Server.js

const express = require('express');

const cors = require('cors');

const bodyParser = require('body-parser');

const config = require('./Config');

var mongoose = require('mongoose');

const app = express();

const router = express.Router();

app.use(cors());

mongoose.connect(config.collection, { useNewUrlParser: true, useUnifiedTopology: true, useFindAndModify: false })

.then(() => console.log('connection successful'))

.catch((err) => console.error(err));

router.use(bodyParser.urlencoded({ extended: false }));

router.use(bodyParser.json());

// Message from pinging server

router.get('/', (req, res) => {

res.status(200).send('This is an authentication server');

});

// Creates listener for http requests

app.use(router);

const port = process.env.PORT || 3000;

app.listen(port, () => {

console.log('Server listening at http://localhost:' + port);

});

var UserSchema = new mongoose.Schema({

user: String,

venueNames: [String],

venueIDs: [String]

});

const Users = mongoose.model('Users', UserSchema);

// --------------------------------------------------------

// Router functions

// --------------------------------------------------------

router.post('/addFavorite/:id', (req, res, next) => {

Users.findByIdAndUpdate(req.params.id,

{

$push: {venueNames: req.body.name, venueIDs: req.body.id}

},

(err, post) => {

if (err) return next(err);

res.json(post);

});

});

router.get('/getFavorites/:id', (req, res, next) => {

Users.findById(req.params.id,

(err, post) => {

if (err) return next(err);

res.json(post);});

});

router.get('/loginUser/:user', (req, res, next) => {

Users.findOneAndUpdate({ user: req.params.user },

{},

{new: true, upsert:true},

(err, post) => {

if (err) return next(err);

res.json(post);});

});

App.module.ts

import { BrowserModule } from '@angular/platform-browser';

import { NgModule } from '@angular/core';

import { HttpClientModule } from '@angular/common/http';

import { FormsModule } from '@angular/forms';

import { AppComponent } from './app.component';

import { HomeComponent } from './home/home.component';

import { LoginComponent } from './login/login.component';

import { InfoComponent } from './info/info.component';

import { FavoritesComponent } from './favorites/favorites.component';

import { AppRoutingModule } from './app-routing.module';

@NgModule({

declarations: [

AppComponent,

LoginComponent,

InfoComponent,

FavoritesComponent,

HomeComponent

],

imports: [

BrowserModule,

FormsModule,

HttpClientModule,

AppRoutingModule

],

providers: [],

bootstrap: [AppComponent]

})

export class AppModule { }

app-routing.module.ts

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import { HomeComponent } from './home/home.component';

import { LoginComponent } from './login/login.component';

import { InfoComponent } from './info/info.component';

import { FavoritesComponent } from './favorites/favorites.component';

const routes: Routes = [

{ path: '', redirectTo: '/login', pathMatch: 'full' },

{ path: 'home', component: HomeComponent },

{ path: 'login', component: LoginComponent },

{ path: 'info/:id', component: InfoComponent },

{ path: 'favorites', component: FavoritesComponent }

];

@NgModule({

imports: [RouterModule.forRoot(routes)],

exports: [RouterModule]

})

export class AppRoutingModule { }

app.component.html

<nav class="navbar navbar-expand-md bg-dark navbar-dark">

<a class="navbar-brand" href="#">Search for Places Nearby</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse" aria-controls="navbarCollapse"

aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarCollapse">

<ul class="navbar-nav mr-auto">

<li class="nav-item">

<a class="nav-link" routerLink="/home">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" routerLink="/favorites">Favorites</a>

</li>

</ul>

</div>

</nav>

<div class="container">

<router-outlet></router-outlet>

</div>

Data.service.ts

import { Injectable } from '@angular/core';

import { HttpClient, HttpHeaders } from '@angular/common/http';

import { environment } from '../../environments/environment';

import { Observable } from 'rxjs';

import { Router } from '@angular/router';

@Injectable({

providedIn: 'root'

})

export class DataService {

private readonly header: any;

user: any;

constructor(private http: HttpClient, private router: Router) {

this.header = new HttpHeaders({

'Content-Type': 'application/json',

Authorization: 'Bearer ' + environment.yelp\_key });

