

Delhi Metro Rail Corporation



An Internship Report

on

A file-sharing web application for DMRC employees using python-django

Submitted By-

VEDANTA SHARAN(21BCE2970)

BACHELOR OF TECHNOLOGY

In

Computer Science and Engineering(Core)

Vellore Institute of Technology, Vellore

Under the guidance of:

Mr. Manu Bhardwaj

Mr. Saurabh Sharma

ACKNOWLEDGEMENT

I take this opportunity to express my sincere thanks to Mr. Saurabh Sharma and Mr. Manu Bhardwaj and all other esteemed engineers of Delhi Metro Rail Corporation for having given me a chance to complete my training. I convey my thanks for their cooperation valuable guidance, constructive criticism, and constant encouragement. I am highly grateful to the Delhi Metro Railway Corporation for giving me an opportunity for the training and for providing the required facilities. It was a great experience working in the institution and learning from such knowledgeable engineers. I hope the knowledge gained here during the training would help me in the development of my career as an engineer.

ABSTRACT

Rapid urbanization and population growth in places like Delhi have raised the demand for reliable public transportation. The Delhi Metro, known for its extensive network and simple connections, is relied on daily by millions of passengers. Therefore systematic and secure sharing of data especially confidential is extremely important for the organization.

Thus, this project was an attempt of creating a personal application for the Delhi Metro Corporation.

CONTENT

- Acknowledgement.....2
- Abstract.....3
- Content.....4
- Introduction.....5-6
- Project Description.....7-13
- Database Design.....13-14
- Code Snippets.....14-16
- Results and discussion....16
- Conclusion and future scope.16
- References.....17

