**Project Documentation**:

**Introduction**:

This application provides a monthly overview of the travel allowance for the employees of a company for the current year.

**Requirements**:

* Travel allowance is based on the mode of transportation, distance traveled, and current compensation rate.
* The compensation rate is obtained from an API endpoint.
* Compensation/Allowance is paid on each first Monday of next month.

**Technologies**:

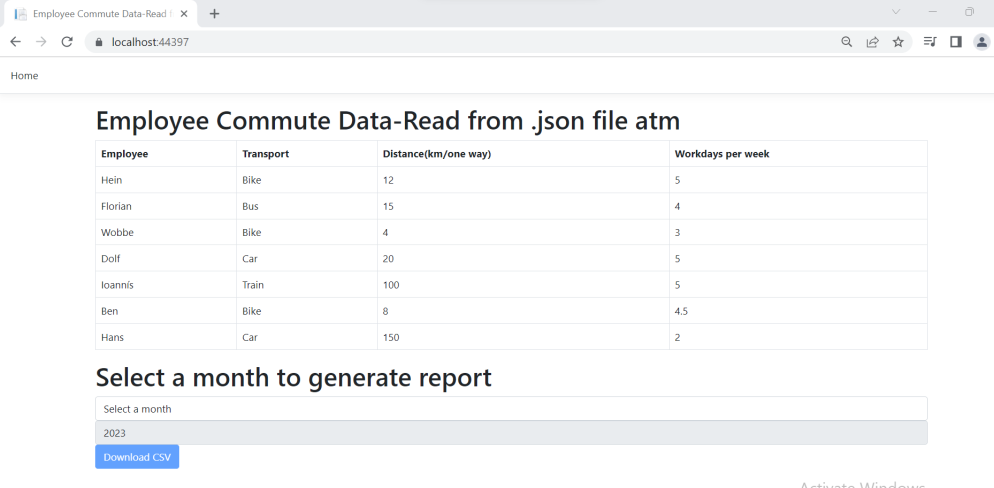
* ASP.NET Core 6.0, C# , LINQ, HttpClient for connection with external API(<https://api.staging.yeshugo.com/applicant/travel_types>.)
* Design Pattern : Dependency injection

