

Assignment Report On

EMPLOYEE APPRAISAL MANAGEMENT

for the course

IT254 : Web Technology and Applications

Submitted by

Kotla Karthik Reddy (181IT123)

IV SEM B.Tech (IT)

Under the guidance of

Mrs. Priyadarshini
Dept of IT, NITK Surathkal

in partial fulfillment for the award of the degree

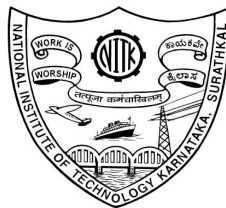
of

Bachelor of Technology

in

Information Technology

at



Department of Information Technology

National Institute of Technology Karnataka, Surathkal.

June 2020

Abstract

In computer programming, create, read, update, and delete(CRUD) are the four basic functions of persistent storage. Alternate words are sometimes used when defining the four basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of delete. CRUD is also sometimes used to describe user interface conventions that facilitate viewing, searching, and changing information; often using computer-based forms and reports. The term was likely first popularized by James Martin in his 1983 book Managing the Data-base Environment. The acronym may be extended to CRUDL to cover listing of large data sets which bring additional complexity such as pagination when the data sets are too large to be easily held in memory.

A single page application for employee appraisal management system using AngularJS, MongoDB as a database is developed. This is a simple MEAN stack application along with the implementation of some CRUD (Create, Read, Update, Delete) functionalities. The application developed is deployed into MVC architecture along with responsiveness provided. Using the developed employee appraisal management application, details of employees can be added by the head of particular departments in which the employee works along with the performance of the employee considered based on the performance in the previous tasks, update the details and delete them. Also, we can retrieve the list of all the employees we have stored in the database.

Problem Statement

1. Create a single page application for the employee appraisal management system using AngularJS, Database to be used MongoDB. Perform CRUD operations.
2. Create MVC architecture for the above-given application and make sure the application is responsive in nature.

Aim of the Assignment

To build a simple employee appraisal system with basic database operations of CRUD (Create, Read, Update and Delete).

