

Fullstack Challenge

Loan management application

[Problem Statement]

Develop a simple application for loan management for clients of a bank.

You are expected to upgrade a Web app written in Angularjs, Angular, React and NodeJS stack, MongoDB should be used as the database

The source code for this application has been provided along with a description of the tasks needed to accomplish.

To run skeleton code use:

Angular 2:

ng serve

Angular 1:

grunt serve

React :

npm start

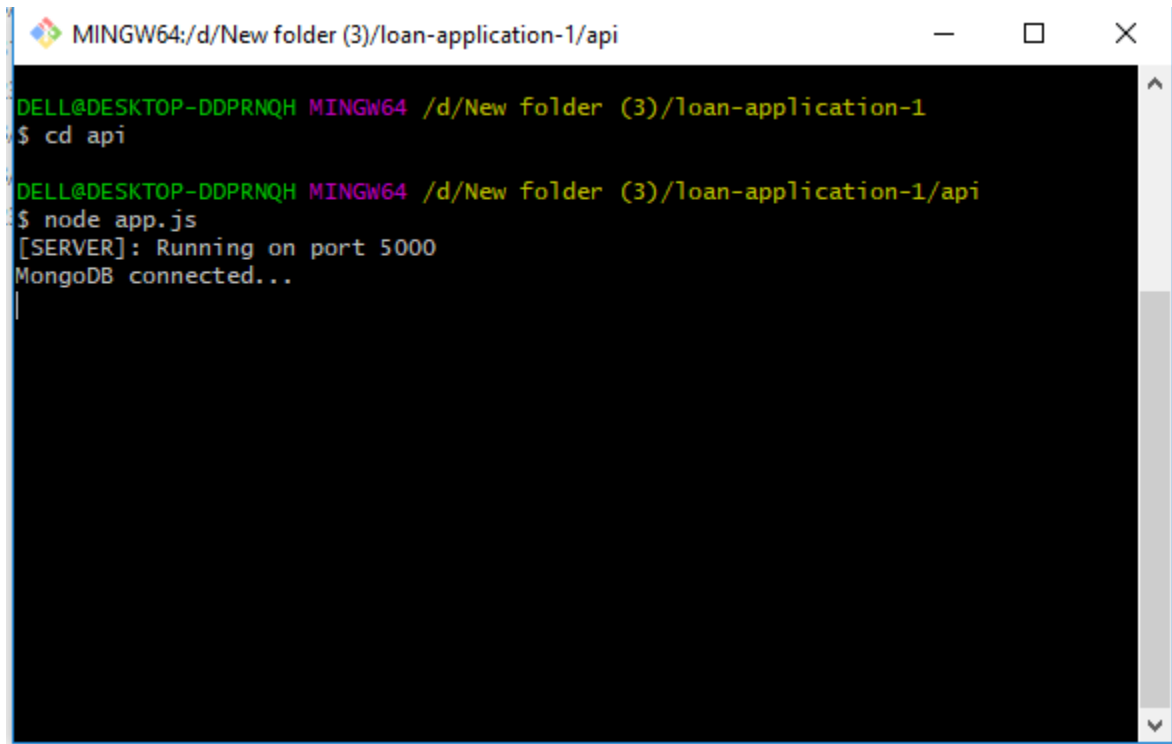
node [backend]:

Path :

