



PROJECT REPORT
ON
“ONLINE COMPLAINT REGISTRATION
AND MANAGEMENT SYSTEM”

A Project Report Submitted in Partial fulfillments of Requirements for the Award of the Degree of

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted To



Submitted By

S.NO	Team Id	Student ID (Hall Ticket Number)	Student Name
1	LTVIP2024TMID13295	207T1A0402	Shaik Rehaman
2		207T1A0401	Borra Goutham
3		207T1A0403	Shaik Shanavaz
4		207T5A0402	Thokala Venu

Contents

Chapter 1: INTRODUCTION

- 1.1 Overview A brief description about your project
- 1.2 Overview of Scenario based case study of your project
- 1.3 Purpose The use of this project. What can be achieved using this.

Chapter 2: LITERATURE SURVEY

- 2.1 Existing problem Existing approaches or method to solve this problem
- 2.2 Proposed solution What is the method or solution suggested by you?

Chapter 3: THEORITICAL ANALYSIS

- 3.1 Block diagram Diagrammatic overview of the project.
- 3.2 Hardware / Software designing Hardware and software requirements of the project
- 3.3 Project structure
- 3.4 Application FLOW of your project

Chapter 4: RESULT

- 4.1 Final findings (Output) of the project along with screenshots.

Chapter 5: ADVANTAGES & DISADVANTAGES

- 5.1 List of advantages and disadvantages of the proposed solution

Chapter 6: APPLICATIONS

- 6.1 The areas where this solution can be applied

Chapter 7: CONCLUSION

- 7.1 Conclusion summarizing the entire work and findings.

Chapter 8: FUTURE SCOPE

Enhancements that can be made in the future.

Chapter 1

INTRODUCTION

In the digital age, effective complaint registration and management systems are pivotal for organizations to uphold customer satisfaction and operational integrity. The proposed Online Complaint Registration and Management System aims to address this need by leveraging Full Stack Development technologies, specifically the MERN (MongoDB, Express.js, React.js, Node.js) stack. This system will revolutionize the way complaints are lodged, tracked, and resolved, benefiting both customers and service providers.

Traditional complaint handling methods often suffer from inefficiencies, such as delayed responses, lack of transparency, and cumbersome paperwork. By transitioning to an online platform, organizations can streamline the entire complaint lifecycle, from submission to resolution.

Key objectives of the system include ensuring user authentication for security, providing intuitive complaint submission forms, enabling real-time tracking and monitoring, automating assignment and escalation processes, facilitating seamless communication channels, offering robust reporting and analytics, and ensuring accessibility and responsiveness across devices.

The system's implementation will not only enhance operational efficiency but also improve customer satisfaction by providing transparency, timely responses, and effective issue resolution. Moreover, the integration of stringent security measures will safeguard user data and uphold confidentiality throughout the complaint handling process. Through this project, organizations can cultivate stronger relationships with their customers and foster a culture of continuous improvement and accountability.

1.1 Overview A brief description about your project

The Online Complaint Registration and Management System is a comprehensive web-based platform designed to streamline the process of handling complaints for organizations across various industries. Developed using Full Stack Development technologies, specifically the MERN (MongoDB, Express.js, React.js, Node.js) stack, this system offers a user-friendly interface for both customers and service providers.

At its core, the system enables users to register complaints online, providing details such as category, severity, and attachments through intuitive submission forms. Once lodged, complaints are efficiently tracked in real-time, allowing users to monitor progress and receive updates throughout the resolution process.

Automated assignment and escalation protocols ensure that complaints are routed to the appropriate staff members and escalated when necessary to meet predefined service level agreements. Built-in communication channels facilitate seamless interaction between customers and staff, fostering transparency and accountability.

Robust reporting and analytics tools provide insights into complaint trends, resolution times, and customer feedback, empowering organizations to make data-driven decisions for continuous improvement.

With a mobile-responsive design and stringent security measures in place, the Online Complaint Registration and Management System aims to enhance operational efficiency, improve customer satisfaction, and uphold confidentiality in the complaint handling process.

1.2 Overview of Scenario based case study of your project

The scenario-based case study for the Online Complaint Registration and Management System revolves around a fictional organization, "CustomerCare Inc.," a leading telecommunications company with a large customer base. Facing challenges with their current complaint handling processes, CustomerCare Inc. decides to implement an Online Complaint Registration and Management System to improve efficiency and customer satisfaction.

The case study follows the journey of CustomerCare Inc. through various stages of implementing and utilizing the system:

Current Challenges: CustomerCare Inc. struggles with manual and inefficient complaint handling processes, leading to delayed responses, lack of transparency, and dissatisfied customers. The organization realizes the need for a modernized solution to streamline complaint management.

Selection of Solution: After thorough research and evaluation, CustomerCare Inc. decides to implement an Online Complaint Registration and Management System built using the MERN stack. The decision is based on the system's comprehensive features, scalability, and compatibility with the organization's existing infrastructure.

Implementation: The implementation process involves collaboration between CustomerCare Inc.'s IT department and the development team responsible for building the system. Customizations are made to align the system with CustomerCare Inc.'s specific requirements and branding.

Training and Adoption: CustomerCare Inc. conducts training sessions for staff members to familiarize them with the new system's functionalities. User guides and support resources are provided to ensure a smooth transition.

Operationalization: With the system fully implemented, CustomerCare Inc. begins using it to register, track, and resolve complaints. The system's automated features, real-time tracking, and seamless communication channels significantly improve complaint handling efficiency.

Results and Benefits: CustomerCare Inc. observes positive outcomes from the implementation, including reduced response times, improved customer satisfaction scores, and better insights into complaint trends through the system's reporting and analytics capabilities.

Future Enhancements: CustomerCare Inc. plans to continuously refine and enhance the system based on user feedback and emerging industry trends, ensuring its effectiveness in the long term.

