

Software Project I, Winter 2023  
Staff Help Desk Software

## Project Overview

Students are tasked with developing a robust Help Desk software application using the MERN stack (MongoDB, Express.js, React, Node.js). The software aims to streamline the support and ticketing process, enhance communication between support agents and users, and provide a comprehensive knowledge base for quick problem resolution.

Thanks to ECS for their collaboration and project proposal.

Copyright© 2023 ECS

## Features

### 1. User Management

- **User Authentication and Role-Based Access Control:**  
Users should be able to securely log in with their credentials.  
Administrators can create new user accounts and assign roles to manage access.
- **User Profile Management:**  
Users can update their profile information to ensure accuracy.

### 2. Ticketing System

- **Create, Update, and Close Support Tickets:**  
Users can create new support tickets, providing detailed information about their issues.  
Support agents can update and close tickets, providing resolution details.  
We will have 3 support agents for our system.
- **Categorization and Prioritization of Tickets:**  
Tickets should be categorized based on issue types: Software, Hardware, and Network.  
Where Hardware Issues: Desktops, Laptops, Printers, Servers, Networking equipment  
Software issues: Operating system, Application software, Custom software, Integration issues  
and Network Issues: Email issues, Internet connection problems, Website errors.  
So, each sub category will have it's own priority.  
Agent 1 will be responsible 90% of the time to software issues, 5% to hardware and 5% to network.  
Agent 2 will be responsible 90% of the time to hardware issues, 5% to software and 5% to network.  
Agent 3 will be responsible 90% of the time to network issues, 5% to hardware and 5% to software.
- **Custom Workflow for Issue Reporting:**  
When a user reports an issue, a custom workflow will be provided by the assigned agent.

### 3. Knowledge Base

- **Organized Repository of FAQs and Solutions:**  
Users can access a Knowledge Base with solutions to common issues.  
Knowledge Base should be organized for easy navigation.
- **Search Functionality:**  
Users can search the Knowledge Base for quick problem resolution.

### 4. Communication Tools

- **Integrated Email and Notification System:**  
Users receive email and notifications for ticket updates.  
Support agents can communicate with users through an integrated messaging system (email).
- **Real-Time Chat Functionality:**  
Users and support agents can engage in real-time chat for immediate assistance.  
Real-time chats will be saved in the database for future reference.

### 5. Reporting and Analytics

- **Generate Reports and Analytics:**  
Manager can generate reports on ticket status, resolution time, and agent performance based on user ratings.  
Analytics to identify common issues and trends like chart,etc  
Reports and analytics will be viewed by manager.

### 6. Automation and Workflows

- **Automate Repetitive Tasks:**  
Implement automated workflows for specific issue types.  
When a user reports an issue, a custom workflow will be provided by the assigned agent.  
Automatically route and assign tickets based on priority and availability of agent.

### 7. Customization and Branding

- **Customize Look and Feel:**  
Administrators can customize the Help Desk's appearance to align with the organization's branding for example changing website colors.

### 8. Security and Data Protection

- **Implement Security Measures:**  
Security measures will include encryption and decryption.  
Ensure the protection of sensitive customer data.  
Data backup and recovery procedures remain a priority for system robustness.

## Additional Features

Teams consist of 6-8 members will choose 1 out of 3 additional features.  
Teams consist of 9 members will choose 2 out of 3 additional features.  
Teams consist of 10 members will implement the 3 additional features.

## Data Science: Ticket Routing System

- **Integrate Pretrained Models:**

Train a machine learning model to automatically route (classify) tickets to the most suitable support agents based on issue type and priority using sample generated dataset.

## Security: Multi-Factor Authentication (MFA)

- **Enhanced Security:**

Add Multi-Factor Authentication (MFA) as an authentication option for all users to enhance overall system security.

## Software Engineering: Error and Exception Logging

- **Develop Robust Logging System:**

Create a robust error and exception logging system to capture and track application issues in real-time. This system will facilitate rapid troubleshooting and debugging, ensuring the application's stability.

