# Dharmsinh Desai University, Nadiad

# Faculty of Technology,

# Department of Computer Engineering

# Tech. CE Semester – VI

# Subject: System design practice

# Project title : Personal digital storage and secure sharing

By

1. **Bhaumik joshi , roll no: CE-50 , Id: 17ceuos030**
2. **Adit modi, roll no: CE-70, Id: 17ceubs097**
3. **Smit panchal , roll no : CE-75, Id:17ceubs054**

**Guided by : Prof. Jigar M Pandya**

Contents

1.Abstract 4

2.Introduction 4

3.Software Requirement Specifications 4

4.Design

I.Use case diagrams 13

II. Class diagram 15

III. Sequence Diagrams 16

IV. Activity Diagrams 17

V. State Diagrams 21

VI. ER Diagram 23

VII. Data-Dictionary 24

5.Implementation Detail 24

6.Testing 25

7.Screen-Shots 26

8.Conclusion 28

9.Limitation and Future extension 28

10.Bibliography 28

**DHARMSINH DESAI UNIVERSITY**

**NADIAD-387001, GUJARAT**



**CERTIFICATE**

This is to certify that the project entitled “**Personal digital storage and secure sharing**” is a bonafied report of the work carried out by

1) **Mr.Bhaumik Joshi,** Student ID No: **17CEUOS030**

2) **Mr. Adit Modi,** Student ID No:**17CEUBS097**

3) **Mr. Smit Panchal,** Student ID No:**17CEUBS054**

of Department of Computer Engineering, semester VI, under the guidance and supervision for the subject System design practice. They were involved in Project training during academic year April, 2020.

|  |  |
| --- | --- |
| Prof. Jigar M Pandya  (Project Guide)  Department of Computer Engineering,  Faculty of Technology,  Dharmsinh Desai University, Nadiad  Date: | Dr. C.K Bhensdadia  Head of the Department  Department of Computer  Engineering,  Faculty of Technology,  Dharmsinh Desai University,  Nadiad  Date: |
|  |  |

Abstract

Data is an important asset in any business, and for some companies it’s one of the most important assets. The way you store your data plays a major role in how easy it is to access, use, and keep secure. What if there are some crucial files of the company that are in the PC or Laptop Or my pen drive.

Amount of data growing everyday is quite huge, One reason of such tremendous data growth is duplication of same data on various local devices.

So the general problem is : What if we need a platform for **digital storage** which is **secure, cost efficient, can be shared** to people you want to and easily accessible , basically All these features in one.

.

Introduction

Our project is mainly focused on storage of data and secure sharing. Storage of data in an efficient manner in very similar way of any OS that gives ease of access.

A Personal Storage for Clients where any files, folders, zip files can be stored and even it can be your private vault to keep secret data. As well as easy and secure sharing of data.

Following is list of all technologies , services and platform used for development of our project.

Front end:

* Angular 8

Backend :

* NodeJs
* Express JS
* Mongo Atlas
* Heroku hosting service
* AWS S3
* AWS cognito
* AWS amplify
* AWS cloudfront
* AWS lexbot

Along with above list , Git for versioning and VS code editor for codebase editor.

Software requirement specifications

**1) User**

**R1) Login**

User should be able to login to their personal account.

Input: username and password

Output: On valid credentials user is prompted to system.

**R2) Sign up**

User should be able to create their personal account.

Input : Personal details

Output : After verification, user account is created.

**R3) Upload files**

Registered users should be able to upload single or multiple files via selection or drag and drop.

Input : Files selection/drag and drop

Output: Files gets uploaded to users’ personal storage.

**R4) Share files**

Registered users should be able to share file via various options.

**R4.1) Share files via link**

Input: user command

Output: File link is copied to clipboard.

**R4.2) Share files via qr-code**

Input: user command

Output: Qr-code of a file is generated and displayed.

**R5) Download file**

Registered users should be able to download any file of their personal storage.

Input: User selection

Output: Selected file gets downloaded to client machine.

**R6) Delete file**

Registered users should be able to delete any file stored in their personal storage space.

Input: User command

Output: File is removed from user storage space.

**R7) Upload folder**

Registered users should be able to upload folders from their local machine to storage space.

Input: Folder selection

Output: Folder is uploaded.

**R8) Traverse folders**

Registered users should be able to traverse all folders and sub folders in their personal storage.

Input: User command

Output: Current directory and all folders and files inside traversed directory is displayed.

**R9) Create folder**

Registered users should be able to create folders and sub-folders inside root directory of their storage space.

Input: Folder name

Output: Folder is created inside current traversed directory

**R10) Delete folders**

Registered users should be able to delete any folder other than their root directory.

Input: User selection

Output: Folder is removed

**R11) Display profile**

Registered users should be able to view their profile info such as email , contact number, used space , free space etc.

**R12) Edit profile**

Registered users should be able to edit their profile information such as contact number , address, profile picture etc.

**R13) Set alerts**

Registered users should be able to set alert to be send to their registered email when some amount of storage is used.

Input: Storage limit

Output: On exceeding storage limit alert mail is sent.

**R14) Log-out**

**R15) Display storage statistics**

Registered users should be able to show basic statistics such chart of their data usage.

**2) Premium user**

Along with above listed functionalities, a premium user should have following additional functionalities.

**R16) Share folders**

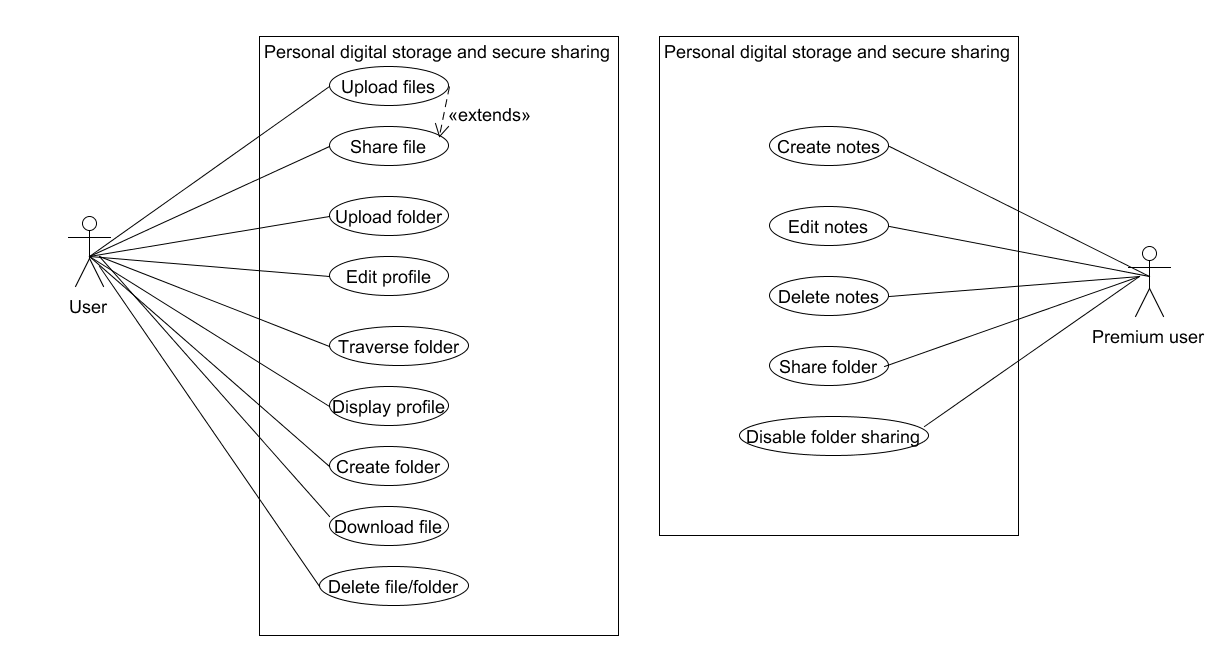
A premium user should be able to share his/her whole folder maintaining its sub directory structure securely between multiple users without data duplication.

**R17) Create notes**

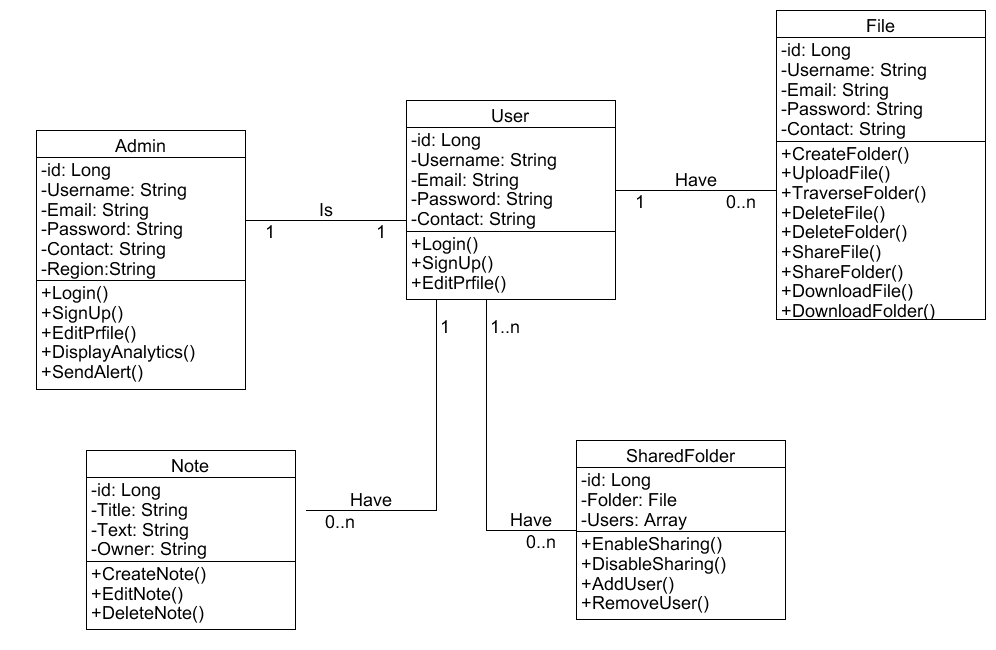
A premium user should be able to create sticky notes directly in their personal space as well as edit it in editor.