Through this scenario-based case study, readers gain insights into the challenges, solutions, and benefits associated with implementing an Online Complaint Registration and Management System, providing a real-world context for understanding its significance and impact on organizational operations.

1.3 Purpose The use of this project. What can be achieved using this.

The Online Complaint Registration and Management System serves as a multifaceted tool with several offers several benefits and achievements for both organizations and their customers. The project's primary objectives and what can be achieved using it include:

The Online Complaint Registration and Management System serves as a multifaceted tool with several key purposes, aiming to address the needs of both organizations and their customers. The project's primary objectives and what can be achieved using it include:

1. **Efficient Complaint Handling:** The system streamlines the process of registering, tracking, and resolving complaints, leading to quicker response times and improved issue resolution.
2. **Enhanced Customer Satisfaction:** By providing a user-friendly platform for lodging complaints and transparently tracking their progress, the system enhances customer satisfaction levels, fostering positive relationships between organizations and their customers.
3. **Improved Operational Efficiency:** Automated assignment and escalation of complaints, along with real-time tracking and monitoring features, contribute to improved operational efficiency within organizations, reducing manual workload and minimizing errors.
4. **Data-Driven Decision Making:** The system's reporting and analytics capabilities provide valuable insights into complaint trends, resolution times, and customer feedback, enabling organizations to make informed decisions for process improvement and resource allocation.
5. **Transparency and Accountability:** Built-in communication channels facilitate seamless interaction between customers and staff, promoting transparency and accountability throughout the complaint handling process.
6. **Scalability and Adaptability:** The project's architecture, based on the MERN stack, offers scalability and adaptability to accommodate the evolving needs of organizations as they grow and change over time.

7. **Compliance and Risk Management:** By centralizing complaint data and implementing robust security measures, the system helps organizations comply with regulatory requirements and mitigate risks associated with complaint handling.

In essence, the purpose of the Online Complaint Registration and Management System is to empower organizations to effectively manage complaints, prioritize customer satisfaction, and drive continuous improvement in their operations. By leveraging modern technologies and best practices in complaint management, organizations can achieve greater efficiency, transparency, and customer-centricity in their service delivery.

Chapter 2

LITERATURE SURVEY

2.1 Existing problem Existing approaches or method to solve this problem

Existing Problem:

Traditional complaint registration and management processes are often manual, time-consuming, and prone to errors. They may involve paper-based forms, email communication, or phone calls, leading to inefficiencies, delays in resolution, and poor customer experiences. These methods can also lack transparency and make it challenging for organizations to track and analyze complaint data effectively.

Existing Approaches or Methods to Solve this Problem:

Several approaches or methods have been employed to address the challenges associated with complaint registration and management:

1. **Manual Systems:** Some organizations still rely on manual systems, such as paper-based forms or spreadsheets, to register and track complaints. While simple, these methods are often inefficient and prone to errors, especially as complaint volumes increase.
2. **Email and Phone Support:** Many organizations offer email or phone support for customers to lodge complaints. However, these channels can be slow, lack transparency, and result in inconsistent handling of complaints due to reliance on individual staff members.
3. **Ticketing Systems:** Ticketing systems, often used in customer service departments, allow organizations to track and manage complaints more effectively. These systems assign a unique ticket number to each complaint, enabling easy tracking and escalation as needed. However, they may lack integration with other systems and require manual data entry.
4. **Customer Relationship Management (CRM) Software:** CRM software includes complaint management features that allow organizations to track interactions with customers, including complaints. While CRM systems offer comprehensive customer data management capabilities, they may be complex and expensive to implement.
5. **Custom-Built Solutions:** Some organizations develop custom-built complaint management systems tailored to their specific needs. These systems can offer greater flexibility and integration with existing processes but require significant time and resources to develop and maintain.
6. **Third-Party Complaint Management Platforms:** There are third-party platforms and software solutions available that specialize in complaint management. These platforms often offer features such as automated workflows, real-time tracking, and reporting capabilities, providing organizations with a comprehensive solution without the need for in-house development.

While each approach has its benefits and limitations, the adoption of modern, digital complaint registration and management systems, such as the Online Complaint Registration and Management System proposed in this project, offers the potential to overcome many of the challenges associated with traditional methods. These systems leverage technology to streamline processes, improve transparency, and enhance the overall customer experience.

2.2 Proposed solution What is the method or solution suggested by you?

The proposed solution is the development of an Online Complaint Registration and Management System using Full Stack Development technologies, specifically the MERN (MongoDB, Express.js, React.js, Node.js) stack. This system will offer a comprehensive and efficient platform for organizations to streamline the process of handling complaints from customers.

Key components of the proposed solution include:

1. **User-friendly Interface:** The system will feature an intuitive and user-friendly interface for both customers and staff members to register, track, and manage complaints seamlessly.
2. **Secure Authentication:** User authentication will be implemented to ensure the security and confidentiality of complaint data, with role-based access controls for staff members.
3. **Complaint Registration Form:** Customers will have access to a complaint registration form where they can provide details such as complaint category, description, and any supporting documents.
4. **Real-time Tracking:** The system will enable real-time tracking of complaint status, allowing customers to monitor the progress of their complaints and receive updates as they are resolved.
5. **Automated Workflows:** Automated workflows will be implemented to assign complaints to appropriate staff members based on predefined criteria, ensuring efficient handling and escalation as needed.
6. **Communication Channels:** Built-in communication channels, such as messaging or email notifications, will facilitate seamless interaction between customers and staff throughout the complaint resolution process.
7. **Reporting and Analytics:** The system will offer robust reporting and analytics capabilities, providing insights into complaint trends, resolution times, and customer feedback for continuous improvement.
8. **Mobile Responsiveness:** The system will be designed with a mobile-responsive interface to ensure accessibility across various devices, enabling customers to register and track complaints conveniently.

9. **Security Measures:** Stringent security measures will be implemented to protect complaint data and ensure compliance with regulatory requirements, safeguarding against potential threats and breaches.

