



If you do not pay any relevant course fee to maintain Earth University BIT platform,  
Be ethical enough to contribute while you are using our Academic Contents

www.earth.lk/community  
2004-2024 (20th Anniversary)

# Sprint-2 Plan

## Employee Detail Management

1. Establishing Project Folder Architecture
2. Server App Initialization
3. **Sprint-2 Execution** [ 2(a), **2(b)**, 2(c), 2(d), 2(e), 2(f) ]
4. Completing Sprint-1 Objectives

### 2(a) Employee Module - Analysis, Design & DB-Preparation

Data Design, UI Design, Sample Data, Database Preparation

### 2(b) Employee View

1. [Controller Design](#)
2. [Coding-Server App](#)
  - [Version-1 \(Client Pagination\)](#)
    - [Entity Coding, DAO Coding, Controller Coding, Testing](#)
  - Version-2 (Server Pagination)
    - Controller Modification, Testing
3. [Coding-Client App](#)
  - [Version-1 \(Client Pagination\)](#)
    - [UI Preparation, Entity Coding, Service Coding, Controller Coding, Testing](#)
  - Version-2 (Server Pagination)
    - Service Modification, Controller Modification, Testing

### 2(c) Employee Search

### 2(d) Employee Insert

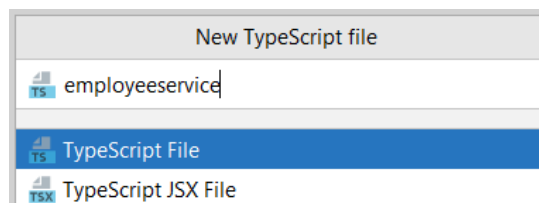
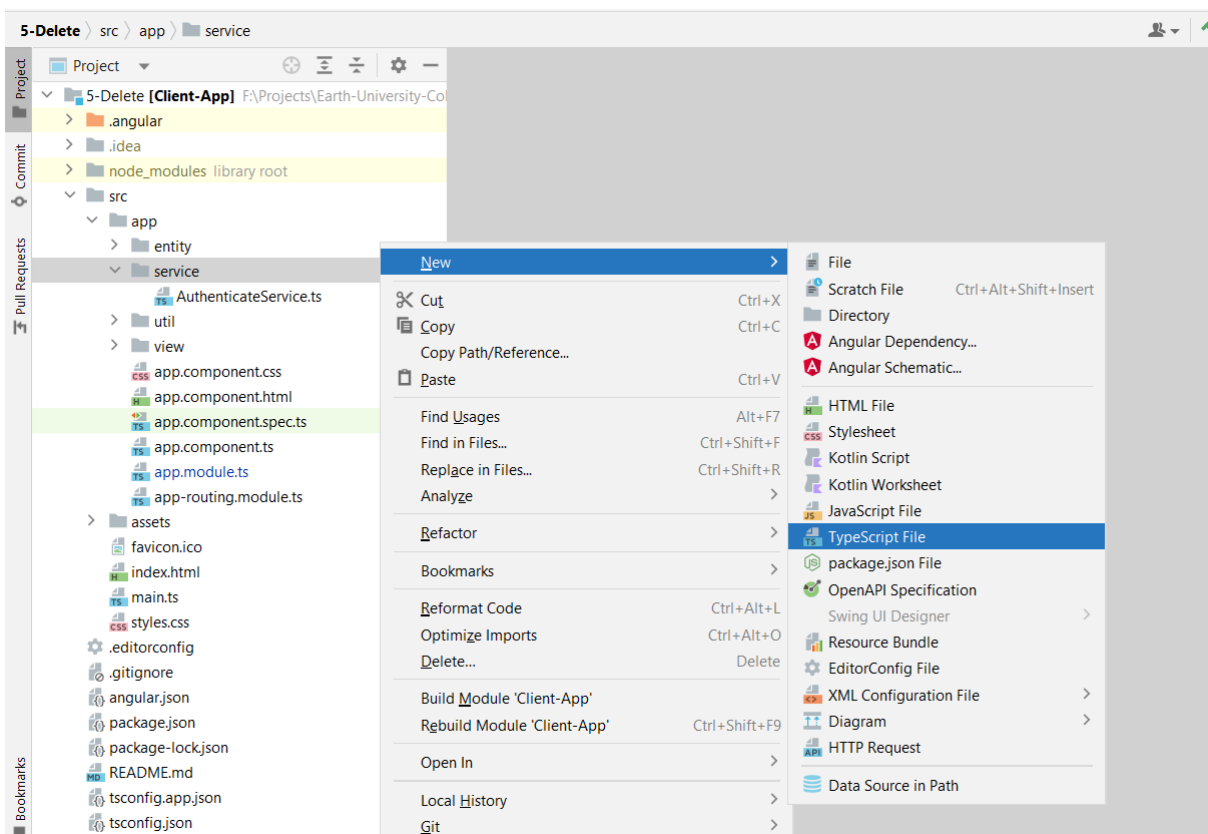
### 2(e) Employee Update

