Title of the Project

A PROJECT REPORT

Submitted by

DHARANI SHREE P (8115U23EC018)

in partial fulfillment of requirements for the award of the course

EGB1201 - JAVA PROGRAMMING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

K. RAMAKRISHNAN COLLEGE OF ENGINEERING
(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

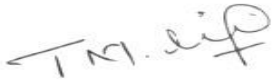
DECEMBER - 2024

K. RAMAKRISHNAN COLLEGE OF ENGINEERING (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **" HELPDESK MANAGEMENT SYSTEM"** is the bonafide work of **DHARANI SHREE .P (8115U23EC018)** who carried out the project work during the academic year 2024 - 2025 under my supervision.



SIGNATURE

Dr. T. M. NITHYA, M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

ASSOCIATE PROFESSOR

Department of CSE

K.Ramakrishnan College of Engineering (Autonomous)

Samayapuram–621112.

SIGNATURE

Mr.V.KUMARARAJA, M.E.,(Ph.D.,),

SUPERVISOR

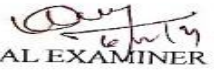
ASSISTANT PROFESSOR

Department of CSE

K.Ramakrishnan College of Engineering (Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on 06/12/24



INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I declare that the project report on **"HELPDESK MANAGEMENT SYSTEM"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **EGB1201 - JAVA PROGRAMMING.**

Signature

DHARANI SHREE P

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Engineering (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. D. SRINIVASAN, B.E, M.E., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. M. NITHYA, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **MR.V.KUMARARAJA, M.E., (Ph.D.,),** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

INSTITUTE VISION AND MISSION

VISSION OF THE INSTIITUTE:

To achieve a prominent position among the top technical institutions.

MISSION OF THE INSTIITUTE:

M1: To best standard technical education par excellence through state of the art infrastructure, competent faculty and high ethical standards.
M2: To nurture research and entrepreneurial skills among students in cutting edge technologies.

M3: To provide education for developing high-quality professionals to transform the society.

DEPARTMENT VISION AND MISSION

DEPARTMENT OF CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

Vision of the Department

To become a renowned hub for Artificial Intelligence and Machine Learning technologies to produce highly talented globally recognizable technocrats to meet industrial needs and societal expectations. **Mission of the Department**

M1: To impart advanced education in Artificial Intelligence and Machine Learning, built upon a foundation in Computer Science and Engineering. **M2:** To foster Experiential learning equips students with engineering skills to tackle real-world problems**.**
M3: To promote collaborative innovation in Artificial Intelligence, machine learning, and related research and development with industries.
M4: To provide an enjoyable environment for pursuing excellence while upholding strong personal and professional values and ethics**.**

Programme Educational Objectives (PEOs):

Graduates will be able to:

PEO1: Excel in technical abilities to build intelligent systems in the fields of Artificial Intelligence and Machine Learning in order to find new opportunities**. PEO2:** Embrace

new technology to solve real-world problems, whether alone or as a team, while prioritizing ethics and societal benefits.

PEO3: Accept lifelong learning to expand future opportunities in research and product development.

Programme Specific Outcomes (PSOs):

PSO1: Ability to create and use Artificial Intelligence and Machine Learning algorithms, including supervised and unsupervised learning, reinforcement learning, and deep learning models.

PSO2: Ability to collect, pre-process, and analyze large datasets, including data cleaning, feature engineering, and data visualization.

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and

safety, and the cultural, societal, and environmental considerations.

1. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

2. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

3. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

4. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Project management and finance: Demonstrate knowledge and

understanding of the engineering and management principles and apply

these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

A HelpDesk Management System is a comprehensive solution for managing customer support processes within a company. It enables users to log and track support tickets for issues or queries, providing a centralized platform to monitor ticket statuses such as open, pending, or resolved. The system ensures efficient allocation of tickets to relevant support agents based on priority or expertise, streamlining communication between customers and support teams. Key features include automated ticket categorization, escalation mechanisms, response time tracking, and a knowledge base for self-service. Additionally, it offers real-time notifications, analytics dashboards, and reporting tools to assess team performance and identify recurring issues. By improving workflow efficiency, reducing response times, and enhancing transparency, the HelpDesk Management System fosters better customer relationships, ensures accountability, and supports continuous operational improvement.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5 : BUILD JAVA APPLICATIONS FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	POs MAPPED	PSOs MAPPED

