

# ICP6

Keenan Flynn – [kpfxn8@umsystem.edu](mailto:kpfxn8@umsystem.edu) – <https://github.com/kfly2fly/Web-Mobile-Spring-2022>  
Jasmine Naraine- [jnytc@umstyem.edu](mailto:jnytc@umstyem.edu) -<https://github.com/JNaraine/Web>

This ICP focuses on the REST APIs. RESTful APIs grant applications access to resources across the internet. These APIs allow us to use algorithms from 3rd parties such as FourSquare and Edamam in our own projects. REST uses HTTP protocols. These protocols include GET, PUT, POST, and DELETE. In this ICP, we used the `get()` method provided with Angular. This method retrieves the current state of the FourSquare and Edamam database and returns that information in the form of a JSON object. We can then query this object to get useful information.

To query the FourSquare and Edamam APIs, we needed to pass some values. Each API has a standard way of passing values in the URL, which we looked up on the developer part of each respective website. There were some challenges related to getting the desired information. For example, the FourSquare api was recently updated and needed `HttpHeaders`. We had to figure out how to pass these headers in the `get()` request so that we could get the right response from the website.

Another aspect of web design is routing. Routing allows you to navigate to different pages based on the url. You can also set up aliases so that one url will redirect you to another url or component. We have basic routing set up in this ICP so that when the user is directed to the recipe-search module with the url <http://localhost:4200/search-recipe> and the url `localhost:4200/`.

The UI consists of 2 form fields where the user can enter a food and a place or city. This information is passed to the APIs and then displayed so that the user can select between different recipes or different restaurants to get that food.

## Default Screen

Find your Recipe!!!

Search for your favorite food!!!

Recipe name - Place

Find

Here's some recipes!

Check out these places!

## Search for Pizza in Kansas City

Search for your favorite food!!!

pizza - kansas city

Find

Here's some recipes!



Pizza Dough

<http://www.lottieanddoof.com/2010/01/pizza-pulp-fiction-jim-lahey/>

Calories: 2,620.98 kcal

Ingredients:

- 500 g bread flour(3 3/4 cups)
- 2 1/2 tsp Dry Yeast instant or active (10 grams)
- 3/4 tsp Table Salt(5 grams)
- 3/4 tsp Sugar, plus a pinch (about 3 grams)
- 1 1/2 cup water at room temperature
- extra-virgin olive oil for pans
- 2 x yellow onions(medium), finely chopped (pizza cipolla)
- 1/3 cup Heavy Cream(pizza cipolla)
- 1 tsp Kosher Salt(pizza cipolla)
- 2 tsp fresh thyme, coarsely chopped(pizza cipolla)
- 7 oz diced tomatoes, drained(pizza pomodoro)
- 3/4 cup Canned Tomatoes (reserved juice) (pizza pomodoro)
- 2 tsp Extra Virgin Olive Oil(pizza pomodoro)
- 1/2 tsp Kosher Salt(pizza pomodoro)
- 1 pinch Red Pepper Flakes(pizza pomodoro)

Check out these places!

Bella Napoli  
Italian Restaurant



Trezo Mare Restaurant and Lounge  
Restaurant



Bristol Seafood + Steak + Social  
Seafood Restaurant



Milwaukee Delicatessen Company  
Bar



Beer Kitchen  
Burger Joint



Cafe Gratitude  
Burger Joint



Up-Down  
Arcade



Cupini's  
Deli



Martin City Pizza & Tap Room  
Brewery



Il Lazzarone  
Pizzeria



## Search for Ice Cream in Kansas City

### Search for your favorite food!!!

#### Here's some recipes!



##### Rainbow Ice Cream Pops Recipe

<http://www.seriousseats.com/recipes/2012/07/rainbow-ice-cream-pops-recipe.html>

Calories: 307.39 kcal

##### Ingredients:

- 3 tablespoons pink-tinted ice cream, softened
- 3 tablespoons orange-tinted ice cream, softened
- 3 tablespoons yellow-tinted ice cream, softened
- 3 tablespoons green-tinted ice cream, softened
- 3 tablespoons blue-tinted ice cream, softened
- 3 tablespoons violet-tinted ice cream, softened



#### Check out these places!

The Nelson-Atkins Museum of Art  
Museum



Cooper's Hawk Winery & Restaurant  
American Restaurant



Kauffman Center for the Performing Arts  
Business and Professional Services



Cosentino's Market  
Grocery Store / Supermarket



Betty Rae's Ice Cream  
Ice Cream Parlor



Country Club Plaza  
Retail



Glacé Artisan Ice Cream  
Ice Cream Parlor



Trezo Mare Restaurant and Lounge  
Restaurant



Crows Coffee  
Coffee Shop



Andy's Frozen Custard  
Ice Cream Parlor



environment.ts

```
export const environment = {  
  production: false,  
  //Edaman application ID and KEY  
  APP_ID: "e2f0f7f3",  
  APP_KEY: "42de8fcd337d2185069fdacc8c6e9adf"  
};
```

## search-recipe-component.ts

```
app > search-recipe > ts search-recipe.component.ts > SearchRecipeComponent > getvenues
    });
  }

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

    if (this.recipeValue !== null) {
      //map storing data for the recipe
      this.searchService.getRecipes(this.recipeValue).subscribe(data => {
        this.recipeList = data.map(e => e.recipe)
      })
    }
    //if the user entered a place and a food, then display a list of restaurants
    if (this.placeValue !== null && this.placeValue !== '' && this.recipeValue !== null && this.recipeValue !== '') {
      //Set up http headers for the get request. These are needed so that foursquare can verify the account information
      const headers= new HttpHeaders()
        .set('Accept', 'application/json')
        .set('Authorization', 'fsq3lNgWYv9sczLrIgyn5/WI4acjplEA0JXYzR7DWfDEXmA=');

      //set a variable that contains our dynamic url
      var url = 'https://api.foursquare.com/v3/places/search?exclude_all_chains=true&sort=RATING&limit=50&query=' + this.recipeValue;

      //create a get request with headers
      this._http.get(url, { 'headers': headers })
        // subscribe triggers when data is successfully retrieved
        .subscribe((data:any) => {
          this.isLoading=false;
          // query the json object. the results is the output of the query
          this.venueList = Object.keys(data.results).map(function (k){
            var i = data.results[k];
            console.log(i);
            // return the needed restaurant name and location
            return {name: i.name, location: i.location, category: i.categories[0].name}
          })
          //Log the results to the console for testing
          console.log(this.venueList);
        });
    }
  }
}
```

## search-recipe.service.ts

```
app > search-recipe > ts search-recipe.service.ts > SearchRecipeService
import { Injectable } from '@angular/core';
import {environment as env} from './../../environments/environment';
import {HttpClient} from '@angular/common/http';
import { map } from 'rxjs/operators';
import { Observable } from 'rxjs';

@Injectable({
  providedIn: 'root'
})
export class SearchRecipeService {

  constructor(private http:HttpClient) { }
  //api url with the app id and app key to search for any recipe
  getRecipes(recipeName:String): Observable<any> {
    const EDAMAM_API_URL = `https://api.edamam.com/search?q=${recipeName}&app_id=${env.APP_ID}&app_key=${env.APP_KEY}`;
    return this.http.get(EDAMAM_API_URL).pipe(map(e => e['hits']))
  }
}
```

## search-recipe.html

```
app > search-recipe > search-recipe.component.html > div.container > div.row > div.col-lg-6 > div.panel.panel-default.col-6 > div.panel-body

```