

COMPLAINT MANAGEMENT SYSTEM

GROUP - 3

APPLIED DATABASES PRACTICUM

CS207

DR. VARUN DUTT



INDEX

I.	The Team.....	2
II.	Acknowledgement.....	2
III.	Abstract.....	3
IV.	Introduction.....	4
V.	Methodology.....	5
VI.	Result/Output.....	7
VII.	Conclusion.....	11



THE TEAM

Group Number 3

1. Yash Sharma B20241
2. Saransh Bansal B20132
3. Aryansh Singla B20085
4. Abhay Gupta B20075
5. Aayushmaan Jha B20175
6. Vastav Bansal B20325
7. Vishwas Garg B20143

ACKNOWLEDGEMENT

We would like to extend a heartfelt gratitude to Dr. Varun Dutt, the course instructor of CS207-Applied Database Practicum for providing us with an opportunity to work on the Complaint Management System project, and for enriching us through your very informative lectures and guidance.

We extend our gratitude to Shashank Uttrani sir and Aadhar Gupta sir and all the TAs for helping us at every step.

We are indebted to our friends and team members for the constant support and motivation.

ABSTRACT

Problem Statement :- Develop an application for students to register and login and raise complaints against their problem. The portal should contain a form to request necessary details of the complaints and should be directed to concerned authorities.

The complaint management system is a web based application and it is designed to keep track of complaints registered by the students. Students need to register and login to the system to raise their complaints. System will direct all the complaints to the concerned authorities. System also keeps track of previous complaints registered by the particular student. Students can see their past complaints as well.

This system helps students to raise their complaints easily. Each student has a password protected account to ensure privacy and security. Also, no student can see any other student's registered complaints. This makes students more comfortable to post their complaints without any hesitation.

INTRODUCTION

This project aims to create an effective Complaint Management system using Databases, to be used as an application for students to register their complaints, regarding the problems they face.

The tasks at hand included building a website consisting of various web pages, which included the Sign-in page, the Sign-up page, the page for Forget Password, and the page for the submission of the complaints.

For all the pages, separate EJS files (Embedded JavaScript files) were used to store the basic layout. Corresponding CSS files were created for designing the pages. NodeJS software was used to complete the JavaScript part.

Various Libraries were used to perform numerous functions . Examples include Nodemailer, Bcrypt etc.The link to the github repository where the project was created is:

<https://github.com/yashcode00/Complaint-management-system>

The user is directed to the login page whenever he opens the link for the complaint registration site. The user needs to fill in his credentials if he has already registered, else there is an option to sign up for a new user where he initially submits his details and sets a unique username and a password to create the account. The password must satisfy the criteria as mentioned on the site and confirm the password submitted before. If he fails to fulfill any of the criteria he won't be able to register and hence he won't be able to create the account. Besides, we have also developed an option of 'forgot password' in case someone just forgets his password.If he clicks on '[Forgot your password?](#)' , he will have to enter his username and a random 4-digit OTP will be sent to his registered mail. The user can change his password by filling the OTP sent to him and then setting a new password will get directed to the login page and can fill in the credentials to register the complaint. Now, he can register a new complaint depicting the type of the complaint and the issue. The user can also see the complaints lodged in the past. The project is hosted at :

<https://complaint-management-sys.herokuapp.com/login>

METHODOLOGY

The main objective of our project was to build a complete working website (frontend+backend) for students. This is a management application for students to register and login and raise complaints against their problems. The portal contains a form to request necessary details of the complaints and directs them to concerned authorities. This way this application helps to deal with various complaints of the student community effectively via directing them to their concerned authority.

Various elements are used as building blocks of this website. We will start with the Frontend. The frontend is basically made using simple *EJS or Embedded Javascript Templating* which is a templating engine used by Node.js. This Template engine helps to create an HTML template with minimal code. Also, it can inject data into an HTML template at the client side and produce the final HTML. Moreover the presentation of the website is done using CSS (the styling part). In the end, to make our website more responsive i.e. buttons and tables have special effects etc. javascript has also been used along with these. All css and js files are present in the “Public” folder along with the images used in this project. The EJS files are present in views folders along with partials folder which contains a common footer for all EJS files. This way we have completed our frontend using the listed technologies.

