

Streamlining Ticket Assignment for Efficient Support Operations

1. INTRODUCTION

1.1 Project Overview

Customer support systems often struggle with efficiently assigning support tickets to the right agents, leading to delays and dissatisfaction. This project proposes an automated ticket assignment solution that uses rule-based logic and AI to streamline support operations.

1.2 Purpose

To develop a system that automates and optimizes ticket assignment, reducing response time and improving service quality by aligning the right issues with the right support personnel.

2. IDEATION PHASE

2.1 Problem Statement

Manual or poorly designed ticket assignment systems lead to inefficiencies such as delays, overburdened staff, and missed service levels, ultimately reducing customer satisfaction.

2.2 Empathy Map Canvas

Says	Thinks	Feels	Does
"Why is my ticket taking so long?"	"I hope this issue gets resolved quickly."	Frustrated, ignored	Keeps checking status, contacts again

2.3 Brainstorming

Ideas explored:

- Rule-based assignment (priority, department)
- Round-robin and load-based allocation
- AI/ML for topic detection
- Integration with platforms like Freshdesk or Zendesk

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

Stage	Customer Action	System Response
Raise Issue	Submits ticket	Acknowledges receipt
Wait	Awaits resolution	Assigns ticket, updates status
Resolve	Confirms resolution	Collects feedback

3.2 Solution Requirement

- Auto-routing based on keywords or tags
- Agent availability tracking
- SLA timer and escalation rules
- Reporting and analytics dashboard

3.3 Data Flow Diagram

Customer → Ticketing Interface → Ticket Classifier → Assignment Engine → Agent Dashboard

3.4 Technology Stack

- Frontend: HTML/CSS, React (optional)
- Backend: Node.js / Python Flask
- Database: MySQL or Firebase
- AI: NLP for ticket classification
- Platform: Integration with support systems via APIs

4. PROJECT DESIGN

4.1 Problem Solution Fit

A well-mapped solution using automation ensures that tickets are handled efficiently, matching customer expectations with timely and expert resolution.

4.2 Proposed Solution

Build a smart ticket routing system that:

- Classifies tickets based on topic and priority
- Matches them with available agents
- Uses real-time agent load balancing
- Tracks SLA and escalates when needed

4.3 Solution Architecture

Architecture Layers:

1. Ticket Ingestion – Customer submits ticket
2. AI-based Classifier – Tags and categorizes issue
3. Assignment Engine – Applies business logic to assign
4. Notification & SLA Tracker – Notifies agent, tracks time
5. Dashboard – Visualizes metrics and progress

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Week	Task
1	Requirement gathering & research
2	System architecture design
3	Frontend and backend setup
4	AI model integration
5	Testing and refinement
6	Report and final deployment

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- Load Test: Multiple simultaneous ticket submissions
- Response Time: Ticket assignment under 3 seconds
- Agent Load: No agent receives more than 2x average load

7. RESULTS

7.1 Output Screenshots

(Insert interface screenshots, ticket dashboard, agent view, and load graph here)

8. ADVANTAGES & DISADVANTAGES

Advantages	Disadvantages
Faster ticket resolution Reduced agent burnout SLA compliance	Requires integration with existing systems May need initial training for AI model Maintenance of agent skill database

9. CONCLUSION

Automated ticket assignment improves efficiency, customer satisfaction, and reduces errors. With AI and proper system design, support teams can focus on resolving issues instead of managing workflows.

10. FUTURE SCOPE

- Integrate chatbot for pre-ticket filtering
- Predict agent fatigue using historical data
- Implement feedback-driven routing improvements
- Multilingual ticket classification

11. APPENDIX

GitHub & Project Demo Link: [<https://github.com/Chandu-1122/Streamlining-Ticket-Assignment-for-Efficient-Support-Operations>]