}

loginUser(user: string) {

if (this.user !== null && this.user !== '') {

this.http.get(`${environment.app\_server\_address}/loginUser/${user}`)

.subscribe((data: any) => {

this.user = data.\_id;

this.router.navigateByUrl('/home');

});

}

}

getFavorites(id: string): Observable<any> {

return this.http.get<any>(`${environment.app\_server\_address}/getFavorites/${id}`);

}

addFavorite(favorite: any) {

return this.http.post(`${environment.app\_server\_address}/addFavorite/${this.user}`, favorite);

}

getVenues(location, venue) {

return this.http.get('https://api.foursquare.com/v2/venues/search?query=' + venue +

'&near=' + location +

'&limit=5' +

'&client\_id=' + environment.foursquare\_id +

'&client\_secret=' + environment.foursquare\_secret +

'&v=20200626');

}

getVenueID(venue) {

return this.http.get('https://cors-anywhere.herokuapp.com/https://api.yelp.com/v3/businesses/matches' +

'?name=' + venue.name +

'&address1=' + venue.location.address +

'&city=' + venue.location.city +

'&state=' + venue.location.state +

'&country=' + venue.location.cc +

'&limit=5', { headers: this.header }).subscribe((business: any) => {

if (business.businesses.length === 0) {

console.log('No information on this location');

} else {

this.router.navigateByUrl(`/info/${business.businesses[0].id}`);

}

});

}

getVenueDetails(id) {

return this.http.get('https://cors-anywhere.herokuapp.com/https://api.yelp.com/v3/businesses/' +

id, { headers: this.header });

}

}

Login.component.ts

import { Component, OnInit } from '@angular/core';

import { DataService } from '../services/data.service';

@Component({

selector: 'app-login',

templateUrl: './login.component.html',

styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

user: string;

constructor(private dataService: DataService) { }

ngOnInit(): void {

document.body.classList.add('bg-img');

}

loginUser() {

this.dataService.loginUser(this.user);

}

}

Login.component.html

<form class="border border-light p-5">

<p class="h4 mb-4">Please Enter your Username to Login</p>

<label class="label" for="userInput" style="font-size: xx-large"> Username </label>

<input type="text" class="form-control mb-4" placeholder="Enter Username" [(ngModel)]="user" name="user" id="userInput">

<button (click)="loginUser()" type="button" class="btn btn-primary">Login</button>

</form>

Home.component.html

<form class="border border-light p-5">

<p class="h4 mb-4">Enter the place you want to find in an area</p>

<label class="label" for="venueInput" style="font-size: xx-large"> Venue </label>

<input type="text" class="form-control mb-4" placeholder="Venue" [(ngModel)]="venue" name="Venue" id="venueInput">

<label class="label" style="font-size: xx-large" for="locationInput"> Location </label>

<input type="text" class="form-control mb-4" placeholder="Location" [(ngModel)]="location" name="Location" id="locationInput">

<button (click)="searchVenues()" type="button" class="btn btn-primary" style="margin-right: 5px">Search</button>

<a routerLink="/favorites" ><button type="button" class="btn btn-primary">Favorites</button ></a>

</form>

<div \*ngIf="venueList.length !== 0" class="col-lg-12">

<div class="card bg-transparent">

<div class="card-heading"><b>Nearby Places</b></div>

<div class="card-body bg-transparent">

<div \*ngFor="let venue of venueList; index as i" class="list-group-item clearfix bg-transparent" (click)="getDetails(i)">

<h4 class="list-group-item-heading">{{ venue.name }}, {{ venue.location.city }}, {{ venue.location.country }}</h4>

<span class="pull-right">

<a href="http://maps.google.com/maps?saddr={{currentLat}},{{currentLong}}

&daddr={{venue.location.address1}},{{ venue.location.city}}" target="\_blank">

<img src="../../assets/google.png" style="width: 30px; height: 30px;"></a>

</span>

</div>

</div>

</div>

</div>

Home.component.ts

import { Component, OnInit } from '@angular/core';

import { DataService } from '../services/data.service';

import {HttpHeaders} from '@angular/common/http';

import {environment} from '../../environments/environment';

