4.WebAPI Handson

**1. Objective:**

Demonstrate how to update employee data using the HTTP PUT method in ASP.NET Core Web API. The update should validate the employee ID and return proper responses. The data is extracted from the request body using [FromBody], and Swagger is used to test the endpoint.

**2. Tools Used:**

* Visual Studio 2022
* .NET 6/7 Core Web API
* Swagger (for API testing)
* C# Programming Language

**3. Employee Model Code (Models/Employee.cs):**

namespace EmployeeApi.Models

{

public class Employee

{

public int Id { get; set; }

public string? Name { get; set; }

public string? Department { get; set; }

public double Salary { get; set; }

}

}

**4. Controller Code (Controllers/EmployeeController.cs):**

using Microsoft.AspNetCore.Mvc;

using EmployeeApi.Models;

namespace EmployeeApi.Controllers

{

[ApiController]

[Route("api/[controller]")]

public class EmployeeController : ControllerBase

{

private static List<Employee> employees = new List<Employee>

{

new Employee { Id = 1, Name = "Alice", Department = "HR", Salary = 50000 },

new Employee { Id = 2, Name = "Bob", Department = "IT", Salary = 60000 },

new Employee { Id = 3, Name = "Charlie", Department = "Finance", Salary = 70000 }

};

[HttpPut("{id}")]

public ActionResult<Employee> UpdateEmployee(int id, [FromBody] Employee updatedEmp)

{

if (id <= 0)

{

return BadRequest("Invalid employee id");

}

var existingEmp = employees.FirstOrDefault(e => e.Id == id);

if (existingEmp == null)

{

return BadRequest("Invalid employee id");

}

existingEmp.Name = updatedEmp.Name;

existingEmp.Department = updatedEmp.Department;

existingEmp.Salary = updatedEmp.Salary;

return Ok(existingEmp);

}

}

}

**5. API Testing via Swagger**

Screenshot the Swagger UI after:

* Opening PUT /api/Employee/{id}
* Clicking **"Try it out"**
* Passing id = 2 and a sample JSON request body like:

{

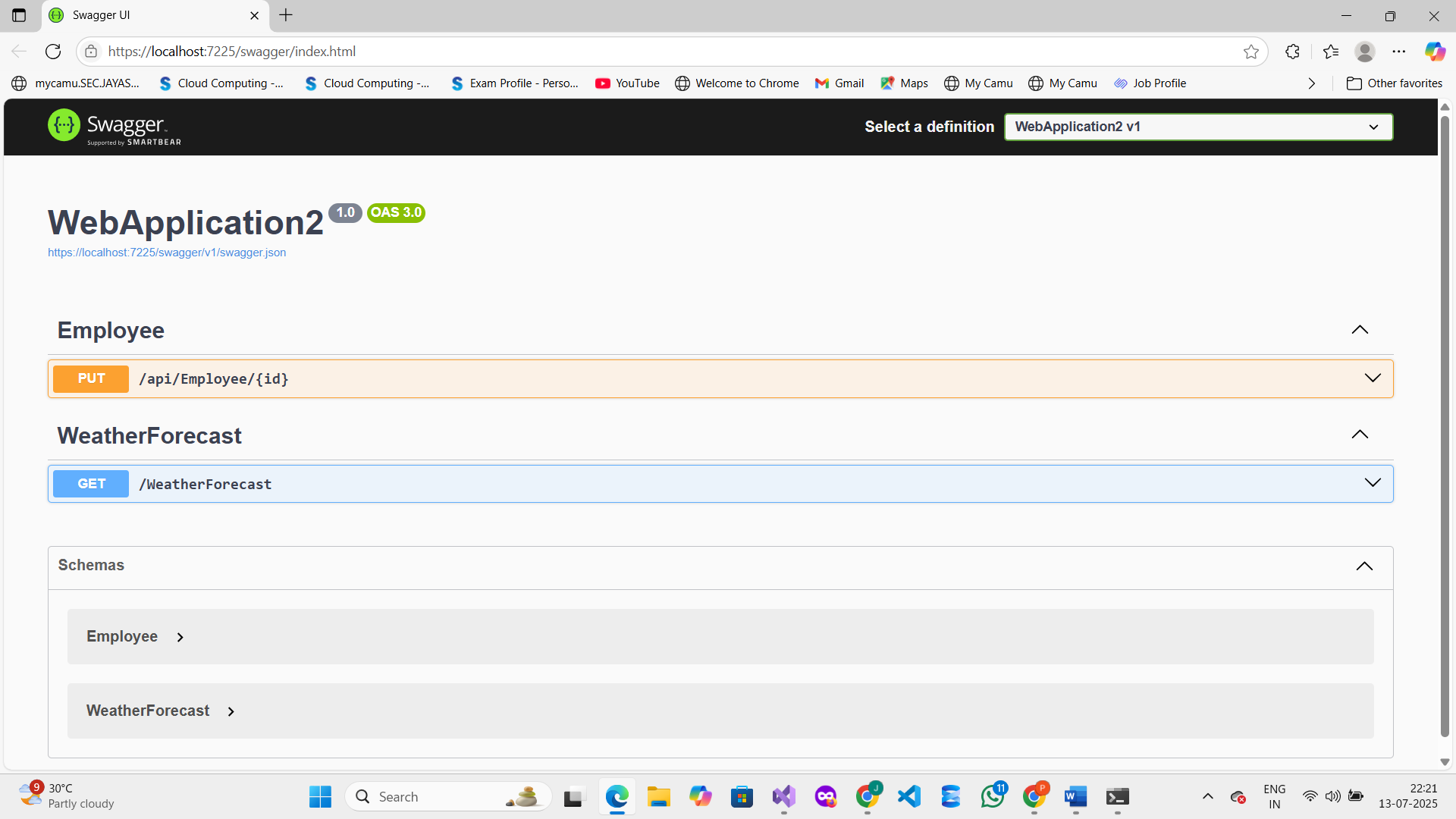
"id": 2,

"name": "Robert",

"department": "IT Support",

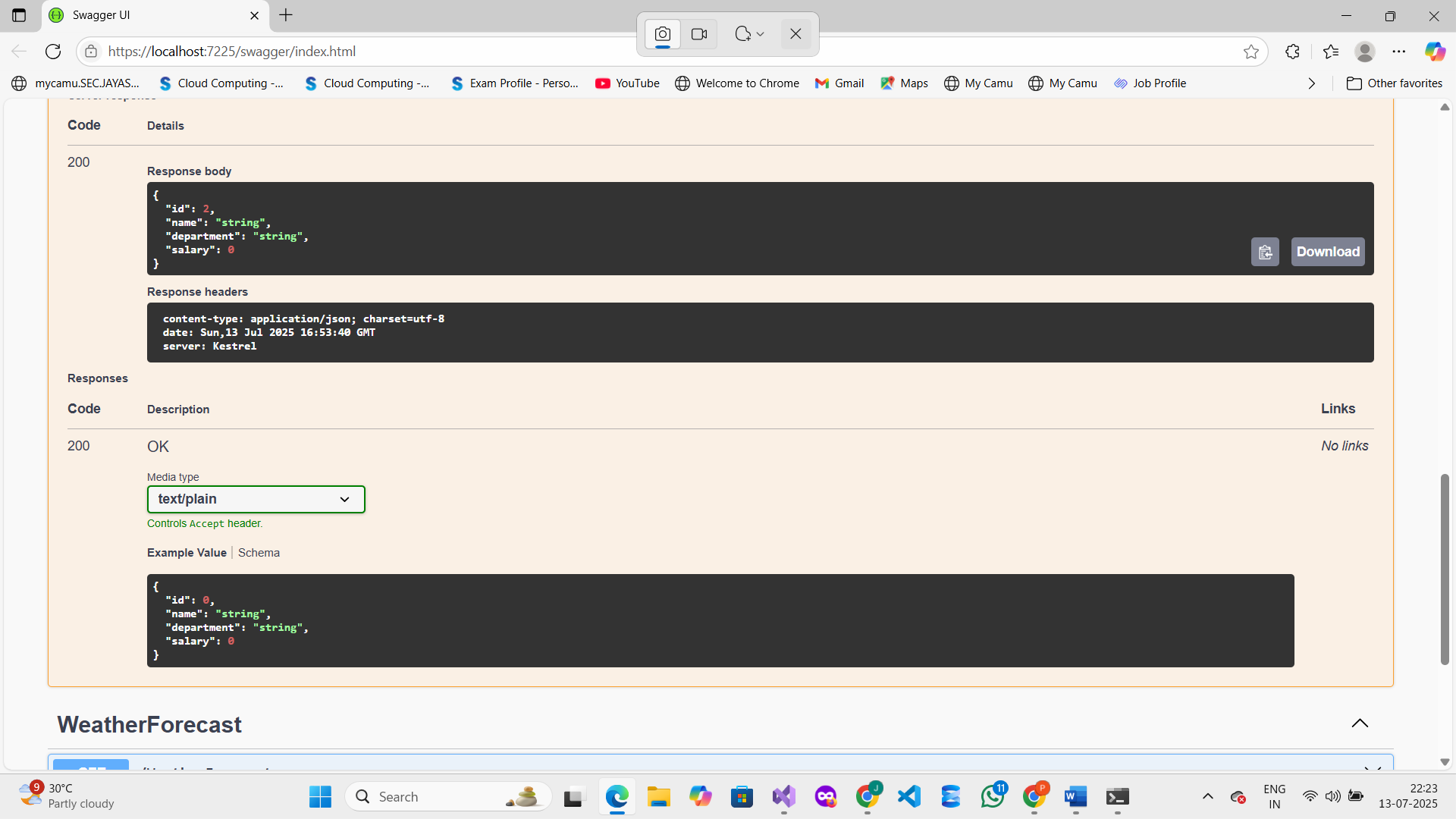
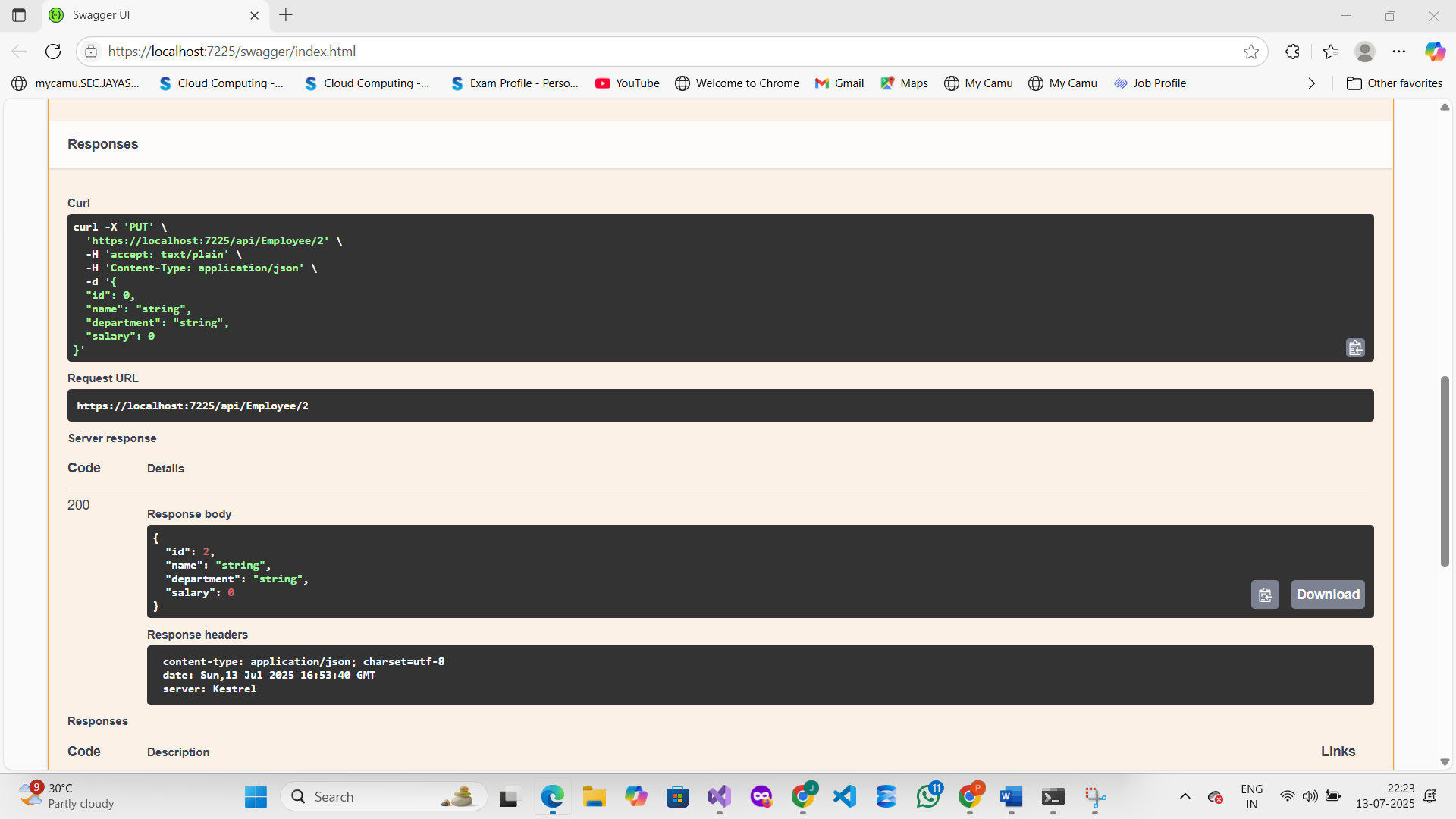
