

REPORT ON
15 WEEKS OF INTERNSHIP

Carried out at

CLOVER BAY TECHNOLOGIES

Submitted to

NMAM INSTITUTE OF TECHNOLOGY, NITTE
(An Autonomous Institution under VTU, Belagavi)

In partial fulfilment of the requirements for the award of the

Degree of Bachelor of Engineering in
Information Science & Engineering

by

GAURAV SHENOY K
4NM21IS216

Under the guidance of

Mr. Manjunath BS
(CTO)

CERTIFICATE

*This is to certify that the “Internship report” submitted by Gaurav Shenoy K bearing USN 4NM21IS216 of VIII semester B.E., a Bonafide student of NMAM Institute of Technology, Nitte, has undergone 15 weeks of internship at Clover Bay technologies during **January 6 2025 to April 19 2025** fulfilling the requirements for the award of degree of Bachelor of Engineering in **Information Science & Engineering** at NMAM Institute of Technology, Nitte.*

Name and Signature of Mentor

Signature of HOD

CERTIFICATE

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Clover bay Technologies for providing me with the opportunity to pursue my internship and gain practical exposure to the industry. I am especially thankful to Mr. Manjunath B S, Chief Technology Officer of Clover bay Technologies, for his constant guidance, valuable insights, and encouragement throughout the internship period.

I also extend my heartfelt thanks to the entire team at Cloverbay Technologies for their support, cooperation, and for creating a positive learning environment. Their assistance and feedback have been instrumental in the successful completion of this project.

TABLE OF CONTENTS

Content	Page no.
Abstract	6
Problem statement	7
Objectives	8
Methodology	9
Implementation	10-11
Conclusion	12
References	13

ABSTRACT

The Ticketing System is a web-based application designed to simplify and automate the process of managing customer queries, technical issues, and service requests within an organization. During my internship, I developed this project to address the common challenges faced by support teams, such as scattered communication, lack of centralized tracking, and delayed resolution of reported problems.

This system enables users to create tickets by submitting their issues through an intuitive frontend interface. Each ticket is assigned a unique identifier and can be tracked throughout its lifecycle, from creation to resolution. Administrators have the ability to view, assign, update, and close tickets, ensuring clear task ownership and accountability at every stage. The system also provides real-time status updates to users, helping reduce follow-up efforts and improving user satisfaction.

The project adopts a full-stack development approach, utilizing React and TypeScript for the frontend to ensure a smooth and responsive user experience, while Java Spring Boot powers the backend services, handling business logic, API routing, and secure data persistence. This separation of concerns ensures scalability and maintainability of the codebase.

Through the implementation of this Ticketing System, organizations can streamline their support processes, minimize communication gaps, and maintain a historical record of all raised and resolved issues for future reference. The system is designed with flexibility in mind, allowing easy adaptation to various business domains beyond technical support, such as human resources, facility management, and client service desks.

The Ticketing System ultimately enhances organizational efficiency by ensuring a structured and transparent workflow for issue tracking and resolution.

PROBLEM STATEMENT

Organizations often face difficulties in managing customer queries and internal issues due to the lack of a centralized and structured system. Manual handling of support requests through emails or direct communication leads to delays, missed issues, lack of accountability, and poor tracking. This creates inefficiencies in resolving problems and maintaining service quality. A reliable ticketing system is essential to streamline issue reporting, automate status tracking, and ensure timely resolution of all raised requests.

OBJECTIVES

1. Ticket Lifecycle Management:

Design and implement a platform to streamline ticket creation, assignment, status updates, and closure, ensuring clear tracking throughout the lifecycle.

2. Role and Privilege-Based Access Control:

Implement secure role-based access to ensure that users, support agents, and administrators can only perform actions permitted for their assigned roles.

3. Multi-Tenant Support:

Architect the system to support multiple tenants, enabling independent ticket data and configurations for different organizations or departments.

4. Real-Time Status Updates and Notifications:

Ensure real-time updates for ticket status changes and trigger email notifications to inform users and agents promptly about important actions.

5. User-Friendly Interface:

Provide a clean, responsive, and intuitive user interface that enhances user experience across desktops, tablets, and mobile devices.

6. Audit Trail and Analytics:

Maintain historical records of all ticket activities and enable performance monitoring to identify trends, bottlenecks, and areas for process improvement.

METHODOLOGY

1. **Issue Reporting:**

Users initiate the process by reporting their issue or requirement via email or directly within the application.

2. **Ticket Generation:**

The system generates a new ticket with details such as issue description, category, type, and priority for structured problem handling.

3. **Ticket Classification:**

The ticket information is validated and classified based on its issue type and category to ensure efficient routing.

4. **Ticket Assignment:**

The system automatically or manually assigns the ticket to the appropriate support staff or team based on the issue type.

5. **Notification Dispatch:**

Once a ticket is assigned, the relevant user receives an immediate notification to acknowledge and start working on the issue.

6. **Response and Status Update:**

The assigned user communicates with the requester, provides solutions, and updates the ticket's status as it moves through the resolution workflow.

7. **Role-Based Access Management:**

Administrators manage user roles and permissions, ensuring only authorized personnel can access or modify specific functionalities.