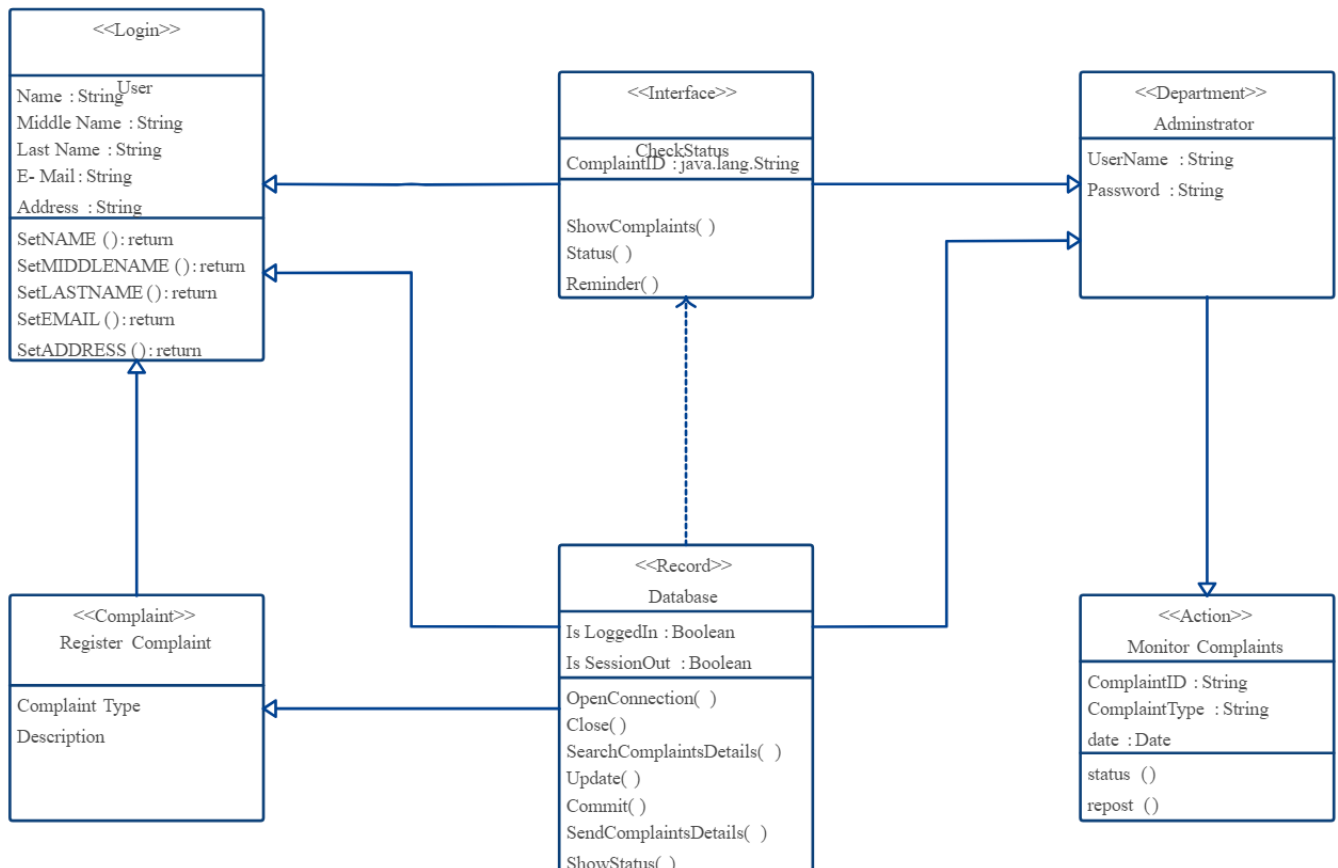
By leveraging the capabilities of the MERN stack and incorporating modern web development best practices, the proposed solution aims to address the inefficiencies and challenges associated with traditional complaint management methods, providing organizations with a comprehensive and efficient platform to prioritize customer satisfaction and streamline complaint handling processes.

Chapter 3

THEORITICAL ANALYSIS

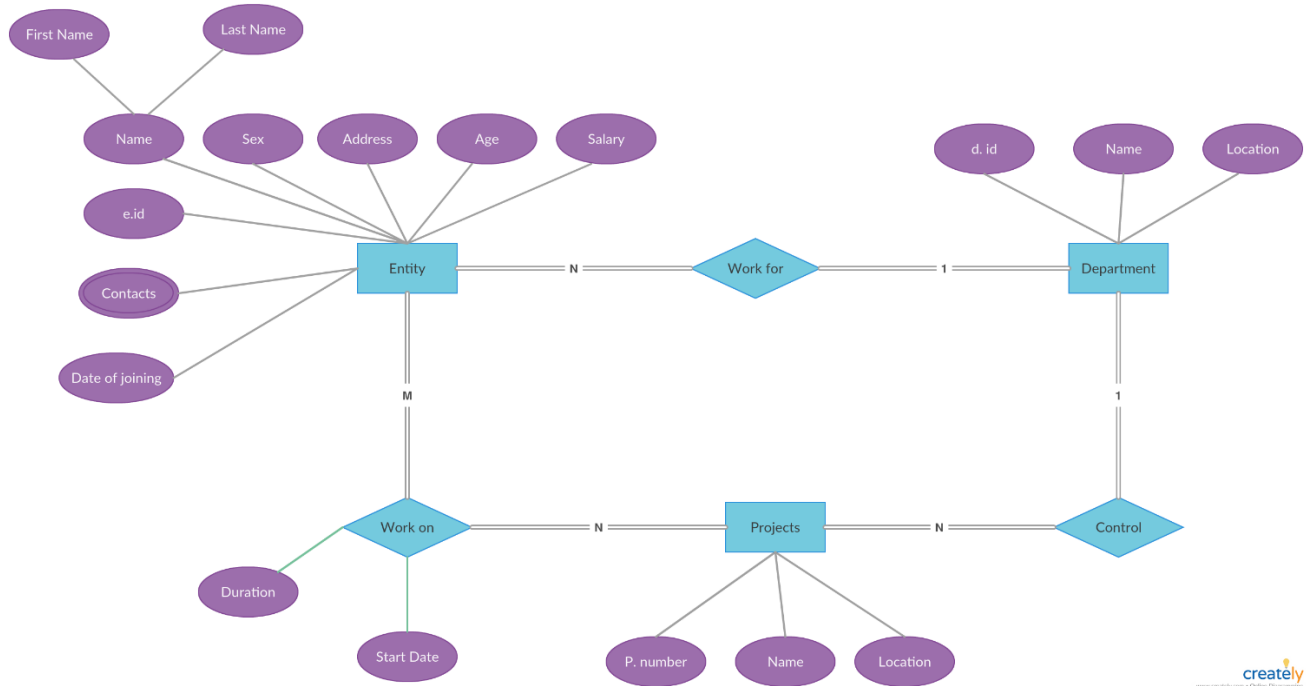
3.1 Block diagram Diagrammatic overview of the project.

