Sarath Chandra kunisetty – skkh2@umkc.edu

GitHub link: https://github.com/kunisettysarath/WebMobileProgramming-Spring22/tree/main/WebDevelopment/ICP's/ICP6

Venubabu Linga- vl6hw@umsystem.edu

GitHub link: https://github.com/VenubabuLinga/WebDevCourse/tree/main/WEB_DEV/ICP6

ICP 2

Objective:

The objective of this ICP6 is to learn about routers, services, HTTP, and RESTful-APIs in Angular

Introduction:

Routers are the essential part of single page applications(SPA) concept of angular. Using routers user will be able to navigate to different pages of a web-application without any need of reloading the webpage. Main advantage of SPA is that user doesn't have to wait for the webpage to load while navigating between pages, data is displayed dynamically.

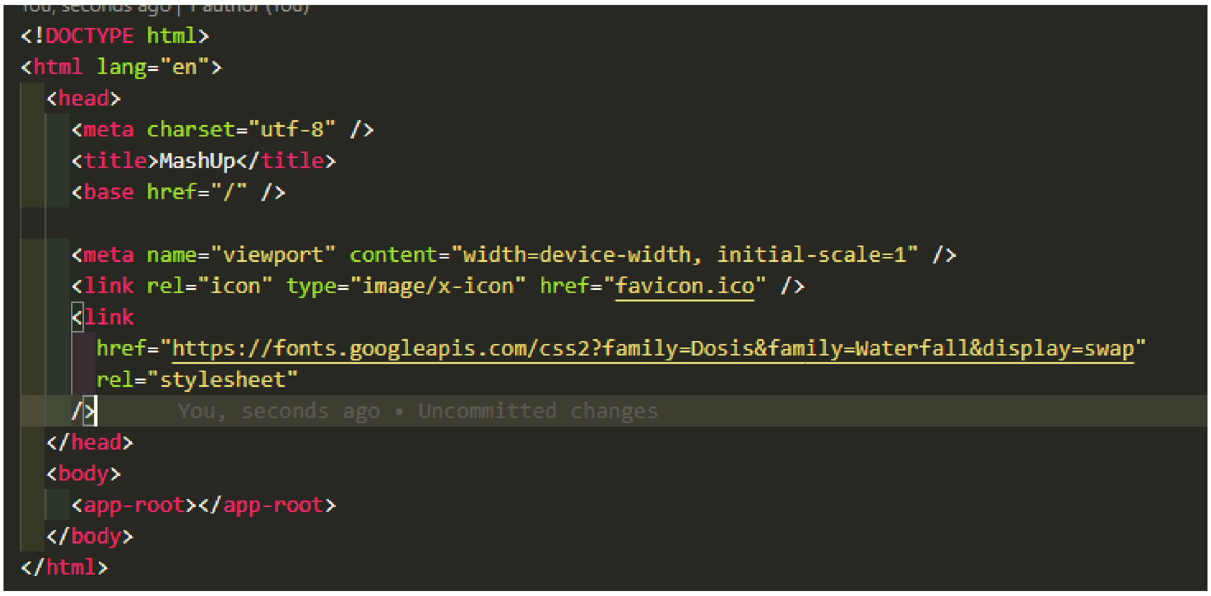
Services in angular are used to reduce the usage of redundant code. If a part of code logic needed to be used between different components then rather than writing that logic in all the components a service can be created and could be reused between all the components.

Restful-api's are used to get, update data from different servers across the globe. These are used to provide interoperability between various systems. Because of the less restriction, features and its flexibility Rest is preferred over Soap

Task:

1. Created "ICP6-submission" branch in the remote repository and have checkout to that branch using GitHub desktop.
2. Then required folders for source and documentation was created for icp6 and the source folder was imported in VS code
3. Created index.html file and have added all the required elements specified as part of the task.
4. Accessed api keys from foursquare and edamam
5. Using HTTPclient we were able to access the required details from the api's and have displayed the details are per the users request
6. These changes were automatically tracked in the GitHub desktop and then the code was committed and pushed into the remote repository using GitHub desktop.
7. As the changes were in another branch apart from main a pullrequest was created for "ICP6-submission" branch and the branch is merged into the main branch.