### 2(f) Employee Delete

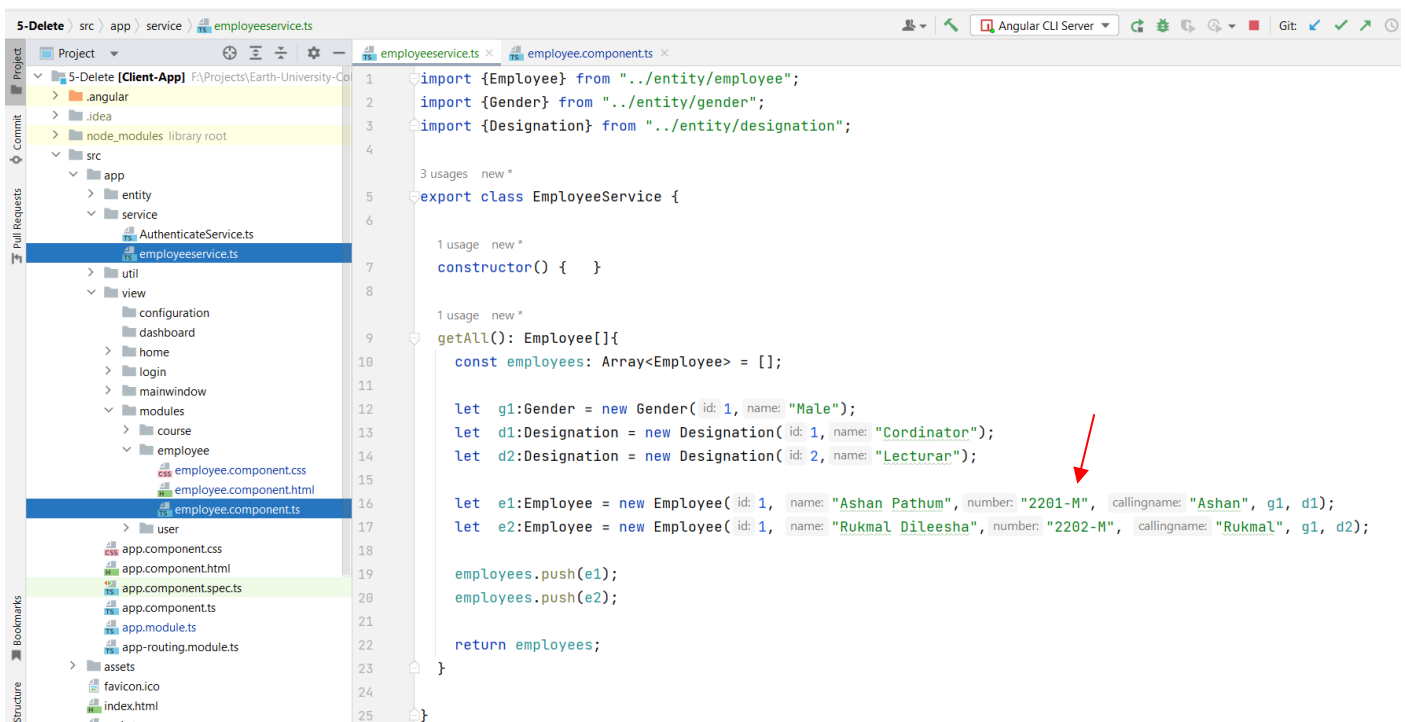
### 2(b) Employee View

1. [Controller Design](#)
2. [Coding Server-App](#)
3. [Coding Client-App](#)
  - Version-1 (Client Pagination)**
    1. Client App Preparation for Sprint-2
    2. UI Preparation with Static Data & Testing
    3. Coding Entity Classes
