Assignment 1

A company is having manufacturing plants and warehouses in various parts of the country. They manufacture ice-cream and milk products. They want to build software to achieve two goals.

- **Manage the inventory**
- **Quickest delivery to the customers**

1. **Please make a BRD which can be presented to the client along with complete development and resource plan.**

1. Executive Summary

This document outlines the requirements for developing software for XYZ, a manufacturer of ice-cream and milk products with a network of manufacturing plants and warehouses across the country. The proposed system will focus on two primary objectives: effective inventory management and enabling the quickest delivery to customers. By achieving these goals, the company aims to enhance operational efficiency, minimize waste, and improve customer satisfaction.

---

2. Business Goals

1. **Streamlined Inventory Management**: Ensure real-time monitoring of inventory levels across all warehouses and plants to reduce spoilage and optimize stock levels.
2. **Enhanced Delivery Efficiency**: Implement a system to identify the fastest delivery routes and automate order allocation based on proximity and inventory availability.
3. **Improved Customer Satisfaction**: Minimize order fulfillment time and ensure product freshness at delivery.

---

3. Business Objectives

1. Develop a centralized system to track and manage inventory across all locations.
2. Automate order processing, allocation, and dispatch based on inventory and location proximity.
3. Integrate delivery route optimization with real-time traffic and weather data.
4. Reduce inventory wastage by implementing an expiry-date tracking mechanism.
5. Enable reporting and analytics for better decision-making and demand forecasting.

---

4. Business Rules

1. Inventory must be updated in real-time upon receipt, dispatch, or adjustment.
2. Orders should only be fulfilled if the inventory is available and meets the required shelf-life criteria.
3. Delivery routes must prioritize freshness while minimizing transportation costs.
4. Customer priority orders (e.g., bulk orders) must be flagged for immediate action.
5. Warehouse reordering thresholds should trigger automated purchase orders.

---

5. Scope of the System

In-Scope:

- Inventory tracking at manufacturing plants and warehouses.
- Order management and allocation.
- Delivery route optimization.
- Analytics and reporting dashboards.

Out-of-Scope:

- Manufacturing process management.
- Customer relationship management (CRM).

---

6. Assumptions

1. All warehouses and plants have internet connectivity for real-time updates.
2. Delivery vehicles are GPS-enabled to support route optimization.
3. Data from external sources (e.g., traffic, weather) is available via APIs.

---

7. Constraints

1. Budget limitations may restrict the scope of features.
2. Integration with legacy systems may pose technical challenges.
3. Project timelines must align with the company’s peak season schedules.

---

8. Risk Analysis

Technical Risks:

- Integration issues with existing systems.
- System scalability with increasing business demand.

Political Risks:

- Resistance from employees due to new processes and technology.
- Potential vendor lock-in with third-party tools.

Requirement Risks:

- Incomplete or evolving requirements from stakeholders.

Business Risks:

- Downtime during system rollout.
- Customer dissatisfaction due to transition delays.

---

9. Business Process Overview

AS-IS:

- Inventory is managed manually or using isolated systems.
- Order allocation is manually determined, leading to inefficiencies.
- Delivery routes are planned based on experience, not optimized.

TO-BE:

- A centralized system enables automated inventory management.
- Orders are allocated and dispatched through an intelligent algorithm.
- Delivery routes are optimized dynamically for speed and cost.

---

10. Business Requirements

1. **Inventory Management Module**:
    - Real-time inventory tracking across locations.
    - Expiry-date monitoring and alerts for perishable items.
2. **Order Management Module**:
    - Automated order allocation based on location and inventory.
    - Bulk order prioritization.
3. **Delivery Optimization Module**:
    - Integration with GPS and real-time data for route planning.
    - Dynamic rerouting in case of delays.

4. **Reporting Module**:
    - Insights into inventory levels, order trends, and delivery performance.

---

11. Development and Resource Plan

Development Plan:

1. **Phase 1**: Requirement gathering, stakeholder workshops, and system design (4 weeks).
2. **Phase 2**: Development of core modules (Inventory, Order, Delivery) (12 weeks).
3. **Phase 3**: Integration with external systems (APIs, GPS) and legacy systems (6 weeks).
4. **Phase 4**: Testing and quality assurance (4 weeks).
5. **Phase 5**: Deployment, training, and support (4 weeks).

