Customer Query Management System

Presented By:
Mohit Kumar
B.Tech (C.S.E)
4th Year
(2104500100036)

CONTENT

- Introduction to SoftPro India Pvt. Ltd

- Role and Responsibility: MERN Intern

- Project Overview : Customer Query Management System

- Web Development

- Methodology

- Results and Discussion

- Future Enhancement

- Conclusion

- References

INTRODUCTION TO SOFTPRO INDIA PVT. LTD.

Softpro India Computer Technologies (P) Ltd is a complete Technology Services and Solutions provider to the clients. It is an ISO 9001:2015 Certified Company, incorporated in the year 2004, by Technocrats from IIT Kanpur and IET Lucknow.

The Company is devoted to provide Software Solutions to patrons in verticals like Manufacturing, IT Services, Pharmaceutical and Education Sector. Softpro is an acclaimed member of UPDESCO which facilitates software development in government sectors in UP.

Softpro Learning Centre: This vertical is the training division of the company commenced in 2008 with a perception to bridge the Technology Gap prevalent between students and professionals. Softpro Learning Centre has embarked itself as one of the most propitious centres for learning across UP and nearby states.

ROLE AND RESPONSIBILITY: MERN INTERN

Primary Focus: Frontend And Backend development of CQMS web project.

Key Responsibilities:

- **UI/UX Implementation:** Contributed to the design to create intuitive and accessible user interfaces.

- **Responsive Design:** Ensured optimal user experience across desktops, tablets, and mobile devices.

- **Interactive Elements:** Added dynamic features using JavaScript to enhance user engagement.

- **Performance Optimization:** Improved website load times through code optimization and efficient resource management.

PROJECT OVERVIEW: CUSTOMER QUERY MANAGEMENT SYSTEM

Introduction:

The Customer Query Management System (CQMS) is a web-based application developed to streamline the process of handling customer queries. In today's competitive market, customer satisfaction is crucial, and the CQMS aims to enhance the efficiency of customer support teams by automating and managing the lifecycle of customer queries.

Developed using the MERN stack, the system allows for real-time interaction between customers and support teams, ensuring that queries are resolved quickly and efficiently. The system includes features such as query submission, tracking, real-time notifications, and detailed analytics.

Objective:

1. To reduce response times by automating the process of query handling .

2. To providing a user-friendly interface for customers to submit and track their queries.

3. To send real-time updates and notifications to customers regarding their queries.

MERN TECHNOLOGY OVERVIEW

Frontend Technologies:

• **React JS**

React.js is a frontend library used to build the user interface of CQMS.

> • **Component-Based Architecture:** React allows for the development of reusable UI components, reducing development time and improving maintainability.

> • **Virtual DOM**: React's virtual DOM optimizes rendering, providing a fast and responsive user experience.

> • **State Management:** State management with React and Redux ensures that the application's data flow is predictable and manageable.

Backend Technologies:

- **Node.js:** Utilized for server-side scripting, Node.js enabled the creation of a scalable and efficient backend capable of handling multiple simultaneous requests, managing sessions, and interacting with the database.

- **Express.js:** A framework for Node.js that simplified server-side development by managing routing, middleware, and creating RESTful APIs for user operations.

- **MongoDB:** Employed as the database for storing user data, MongoDB's NoSQL structure offered flexibility for managing diverse data types and scalability for handling large volumes of unstructured data.

METHODOLOGY

- **System Design:** The system architecture was designed to ensure scalability and flexibility. Technologies like MongoDB, Express.js, React.js, and Node.js were chosen for their interoperability and performance.

- **Implementation:** The implementation phase involved developing the backend RESTful APIs, the frontend user interface, and integrating the system with MongoDB.

- **Deployment:** The system was deployed on a cloud platform (frontend on <u>Vercel</u> and backend on <u>Render</u>, making it accessible to users across different geographical locations.
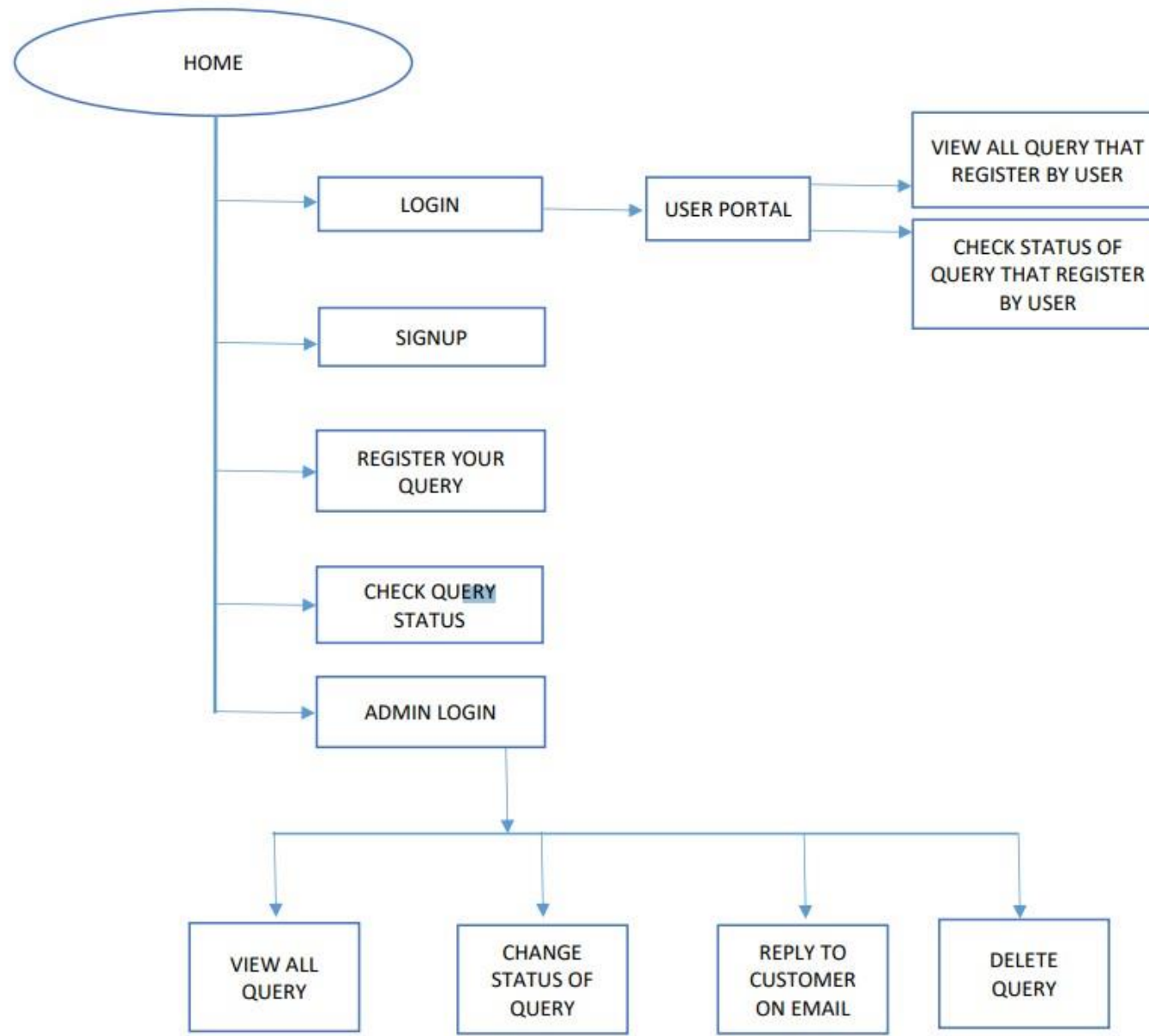
FIG. 1: Working Module Of CQMS

