Principles of Database Systems

Project Part #5

Usman Zia

OVERVIEW

The Technologies used:

- This web application uses AngularJS, a popular javascript-based frontend framework, for the frontend.
- This web application uses ExpressJS, a popular javascript-based backend framework, for the backend.
- This web application uses XAMPP as the Database Management System.

Running the Web application:

- The AngularJS frontend can be started using the "ng serve" command once you cd into the AngularJS directory and use the “npm install” command to download all the necessary dependencies to serve the frontend on http://localhost:4200/
- The ExpressJS backend can be started using the "node app.js" command once you cd into the ExpressJS and use the “npm install” command to download all the necessary dependencies to serve the backend on http://localhost:8080/
- The application uses XAMPP for the database, so for the application to run the XAMPP SQL server must be running on http://localhost:3306/

Is the error checking done by Application or by Database?

All error checking against bad inputs in this web application is done by the application to ensure only valid and accurate inputs are received by the database and that the user receives immediate error feedback.

Explain the trade-offs and give examples of both.

Pros and Cons of Application Error Checking

Application error checking is advantageous to Database error checking because it provides the user with immediate error feedback, and it is less costly in time and resources since no information is sent to the database just for the input to be rejected because it is invalid.

Application error checking is disadvantageous to Database error checking since it requires extra coding work for the programmer and the possibility of errors in that extra code. Everywhere in the application where you accept the input you will need the appropriate error-checking code, which leads to redundant and repetitive code.

Example of Application Error Checking: The ssn input given by the user is checked using javascript logic to ensure that the input string is all numerical values and has a length of 9 before passing it to the database.

Pros and Cons of Database Error Checking

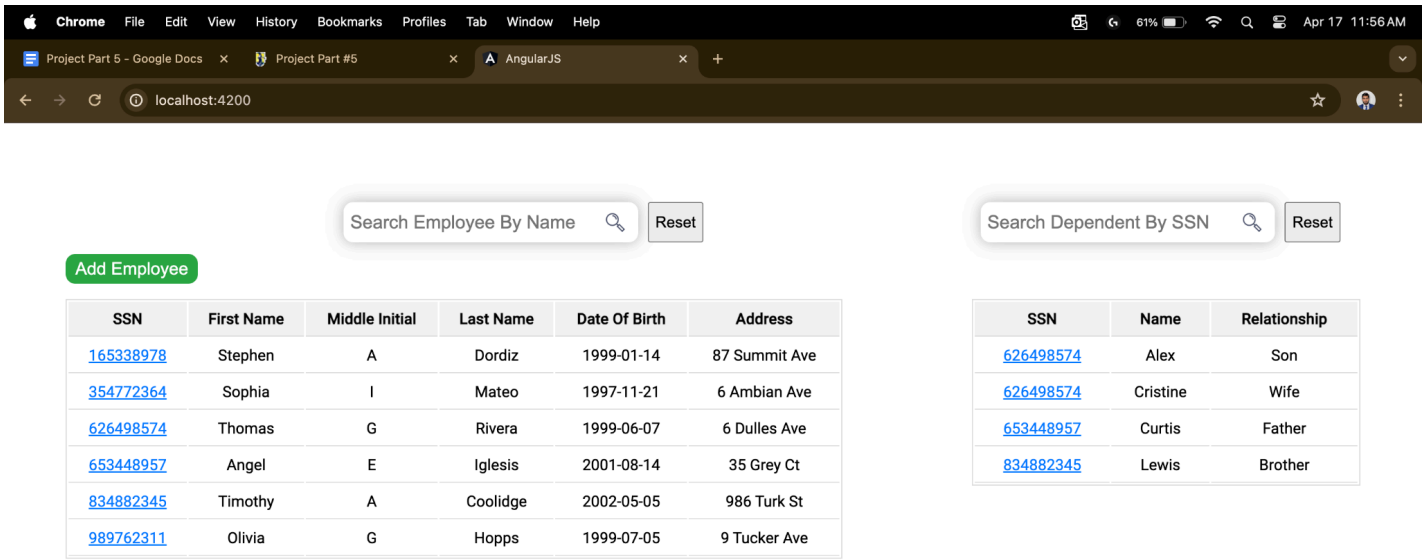
Database error checking is advantageous to Application error checking because you avoid repetitive error-checking code in the application and instead have the database error check the input.

Database error checking is disadvantageous to Application error checking because it is a slow resource-consuming process in which input, as given by the user, is sent directly to the database without a validity check by the application, and error feedback is not immediately visible to the user.

Example of Database Error Checking: The ssn input given by the user is passed to the database without any error checking by the application for the database to check the input and decide whether to accept the input or reject it with an error.

