

## Project Description and Approach

1. **Manager Home Screen** – Altered table employee and skills, added **employee\_id** as **primary key** to connect skill, skilllock and employee table to generate query to get data,

```
ALTER TABLE `wfm`.`employees`
```

```
ADD PRIMARY KEY (`employee_id`);
```

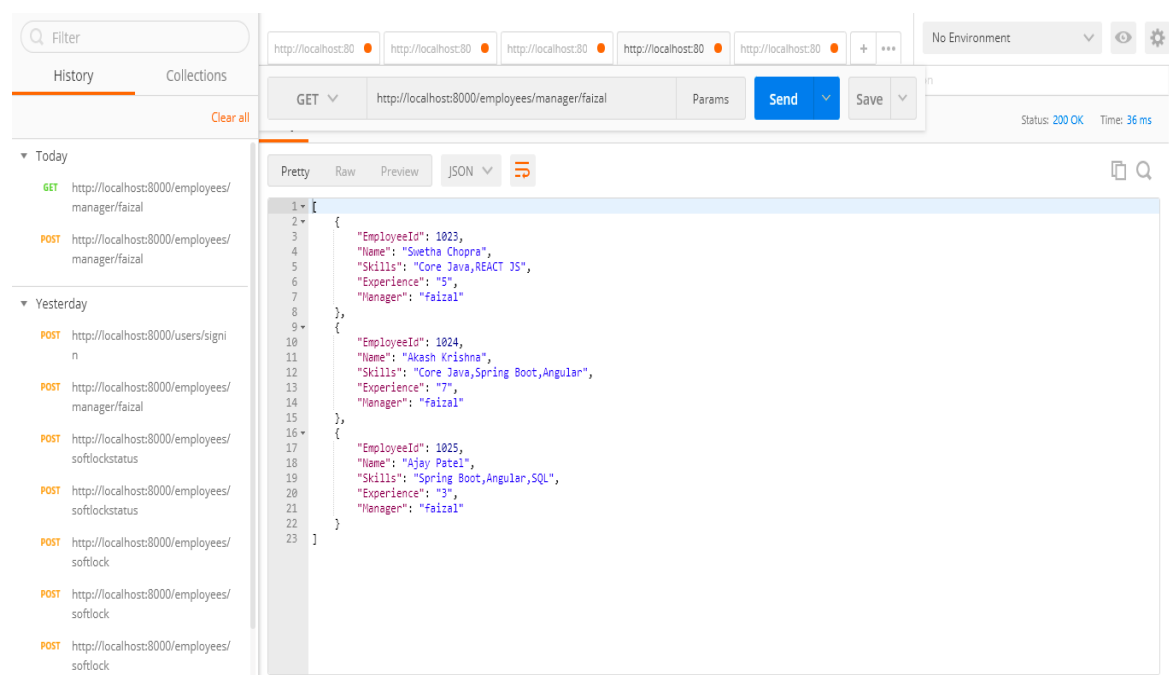
```
ALTER TABLE `wfm`.`skillmap`
```

```
RENAME TO `wfm`.`skillmaps`;
```

API – <http://localhost:8000/employees/manager/faizal>

{Added Authentication token in header} – GET request

### API Response Screenshot -



**Manager Home Screen Kendo Grid UI** – showing current logged In user data with pagination

#### Manager Home Screen

##### Managers Request Lock Table

Log out

Employee Id	Name	Skills	Experience	Manager	Request Lock
1023	Swetha Chopra	Core Java REACT JS	5	faizal	Request Lock
1024	Akash Krishna	Core Java Spring Boot Angular	7	faizal	Request Lock
1025	Ajay Patel	Spring Boot Angular SQL	3	faizal	Request Lock

1 - 3 of 3 items

2. **Manager Screen Request Lock Popup** - Clicking on request detail raising request for current user to lock resource