RESULTS AND DISCUSSION

The Customer Query Management System was successfully developed and deployed, meeting all the initial project objectives. The system's real-time notification feature has significantly improved response times, and the user interface has been well-received for its simplicity and ease of use

System Performance:

- **Query Handling:** The system handles multiple simultaneous queries efficiently, with minimal lag or downtime.

- **User Satisfaction:** Initial user feedback indicates a high level of satisfaction with the system's ability to provide timely updates on query status.

- **Scalability**: The system architecture supports scaling, with MongoDB's sharding feature allowing for the addition of more database servers as needed.

Challenges:
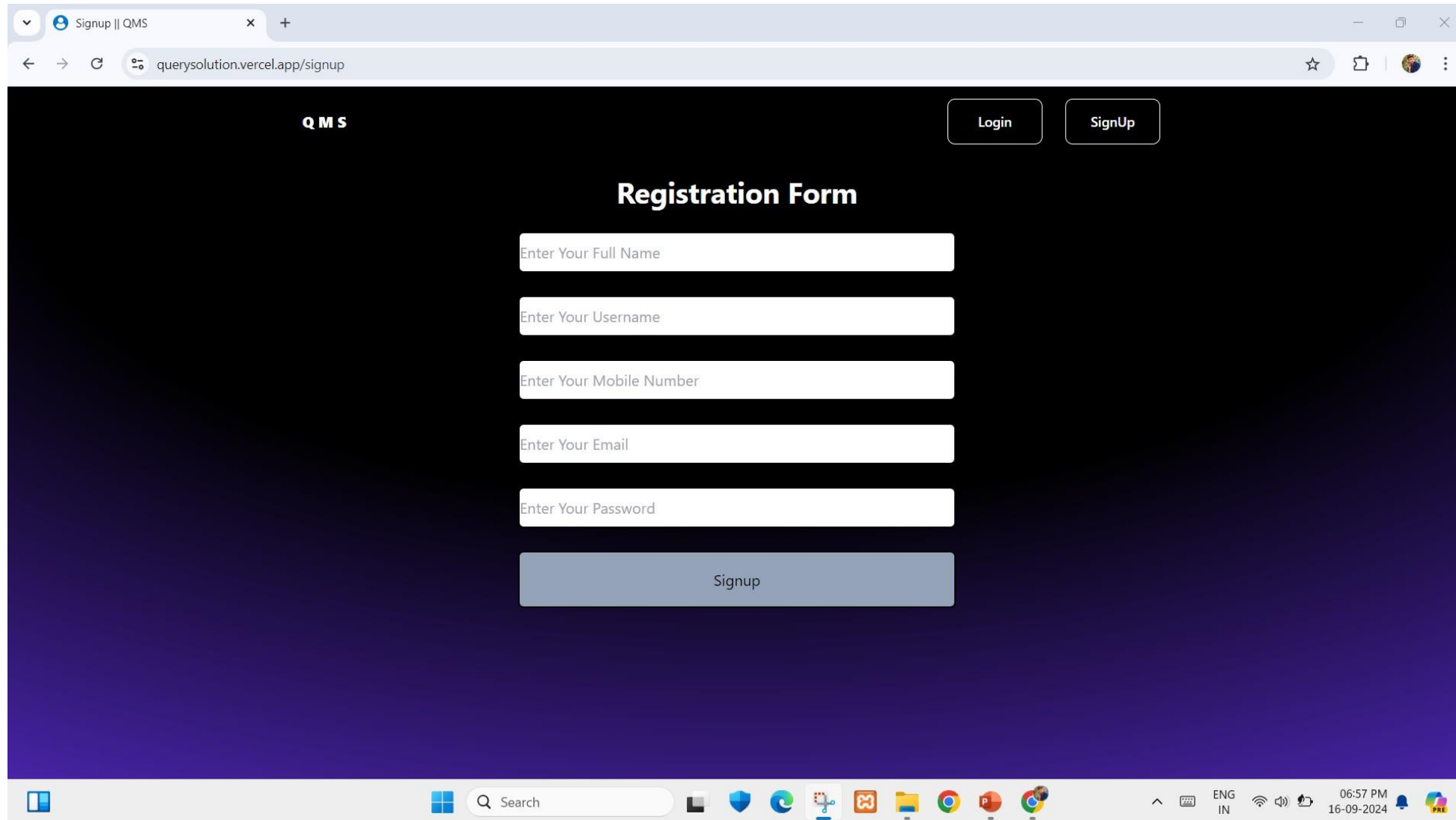
- **Data Security:** Ensuring data security and privacy was a challenge, particularly with the handling of sensitive customer information. The implementation of JWT and encrypted data storage addressed these concerns.