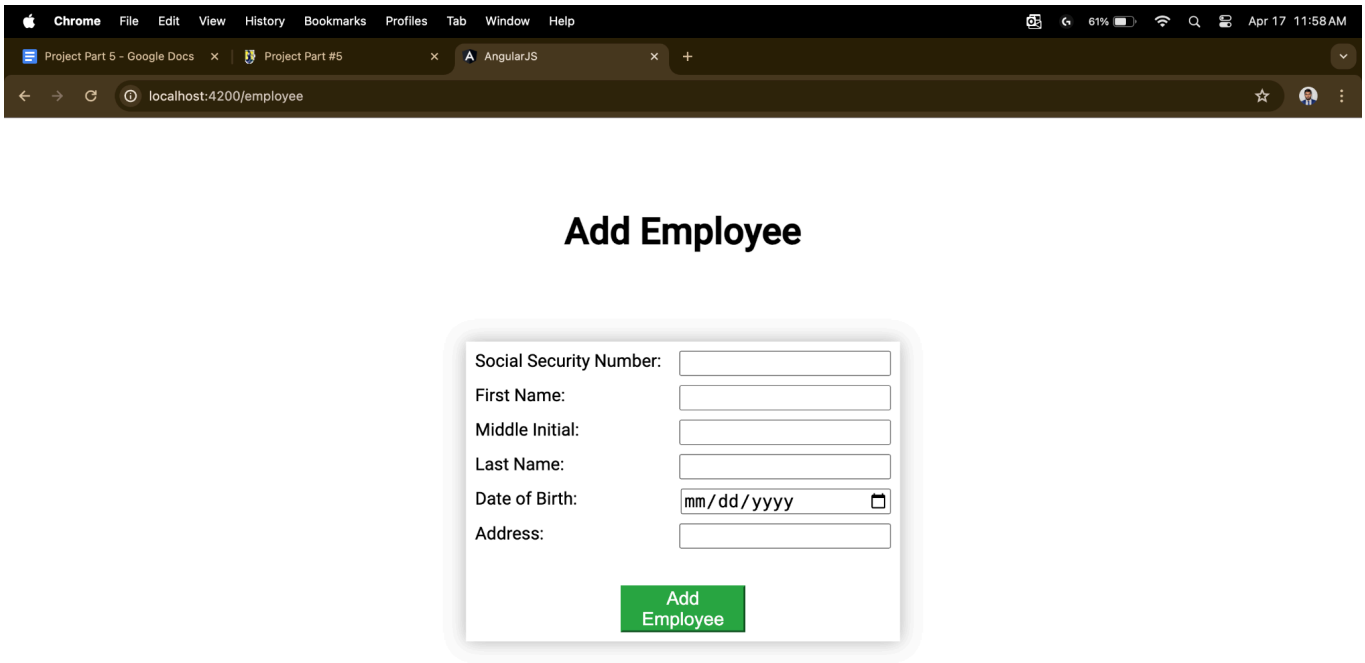
DEMO

This is the starting page of the Web application, this page shows all information about Employees and Dependents