Design

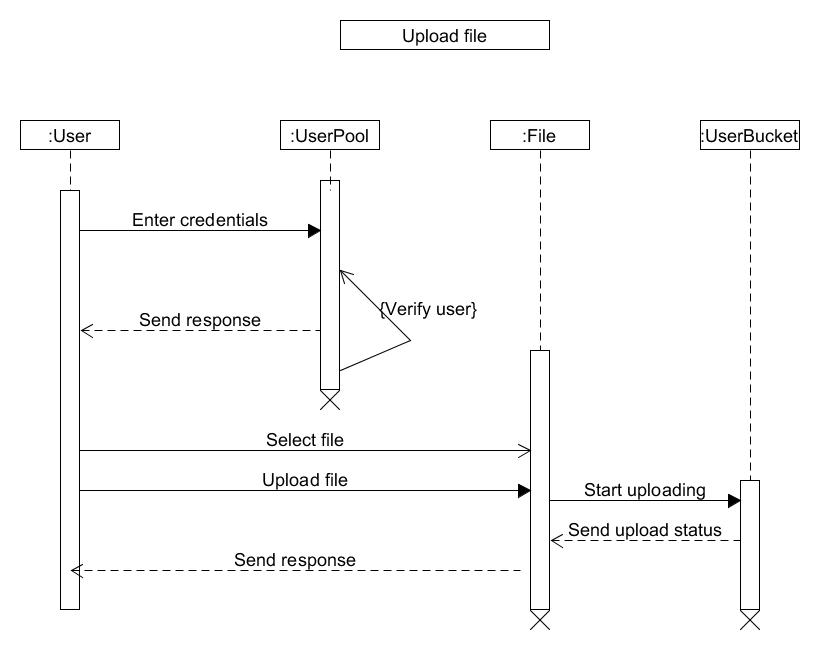
# Use case diagrams



# Class diagram



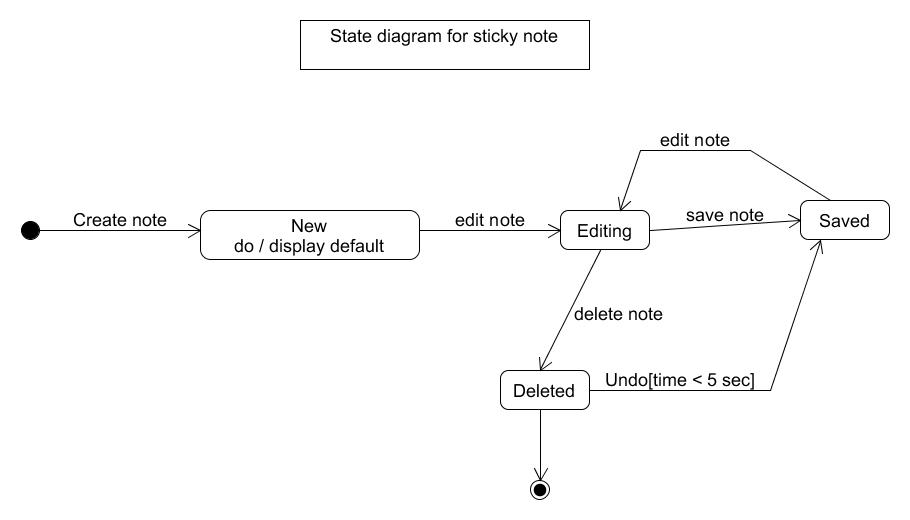
# Sequence diagrams



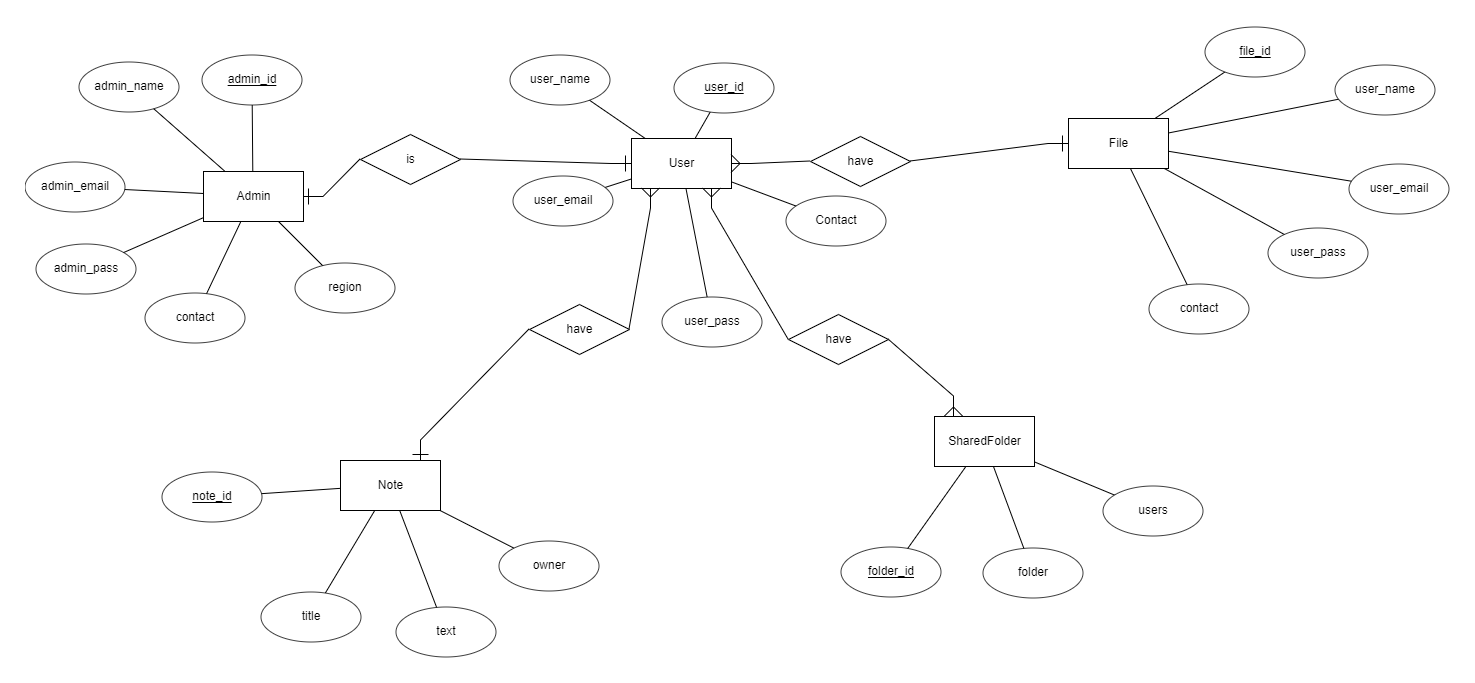
# Activity diagrams

# C:\Users\joshi\Downloads\SAVE_20200424_225920.jpg

# State diagrams



# ER diagram



# Data dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **User** | | | | | |
| **Column Name** | **Data Type** | **Required** | **Unique** | **Pk/Fk** | **Reference Table** |
| user\_id | Long | Yes | Yes | Primary key | - |
| user\_name | Varchar(20) | Yes | No | - | - |
| user\_email | Varchar(20) | Yes | No | - | - |
| user\_pass | Varchar(8) | Yes | No | - | - |
| Contact | Varchar(10) | No | No | - | - |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Admin** | | | | | |
| **Column Name** | **Data Type** | **Required** | **Unique** | **Pk/Fk** | **Reference Table** |
| admin\_id | Long | Yes | Yes | Primary key | - |
| admin\_name | Varchar(20) | Yes | No | - | - |
| admin\_email | Varchar(20) | Yes | No | - | - |
| admin\_pass | Varchar(8) | Yes | No | - | - |
| Contact | Varchar(10) | No | No | - | - |
| Region | Varchar(20) | No | No | - | - |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **File** | | | | | |
| **Column Name** | **Data Type** | **Required** | **Unique** | **Pk/Fk** | **Reference Table** |
| file\_id | Long | Yes | Yes | Primary key | - |
| user\_name | Varchar(20) | Yes | No | - | - |
| user\_email | Varchar(20) | Yes | No | - | - |
| user\_pass | Varchar(8) | Yes | No | - | - |
| Contact | Varchar(10) | No | No | - | - |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Note** | | | | | |
| **Column Name** | **Data Type** | **Required** | **Unique** | **Pk/Fk** | **Reference Table** |
| note\_id | Long | Yes | Yes | Primary key | - |
| title | Varchar(20) | Yes | No | - | - |
| text | Varchar(20) | Yes | No | - | - |
| owner | Varchar(8) | Yes | No | - | - |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SharedFolder** | | | | | |
| **Column Name** | **Data Type** | **Required** | **Unique** | **Pk/Fk** | **Reference Table** |
| folder\_id | Long | Yes | Yes | Primary key | - |
| folder | Varchar(20) | Yes | No | - | - |
| users | Varchar(20) | Yes | No | - | - |
|  |  |  |  |  |  |

Implementation detail

Following is brief description of some of important components and services of the system.