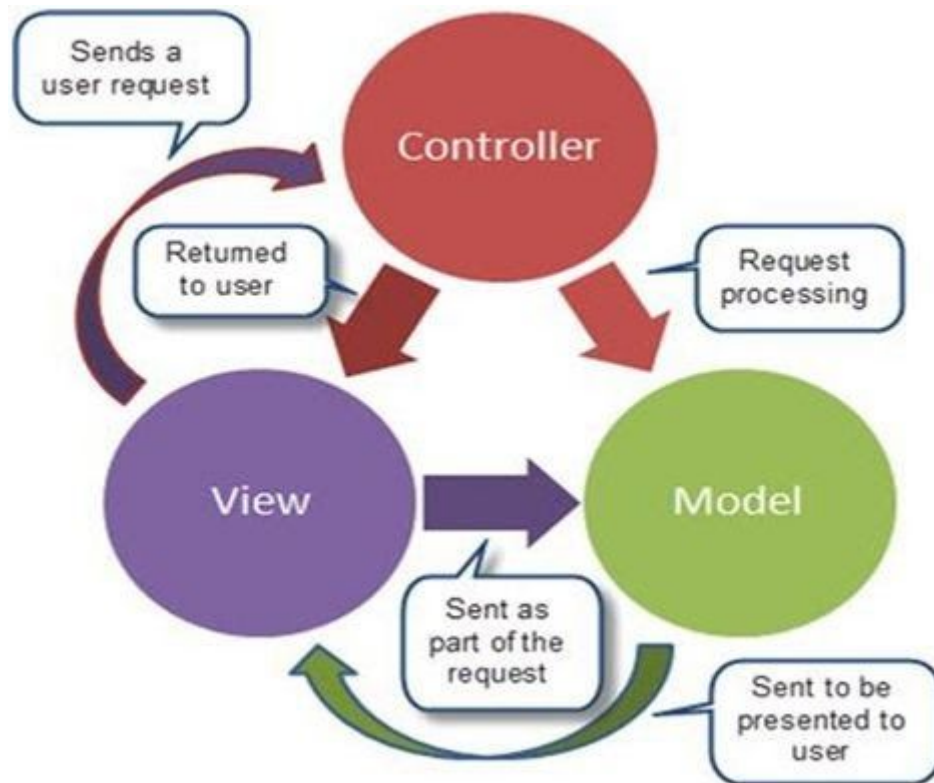
Tools and Technologies

- Angular6 - Frontend
- NodeJs - Backend
- Express - Routing library
- MongoDB - Database

Working

This app is to be used by Employers to maintain appraisal details. The employer can add a new employee, update the details of an existing employee, or delete an employee's data. Reviews can be given only to employees that exist in the database. These reviews can be updated or deleted.

Architecture



This project uses an MVC architecture which stands for Model, View and Control.

The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application.

Model

- The model is responsible for managing the data of the application.
- It responds to the request from the view and it also responds to instructions from the controller to update itself.

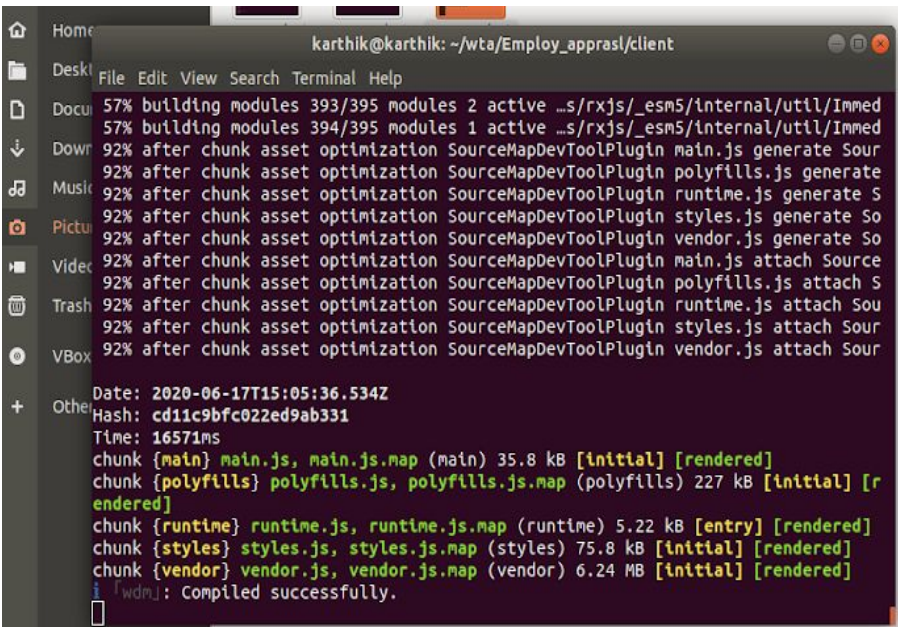
View

- It means presentation of data in a particular format, triggered by a controller's decision to present the data. It is used for all the UI logic of the application.

Controller

- The controller is responsible for responding to the user input and performing interactions on the data model objects.
- The controller receives the input, it validates the input and then performs the business operation that modifies the state of the data model.

