Alternative Non-Collapsable Navigation Bar

The way we added it, the Navbar will collapse on smaller screens. Since we didn't implement a Hamburger menu, that means that there's no way of accessing our links on smaller screens.

You can either add such a menu on your own (see below), or you replace `collapse navbar-collapse` with just `navbar-default`.

Adding a Hamburger Menu:

Alternatively, if you want to make the navigation bar responsive, please replace these lines in `header.component.html`:

```html
1  <div class="navbar-header">
2    <a routerLink="/" class="navbar-brand">Recipe Book</a>
3  </div>
4  <div class="collapse navbar-collapse">
```

with these lines:

```html
1  <div class="navbar-header">
2    <button type="button" class="navbar-toggle" (click)="collapsed =
   !collapsed">
3      <span class="icon-bar" *ngFor="let iconBar of [1, 2, 3]"></span>
4    </button>
5    <a routerLink="/" class="navbar-brand">Recipe Book</a>
6  </div>
7  <div class="navbar-collapse" [class.collapse]="collapsed"
   (window:resize)="collapsed = true">
```

and add this line to `header.component.ts`:

```typescript
1  collapsed = true;
```

Alternative Non-Collapsable Navigation Bar

The way we added it, the Navbar will collapse on smaller screens. Since we didn't implement a Hamburger menu, that means that there's no way of accessing our links on smaller screens.

You can either add such a menu on your own (see below), or you replace `collapse navbar-collapse` with just `navbar-default`.

Adding a Hamburger Menu:

Alternatively, if you want to make the navigation bar responsive, please replace these lines in `header.component.html`:

```html
1   <div class="navbar-header">
2     <a routerLink="/" class="navbar-brand">Recipe Book</a>
3   </div>
4   <div class="collapse navbar-collapse">
```

with these lines:

```html
1   <div class="navbar-header">
2     <button type="button" class="navbar-toggle" (click)="collapsed = !collapsed">
3         <span class="icon-bar" *ngFor="let iconBar of [1, 2, 3]"></span>
4     </button>
5     <a routerLink="/" class="navbar-brand">Recipe Book</a>
6   </div>
7   <div class="navbar-collapse" [class.collapse]="collapsed" (window:resize)="collapsed = true">
```

and add this line to `header.component.ts`:

```
1   collapsed = true;
```

Alternative Non-Collapsable Navigation Bar

The way we added it, the Navbar will collapse on smaller screens. Since we didn't implement a Hamburger menu, that means that there's no way of accessing our links on smaller screens.

You can either add such a menu on your own (see below), or you replace `collapse navbar-collapse` with just `navbar-default`.

Adding a Hamburger Menu:

Alternatively, if you want to make the navigation bar responsive, please replace these lines in `header.component.html`:

```
1    <div class="navbar-header">
2      <a routerLink="/" class="navbar-brand">Recipe Book</a>
3    </div>
4    <div class="collapse navbar-collapse">
```

with these lines:

```
1    <div class="navbar-header">
2      <button type="button" class="navbar-toggle" (click)="collapsed =
     !collapsed">
3          <span class="icon-bar" *ngFor="let iconBar of [1, 2, 3]"></span>
4      </button>
5      <a routerLink="/" class="navbar-brand">Recipe Book</a>
6    </div>
7    <div class="navbar-collapse" [class.collapse]="collapsed"
     (window:resize)="collapsed = true">
```

and add this line to `header.component.ts`:

```
1    collapsed = true;
```

Alternative Non-Collapsable Navigation Bar

The way we added it, the Navbar will collapse on smaller screens. Since we didn't implement a Hamburger menu, that means that there's no way of accessing our links on smaller screens.

You can either add such a menu on your own (see below), or you replace `collapse navbar-collapse` with just `navbar-default`.

Adding a Hamburger Menu:

Alternatively, if you want to make the navigation bar responsive, please replace these lines in `header.component.html`:

```html
1    <div class="navbar-header">
2      <a routerLink="/" class="navbar-brand">Recipe Book</a>
3    </div>
4    <div class="collapse navbar-collapse">
```

with these lines:

```html
1    <div class="navbar-header">
2      <button type="button" class="navbar-toggle" (click)="collapsed =
     !collapsed">
3          <span class="icon-bar" *ngFor="let iconBar of [1, 2, 3]"></span>
4      </button>
5      <a routerLink="/" class="navbar-brand">Recipe Book</a>
6    </div>
7    <div class="navbar-collapse" [class.collapse]="collapsed"
     (window:resize)="collapsed = true">
```

and add this line to `header.component.ts`:

```typescript
1    collapsed = true;
```

Alternative Non-Collapsable Navigation Bar

The way we added it, the Navbar will collapse on smaller screens. Since we didn't implement a Hamburger menu, that means that there's no way of accessing our links on smaller screens.

You can either add such a menu on your own (see below), or you replace `collapse navbar-collapse` with just `navbar-default`.

Adding a Hamburger Menu:

Alternatively, if you want to make the navigation bar responsive, please replace these lines in `header.component.html` :

```
1   <div class="navbar-header">
2     <a routerLink="/" class="navbar-brand">Recipe Book</a>
3   </div>
4   <div class="collapse navbar-collapse">
```

with these lines:

```
1   <div class="navbar-header">
2     <button type="button" class="navbar-toggle" (click)="collapsed =
    !collapsed">
3         <span class="icon-bar" *ngFor="let iconBar of [1, 2, 3]"></span>
4     </button>
5     <a routerLink="/" class="navbar-brand">Recipe Book</a>
6   </div>
7   <div class="navbar-collapse" [class.collapse]="collapsed"
    (window:resize)="collapsed = true">
```

and add this line to `header.component.ts` :

```
1   collapsed = true;
```