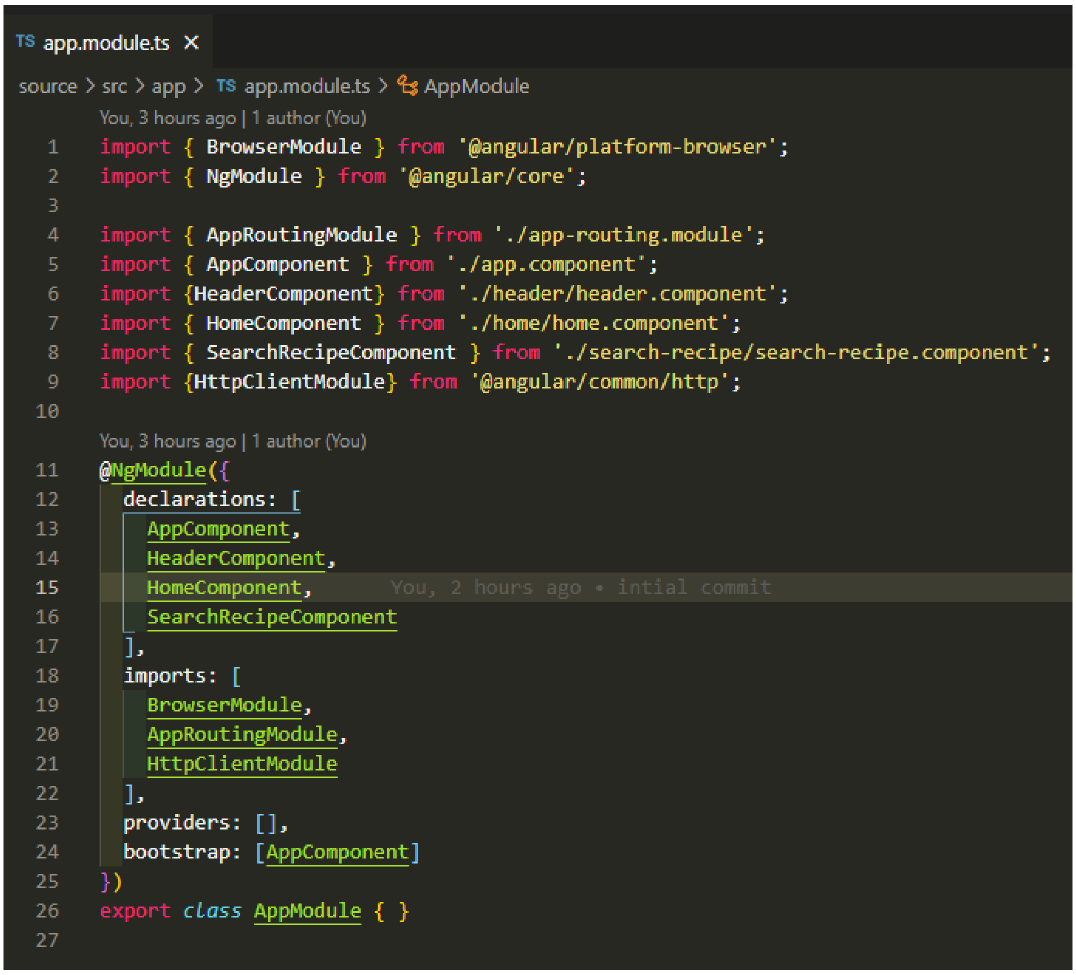
Index.html



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>MashUp</title>
    <base href="/" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="icon" type="image/x-icon" href="favicon.ico" />
    <link
      href="https://fonts.googleapis.com/css2?family=Dosis&family=Waterfall&display=swap"
      rel="stylesheet"
    />       You, seconds ago • Uncommitted changes
  </head>
  <body>
    <app-root></app-root>
  </body>
</html>
```

Here the actual execution starts where “<app-root></app-root>” represents the app component which is the main component containing all other components.

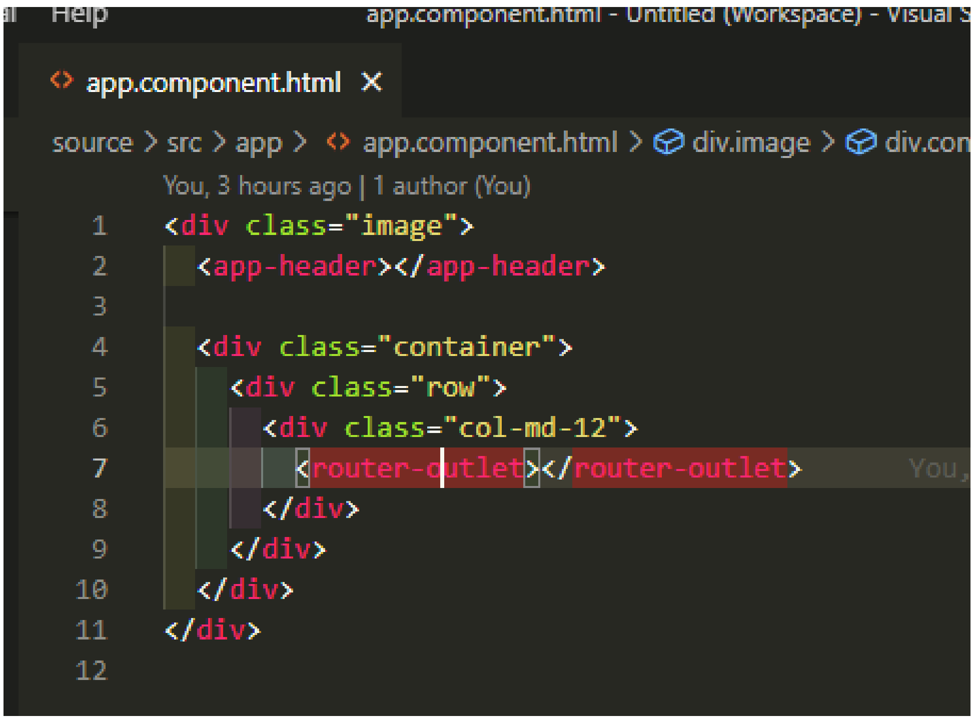
App.module.ts:



```typescript
You, 3 hours ago | 1 author (You)
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import {HeaderComponent} from './header/header.component';
import { HomeComponent } from './home/home.component';
import { SearchRecipeComponent } from './search-recipe/search-recipe.component';
import {HttpClientModule} from '@angular/common/http';

You, 3 hours ago | 1 author (You)
@NgModule({
  declarations: [
    AppComponent,
    HeaderComponent,
    HomeComponent,       You, 2 hours ago • intial commit
    SearchRecipeComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    HttpClientModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