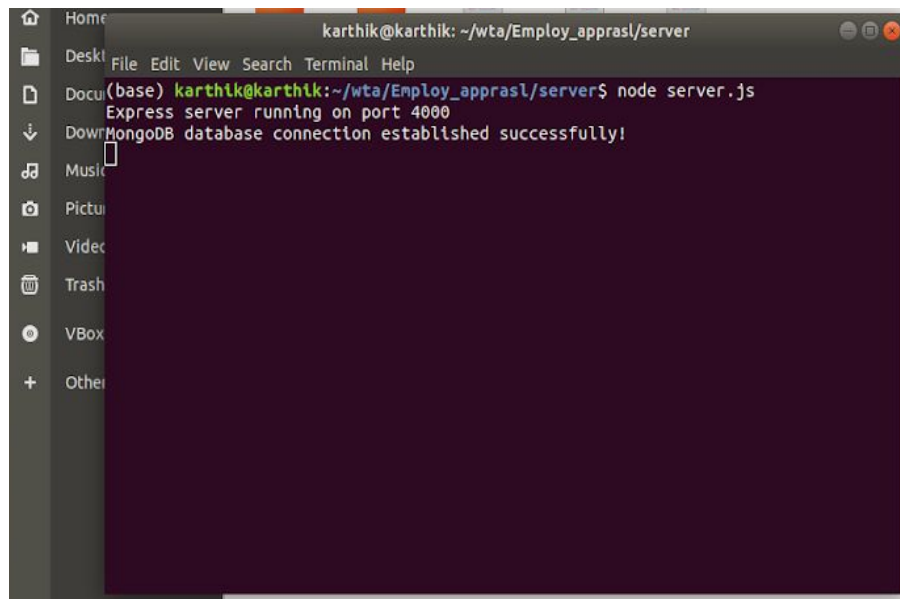
Screenshots



```
karthik@karthik: ~/wta/Employ_apprasl/client
File Edit View Search Terminal Help
57% building modules 393/395 modules 2 active ...s/rxjs/_esm5/internal/util/Immed
57% building modules 394/395 modules 1 active ...s/rxjs/_esm5/internal/util/Immed
92% after chunk asset optimization SourceMapDevToolPlugin main.js generate Sour
92% after chunk asset optimization SourceMapDevToolPlugin polyfills.js generate
92% after chunk asset optimization SourceMapDevToolPlugin runtime.js generate S
92% after chunk asset optimization SourceMapDevToolPlugin styles.js generate So
92% after chunk asset optimization SourceMapDevToolPlugin vendor.js generate So
92% after chunk asset optimization SourceMapDevToolPlugin main.js attach Source
92% after chunk asset optimization SourceMapDevToolPlugin polyfills.js attach S
92% after chunk asset optimization SourceMapDevToolPlugin runtime.js attach Sou
92% after chunk asset optimization SourceMapDevToolPlugin styles.js attach Sour
92% after chunk asset optimization SourceMapDevToolPlugin vendor.js attach Sour

Date: 2020-06-17T15:05:36.534Z
Hash: cd11c9bfc022ed9ab331
Time: 16571ms
chunk {main} main.js, main.js.map (main) 35.8 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 227 kB [initial] [r
endered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 5.22 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 75.8 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 6.24 MB [initial] [rendered]
[wdm]: Compiled successfully.
```

Fig 1: Compilation of client side

A terminal window titled 'karthik@karthik: ~/wta/Employ_apprasl/server'. The prompt is '(base) karthik@karthik:~/wta/Employ_apprasl/server\$'. The user has entered 'node server.js'. The output shows 'Express server running on port 4000' and 'MongoDB database connection established successfully!'.

```
karthik@karthik: ~/wta/Employ_apprasl/server
(base) karthik@karthik:~/wta/Employ_apprasl/server$ node server.js
Express server running on port 4000
MongoDB database connection established successfully!
```