## File upload service

This is the most important service of system which is responsible for interacting with user files stored in AWS S3 bucket. This service contains various methods for fetching all user files, uploading multiple files, deleting files etc.

## Form Upload component

This is very important component of system. This component provides UI for uploading file or folder and renders list of user files and folders with help of its child component.

## List Upload component

It is child component of file upload component. This component along with its child details component jointly handles user files and folder UI for accessing and sharing files and traversing folders.

## Notes Service

This service interacts with node js API for handling requests of sticky notes created by user that are stored in mongo db.

## Shared service

This service interacts with Node Api that handles requests related to sharing of folders between multiple users.

## Storage service

This service is used to calculate statistics of user data and keep track of storage usage.

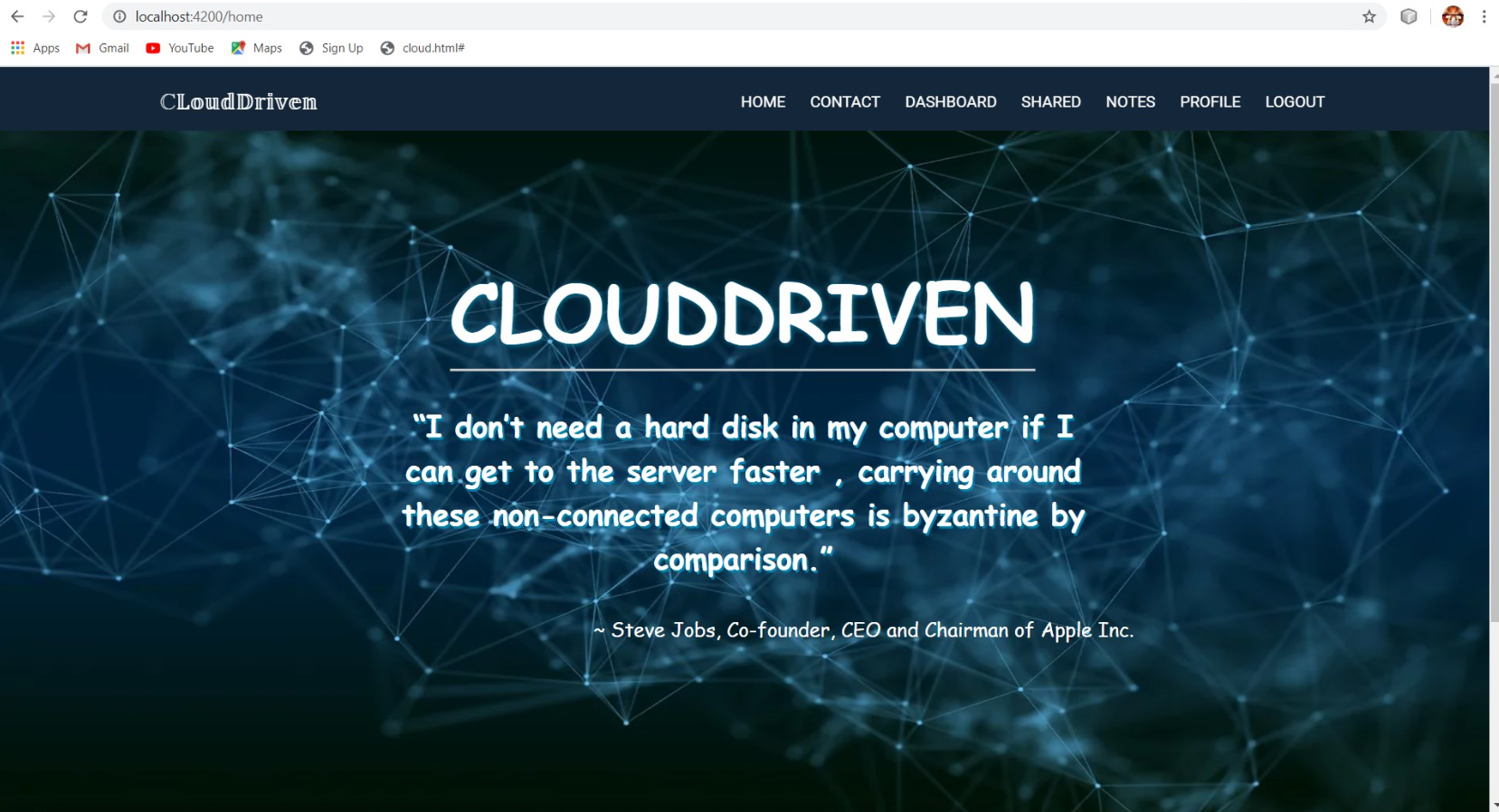
Testing

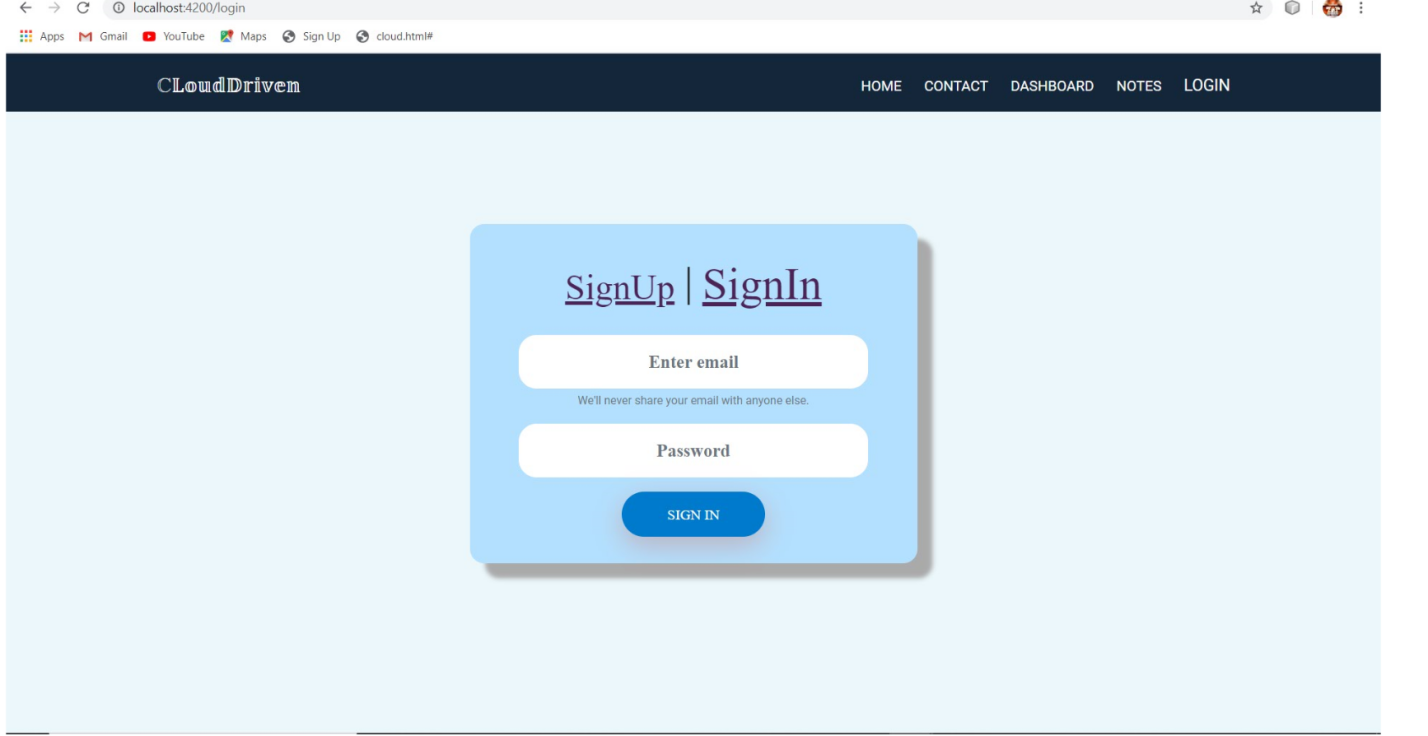
Unit testing is done on individual components of the system.

