Angular 15 Task Manager Project with Bootstrap 4

Introduction

This document provides a step-by-step guide to create a mini project using Angular 15 with Bootstrap 4. The project is a simple Task Manager application where users can add, view, and delete tasks.

Step 1: Create a New Angular Project

1. Create a new Angular project by running the following commands:

```sh
ng new task-manager
cd task-manager
```

Step 2: Install Bootstrap 4

2. Install Bootstrap 4 via npm and add it to `angular.json`:

```sh
npm install bootstrap@4
```

Add Bootstrap to `angular.json`:

```json
"styles": [
  "node_modules/bootstrap/dist/css/bootstrap.min.css",
  "src/styles.css"
],
"scripts": [
  "node_modules/bootstrap/dist/js/bootstrap.min.js"
]
```

Step 3: Create Components
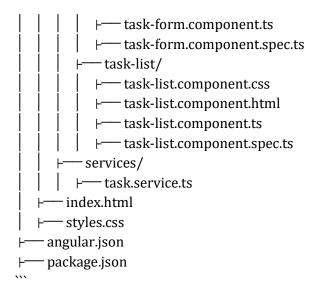
3. Generate the required components using Angular CLI:

```sh
ng generate component components/task-list
ng generate component components/task-form
```

Step 4: Implement Task Service

4. Generate a service to manage tasks and implement the task service (`task.service.ts`):

```sh
ng generate service services/task
```

Implement the task service (`task.service.ts`):

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

@Injectable({
  providedIn: 'root'
})
export class TaskService {
  private tasks: string[] = [];

  constructor() { }

  getTasks(): string[] {
    return this.tasks;
  }

  addTask(task: string) {
    this.tasks.push(task);
  }

  deleteTask(index: number) {
    this.tasks.splice(index, 1);
  }
}
```

Step 5: Implement Task Form Component

5. Update the `task-form.component.ts` and `task-form.component.html`:

```typescript
import { Component } from '@angular/core';
import { TaskService } from '../../services/task.service';

@Component({
  selector: 'app-task-form',
  templateUrl: './task-form.component.html',
```

```
  styleUrls: ['./task-form.component.css']
})
export class TaskFormComponent {
 task: string = '';

 constructor(private taskService: TaskService) { }

 addTask() {
  if (this.task) {
   this.taskService.addTask(this.task);
   this.task = '';
  }
 }
}
```

```html
<div class="card">
 <div class="card-body">
   <h5 class="card-title">Add Task</h5>
   <form (submit)="addTask()">
    <div class="form-group">
     <input type="text" class="form-control" [(ngModel)]="task" name="task"
placeholder="Enter task" required>
    </div>
    <button type="submit" class="btn btn-primary">Add Task</button>
   </form>
 </div>
</div>
```

Step 6: Implement Task List Component

6. Update the `task-list.component.ts` and `task-list.component.html`:

```typescript
import { Component } from '@angular/core';
import { TaskService } from '../../services/task.service';

@Component({
 selector: 'app-task-list',
 templateUrl: './task-list.component.html',
 styleUrls: ['./task-list.component.css']
})
export class TaskListComponent {
```

```
  constructor(public taskService: TaskService) { }

  deleteTask(index: number) {
    this.taskService.deleteTask(index);
  }
}
```

```html
<div class="card mt-4">
 <div class="card-body">
  <h5 class="card-title">Task List</h5>
  <ul class="list-group">
   <li *ngFor="let task of taskService.getTasks(); let i = index" class="list-group-item d-flex
justify-content-between align-items-center">
    {{ task }}
    <button class="btn btn-danger btn-sm" (click)="deleteTask(i)">Delete</button>
   </li>
  </ul>
 </div>
</div>
```

Step 7: Update App Component

7. Update the `app.component.html`:

```html
<div class="container mt-4">
 <h1 class="text-center">Task Manager</h1>
 <app-task-form></app-task-form>
 <app-task-list></app-task-list>
</div>
```

Solution Structure

```plaintext
task-manager/
├── src/
│   ├── app/
│   │   ├── components/
│   │   │   ├── task-form/
│   │   │   │   ├── task-form.component.css
│   │   │   │   ├── task-form.component.html
```

```
│   │   │   │   ├── task-form.component.ts
│   │   │   │   ├── task-form.component.spec.ts
│   │   │   ├── task-list/
│   │   │   │   ├── task-list.component.css
│   │   │   │   ├── task-list.component.html
│   │   │   │   ├── task-list.component.ts
│   │   │   │   ├── task-list.component.spec.ts
│   │   ├── services/
│   │   │   ├── task.service.ts
│   ├── index.html
│   ├── styles.css
├── angular.json
├── package.json
```

Final Steps

1. Run the application by executing the following command:

```sh
ng serve
```

2. Open the application in the browser by navigating to `http://localhost:4200`.