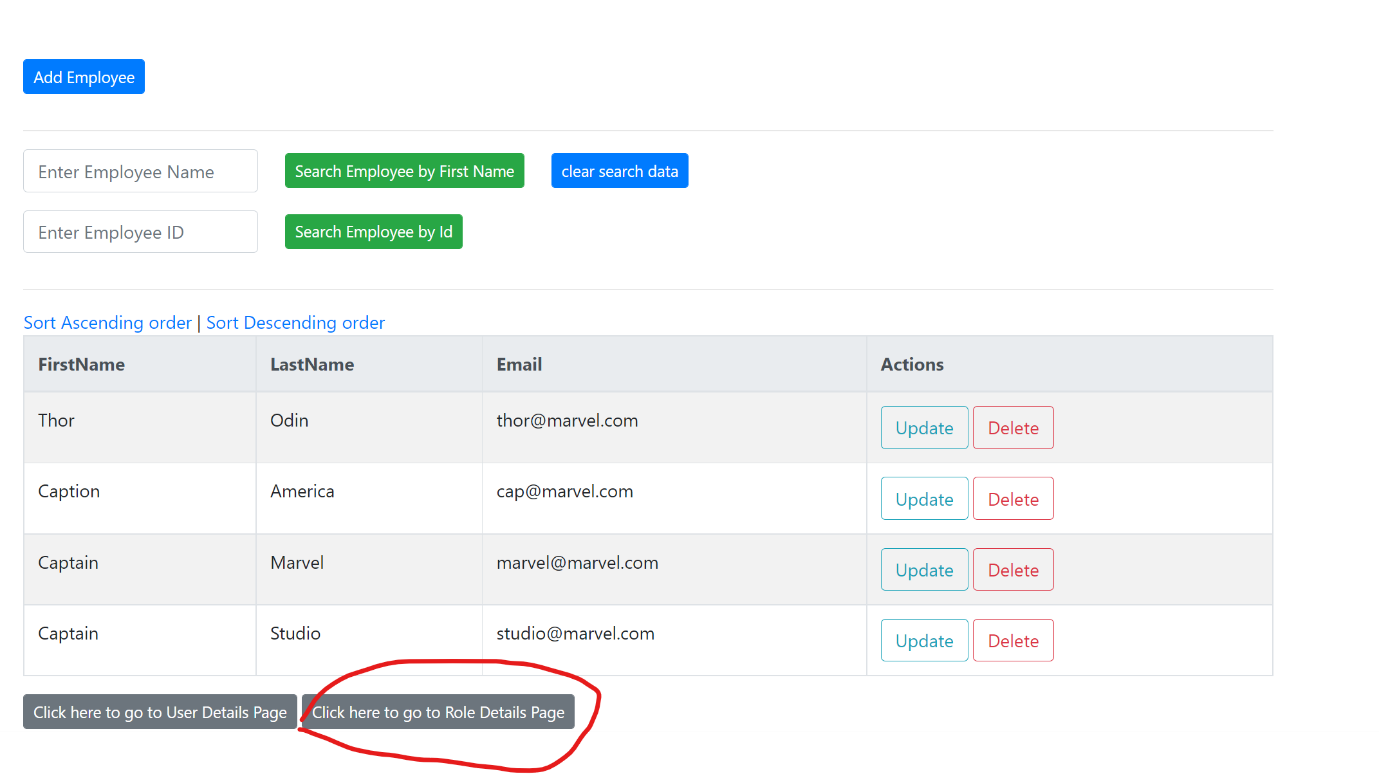
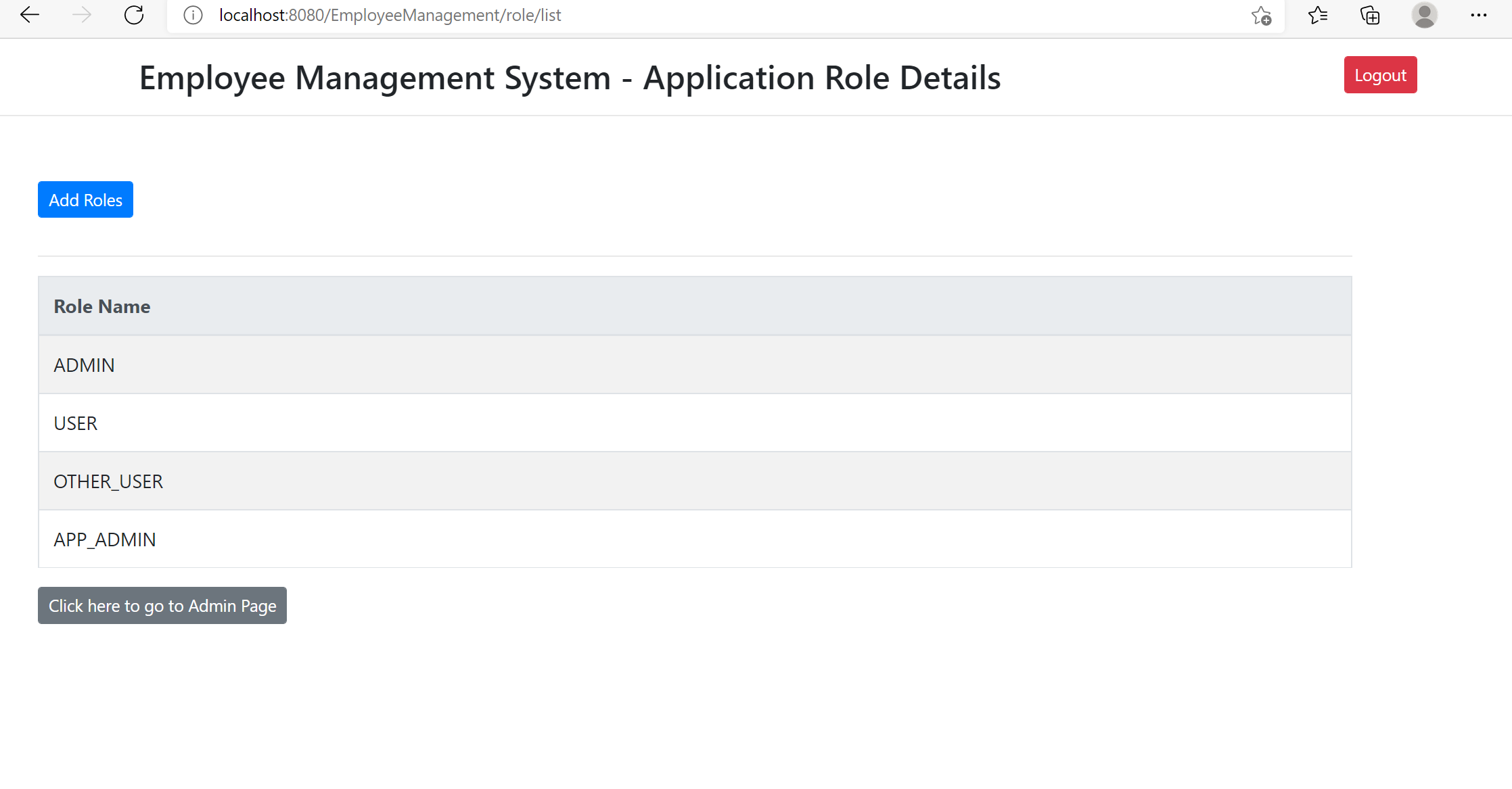
EMPLOYEE MANAGEMENT SYSTEM