TECHINICAL ARCHITECTURE:



This Online Complaint System is an invaluable tool for software development teams. It enables users to submit and track all types of complaints, identify issues and report on the status of resolutions. Teams can quickly and effectively log, track, escalate and close customer issues to ensure customer satisfaction. The system also allows users to set up customizable notifications and reports to keep everyone up to date on complaint resolution statuses. It helps teams identify problem areas and fix them proactively, reducing the number of complaints over time.

ER DIAGRAM:



3.2 Hardware / Software designing Hardware and software requirements

Hardware Requirements:

1. Server:

- High-performance server capable of running Node.js and MongoDB efficiently.
- Recommended specifications:
 - CPU: Multi-core processor (e.g., Intel Core i7 or AMD Ryzen series)
 - RAM: Minimum 8GB (16GB or higher recommended)
 - Storage: SSD for faster read/write operations

2. Network Infrastructure:

- Stable internet connection to ensure continuous access to the application.
- Firewall and security measures to protect against unauthorized access.

3. Client Devices:

- Desktop computers, laptops, tablets, and smartphones for accessing the web application.
- Internet browsers such as Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge.

Software Requirements:

1. Operating System:

- Server: Linux-based operating system (e.g., Ubuntu, CentOS)
- Client Devices: Compatible with major operating systems like Windows, macOS, Linux, Android, and iOS.

2. Development Tools:

- Text Editor or Integrated Development Environment (IDE) for coding (e.g., Visual Studio Code, Sublime Text, Atom).
- Version Control System (e.g., Git) for collaborative development and code management.

3. Backend:

- Node.js: JavaScript runtime environment for building server-side applications.
- Express.js: Web application framework for Node.js, providing tools and utilities for creating APIs and handling HTTP requests.
- MongoDB: NoSQL database for storing complaint data.
- Other Node.js modules and libraries as needed for authentication, data validation, and other functionalities.

4. Frontend:

- React.js: JavaScript library for building user interfaces, providing reusable UI components and efficient rendering.
- HTML5, CSS3, JavaScript: Standard web development technologies for structuring, styling, and adding interactivity to web pages.
- Axios or Fetch API for making HTTP requests to the backend server.
- Redux or Context API for state management, if needed.

5. Deployment:

- Web Server: NGINX or Apache for serving static files and proxying requests to the Node.js server.
- Deployment Platforms: Services like Heroku, AWS, DigitalOcean, or Azure for deploying and hosting the web application.
- Continuous Integration/Continuous Deployment (CI/CD) tools for automating the deployment process (e.g., Jenkins, CircleCI, GitHub Actions).

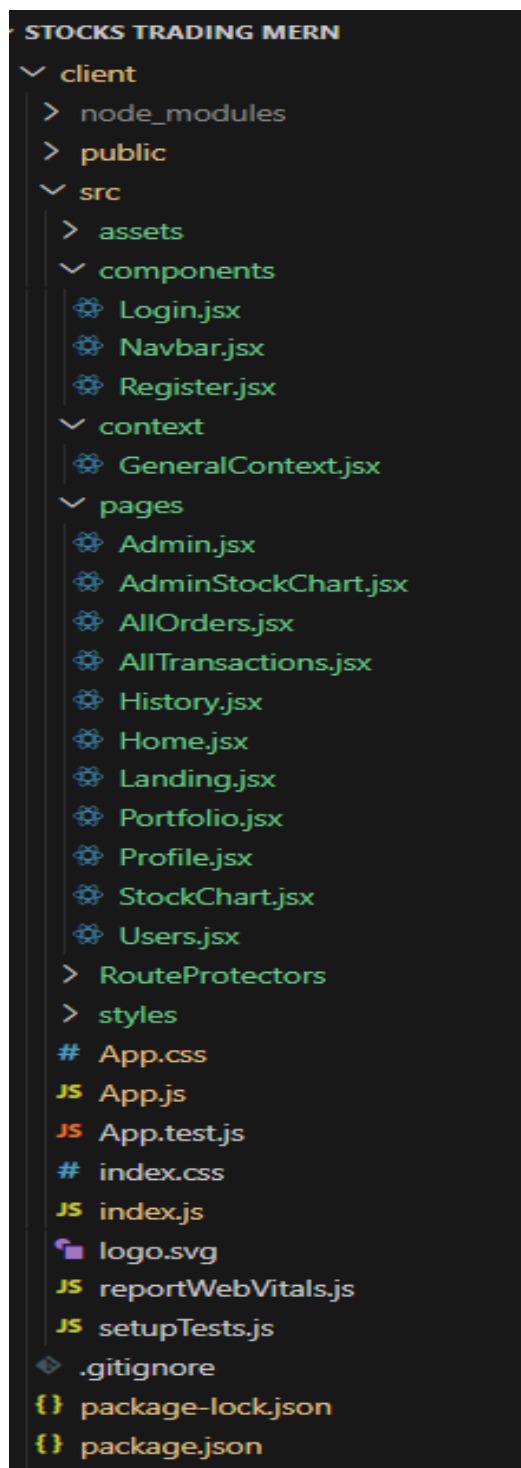
6. Security:

- SSL/TLS Certificate for encrypting data transmitted between the client and server.
- Security best practices for securing the application against common vulnerabilities (e.g., XSS, CSRF, SQL injection).
- Authentication mechanisms such as JSON Web Tokens (JWT) or OAuth for user authentication and authorization.

