Header.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-header',
  templateUrl: './header.component.html'
})
export class HeaderComponent {
}
```

Header.component.html

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-header',
  templateUrl: './header.component.html'
})
export class HeaderComponent {
}
```

Src > app > shopping-list > shopping-edit

Shopping-edit.component.ts

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

import { Ingredient } from '../../shared/ingredient.model';
import { ShoppingListService } from '../shopping-list.service';

@Component({
  selector: 'app-shopping-edit',
  templateUrl: './shopping-edit.component.html',
  styleUrls: ['./shopping-edit.component.css']
})
export class ShoppingEditComponent implements OnInit {
  @ViewChild('nameInput', { static: false }) nameInputRef: ElementRef;
  @ViewChild('amountInput', { static: false }) amountInputRef: ElementRef;

  constructor(private slService: ShoppingListService) { }

  ngOnInit() {
  }

  onAddItem() {
    const ingName = this.nameInputRef.nativeElement.value;
    const ingAmount = this.amountInputRef.nativeElement.value;
    const newIngredient = new Ingredient(ingName, ingAmount);
    this.slService.addIngredient(newIngredient);
  }

}
```

Shopping-edit.component.html

```
<div class="row">
  <div class="col-xs-12">
    <form>
      <div class="row">
        <div class="col-sm-5 form-group">
          <label for="name">Name</label>
          <input
            type="text"
            id="name"
            class="form-control"
            #nameInput>
        </div>
        <div class="col-sm-2 form-group">
          <label for="amount">Amount</label>
          <input
            type="number"
            id="amount"
            class="form-control"
            #amountInput>
        </div>
      </div>
      <div class="row">
        <div class="col-xs-12">
          <button class="btn btn-success" type="submit" (click)="onAddItem()">Add</button>
          <button class="btn btn-danger" type="button">Delete</button>
          <button class="btn btn-primary" type="button">Clear</button>
        </div>
      </div>
    </form>
  </div>
</div>
```

Shopping-list.service.ts

```
import { Ingredient } from '../shared/ingredient.model';
import { EventEmitter } from '@angular/core';

export class ShoppingListService {
  ingredientsChanged = new EventEmitter<Ingredient[]>();
  private ingredients: Ingredient[] = [
    new Ingredient('Apples', 5),
    new Ingredient('Tomatoes', 10),
  ];

  getIngredients() {
    return this.ingredients.slice();
  }

  addIngredient(ingredient: Ingredient) {
    this.ingredients.push(ingredient);
    this.ingredientsChanged.emit(this.ingredients.slice());
  }

  addIngredients(ingredients: Ingredient[]) {
    // for (let ingredient of ingredients) {
    //   this.addIngredient(ingredient);
    // }
    this.ingredients.push(...ingredients);
    this.ingredientsChanged.emit(this.ingredients.slice());
  }
}
```

Shopping-list.component.ts

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

import { Ingredient } from '../shared/ingredient.model';
import { ShoppingListService } from './shopping-list.service';

@Component({
  selector: 'app-shopping-list',
  templateUrl: './shopping-list.component.html',
  styleUrls: ['./shopping-list.component.css']
})
export class ShoppingListComponent implements OnInit {
  ingredients: Ingredient[];

  constructor(private slService: ShoppingListService) { }

  ngOnInit() {
    this.ingredients = this.slService.getIngredients();
    this.slService.ingredientsChanged
      .subscribe(
        (ingredients: Ingredient[]) => {
          this.ingredients = ingredients;
        }
      );
  }
}
```

Shopping-list.component.html

```
<div class="row">
  <div class="col-xs-10">
    <app-shopping-edit></app-shopping-edit>
    <hr>
    <ul class="list-group">
      <a
        class="list-group-item"
        style="cursor: pointer"
        *ngFor="let ingredient of ingredients"
      >
        {{ ingredient.name }} ({{ ingredient.amount }})
      </a>
    </ul>
  </div>
</div>
```