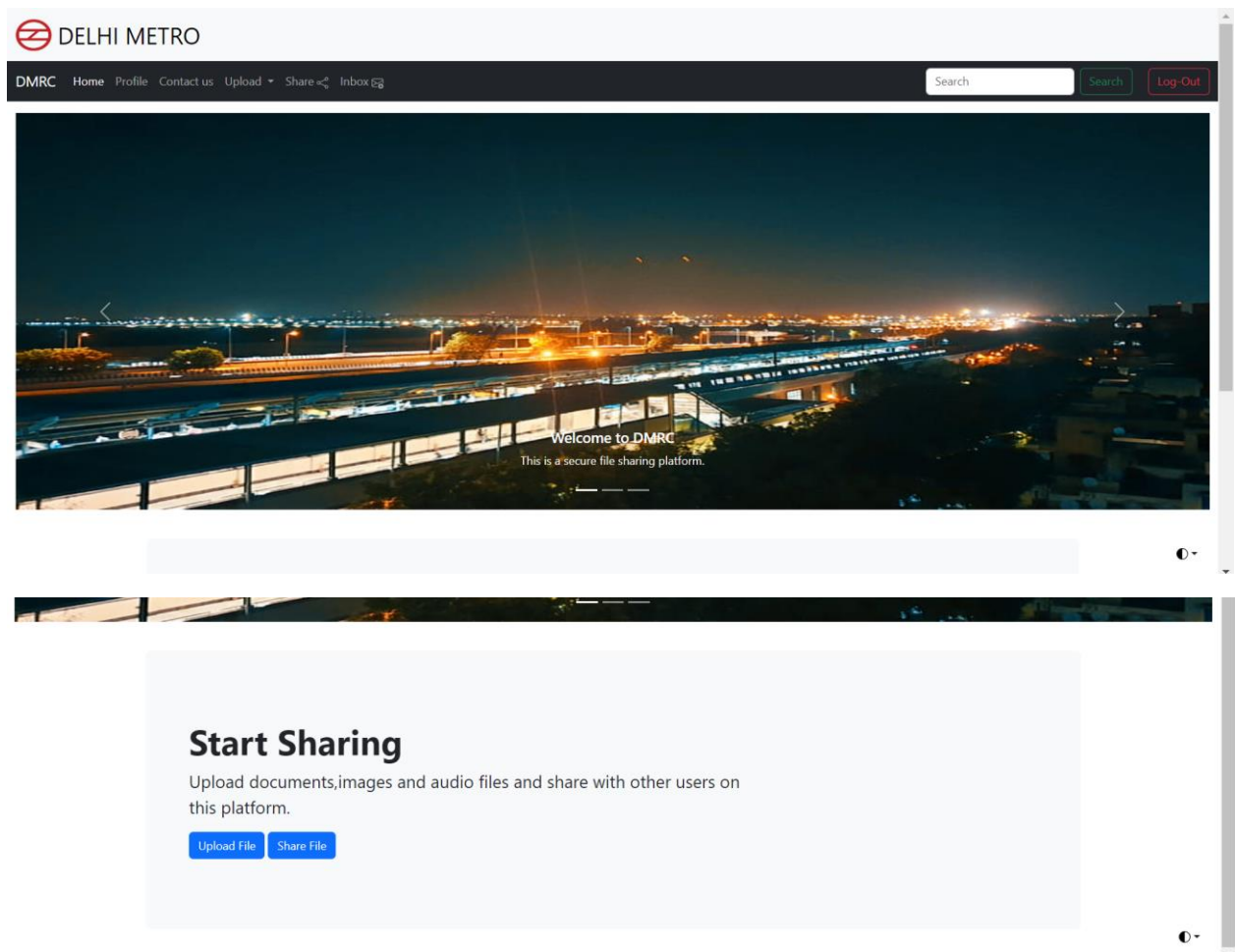
INTRODUCTION

The Delhi Metro has quickly become an integral part of the city's transportation network. Since its introduction in 2002, it has helped tremendously with the transportation needs of the city's expanding population. The metro system has benefitted millions of commuters by cutting down on traffic and offering a more comfortable mode of transportation because of its extensive coverage and efficient connectivity.

Effective communication is essential for the functioning of any organisation. It is important that the data being shared in the organisation remains intact and only accessible by authorised members of the organisation.

Thus , this project aims to create a file sharing platform for DMRC which has security measures put in place to keep the data safe from any hackers or unethical use. It uses Django and its in-built functions along with html and css code

connected to a No-Sql database(MongoDB) to create a fully functioning website.



OBJECTIVE:

The aim is to create a website which allows users to create an account, log in to the site , save files in their accounts for future use and share files with other users along with proper security measures to keep the data safe from potential threats.

TECHNOLOGIES USED:

- Python
- Django

- MongoDB
- Html ,Css , Javascript using Bootstrap

Project Description

The web application is created using python's web framework Django.

It consists of the following components:

1. The Home page:



It consists of a carousel from bootstrap with images of personal choice and can be changed at any time.


Apart from the carousel it has two buttons which are linked to two pages i.e the file upload page and the file sharing page respectively.


2.The Profile page:


- It consists of the profile of the user. It has a profile-picture component which has the profile picture set to the dmrc logo as default.
- It displays the username,email and a list of files sent to another user with the receiver's name, date and time and the file itself altogether.
- It also has a 'Files received' component which on clicking sends the user to the inbox page.

USER PROFILE


ProfilePic:



Username:
krish


Email:
krish8844@gmail.com


Files received
[Click here](#)

Files sent:	
	<ul style="list-style-type: none"> <div> <div>Receiver:</div> <div>Shared</div> </div> <div> <div>ShreeRam</div> <div>At: Oct. 11, 2023, 10:10 p.m.</div> <div>File:</div> </div> sharedfiles/4k-ultra-instinct-goku-dragon-ball-super-wallpaper-preview_IVA0H5a.jpg <ul style="list-style-type: none"> <div> <div>Receiver:</div> <div>Shared</div> </div> <div> <div>ShreeRam</div> <div>Shared At: Oct. 11, 2023, 10:11 p.m.</div> <div>File: sharedfiles/4k-ultra-instinct-goku-dragon-ball-super-wallpaper-preview_KsR3T3E.jpg</div> </div> <div> <div>Receiver:</div> <div>Shared</div> </div> <div> <div>ShreeRam</div> <div>Shared At: Oct. 11, 2023, 10:12 p.m.</div> <div>File: sharedfiles/4k-ultra-instinct-goku-dragon-ball-super-wallpaper-preview_1TRbA7Y.jpg</div> </div>

3.The Contact Us page:

- It consists of the username, email and a description of the issue the user is facing.
- The data after submission can be viewed in the mongodb database as well as the Django-admin's portal.