- Regular software updates and patches to address security vulnerabilities in dependencies and libraries.

These hardware and software requirements provide a foundation for developing and deploying the Online Complaint Registration and Management System using the MERN stack. Adjustments may be necessary based on specific project needs, scalability requirements, and organizational constraints.

3.3 Project structure:



The frontend structure assumes a React app and follows a modular approach. Here's a brief explanation of the main directories and files:

- **src/components:** This directory has minor components such as Login, Register, etc.,
- **src/pages:** The pages folder contains all the pages of the application like landing page, home page, etc.,

3.4 Application FLOW of your project

The application flow of the Online Complaint Registration and Management System can be outlined as follows:

1. **User Registration and Authentication:**
 - Users (both customers and staff members) register for an account or log in to the system using their credentials.
 - Authentication mechanisms verify user identity and grant access to the appropriate functionalities based on user roles (e.g., customer, support staff, administrator).
2. **Complaint Registration:**
 - Customers access the complaint registration form through the user interface.
 - Customers provide details about their complaint, including category, description, severity level, and any relevant attachments.
 - Upon submission, the complaint is stored in the database and assigned a unique identifier for tracking purposes.
3. **Complaint Assignment and Escalation:**
 - Automated workflows or manual processes assign complaints to appropriate staff members based on predefined criteria (e.g., category, severity).
 - Complaints are escalated to higher levels of support or management if they cannot be resolved within specified timeframes or require additional attention.
4. **Real-time Tracking and Monitoring:**
 - Customers and staff members can monitor the status of complaints in real-time through the user interface.
 - Updates, including assignment changes, status changes, and comments from staff members, are reflected immediately to provide transparency and visibility into the resolution process.
5. **Communication and Collaboration:**
 - Built-in communication channels facilitate seamless interaction between customers and staff members throughout the complaint resolution process.
 - Customers can provide additional information or clarify details, while staff members can request further details, provide updates, or request feedback.
6. **Resolution and Closure:**
 - Staff members work to resolve complaints efficiently and effectively, taking necessary actions such as investigating issues, communicating with relevant parties, and implementing solutions.
 - Once a resolution is reached, staff members update the complaint status accordingly and notify the customer of the outcome.
7. **Reporting and Analytics:**
 - The system provides reporting and analytics capabilities to track complaint trends, analyze resolution times, and gather customer feedback.
 - Reports and analytics dashboards offer insights into areas for improvement, performance metrics, and overall complaint management effectiveness.

8. Feedback and Continuous Improvement:

- Customers may provide feedback on the complaint resolution process, which can be used to drive continuous improvement initiatives.
- Organizations analyze feedback, complaint data, and performance metrics to identify opportunities for enhancing customer satisfaction and operational efficiency.

This application flow ensures a streamlined and transparent process for registering, tracking, and resolving complaints, ultimately improving customer satisfaction and organizational effectiveness.

Project Flow:

Milestone 1: Project Setup and Configuration:

Let's do project setup and configure it as per our requirements.

1. Create project folders and files:

Now, firstly create the folders for frontend and backend to write the respective code and install the essential libraries.

- Client folders.
- Server folders

2. Install required tools and software:

For the backend to function well, we use the libraries mentioned in the prerequisites. Those libraries includes

- Node.js.
- MongoDB.
- Bcrypt
- Body-parser

Also, for the frontend we use the libraries such as

- React Js.
- Material UI
- Bootstrap
- Axios

After the installation of all the libraries, the package.json files for the frontend looks like the one mentioned below.

```
{ } package.json M X
client > { } package.json > ...
1  {
2    "name": "client",
3    "version": "0.1.0",
4    "private": true,
5    "dependencies": {
6      "@testing-library/jest-dom": "^5.17.0",
7      "@testing-library/react": "^13.4.0",
8      "@testing-library/user-event": "^13.5.0",
9      "apexcharts": "^3.41.1",
10     "axios": "^1.4.0",
11     "bootstrap": "^5.3.1",
12     "react": "^18.2.0",
13     "react-apexcharts": "^1.4.1",
14     "react-dom": "^18.2.0",
15     "react-google-charts": "^4.0.1",
16     "react-icons": "^4.10.1",
17     "react-router-dom": "^6.14.2",
18     "react-scripts": "5.0.1",
19     "web-vitals": "^2.1.4"
20   },
```

After the installation of all the libraries, the package.json files for the backend looks like the one mentioned below.

```
{ } package.json X
server > { } package.json > ...
1  {
2    "name": "server",
3    "version": "1.0.0",
4    "description": "",
5    "type": "module",
6    "main": "index.js",
7    "scripts": {
8      "test": "echo \\\"Error: no test specified\\\" && exit 1"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "bcrypt": "^5.1.0",
15     "body-parser": "^1.20.2",
16     "cors": "^2.8.5",
17     "express": "^4.18.2",
18     "mongoose": "^7.4.2"
19   }
20 }
```


Milestone 2: Backend Development:

- **Set Up Project Structure:**
 - Create a new directory for your project and set up a package.json file using npm init command.
 - Install necessary dependencies such as Express.js, Mongoose, and other required packages.
- **Create Express.js Server:**
 - Set up an Express.js server to handle HTTP requests and serve API endpoints.
 - Configure middleware such as body-parser for parsing request bodies and cors for handling cross-origin requests.
- **Define API Routes:**
 - Create separate route files for different API functionalities such as authentication, stock actions, and transactions.
 - Implement route handlers using Express.js to handle requests and interact with the database.
- **Implement Data Models:**
 - Define Mongoose schemas for the different data entities like Bank, users, transactions, deposits and loans.
 - Create corresponding Mongoose models to interact with the MongoDB database.
 - Implement CRUD operations (Create, Read, Update, Delete) for each model to perform database operations.
- **User Authentication:**
 - Implement user authentication using strategies like JSON Web Tokens (JWT) or session-based authentication.
 - Create routes and middleware for user registration, login, and logout.
 - Set up authentication middleware to protect routes that require user authentication.
- **Handle new transactions:**
 - Allow users to make transactions to other users using the user's account id.
 - Update the transactions and account balance dynamically in real-time.
- **Admin Functionality:**
 - Implement routes and controllers specific to admin functionalities such as fetching all the data regarding users, transactions, stocks and orders.
- **Error Handling:**
 - Implement error handling middleware to catch and handle any errors that occur during the API requests.
 - Return appropriate error responses with relevant error messages and HTTP status codes.

Milestone 3: Database Development:

- Set up a MongoDB database either locally or using a cloud-based MongoDB service like MongoDB Atlas.
 - Create a database and define the necessary collections for users, transactions, stocks and orders.
 - Also let's see the detailed description for the schemas used in the database.
- 1. User Schema:**

- Schema: userSchema
- Model: 'User'
- The User schema represents the user data and includes fields such as username, email, balance, and password.
- It is used to store user information for registration and authentication purposes.
- The email field is marked as unique to ensure that each user has a unique email address.
- The balance field represents the amount in users account and it gets updated with transactions.

2. Transactions Schema:

- Schema: transactionsSchema
- Model: 'Transactions'
- The Transactions schema represents the transactions data such as user details, transaction type, amount, time, etc.,

3. Stocks Schema:

- Schema: stocksSchema
- Model: 'Stocks'
- The Stocks schema represents the stock data such as user details, stock details, stock exchange, stock price, etc.,

4. Orders Schema:

- Schema: ordersSchema
- Model: 'Orders'
- The Orders schema represents the stock order data and includes fields such as user details, stock details, total amount, order type, etc.,

For the connection of database use the code given below

```
const PORT = 6001;
mongoose.connect('mongodb://localhost:27017/Stocks', {
  useNewUrlParser: true,
  useUnifiedTopology: true,
})
.then(()=>{
```

```
  app.listen(PORT, ()=>{
    console.log(`Running @ ${PORT}`);
  });
}).catch((e)=> console.log(`Error in db connection ${e}`));
```

Change the URL string according to the MongoDB connection string from the Atlas(cloud) / MongoDB Compass(local). The schemas for the database are given below

```

JS Schema.js x
server > JS Schema.js > [⌘] transactionSchema > time
1  import mongoose from 'mongoose';
2
3  const userSchema = new mongoose.Schema({
4    username: { type: String, required: true },
5    email: { type: String, required: true, unique: true },
6    usertype: { type: String, required: true },
7    password: { type: String, required: true },
8    balance: { type: Number, default: 0 }
9  });
10
11 const transactionSchema = new mongoose.Schema({
12   user: { type: String, required: true },
13   type: { type: String, required: true },
14   paymentMode: { type: String, required: true },
15   amount: { type: Number, required: true },
16   time: [{ type: String }]
17 });
18
19 const stocksSchema = new mongoose.Schema({
20   user: { type: String },
21   symbol: { type: String },
22   name: { type: String },
23   price: { type: Number },
24   count: { type: Number },
25   totalPrice: { type: Number },
26   stockExchange: { type: String }
27 });
28
29 const ordersSchema = new mongoose.Schema({
30   user: { type: String },
31   symbol: { type: String },
32   name: { type: String },
33   price: { type: Number },
34   count: { type: Number },
35   totalPrice: { type: Number },
36   stockType: { type: String }, // intraday / delivery
37   orderType: { type: String }, // buy / sell
38   orderStatus: { type: String }
39 });
40
41 export const User = mongoose.model('users', userSchema);
42 export const Transaction = mongoose.model('transactions', transactionSchema);
43 export const Stock = mongoose.model('stocks', stocksSchema);
44 export const Order = mongoose.model('orders', ordersSchema);

```

Milestone 4: Frontend Development:

1. Setup React Application:

Bringing SB Stocks to life involves a three-step development process. First, a solid foundation is built using React.js. This includes creating the initial application structure, installing necessary libraries, and organizing the project files for efficient development. Next, the user interface (UI) comes to life. To start the development process for the frontend, follow the below steps.

- Install required libraries.
- Create the structure directories.

2. Design UI components:

Reusable components will be created for all the interactive elements you'll see on screen, from stock listings and charts to buttons and user profiles. Next, we'll implement a layout and styling scheme to define the overall look and feel of the application. This ensures a visually-appealing and intuitive interface. Finally, a navigation system will be integrated, allowing you to

effortlessly explore different sections of SB Stocks, like viewing specific stocks or managing your virtual portfolio.

3. Implement frontend logic:

In the final leg of the frontend development, we'll bridge the gap between the visual interface and the underlying data. It involves the below stages.

- Integration with API endpoints.
- Implement data binding.

Milestone 5: Project Implementation:

On completing the development part, we then run the application one last time to verify all the functionalities and look for any bugs in it.