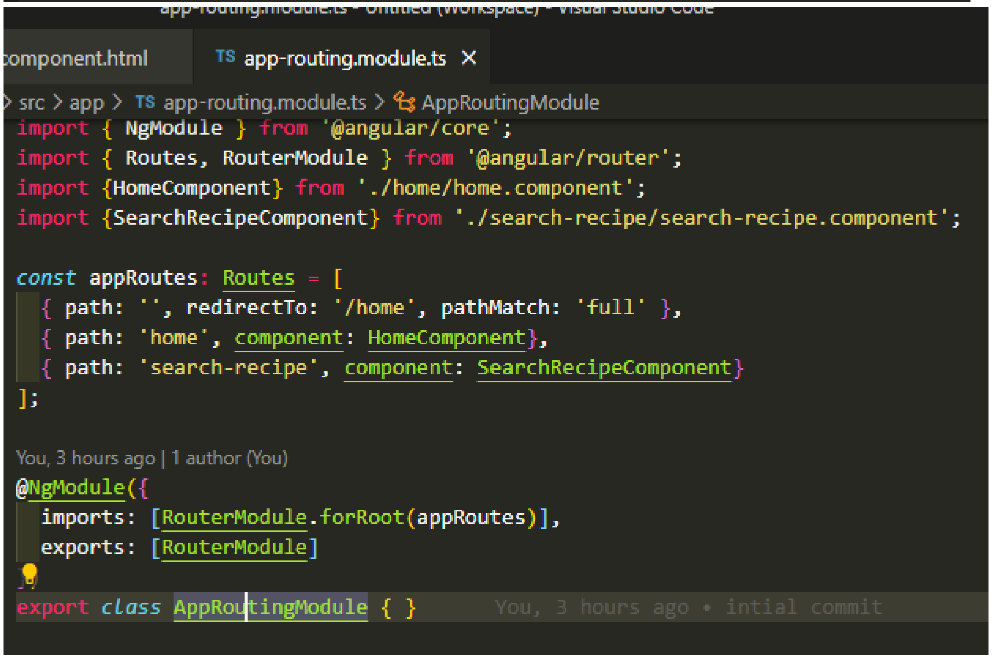
In here we import all the libraries that are required across all the components.

App.component.ts:



Help
app.component.html - Untitled (Workspace) - Visual S
app.component.html
source > src > app > app.component.html > div.image > div.con
You, 3 hours ago | 1 author (You)

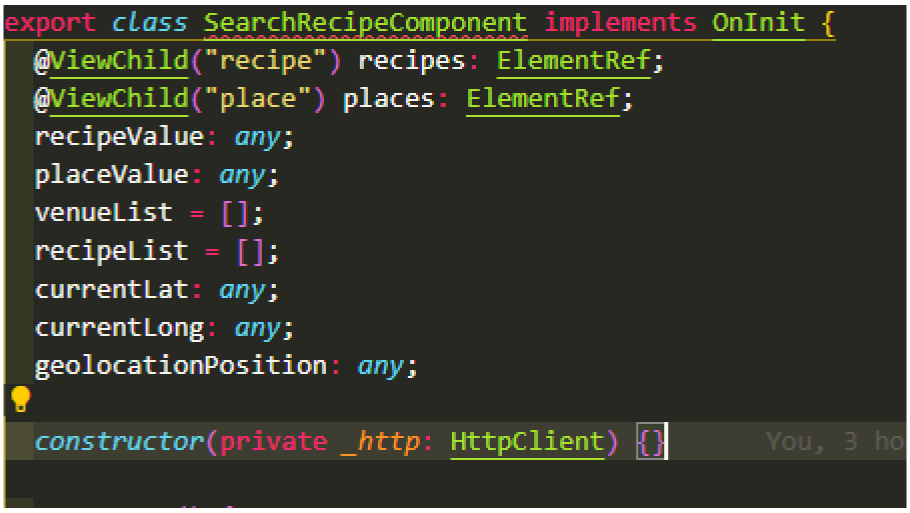
```html
<div class="image">
  <app-header></app-header>

  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <router-outlet></router-outlet>
      </div>
    </div>
  </div>
</div>
```



app-routing.module.ts - Untitled (Workspace) - Visual Studio Code
component.html
app-routing.module.ts
src > app > app-routing.module.ts > AppRoutingModule

```typescript
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';
import {HomeComponent} from './home/home.component';
import {SearchRecipeComponent} from './search-recipe/search-recipe.component';

const appRoutes: Routes = [
  { path: '', redirectTo: '/home', pathMatch: 'full' },
  { path: 'home', component: HomeComponent},
  { path: 'search-recipe', component: SearchRecipeComponent}
];

You, 3 hours ago | 1 author (You)
@NgModule({
  imports: [RouterModule.forRoot(appRoutes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }      You, 3 hours ago • intial commit
```

