PROJECT 4 – The School -

Front end:

├── school-clientside

| ├── css

| | ├──style

| ├── imgs

| ├── js

| | ├──controlers

| | | ├── nav-bar-controller.js

| | | ├── page1-login-controler.js

| | | ├── page2-mail-insert-controller.js

| | | ├── page3-confirmation-controler.js

| | | ├── page7-School-controler.js

| | | ├── page8-admnistration-controler.js

| | ├──app.js

| | ├──factories.js

| ├── templates

| | ├──Courses

| | | ├── course-create-template.js

| | | ├── course-delete-template.js

| | | ├── course-edit-template.js

| | | ├── course-template.js

| | ├──Manager

| | | ├── manager-create-template.js

| | | ├── manager-delete-template.js

| | | ├── manager-edit-template.js

| | | ├── manager-template.js

| | ├──Student

| | | ├── student-create-template.js

| | | ├── student-delete-template.js

| | | ├── student-edit-template.js

| | | ├── student-template.js

| | └── login-template.html

| ├── views

| | ├── page1-login.html

| | ├── page2-mail-insert.html

| | ├── page3-confirmation.html

| | ├── page7-School.html

| | ├── page8-administration.html

| └── index.js

Back end:

!!! –node\_modules is empty, requires npm install to deploy

!!! –public and views folders are not in use, separated front end so there Is no rendering of pages

├── school-serverside

| ├── bin

| | ├──www

| ├── config

| | ├── JWT-config.js

| | ├── passport.js

| ├── controllers

| | ├── admins\_controller.js

| | ├── courses\_controler.js

| | ├── students\_controller.js

| ├── imgs

| ├── models

| | ├── admins\_model.js

| | ├── courses\_model.js

| | ├── students\_model.js

| ├── node\_modules

| ├── public

| ├── routes

| | ├── admins.js

| | ├── courses.js

| | ├── students.js

| | ├── index.js

| | ├── users.js

| ├── views

├── app.js

├── Create-DB.js

├── package.json

├── package-lock.json

Front end implemented using angularJS backend implemented using node.JS

In frontend side the file named app.js contains both route configuration and service named “sharedProperties” that keeps several variables transferable between different controllers.

On the first run to create the DB: node Create-DB

Back end needs to be started by either:

1. Npm start
2. Nodemon bin/www

Structured according to MVC principals as much as possible

Routes that concern students and courses go to students and courses model respectively

While admin model is used by both administrators display and editing as well as to log into the system. So routes concerning login go through route named users, routes concerning CRUD operations go through admins route.

Inside config folder, all elements regarding passport model are in passport.js file, all the definitions regarding creation of JWT and checking for authorization level as coded in the JWT is in JWT-config.js