

# Weather Forecast Application using [VB.NET](#) MVC

## Problem Statement:

Develop a weather forecasting web application using VB.NET and ASP.NET MVC. The application should allow users to upload a CSV file containing a list of geographic locations, including latitude, longitude, and location name.

Upon upload, the application should parse the CSV file, process each location, and fetch the weather forecast for the upcoming days using an external weather service API.

After processing, the application must display the list of uploaded locations along with their corresponding weather forecasts in a user-friendly interface.

## Flowchart:

[https://miro.com/app/board/uXjVJc3VFus=](https://miro.com/app/board/uXjVJc3VFus=/)

## Technology Stack

**Backend:** VB.NET, ASP.NET MVC, LINQ

**Frontend:** HTML5, CSS3, JavaScript, jQuery, Bootstrap, Chart.js

**External API:** Open-Meteo

**Storage:** Local SQL Database (used as a 6-hour cache)

## Features:

### Upload CSV with location data

Users can upload a CSV file containing a list of locations. Each row should include Latitude, Longitude, and Location Name.

### Parse and validate CSV

After upload, the application reads the CSV and checks for correctness ensuring all required fields are present and properly formatted. If the file is invalid (missing columns or values), the user is notified with a clear error message.

### Fetch weather forecast using external API

For each valid location in the CSV, the app sends a request to the Open-Meteo API to retrieve a 7-day weather forecast. The response includes daily minimum and maximum temperatures.

**Display results in a tabular view (showing minimum and maximum temperature for seven days)**

The weather data for all locations is presented in a table format, allowing users to compare and view forecasted temperatures at a glance.

**Chart view with dropdown to select locations**

A dropdown is populated with all the uploaded locations. Users can select a location to view its forecast visually.

**Line chart showing min and max temperatures**

When a user selects a location from the dropdown, a line chart appears displaying the minimum and maximum temperatures for that location over the next 7 days.

**Re-upload CSV from the same page**

Users don't need to navigate to a new page to upload a different CSV. The upload feature is available on the same page for convenience and better user experience.