For the backend, Node.js has been used. A file named app.js has been made in our code folder to handle all code for the backend. Along with it to create and store data, a NoSQL database program, MongoDB has been used.

To make the backend process various modules available with NODe.js have been used. This could be easily installed via using the command line `npm install module_name`. Some of the modules used are listed below:

1. ExpressJs: ExpressJS is a prebuilt Node JS framework that can help us in creating server-side web applications faster and smarter. Simplicity, minimalism, flexibility, scalability are some of its characteristics and since it is made in NodeJS itself, it inherited its performance as well. It helped us to host our project locally on port 3000 (localhost:3000) and also deal with various get and post requests.

- 
2. Mongoose: Mongoose is a Node.js based Object Data Modeling (ODM) library for MongoDB. It is akin to an Object Relational Mapper (ORM) such as SQLAlchemy for traditional SQL databases. The problem that Mongoose aims to solve is allowing developers to enforce a specific schema at the application layer.
 3. Bcrypt: Bcrypt is a popular and trusted method for salt and hashing passwords. All the passwords are hashed using this module only.
 4. Nodemailer: Nodemailer is a module for Node.js applications to allow easy as cake email sending. All emails sent whether it's for forgetting password or it is to the concerned authority all are done using this module only.
 5. Connect-flash: The flash is a special area of the session used for storing messages. Messages are written to the flash and cleared after being displayed to the user. The flash is typically used in combination with redirects, ensuring that the message is available to the next page that is to be rendered. All the error or success notifications are done using flash only.
 6. Cookie-parser: Cookies are small data that are stored on a client side and sent to the client along with server requests. In this project various things are stored in cookies. This is done using this module only. The Cookie-parser module of npm which provides middleware for parsing of cookies.

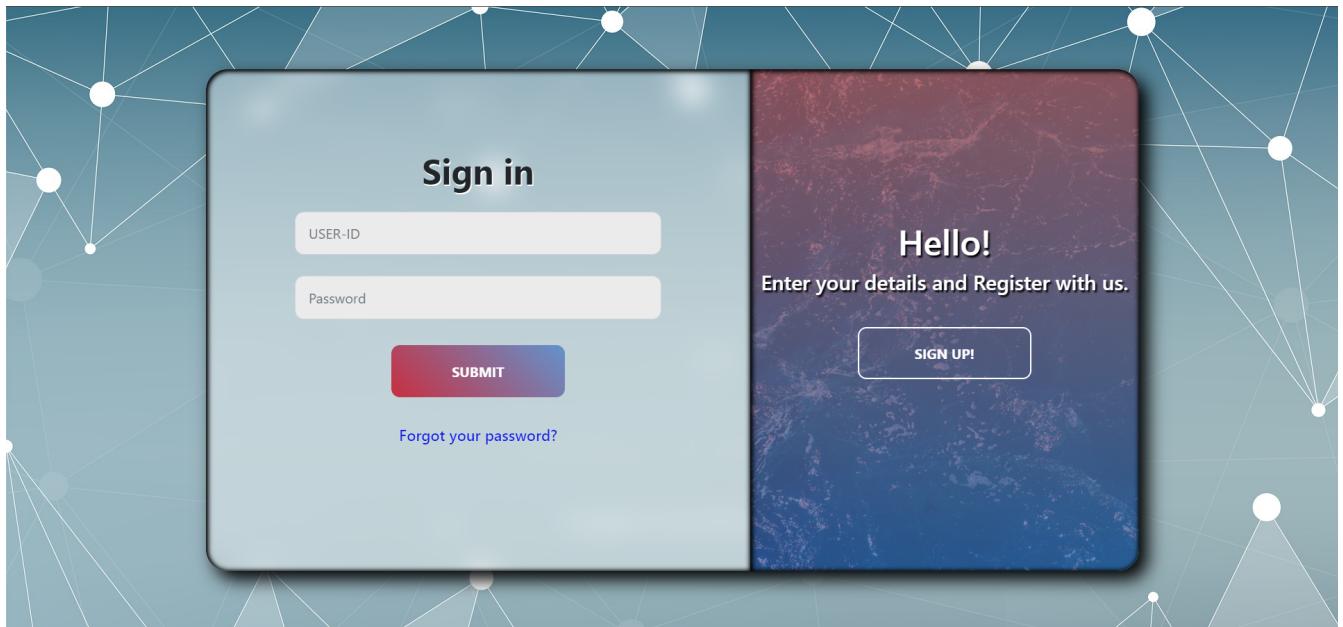
All the modules with used version details are present in a file in the project folder named : *package.json*.