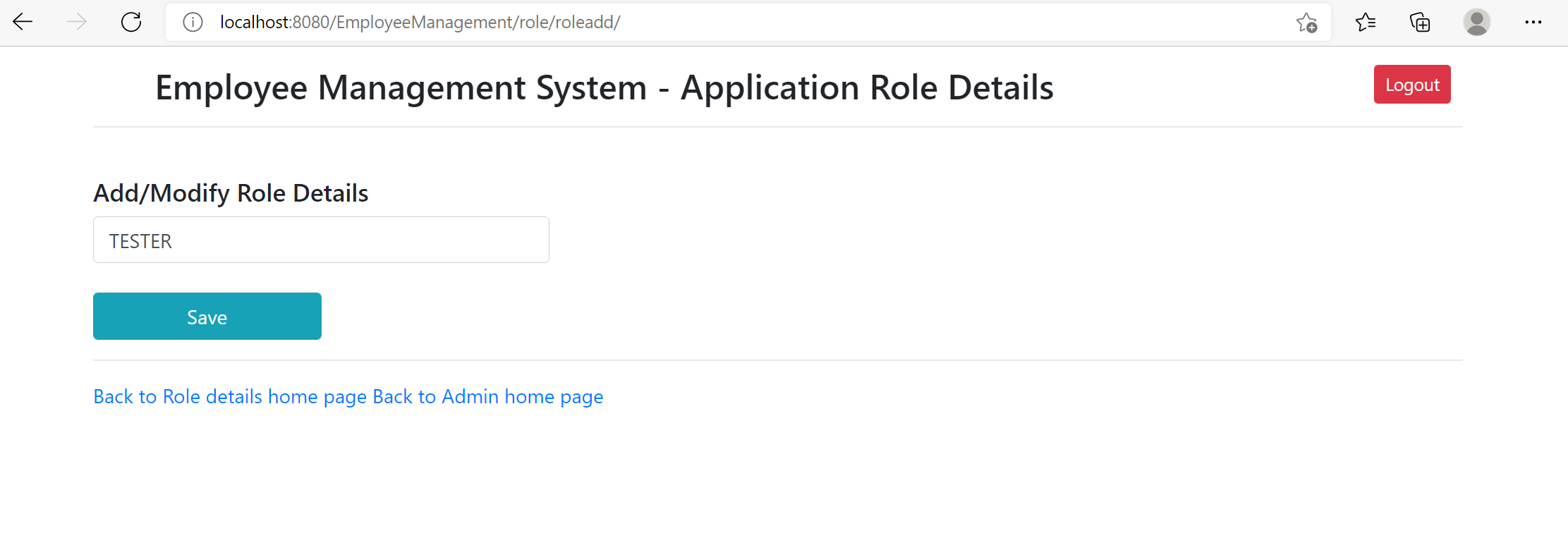
1.Your application should be able to add roles in the database dynamically in the db.