**Architecture**:

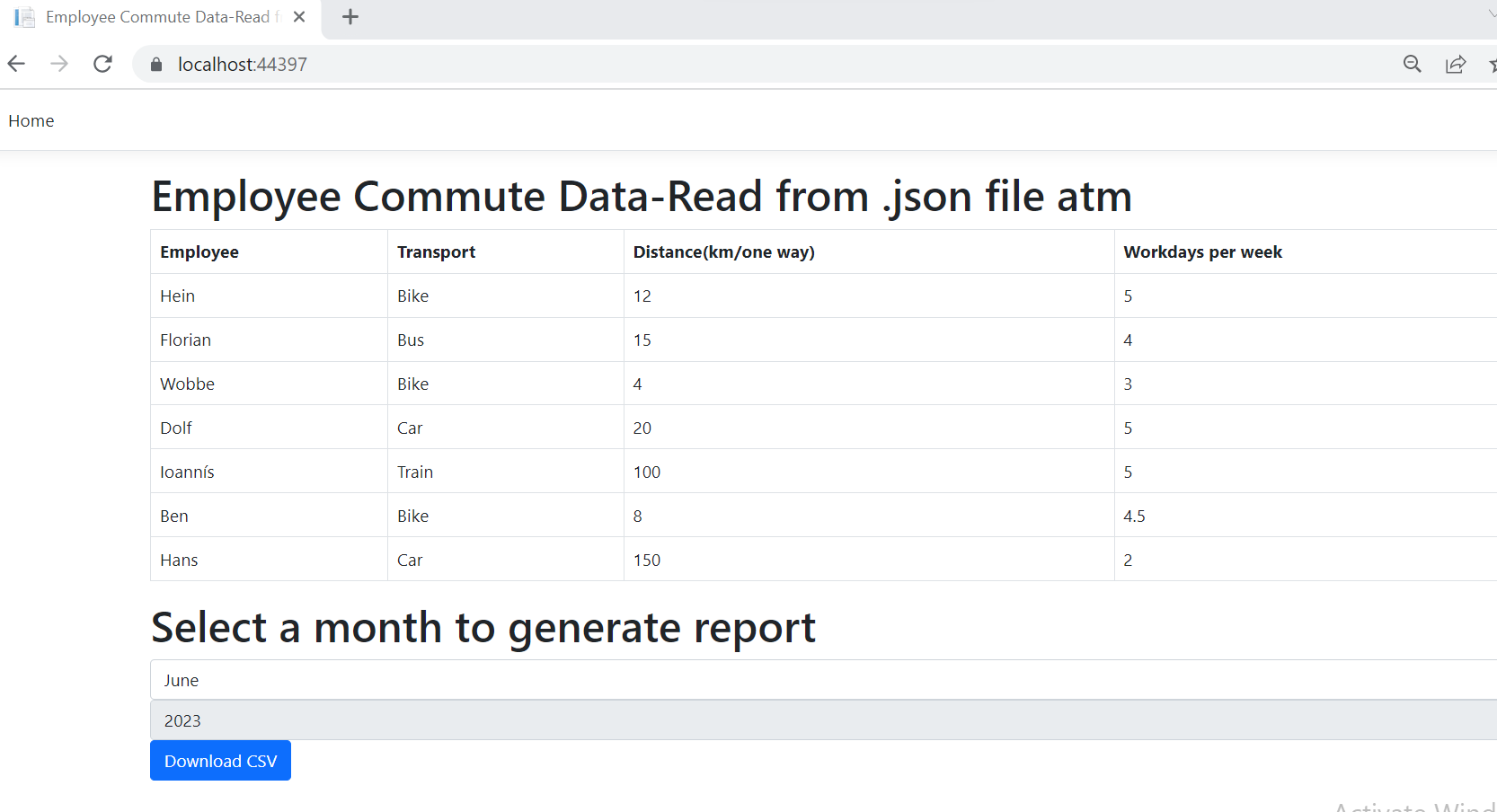
* The application consists of a front-end MVC framework and a back-end API.
* The front-end takes input and output, while the back-end handles business logic and data manipulation.
* A separate service class is written to handle all the business logic. I have tried to make it as headless possible

**User Interface**:

* It shows the user commute and work week information. This information is read from .json file.



* It provides option to choose month. The Download button is enabled when month is selected.



* Clicking on the download button downloads the compensation report in travel\_allowance\_MM\_YEAR.CSV format to your machine.

**Business Logic and Assumption**:

* As per requirement, employees work days per week is different. With the given information, we do not know on which days of a week an employee actually work/will work (if he works for less than 5 days). Hence, following is the example of my logic to calculate compensation for a given month.

There are **52** weeks in a year. So if an employee works **3** days a week and he travels **4**km/one way, then

Total distance he travelled for a year = (**4**\*2) \* **3** \* **52**=1248

Distance covered in a month = 1248/12 = 104.

Monthly compensation = 104 \* Compensation rate per km (0.33) = 34.32.

* Compensation rate per km of travel for different modes of commute is fetched from the given API
* Payment Date is calculated such that it is always the first Monday of the subsequent month.
* For instance, if allowance report is created for month February 2023, then payment date will be 6th March 2023

**API Integration**:

Code snippets to illustrate the implementation of the API integration.

// Get the compensation rates for different travel types from the API

var response = await \_httpClient.GetAsync("applicant/travel\_types");

var content = await response.Content.ReadAsStringAsync();

IEnumerable<CompensationRate>? employeetravelModes = JsonConvert.DeserializeObject<IEnumerable<CompensationRate>>(content);

**Manual Execution**:

As I did not deploy the application anywhere you can always open this via Visual Studio 2022 and run directly to test the feature.

**Conclusion**:

Future enhancements that could be added to the application.

* Employee punched-in information should also be considered so that exact compensation for the month can be calculated.
* Employee data can be retrieved from other ways also like database, webapis, user inputs.