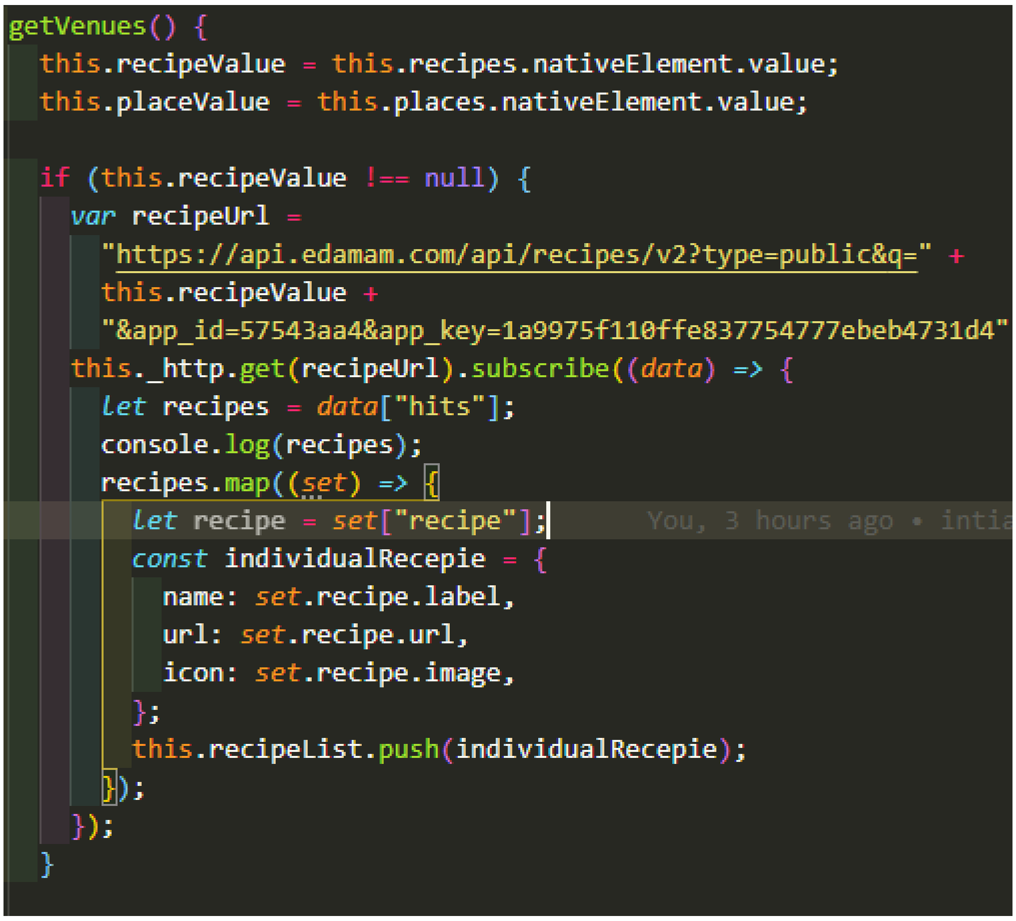
We have used "<router-outlet>" tag which is used for routing where the page will be redirected to respective page upon clicking by the user and in app-routing.module.ts we specify the components that needs to be displayed for a route