Contact us

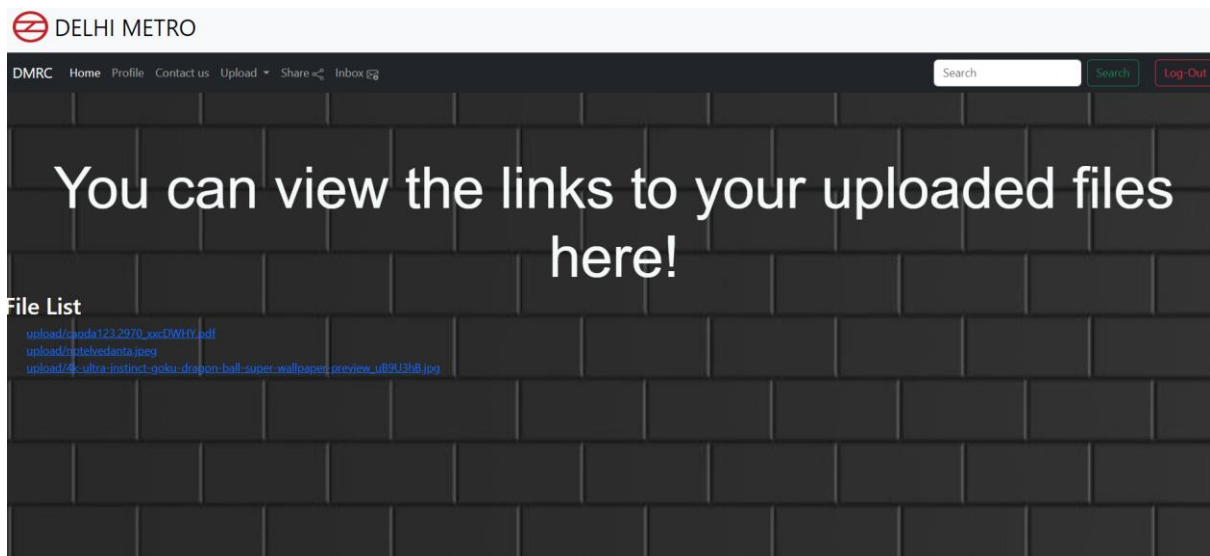
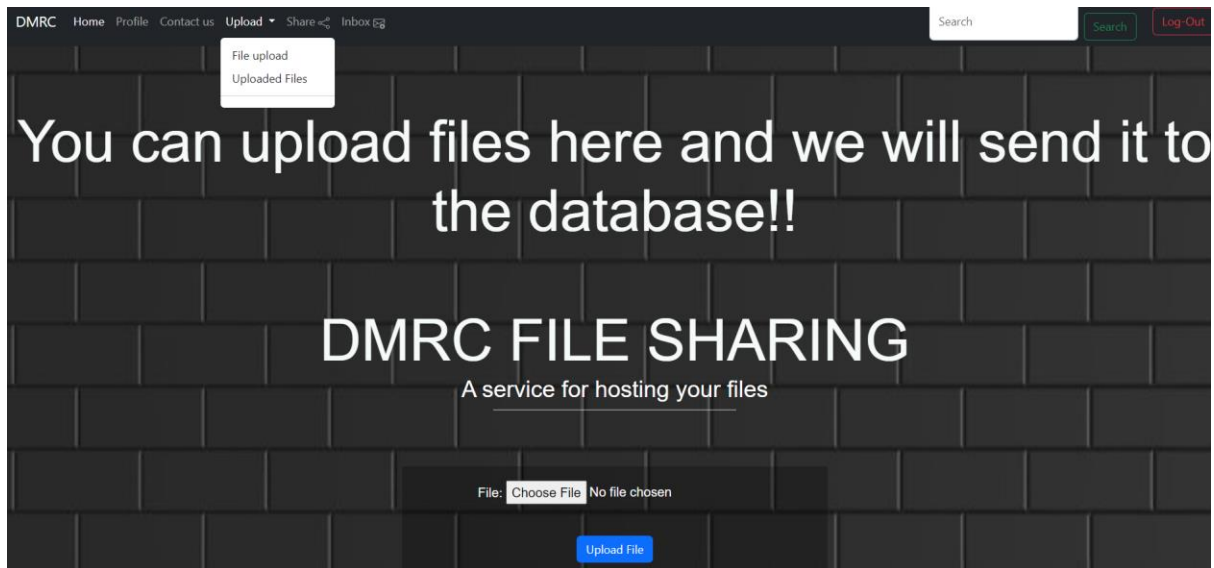
Username:

Email:

Desc:

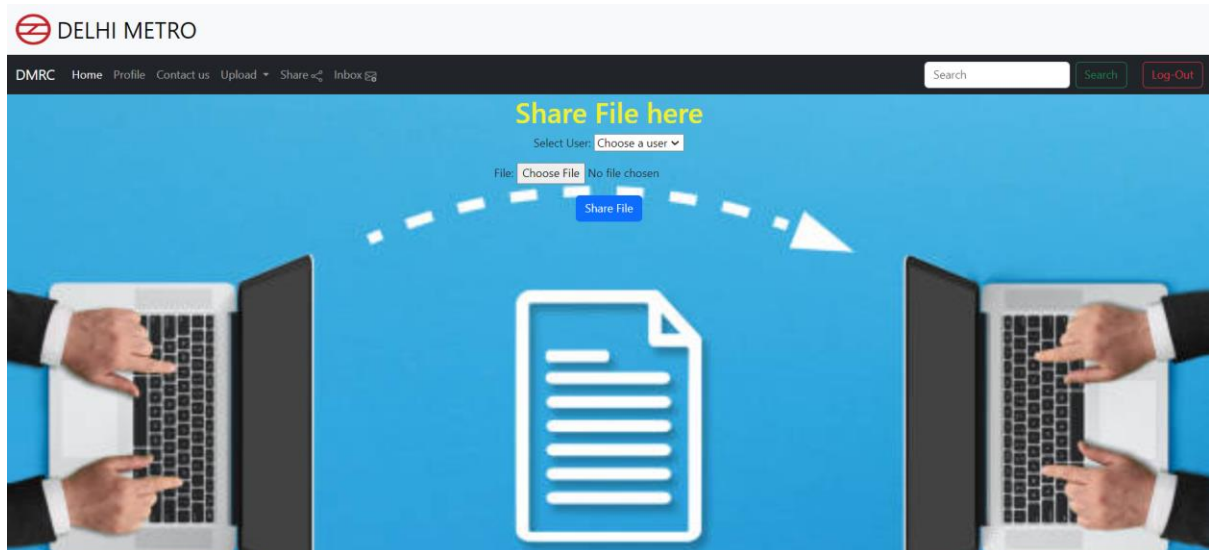
4.The Upload dropdown:

- It has two options:
- File Upload: It allows the user to upload the file and save the link to the file on the Uploaded files page.
- Uploaded files: It consists of the list of file links that the user has uploaded.

