinkedIn to authenticate with. If the users are unable to authenticate they will not be able to access all the features that require an account. They will only be able to navigate to what is available on the landing page. This is part of the model in the MVC architecture.