Sri Lanka Institute of Information Technology

Student and Instructor Information System

Technical Report

APPLICATION FRAMEWORKS - SE3040

Submitted by:

IT17016230 - Saranga S.A.G.
IT17006880 - De Silva N.
IT17012966 - L.S. Jayasinghe
IT17410250 - M.V Lakshitha

Weekend Batch

Group No: **WE 25**

Submitted to:

(Supervisor's signature)
……………………………..

Name of the supervisor

23/06/2019
Date of submission

Contents

ABSTRACT

The following project report contains all the necessary technical details of the **Student Instructor Information System** developed the members of this project.

The full source code of this project along with the URL in which this application is hosted is given below.

GitHub link- https://github.com/gihanrcg/af_final_project.git

Application link- http://sms-af.herokuapp.com/

Introduction

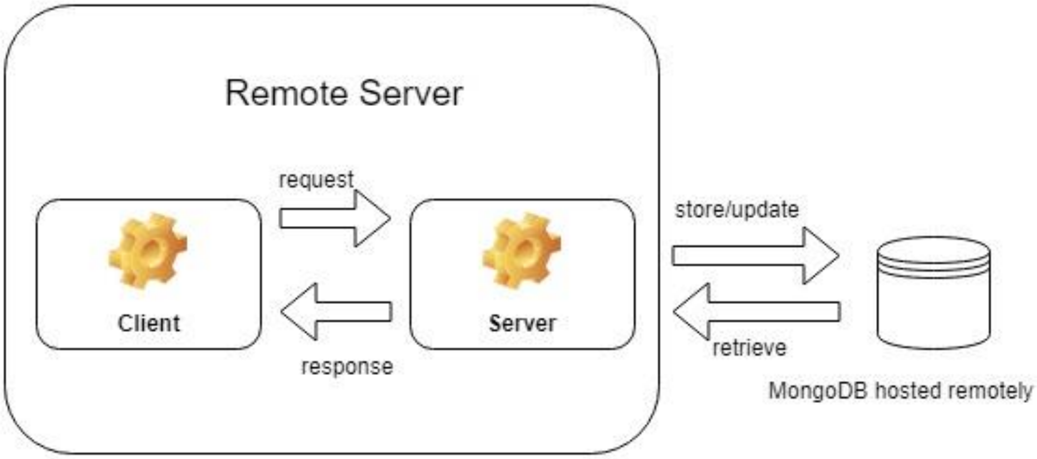
This project is a Student Instructor information System developed making the using of REST API technologies with the help of ExpressJS to develop the backend services and ReactJS for the front-end components. The developed project provides the following features and functionalities to the users which mainly consist of Student, Instructor and Administrator user roles.

The following are the key features and functionalities implemented in this project,

- Allow administrators to create administrator, student and instructor roles
- Notify user creation/registration via email
- Allow administrators to create a course and add instructors to a course
- Notify instructors about new course creations by administrators
- Allow instructors to accept/reject new course creations by administrators
- Allow students to join courses
- Allow instructors to create assignments and exams for a course with due dates as well as notify students about updates
- Allow students to upload files for assignments and exams before the due date.
- Allow instructor to view and download the uploaded files by the students
- Publish student marks and notify the students
- Allow instructors to create question papers and exams for students as well as publish the results of those exams

The frontend and backend and database of this application are all hosted on remote servers.

A Simple high-level diagram of the developed application is as follows



Tools and Technologies

- Backend- NodeJS / ExpressJS
- Frontend- ReactJS, JavaScript (jQuery), HTML, CSS (Bootstrap and React-Bootstrap)
- Database-MongoDB

Backend and Frontend Implementation

Backend

Components

Backend components were mainly divided into two main components:

- **Models** – These are the Schema object which represent the structure of the relevant collections stored in the database
- **Routes** – These are the API endpoints making use of REST principles to retrieve, handle and modify data store in the database with the help of the model objects.

Server- The server.js file contains all the configurations for setting up the port including the configurations for production hosting on the server.

Middleware used for the application are also defined in the server.js file and the key middleware used in the application are as listed below.

- Body-parser
- Base URL routes for the developed endpoints
- Authentication route
- express.static to serve static files

The key dependencies used in developing the backend application are as follows:

- Router
- bcryptjs
- body-parser
- bootstrap
- concurrently
- config

- dotenv
- express
- jsonwebtoken
- mongoose
- multer
- nodemailer
- nodemon
- passport
- passport-jwt
- react
- react-dom
- react-router-dom
- reactstrap

Frontend

The frontend of the application is mainly comprised of react components used in rendering the UI. Styling of UI components is done with the help of Bootstrap and React-Bootstrap along with jQuery to handle user interactions.

The key dependencies used in developing the frontend application are as follows:

- material-ui/core
- axios
- material-ui
- react
- react-avatar
- react-dom
- react-filterable-table
- react-imageloader
- react-notifications
- react-scripts
- react-scroll-to-component
- react-select
- react-slideshow-image
- react-sort-search-table
- sweetalert
- material-ui/icons

- bootstrap-4-react
- react-bootstrap
- react-bootstrap-grid

Database

The database is a MongoDB database hosted remotely and is used to store all the data collections required for the application.

Authentication and Security

Authentication and security have been a key concern when developing the application and necessary steps have been taken in order to ensure user authentication, access and data security.

The list of implemented security features are given below.

- Usage of JSON web token to provide user authentication
- Hashing of user passwords in the database
- Confirmation of user email when a new account is created using a uniquely generated token sent through email
- User session management