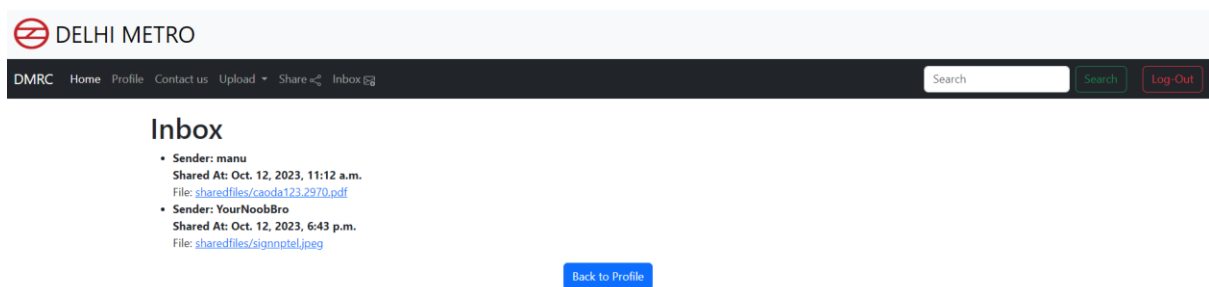


5. Share page: It consists of two fields:

- Receiver: to whom the file is sent to.
- File: The file to be shared.
- After sharing the user is redirected to the profile page with a success message and the file sent can be viewed in the file sent field of the profile page.




6. Inbox page: It consists of the list of files that have been sent by other users along with the sender's name and the date and time of sharing.



7. Signup page:

The signup page is used to create a custom user that can access the website by logging in and share files. Django's inbuilt UserCreationForm was used to create the page and it has security measures to ensure safe password creation.

Head to login page: [Login](#)



Welcome to DMRC

Create Account

Full name:

Email:

Username:

Required. 150 characters or fewer.
Letters, digits and @./+/-/_ only.

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation: Enter


the same password as before, for verification.

[Register](#)

If you already have an account, [login](#) instead.

8. Login Page: Finally , the login page allows the user to access the website by using their username and password. Only valid users created by the signup form(apart from the django's superuser) can access the website.

Don't have an account? [Sign Up](#)



Welcome to DMRC

Sign in

Username

Password

☐ Remember me

[Sign in](#)

© 2023

9.The update Profile page:

Update Profile

Profilepic: Currently: [profile_images/collegeidvedanta_UVCNvmH.jpg](#)

Change: [Choose File](#) No file chosen

Full name:

Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email:

Password:

No password set.

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using [this form](#).

[Update](#) [Change Password](#)

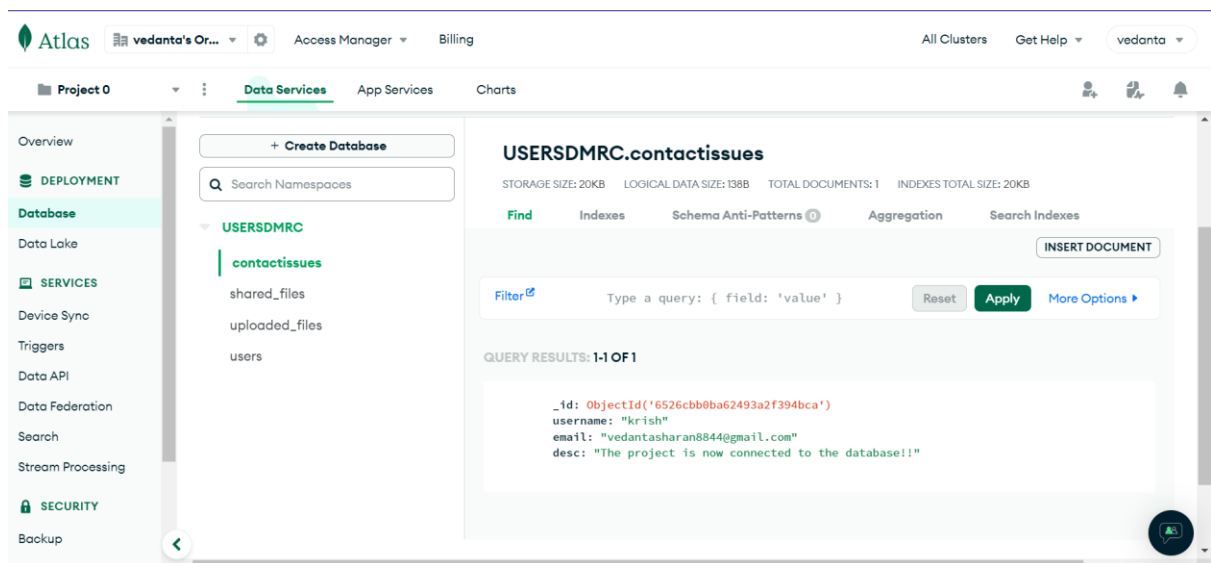
This page allows to update the information of the users and also allows them to change their password.