Src > app > shared

Dropdown.directive.ts

```
import { Directive, HostListener, HostBinding } from '@angular/core';

@Directive({
  selector: '[appDropdown]'
})
export class DropdownDirective {
  @HostBinding('class.open') isOpen = false;

  @HostListener('click') toggleOpen() {
    this.isOpen = !this.isOpen;
  }
}
```

Ingredient.model.ts

```
export class Ingredient {
  constructor(public name: string, public amount: number) {}
}
```

Src > app > recipes > recipe-list > recipe-item

Recipe-item.component.ts

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

import { Recipe } from '../../recipe.model';

@Component({
  selector: 'app-recipe-item',
  templateUrl: './recipe-item.component.html',
  styleUrls: ['./recipe-item.component.css']
})
export class RecipeItemComponent implements OnInit {
  @Input() recipe: Recipe;
  @Input() index: number;

  ngOnInit() {
  }
}
```

Recipe-item.component.html

```
<a
  style="cursor: pointer;"
  [routerLink]="[index]"
  routerLinkActive="active"
  class="list-group-item clearfix">
  <div class="pull-left">
    <h4 class="list-group-item-heading">{{ recipe.name }}</h4>
    <p class="list-group-item-text">{{ recipe.description }}</p>
  </div>
  <span class="pull-right">
        <img
          [src]="recipe.imagePath"
          alt="{{ recipe.name }}"
          class="img-responsive"
          style="max-height: 50px;">
      </span>
</a>
```

Src > app > recipes > recipe-list

Recipe-list.component.ts

```
import { Component, OnInit } from '@angular/core';
import { Router, ActivatedRoute } from '@angular/router';

import { Recipe } from '../recipe.model';
import { RecipeService } from '../recipe.service';

@Component({
  selector: 'app-recipe-list',
  templateUrl: './recipe-list.component.html',
  styleUrls: ['./recipe-list.component.css']
})
export class RecipeListComponent implements OnInit {
  recipes: Recipe[];

  constructor(private recipeService: RecipeService,
              private router: Router,
              private route: ActivatedRoute) {
  }

  ngOnInit() {
    this.recipes = this.recipeService.getRecipes();
  }

  onNewRecipe() {
    this.router.navigate(['new'], {relativeTo: this.route});
  }
}
```

Recipe-list.component.html

```
<div class="row">
  <div class="col-xs-12">
    <button class="btn btn-success" (click)="onNewRecipe()">New
Recipe</button>
  </div>
</div>
<hr>
<div class="row">
  <div class="col-xs-12">
    <app-recipe-item
      *ngFor="let recipeEl of recipes; let i = index"
      [recipe]="recipeEl"
      [index]="i"></app-recipe-item>
  </div>
</div>
```

Recipes > recipe-start

Recipe-start.component.ts

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

@Component({
  selector: 'app-recipe-start',
  templateUrl: './recipe-start.component.html',
  styleUrls: ['./recipe-start.component.css']
})
export class RecipeStartComponent implements OnInit {

  constructor() { }

  ngOnInit() {
  }

}
```

Recipe-start.component.html

```
<h3>Please select a Recipe!</h3>
```

Recipes > recipe-edit

Recipe-edit.component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Params } from '@angular/router';

@Component({
  selector: 'app-recipe-edit',
  templateUrl: './recipe-edit.component.html',
  styleUrls: ['./recipe-edit.component.css']
})
export class RecipeEditComponent implements OnInit {
  id: number;
  editMode = false;

  constructor(private route: ActivatedRoute) { }

  ngOnInit() {
    this.route.params
      .subscribe(
        (params: Params) => {
          this.id = +params['id'];
          this.editMode = params['id'] != null;
        }
      );
  }

