

Cross Platform Mobile Programming

IF733

Ionic with Angular (II)

Andre Rusli

Review

- Installing Ionic and starting an Ionic Angular app
- Working with Page, Interface, and Services

src/app/recipes/recipes.service.ts

```
constructor() { }

getAllRecipes() {
  return [...this.recipes];}

getRecipe(recipeId: string) {

}

deleteRecipe(recipeId: string) {
}
```

Agenda

- Route Parameters
- Navigate Between Pages Using Angular Router
- Ionic Controller Component: Alert & Toast

Route Params

1/3

Working with Route Params

-Setting Up the Routing Module-

Set which page component gets loaded
when a specific URL with parameter is requested.

src/app/app-routing.module.ts

```
const routes: Routes = [
  { path: '', redirectTo: 'recipes', pathMatch: 'full' },
  {
    path: 'recipes',
    children: [
      {
        path: '',
        loadChildren: './recipes/recipes.module#RecipesPageModule'
      },
      {
        path: ':recipeId',
        loadChildren: './recipes/recipe-detail/recipe-detail.module#RecipeDetailPageModule'
      }
    ]
];
```

Working with Route Params

-Getting the params using ActivatedRoute-

Receive the parameter, get data from Service,
and store it in a local variable.

src/app/recipes/recipe-detail/recipe-detail.page.ts

```
export class RecipeDetailPage implements OnInit {
  loadedRecipe: Recipe;
  constructor(private activatedRoute: ActivatedRoute, private recipesSvc: RecipesService) { }

  ngOnInit() {
    this.activatedRoute.paramMap.subscribe(
      paramMap => {
        if (!paramMap.has('recipeId')) { return; }
        this.loadedRecipe = this.recipesSvc.getRecipe(paramMap.get('recipeId'));
      });
  }
}
```

Working with Route Params

-Displaying the parameter's value-

src/app/recipes/recipe-detail/recipe-detail.page.html

```
<ion-header>
  <ion-toolbar>
    <ion-title>{{ loadedRecipe.title }}</ion-title>
  </ion-toolbar>
</ion-header>

<ion-content>
  <ion-grid>
    <ion-row>
      <ion-col class="ion-no-padding">
        <ion-img [src]="loadedRecipe.imageUrl"></ion-img>
      </ion-col>
    </ion-row>
    <ion-row>
      <ion-col>
        <ion-item *ngFor="let i of loadedRecipe.ingredients">{{ i }}</ion-item>
      </ion-col>
    </ion-row>
  </ion-grid>
</ion-content>
```

<http://localhost:8100/recipes/r3>

Pizza Margerita



Adonan Pizza

Keju

Saus Tomat

<http://localhost:8100/recipes/r1>

Gado-gado



Timun

Telur

Bumbu Kecap

Tauge

Navigate Between Pages Using Angular Router

2/3

RouterLink and <ion-back-button>

src/app/recipes/recipes.page.html

```
ion-item *ngFor="let recipe of recipes" [routerLink]="['./', recipe.id]">
  <ion-avatar slot="start">
    <ion-img [src]="recipe.imageUrl"></ion-img>
  </ion-avatar>
  <ion-label>{{ recipe.title }}</ion-label>
</ion-item>
```

src/app/recipes/recipe-detail/recipe-detail.page.html

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button defaultHref="/recipes"></ion-back-button>
    </ion-buttons>
    <ion-title>{{ loadedRecipe.title }}</ion-title>
  </ion-toolbar>
</ion-header>
```

RouterLink and <ion-back-button>

<http://localhost:8100/recipes>

Recipes

-  Gado-gado >
-  Ketupat >
-  Pizza Margerita >

<http://localhost:8100/recipes/r1>

< Back Gado-gado



Timun

Telur

Bumbu Kecap

Tauge

Adding a Delete Button

src/app/recipes/recipe-detail/recipe-detail.page.html

```
<ion-header>
  <ion-toolbar>
    <ion-buttons slot="start">
      <ion-back-button defaultHref="/recipes"></ion-back-button>
    </ion-buttons>
    <ion-buttons slot="primary">
      <ion-button color="danger" (click)="deleteRecipe()">
        <ion-icon name="trash" slot="icon-only"></ion-icon>
      </ion-button>
    </ion-buttons>
    <ion-title>{{ loadedRecipe.title }}</ion-title>
  </ion-toolbar>
</ion-header>
```

Adding a Delete Button

[Back](#) **Gado-gado**



Timun

Telur

Bumbu Kecap

Tauge

Using Angular Router

and Deleting a Recipe

src/app/recipes/recipe-detail/recipe-detail.page.html

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

...
export class RecipeDetailPage implements OnInit {
  loadedRecipe: Recipe;
  constructor(private activatedRoute: ActivatedRoute, private recipesSvc: RecipesService,
  private router: Router) { }

  ...
  deleteRecipe() {
    this.recipesSvc.deleteRecipe(this.loadedRecipe.id);
    this.router.navigate(['/recipes']);
  }
}
```