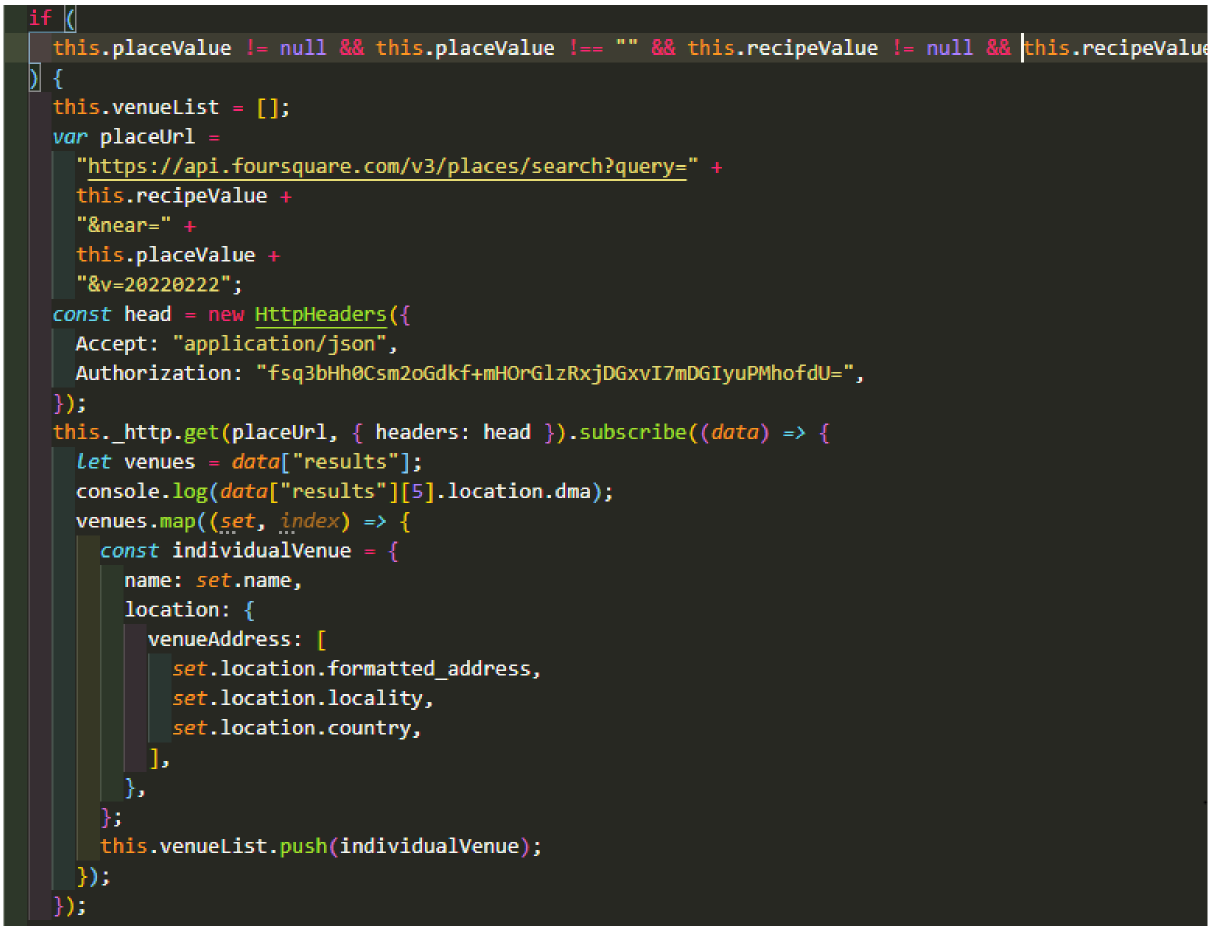
```
export class SearchRecipeComponent implements OnInit {
  @ViewChild("recipe") recipes: ElementRef;
  @ViewChild("place") places: ElementRef;
  recipeValue: any;
  placeValue: any;
  venueList = [];
  recipeList = [];
  currentLat: any;
  currentLong: any;
  geolocationPosition: any;

  constructor(private _http: HttpClient) {}        You, 3 ho
```

Here two values “recipe” and “place” both variables are used as an element reference inorder to get the users input and through constructor “_http” is assigned for HttpClient

getVenues() method is called upon clicking submit button in UI. The following part deals with creating a recipes request based on the users input and the recipes list obtained from the response is passed back to the “search-recipe.component.html” inorder to display in UI.



