

Created Add Employee Form with basic details.

No address is added.

The result json response after adding new Employee.

​​

{

"empId": 44,

"firstName": "Shireen",

"lastName": "Syed",

"everestEmailId": "syed.shireen@everest.engineering",

"password": "Sohail@0553",

"personalEmailId": "syedshireen@gmail.com",

"dateOfBirth": **null**,

"dateOfJoin": **null**,

"designation": **null**,

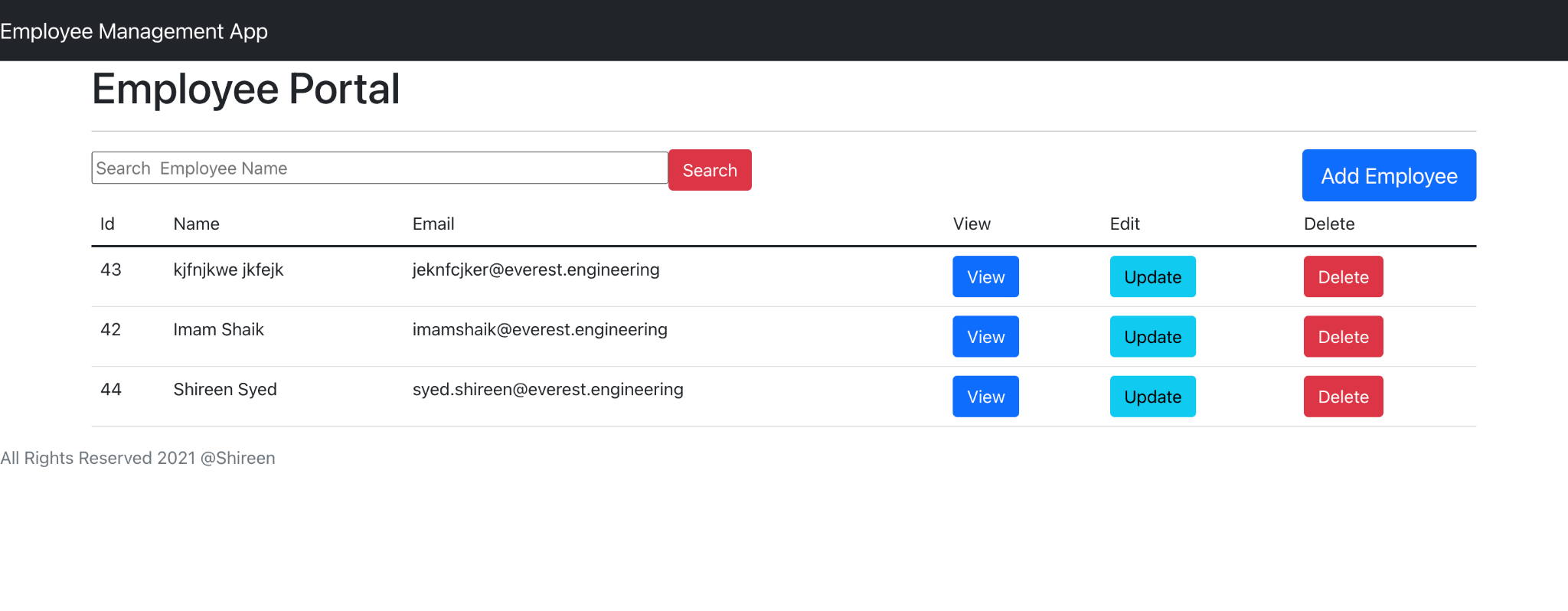
"experience": **null**,