List roles - <http://localhost:8080/EmployeeManagement/role/list>

Add Roles - http://localhost:8080/EmployeeManagement/role/roleadd/



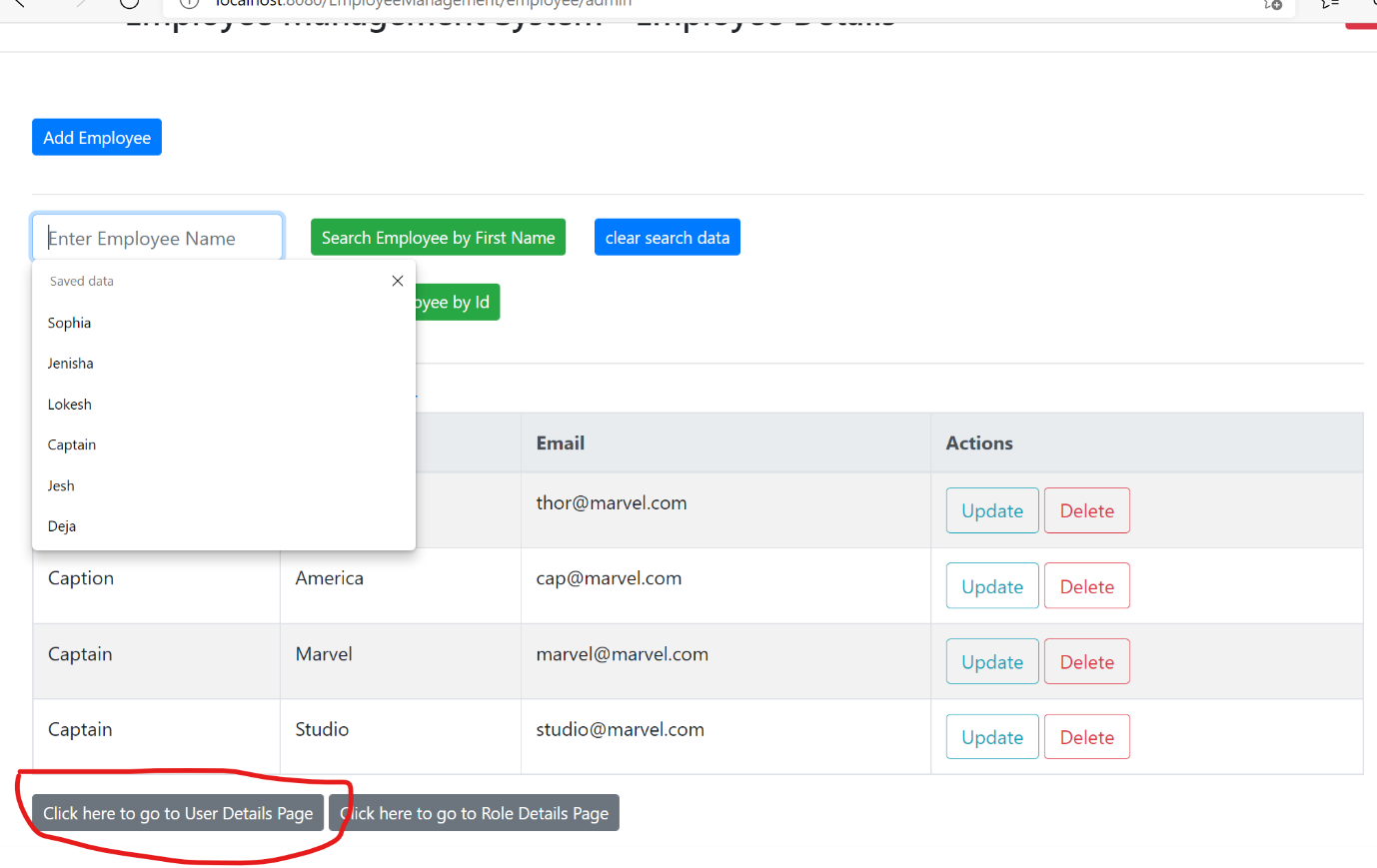


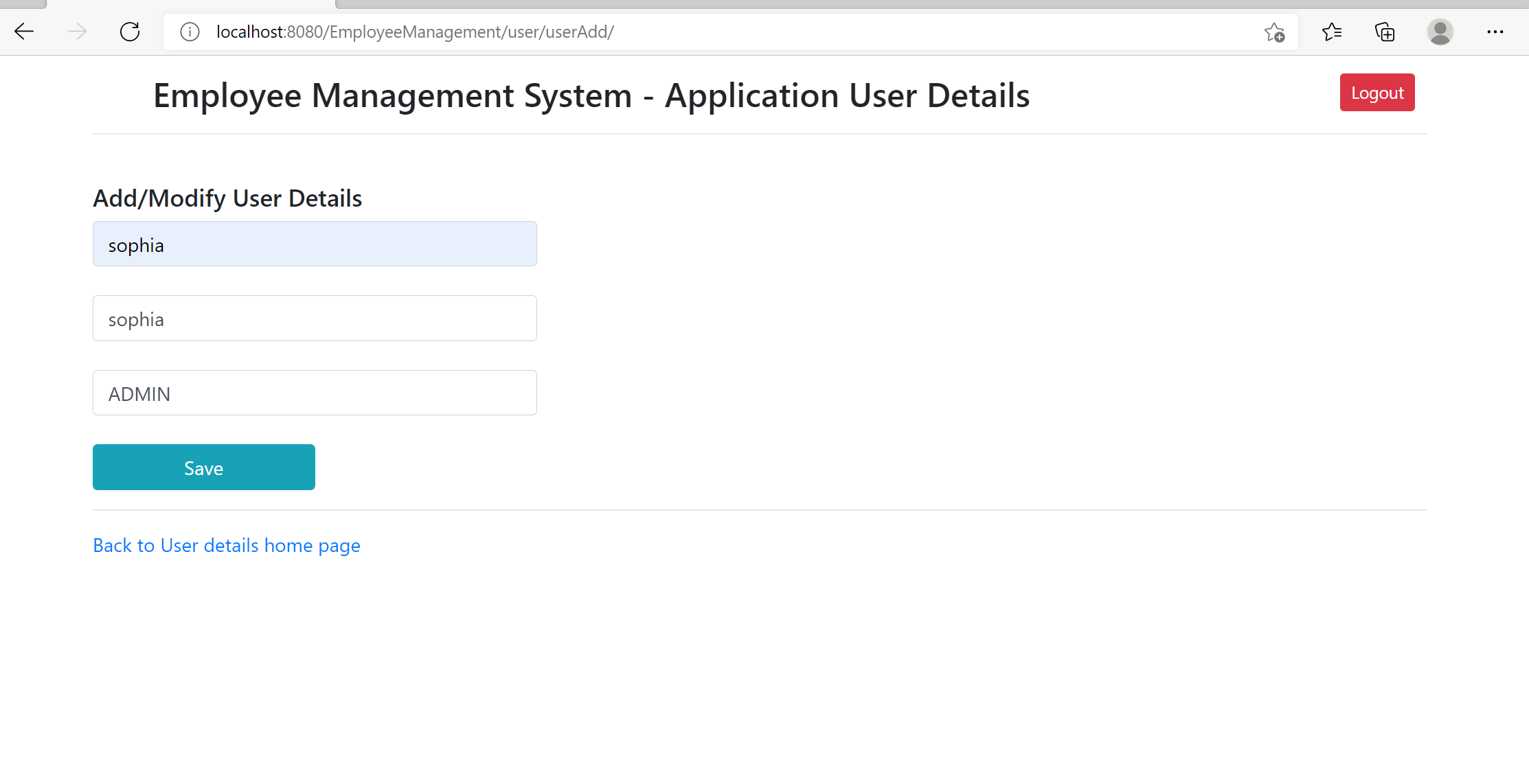


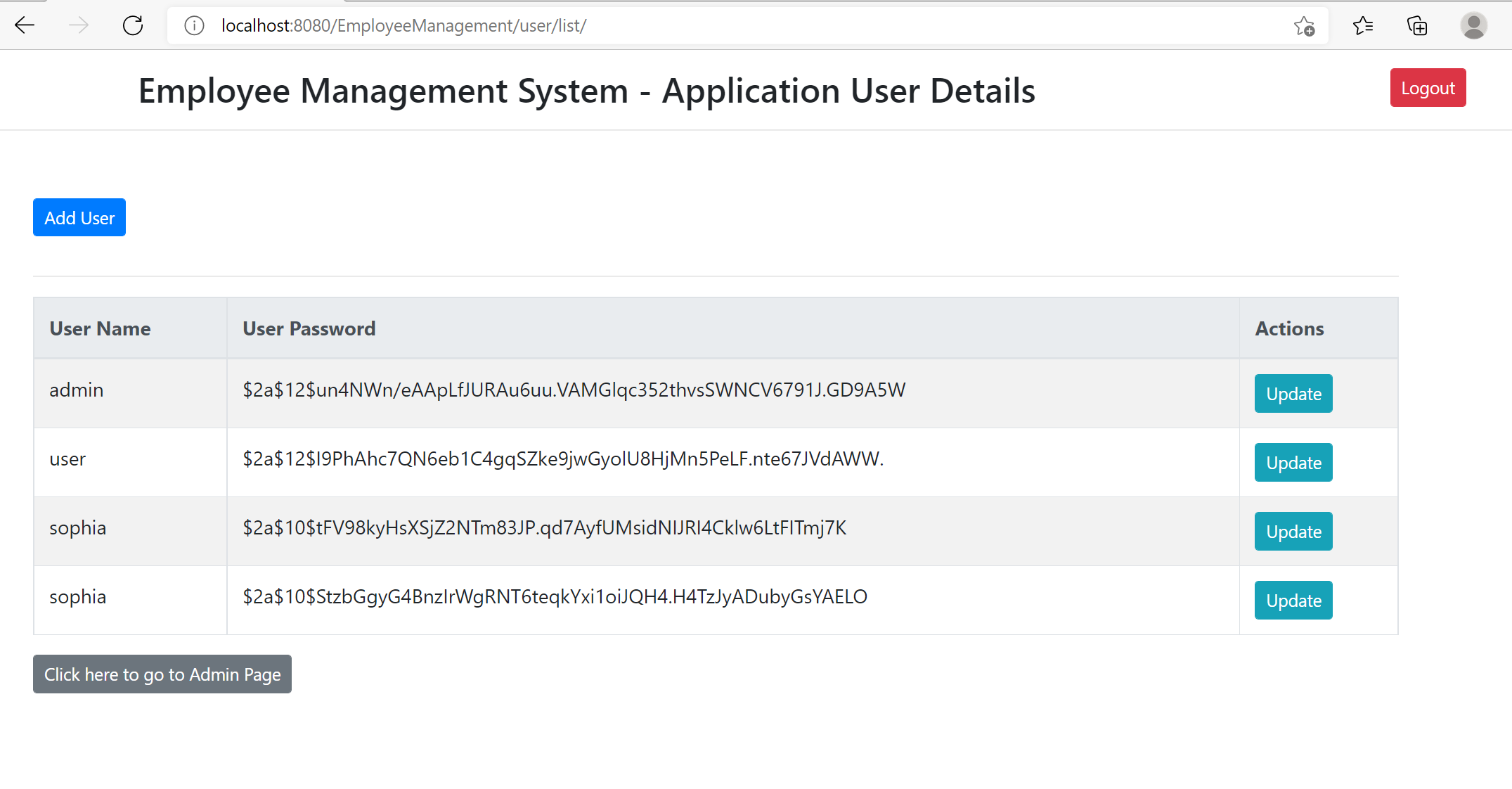
2. Your application should be able to add Users in the db which can be used for authentication purposes.