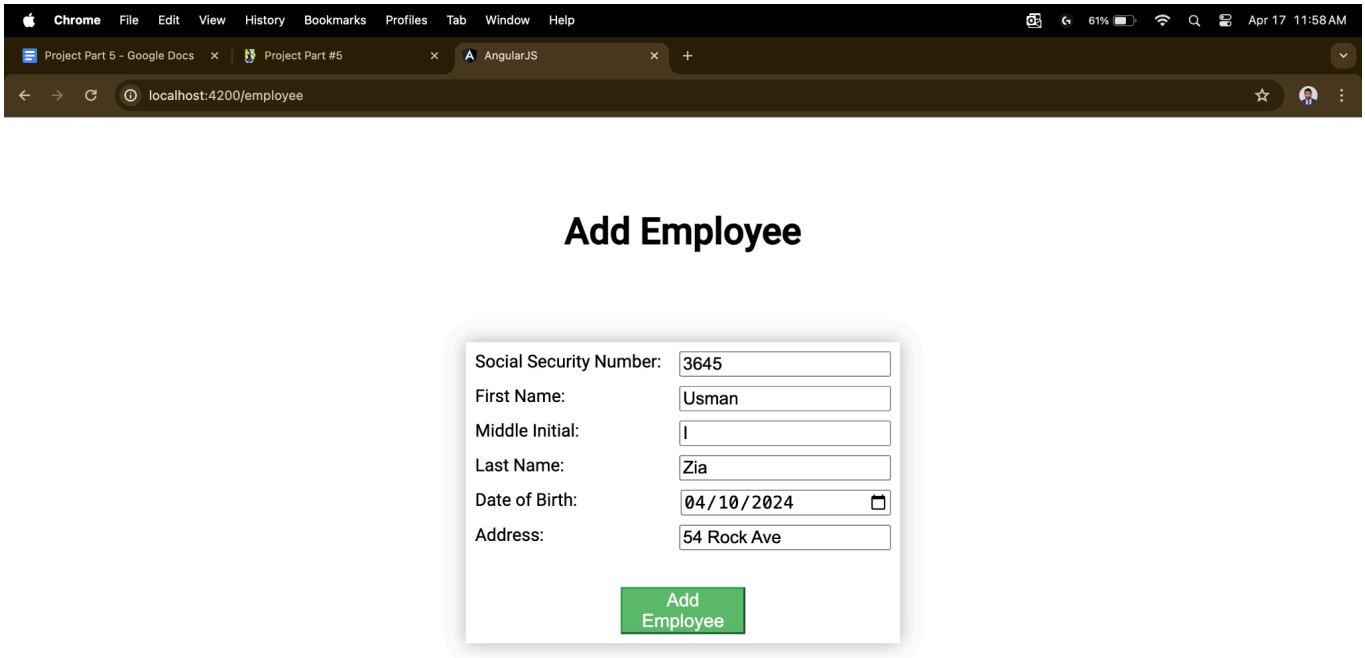


Demonstrating the Inserting functionality

Let’s click the “Add Employee” button to add a new employee

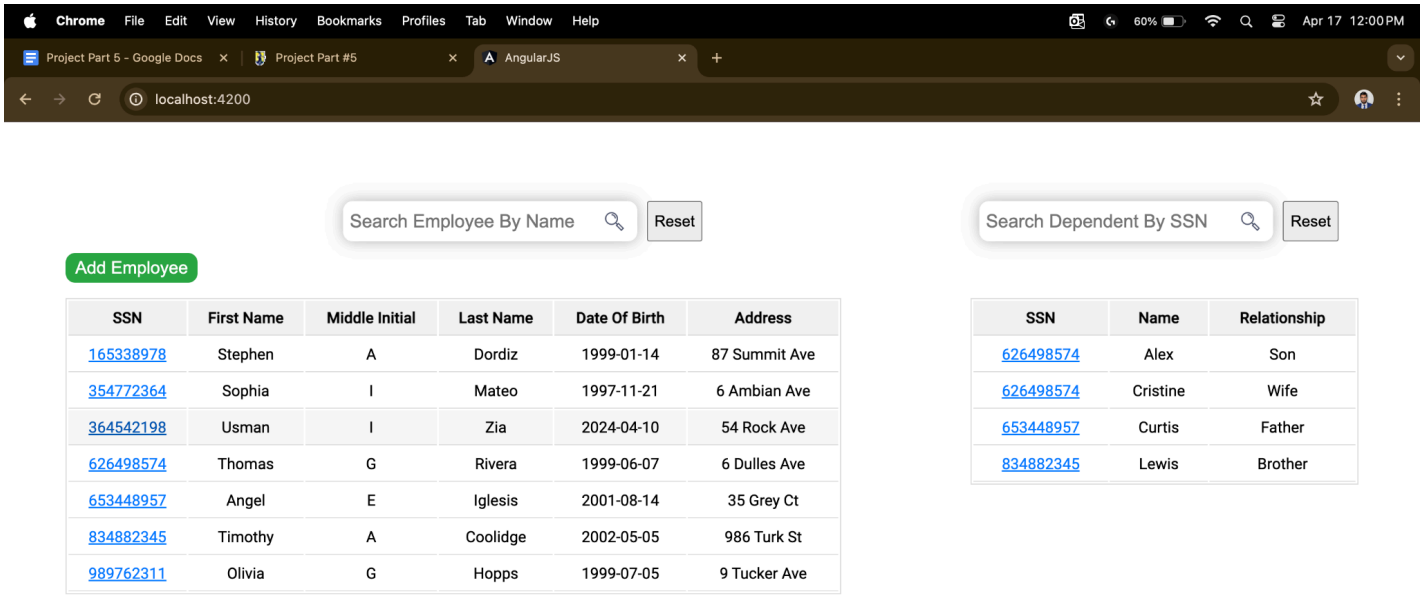


On this page, we will input all information about the new employee



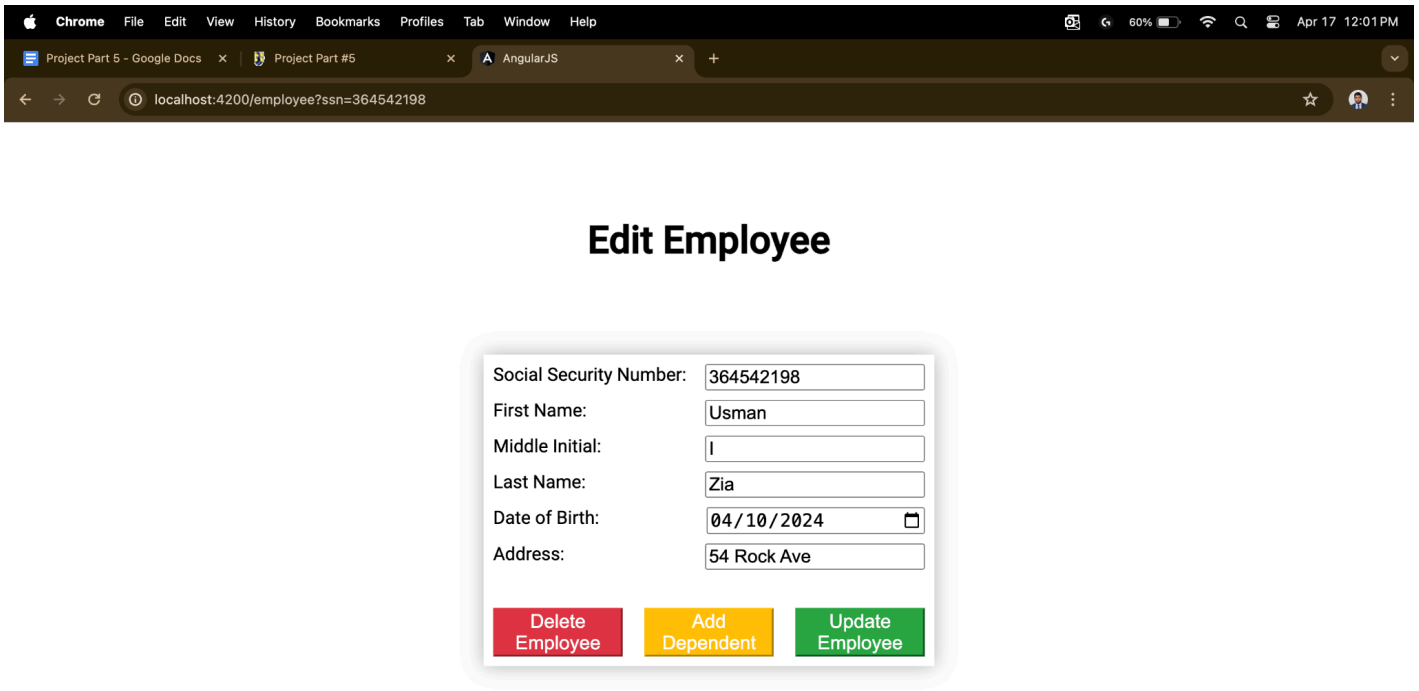
As you can see, the application does thorough error checking on the user input to ensure that all input requirements are met and that the input is valid. Until the input requirements are met, no employee can be added. So let's provide the full ssn number and click "Add Employee"