A HelpDesk Management System is a comprehensive solution for managing customer support processes within a company. It enables users to log and track support tickets for issues or queries, providing a centralized platform to monitor ticket statuses such as open, pending, or resolved. The system ensures efficient allocation of tickets to relevant support agents based on priority or expertise, streamlining communication between customers and support teams.	**PO1 -3** **PO2 -3** **PO3 -3** **PO4 -3** **PO5 -3** **PO6 -3** **PO7 -3** **PO8 -3** **PO9 -3** **PO10 -3** **PO11-3** **PO12 -3**	**PSO1 -3** **PSO2 -3**

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1 Objective

The objective of the Help Desk Management System is to provide a centralized platform for managing customer support requests efficiently. It aims to streamline the process of logging, prioritizing, and resolving tickets, ensuring faster response times and improved issue tracking. By offering real-time updates, automated workflows, and performance analytics, the system enhances communication between customers and support teams while optimizing team productivity. Ultimately, the goal is to deliver effective solutions promptly, leading to higher customer satisfaction and operational excellence.

1.2 Overview

The HelpDesk Management System is a comprehensive solution designed to manage customer support requests efficiently within a company. It serves as a centralized platform where customers can log issues, and support teams can track, prioritize, and resolve tickets. The system includes features such as automated ticket assignment, realtime status updates, and notifications, ensuring streamlined communication between customers and support staff. With built-in analytics and reporting tools, it helps monitor team performance, identify bottlenecks, and improve overall response times. This system not only enhances customer satisfaction by ensuring prompt and effective issue resolution but also optimizes the workflow of support teams, making it an essential tool for modern businesses.

1.3 Java Programming Concepts

The HelpDesk Management System incorporates several foundational and advanced Java programming concepts. Here's a breakdown of the concepts used in this program:

1. Object-Oriented Programming (OOP) Classes and Objects:
    - Classes like Student, Room, and HostelBookingSystem define the blueprint for entities, while objects are their runtime instances.
- Encapsulation:
    - Attributes in classes are private (e.g., name, roomNumber), and access is controlled via getter and setter methods, ensuring data integrity.
- Abstraction:
    - Complex logic (e.g., booking, cancellation) is abstracted into methods like bookRoom() and cancelBooking().
- Polymorphism (Method Overriding):
    - The toString() method in Student and Room classes is overridden to provide customized string representations.

2. Core Java Collections ArrayList:
    - Used to store and manage dynamic lists of students (students) and rooms (rooms), allowing flexible storage and iteration.

3. Control Structures
   - Conditional Statements (if-else):
     - Used to check conditions like room availability (if (!room.isBooked())).
   - Loops: ○ for loop: Initializes room objects and iterates over them for display.
     - while loop: Implements the main menu logic to keep the program running until the user exits.

4. Exception Handling
   - Scanner Input Handling:
     - Used for handling user input (e.g., nextLine(), nextInt()) and preventing runtime errors by handling invalid input gracefully.

5. String Handling
   - Manipulation of strings for storing and displaying student names, IDs, and email addresses.
   - Methods like nextLine() capture and process string input.

6. Modularity
   - Methods:
     - Functions like registerStudent(), viewRooms(), and bookRoom() modularize the code, improving readability and reusability.

7. Basic Input/Output
   - Scanner:
     - Used for taking user input from the console.
   - System.out.println:
     - Used for displaying messages, menus, and outputs.

8. Data Handling
   - Real-time updates to room booking statuses (isBooked attribute).
   - Management of dynamic student and room records using ArrayLists.

9. Menu-Driven Programming
   - The system employs a menu-driven approach, enabling the user to navigate and execute specific functionalities interactively.

10. Program Flow Control Switch Statements:
    - Used in the main menu to handle different user choices efficiently.

11. Overriding and Custom Logic
    - The toString() method is overridden to provide meaningful string representations of objects like Student and Room.