DATABASE DESIGN

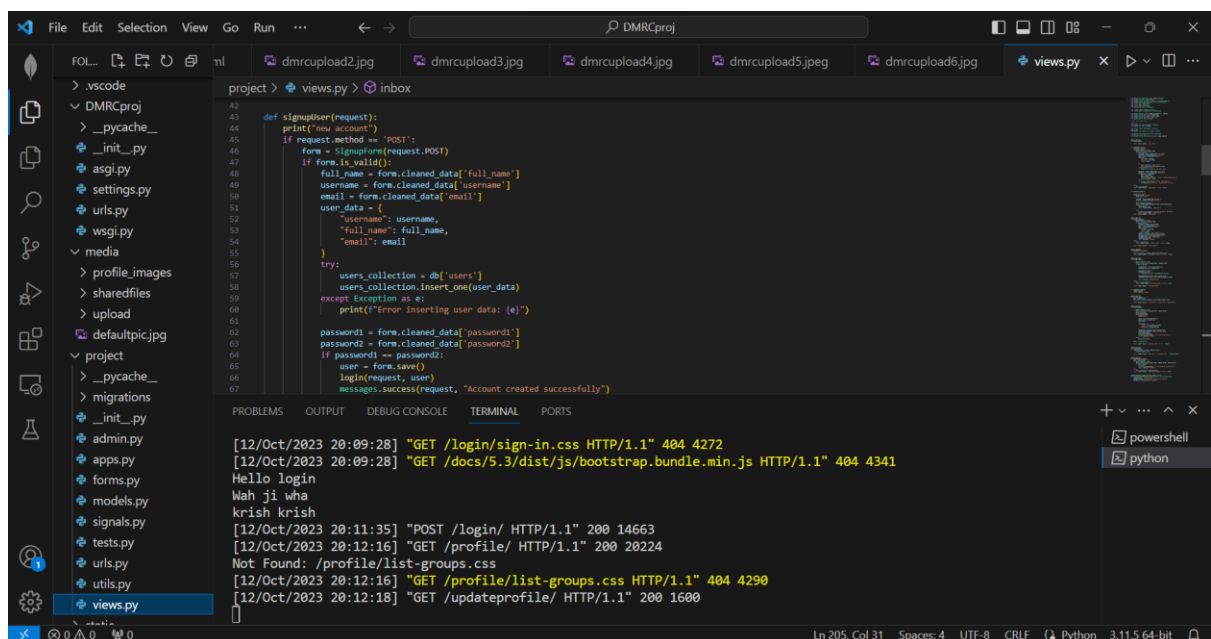
A No-Sql database MongoDB is used. The Django project is connected to the cloud storage of MongoDB called MongoDB atlas using PyMongo.