**Altered softlock table-**

```
ALTER TABLE `wfm`.`softlock`
```

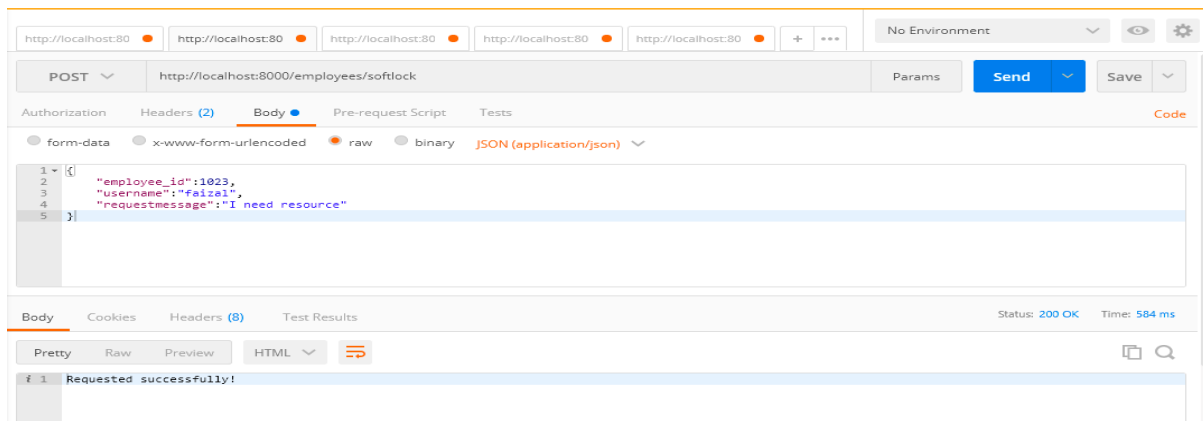
```
RENAME TO `wfm`.`softlocks` ;
```

**API** - <https://localhost:8000/employees/softlock> - Post request

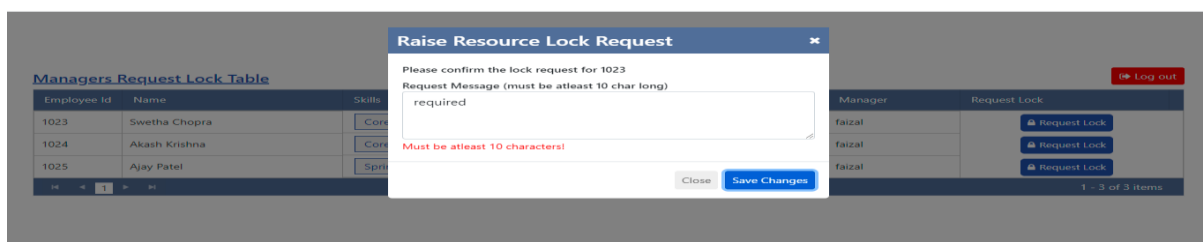
**Post Body-**

```
{  
  "employee_id":1023,  
  "username":"faizal",  
  "requestmessage":"I need resource"  
}
```

**API Response Screenshot –**



**Manager Lock Request Popup UI – with validation to have atleast 10 char**



## Manager Home Screen kendo Grid after Raising request-

[Manager Home Screen](#)

[Managers Request Lock Table](#) Log out

Employee Id	Name	Skills	Experience	Manager	Request Lock
1024	Akash Krishna	Core Java Spring Boot Angular	7	faizal	Request Lock
1025	Ajay Patel	Spring Boot Angular SQL	3	faizal	Request Lock

1 - 2 of 2 items

## 3.WFM Screen – Hold raised request to particular wfm manager

API -<http://localhost:8000/employees/wfm/lokesh> - GET request

## API Response Screenshot-

```
{
  "EmployeeId": 1017,
  "Manager": "rohit",
  "reqDate": "2020-05-27",
  "requestmessage": "I Want This Resource",
  "wfm_manager": "lokesh"
},
{
  "EmployeeId": 1019,
  "Manager": "rohit",
  "reqDate": "2021-12-06",
  "requestmessage": "Need this employee",
  "wfm_manager": "lokesh"
}
```

## WFM Home Screen React kendo grid UI-

[WFM Home Screen](#)

[Request Awaiting Approval](#) Log out

Employee Id	Requestee	Request Date	Employees Manager	Approve Request
1017	rohit	2020-05-27	lokesh	View Details
1019	rohit	2021-12-06	lokesh	View Details

1 - 2 of 2 items

#### 4.Approve Lock Request Popup –

API -<http://localhost:8000/employees/softlockstatus> POST request

POST Body –

```
{  
  "employee_id":1023,  
  "status":"accepted"  
}
```

#### API Response Screenshot-

The screenshot shows a REST client interface. The top bar indicates a POST request to `http://localhost:8000/employees/softlockstatus`. The 'Body' tab is selected, showing a JSON payload: `{ "employee_id":1023, "status":"accepted" }`. The response status is `200 OK` with a time of `272 ms`. The response body is displayed as `Requested status updated successfully!`.

#### WFM Request Lock Popup UI-

The screenshot displays the 'Soft Lock Request Confirmation' popup. The popup contains the following information:

- Status Update for Request Lock**
- Employee ID:** 1017
- Requestee:** rohit
- Employee Manager:** lokesh
- Request Description:** I Want This Resource
- Status:** Update (dropdown menu)

At the bottom of the popup are 'Close' and 'Save Changes' buttons. In the background, a table titled 'Request Awaiting Approval' is visible, showing two rows of employee data.

Employee Id	Requestee
1017	rohit
1019	rohit

## WFM Home Screen after Approving/Rejecting request-

### [WFM Home Screen](#)

#### [Request Awaiting Approval](#)

 Log out

Employee Id	Requestee	Request Date	Employees Manager	Approve Request
1019	rohit	2021-12-06	lokesh	<a href="#">View Details</a>

  **1**  

1 - 1 of 1 items