"bio": **null**,

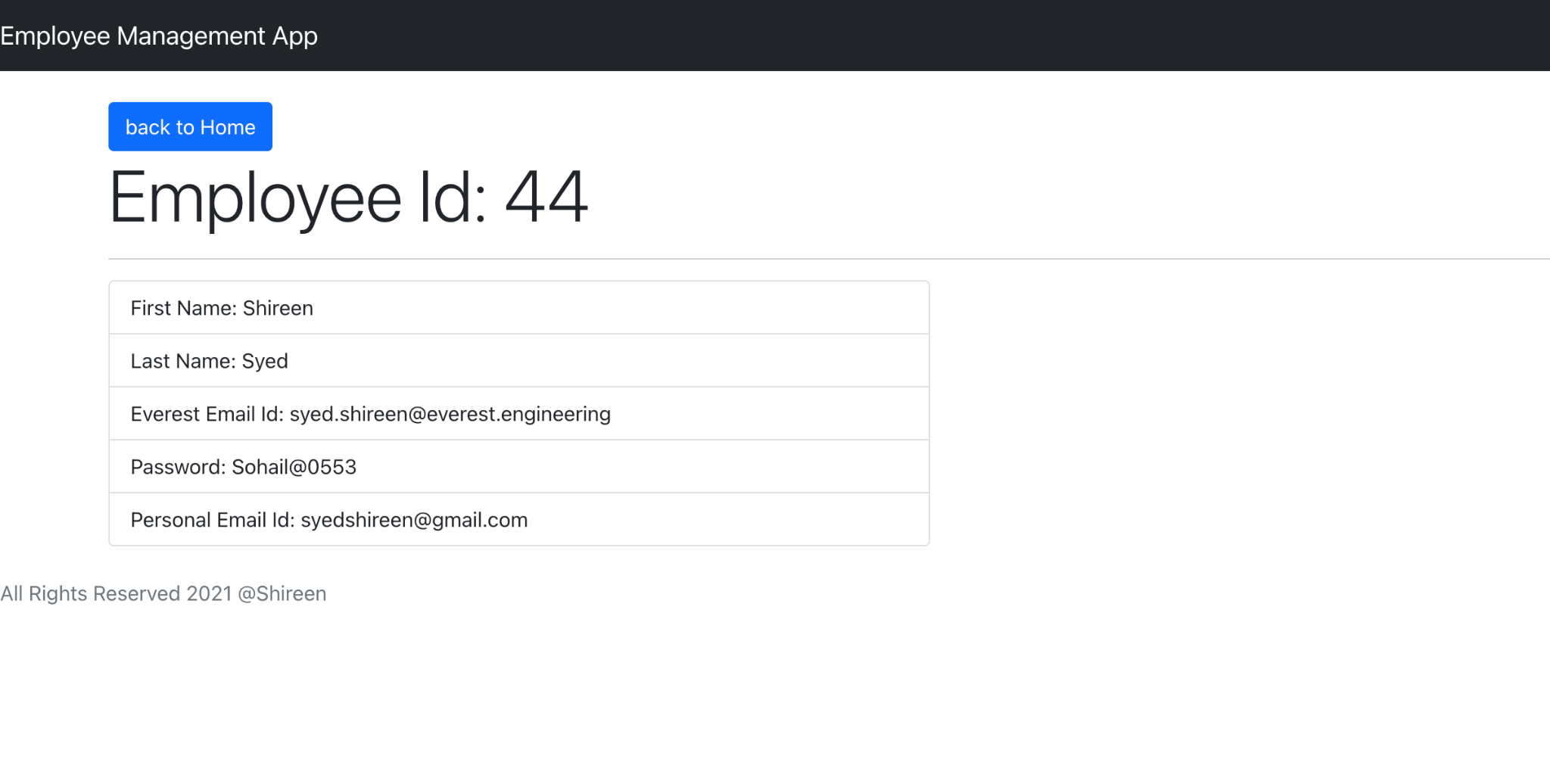
"presentAddress": **null**,

"permanentAddress": **null**

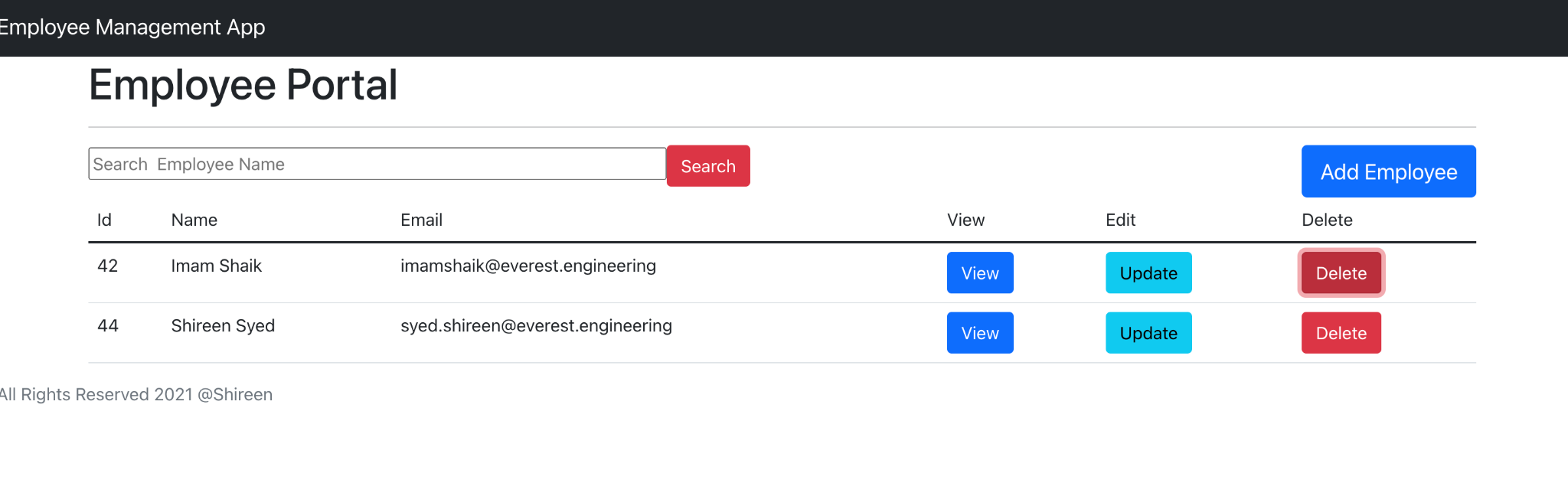
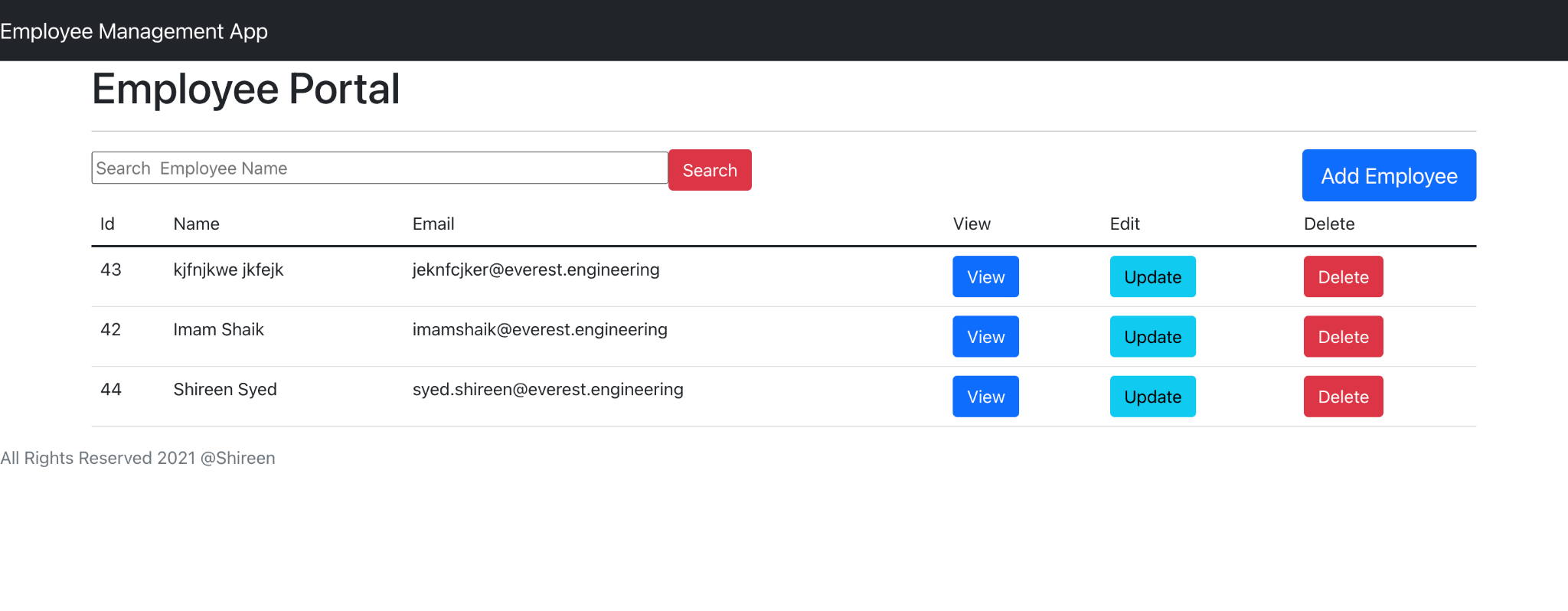
}



Fetched Employees Successfully.