Sample Screenshots:
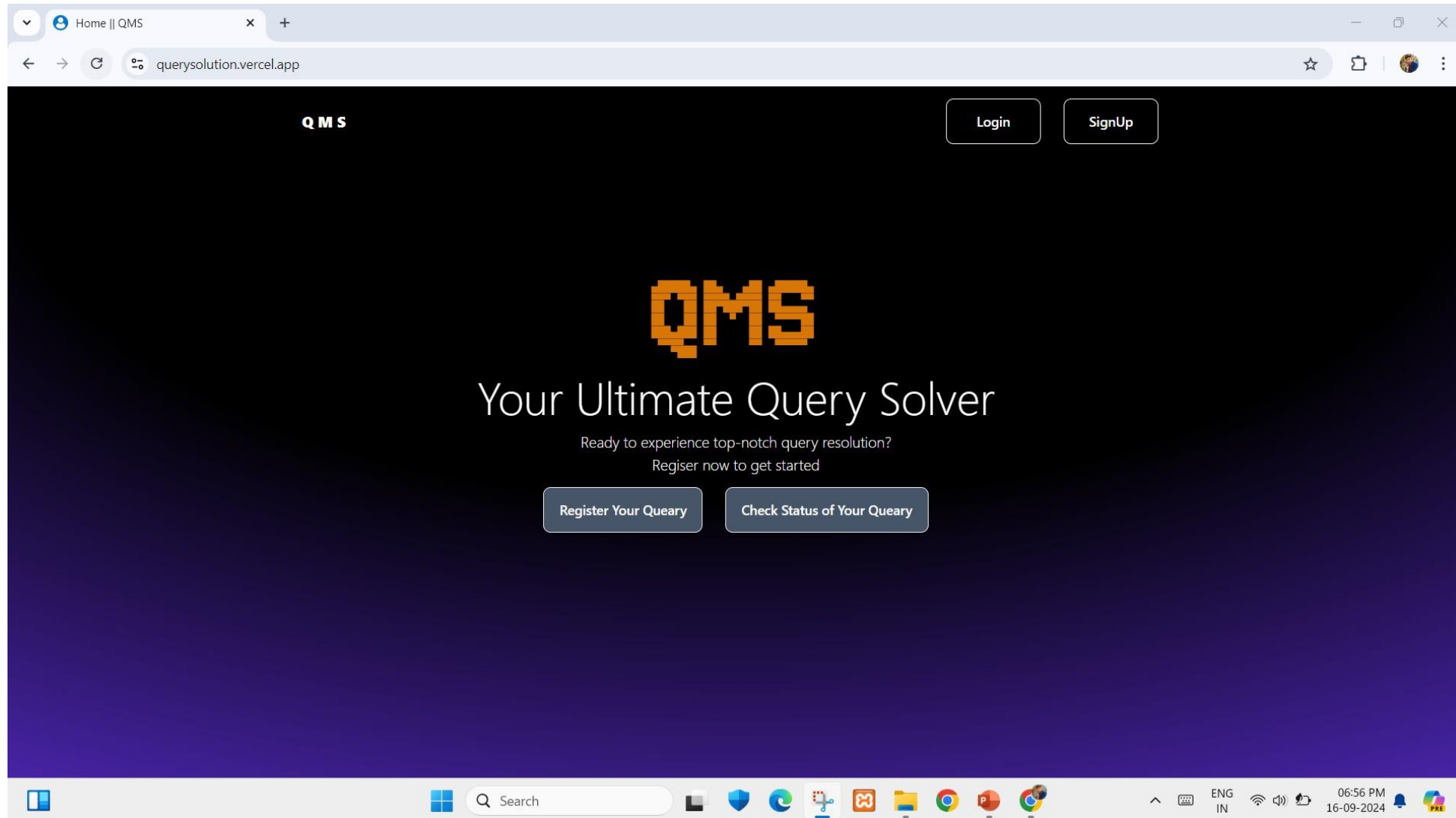
Home Page

Sign-up Page: Showcasing secure and user-friendly account create.
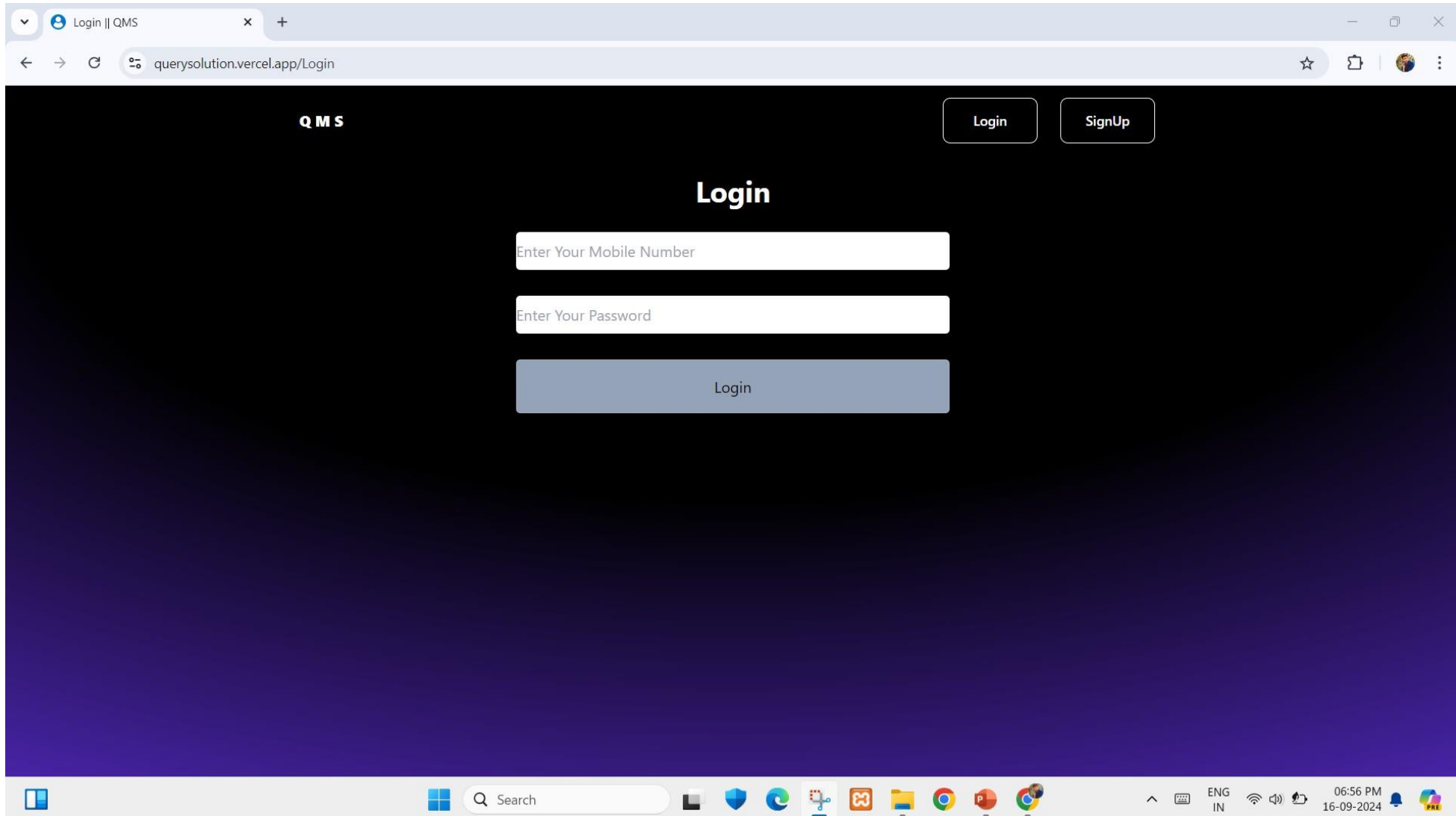