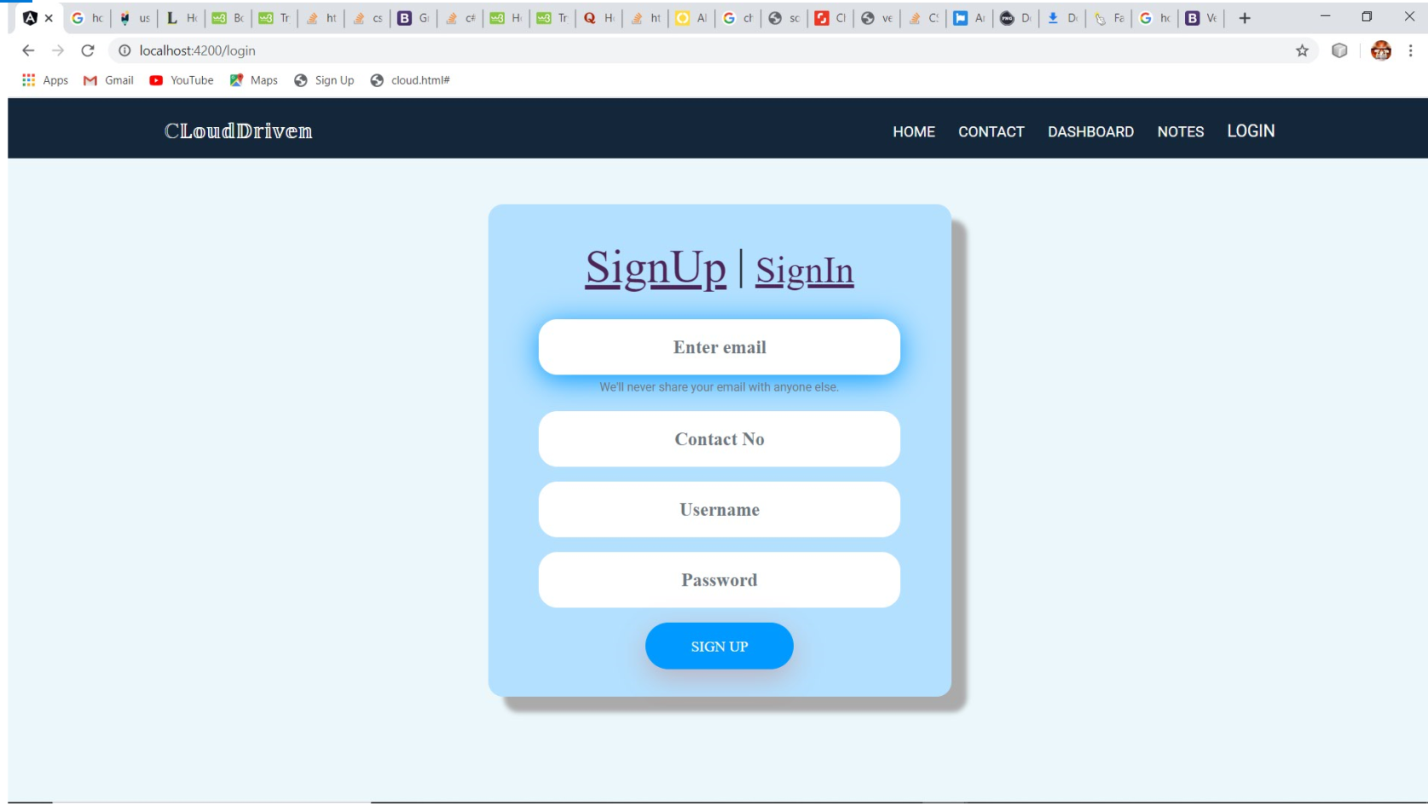
Below is example of login test case

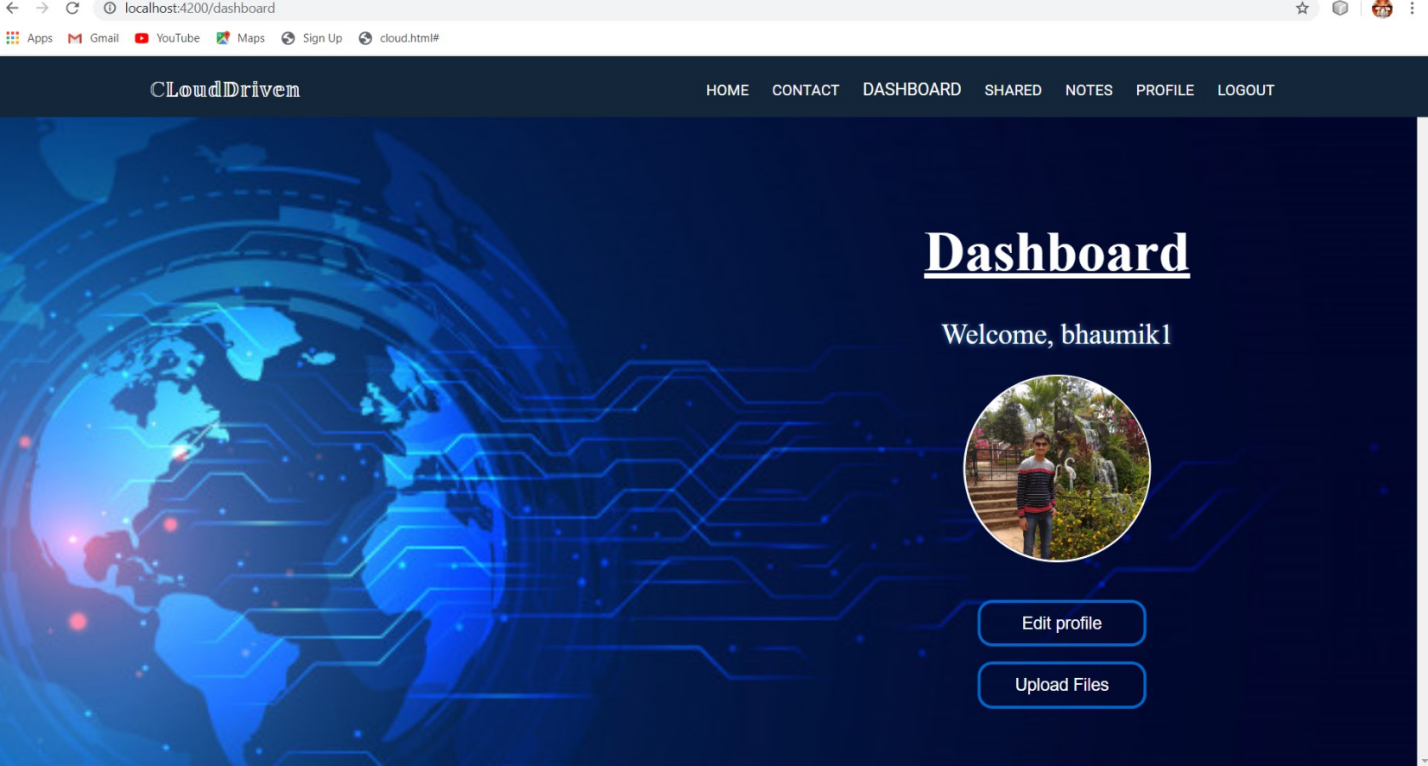
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Id | Test scenario | Test steps | Test data | Expected results | Actual results | Pass/Fail |
| 1  2 | Check user login with valid data  Check user login with invalid data | 1.Open login page  2.Enter email and password  3.Cick on login button  1.Open login page  2.Enter email and password  3.Cick on login button | [Email=admin@admin.com](mailto:Email=admin@admin.com)  Password=admin123  [Email=admin@admin.com](mailto:Email=admin@admin.com)  Password=admin456 | User should login into application  User should not be able to login and get invalid credential message | As expected  As expected | Pass  Pass |
|  |  |  |  |  |  |  |