IMPLEMENTATION

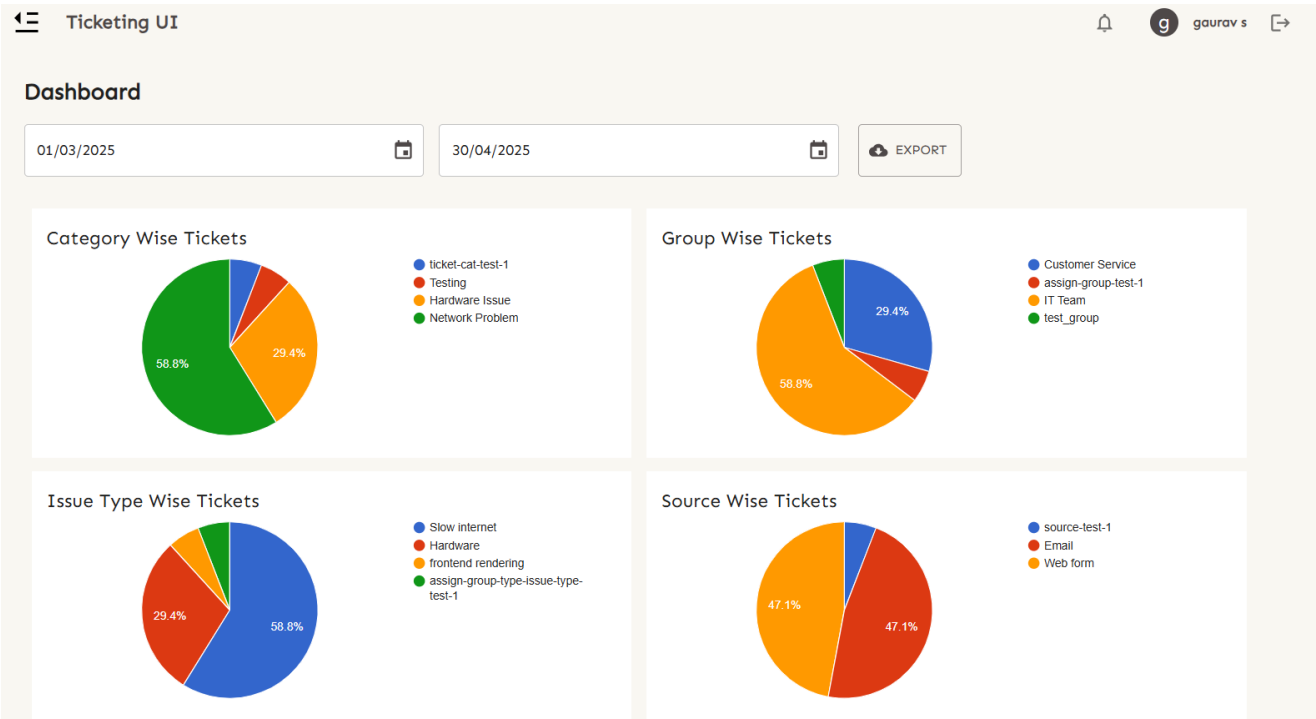


Fig1. Dashboard for It support to see the Ticket Details

Create ticket

Customer Id Profile Details Customer email

Ticket category * Ticket sub-category *

Issue type * Ticket type *

Ticket source * Ticket sub-source *

Key attribute * Child attribute *

Order Id Order Details Assign to group *

Assignee * Promised resolution date *

Is escalation * Ticket Status *

Description

ADD ATTACHMENT

CREATE AND ASSIGN TICKET

Fig2. Ticket creation and Assign Ticket

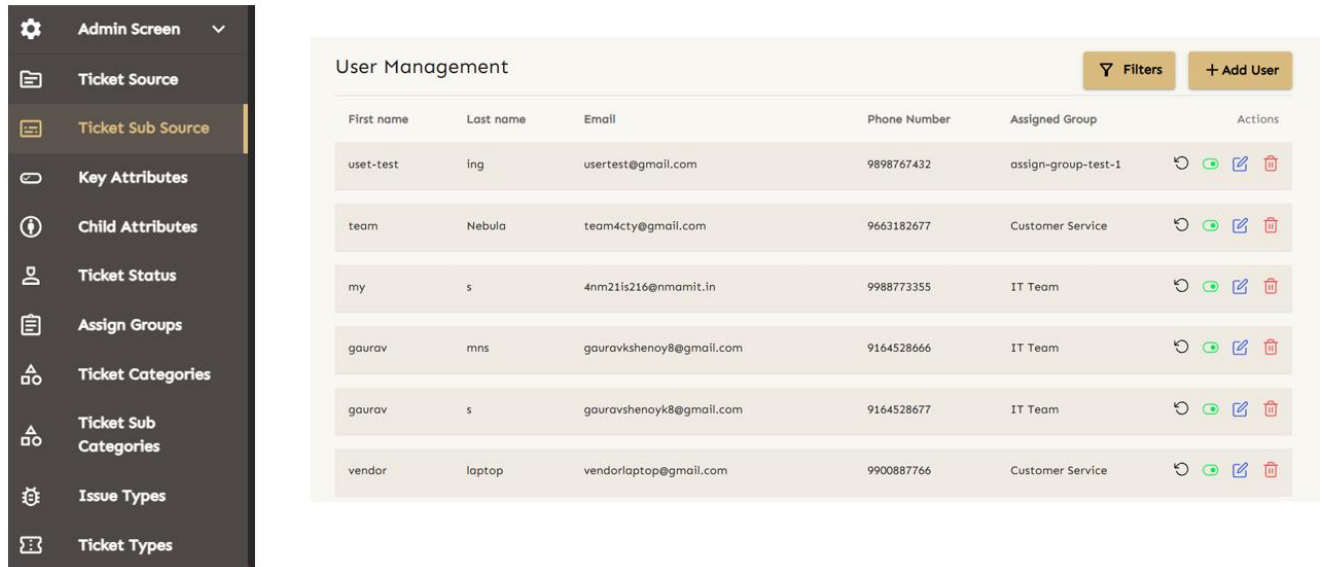


Fig3. Admin Screen Today New Properties to the ticket and User Management

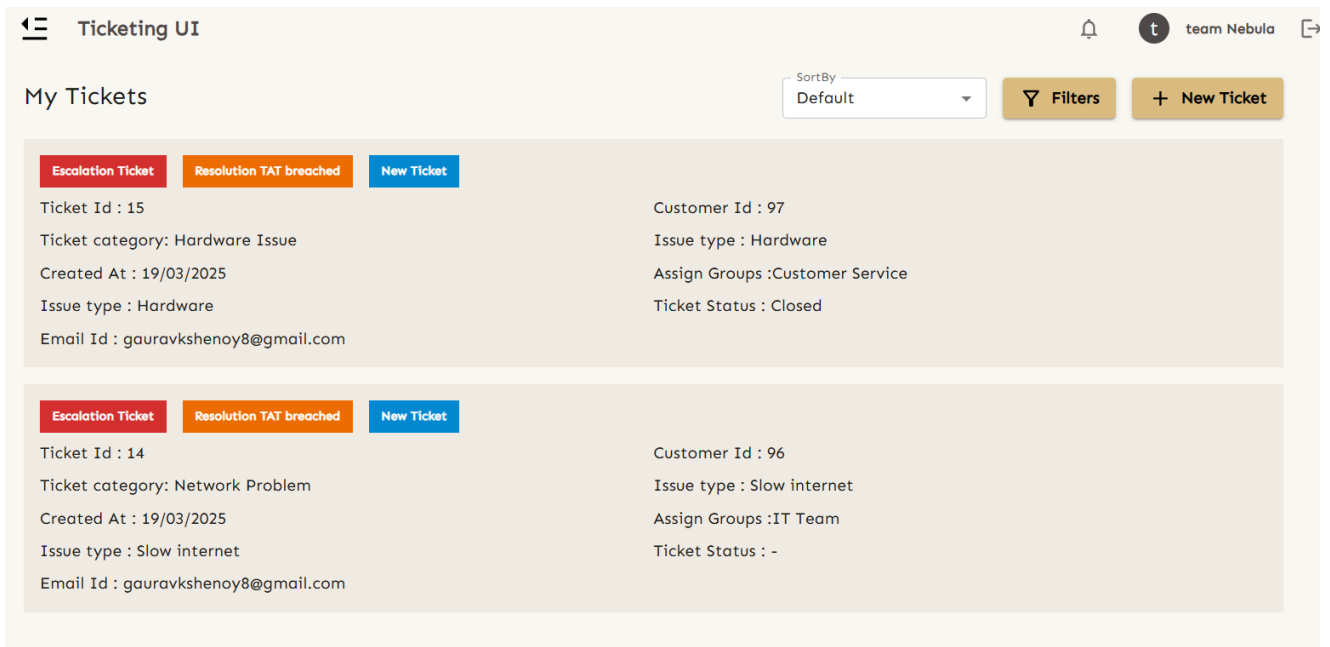


Fig4. Assigned User screen

CONCLUSION

The Ticketing System project was designed and developed to streamline the process of handling technical issues and service requests within an organization. By implementing a structured and automated approach, the system ensures that every user-reported issue is properly documented, assigned, and resolved in an organized manner. The platform not only helps IT support teams manage incoming requests more efficiently but also improves overall communication between users and support staff through timely notifications and status updates.

Role-based access control was introduced to maintain security and ensure that users, support agents, and administrators only interact with the components relevant to their responsibilities. The system's multi-tenant support also enables it to scale and serve multiple teams or organizations independently, without data overlap or conflicts.

Additionally, the integration of email notifications, real-time ticket status updates, and intuitive UI design enhances the user experience, making the platform more reliable and easier to use. The project successfully meets its objective of improving ticket visibility, tracking, and resolution efficiency. Overall, this system creates a productive environment for support operations and lays a strong foundation for future enhancements like reporting dashboards, performance analytics, and AI-driven issue classification.

REFERENCES

1. [Documentation - Keycloak](#)
2. <https://docs.spring.io/spring-boot/redirect.html>
3. Stack overflow