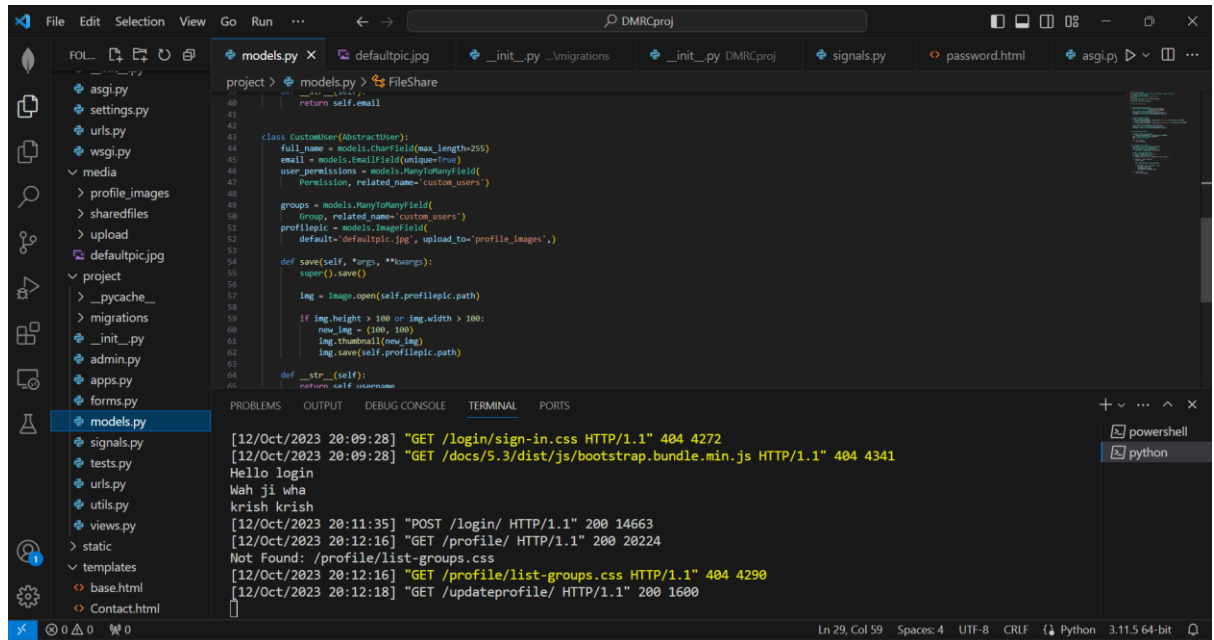
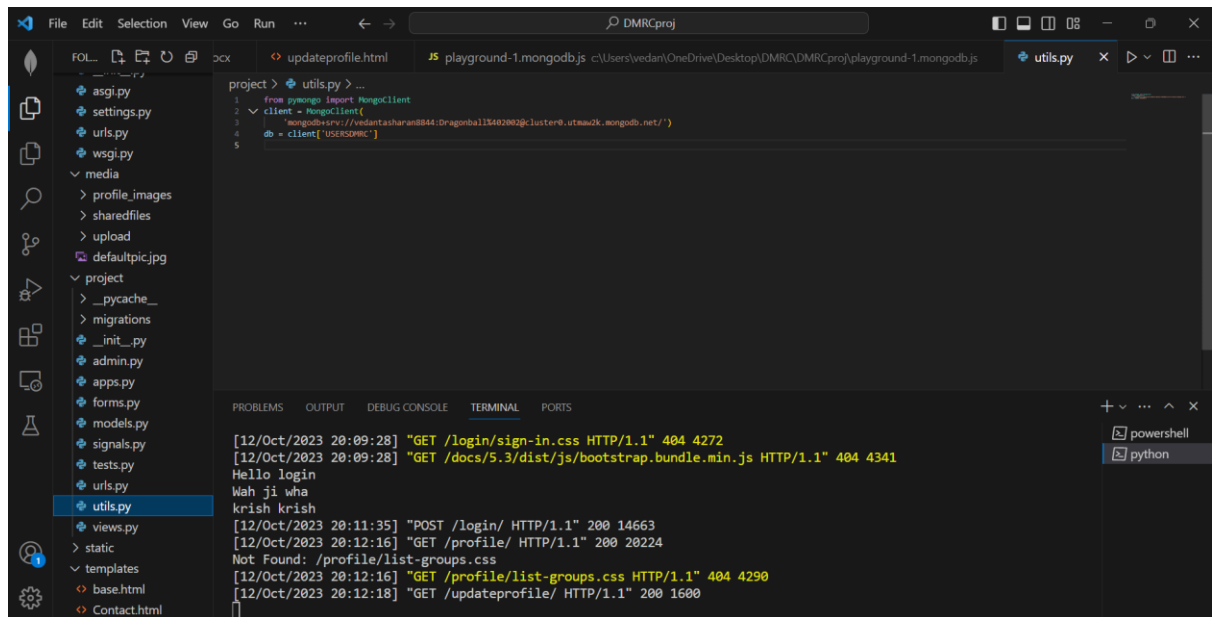
Four collections have been included to store data:

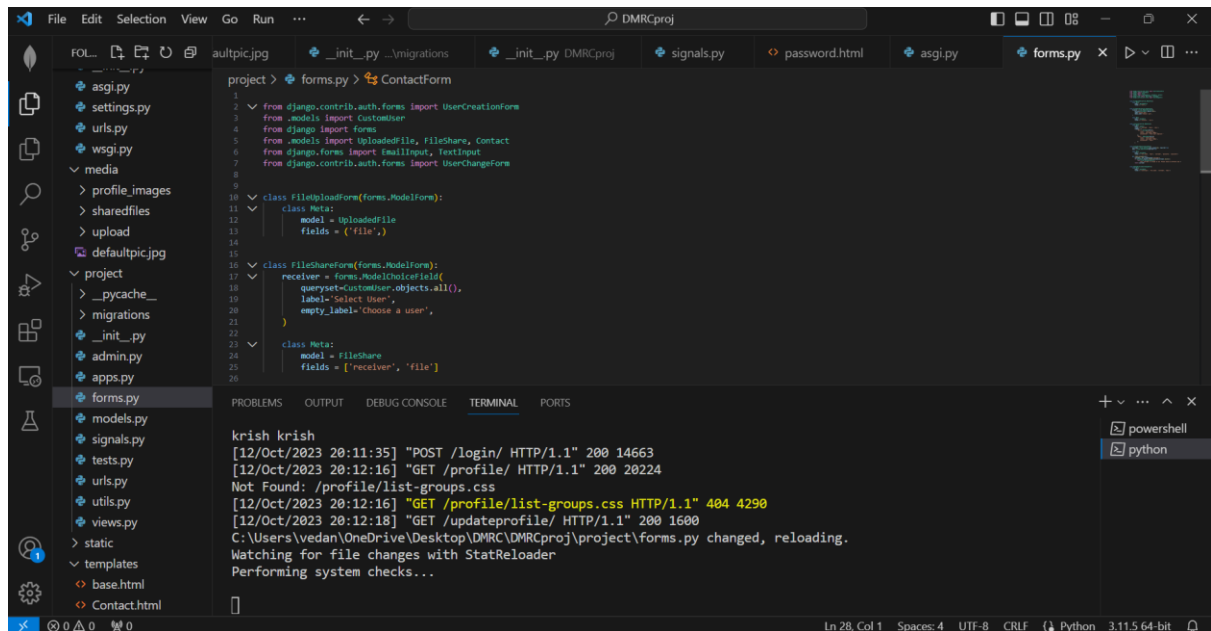
1. Users: To store the required user information.
2. Contactissues: To store the data posted in the 'Contact Us' page.
3. Uploaded_files: To store the file path of the uploaded files on the file upload page.
4. Shared_files: To share the file path of the shared files on the file share page.



CODE SNIPPETS







RESULTS AND DISCUSSION

The project was created by using Django integrated with a MongoDB database. The user can create an account, log-in, update the automatically created profile, write complaints to the administrator, upload and save files and share files with other users. The security features while user creation are implemented using django's built in features like Django.auth.

CONCLUSION AND FUTURE SCOPE

An attempt was made to create a secure file sharing platform for the Dmrc employees. The project has a lot of future scope related to design, security features and database implementation with a powerful framework like Django.

REFERENCES

- <https://docs.djangoproject.com/en/4.2/>
- <https://getbootstrap.com/>
- <https://www.mongodb.com/>
- www.google.com
- Stackoverflow.com
- Youtube

Github link to project:

[**https://github.com/Vedanta8844/Dmrcinternproject.git**](https://github.com/Vedanta8844/Dmrcinternproject.git)