Ionic Controller Component: Alert & Toast

3/3

Ionic Controller Component

-AlertController-

Usage

ANGULAR

JAVASCRIPT

REACT

VUE

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

@Component({
  selector: 'alert-example',
  templateUrl: 'alert-example.html',
  styleUrls: ['./alert-example.css'],
})
export class AlertExample {

  constructor(public alertController: AlertController) {}

  async presentAlert() {
    const alert = await this.alertController.create({
      header: 'Alert',
      subHeader: 'Subtitle',
      message: 'This is an alert message.',
      buttons: ['OK']
    });

    await alert.present();
  }
}
```

```
async presentAlert() {
  const alert = await this.alertController.create({
    header: 'Delete Recipe',
    message: 'Are you sure you want to delete this
recipe?',
    buttons: [
      {
        text: 'YES',
        handler: () => this.deleteRecipe()
      },
      {
        text: 'Cancel',
        role: 'cancel'
      }
    ]
  });
  await alert.present();
}
```

Ionic Controller Component

-AlertController-

Properties

| animated | |
|-----------------|--|
| Description | If <code>true</code> , the alert will animate. |
| Attribute | <code>animated</code> |
| Type | <code>boolean</code> |
| Default | <code>true</code> |
| backdropDismiss | |
| Description | If <code>true</code> , the alert will be dismissed when the backdrop is clicked. |
| Attribute | <code>backdrop-dismiss</code> |
| Type | <code>boolean</code> |
| Default | <code>true</code> |
| buttons | |
| Description | Array of buttons to be added to the alert. |
| Type | <code>(string AlertButton)[]</code> |
| Default | <code>[]</code> |

header

| | |
|-------------|---|
| Description | The main title in the heading of the alert. |