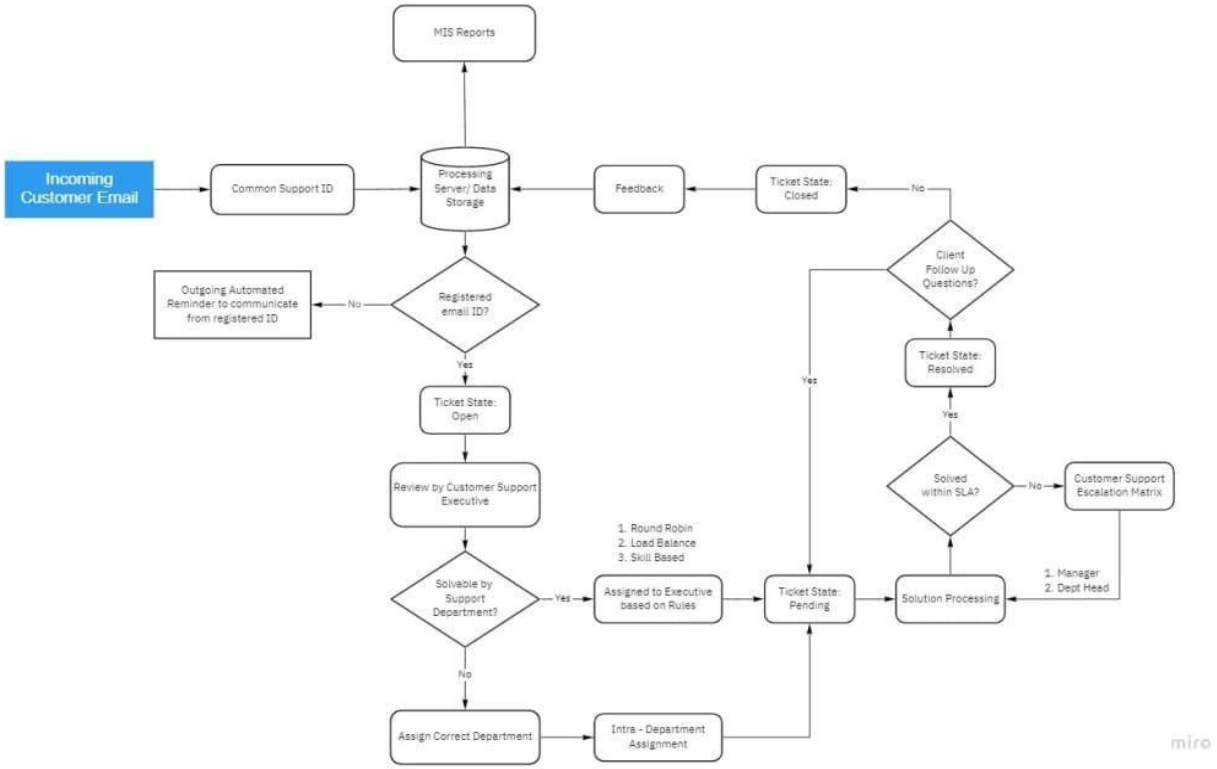
CHAPTER 2 PROJECT METHODOLOGY

2.1 Proposed Work

The proposed work for the Help Desk Management System focuses on developing a centralized platform to streamline customer support operations. The system will enable customers to log and track their support requests while automating ticket categorization and prioritization based on predefined criteria. Role-based access will ensure secure handling of data, and real-time notifications will enhance communication between customers and support teams. Additionally, the system will include analytics and reporting tools to monitor team performance and resolution efficiency. Integration with external tools, scalability, and ease of maintenance are

key considerations to ensure the system meets the evolving needs of the company while improving customer satisfaction.

2.2 Block Diagram



CHAPTER 3 MODULE DESCRIPTION

3.1 User Management Module

Description: Handles the registration, authentication, and management of users, including customers, support agents, and administrators.

Key Features:

User roles and permissions.

Secure login and account management.

3.2 Ticket Management Module

Description: Enables customers to log issues and track their status throughout the resolution process.