Sign-up Page

Email received to user: User gets email after successful Sign-Up.
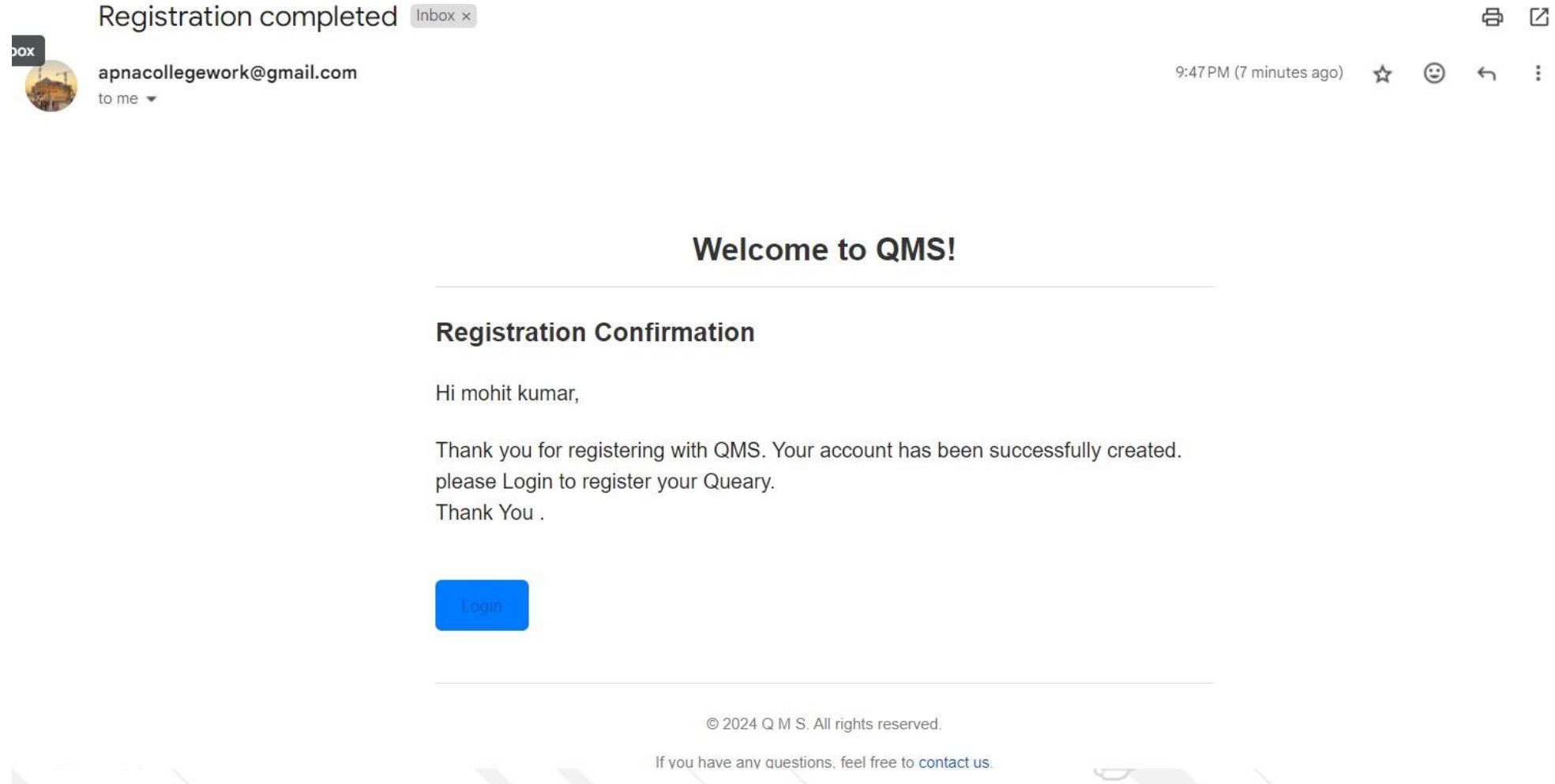
Sign-Up Confirmation Email