As you can see, our new employee has now been successfully added

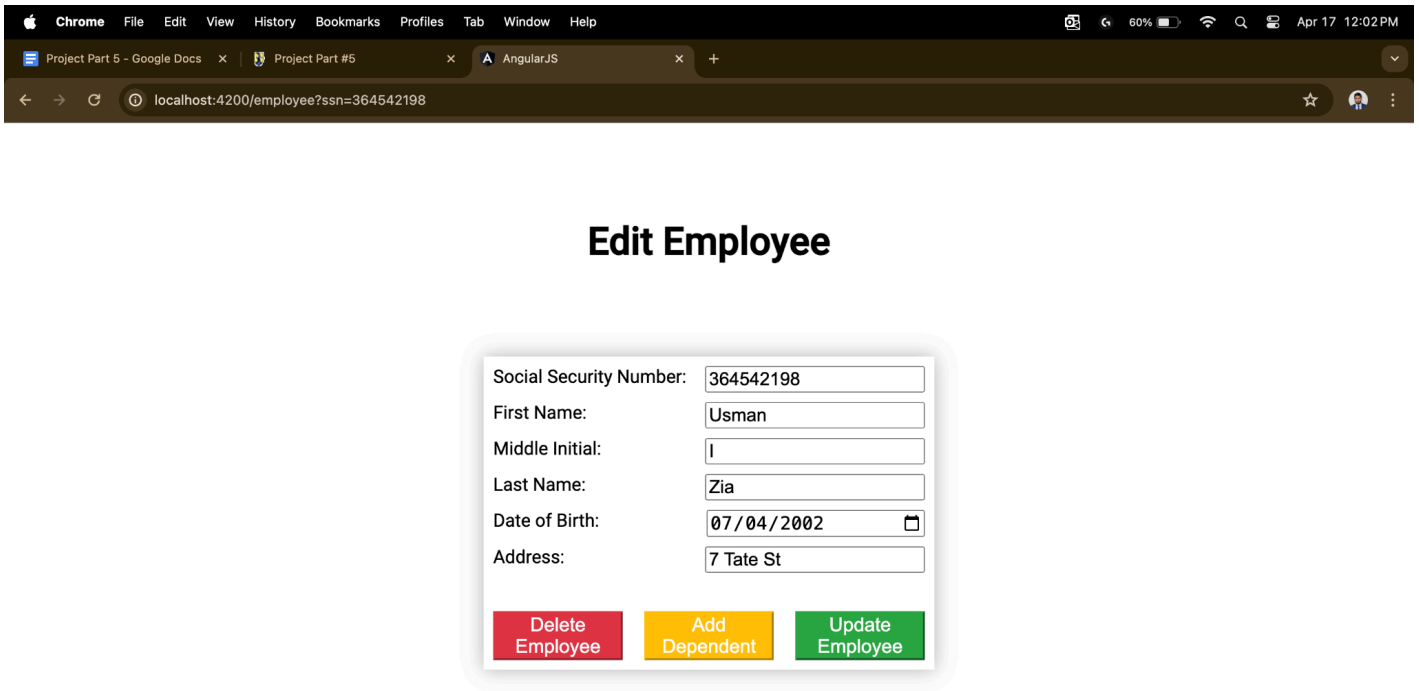


Demonstrating the Updating functionality

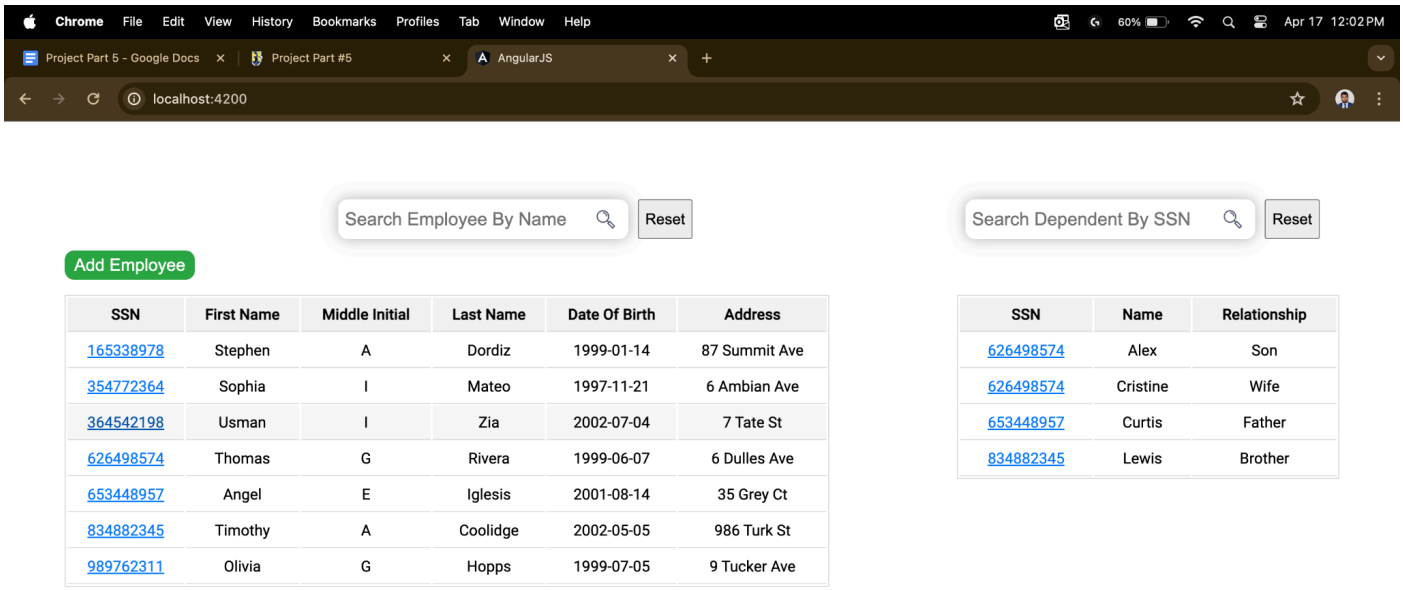
Let’s click the Employee’s SSN to update the employee information