List user details <http://localhost:8080/EmployeeManagement/user/list>

Add user details - <http://localhost:8080/EmployeeManagement/user/userAdd/>





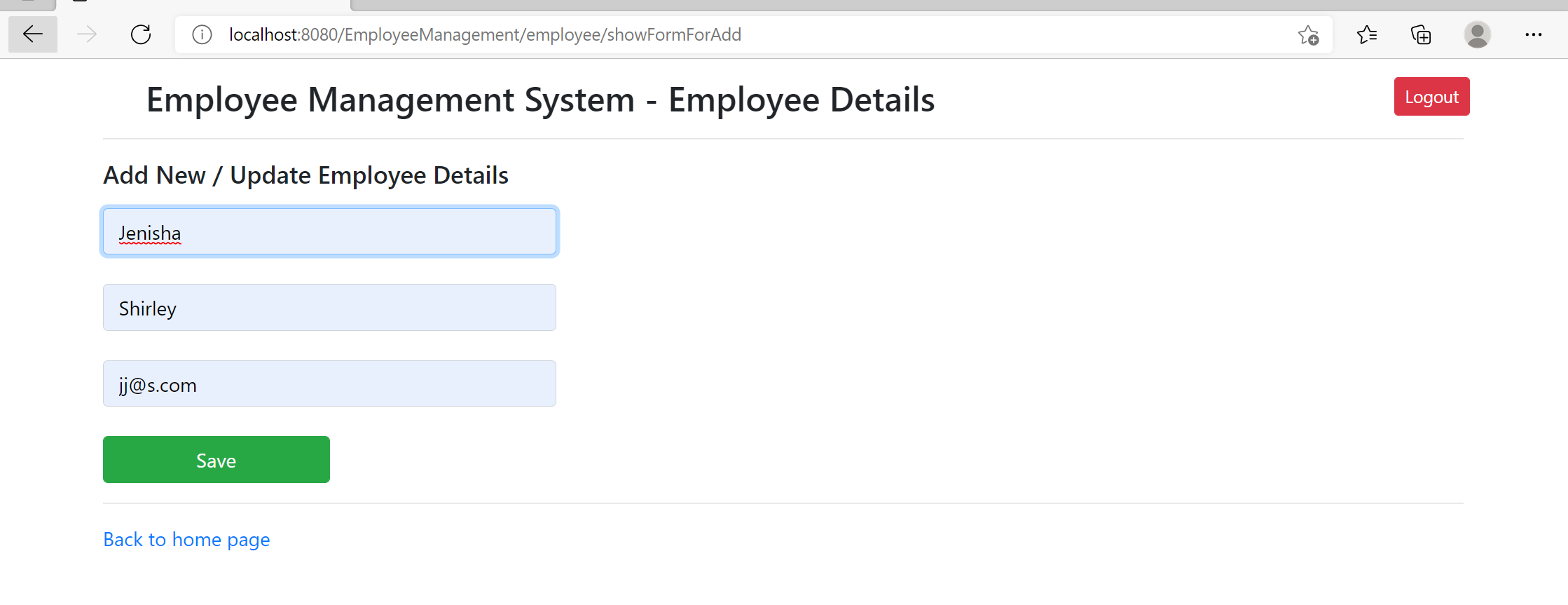


3. Now Your application should be able to add employees data in the db if and only if the authenticated user is **ADMIN**-