Key Features:

Ticket creation with unique IDs.

Categorization and prioritization of tickets.

Real-time updates on ticket status.

3.3 Workflow Automation Module

Description: Automates the assignment and prioritization of tickets to ensure efficient handling.

Key Features:

Auto-assignment based on predefined rules.

SLA (Service Level Agreement) tracking and escalation for overdue tickets.

3.4 Notification Module

Description: Manages communication between customers and support teams through alerts and updates.

Key Features:

Email and SMS notifications.

Real-time alerts for status changes or escalations.

3.5 Analytics and Reporting Module

Description: Provides tools for analyzing support performance and generating detailed reports.

Key Features:

Metrics for ticket resolution times and team efficiency.

Customer satisfaction analytics.

Exportable reports for decision-making.

3.6 Knowledge Base Module

Description: Offers a self-service portal where users can find solutions to common issues.

Key Features:

FAQs and troubleshooting guides.

Search functionality for quick access to information.

3.7 Integration Module

Description: Facilitates seamless integration with third-party tools and systems.

Key Features:

Integration with CRM, email, and messaging platforms.

API support for custom integrations.

3.8 Administration Module

Description: Allows administrators to manage system settings, user roles, and overall operations.

Key Features:

Dashboard for monitoring system health and activities.

Configuration of workflows and access controls.

These modules work together to ensure efficient and effective customer support management, improving both the user and team experience.

CHAPTER 4 CONCLUSION AND FUTURESCOPE

Create Ticket:

```
Helpdesk Management
1. Create Ticket
2. View Tickets
3. Resolve Ticket
4. Exit
Choose an option: 1
Enter issue description: ABC
Ticket created successfully: ID: 1, Issue:
    ABC, Status: Open
```

Assign Ticket:

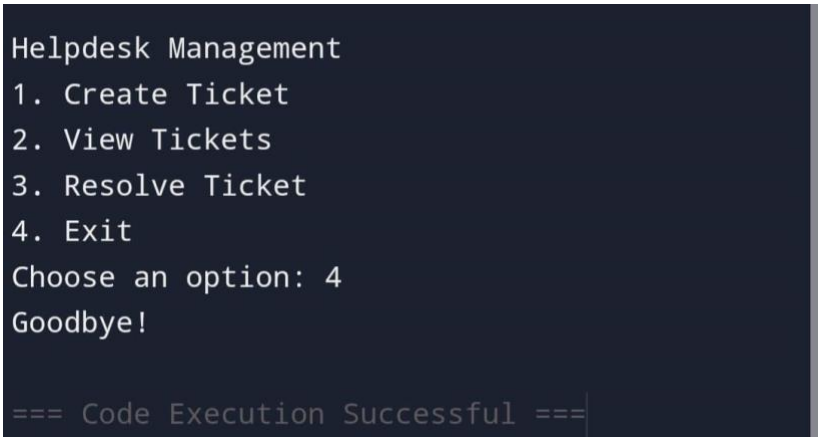
```
Helpdesk Management
1. Create Ticket
2. View Tickets
3. Resolve Ticket
4. Exit
Choose an option: 2
List of Tickets:
ID: 1, Issue: ABC, Status: Open
```

K.RAMAKRISHNAN
COLLEGE OF ENGINEERING
An Autonomous Institution
Permanently Affiliated to Anna University Chennai, Approved by AICTE New Delhi,
ISO 9001:2015, 14001:2015 certified institution, Accredited by NBA and with A grade by NAAC
Samayapuram, Tiruchirappalli – 621 112, Tamilnadu, India.

Resolve Ticket:

```
Helpdesk Management
1. Create Ticket
2. View Tickets
3. Resolve Ticket
4. Exit
Choose an option: 3
Enter Ticket ID to resolve: 1
Ticket resolved: ID: 1, Issue: ABC, Status:
    Resolved
```

Exit:



CHAPTER 5 CONCLUSION

In conclusion, the HelpDesk Management System is an essential tool for improving customer support efficiency and enhancing user satisfaction. By centralizing the ticketing process, automating workflows, and providing realtime updates, the system streamlines communication between customers and support teams. Its robust features, such as analytics, reporting, and integration capabilities, enable businesses to monitor performance and make data-driven improvements. With a focus on scalability, security, and ease of use, the system ensures that companies can handle growing support demands while maintaining high-quality service. Implementing this solution will lead to optimized operations, improved response times, and a better overall customer experience. It is a vital tool for organizations to ensure efficient customer support and issue resolution. By centralizing communication, tracking requests, and streamlining workflows, it significantly enhances the ability to provide timely and effective solutions to users. The system improves both customer satisfaction and internal operations, allowing for better resource management and decision-making.