Fig 2: Compilation of server side

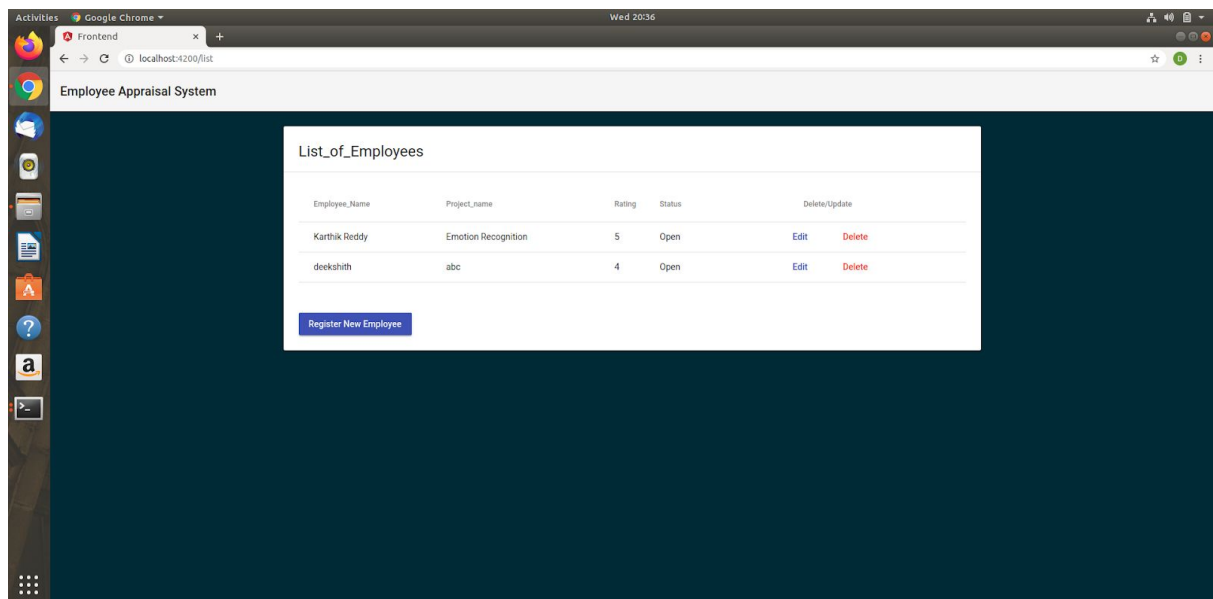


Fig 3: List of employees

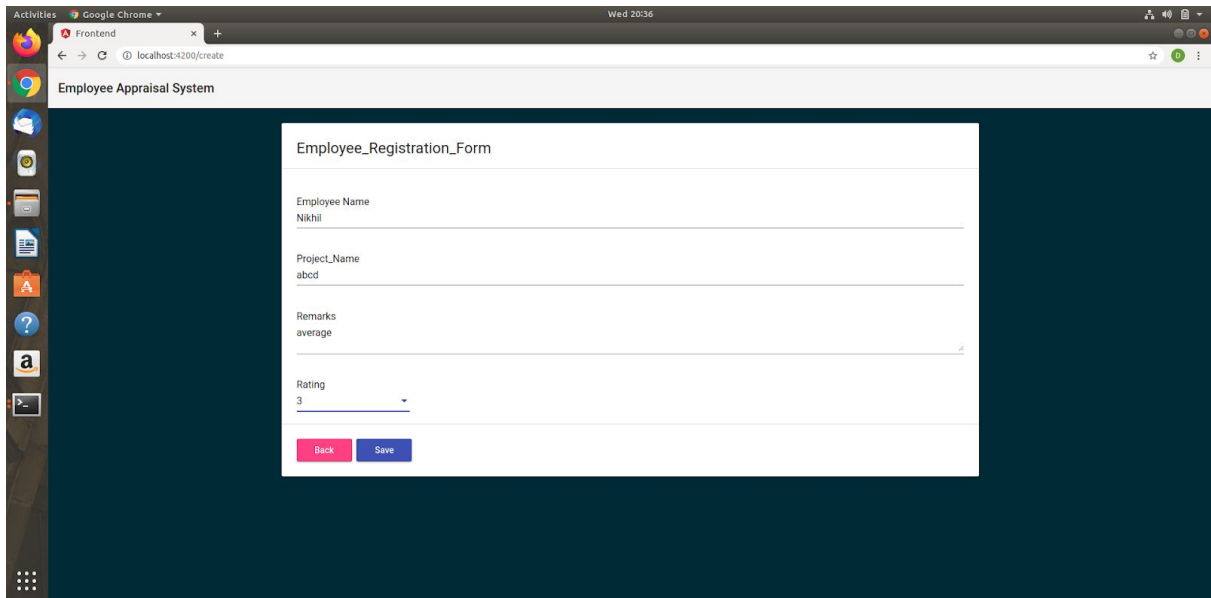