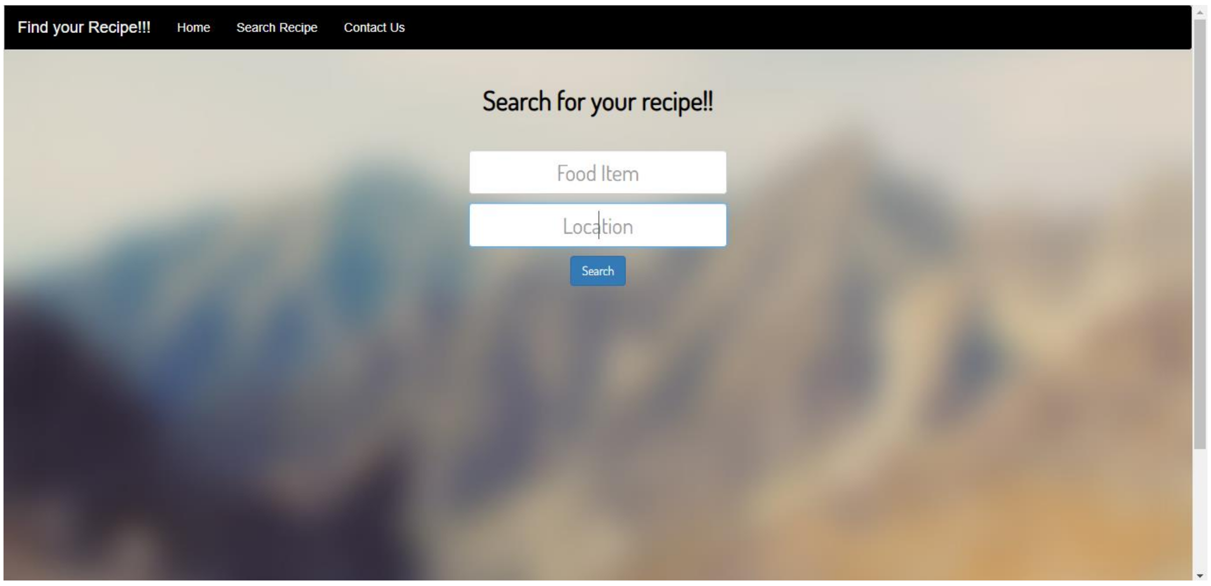
```
getVenues() {
  this.recipeValue = this.recipes.nativeElement.value;
  this.placeValue = this.places.nativeElement.value;

  if (this.recipeValue !== null) {
    var recipeUrl =
      "https://api.edamam.com/api/recipes/v2?type=public&q=" +
      this.recipeValue +
      "&app_id=57543aa4&app_key=1a9975f110ffe837754777ebeb4731d4"
    this._http.get(recipeUrl).subscribe((data) => {
      let recipes = data["hits"];
      console.log(recipes);
      recipes.map((set) => {
        let recipe = set["recipe"];        You, 3 hours ago • intia
        const individualRecepie = {
          name: set.recipe.label,
          url: set.recipe.url,
          icon: set.recipe.image,
        };
        this.recipeList.push(individualRecepie);
      });
    });
  }
```