}
```

Recipe-edit.component.html

```
<p>
  recipe-edit works!
</p>
```

Recipes > recipe-detail

Recipe-detail.component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Params, Router } from '@angular/router';

import { Recipe } from '../recipe.model';
import { RecipeService } from '../recipe.service';

@Component({
  selector: 'app-recipe-detail',
  templateUrl: './recipe-detail.component.html',
  styleUrls: ['./recipe-detail.component.css']
})
export class RecipeDetailComponent implements OnInit {
  recipe: Recipe;
  id: number;

  constructor(private recipeService: RecipeService,
              private route: ActivatedRoute,
              private router: Router) {
  }

  ngOnInit() {
    this.route.params
```

```
      .subscribe(
        (params: Params) => {
          this.id = +params['id'];
          this.recipe = this.recipeService.getRecipe(this.id);
        }
      );
  }

  onAddToShoppingList() {

this.recipeService.addIngredientsToShoppingList(this.recipe.ingredients
);
  }

  onEditRecipe() {
    this.router.navigate(['edit'], {relativeTo: this.route});
    // this.router.navigate(['../', this.id, 'edit'], {relativeTo:
this.route});
  }

}
```

Recipe-detail.component.html

```
<div class="row">
  <div class="col-xs-12">
    <img
      [src]="recipe.imagePath"
      alt="{{ recipe.name }}"
      class="img-responsive"
      style="max-height: 300px;">
  </div>
</div>
<div class="row">
  <div class="col-xs-12">
    <h1>{{ recipe.name }}</h1>
  </div>
</div>
<div class="row">
```

```
  <div class="col-xs-12">
    <div
      class="btn-group"
      appDropdown>
      <button
        type="button"
        class="btn btn-primary dropdown-toggle">
        Manage Recipe <span class="caret"></span>
      </button>
      <ul class="dropdown-menu">
        <li><a (click)="onAddToShoppingList()" style="cursor: pointer;">To Shopping List</a></li>
        <li><a style="cursor: pointer;" (click)="onEditRecipe()">Edit Recipe</a></li>
        <li><a style="cursor: pointer;">Delete Recipe</a></li>
      </ul>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-xs-12">
    {{ recipe.description }}
  </div>
</div>
<div class="row">
  <div class="col-xs-12">
    <ul class="list-group">
      <li
        class="list-group-item"
        *ngFor="let ingredient of recipe.ingredients">
        {{ ingredient.name }} - {{ ingredient.amount }}
      </li>
    </ul>
  </div>
</div>
```

App > recipes

Recipes.component.ts

```
import { Component, OnInit } from '@angular/core';
import { Recipe } from './recipe.model';
import { RecipeService } from './recipe.service';

@Component({
  selector: 'app-recipes',
  templateUrl: './recipes.component.html',
  styleUrls: ['./recipes.component.css'],
  providers: [RecipeService]
})
export class RecipesComponent implements OnInit {
  selectedRecipe: Recipe;

  constructor(private recipeService: RecipeService) { }

  ngOnInit() {
    this.recipeService.recipeSelected
      .subscribe(
        (recipe: Recipe) => {
          this.selectedRecipe = recipe;
        }
      );
  }
}
```

Recipes.component.html

```
<div class="row">
  <div class="col-md-5">
    <app-recipe-list></app-recipe-list>
  </div>
  <div class="col-md-7">
    <router-outlet></router-outlet>
  </div>
</div>
```

Recipes.service.ts

```
import { EventEmitter, Injectable } from '@angular/core';

import { Recipe } from './recipe.model';
import { Ingredient } from '../shared/ingredient.model';
import { ShoppingListService } from
'../shopping-list/shopping-list.service';

@Injectable()
export class RecipeService {
  recipeSelected = new EventEmitter<Recipe>();

  private recipes: Recipe[] = [
    new Recipe(
      'Tasty Schnitzel',
      'A super-tasty Schnitzel - just awesome!',

'https://upload.wikimedia.org/wikipedia/commons/7/72/Schnitzel.JPG',
      [
        new Ingredient('Meat', 1),
        new Ingredient('French Fries', 20)
      ]),
    new Recipe('Big Fat Burger',
      'What else you need to say?',

'https://upload.wikimedia.org/wikipedia/commons/b/be/Burger_King_Angus_
Bacon_%26_Cheese_Steak_Burger.jpg',
      [
        new Ingredient('Buns', 2),
        new Ingredient('Meat', 1)
      ])
  ];

  constructor(private slService: ShoppingListService) {}

  getRecipes() {
    return this.recipes.slice();
  }

  getRecipe(index: number) {
    return this.recipes[index];
  }

  addIngredientsToShoppingList(ingredients: Ingredient[]) {
    this.slService.addIngredients(ingredients);
  }
}
```