Fig 4: Adding new Employee

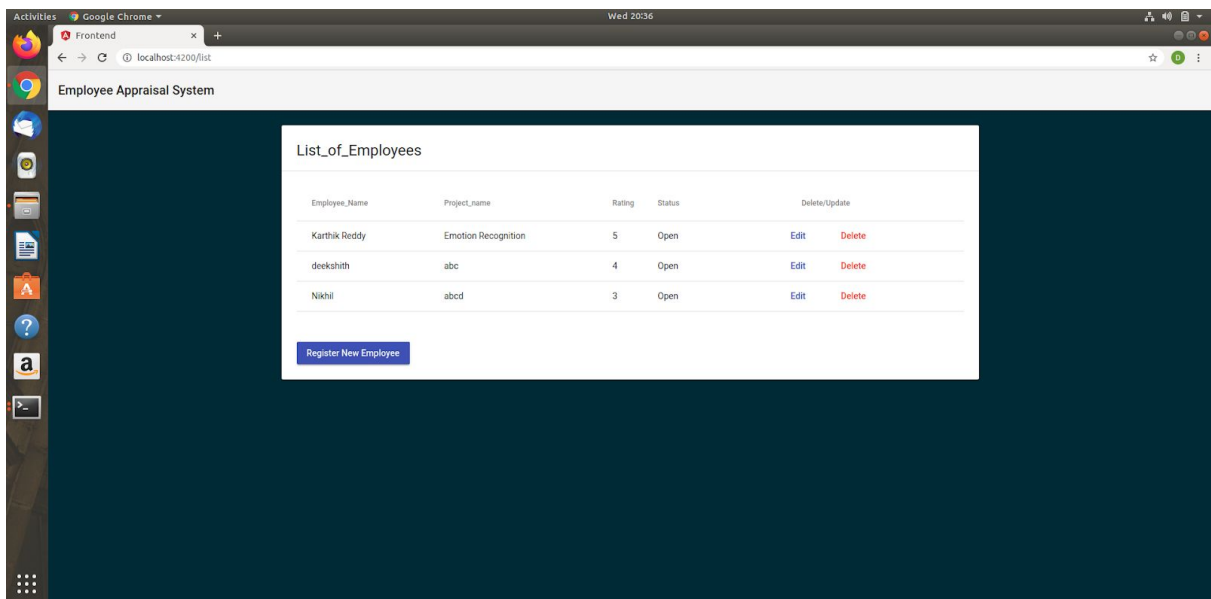


Fig 5: List of Employees after adding new Employee

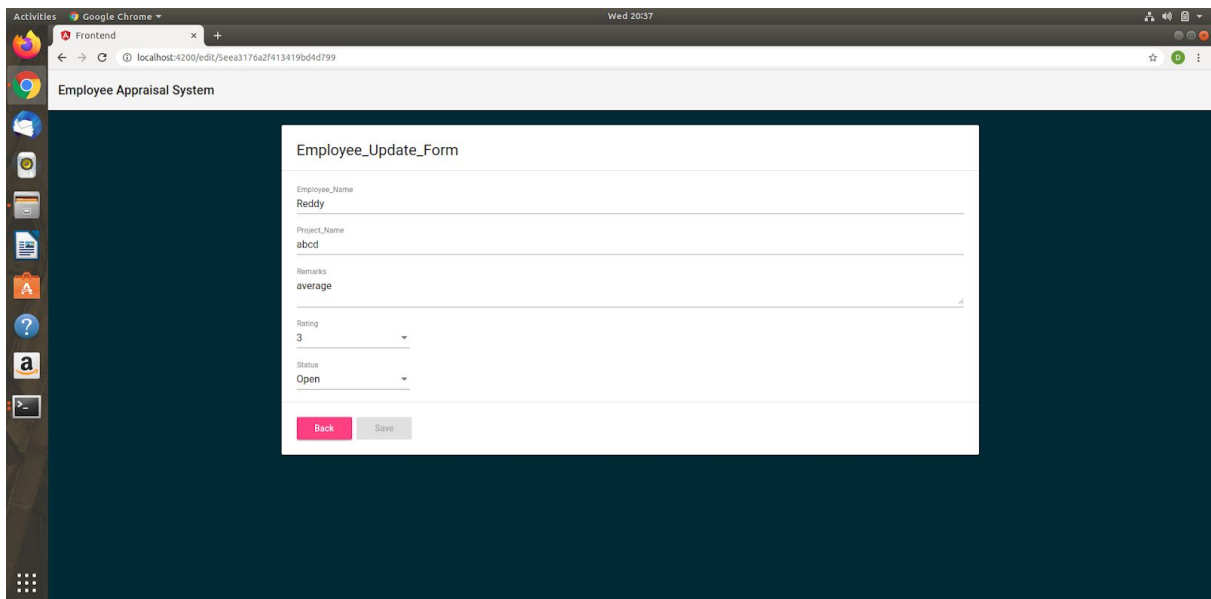


