Index.ejs

<!DOCTYPE html>

<html>

<head>

<meta charset='utf-8'>

<meta http-equiv='X-UA-Compatible' content='IE=edge'>

<title>Intern's Onboard</title>

<meta name='viewport' content='width=device-width, initial-scale=1'>

<link rel='stylesheet' type='text/css' media='screen' href='main.css'>

<script src='main.js'></script>

<style>

a {

text-decoration: none;

color: #000;

padding: 10px;

border: 1px solid #000;

border-radius: 8px;

margin: 10px;

}

</style>

</head>

<body>

<h1>Intern's Onboard</h1>

<nav id="common-nav" class="common-nav" style="display: flex; flex-wrap: wrap;">

<!-- Will add approaching colleges and setting orientation as a field in the provide counts -->

<a href="/provide-counts">Provide Interns Count</a>

<a href="/show-stats">Show Universities Stats</a>

<!-- Will add field to Universities and counts to send resume forms Once clicked on send resume form the university will recieve an email with the link to upload the resume forms of all the candidates -->

<a href="/show-resumes">Show Resumes</a>

<!-- Show resumes will contain the list of universities and followed by the number of interns who have uploaded their resume forms -->

<!-- Each resume uploaded will count as one intern of the mentioned university. -->

<!-- Will have a selected or rejected button on the show resume page. -->

<a href="/resume">The Resume Form</a>

</nav>

</body>

</html>

server.js

const express = require('express')

const mongoose = require('mongoose')

const Intern = require('./models/data')

const Resume = require('./models/intern\_form')

const app = express()

const router = express.Router()

// app.use(express.json())

app.use(express.urlencoded({extended:true}))

const mongodb = 'mongodb+srv://bishant:mongo\_bishant1234@hrc-cluster.ikeg2.mongodb.net/myFirstDatabase?retryWrites=true&w=majority';

mongoose.connect(mongodb, {useNewUrlParser: true, useUnifiedTopology: true});

app.set('view engine', 'ejs')

//Render the Main landing page

app.get("/", (req, res) => {

res.render('index')

})

// Render the page to initiate adding the data of interns

app.get("/provide-counts", (req, res) => {

res.render('provide\_counts', {intern: new Intern()})

})

// Show the Statistics of the data entered by the users.

app.get("/show-stats", async (req, res) => {

const intern = await Intern.find()

console.log(intern);

res.render('show\_stats', {intern: intern})

})

// Each intern data

app.get('/show\_stats/:id', async (req, res) => {

// const intern = await Intern.findById(req.params.id)

const intern = await Intern.findById(req.params.id)

.catch(err => {console.log("Error: ", err);})

if (intern == null) res.redirect('/')

res.render('show\_stats\_id', {intern: intern})

// res.send(req.params.id)

})

// Each intern data

app.post('/show-stats/delete/', async (req, res) => {

// const intern = await Intern.findById(req.params.id)

console.log("id: ", req.body.id);

const intern = await Intern.findById(req.body.id)

.catch(err => {console.log("Error: ", err);})

const \_ = await intern.remove()

.catch(err => {console.log("Error: ", err);})

const new\_interns = await Intern.find()

res.render('show\_stats', {intern: new\_interns})

})

app.get("/show-resumes", (req, res) => {

res.render('show\_resumes')

})

app.get("/resume", (req, res) => {

res.render('intern\_resume')

})

app.post("/resume/submit/", async (req, res) => {

let data = {

name: req.body.name,

email: req.body.email,

phone: req.body.phone,

college: req.body.college, // This will be a dropdown with populated data of colleges names.

degree: req.body.degree,

year: req.body.year,

resume: req.body.resume,

gpa: req.body.gpa,

criminal\_record: true ? req.body.criminal\_record == 1 : false

}

console.log(data);

let resume = new Resume(data)

const resume\_id = await resume.save()

.catch(err => {console.log("Error in Resume Submitting: ", err);})

res.render('thankyou')

})

// Initial Phase come to the site and provide a college wide count of the interns

app.post("/provide-counts/add/", async (req, res) => {

// console.log("Req: ", req.body);

var data = {

university: req.body.university,

count: req.body.count,

}

let intern = new Intern(data)

try {

intern = await intern.save()

res.redirect(`/show\_stats/${intern.id}`)

} catch (error) {

console.log(error)

res.render('provide\_counts', {intern: intern})

}

// console.log("Data: ", data);

res.render("index")

})

app.listen(5003)

console.log("Running Port at 5003");