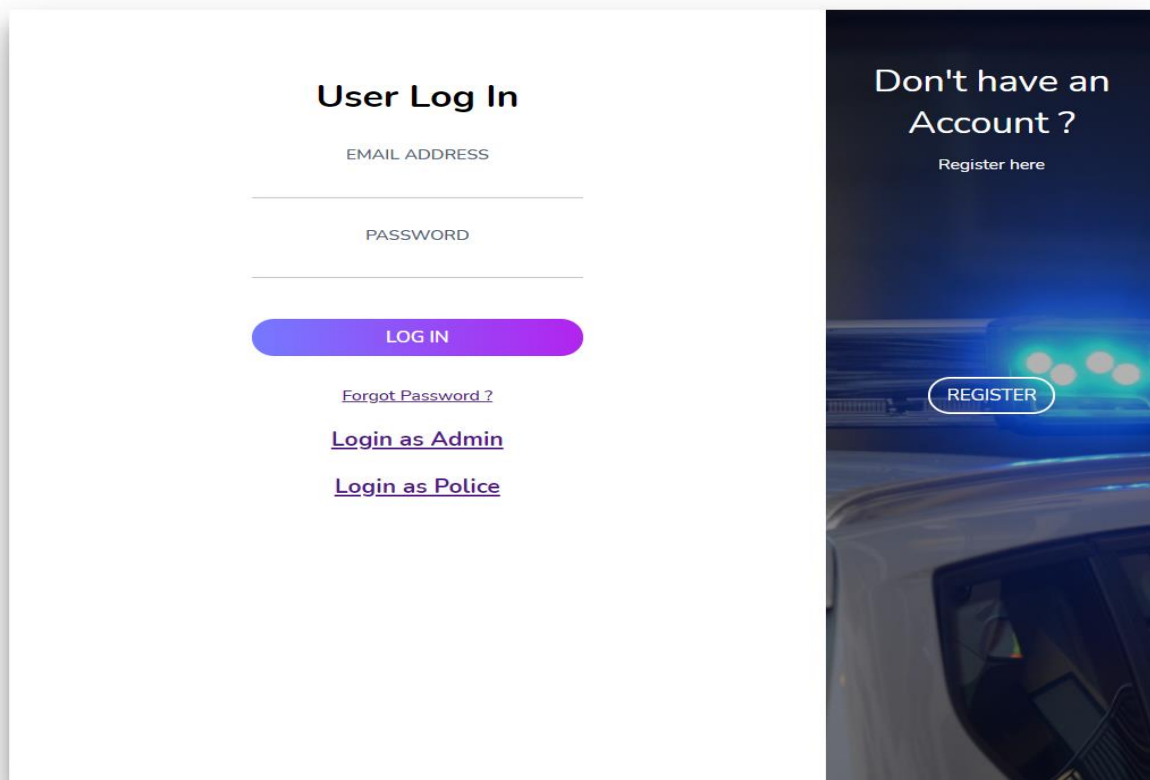
Chapter 4

RESULT

4.1 Final findings (Output) of the project along with screenshots.

On completing the development part, we then run the application one last time to verify all the functionalities and look for any bugs in it. The user interface of the application looks a bit like the one's provided below.

Landing page:



Home page:

If Already have
an Account?

Then just Log in

LOG IN

Registration

NAME

Date of Birth
dd-mm-yyyy

Select Gender
(Please select a state)

CONTACT

EMAIL

PASSWORD

CONFIRM PASSWORD

☐ I Agree to the terms and conditions

REGISTER NOW

All users (Admin):

Admin Log In

USERNAME

PASSWORD

LOG IN

[Login as Police](#)

[Login as User](#)

Please Enter
Your Email and
password to
Login into the
system

Police Login:

Police Log In

POLICE ID


PASSWORD

LOG IN


[Login as Admin](#)

[Login as User](#)

Please Enter
Your Police ID
and password
to Login into the
system



[Logout](#)
Government of India



india.gov.in
national portal of india

COMPLAINT REGISTRATION FORM

If you have witnessed an incident that files under police attribution, please use this online police FIR report form to signalize it. the police will review the report and take the appropriate action. Thank you for being a responsible citizen.

As per the prevailing laws, FIR of a major crime (cognizable crimes like theft, burglary, motor vehicle theft, accident, ci

[NEWS](#) [ABOUT US](#) [STATUS](#)

Incident type ☐ Collision
☐ Theft

Name of the person reporting the incident

First name..

Last name..

Phone

Mobile no..

Email

Email Address..

Gender

☐ Male ☐ Female ☐ Other

Date when incident took place

dd-mm-yyyy



Location of the incident

Street Address..

Pincode

City

City

State

(Please select a state) ▼

Country

India

Please describe the incident

Describe Incident Here..

How many people were involved ?

Upload Incident Proof

Choose File

No file chosen

Select ID Proof

☐ Aadhar Card ☐ Pan Card ☐ VoterID

ZZ

Enter ID Number

Upload ID

Choose File

No file chosen

☐ I certify that the information provided by me in above form is accurate and correct as per best of my knowledge. I understand that withholding of information or giving false information is a criminal offense and may result in legal action against me.

Submit

Disclaimer: As per the prevailing laws, FIR of a major crime (cognizable crimes like theft, burglary, motor vehicle theft, accident, chain-snatching, assault, rape, murder, attempt to commit murder, robbery, dacoity, extortion etc) can only be registered at a Police Station. Please contact your nearest Police Station for the same.

Chapter 5

ADVANTAGES & DISADVANTAGES

5.1 List of advantages and disadvantages of the proposed solution

Advantages of the Online Complaint Registration and Management System:

1. **Enhanced Customer Satisfaction:** The system provides a streamlined process for registering and resolving complaints, leading to improved customer satisfaction and loyalty.
2. **Transparency and Accountability:** Real-time tracking and communication features foster transparency and hold organizations accountable for their response times and resolution actions.
3. **Improved Operational Efficiency:** Automated workflows and centralized data management streamline complaint handling processes, reducing manual effort and operational costs.
4. **Data-Driven Decision Making:** Reporting and analytics capabilities offer valuable insights into complaint trends, enabling organizations to make informed decisions for process optimization and resource allocation.
5. **Secure and Confidential:** The system employs security measures to protect complaint data and ensure compliance with regulatory requirements, safeguarding against potential breaches and unauthorized access.