projectName/api/app.js

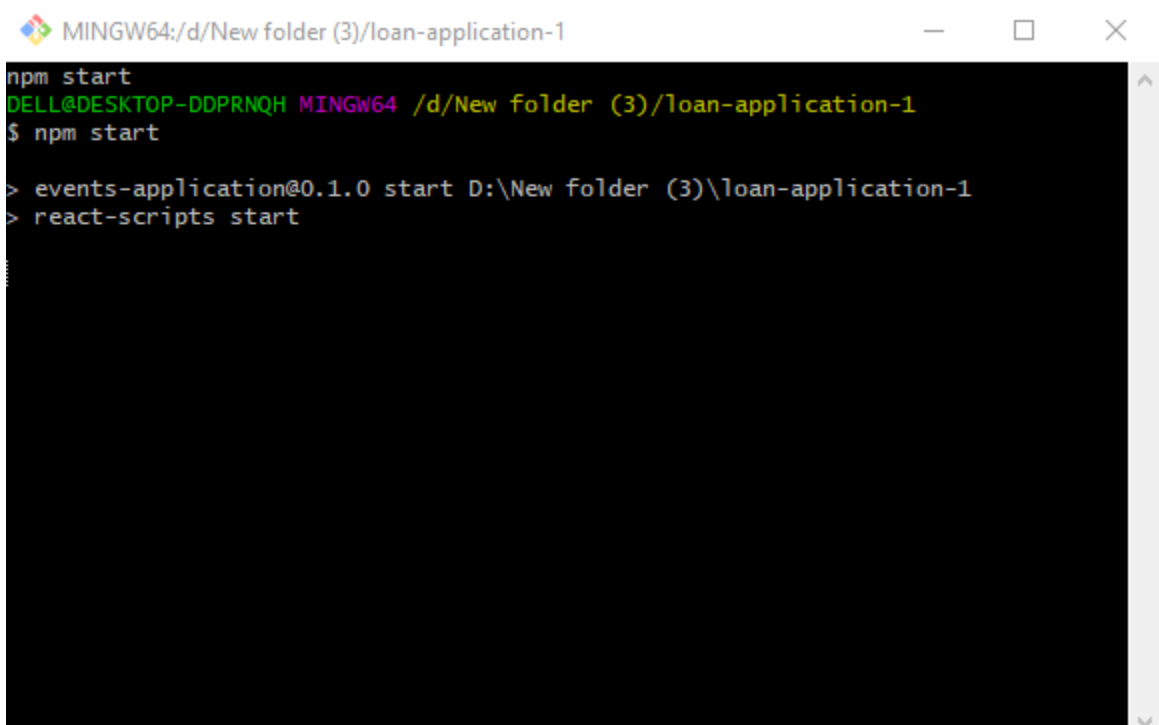
cmd : node app.js

Sample Screenshot :



```
MINGW64:/d/New folder (3)/loan-application-1/api
DELL@DESKTOP-DDPRNQH MINGW64 /d/New folder (3)/loan-application-1
$ cd api
DELL@DESKTOP-DDPRNQH MINGW64 /d/New folder (3)/loan-application-1/api
$ node app.js
[SERVER]: Running on port 5000
MongoDB connected...
```

“npm start ”



```
MINGW64:/d/New folder (3)/loan-application-1
npm start
DELL@DESKTOP-DDPRNQH MINGW64 /d/New folder (3)/loan-application-1
$ npm start
> events-application@0.1.0 start D:\New folder (3)\loan-application-1
> react-scripts start
```

****With the problem statement, we have also provided the skeleton of the solution which is incomplete. You are expected to understand the skeleton thoroughly and complete most of the tasks specified here.****

[Basic Tasks]

[Frontend Tasks]

- Ability to add a loan application
- View all add loan applications with their status
- Ability to Edit and Delete loan applications
- Add amount of loan requested. UI for the same needs to be created.

[Backend Tasks]

- ⌘ Build endpoint for creating a loan application
- ⌘ Proper REST API calls at (\add), (\update<id>) and (\delete<id>)
- ⌘ Pagination of 20 items. After reaching the end the application should make another call for the database for the next 20 items.
- ⌘ Implementation of Login/Logout functionality on the home screen, including user authentication. Either use Google API for authentication or create a manual login system and store password in hash.
- ⌘ Build an endpoint for delete a loan application based on ID
- ⌘ Create an onclick events that deletes the corresponding event.
- ⌘ Developing presentational components which are reusable

[Intermediate Tasks]

[Frontend Tasks]

- Ability to archive loan applications
- Ability to clear all denied loan applications
- Ability to bulk re-apply all refused loan applications (*Hint: This has to do with changing the status of all denied loan applications*)
- ___ Sort and filter based on date, category and alphabetical order
- ___ Developing presentational components which are reusable

- ___ Implement additional features like tags bookmarks ratings
- ___ Implementing bulk delete functionality
- ___ Provide timestamp for add/modify activities on the home page
- ___ Responsive web design for all the views
- ___ UI layer is responsive. Mobile UI should match that of mock provided. Provide a breakpoint at 960px
- ___ First name and last name should take string input with length between 3 and 24 characters. Loan amount should be numeric between 100 and 1 crore and number of installments should be numeric between 1 and 120.

[Backend Tasks]

- Build an endpoint for updating a loan application
- Build an endpoint for get a single loan application
- Use of git to manage source code. Create different branches for FrontEnd, Backend etc, Use of proper design patterns like Factory, Service, Singleton for all CRUD functionalities
- Proper validations should be given if forms are used(messaging/alerts etc.,)
- Proper unit tests and code coverage for Crud functionalities

[Advanced Tasks]

- Filter out loan applications based on name
- ___ Connecting the complete end to end flow through routing mechanism (which includes login /logout functionality)
- ___ Session and state management across the application
- ___ Provide UI test cases using Karma, Selenium etc.
 - Implementing a real time search
 - Acceptable scaffolding/asset management and code management by using gitlab

[Mockup References]:

Homepage

Loan Application Manager

Welcome to Loan Manager

Create Loan Applications

Clear Loan Applications

Archive Applications

type here...