Thus all these frontend and backend when combined then gives us a working portal that is capable to handle any type of given request ,given the request has some legit operation otherwise if a request is received is invalid or has no meaning to the system then the user is directed to to a 404 Error page making the process more smoother to understand.

This way we have obtained a management system as aimed in the beginning.

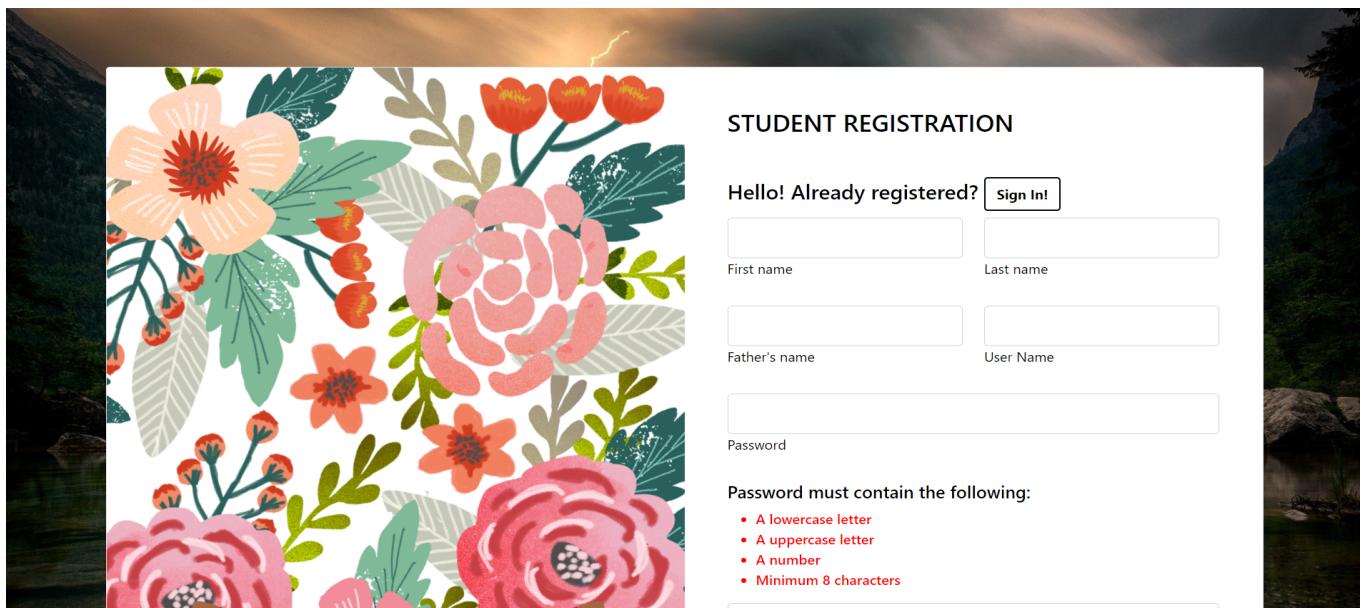
RESULTS AND OUTPUTS

1.