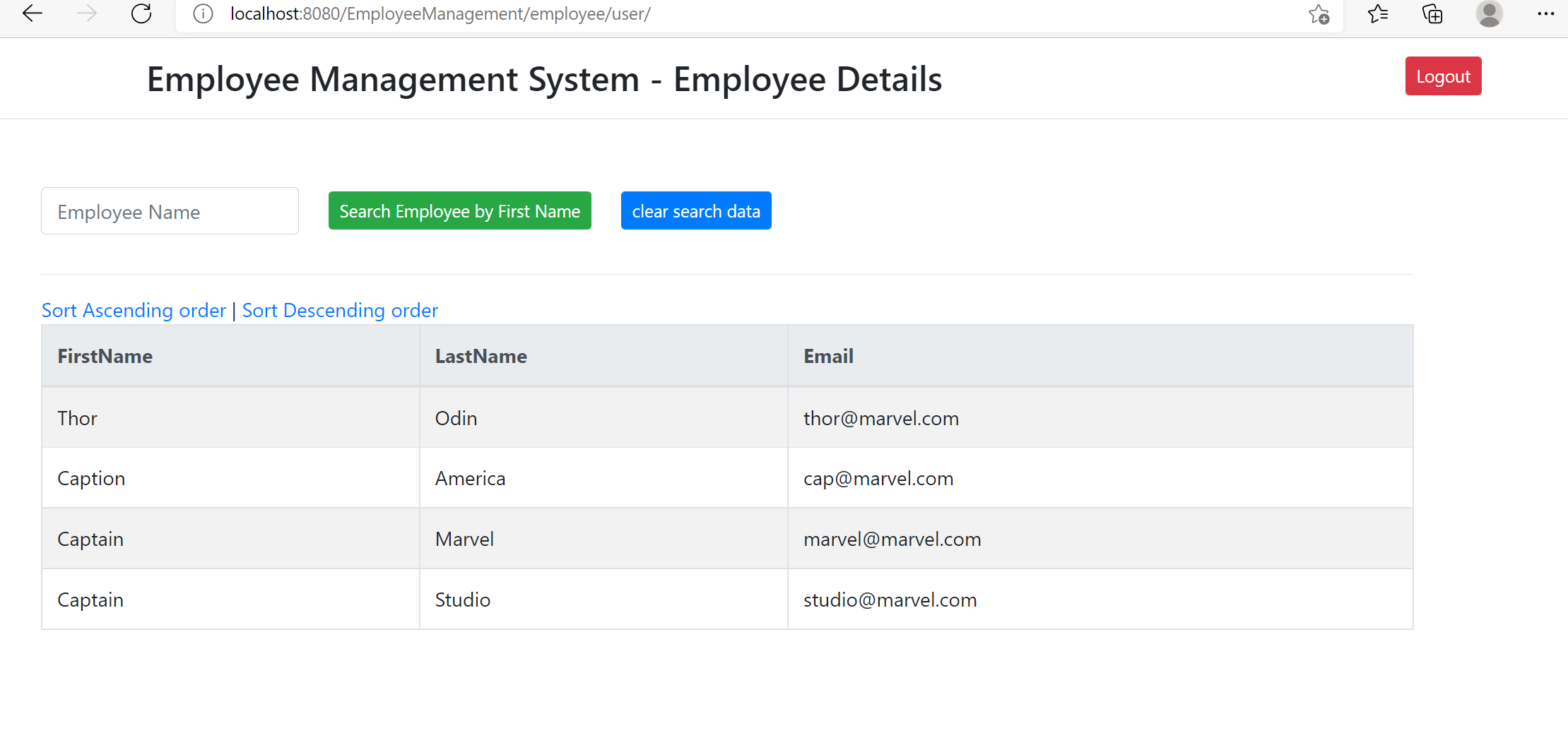
Add user option in admin login

<http://localhost:8080/EmployeeManagement/employee/showFormForAdd>

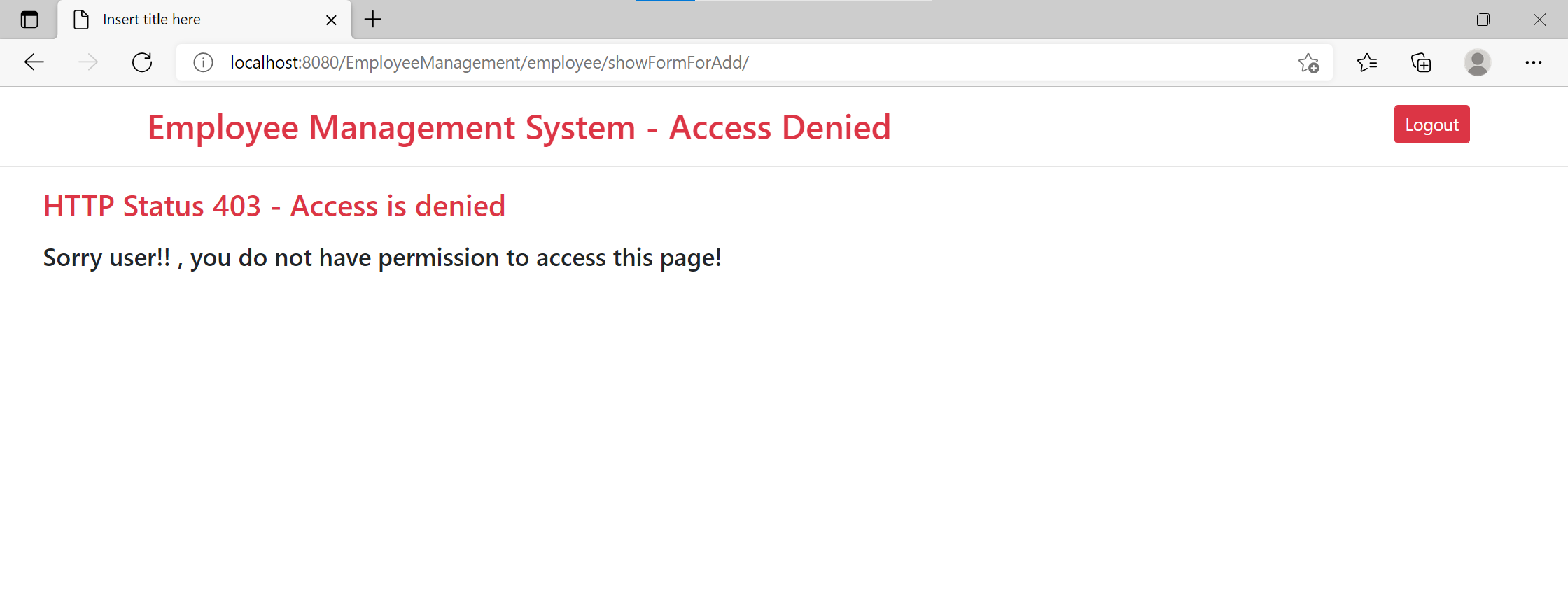
Add user option hidden in User login



User login – Add user option not present