Search

There are no applications.

Create Loan Application

Loan Application Manager

Welcome to Loan Manager

Re-apply Applications

Clear Loan Applications

Create Loan Application

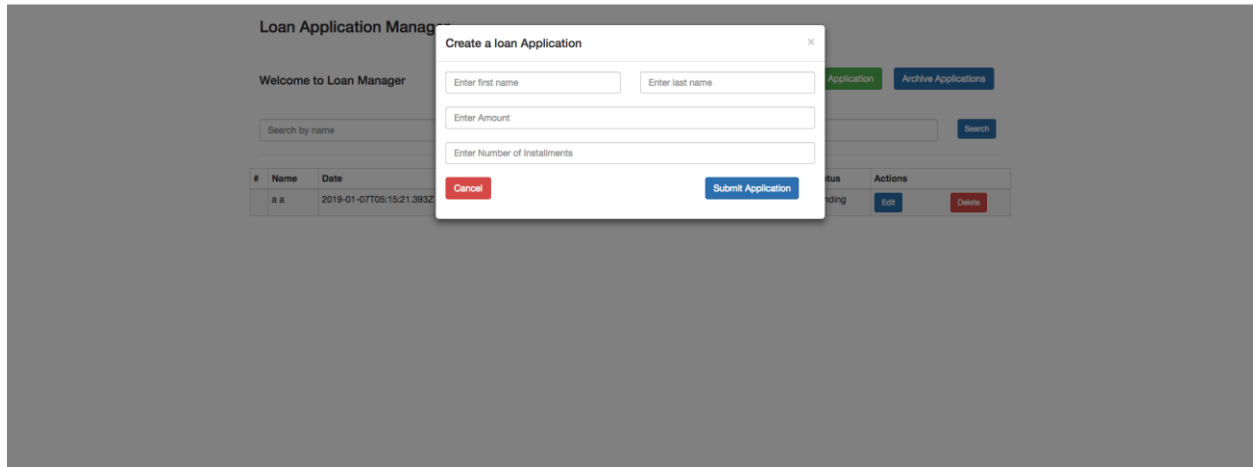
Archive Applications

Search by name

Search

#	Name	Date	Amount Requested	Number of Installements	Status	Actions
---	------	------	------------------	-------------------------	--------	---------

On making a request on ("add")



On making as request at ("update\<id>")



Recommended Mockup

Web Mockup should be as follows:

Loan Application Manager

Welcome to Loan Manager

Create Loan Applications

Clear Loan Applications

Archive Applications

Search

id:18022901

Name:Jon Snow

Date:28/2/2018

Amount Required:1,00,000

No. of installments:16

Status:Accepted

Update

id:18022902

Name:Peter Parker

Date:28/2/2018

Amount Required:2,00,000

No. of installments:16

Status:Accepted

Update

id:18022903

Name:Cindy R.

Date:28/2/2018

Amount Required:1,50,000

No. of installments:16

Status:Accepted

Update

id:18022904

Name:Bruce Banner

Date:28/2/2018

Amount Required:12,00,000

No. of installments:16

Status:Accepted

id:18022905

Name:Darth Vader

Date:28/2/2018

Amount Required:21,00,000

No. of installments:16

Status:Accepted

id:18022901

Name:Sheldon S.

Date:28/2/2018

Amount Required:3,00,000

No. of installments:16

Status:Accepted

Mobile UI should be as follows:

Loan Application Manager

Welcome to Loan Manager

Create Loan Applications

Clear Loan Applications

Archive Applications

Search

id:18022901

Name:Jon Snow

Date:28/2/2018

Amount required:1,00,000

No. of installments:16

Status:Accepted

Update

id:18022902

Name:Peter Parker

Date:28/2/2018

Amount required:2,00,000

[Accepted Solution]

By the end you should have an application with the following features for both frontend and backend:

- Add a loan application
- Edit/Delete a loan application
- View all added loan applications
- Archive loan applications
- Clear loan applications
- Bulk reply for loan applications
- Basic search by name of loan applications

[Solution Approach]

- All the base code has been written, you just need to add in logic for both frontend and backend as per the aforementioned tasks.
- You don't need any css or extra html

[Guide]

Code Skeleton:

TechStack :

Frontend: Angular, Angularjs, Reactjs.

Backend: Mongo, Express, Nodejs