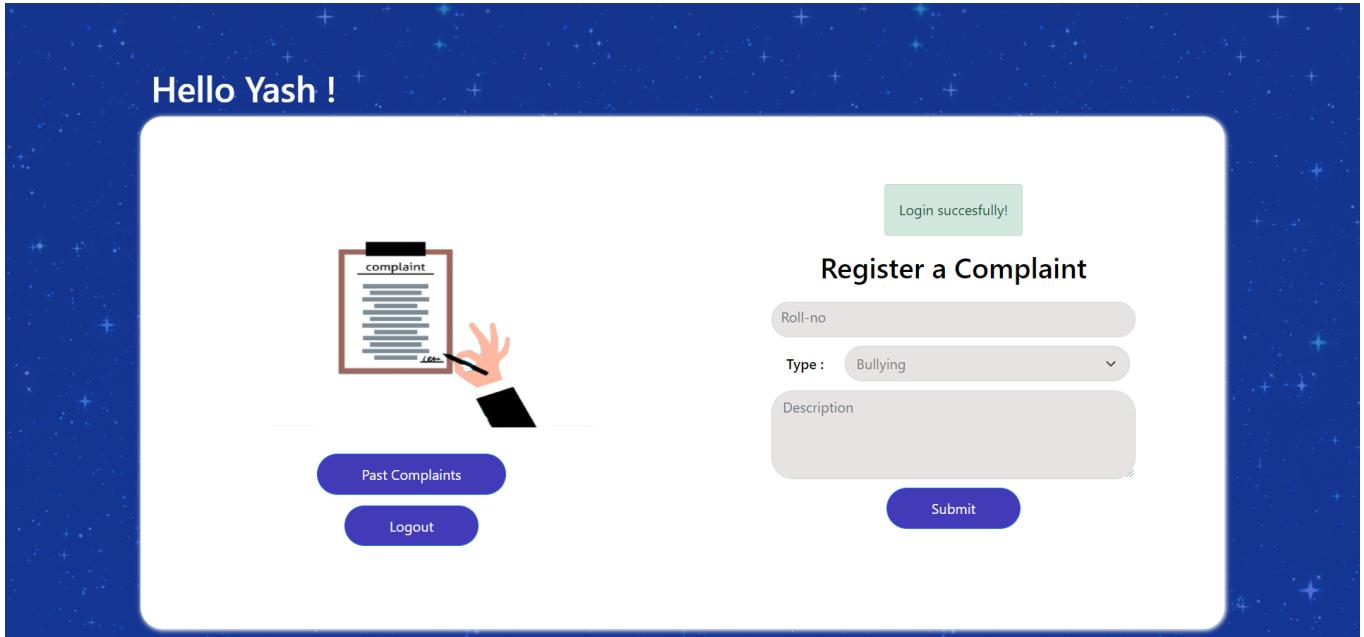
(Home page for log In or Sign up)

2.



(Page for Registration)

3.



(Page for registering your complaint)

4.

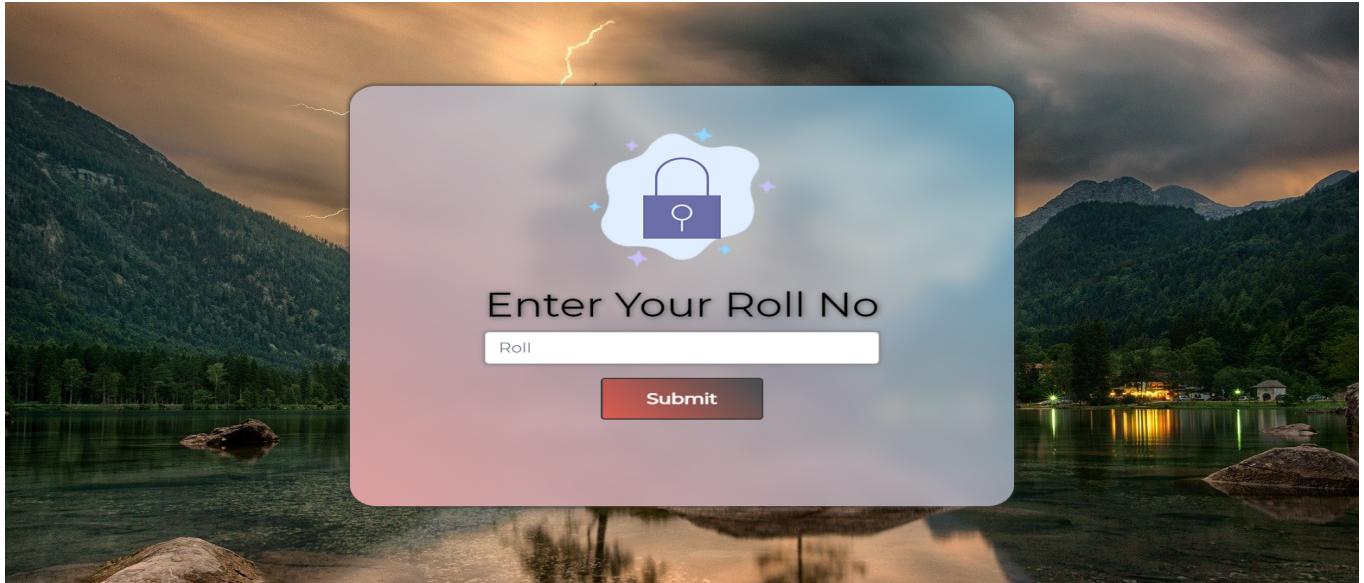
The screenshot shows a table of previous complaints. At the top left, there is a "Back" button. The table has columns for S.No., Timestamp, Username, Roll No, Complaint Category, and Complaint. The data is as follows:

S.No.	Timestamp	Username	Roll No	Complaint Category	Complaint
1	Thu Nov 11 2021 18:38:04 GMT+0530 (India Standard Time)	John_wick	1	Bullying	Bullying Test
2	Thu Nov 11 2021 18:38:33 GMT+0530 (India Standard Time)	John_wick	1	Harrasment	Administration Harasses me!
3	Thu Nov 11 2021 18:39:02 GMT+0530 (India Standard Time)	John_wick	1	Hostel	Never Visited so no Complaints

(Page of Previous Complaints)

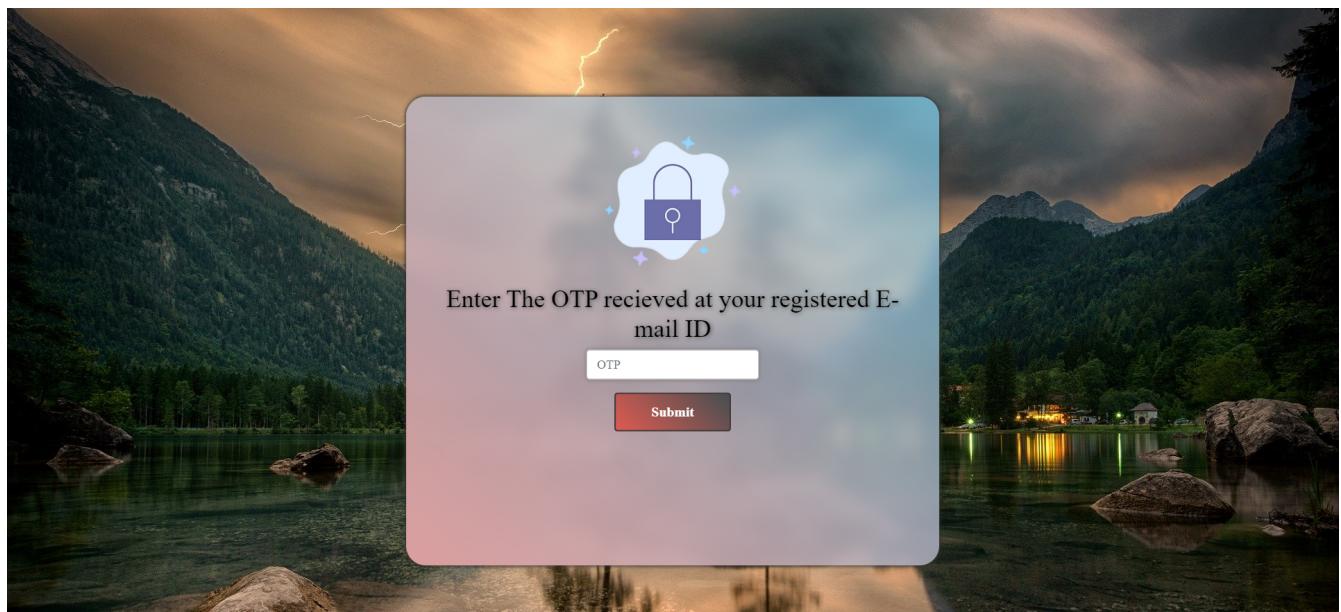


5.



