Project Requirement Document: Ticketstm

Project Name	**Ticketstm [Ticket Generation System]**
Authors	1. Shubham Waykar 2. Ashish Gaikwad
Mentor	Swapnil Akolkar

Overview:-

Ticketstm is an system designed to streamline the process of generating and managing user queries or tickets. This comprehensive tool efficiently handles user inquiries and provides real-time support by generating and organizing tickets based on various criteria. Ticketstm offers a range of features to enhance user experience and query resolution.

How it will help Users with Queries:

Ticketstm is designed to revolutionize the way users interact with support systems, providing a seamless experience that empowers users and enhances query resolution efficiency.

1. **Instant Query Submission:** Users can submit their queries through Ticketstm's user-friendly interface, eliminating the need to navigate through complex support systems. This quick and easy process ensures that users can seek assistance without any hassle.
2. **Clear Communication:** Ticketstm ensures that users' queries are accurately recorded and categorized, preventing any misunderstandings. Users can provide detailed information about their concerns, allowing support teams to comprehend the issue clearly.
3. **Real-time Status Updates:** Once a ticket is generated, users can track the status of their query in real-time. This feature provides transparency and keeps users informed about the progress of their inquiry.
4. **Timely Assistance:** Ticketstm's prioritization system guarantees that urgent queries receive swift attention. This means users with critical concerns don't have to

wait unnecessarily and can get the help they need promptly.

5. **Efficient Query Resolution:** Categorized ticket handling enables support teams to assign queries to the most suitable agents or departments. This ensures that users receive accurate and relevant responses, leading to faster query resolution.

Features:-

1. **Instant Query Submission:** Users can submit their queries through Ticketstm's user-friendly interface, eliminating the need to navigate through complex support systems. This quick and easy process ensures that users can seek assistance without any hassle.

2. **Clear Communication:** Ticketstm ensures that users' queries are accurately recorded and categorized, preventing any misunderstandings. Users can provide detailed information about their concerns, allowing support teams to comprehend the issue clearly.

3. **Real-time Status Updates:** Once a ticket is generated, users can track the status of their query in real-time. This feature provides transparency and keeps users informed about the progress of their inquiry.

4. **Timely Assistance:** Ticketstm's prioritization system guarantees that urgent queries receive swift attention. This means users with critical concerns don't have to wait unnecessarily and can get the help they need promptly.

Future scope:-

- Discussion Forum.
- Ranking of Agnet Based on their Performance.
- Daily and monthly ticket creating Graphic charts.
- Real time Chat with query resolver.
- Dark theme support.

Success Metrics:-

1. **Uniformity**: The UI will be based on a uniform theme. We will be using Material UI to have a consistent and responsive UI.

2. **Compatibility**: The website will be cross-browser compatible
    - The web app should support all browsers like Chrome, Firefox, and Safari.
    - The web app should be responsive across different screen sizes and will provide a smooth user experience to mobile users.
3. **Scalability**: The web app should provide a smooth user experience even during periods of high user traffic.

Non Functional Requirements (NFRs):-

1. Secure and reliable data handling
2. Responsive UI.
3. Cross-browser Support.
4. Consistent user experience across the app.
5. Smooth User Experience during high-traffic periods (Scalability)
6. SEO Friendly (future scope)
6. Maintainable codebase
    - This will also Optimise development efforts while adding new features.

User Stories:-

A typical end-user could be any one of the following :

1. **As a User,** I want to create a new ticket so that I can seek assistance for my query or concern easily.
2. **As a User,** I want to view the status of my submitted tickets so that I can track the progress of issue resolution.
3. **As an Agent,** I want to view and access assigned tickets from the queue so that I can address user queries efficiently.
4. **As an Agent,** I want to communicate with users regarding their tickets to gather more information or provide updates.
5. **As an Administrator,** I want to manage user accounts and roles, assigning agents to specific ticket categories for specialized support.

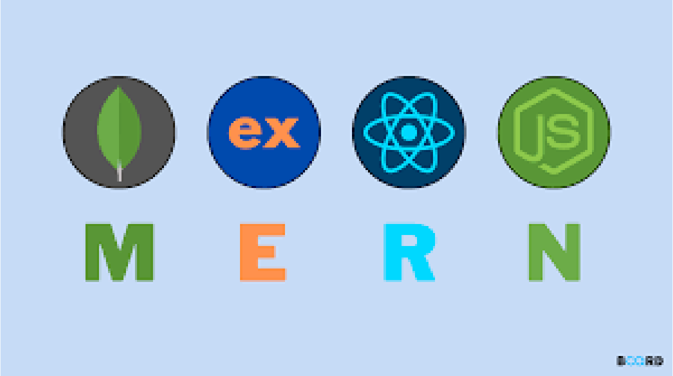
6. **As an Administrator,** I want to send important announcements or updates to users and agents using the system.

End User Requirements:-

#	User Requirements	Description
1	Sign UP and Login	**Ability to Signup, Login and reset password** Features: 1. User should be able to sign up (Username: EmailId; password: minimum length of 6 characters, should be a combination of letters, numbers and special characters, the password should be hashed) 2. User should be able to log in using a username and password 3. User should be able to reset the password (a link should be sent to the user to reset the password) (future scope) 4. Google login integration (future scope)
2	Guest Login	Guest users have the option to proceed without registration and raise tickets for their queries. However, these users won't have access to view their ticket history.
3	Dashboard	The user will able to create ticket , update comment on ticket also apply various filter on tickets.
4	Ticket Page	**Users will be able to see his/her performance in the exam:** - **Marks obtained** = total marks obtained - negative marks - **Negative marks** = wrong attempts * negative factor - **Attempt** = number of attempted questions - **Unattempted** questions - **Correct** answer count - **Wrong** answer count - **Accuracy** = 100 * correct/ attempt
5	Rank Page	The user will be able to check his rank among the survey participants
6	Feedback Page	User should share his/her feedback with the RankIQ admin An email will be triggered to the user submitting the feedback and the RankIQ admin
7	Localisation	The site content should be available in English and Hindi.

Tech Stack:-

Component	Technology
Front-end	React, Material-UI
Backend	Node.js, ExpressJs
DB	MongoDB
Authentication	JWT Token
CI/CD - Frontend	Vercel integration with GitHub
CI/CD - Backend	Render integration with GitHub



Milestones, Timeline and Priorities:-

MVP	Details	Priority	Timeline
Tech stack Integration and setting up scaffold	1. React JS Project setup 2. Backend Project Setup (NodeJS + ExpressJS) 3. database integration with node Js 4. GitHub Repository setup 5. Git environment setup	High	Week 1
High priority action items for app development	1. DB schema development 2. Brainstorming about the different scenario and use cases.	High	Week 1
User Authentication and rank checking functionality (BE)	1. APIs- login, sign-up, store user data, create Ticket. 2. JWT Token Integration	High	Week 2

MVP	Details	Priority	Timeline
User Authentication and rank checking functionality (FE)	1. Login, Sign-up screen, 2. User Dashboard. 3. About us page 4. Landing page of website	High	Week 2
CI/CD, Sentry integration, Hosting environment setup	vercel + render + github integration	Medium	Week 3
Testing	Jest + React Testing Library	High	Week 3
Ticket Detail Page	Show ticket Detail with comments based on User Roles.	High	Week 3
Performance Testing	User Traffic simulation	Medium	Week 3
Non-functional Requirements	1. Responsive UI 2. Cross-browser Support 3. Smooth User Experience during high traffic period 4. SEO Friendly (future scope)	High	week 1 - 3