Sign-in Page: Showcasing secure and user-friendly account access.

Sign-in Page

User Portal: After Successful login user can register their query and see all queries that are registered by this mobile number.
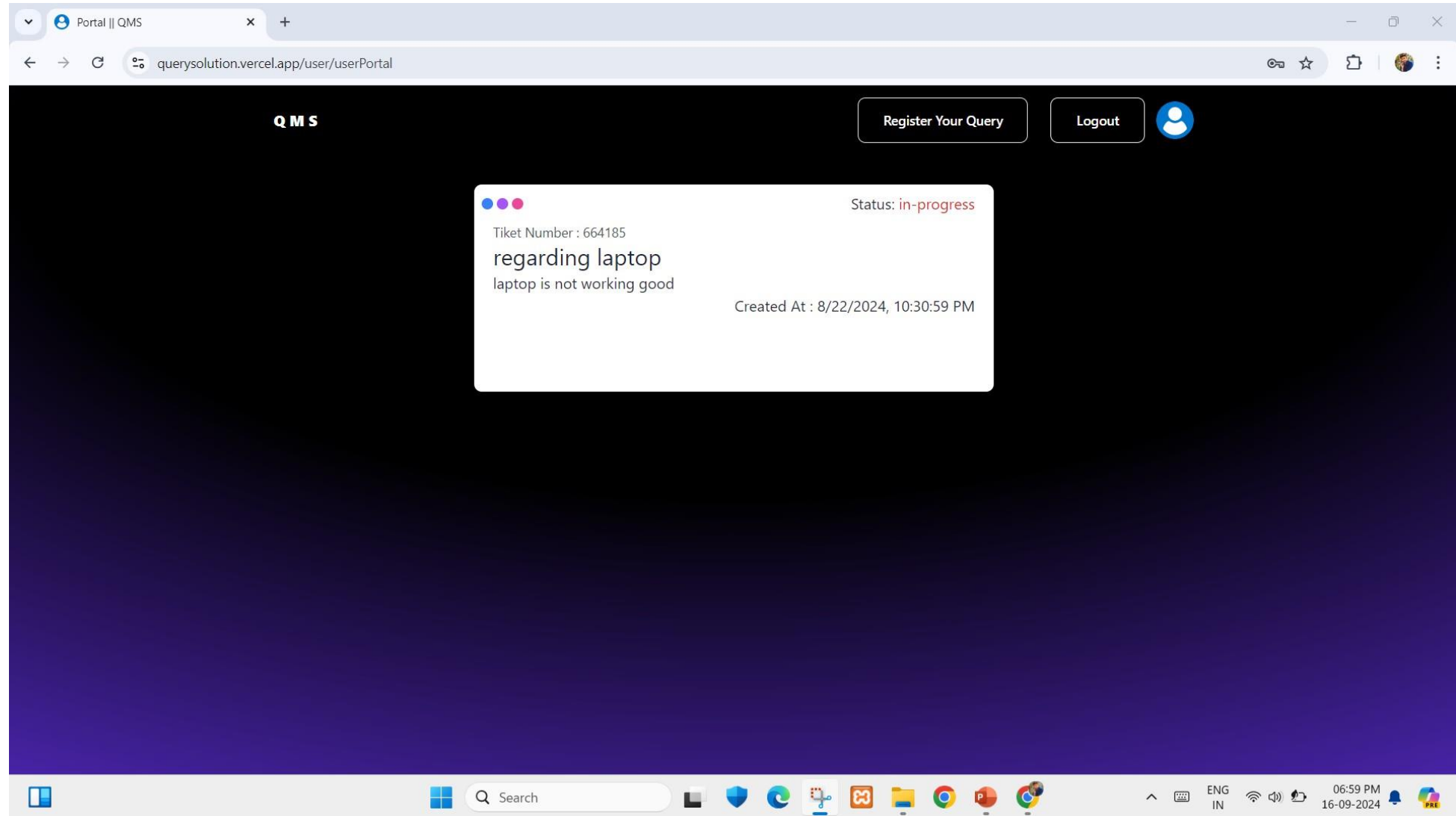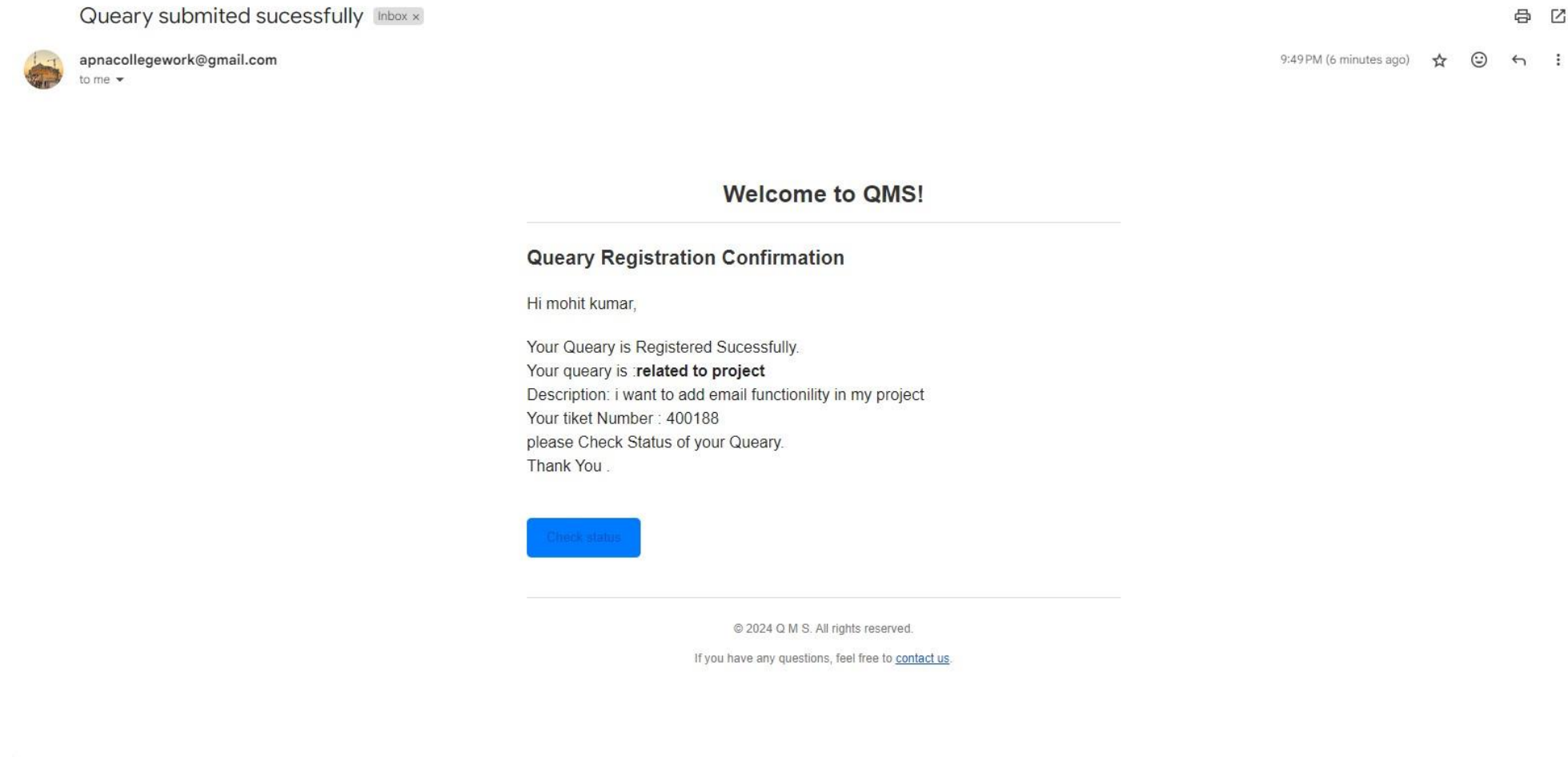