APPENDIX (Coding)

```
import java.util.ArrayList; import
java.util.Scanner;


public class Main { static class
Ticket { privatestatic int id
Counter = 1; private int id;
private String issue; private
String status;


public Ticket(String issue) {
this.id = idCounter++;
this.issue = issue; this.status
= "Open";
        }




 public int getId() {
            return id;
```

```
                                        }
    public String getStatus() {
return status;
    }
    public void resolve() {
this.status = "Resolved";
    }
    @Override public String toString() { return "ID:
" + id + ", Issue: " + issue + ", Status: " + status;
    }
  }


 public static void main(String[] args) {


ArrayList<Ticket> tickets = new ArrayList<>();
    Scanner scanner = new Scanner(System.in);


    while (true) {
      System.out.println("\nHelpdesk Management");
      System.out.println("1. Create Ticket");
```

```
                                    System.out.println("2. View Tickets");
        System.out.println("3. Resolve Ticket");
        System.out.println("4. Exit");
        System.out.print("Choose an option: ");

        int choice = scanner.nextInt();
scanner.nextLine(); // Consume newline

        switch (choice) {
case 1 -> {
            System.out.print("Enter issue description: ");
            String issue = scanner.nextLine();
Ticket ticket = new Ticket(issue);
tickets.add(ticket);
            System.out.println("Ticket created successfully: " + ticket);
          }
case 2 -> { if
(tickets.isEmpty()) {
              System.out.println("No tickets available.");
            } else {
```

```
System.out.println("List of Tickets:");                    for
(Ticket ticket : tickets) {
                    System.out.println(ticket);
                }
            }
}           case 3 ->
{
            System.out.print("Enter Ticket ID to resolve: ");
int ticketId = scanner.nextInt();                scanner.nextLine();
// Consume newline                boolean found = false;
            for (Ticket ticket : tickets) {                    if
(ticket.getId() == ticketId) {                    if
("Resolved".equals(ticket.getStatus())) {
                    System.out.println("Ticket is already resolved.");
                }           else           {
ticket.resolve();
                    System.out.println("Ticket resolved: " + ticket);
                }
found = true;
break;
```

```
                                                    }
  }   if (!found) {                  System.out.println("Ticket not found.");
                    }
}                  case 4 ->
{
   System.out.println("Goodbye!");
                scanner.close();
return;
               }
               default -> System.out.println("Invalid choice. Please try again”);


       }
        }
    }
}
```

REFERENCES:

1. Books and Articles:

Sommerville, Ian. Software Engineering. Pearson, 10th Edition, 2015.

Pressman, Roger S. Software Engineering: A Practitioner's Approach. McGrawHill, 8th Edition, 2014.

2. Online Documentation:

Oracle. "Java Platform, Standard Edition (Java SE) Documentation." https://docs.oracle.com/javase/

Apache. "Apache Tomcat Documentation." https://tomcat.apache.org/documentation.html

3. Websites and Tutorials:

W3Schools. "Java Programming Basics." https://www.w3schools.com/java/

GeeksforGeeks. "Help Desk Ticketing System Overview." https://www.geeksforgeeks.org/

4. Research Papers:

Jansen, A., and Bosch, J. "Software Architecture as a Set of Architectural Design Decisions." In Proceedings of the 5th Working IEEE/IFIP Conference on Software Architecture (WICSA), 2005.

Chakrabarty, Ananya, et al. "Customer Satisfaction and Helpdesk Performance Metrics." International Journal of Advanced Research in Computer Science, Vol. 9, Issue 3, 2018.