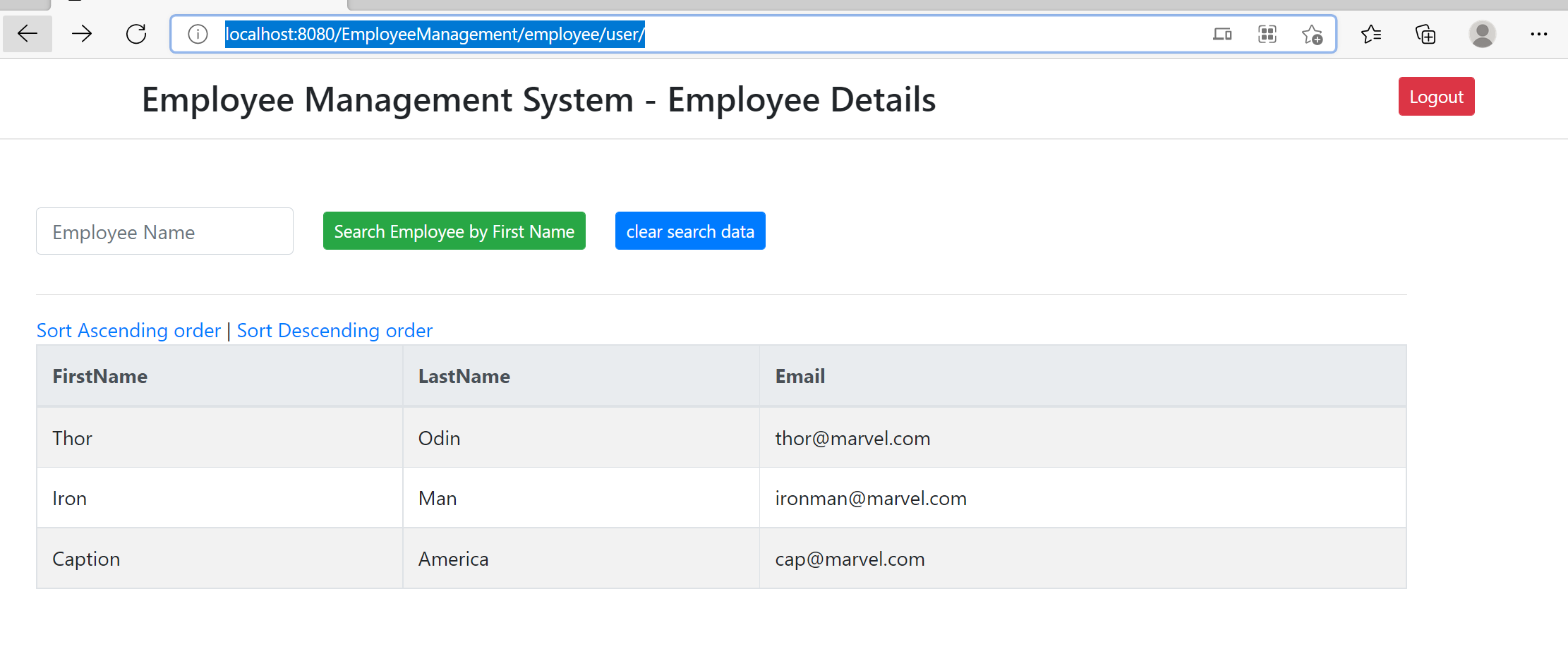
On accessing the endpoint as user



4. Your application should provide an endpoint to list all the employees stored in the database.

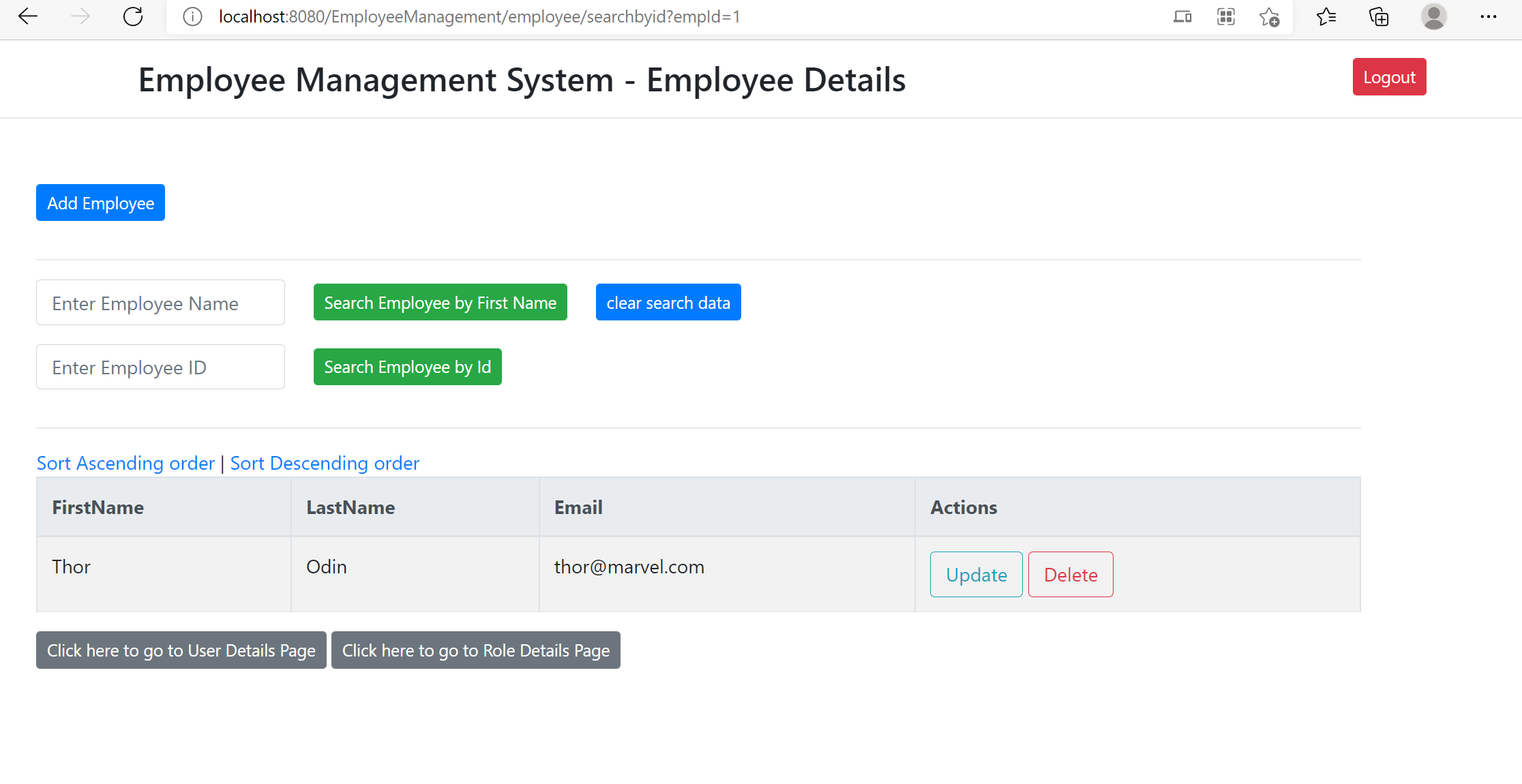
User endpoint -> <http://localhost:8080/EmployeeManagement/employee/user/>