## User Stories

### 1. User Management

- As a user, I want to securely log in with my credentials.
- As an administrator, I want to create new user accounts and assign roles to manage access.
- As a user, I want to update my profile information to ensure accuracy.

### 2. Ticketing System

- As a user, I want to create new support tickets, providing detailed information about my issues.
- As a support agent, I want to update and close tickets, providing resolution details.
- As a user, I want to categorize my ticket based on issue types: Software, Hardware, and Network.
- As a user, I want to choose sub-category and based on sub category, priorities will be assigned for efficient management.
- As a user, I want a custom workflow provided by the assigned agent when reporting an issue.

### 3. Knowledge Base

- As a user, I want to access a Knowledge Base with solutions to common issues.
- As a user, I want the Knowledge Base to be organized for easy navigation.
- As a user, I want to search the Knowledge Base for quick problem resolution.

### 4. Communication Tools

- As a user, I want to receive email notifications for ticket updates.
- As a support agent, I want to communicate with users through an integrated messaging system.
- As a user, I want to engage in real-time chat with support agents for immediate assistance.
- As a support agent, I want real-time chats to be saved in the database for future reference.

## 5. Reporting and Analytics

- As a manager, I want to generate reports on ticket status, resolution time, and agent performance based on user ratings.
- As a manager, I want to view analytics and reports.

## 6. Automation and Workflows

- As a support agent, I want to implement automated workflows for specific issue types.
- As a user, I want automated ticket routing and assignment based on priority and availability of agent.

## 7. Customization and Branding

- As an administrator, I want to customize the Help Desk's appearance to align with the organization's branding.

## 8. Security and Data Protection

- As a user, I want the system to implement security measures, including encryption and decryption.
- As a user, I want the protection of sensitive customer data.
- As a user, I want Data backup and recovery procedures.

## Additional features user stories

### Data Science: Ticket Routing System

- As a system administrator, I want to integrate pretrained models for ticket routing so that tickets are automatically assigned to the most appropriate support agent based on historical data and patterns.

### Security: Multi-Factor Authentication (MFA)

- As a user, I want the option to enable Multi-Factor Authentication (MFA) to enhance the security of my account.

### Software Engineering: Error and Exception Logging

- As a developer, I want the application to capture and log errors and exceptions in real-time, allowing for rapid troubleshooting and debugging.
- As a system administrator, I want access to a detailed log of errors and exceptions to monitor the application's health and identify potential issues proactively.
- As a user, I want the system to implement security measures, including encryption and decryption.
- As a user, I want the protection of sensitive customer data.

## Bonus features

- Fancy user Experience and user Interface

## Technology Stack

- **Frontend:** React
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Validation:** Joi for input validation, Mongoose for database schema validation
- **Authentication:** JSON Web Tokens (JWT)
- **Real-Time Communication:** Socket.io as suggestion
- **Infrastructure:** Deploy on cloud services like Heroku

## Deliverables

- Fully functional Help Desk software application deployed to a cloud platform.
- Source code on GitHub with history and branch for each member.

## Timeline

- **Milestone 1 (Week 1) (*18th Nov - 24th Nov*):**  
Set up project structure and design database schema (Models).  
Deadline: November 24  
Milestone 1 submission link: <https://forms.gle/abpTAzPdugbtdFgz6>
- **Milestone 2 (Weeks 2-3) (*21st Nov- 5th Dec*):**  
Develop the back-end of the application.  
Deadline: December 5  
Milestone 2 submission link: <https://forms.gle/bpA6nEy1pRqsjaE96>
- **Milestone 3 (Weeks 4-6) (*5th Dec- 26th Dec*):**  
Develop the front-end and deploy the application.  
Deadline: December 26  
Milestone 3 submission link: <https://forms.gle/f4L8fggUBJVL7K5H6>

## Project Presentation

At the end of the project, each team will present their Help Desk software application. The presentation should include a demonstration of key features, an overview of the technology stack, and insights into the development process.