Fig 6: Updating of the Employee

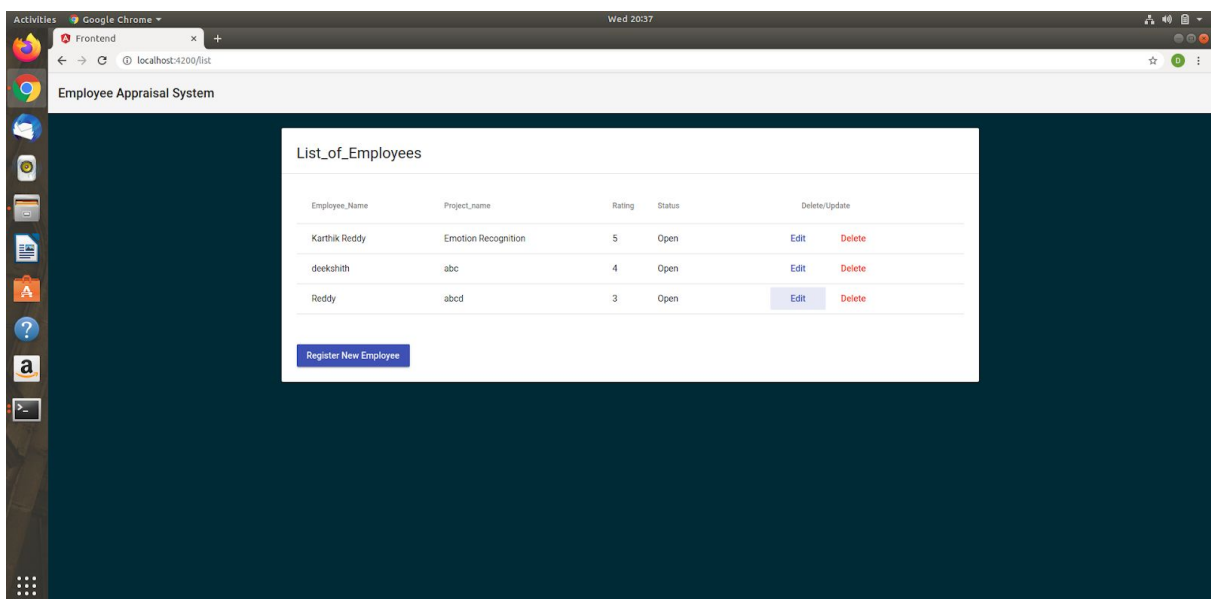


Fig 7: List after updating

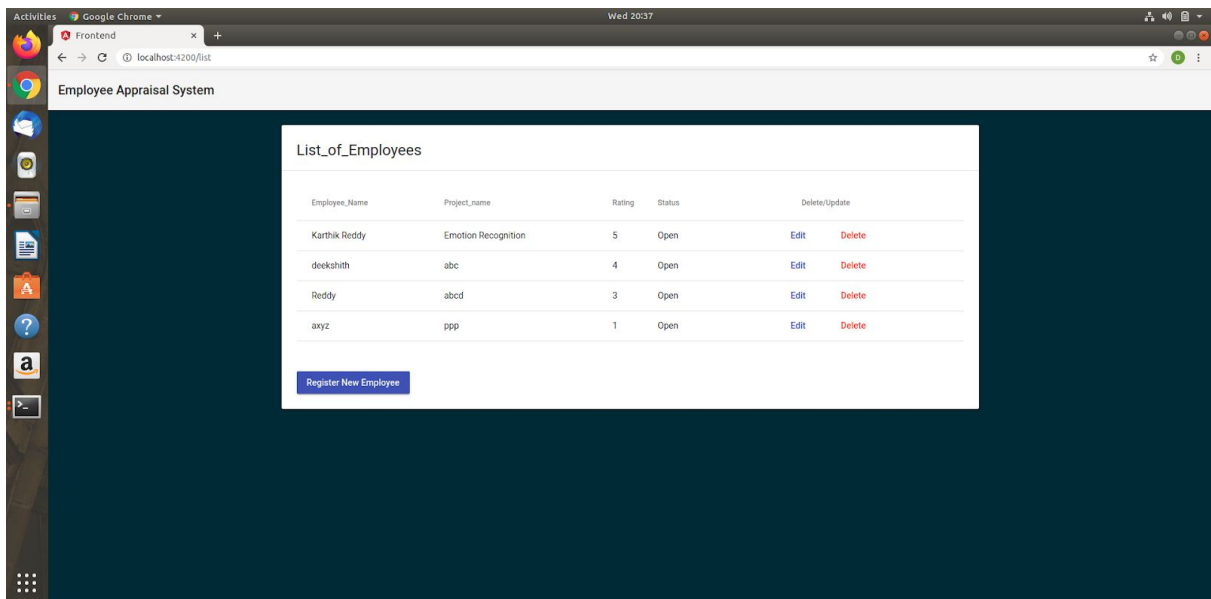