    4. Data Loading with Dynamic Data & Testing
    5. **Coding Service Classes for Dynamic Data & Testing ( With & Without DI)**
    6. Data loading from Server & Testing
    7. Entity Auto generating & Testing
  - Version-2 (Server Pagination)**

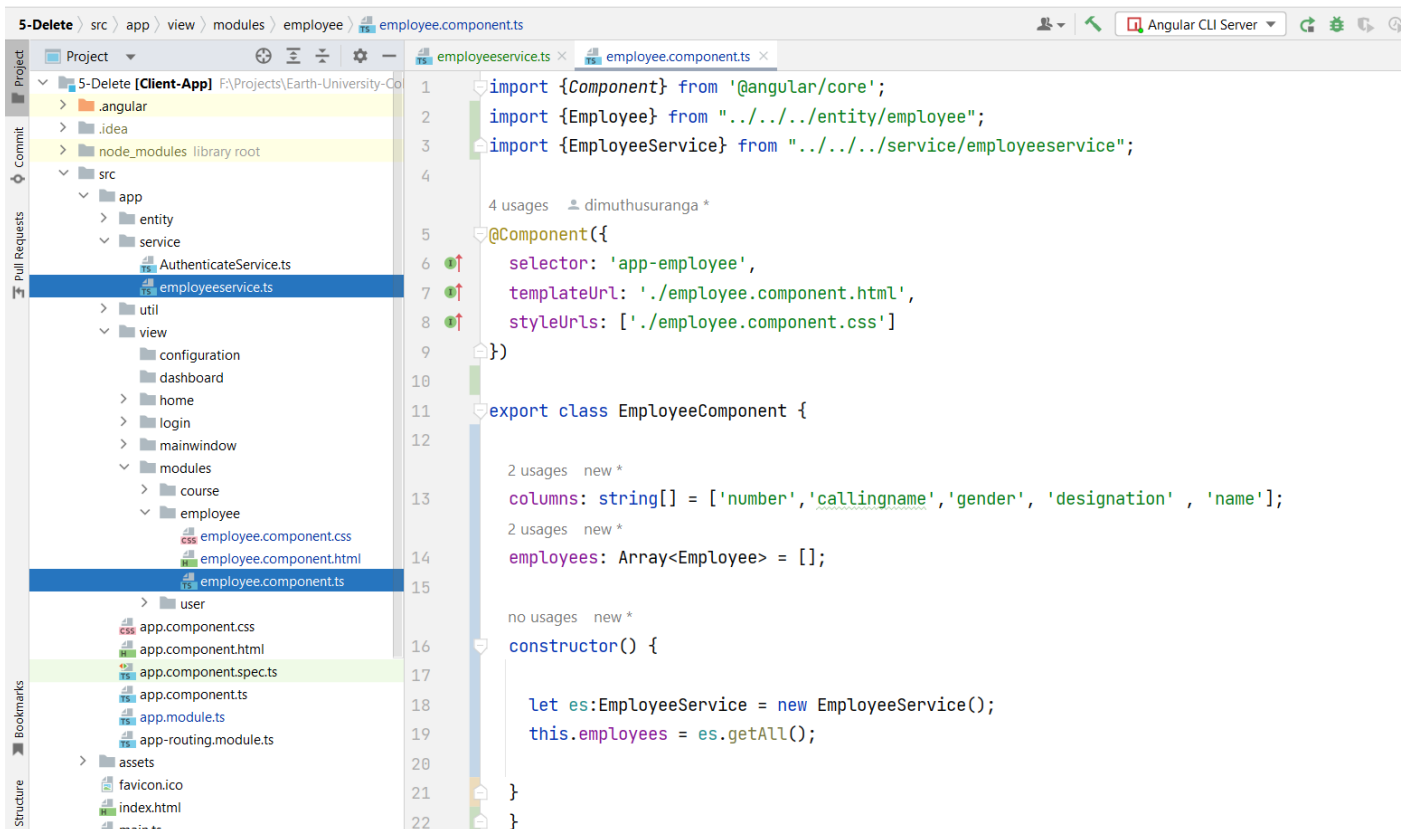
## 1. Create Service Class in the Service Package



