/// <reference types="@types/googlemaps" />

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

import { ListingCardComponent } from "../listing-card/listing-card.component";

import { ISearchResult } from '../../models/search-result.model';

// import { IListing } from "../../models/listing.model";

import { ApiService } from '../../shared/api/api.service';

import { SearchCriteriaService } from '../../services/search-criteria.service';

// import { PageChangedEvent } from 'ngx-bootstrap';

@Component({

selector: 'app-srp',

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

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

})

export class SrpComponent implements OnInit {

@ViewChild('googlemap') googlemapElement: any;

map: google.maps.Map;

infoWindow = new google.maps.InfoWindow();

public searchResult: ISearchResult;

// returnedArray: IListing[];

private \_view: string = "grid";

latitude: number = 41.85003;

longitude: number = -87.6500523;

zoom: number = 5;

previous;

// contentArray = new Array(90).fill('');

// lPageSizes: any[] = [

// { id: 1, Value: '4' },

// { id: 2, Value: '8' },

// { id: 3, Value: '12' },

// { id: 4, Value: '16' }

// ];

// curPageSize: any = this.lPageSizes[0];

constructor(private \_api: ApiService, private \_searchCriteriaService: SearchCriteriaService,

) {

}

async ngOnInit() {

//TODO - populate search criteria, post listing search

const searchCriteria = this.\_searchCriteriaService.getCriteria();

this.searchResult = await this.\_api.getEndPoint<ISearchResult>('/assets/data/search-result.json');

console.log("search result", this.searchResult);

// this.returnedArray = this.searchResult.listings.slice(0, Number(this.curPageSize.Value));

var mapProp = {

center: new google.maps.LatLng(41.850033, -87.6500523),

zoom: 5

};

// this.map = new google.maps.Map(this.googlemapElement.nativeElement, mapProp);

for (let listing of this.searchResult.listings) {

if (listing.latitude != 0 && listing.longitude != 0) {

let location = new google.maps.LatLng(listing.latitude, listing.longitude);

let marker = new google.maps.Marker({

position: location,

map: this.map,

title: listing.address

});

marker.addListener('click', (e) => {

console.log("list", listing);

// this.markerHandler(marker,e)

});

}

}

}

// pageChanged(event: PageChangedEvent): void {

// const startItem = (event.page - 1) \* event.itemsPerPage;

// const endItem = event.page \* event.itemsPerPage;

// this.returnedArray = this.searchResult.listings.slice(startItem, endItem);

// }

// setPageSize(id: any): void {

// this.curPageSize = this.lPageSizes.filter(value => value.id === Number(id))[0];

// this.returnedArray = this.searchResult.listings.slice(0, Number(this.curPageSize.Value));

// }

OnMarkerClick(cardInfoWindow) {

console.log("info", cardInfoWindow);

if (this.previous) {

this.previous.close();

}

this.previous = cardInfoWindow;

}

// markerHandler(marker: google.maps.Marker,e) {

// }

isGridView(): boolean {

return this.\_view === "grid";

}

isMapView(): boolean {

return this.\_view === "map";

}

loadView(view: string): void {

this.\_view = view;

}

}