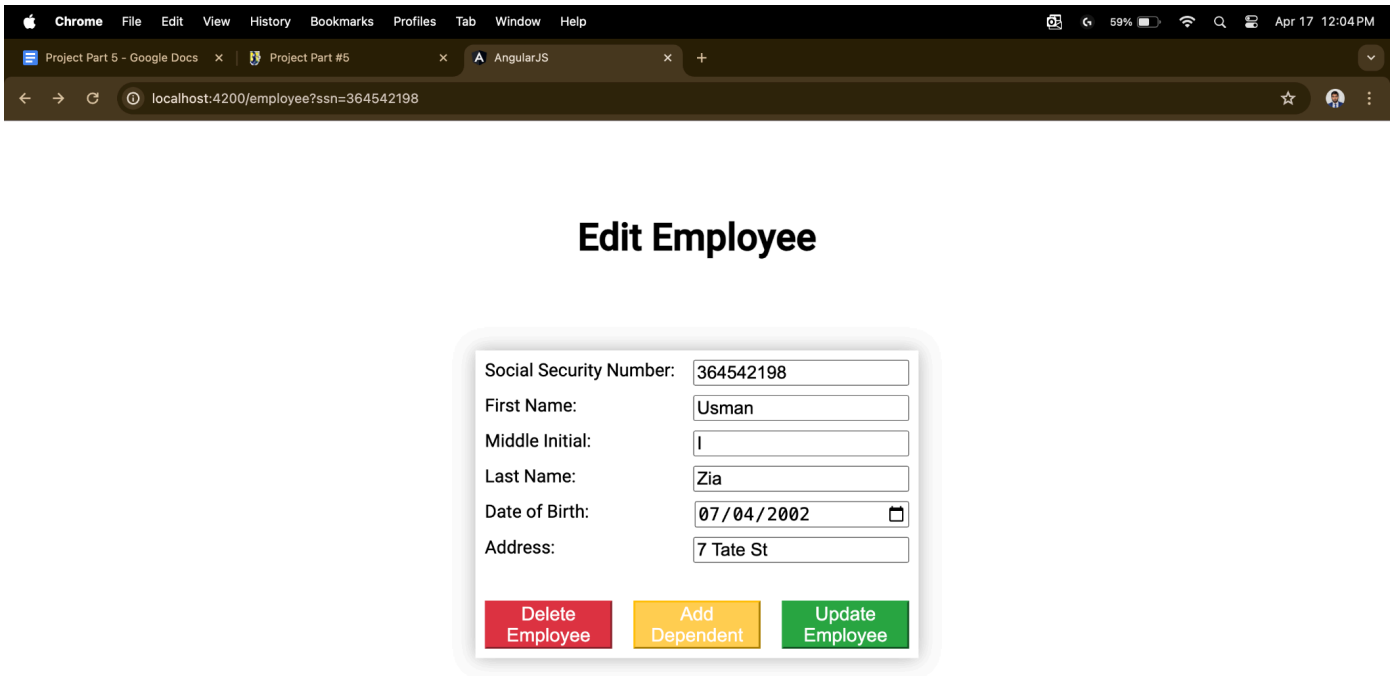
Now we change the employee’s birthday and address, press “Update Employee”



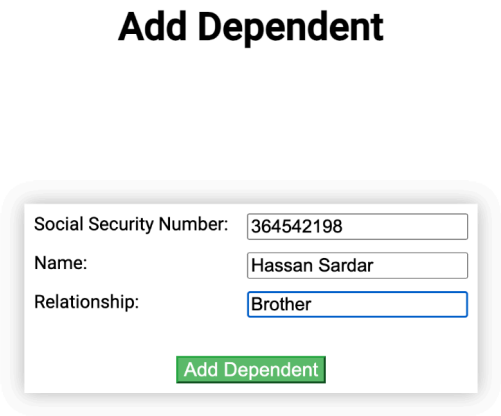
Now, the employee's information has been updated