## 2. Copy the Codes of the Object Array into the Service Class(employeeservice.ts) from the Controller Class(employee-component.ts) ← Change the Staff number “-M” for testing

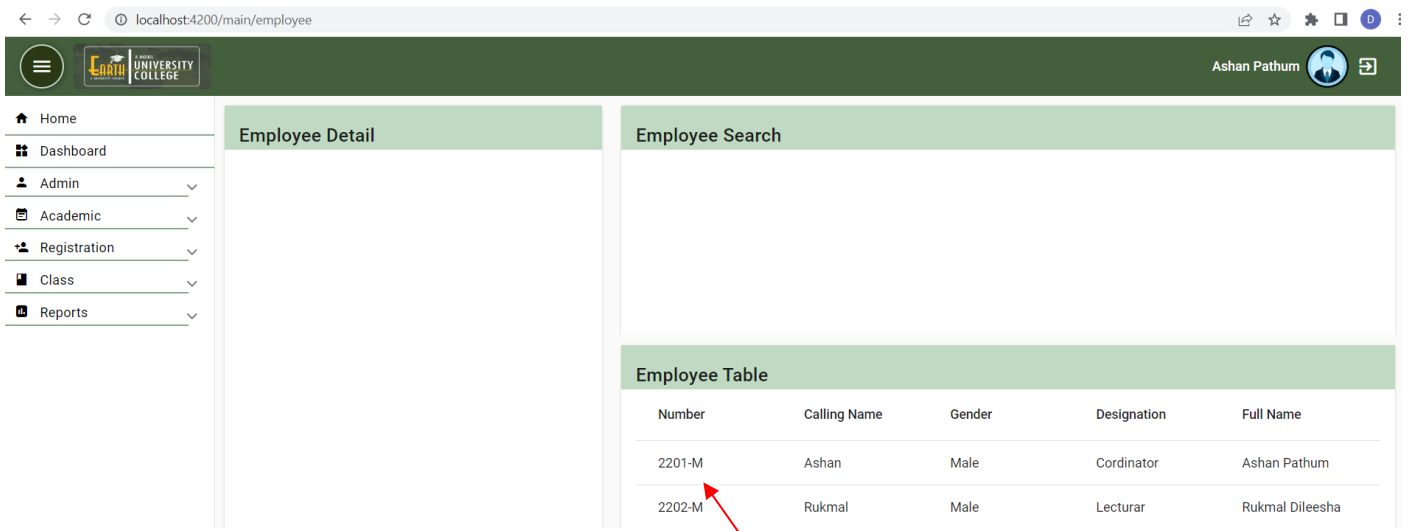


### 3. Change the Controller Codes to use the “getAll()” function of the Service Class using manual construction



```
1 import {Component} from '@angular/core';
2 import {Employee} from "../../entity/employee";
3 import {EmployeeService} from "../../service/employeeservice";
4
5 @Component({
6   selector: 'app-employee',
7   templateUrl: './employee.component.html',
8   styleUrls: ['./employee.component.css']
9 })
10
11 export class EmployeeComponent {
12
13   columns: string[] = ['number', 'callingname', 'gender', 'designation', 'name'];
14   employees: Array<Employee> = [];
15
16   constructor() {
17
18     let es:EmployeeService = new EmployeeService();
19     this.employees = es.getAll();
20   }
21 }
22
```

### 4. Observer the browser output



### 5. Change the Service class into a Injectable Type for auto construction with platform configuration at runtime by the Angular Framework

What is meant by Injectable Types?

Discuss the important of Injectable Type, Especially for Service Classes

Coding changes needed to make the Service Class Injectable

Coding changes needed to consume an Injectable Object in UI-Controller Class

Change the Staff Number with “-A” for testing

```

1 import {Employee} from "../entity/employee";
2 import {Gender} from "../entity/gender";
3 import {Designation} from "../entity/designation";
4 import {Injectable} from "@angular/core";
5
6 @Injectable({
7   providedIn: 'root'
8 })
9 export class EmployeeService {
10   constructor() { }
11
12   getAll(): Employee[] {
13     const employees: Array<Employee> = [];
14
15     let g1:Gender = new Gender( id: 1, name: "Male");
16     let d1:Designation = new Designation( id: 1, name: "Cordinator");
17     let d2:Designation = new Designation( id: 2, name: "Lecturan");
18
19     let e1:Employee = new Employee( id: 1, name: "Ashan Pathum", number: "2201-A", callingname: "Ashan", g1, d1);
20     let e2:Employee = new Employee( id: 1, name: "Rukmal Dileesha", number: "2202-A", callingname: "Rukmal", g1, d2);