User Portal

Register Query Page : Here user can register their query and gets an mail through Admin.
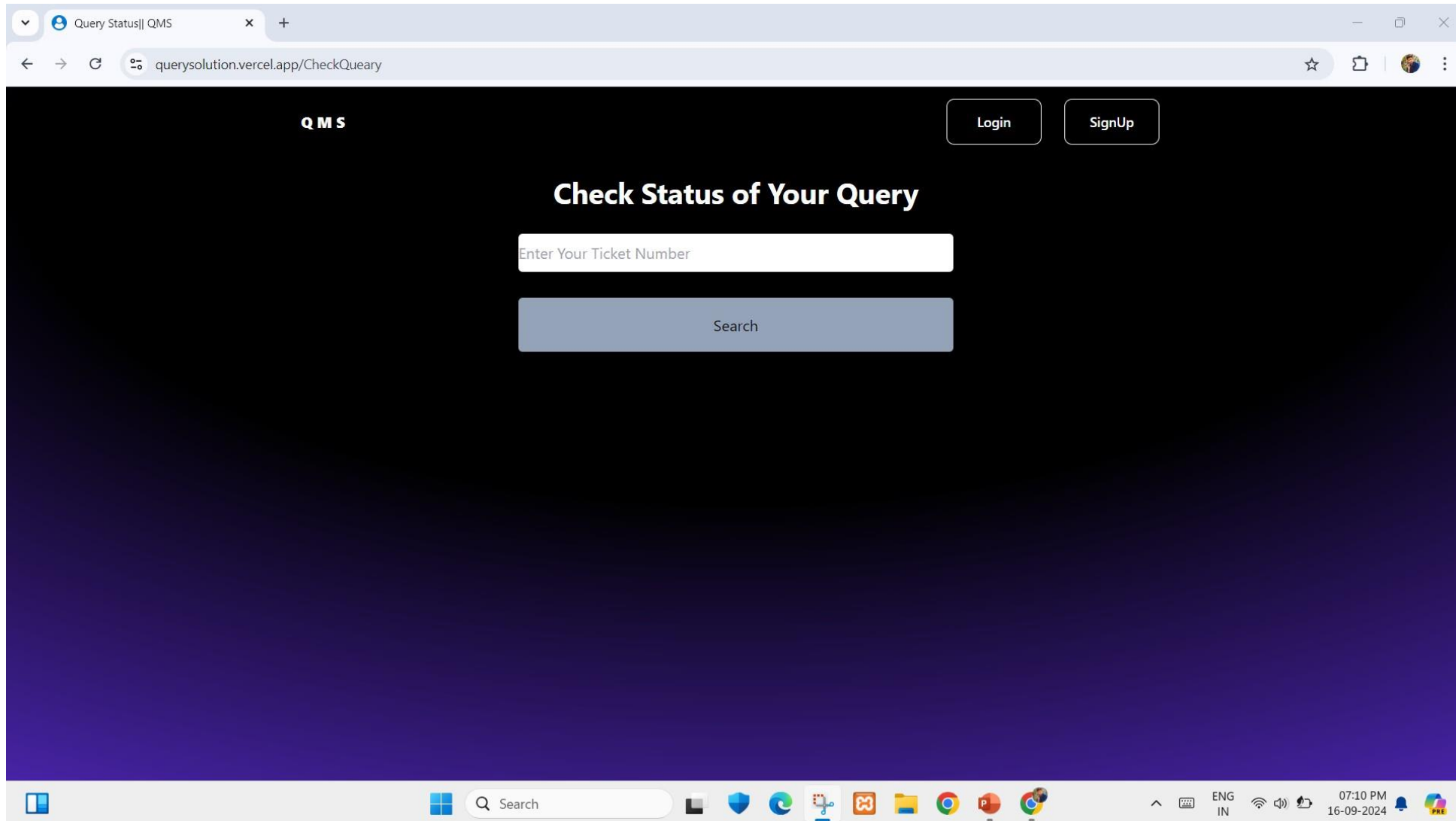