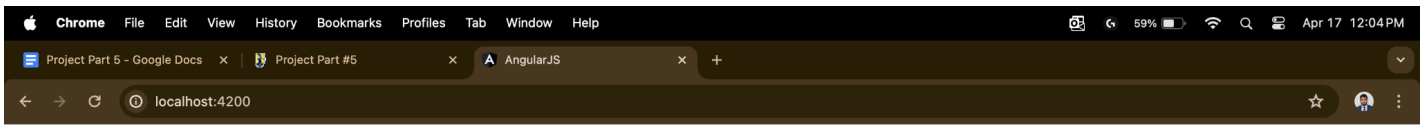
Now, for the newly added employee, let's add some dependents. To do that, click on the Employee SSN



Click the “Add Dependent” button to add a Dependent to an Employee, and Enter the Dependent’s information and click Add Dependent



We can see that the dependent was added successfully



Search Employee By Name Reset

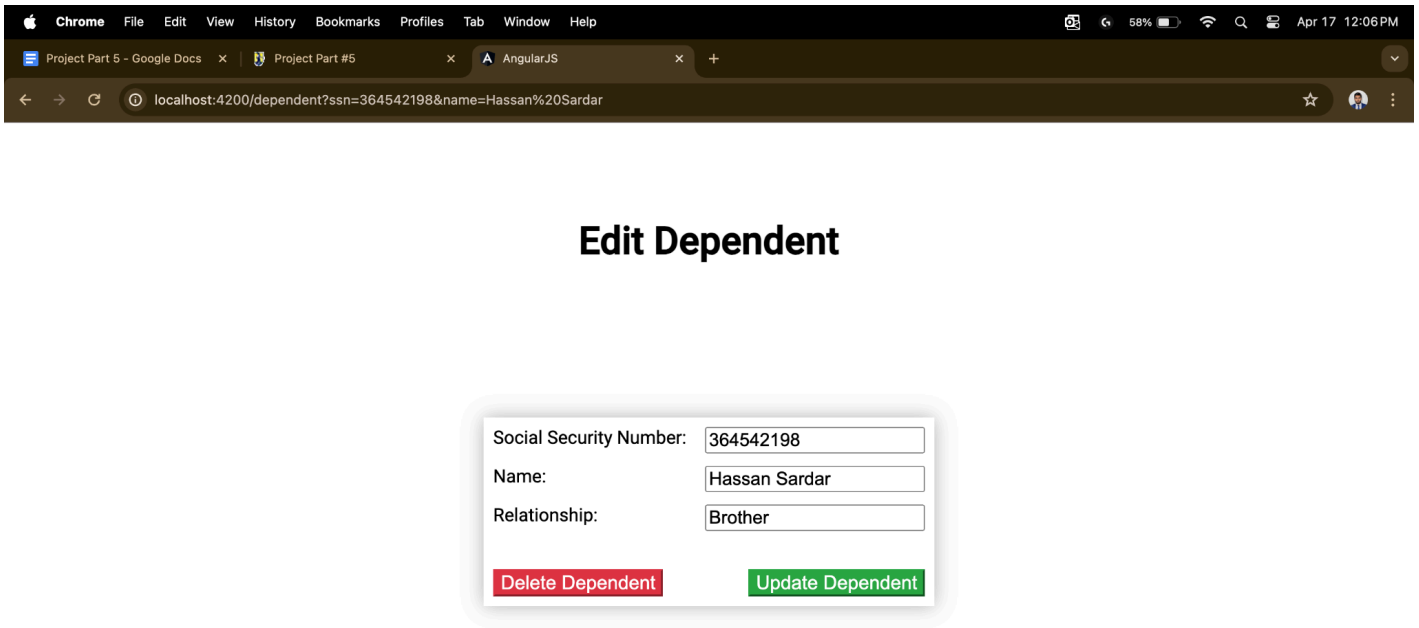
Search Dependent By SSN Reset

Add Employee

SSN	First Name	Middle Initial	Last Name	Date Of Birth	Address
165338978	Stephen	A	Dordiz	1999-01-14	87 Summit Ave
354772364	Sophia	I	Mateo	1997-11-21	6 Ambian Ave
364542198	Usman	I	Zia	2002-07-04	7 Tate St
626498574	Thomas	G	Rivera	1999-06-07	6 Dulles Ave
653448957	Angel	E	Iglesis	2001-08-14	35 Grey Ct
834882345	Timothy	A	Coolidge	2002-05-05	986 Turk St
989762311	Olivia	G	Hopps	1999-07-05	9 Tucker Ave