Fig 8: List after adding one more new Employee

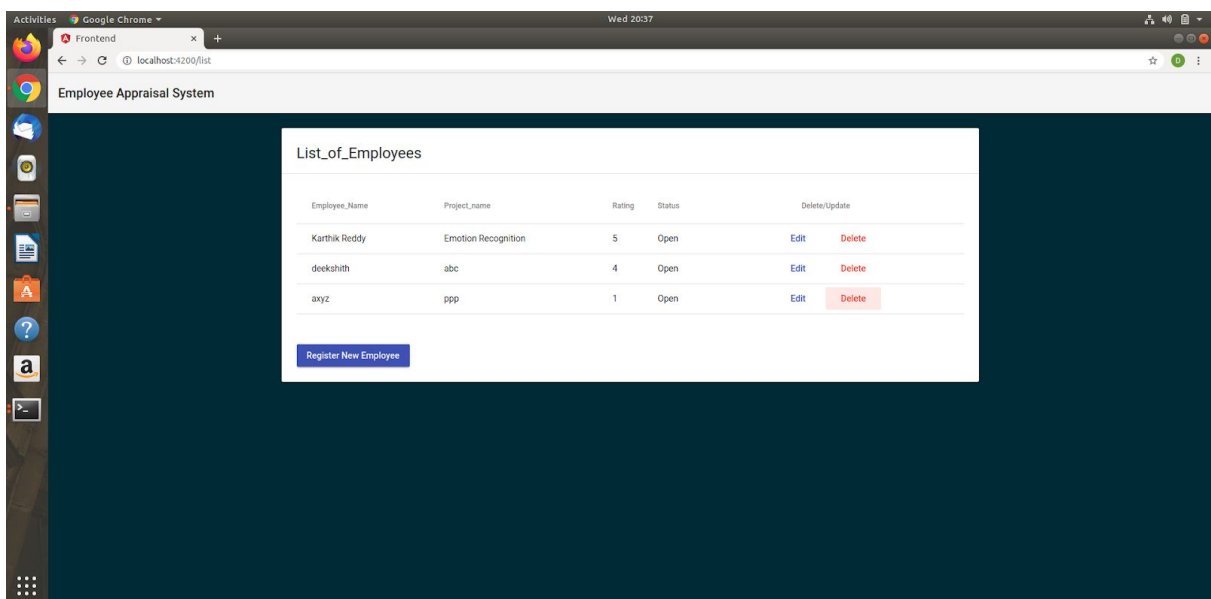


Fig 9: List after deleting an Employee