Admin endpoint -> <http://localhost:8080/EmployeeManagement/employee/admin/>



5. Your application should provide endpoint to fetch or get an employee record specifically based on the id of that employee-

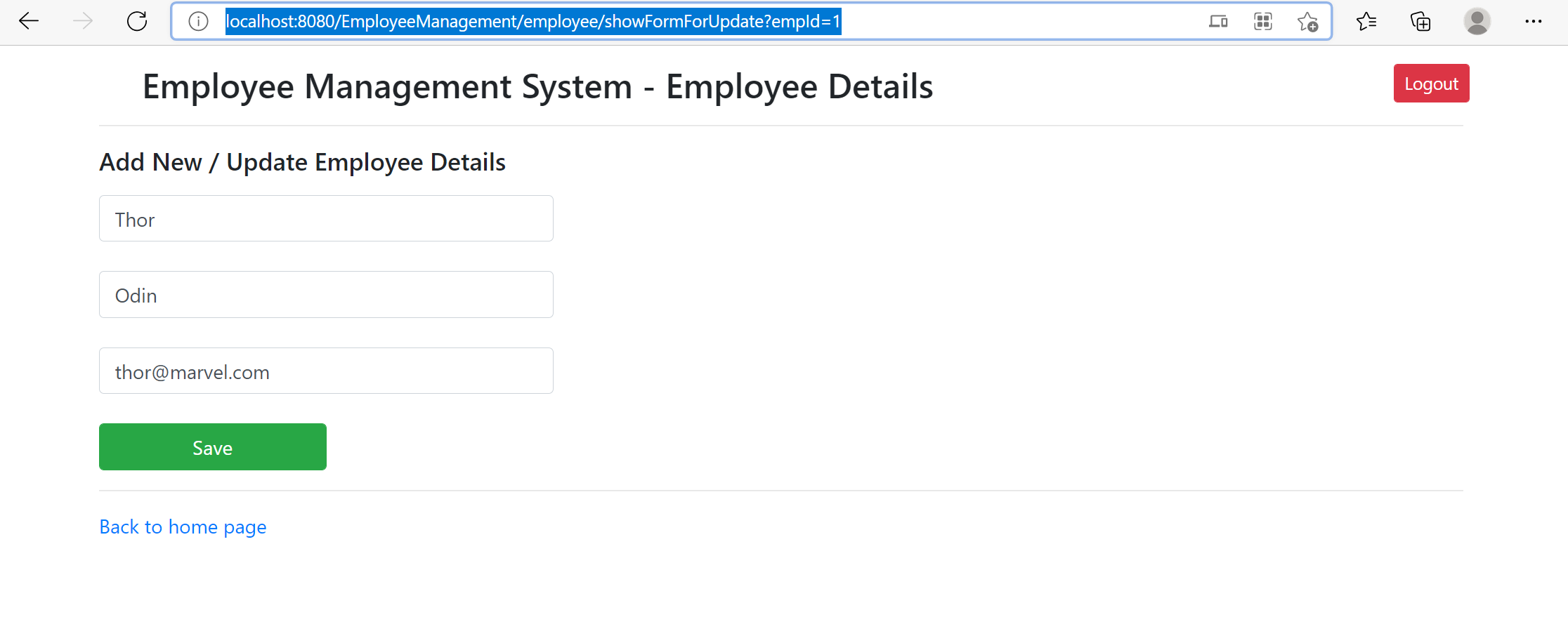
Search by ID -> <http://localhost:8080/EmployeeManagement/employee/searchbyid?empId=1>