Register Query Page

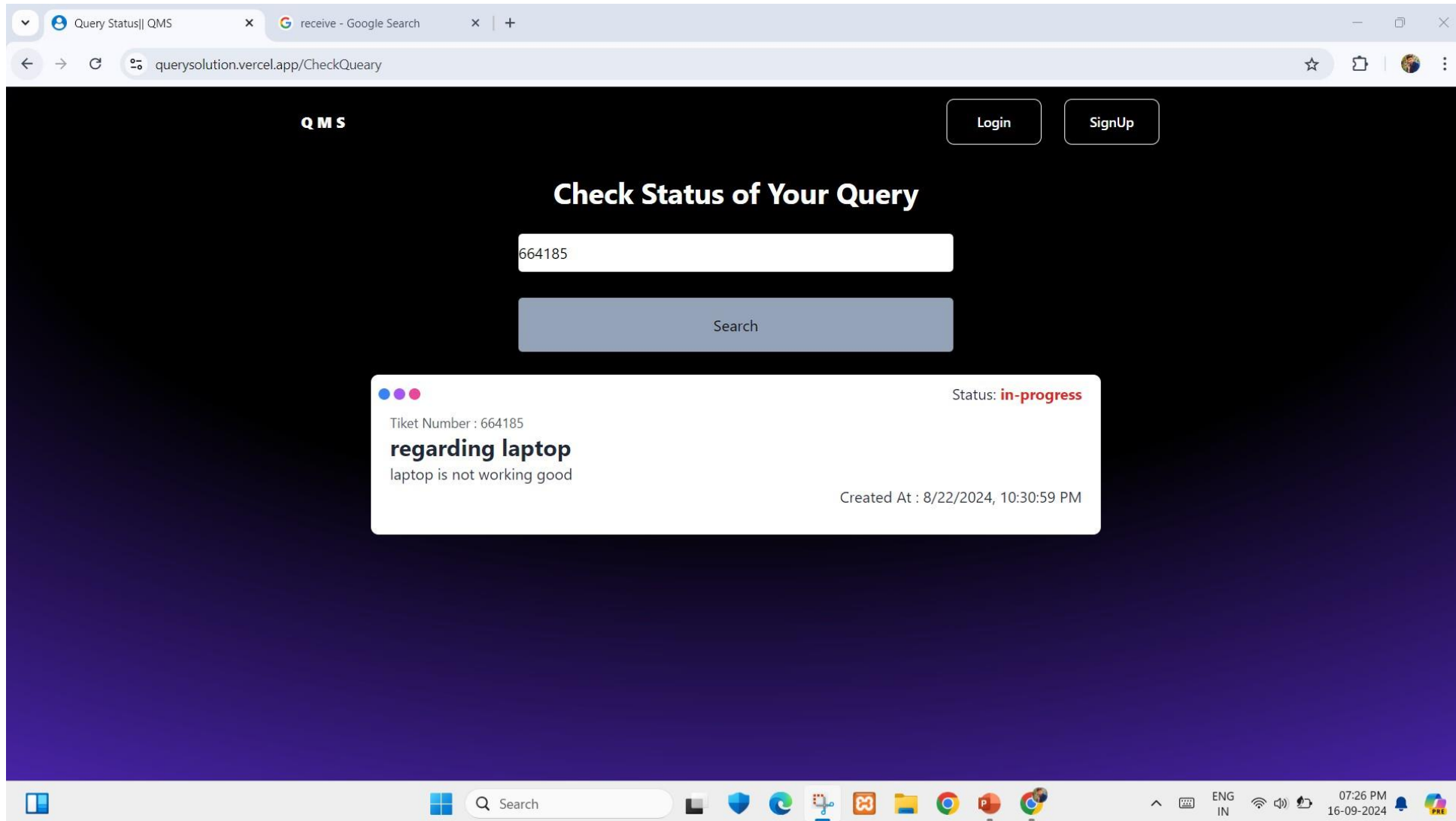**Email received to user**: User gets email after successful register Query .

Query Registration Confirmation Email

check Status Page : Here user can check status of that query by using ticket Number which is send through mail.

Register Query Page

check Status Page : Here user can check status of that query by using ticket Number which is send through mail.

Register Query Page

Admin login Page : here is the admin login Page from where admin can access to the portal.

Admin Login Page

Admin Portal Page : here is the admin Panel from where admin can mange the query and response them.