Recipe.model.ts

```
import { Ingredient } from '../shared/ingredient.model';

export class Recipe {
  public name: string;
  public description: string;
  public imagePath: string;
  public ingredients: Ingredient[];

  constructor(name: string, desc: string, imagePath: string,
ingredients: Ingredient[]) {
    this.name = name;
    this.description = desc;
    this.imagePath = imagePath;
    this.ingredients = ingredients;
  }
}
```

Src > app

App-routing.module.ts

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

import { RecipesComponent } from './recipes/recipes.component';
import { ShoppingListComponent } from
'./shopping-list/shopping-list.component';
import { RecipeStartComponent } from
'./recipes/recipe-start/recipe-start.component';
import { RecipeDetailComponent } from
'./recipes/recipe-detail/recipe-detail.component';
import { RecipeEditComponent } from
'./recipes/recipe-edit/recipe-edit.component';

const appRoutes: Routes = [
  { path: '', redirectTo: '/recipes', pathMatch: 'full' },
  { path: 'recipes', component: RecipesComponent, children: [
    { path: '', component: RecipeStartComponent },
    { path: 'new', component: RecipeEditComponent },
    { path: ':id', component: RecipeDetailComponent },
    { path: ':id/edit', component: RecipeEditComponent },
  ] },
  { path: 'shopping-list', component: ShoppingListComponent },
];

@NgModule({
  imports: [RouterModule.forRoot(appRoutes)],
  exports: [RouterModule]
})
export class AppRoutingModule {

}
```

App.module.ts

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';


import { AppComponent } from './app.component';
import { HeaderComponent } from './header/header.component';
import { RecipesComponent } from './recipes/recipes.component';
import { RecipeListComponent } from
'./recipes/recipe-list/recipe-list.component';
import { RecipeDetailComponent } from
'./recipes/recipe-detail/recipe-detail.component';
import { RecipeItemComponent } from
'./recipes/recipe-list/recipe-item/recipe-item.component';
import { ShoppingListComponent } from
'./shopping-list/shopping-list.component';
import { ShoppingEditComponent } from
'./shopping-list/shopping-edit/shopping-edit.component';
import { DropdownDirective } from './shared/dropdown.directive';
import { ShoppingListService } from
'./shopping-list/shopping-list.service';
import { AppRoutingModule } from './app-routing.module';
import { RecipeStartComponent } from
'./recipes/recipe-start/recipe-start.component';
import { RecipeEditComponent } from
'./recipes/recipe-edit/recipe-edit.component';

@NgModule({
  declarations: [
    AppComponent,
    HeaderComponent,
    RecipesComponent,
    RecipeListComponent,
    RecipeDetailComponent,
    RecipeItemComponent,
    ShoppingListComponent,
    ShoppingEditComponent,
    DropdownDirective,
    RecipeStartComponent,
    RecipeEditComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    AppRoutingModule
  ],
  providers: [ShoppingListService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

App.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  loadedFeature = 'recipe';
  onNavigate(feature: string) {
    this.loadedFeature = feature;
  }
}
```

App.component.html

```
<app-header (featureSelected)="onNavigate($event)"></app-header>
<div class="container">
  <div class="row">
    <div class="col-md-12">
      <router-outlet></router-outlet>
    </div>
  </div>
</div>
```