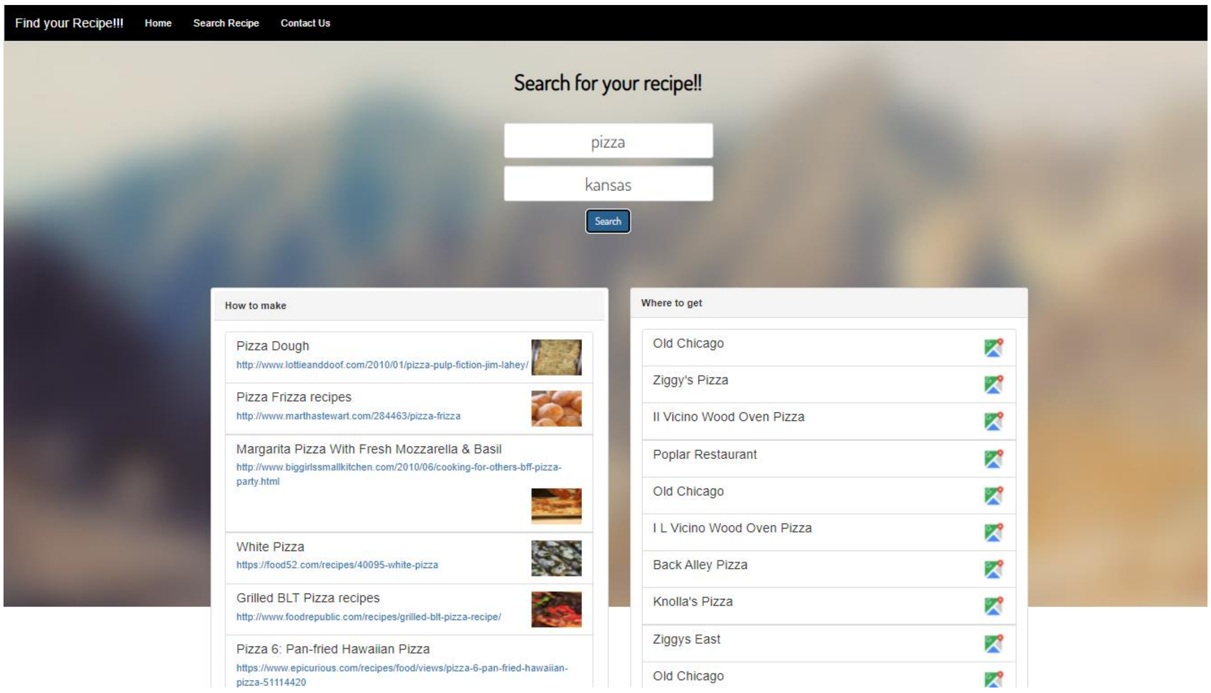
```
if (
  this.placeValue != null && this.placeValue !== "" && this.recipeValue != null && this.recipeValu
) {
  this.venueList = [];
  var placeUrl =
    "https://api.foursquare.com/v3/places/search?query=" +
    this.recipeValue +
    "&near=" +
    this.placeValue +
    "&v=20220222";
  const head = new HttpHeaders({
    Accept: "application/json",
    Authorization: "fsq3bHh0Csm2oGdkf+mHOrGlzRxjDGxvI7mDGIyuPMhofdU=",
  });
  this._http.get(placeUrl, { headers: head }).subscribe((data) => {
    let venues = data["results"];
    console.log(data["results"][5].location.dma);
    venues.map((set, index) => {
      const individualVenue = {
        name: set.name,
        location: {
          venueAddress: [
            set.location.formatted_address,
            set.location.locality,
            set.location.country,
          ],
        },
      };
      this.venueList.push(individualVenue);
    });
  });
```

The above code creates a http request to obtain the places where the user requested food item is available.

<u>Output:</u>



We have hidden the recipe and locations section until the user clicks on search button



How to make

Pizza Dough
http://www.lottieanddoof.com/2010/01/pizza-pulp-fiction-jim-lahey/

Pizza Frizza recipes
http://www.marthastewart.com/284463/pizza-frizza

Margarita Pizza With Fresh Mozzarella & Basil
http://www.biggirlssmallkitchen.com/2010/06/cooking-for-others-bff-pizza-party.html

White Pizza
https://food52.com/recipes/40095-white-pizza

Grilled BLT Pizza recipes
http://www.foodrepublic.com/recipes/grilled-blt-pizza-recipe/

Pizza 6: Pan-fried Hawaiian Pizza
https://www.epicurious.com/recipes/food/views/pizza-6-pan-fried-hawaiian-pizza-51114420

Where to get

Old Chicago
Ziggy's Pizza
Il Vicino Wood Oven Pizza
Poplar Restaurant
Old Chicago
I L Vicino Wood Oven Pizza
Back Alley Pizza
Knolla's Pizza
Ziggys East
Old Chicago

Once the user enters the details and clicks on search button respective recipe and venue details are displayed accordingly.