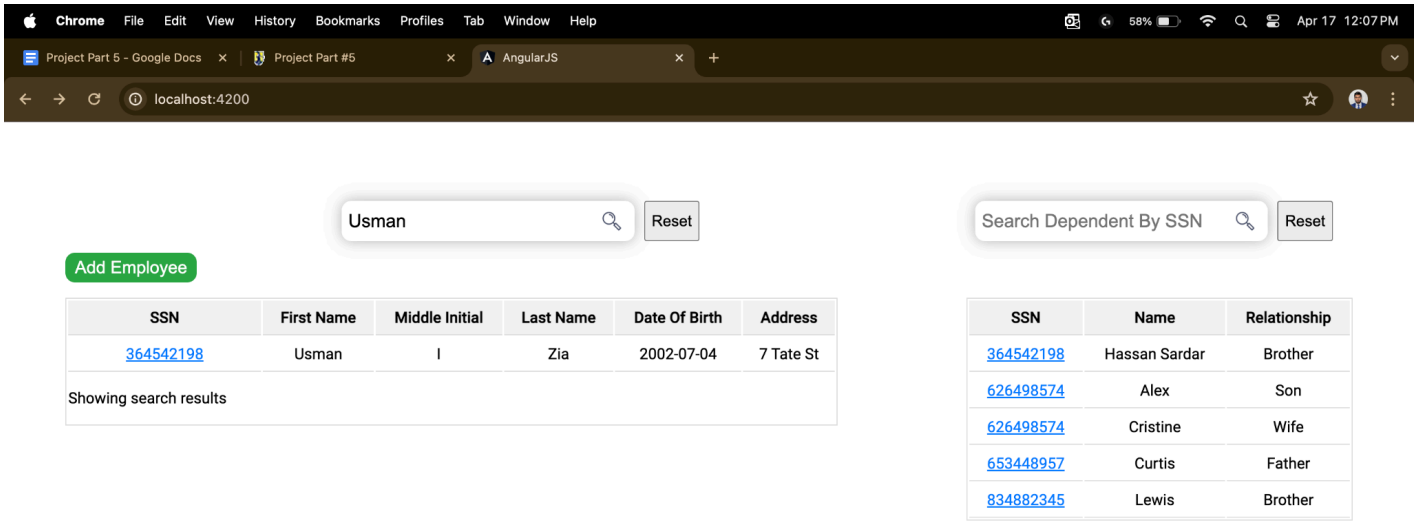
SSN	Name	Relationship
364542198	Hassan Sardar	Brother
626498574	Alex	Son
626498574	Cristine	Wife
653448957	Curtis	Father
834882345	Lewis	Brother

Clicking on the dependent also allows for the dependent information to be updated