@Component({

selector: 'app-home',

templateUrl: './home.component.html',

styleUrls: ['./home.component.css']

})

export class HomeComponent implements OnInit {

venue: string;

location: string;

venueList = [];

currentLat: any;

currentLong: any;

geolocationPosition: any;

constructor(private dataService: DataService) {

}

ngOnInit() {

document.body.classList.add('bg-img');

window.navigator.geolocation.getCurrentPosition(

position => {

this.geolocationPosition = position;

this.currentLat = position.coords.latitude;

this.currentLong = position.coords.longitude;

});

}

// Find a list of venues that matches the location and venue description

searchVenues() {

if (this.location != null && this.location !== '' && this.venue != null && this.venue !== '') {

this.dataService.getVenues(this.location, this.venue).subscribe((data: any) => {

this.venueList = data.response.venues;

});

}

}

// Pass the index of the selected item from the results list

getDetails(index) {

this.dataService.getVenueID(this.venueList[index]);

}

}

Info.component.html

<form class="border border-light p-5">

<button routerLink="/home" type="button" class="btn btn-primary">Back</button>

<p class="h4 mb-4" style="margin-top: 10px">Place Information</p>

<div \*ngIf="info !== undefined" class="col-lg-12" style="margin-top: 10px">

<div class="card bg-transparent">

<div class="card-body bg-transparent">

<h4 class="list-group-item-heading">{{info.name}}</h4>

<h5 class="list-group-item-heading">{{info.display\_phone}}</h5>

<h5 \*ngFor="let line of info.location.display\_address" class="list-group-item clearfix bg-transparent">{{line}}</h5>

</div>

</div>

</div>

<button (click)="addFavorite()" type="button" class="btn btn-primary" style="margin-top: 20px">Favorite</button>

<a routerLink="/favorites" ><button type="button" class="btn btn-primary" style="margin-left: 10px; margin-top: 20px">Favorites Page</button ></a>

</form>

Info.component.ts

import { Component, OnInit } from '@angular/core';

import { DataService } from '../services/data.service';

import { ActivatedRoute } from '@angular/router';

@Component({

selector: 'app-info',

templateUrl: './info.component.html',

styleUrls: ['./info.component.css']

})

export class InfoComponent implements OnInit {

info: any;

constructor(private dataService: DataService, private route: ActivatedRoute) { }

ngOnInit(): void {

document.body.classList.add('bg-img');

this.dataService.getVenueDetails(this.route.snapshot.params.id)

.subscribe((details: any) => {

this.info = details;

});

}

addFavorite() {

this.dataService.addFavorite({

id: this.route.snapshot.params.id,

name: this.info.name

}).subscribe();

}

}

Favorites.component.html

<form class="border border-light p-5">

<button routerLink="/home" type="button" class="btn btn-primary">Back</button>

<p class="h4 mb-4" style="margin-top: 10px">Place Information</p>

<div \*ngIf="favorites !== undefined" class="col-lg-12" style="margin-top: 10px">

<div class="card bg-transparent">

<div class="card-body">

<div \*ngFor="let venue of favorites.venueNames; index as i" class="list-group-item clearfix bg-transparent"

routerLink="/info/{{favorites.venueIDs[i]}}">

<h4 class="list-group-item-heading">{{ venue }}</h4>

</div>

</div>

</div>

</div>

</form>

Favorites.component.ts

import { Component, OnInit } from '@angular/core';

import { DataService } from '../services/data.service';

@Component({

selector: 'app-favorites',

templateUrl: './favorites.component.html',

styleUrls: ['./favorites.component.css']

})

export class FavoritesComponent implements OnInit {

favorites: any;

constructor(private dataService: DataService) { }

ngOnInit(): void {

document.body.classList.add('bg-img');

this.dataService.getFavorites(this.dataService.user).subscribe((data: any) => {

this.favorites = data;

});

}

}