| Attribute | <code>header</code> |
| Type | <code>string undefined</code> |

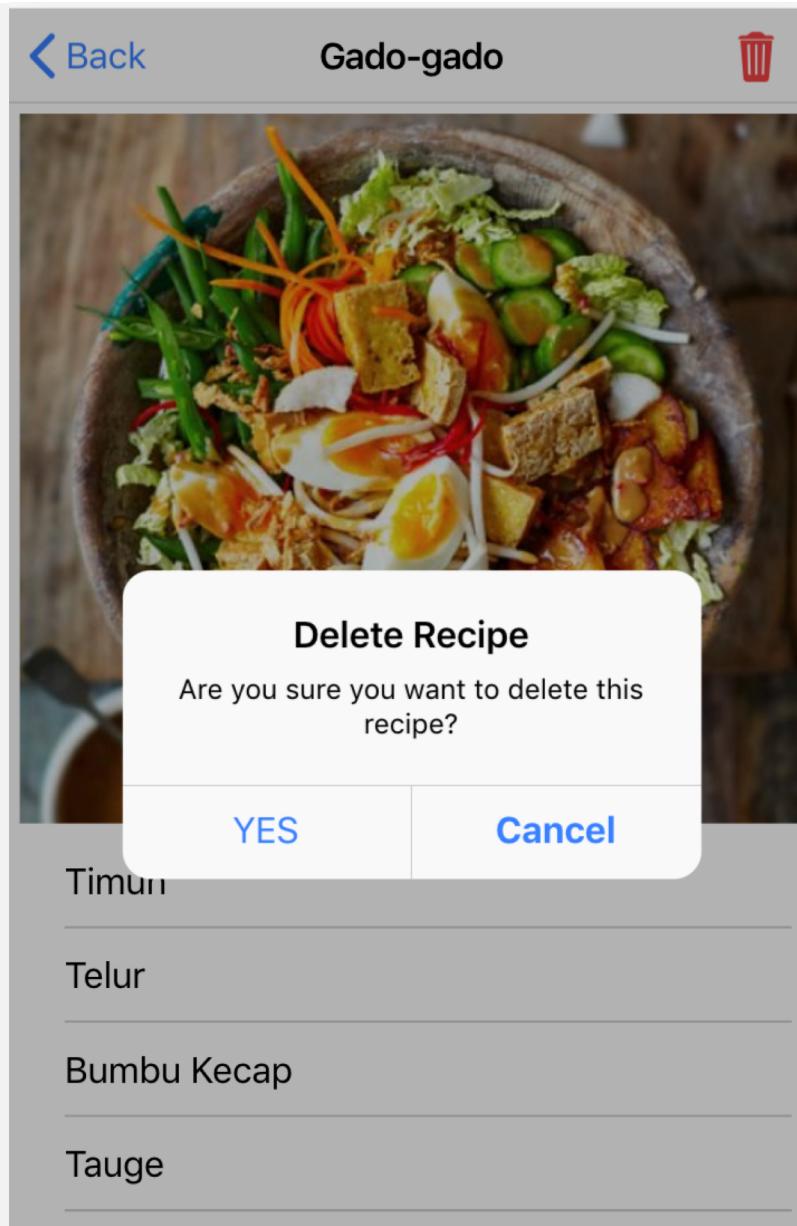
inputs

| | |
|-------------|--------------------------------------|
| Description | Array of input to show in the alert. |
| Type | <code>AlertInput[]</code> |
| Default | <code>[]</code> |

keyboardClose

| | |
|-------------|--|
| Description | If <code>true</code> , the keyboard will be automatically dismissed when the overlay is presented. |
| Attribute | <code>keyboard-close</code> |
| Type | <code>boolean</code> |
| Default | <code>true</code> |

And many more ...



Ionic Controller Component

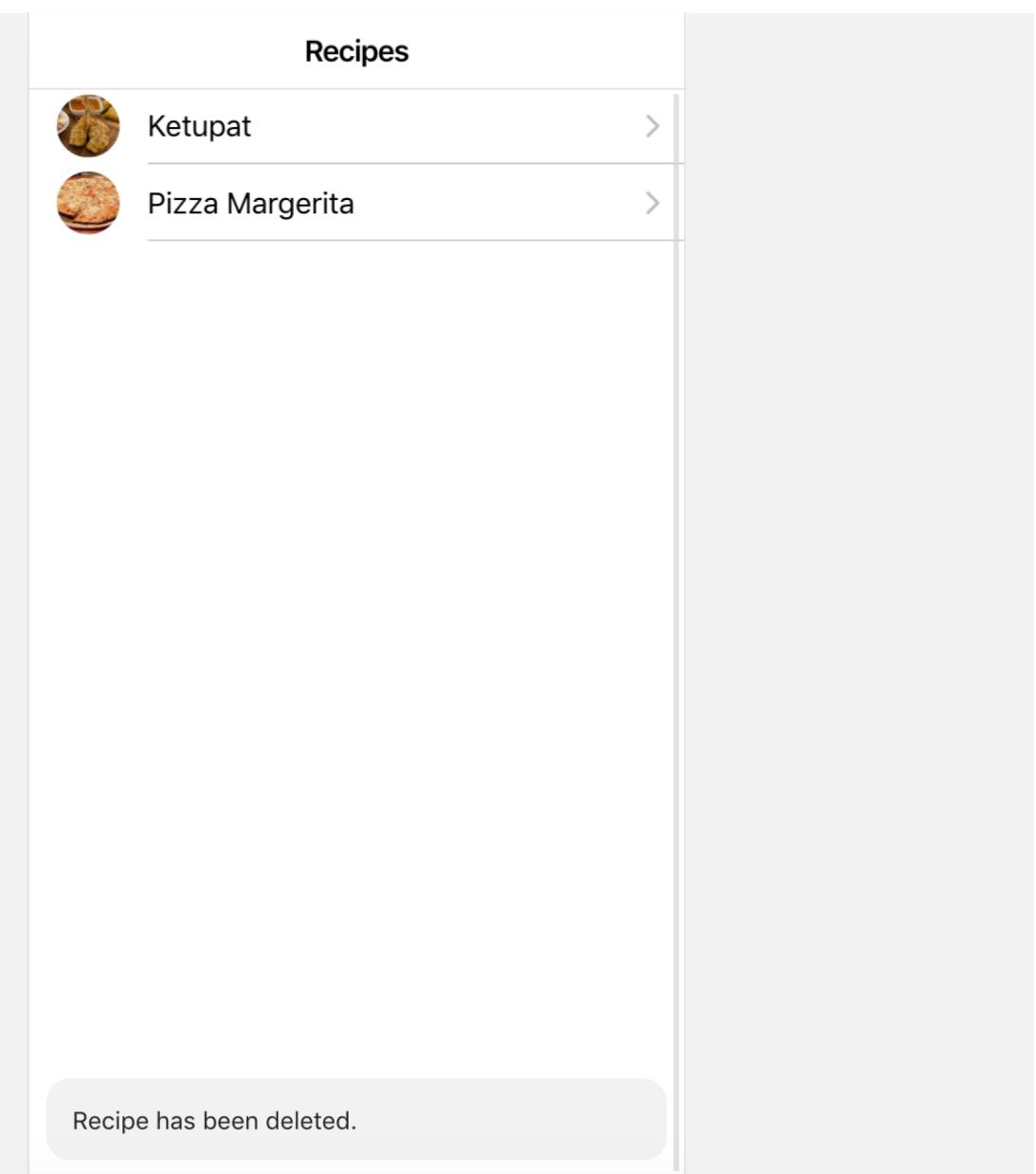
-AlertController-

Ionic Controller Component

-ToastController-

<https://ionicframework.com/docs/api/toast>

Check the documentation on how to do this.



The screenshot shows a mobile application interface with a dark blue header bar at the top. Below the header, there is a white content area. At the top of this area, the word "Recipes" is centered in a bold black font. Below "Recipes", there is a horizontal list of two items, each consisting of a small circular thumbnail image on the left, the item name in the center, and a right-pointing arrow icon on the far right. The first item is "Ketupat" with a thumbnail showing a dish of rice. The second item is "Pizza Margerita" with a thumbnail showing a slice of pizza. At the bottom of the white content area, there is a light gray rounded rectangular toast notification with the text "Recipe has been deleted." inside it.

| Item | Description | Action |
|-----------------|-------------------|--------|
| Ketupat | A dish of rice. | > |
| Pizza Margerita | A slice of pizza. | > |

Recipe has been deleted.

Questions?

Comments?

Thank you