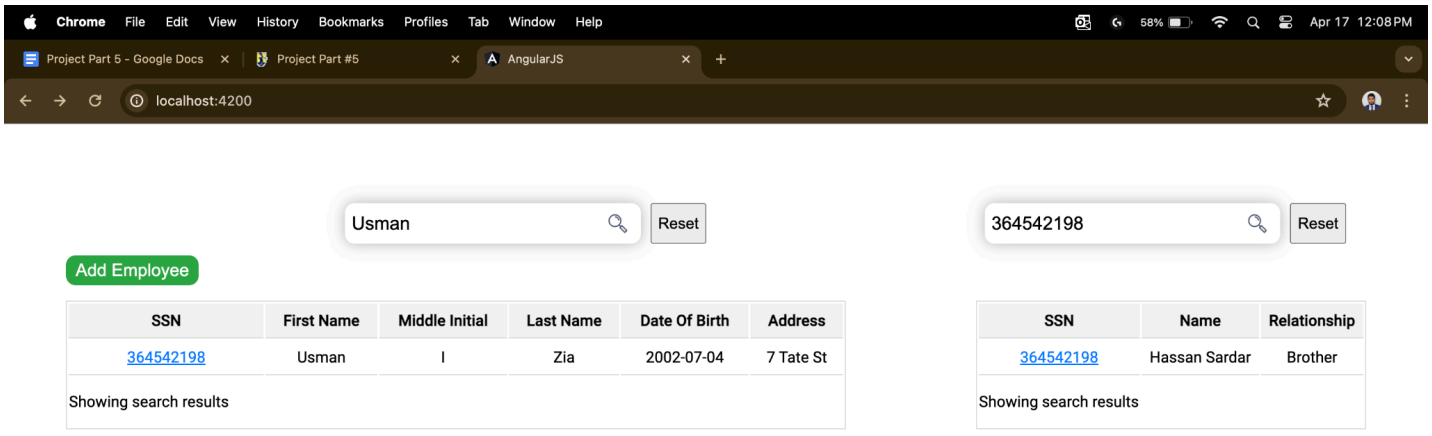


Demonstrating the looking-up functionality

An employee can be searched by their Name

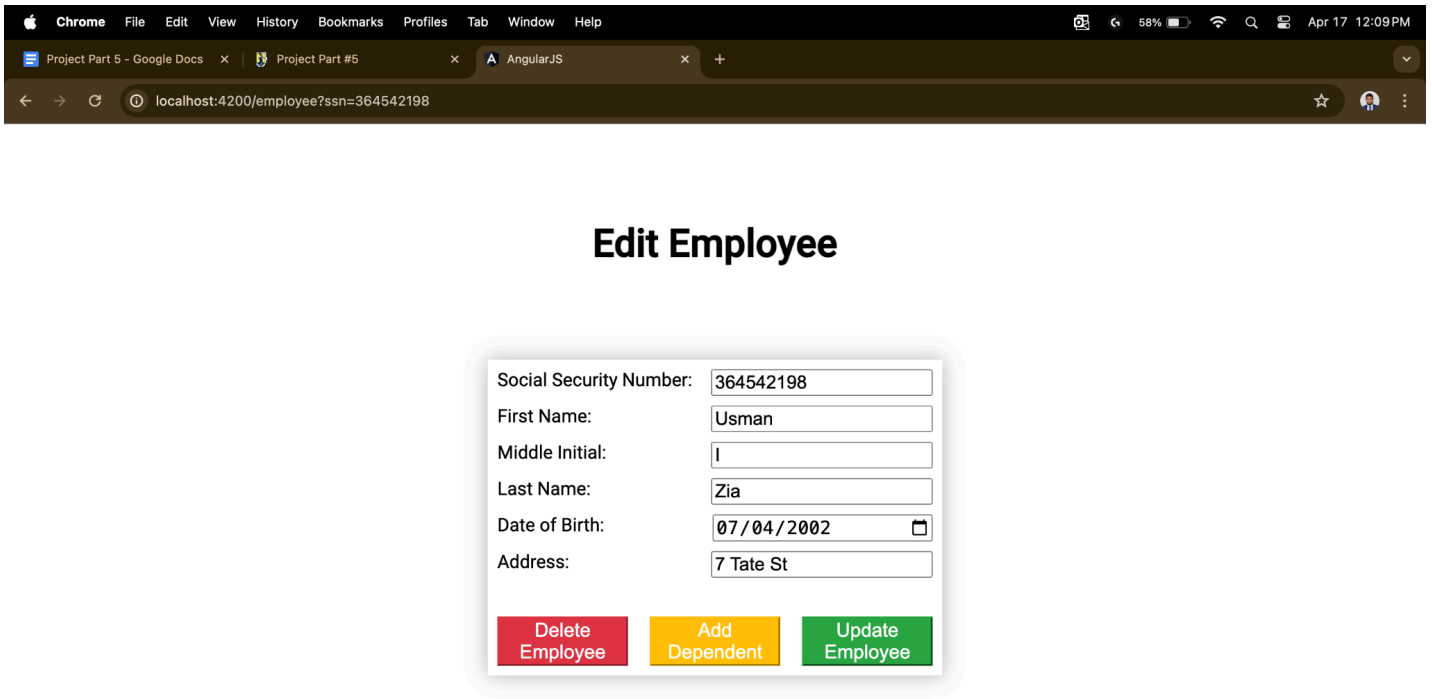


Similarly, the dependents of an employee can be searched by the Employee's SSN

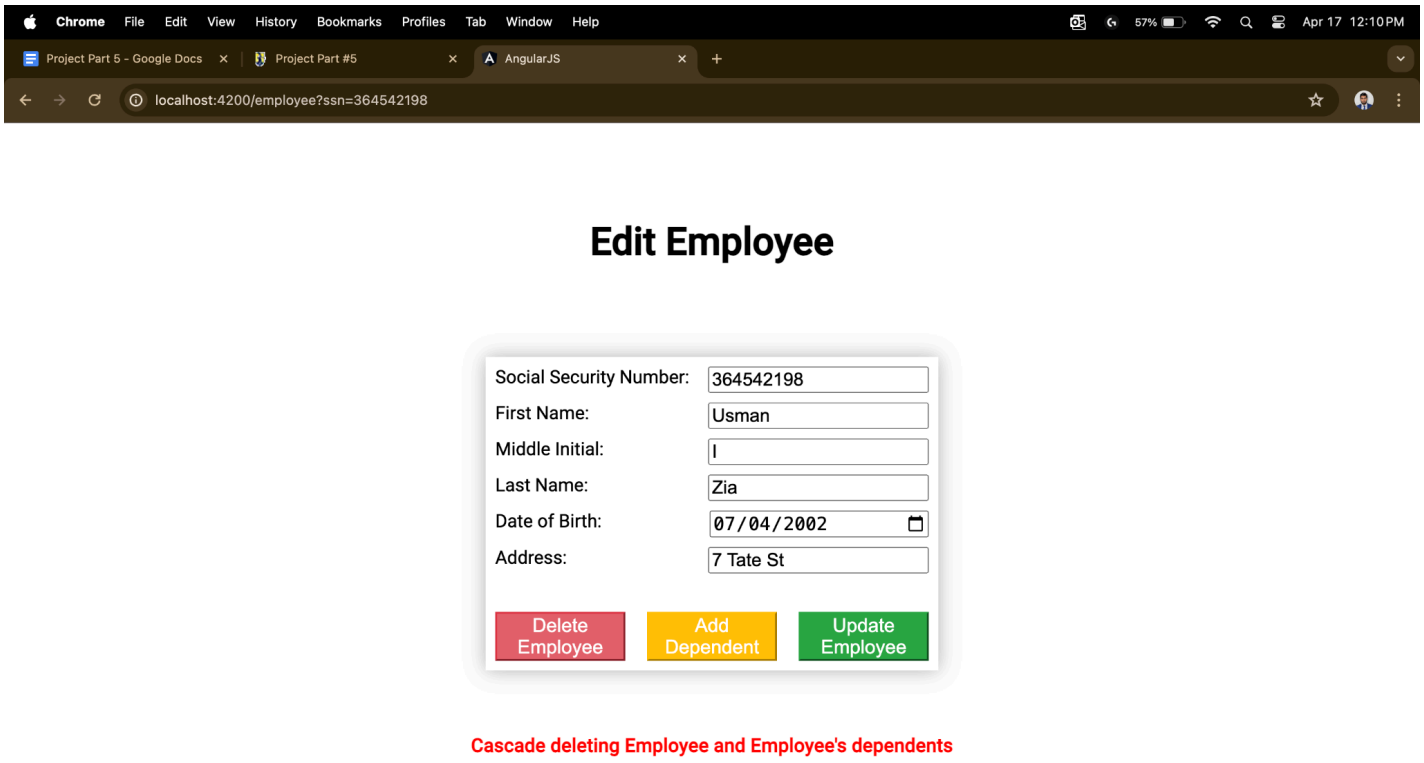


Demonstrating the deleting functionality

Let’s try deleting the newly added Employee, We will click on the Employee SSN and hit the Delete Employee button



Notice that deleting an employee cascade deletes the employee's dependents



Now the employee and their dependents are successfully deleted