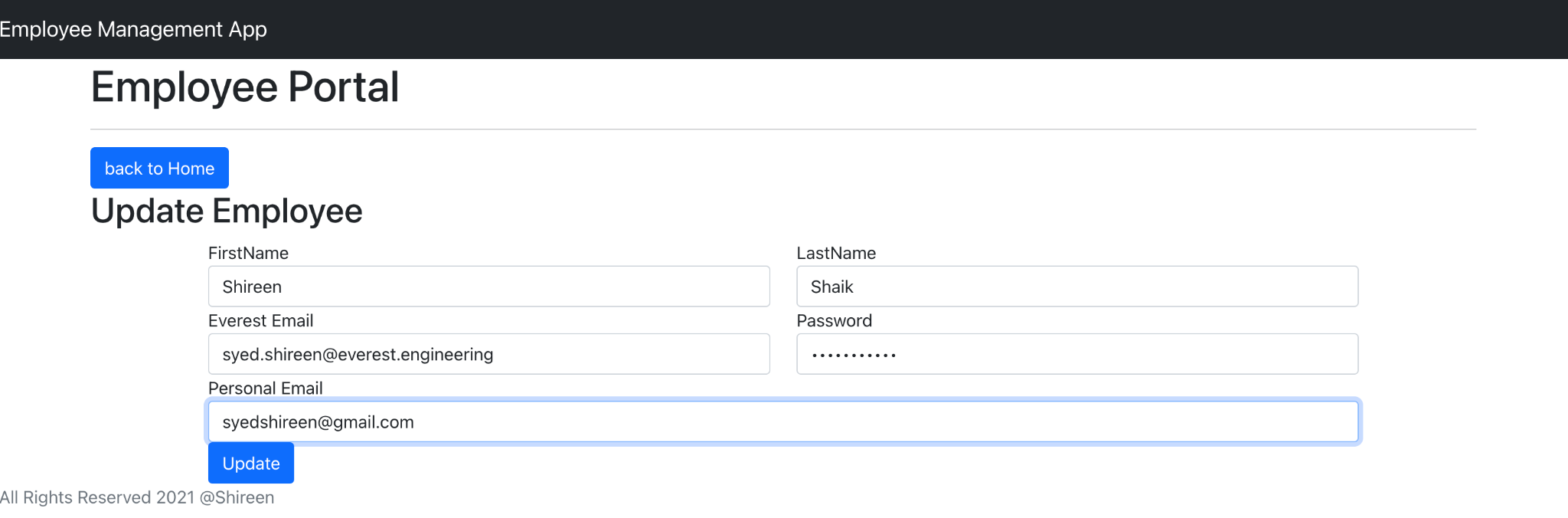


Fetched Employee ById successfully.



Deleted Employee Successfully

UPDATE EMPLOYEES



java.lang.NullPointerException: Cannot invoke "com.everestengineering.employeeportalapplication.entities.Address.getId()" because the return value of "com.everestengineering.employeeportalapplication.entities.Employee.getPresentAddress()" is null

at com.everestengineering.employeeportalapplication.services.EmployeeService.updateEmployee(EmployeeService.java:59) ~[classes/:na]

UpdateEmployee Function

public Employee updateEmployee(Long employeeId, Employee employee) {

Optional<Employee> employeeOldDetails = employeeRepository.findById(employeeId);

if (employeeOldDetails.isPresent()) {

Optional<Employee> employeeExistingWithEverestEmail = employeeRepository.findByEverestEmailId(employee.getEverestEmailId());

Optional<Employee> employeeExistingWithPersonalEmail = employeeRepository.findByPersonalEmailId(employee.getPersonalEmailId());

if (((employeeExistingWithEverestEmail.isPresent()) && (employeeExistingWithEverestEmail.get().getEmpId() != employeeOldDetails.get().getEmpId()))

|| ((employeeExistingWithPersonalEmail.isPresent()) && employeeExistingWithPersonalEmail.get().getEmpId() != employeeOldDetails.get().getEmpId())) {

throw new EmployeesDataNotFoundException("EverestEmail or PersonalEmail already exists");

} else {

employee.setEmpId(employeeOldDetails.get().getEmpId());

employee.getPresentAddress().setId(employeeOldDetails.get().getPresentAddress().getId());

employee.getPermanentAddress().setId(employeeOldDetails.get().getPermanentAddress().getId());

employeeRepository.save(employee);

return employee;

}

} else {

throw new EmployeesDataNotFoundException("Invalid email or password");

}

}

As present and permanent address ids are null,failed to update Employee.

If removed addresses,faced the issue of creating new addresses for employees instead of replacing(Duplication).

ISSUE:

1.Failed to design objects of Temporary address and Permanent address from the frontend side.  
2.If no address is given,failed to set the address Id for objects.