21
22     employees.push(e1);
23     employees.push(e2);
24
25     return employees;
26   }
27 }

```

6. Change Controller Codes that is capable to use Injectable Object (Employee Service) constructed and injected by the Angular Framework (Actually the Router-Module when routing)

```

1 import {Component} from '@angular/core';
2 import {Employee} from "../../entity/employee";
3 import {EmployeeService} from "../../service/employeeservice";
4
5 @Component({
6   selector: 'app-employee',
7   templateUrl: './employee.component.html',
8   styleUrls: ['./employee.component.css']
9 })
10 export class EmployeeComponent {
11
12   columns: string[] = ['number', 'callingname', 'gender', 'designation', 'name'];
13   employees: Array<Employee> = [];
14
15   constructor(private es:EmployeeService) {
16     this.employees = es.getAll();
17   }
18 }

```

Employee UI is built and included into the App by the Routing Module.  
 HTML, CSS will be combined and Modified according to the EmployeeController(EmployeeComponnet.class).  
 So, Routing Module will create the object of EmployeeService and passed into the EmployeController when its constructor called. To achieve this auto construction and auto injection (dependency injection), the service class must be type @Injectable.

## 7. Observe the output with prefix “-A”

The screenshot shows a web application interface for Earth University College. The browser address bar displays 'localhost:4200/main/employee'. The interface has a dark green header with a logo on the left and the user 'Ashan Pathum' on the right. A sidebar on the left contains a menu with items: Home, Dashboard, Admin, Academic, Registration, Class, and Reports. The main content area is divided into two panels. The left panel is titled 'Employee Detail' and is currently empty. The right panel is titled 'Employee Search' and contains an 'Employee Table' with the following data:

Number	Calling Name	Gender	Designation	Full Name
2201-A	Ashan	Male	Cordinator	Ashan Pathum
2202-A	Rukmal	Male	Lecturar	Rukmal Dileesha

A red arrow points to the '2202-A' entry in the 'Number' column of the table.

Department of Information Technology  
Earth University College