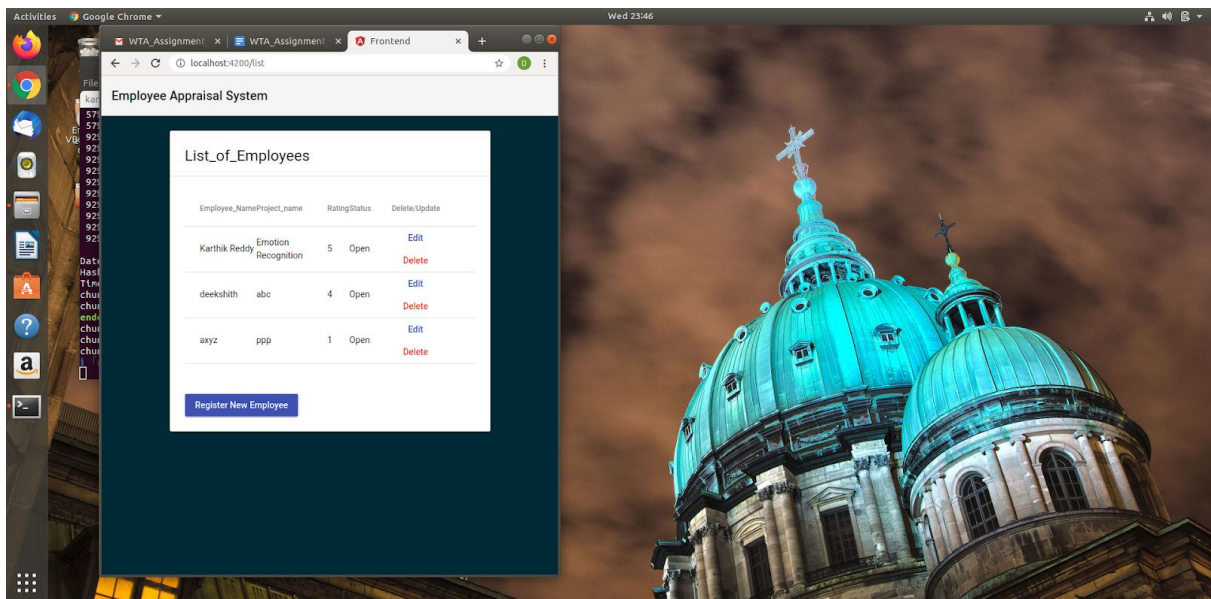


Fig 10: Responsiveness of Application

Github-Link :

https://github.com/karthikreddy1729/Employee_Appraisal_Form