Disadvantages of the Online Complaint Registration and Management System:

1. **Initial Implementation Challenges:** Implementing the system may require significant time, resources, and adjustments to existing processes, potentially causing disruption and resistance from staff members.
2. **Technical Dependencies:** The system relies on stable internet connectivity and compatible devices for access, posing challenges in environments with unreliable internet connections or outdated technology infrastructure.
3. **User Adoption Issues:** Users may encounter difficulties navigating the system or be resistant to change, requiring comprehensive training and support to encourage adoption and maximize utilization.
4. **Overreliance on Technology:** Organizations may become overly dependent on the system, neglecting interpersonal communication and problem-solving skills, which are crucial for effective complaint resolution.
5. **Privacy Concerns:** Collecting and storing complaint data raises privacy concerns, necessitating strict adherence to data protection regulations and ethical practices to ensure confidentiality and trust.

Chapter 6

APPLICATIONS

6.1 The areas where this solution can be applied

The Online Complaint Registration and Management System has various applications across different industries and sectors. Some of the key applications include:

1. **Customer Service Departments:** The system can be used by customer service departments in industries such as telecommunications, banking, retail, and healthcare to efficiently handle and resolve customer complaints.
2. **Government Agencies:** Government agencies can utilize the system to manage complaints related to public services, such as transportation, utilities, healthcare, and education, ensuring timely responses and effective resolution.
3. **Educational Institutions:** Schools, colleges, and universities can employ the system to manage student complaints regarding academic matters, facilities, administrative issues, and student services.
4. **Healthcare Facilities:** Hospitals, clinics, and healthcare organizations can use the system to register and address patient complaints regarding medical care, staff behavior, facilities, and billing issues.
5. **Financial Institutions:** Banks, insurance companies, and financial service providers can utilize the system to handle customer complaints related to transactions, account management, loan processing, and financial products.
6. **Retail and E-commerce:** Retailers and e-commerce companies can implement the system to manage customer complaints about product quality, delivery delays, returns, refunds, and customer service interactions.
7. **Hospitality Industry:** Hotels, restaurants, and travel agencies can utilize the system to manage guest complaints regarding accommodations, reservations, service quality, and facilities.
8. **Manufacturing and Supply Chain:** Manufacturing companies can use the system to track and address complaints related to product defects, shipping delays, supply chain disruptions, and warranty issues.
9. **Utilities and Energy Companies:** Utility providers, such as electricity, water, and gas companies, can employ the system to manage complaints about service disruptions, billing discrepancies, meter readings, and infrastructure maintenance.
10. **Human Resources Departments:** Organizations can utilize the system internally within their human resources departments to manage employee complaints related to workplace harassment, discrimination, grievances, and employment benefits.

Chapter 7

CONCLUSION

7.1 Conclusion summarizing the entire work and findings.

In conclusion, the Online Complaint Registration and Management System presents a transformative solution for organizations to elevate their customer service standards and operational efficiency. By streamlining complaint handling processes, fostering transparency, and harnessing data-driven insights, the system enables organizations to not only address customer concerns effectively but also proactively identify opportunities for improvement. The system's scalability and adaptability ensure its relevance across various industries and sectors, catering to the diverse needs of organizations worldwide. Looking ahead, the future scope of the system holds immense potential for further innovation, including integration with emerging technologies, advanced analytics capabilities, and continuous enhancement of user experiences. As organizations continue to prioritize customer-centricity and embrace digital transformation, the Online Complaint Registration and Management System remains a strategic enabler for driving positive customer outcomes, fostering long-term loyalty, and maintaining a competitive edge in today's dynamic marketplace.

Chapter 8

FUTURE SCOPE

Enhancements that can be made in the future.

The future scope of the Online Complaint Registration and Management System is promising, with several opportunities for further development and expansion. Some of the potential areas of future scope include:

1. **Integration with AI and Machine Learning:** Incorporating artificial intelligence (AI) and machine learning (ML) algorithms can enhance the system's capabilities for automatic categorization and prioritization of complaints, sentiment analysis of customer feedback, and prediction of potential issues before they escalate.
2. **Advanced Analytics and Predictive Modeling:** Enhancing the system's reporting and analytics capabilities can enable organizations to derive deeper insights from complaint data, including predictive modeling to anticipate customer needs, identify emerging trends, and proactively address issues before they arise.
3. **Omni-channel Support:** Expanding the system to support multiple communication channels, such as social media, chatbots, and voice assistants, can provide customers with greater flexibility in registering complaints and receiving support, while also ensuring consistent and seamless experiences across channels.
4. **Enhanced Security and Compliance:** Strengthening the system's security measures and compliance features to meet evolving regulatory requirements and industry standards can help organizations maintain the confidentiality and integrity of complaint data, safeguarding against potential breaches and legal liabilities.
5. **Customer Feedback Integration:** Integrating the system with customer feedback mechanisms, such as surveys and ratings, can enrich complaint data with additional context and insights, enabling organizations to gain a holistic view of customer sentiment and satisfaction levels.
6. **Mobile Application Development:** Developing mobile applications for iOS and Android platforms can extend the reach of the system, allowing customers to register complaints and track their status on-the-go, while also providing staff members with greater flexibility in managing complaints remotely.
7. **Customer Relationship Management (CRM) Integration:** Integrating the system with existing CRM platforms can facilitate seamless data exchange and collaboration between complaint management and other customer-facing departments, enabling organizations to deliver more personalized and proactive customer service.

8. **IoT Integration for Proactive Monitoring:** Leveraging Internet of Things (IoT) devices for proactive monitoring of products and services can enable organizations to identify potential issues in real-time and preemptively address them, minimizing the occurrence of complaints and enhancing overall customer satisfaction.
9. **Continuous User Training and Support:** Providing ongoing user training and support to both customers and staff members can ensure maximum utilization and adoption of the system's features, while also fostering a culture of continuous improvement and innovation within the organization.