Technical Specification

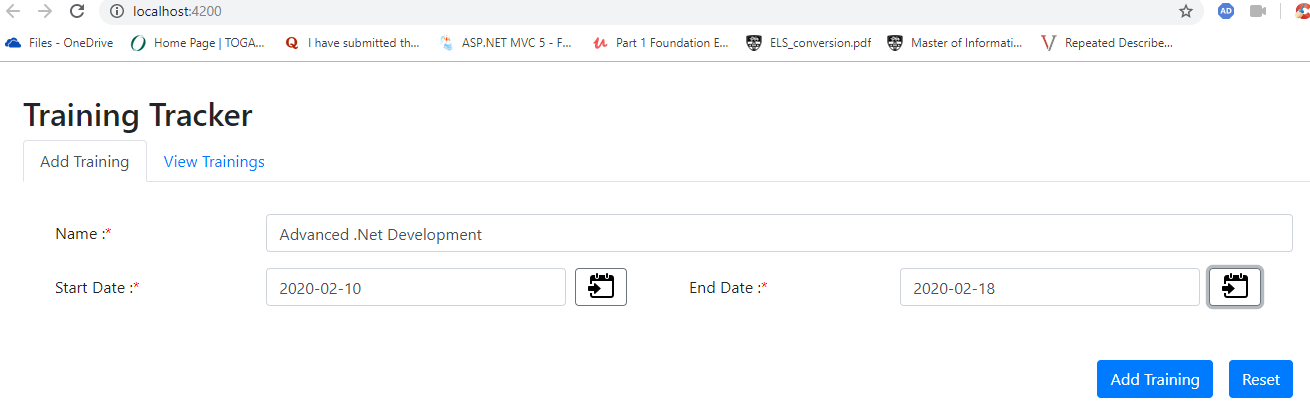
# Requirement Realisation

## Build a .NET/C# REST API application with Angular 2+ and Bootstrap front-end framework.

Application contains 2 parts. The front end (UI) created using Angular 8, bootstrap. The backend Api is created in .Net Core.

## The application should provide a UI form with input fields as training name, start date, end date, and a Confirm button to submit the form.

Application has pages for adding training and another page for viewing the trainings added to the database. Basic client-side validations are provided for the controls in the page.

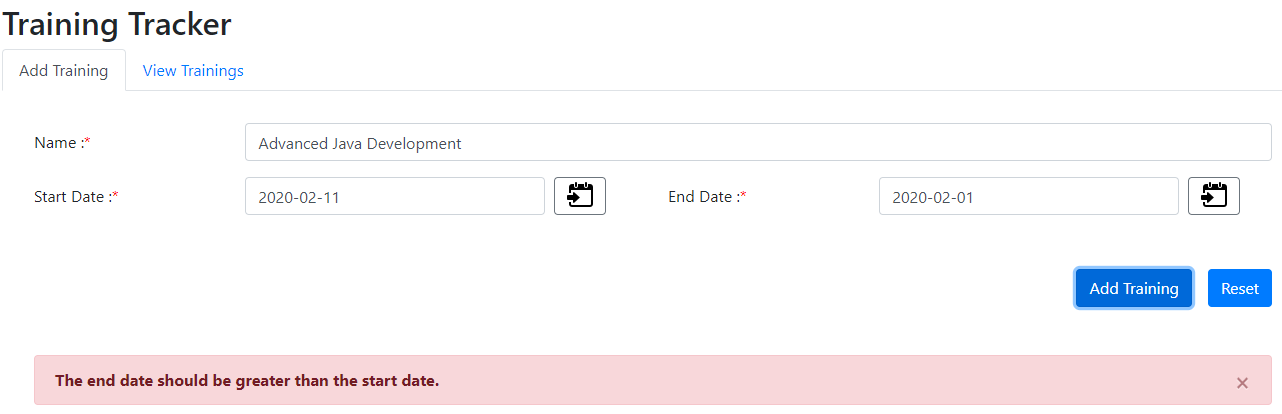


The list page gives the overall trainings. In addition, the list is sortable and pagination is enabled.

## 

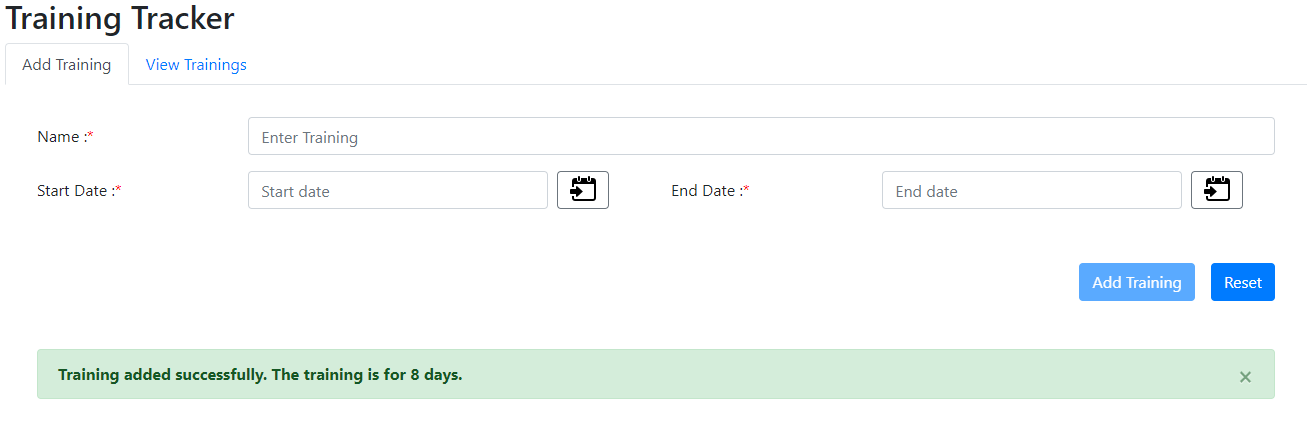
## If the end date is before the start date, the application should display an error message and the message text should be defined externally.