6. Your application should provide an endpoint to update an existing employee record with the given updated json object.

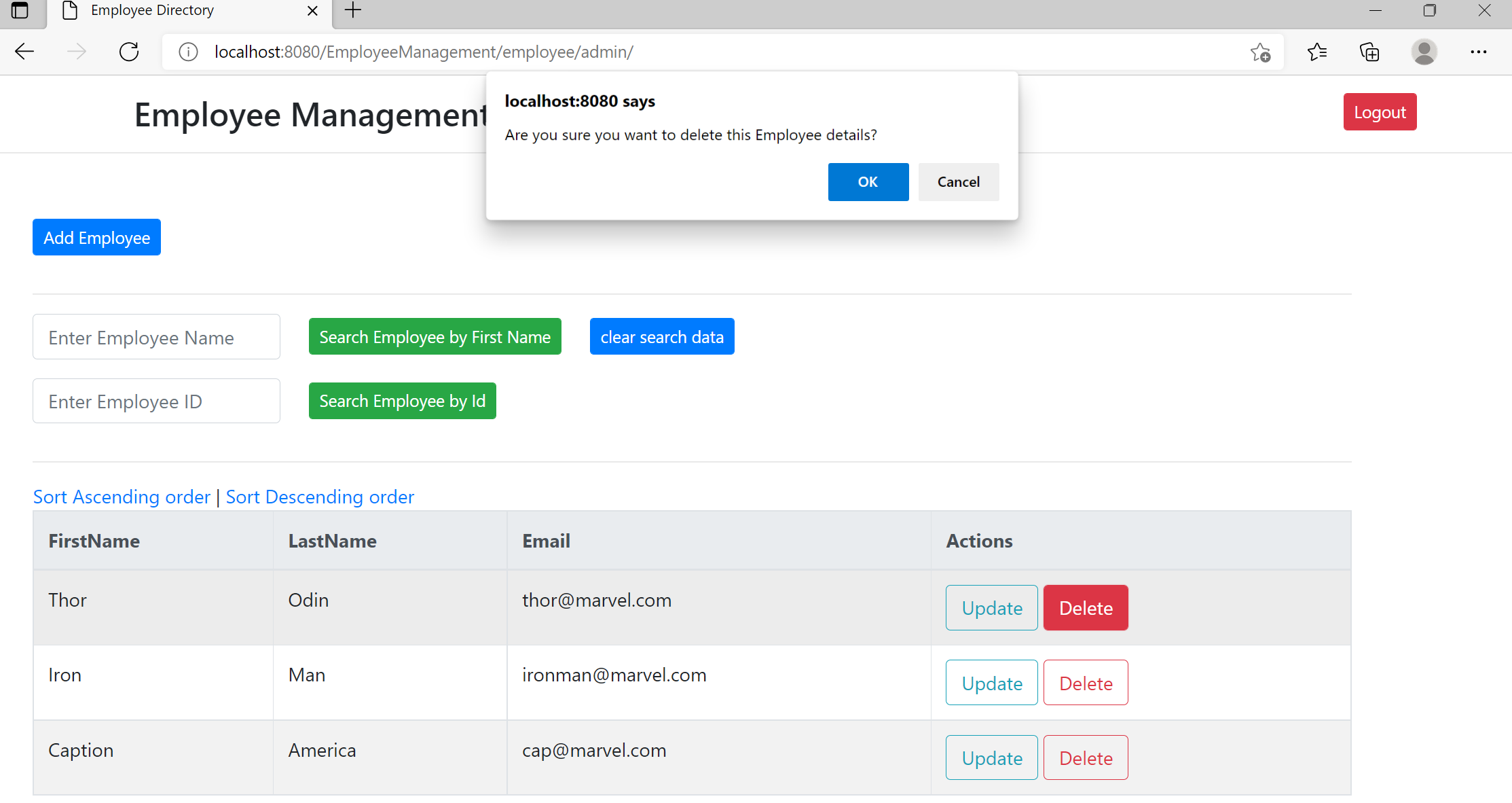
Update Employee details

<http://localhost:8080/EmployeeManagement/employee/showFormForUpdate?empId=1>



7. Your application should also provide an endpoint to delete an existing employee record based on the id of the employee-

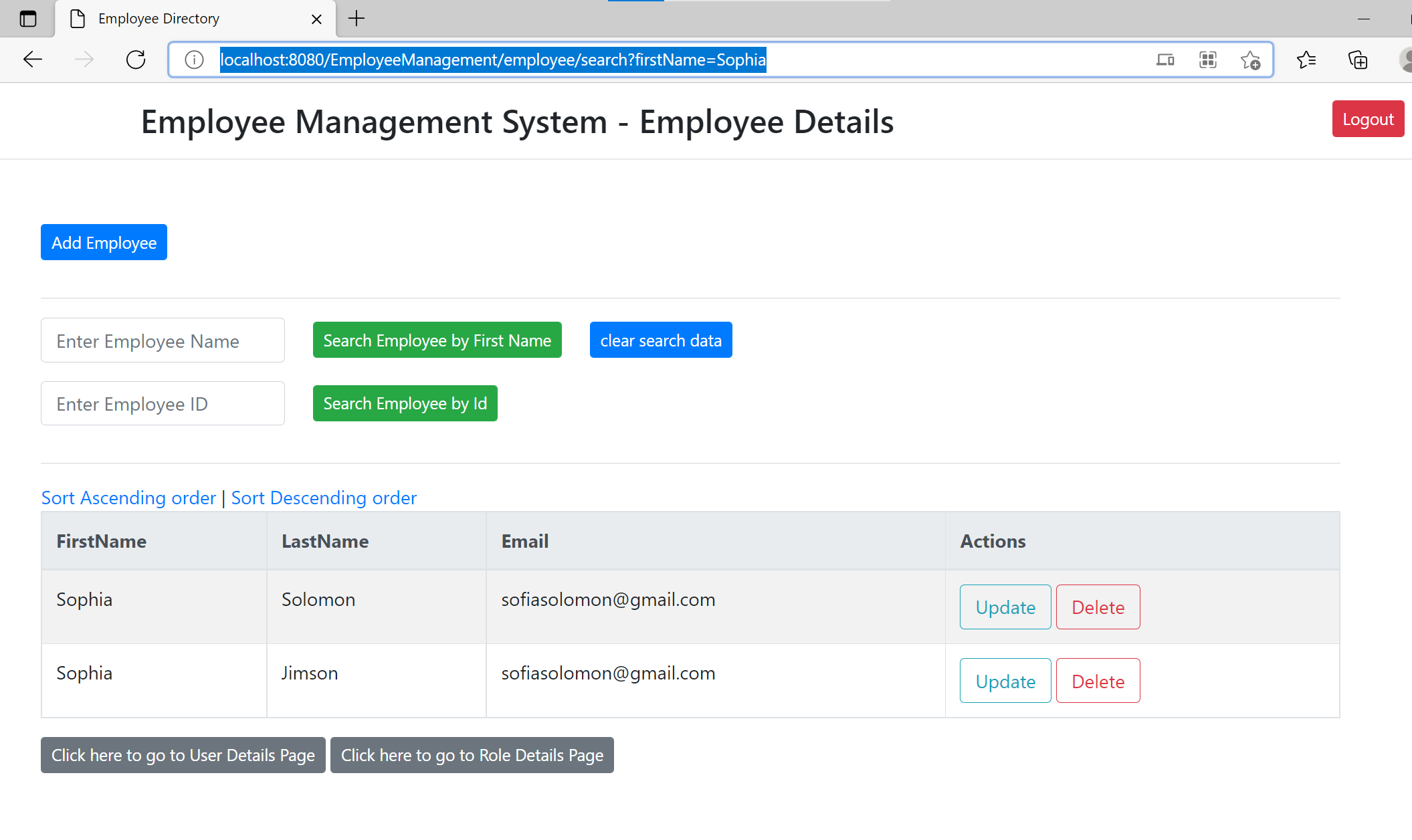
Delete endpoint - [localhost:8080/EmployeeManagement/employee/delete?empId=2](http://localhost:8080/EmployeeManagement/employee/delete?empId=2)



8. Your application should provide an endpoint to fetch an employee by his/her first name and if found more than one record then list them all-

Search Employee -

<http://localhost:8080/EmployeeManagement/employee/search?firstName=Sophia>



9. Your application should be able to list all employee records sorted on their first name in either ascending order or descending order .

Sort ascending order

<http://localhost:8080/EmployeeManagement/employee/adminsort?order=1>

Sort Descending Order

<http://localhost:8080/EmployeeManagement/employee/adminsort?order=2>