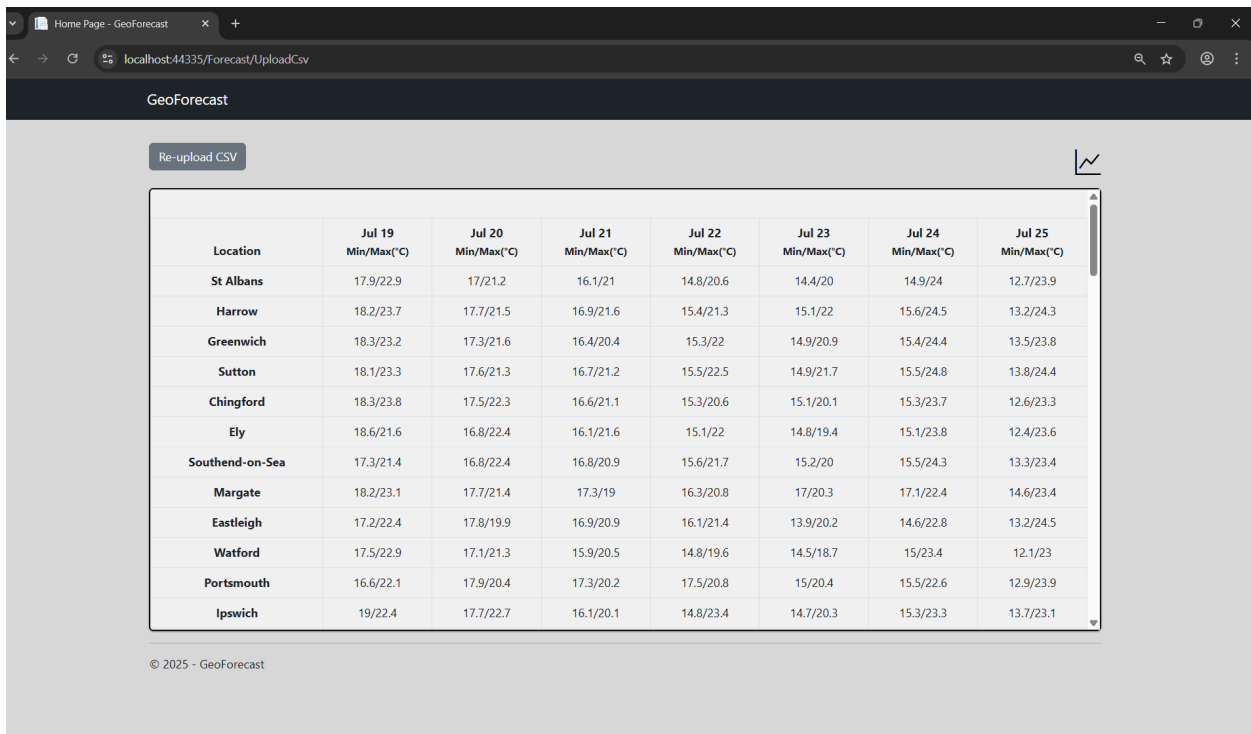
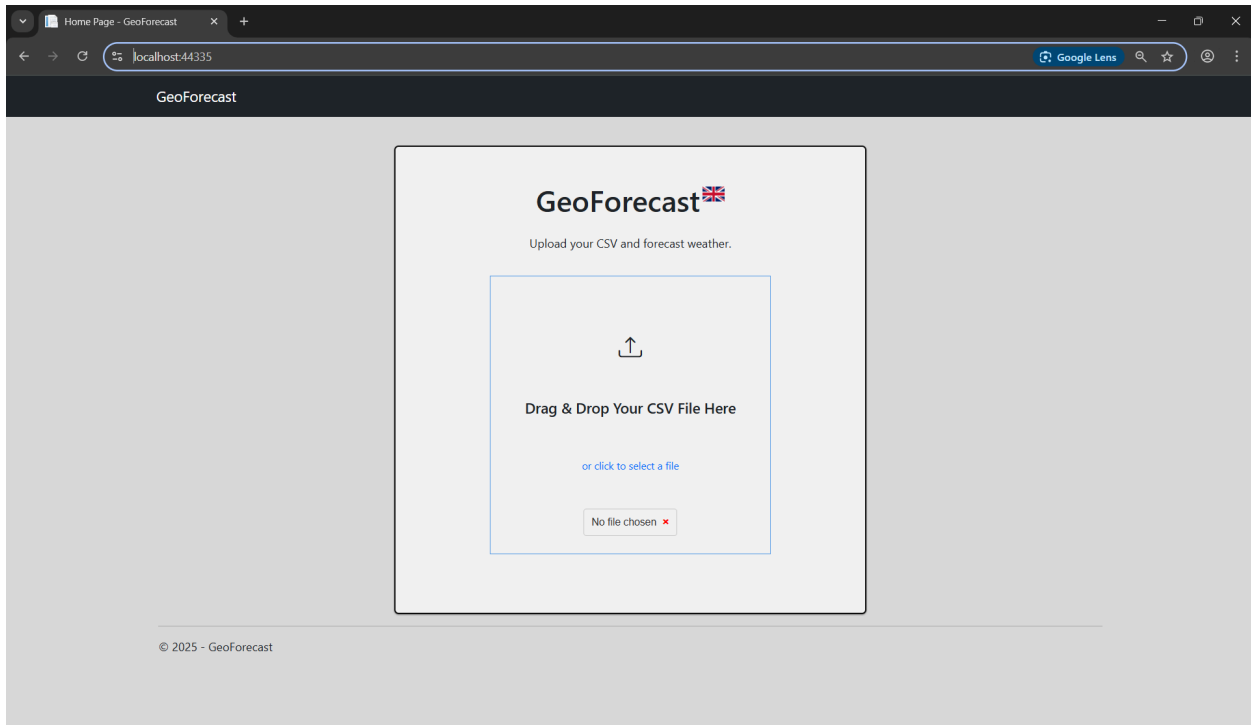
**Caching via local DB to reduce API calls**

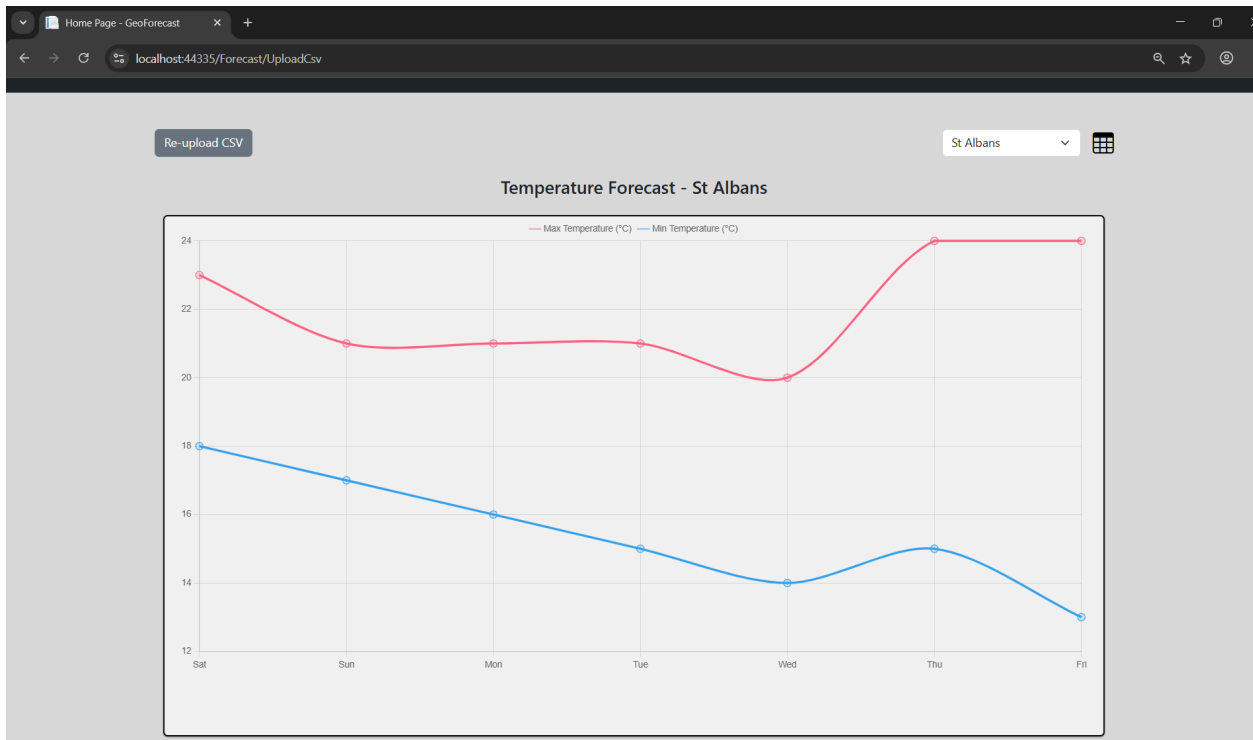
To avoid unnecessary external API calls, the app stores the forecast data in a local database. If data for a location is already fetched in the past 6 hours, the app uses that cached data instead of calling the API again, improving speed and reducing latency.