The error message is defined in the server end in “Appsettings.Json”. The client passes an error code to server to get the right message and displays in the UI.



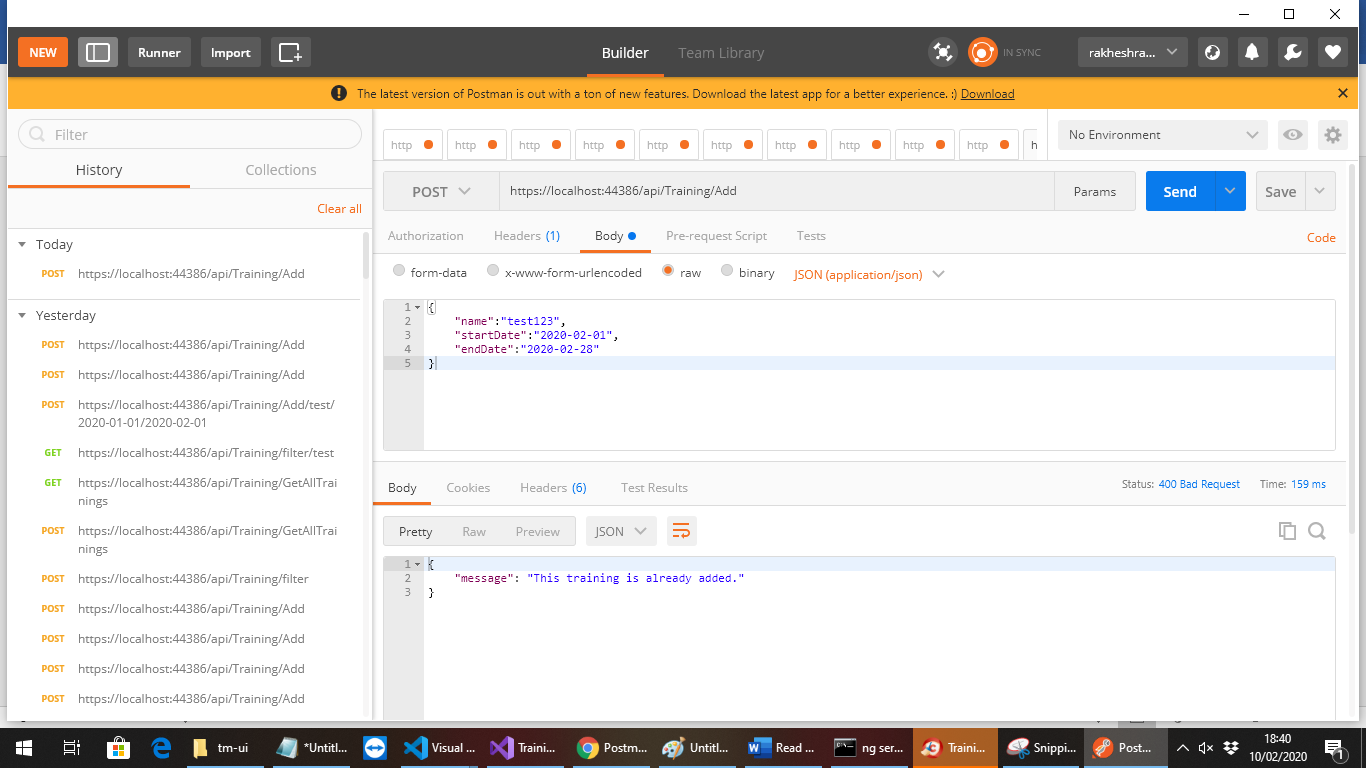
## On Confirmation of saving the training name, start date and end date the rest api should save details to DB and respond with Success message. Along with message, application should display number of days between the two date inputs.

The data is saved in the DB and the server responds with the success message with the difference in dates.



## Unit/Integration test cases are mandatory.

Few test cases are written in “add-training.component.spec.ts”. Api’s are tested using Postman.



# Notes

The “src\app\app.config.ts” file has an entry “apiBaseUrl: 'https://localhost:44386/api/Training'”. Please change it to point it to the API’s hosted.

# Backend Specification

The backend application has 4 API methods which are follows.

## Add

Accepts “Training” class as the input, validates the input and adds to the DB. Returns a response with an appropriate status code and message.

## Filter

Accepts a string as the input, validates it searches in the DB and returns a response with an appropriate status code and message.

## GetAllTrainings

Returns all trainings from the database with appropriate status codes and message.

## GetErrorMessage

Returns error message from appsettings.json based on the “error code” with appropriate status codes.

The Api’s are connected to the Database through entity framework.