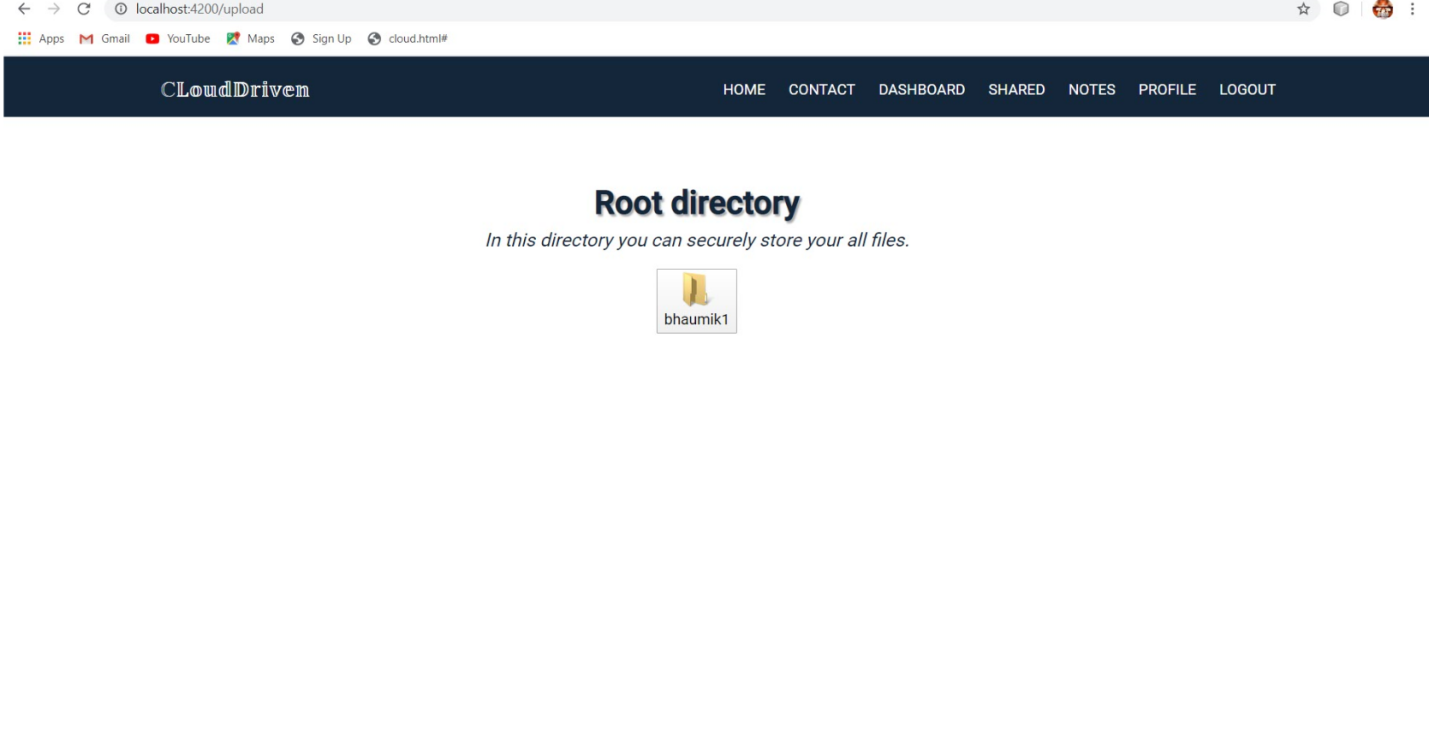
Screen shots

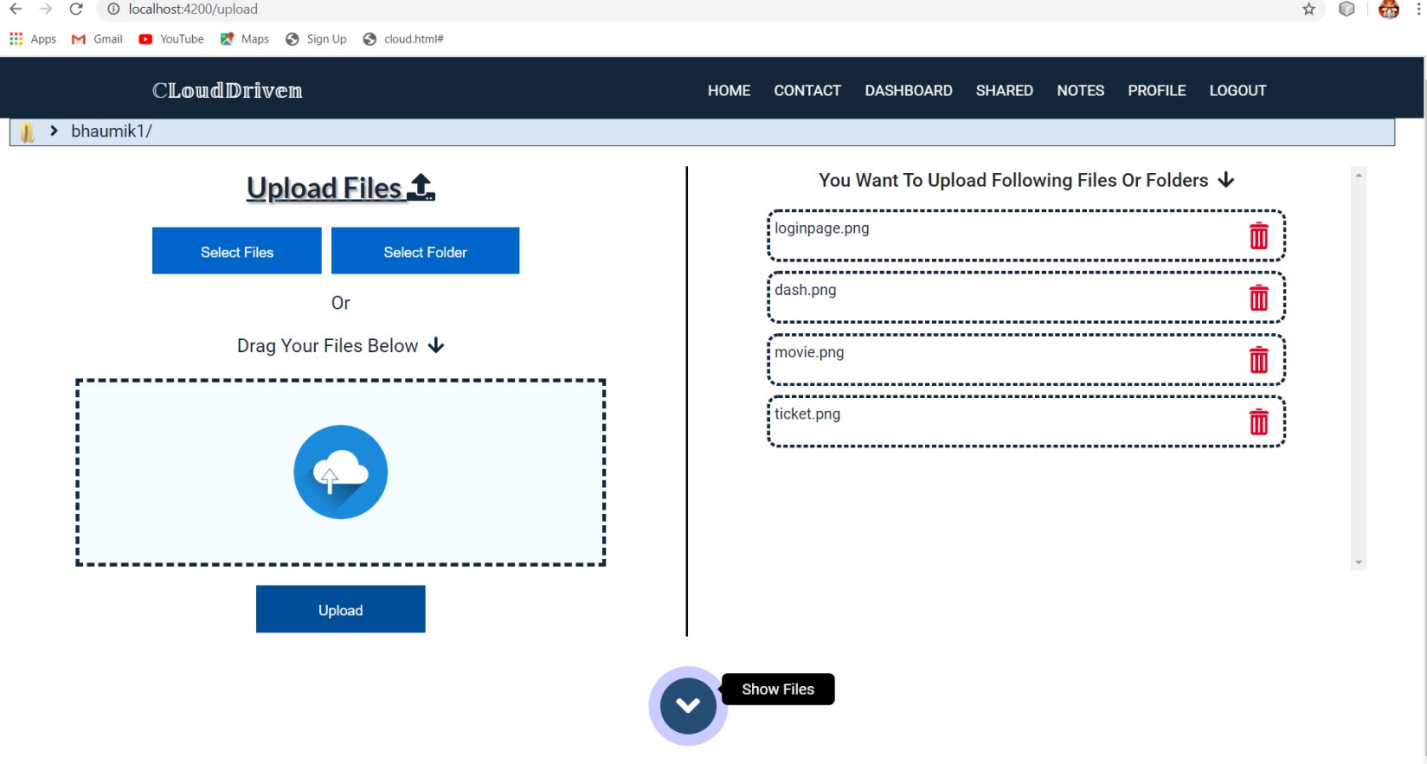


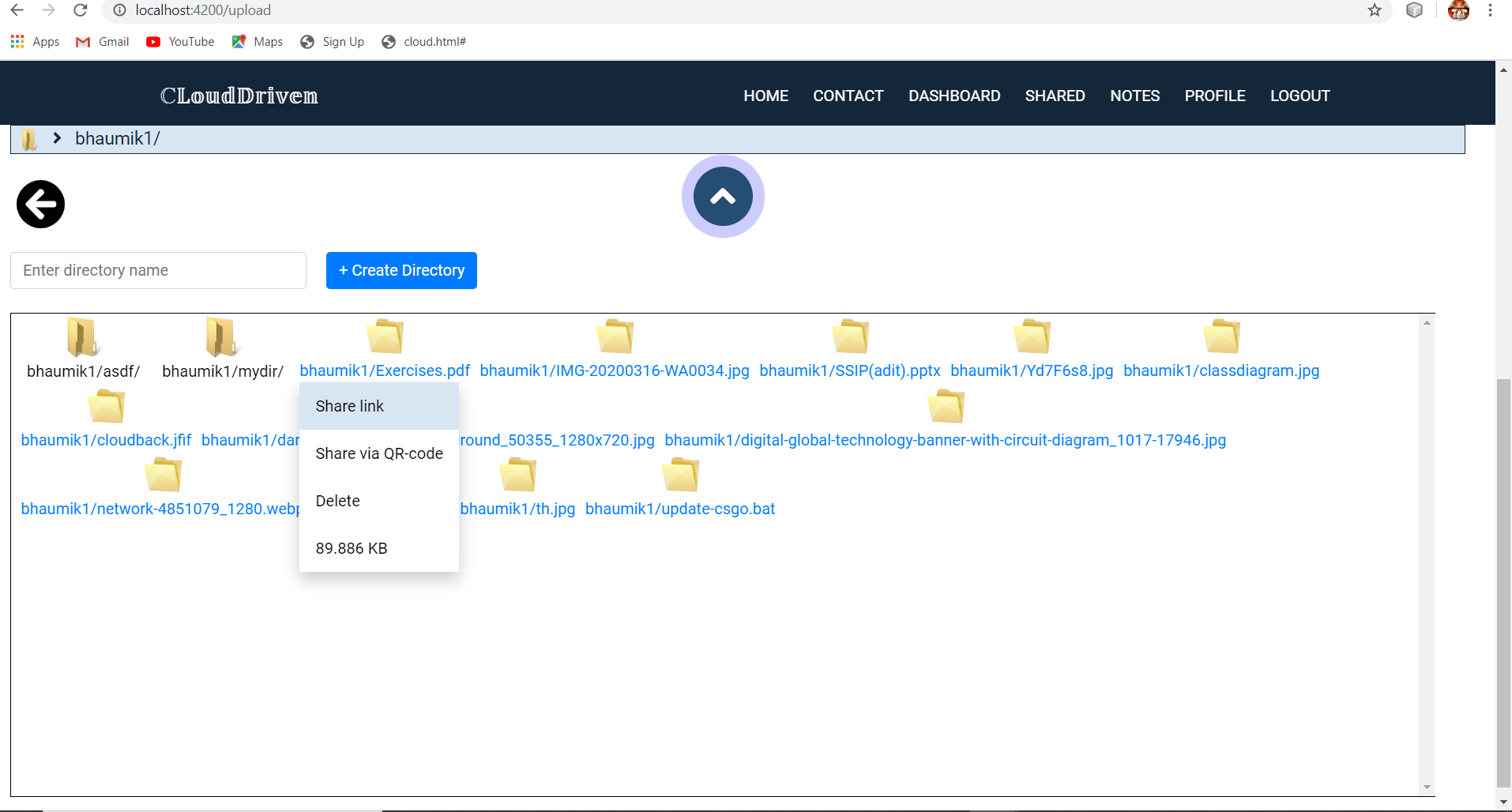


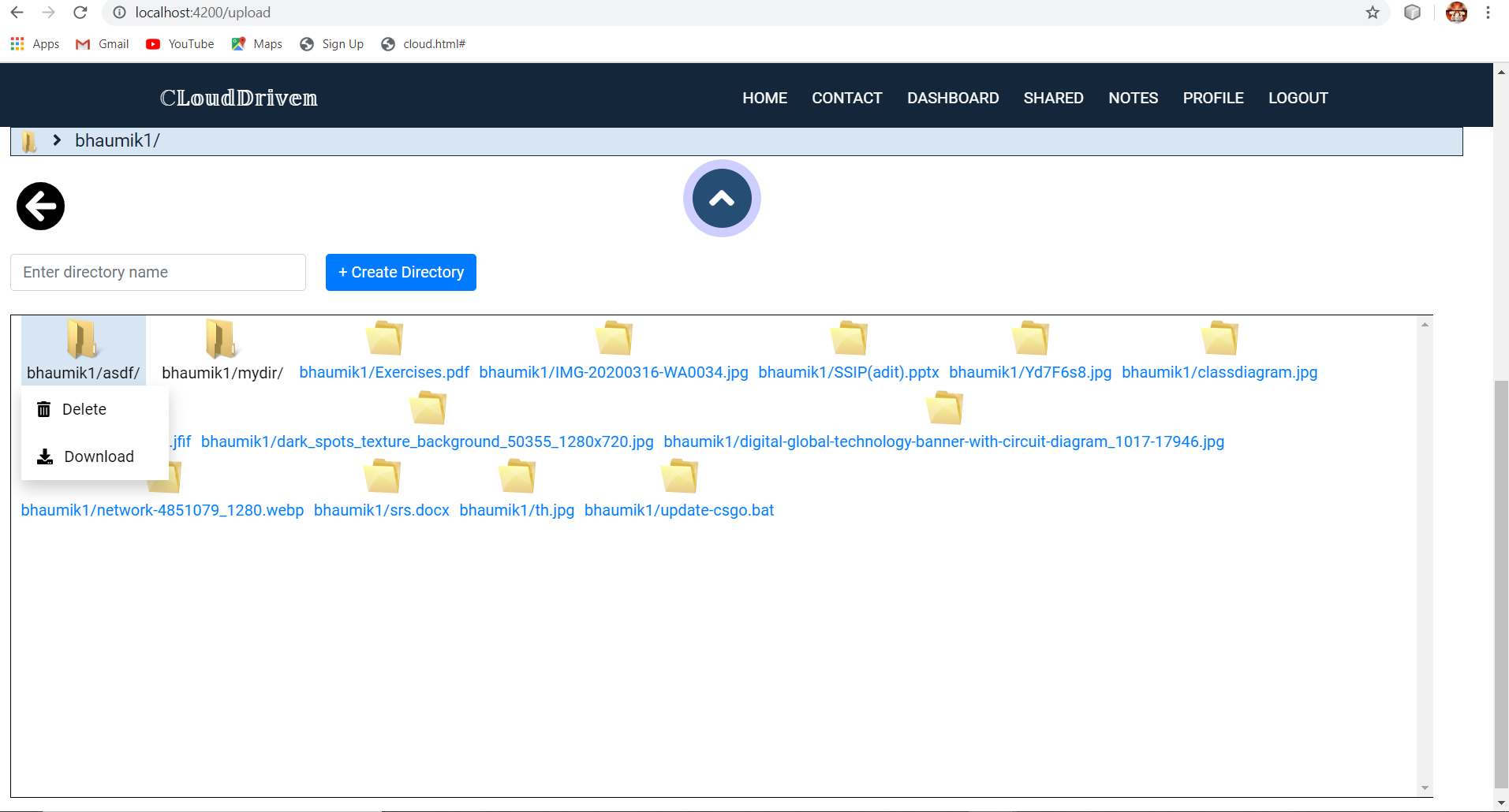


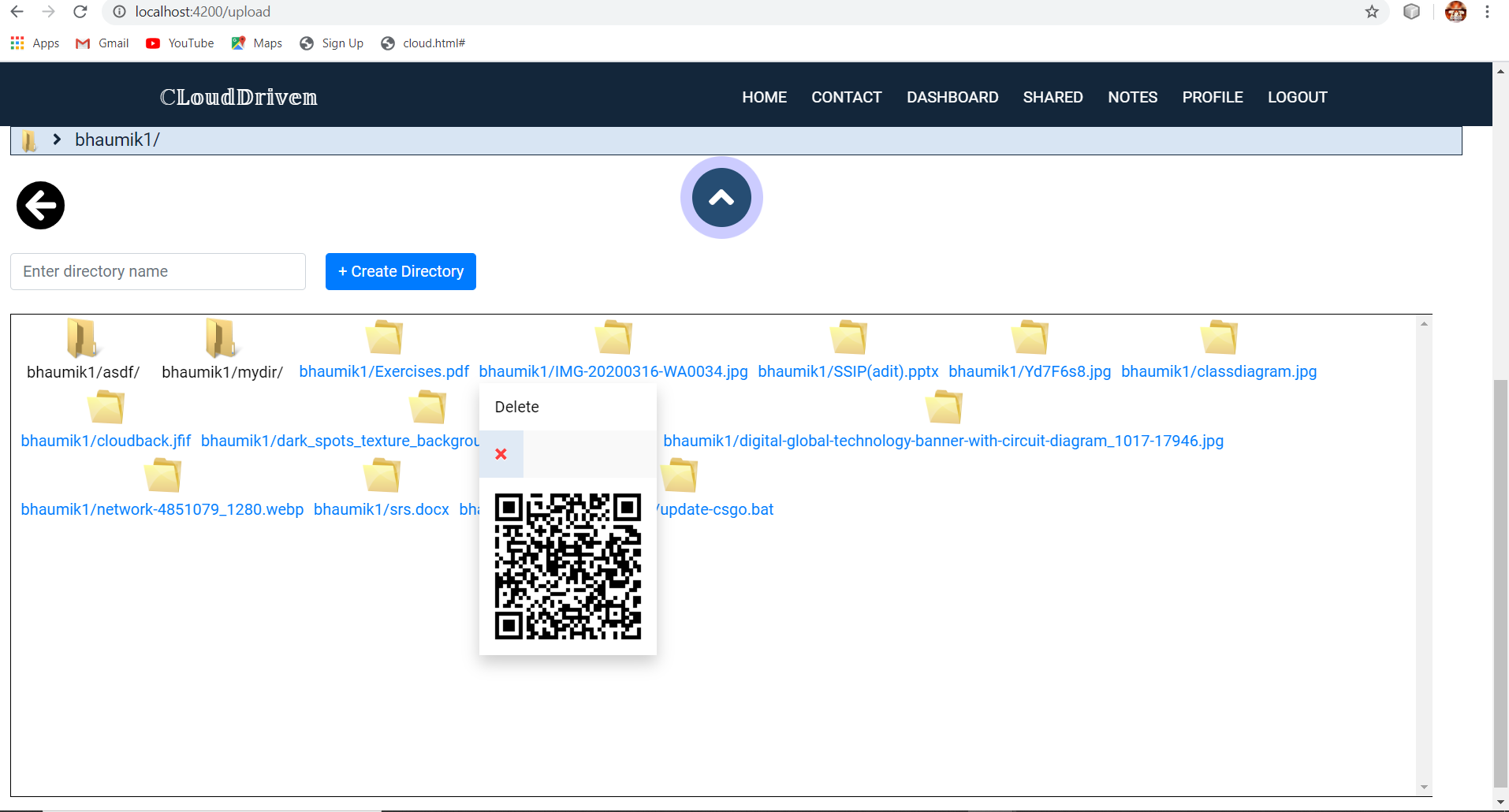


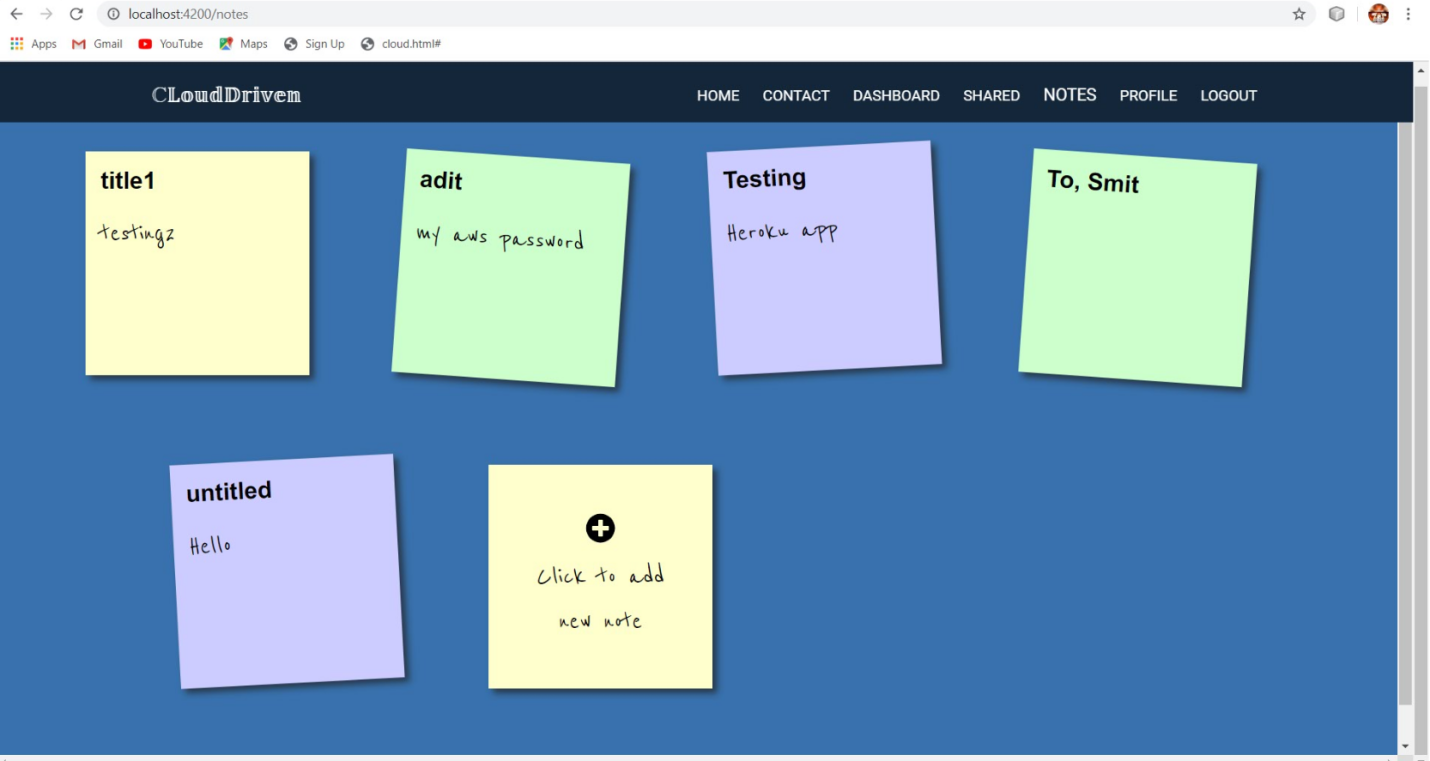


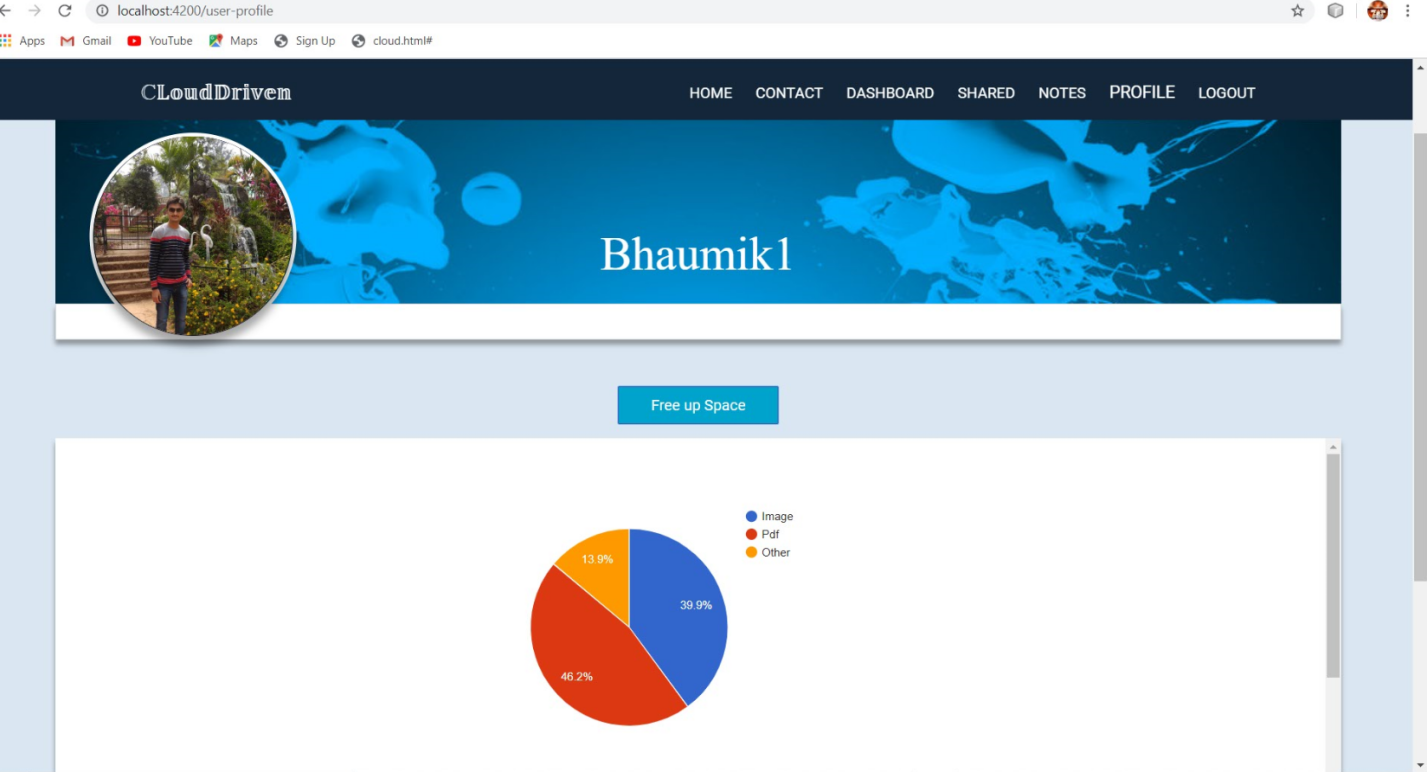


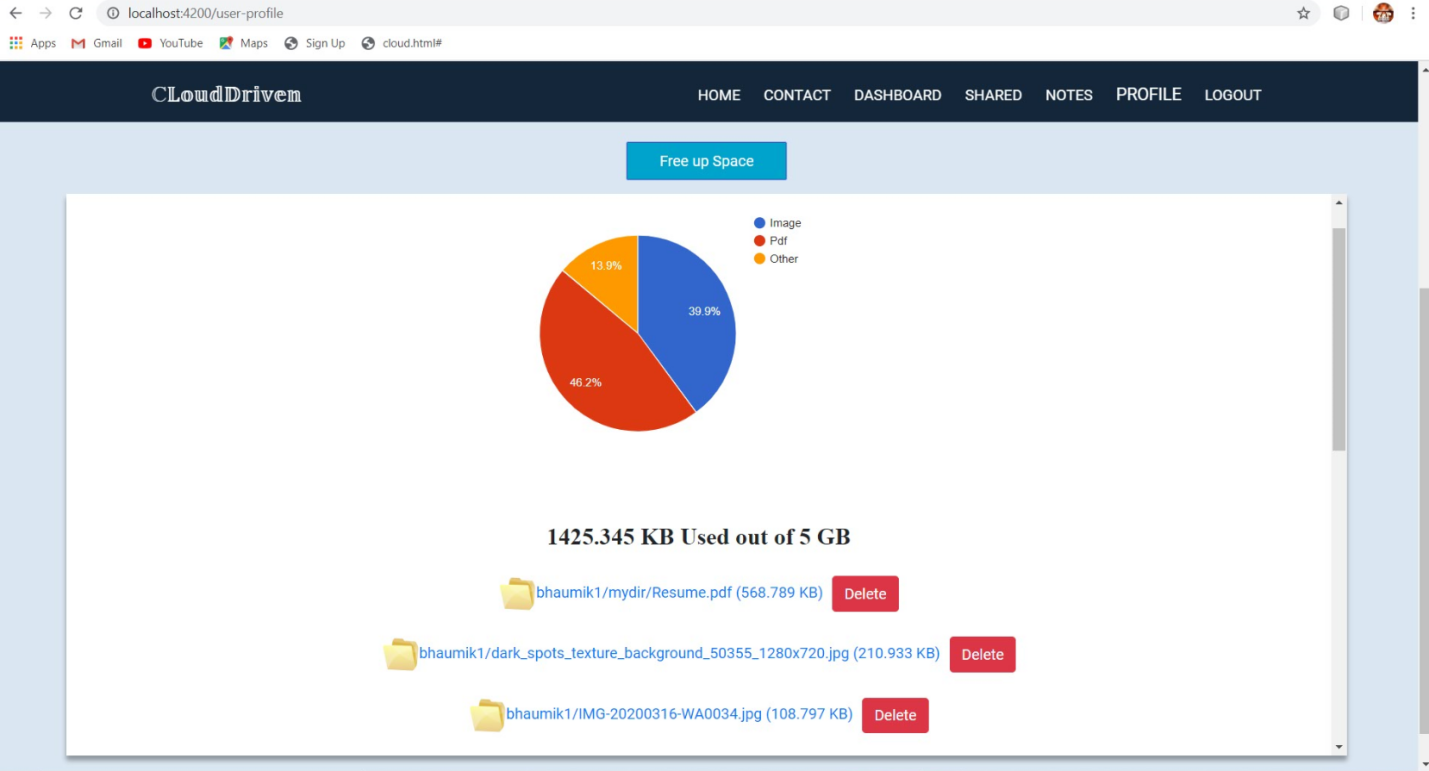


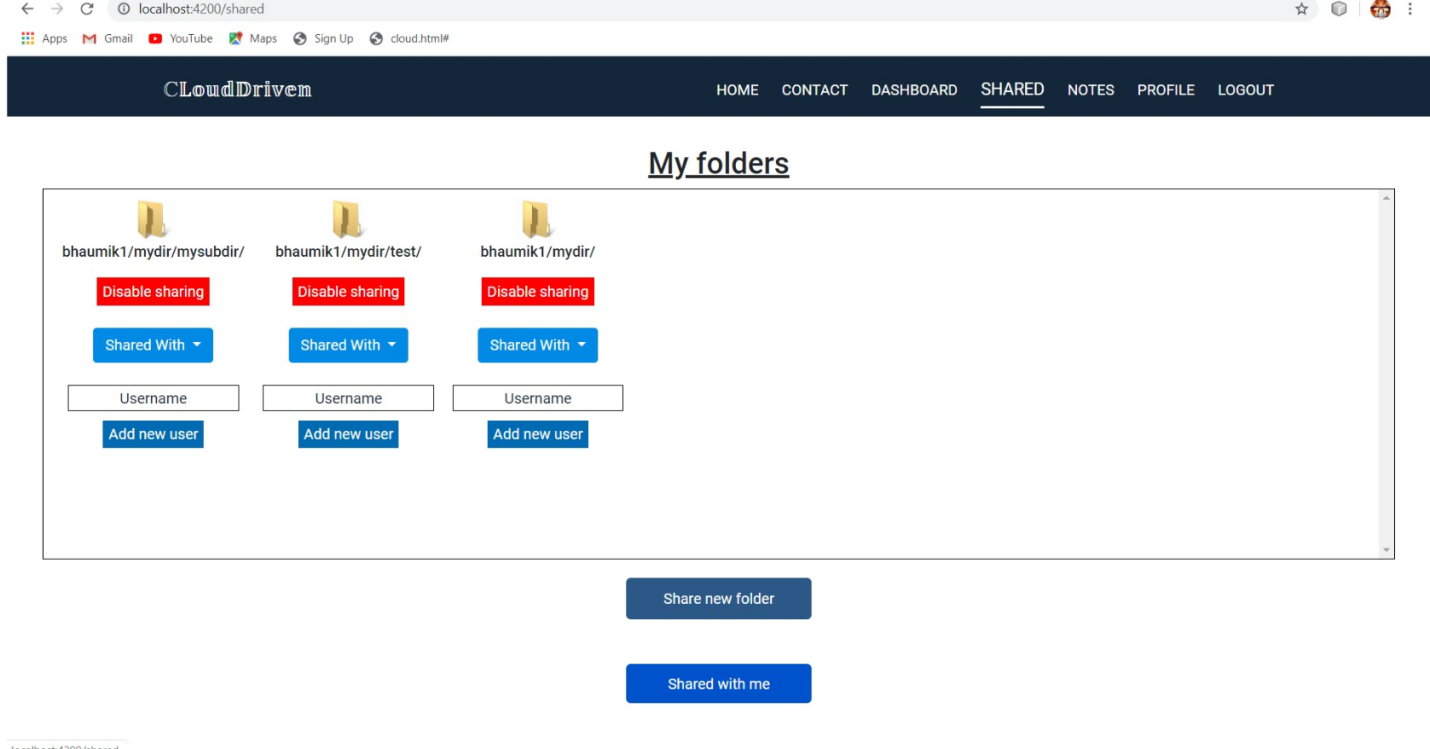


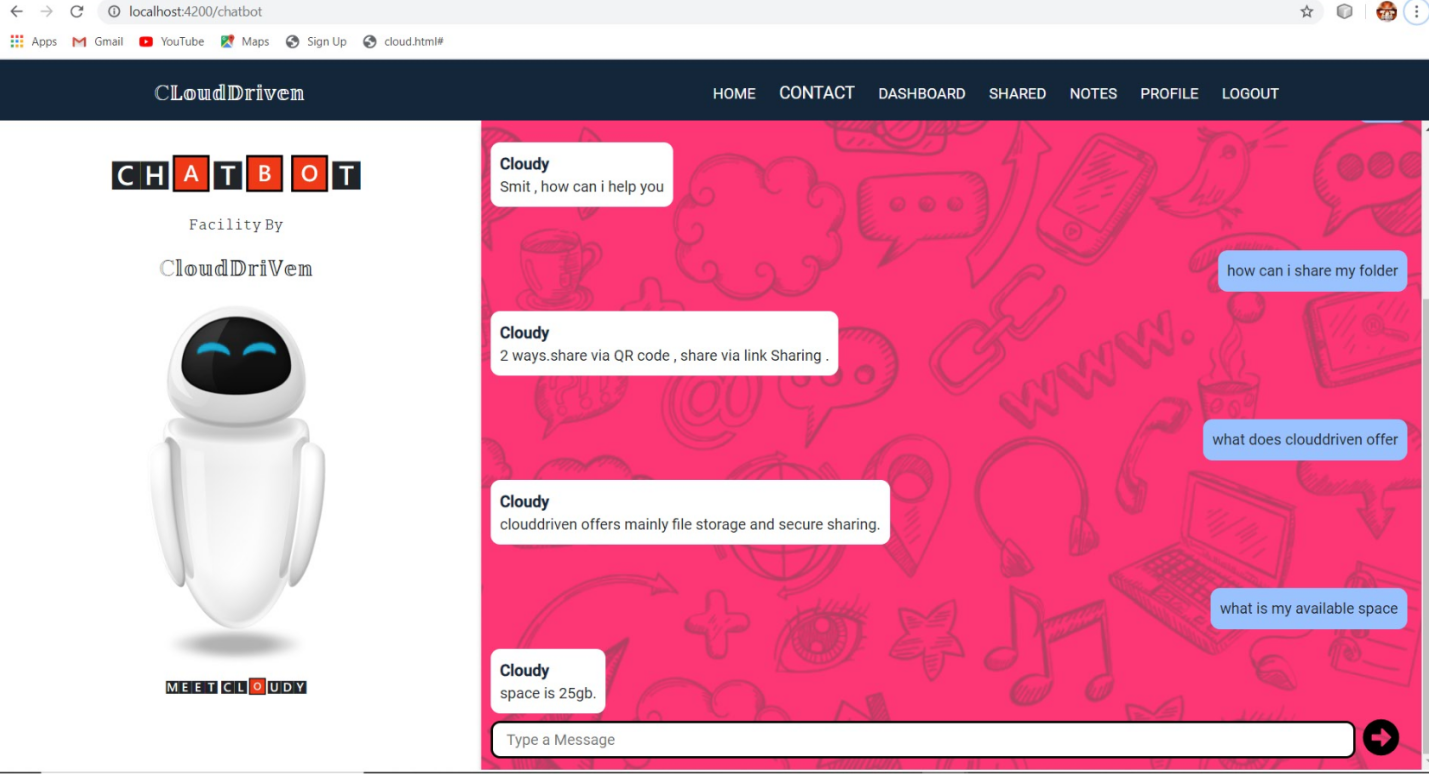












Conclusion

Our project - **Clouddriven** is able to successfully demonstrate a system for personal digital storage and secure sharing in which users can store any types of files and create an efficient directory structure and traverse it in a manner very similar to any traditional Operating system all in a cloud space. More over Users can share files via various methods such as link and Qr code all that in a intuitive and single page UI in angular, Enhanced security using AWS services. Along with these basic functionality, our system is able to provide some additional functionalities such as creating light weight sticky notes and sharing of folders using nodejs server and mongo db.

Limitation and future extension

* Currently we have developed web application in future mobile application can be developed , and it will be much easier because all our aws backend services also support mobile client.
* We have developed a chat bot that helps users to understand functions of our system, currently it is of limited scope , later can be optimized.
* In user data statistics, currently only a pie chart representing different type of files and its storage usage is displayed later more advanced charts can be added as well as analytics can be done after collecting sufficient user data.

Bibliography

<https://docs.aws.amazon.com/s3/>

<https://creately.com/lp/uml-diagram-tool/>

<https://www.umlet.com/>