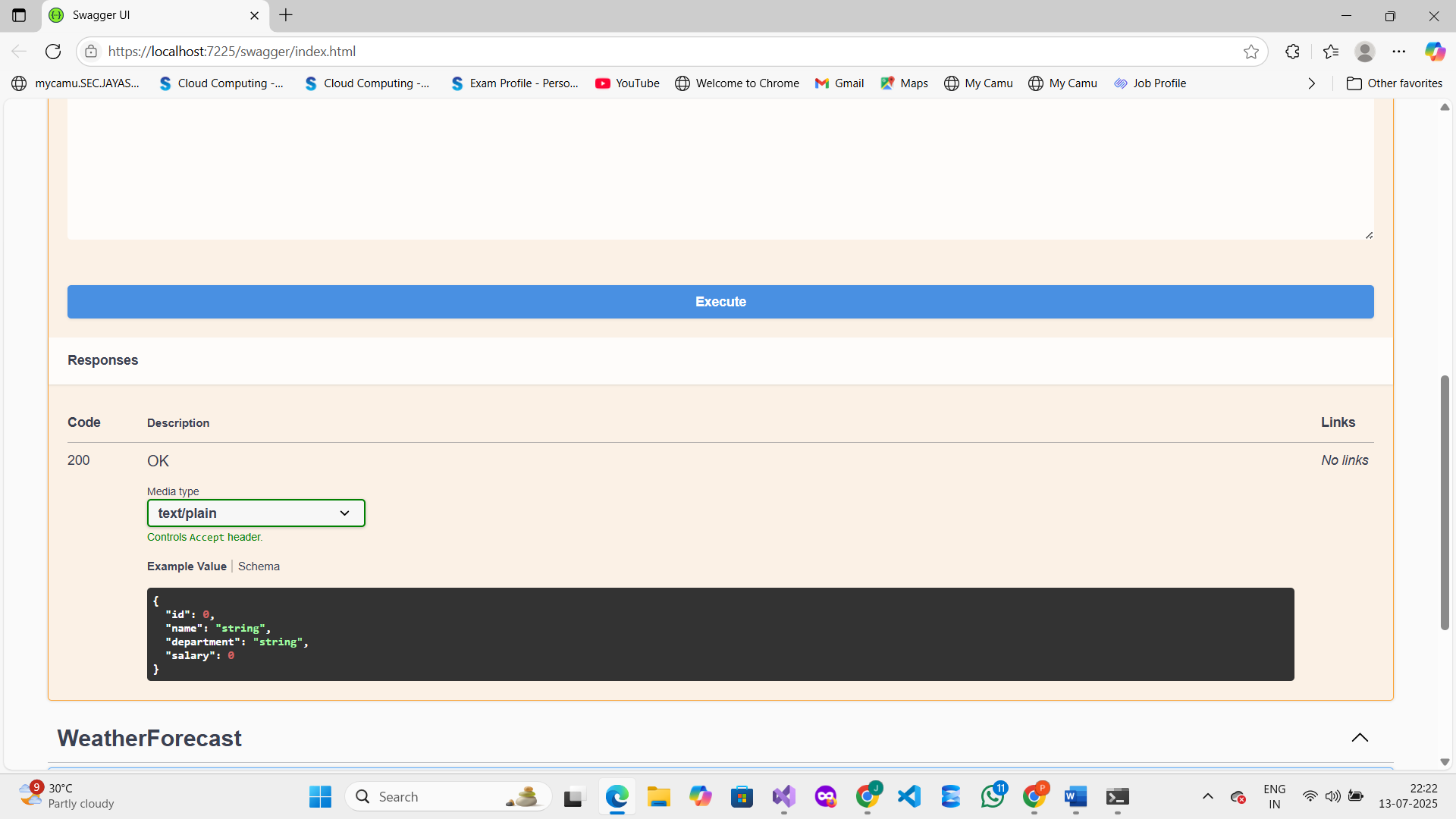
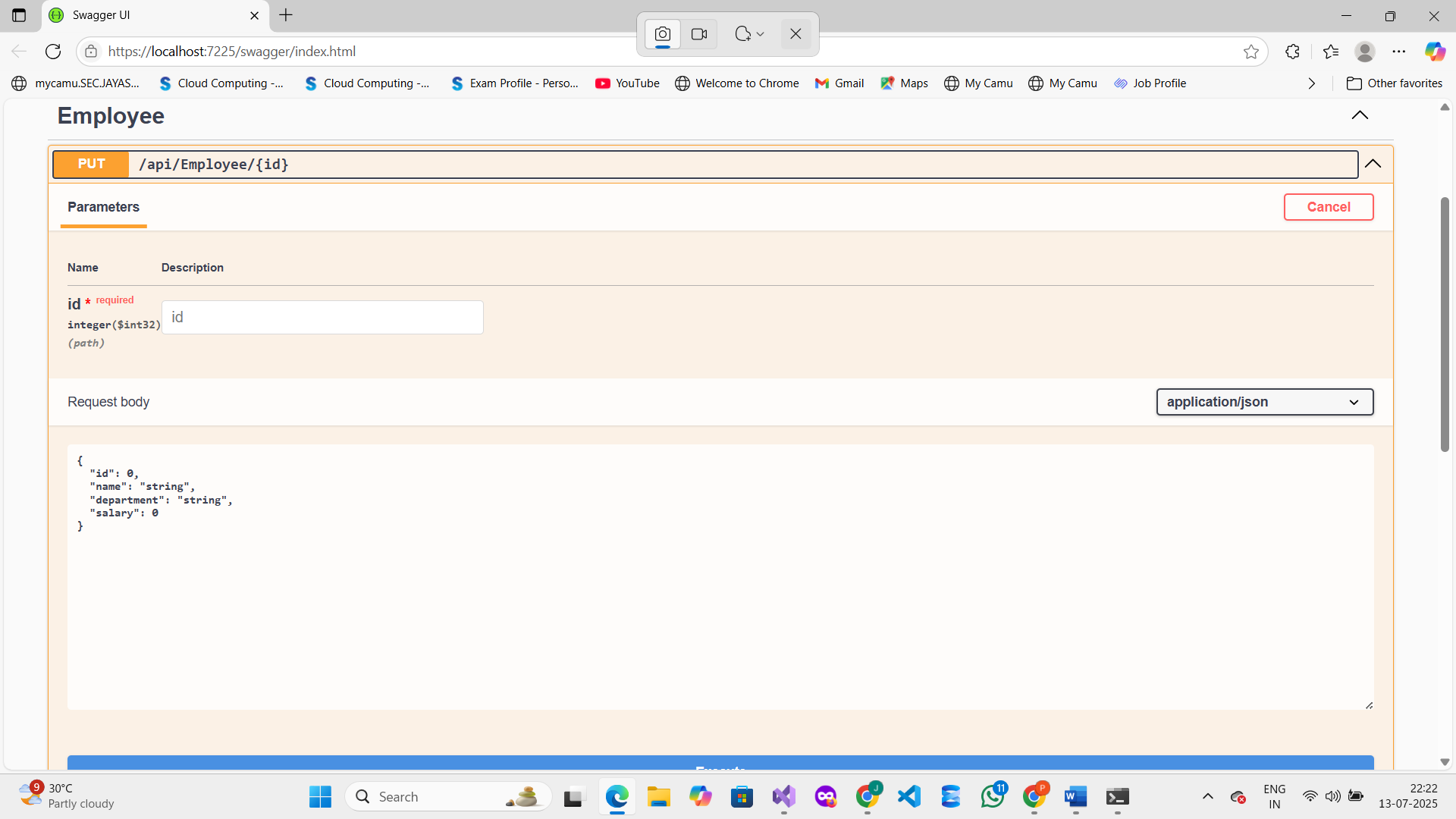
"salary": 65000

}



**🔹 6. Validation Scenarios Tested:**

| **Scenario** | **Result** |
| --- | --- |
| ID <= 0 | 400 Bad Request |
| ID not found in list | 400 Bad Request |
| Valid ID with JSON body | 200 OK |



**7. Conclusion:**

The PUT method was successfully implemented to update employee data in a Web API using [FromBody]. Proper validation was added to handle invalid IDs, and the updated employee record was returned. Swagger was used to test all possible scenarios.