

Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	31 January 3035
Team ID	NM2025TMID08026
Project Name	1 NOVEMBER 2025
Maximum Marks	4 Marks

Technical Architecture:

This project uses an automated ticket assignment system to streamline support operations. Tickets generated in the system are analyzed based on priority, category, and urgency. Agent profile data, including skills and workload, is used to determine the most suitable support engineer for each ticket. The automation logic runs on the platform backend, assigning tickets in real-time without manual effort. A dashboard displays ticket status and assignment logs. The solution ensures faster resolution time, fair ticket distribution, and improved SLA compliance.

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Support team and admin use web dashboard to view and manage tickets.	Web UI (HTML, CSS, JS / Framework UI
2.	Ticket Input Layer	Accepts new support tickets with priority & category	Web Form / Ticket Portal
3.	Automation Logic-1	Assigns tickets based on priority	Backend Rules / Automation Engine
4.	Automation Logic-2	Matches agent skills & checks availability.	Rule Engine / Backend Script
5.	Automation Logic-3	Stores user data, ticket data, and agent skill metadata	Load distribution logic / Workflow rules
6.	Database	Stores user data, ticket data, and agent skill metadata	MySQL / Firebase / Platform DB
7.	Cloud/Server Execution	Hosted application logic and automation engine	Cloud Hosting (AWS / Firebase / Web Server)
8.	External API-1 (Optional)	Can integrate with HR/Agent management systems to fetch skills	REST API in ServiceNow
9.	External API-2	Optional ticketing integration (Zendesk/Jira/ServiceNow API)	REST/Webhooks
10.	Machine Learning Model	(Future upgrade) AI-based smart routing & agent prediction	ML Model (future scope)
11.	Storage & Logs	Activity logs and ticket assignment history stored	Cloud DB / Local storage logs

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Web UI & backend automation can be built using open-source stack	Node.js / Python / React (Example)
2.	Security Implementations	User authentication & role-based access to ticket system	JWT / RBAC / Encrypted DB
3.	Scalable Architecture	Can scale with more users & agents by cloud deployment	Cloud Hosting (AWS/Firebase)
4.	Availability	System accessible anytime via cloud	Load Balancing / Cloud Server
5.	Performance	Fast ticket routing, optimized queries, async processings	GlideRecord, Background Scripts