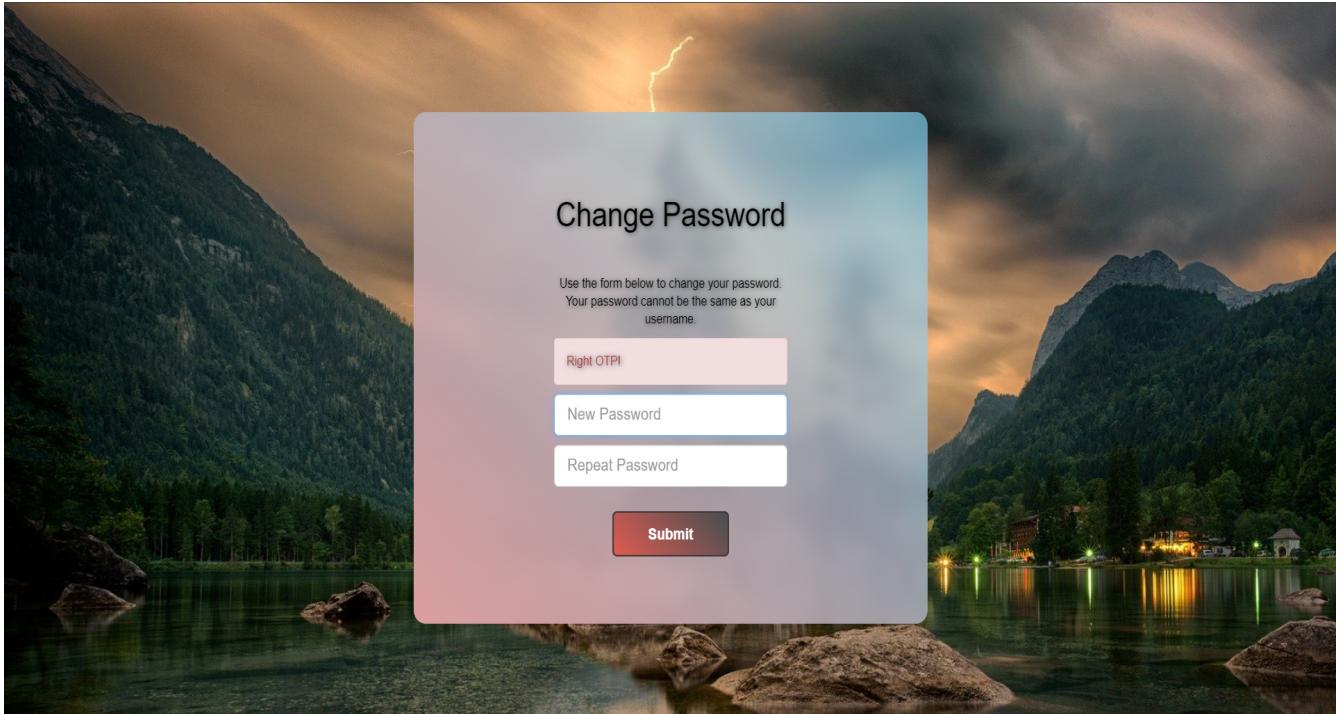
(Page for generating OTP for resetting password)

6.



(Page for entering OTP)

7.



(Page for resetting password)

CONCLUSION

The purpose of this project was to make an effective Complaint Management System using Databases- an application for students to register their complaints against the problems faced by them. HTML, CSS and JavaScript were used for the frontend design, with the Backend part being done using the NodeJS software. MongoDB was used to store and handle the data. The registration and login system is a robust one ,with encrypted passwords and mailing functionality, made using BCrypt and NodeMailer. Further enhancement in the website could be done by introducing, facilitating, and enhancing communication between the users themselves, so that they can interact with one another. This, in retrospect will reduce the number of complaints in total as repetitive complaints would be avoided relatively. There could also be added suggestions for the complaints by other users, available for the user to see while he is registering one of his own, again, in order to avoid repetitions. Suggestions could be taken from the users to solve their own problems and that of others as well, in order to move the website more in the direction of self-sufficiency. The users could be rewarded too based on their participation in resolving other's complaints, in order to increase participation and create a more self-reliant website and a more wholesome environment , making it more popular. However, for all these suggestions to be implemented a lot more work has to be done, with ample amount of time and effort also being required. Till then, this is a basic structure for a Complaint Management System successfully prepared hence.