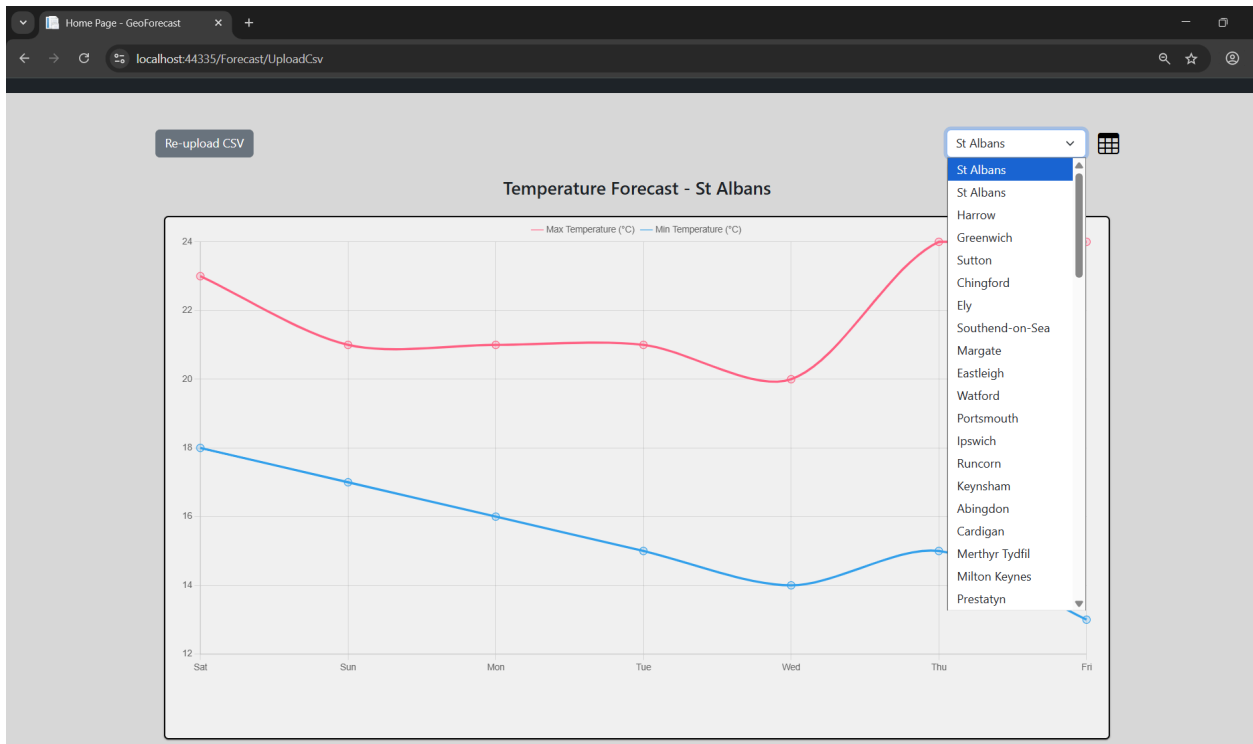
**Error handling**

The application includes proper error handling whether it's a malformed CSV, API failure, or unexpected system issue and provides clear feedback to the user.

# Screenshots:







Home Page - GeoForecast

localhost:44335/Forecast/UploadCsv

GeoForecast

**Error**

The remote server returned an error: (400) Bad Request. Please re-upload the CSV with proper values.

## GeoForecast

Upload your CSV and forecast weather.

**Drag & Drop Your CSV File Here**

[or click to select a file](#)

No file chosen