Admin Portal
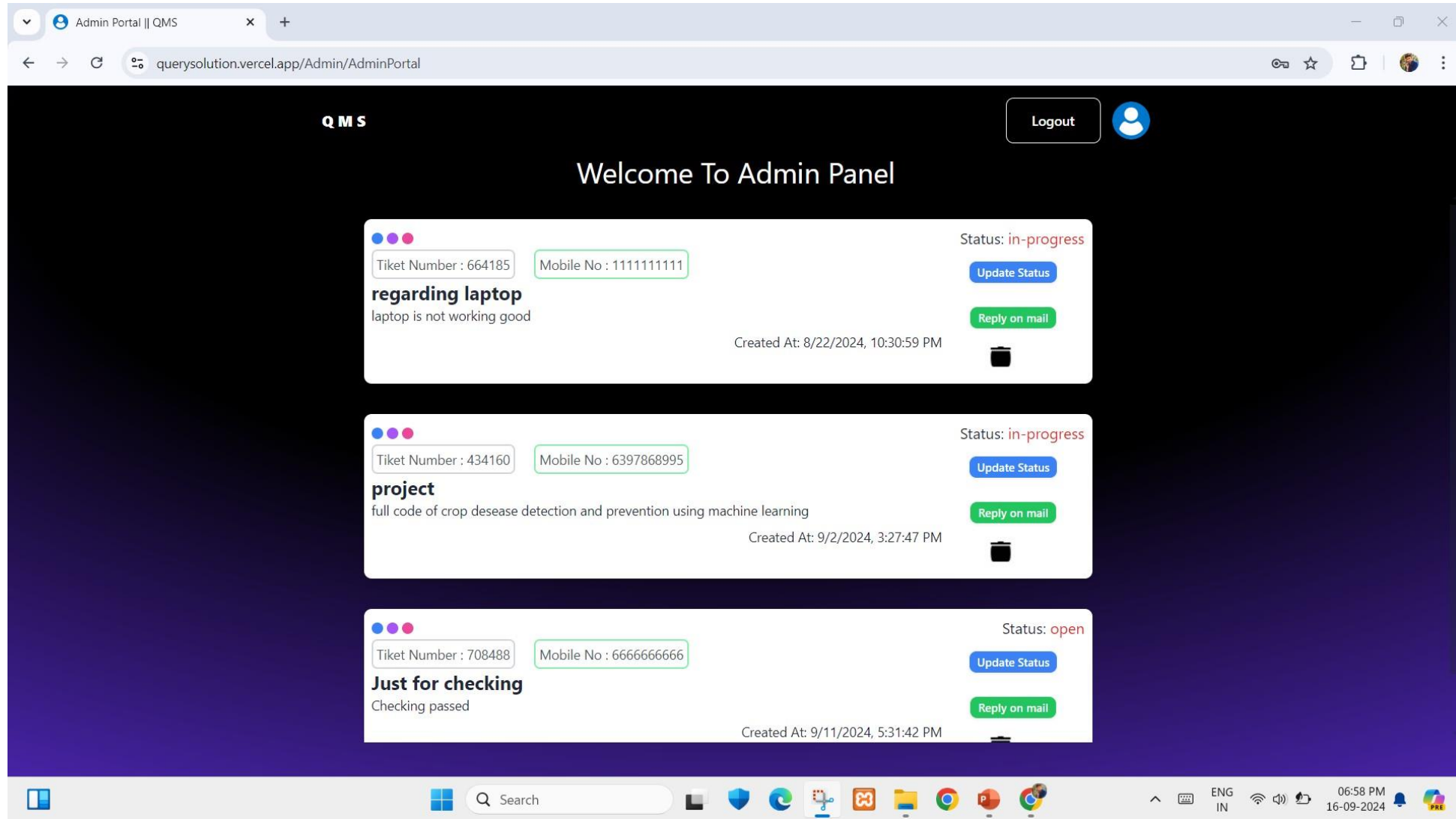
Email received to user: Agent /admin can connect to the user by using mail or on Call to provide the solution .

← ☐ ⓘ 🗑 ✉ 🗂 ⋮   1 of 564 ‹ › ⌨ ▾

regarding your queary regarding related to project Inbox ×   🖨 ↗

apnacollegework@gmail.com   9:52 PM (3 minutes ago) ☆ ☺ ↩ ⋮
to me ▾

Welcome to QMS!

Agent Response

Hi mohit kumar,

Your queary is :**related to project**
Description : i want to add email functionility in my project
Your tiket Number : 400188
Agent Reply: you can use nodemailer to send mail to user
Thank You .

© 2024 Q M S. All rights reserved.

If you have any questions, feel free to contact us.

↩ Reply   → Forward   ☺

Agent Response on Email

FUTURE ENHANCEMENT

To further enhance the Customer Query Management System, several improvements are planned:

- **AI-Powered Query Resolution :**

  - **Chatbots**: Integrating AI-powered chatbots to handle common customer queries automatically.

- **Mobile Application Development :**

  - **Cross-Platform App**: Developing a mobile application using React Native, making the system accessible on both Android and iOS platforms.

- **Enhanced Security Features:**

  - **Two-Factor Authentication**: Implementing two-factor authentication to improve user account security.

CONCLUSION

- **Learning Experience:**

  - Gained extensive hands-on experience in full-stack web development.

  - Developed strong problem-solving and project management skills through real-world application.

- **Project Impact:**

  - Delivered a comprehensive and modern web solution aligning with client needs and industry standards.

- **Future Aspirations:**

  - Eager to apply acquired skills to upcoming projects and continue advancing in web development and design.

REFERENCES

- **React Documentation:** https://legacy.reactjs.org/docs/getting-started.html

- **Node.js Documentation :** https://nodejs.org/docs/latest/api/

- **Express.js Documentation:** https://expressjs.com/en/5x/api.html

- **MongoDB Documentation:** https://www.mongodb.com/docs

- **Project link:** https://querysolution.vercel.app/

THANK YOU