Resource Plan:

- **Project Manager**: 1 FTE
- **Developers**: 3 FTE (Backend, Frontend, Integration specialists)
- **QA Engineers**: 2 FTE
- **Business Analyst**: 1 FTE
- **Support Team**: 2 FTE

---

12. Appendices

1. Glossary of Terms
2. Abbreviations Used

---

13. Related Documents

1. Feasibility Study Report
2. Market Analysis for Inventory and Delivery Optimization Software
3. Stakeholder Requirements Document

Assignment 2

1. Write an introduction letter to a client introducing yourself as a business analyst in charge of working with the client and his team to start the business understanding process.

Subject: Introduction as Your Business Analyst Partner

Dear Sir,

I hope this message finds you well. My name is Shailesh Mishra, and I am delighted to introduce myself as the Business Analyst assigned to collaborate with you and your team on this exciting project.

Understanding the challenges and opportunities within the manufacturing and logistics sectors, particularly in delivering exceptional customer service, is a domain I am passionate about. With your vision of managing inventory and ensuring the quickest delivery of your ice-cream and milk products, my role will be to work closely with you to transform these goals into a robust, tailor-made software solution.

To begin, I aim to thoroughly understand your current processes, challenges, and aspirations. Together, we will explore your operational workflows, identify key requirements, and map out a strategic plan that aligns with your business objectives. Your insights will be invaluable in ensuring the solution we design is practical, scalable, and optimized for your unique needs.

I look forward to discussing your expectations and gathering inputs from your team during our initial meetings. In the meantime, please feel free to share any documents, current processes, or initial thoughts that could help us hit the ground running.

Thank you for the opportunity to collaborate on this project. I am confident that, together, we will develop a solution that adds significant value to your business operations. Please let me know a convenient time for us to connect further.

Looking forward to working with you.

Warm regards,
Shailesh Mishra
Business Analyst
+91 6268644428

2. Prepare a brief BRD and SRS for a project- Ticketing system

Business Requirements Document (BRD)

Project Name: Ticketing System

Prepared By: Shailesh Mishra
Date: 6th December 2024

Executive Summary

The objective of the Ticketing System project is to implement a comprehensive solution for managing customer support tickets effectively. The system will streamline the process of ticket creation, tracking, resolution, and closure. It will be designed to ensure seamless communication between users, support agents, and managers, improving the overall efficiency of customer support operations. Additionally, the system will include advanced features for monitoring service level agreement (SLA) compliance, enhancing reporting capabilities, and integrating with a knowledge base for faster problem resolution.

---

1. Document Revision

Date	Version Number	Document Changes
23/10/2024	1	Initial draft of Requirement Gathering
24/10/2024	2	Made changes in the requirement gathering
25/10/2024	3	Updated RACI
26/10/2024	4	Changed the use case diagram
27/10/2024	5	Updated RTM

2. Approval

Role	Name	Title	Signature	Date
Project Sponsor	Kalpana Sandeep Kumar	Requirements	Kalpana Kale Sandeep Kumar	23/10/2024
Business Owner	Shailesh Mishra	Requirements	Shailesh Mishra	24/10/3034
Project	Parth Desai	BRD	Parth Desai	23/10/2024

Manager				
System Architect	Major Verma	Architecture	Manoj Verma	24/10/2024
Development Lead	Kashi Tayde	Requirement	Kashi Tayde	24/10/2024
User Experience Lead	Pooja Nagul	Design	Pooja Nagul	24/10/2024
Quality Lead	Payal Tiwari	Quality	Payal Tiwari	24/10/2024
Content Lead	Dilip Kumar	Content	Dilip Kumar	24/10/2024

3. RACI

Name	**Position**	*	**R**	**A**	**S**	**C**	**I**
Aditya Jha	**Hirirng Manager**					**Yes**	
Aradhana Sunny	**Recuiter**					**Yes**	
Kalpana Kale,Sandeep Kumar	**Project Sponsor**	**Yes**	**Yes**			**YES**	**Yes**
Piyush Jha	**Business Owner**	**Yes**	**Yes**	**Yes**		**Yes**	**Yes**

4. Introduction:

4.1 Business Goals

- **Streamline Issue Resolution**
  Enable users to easily raise and track tickets while ensuring quick and efficient resolution by support teams.
- **Enhance Transparency**
  Provide users and stakeholders with real-time updates on ticket status and resolution timelines.
- **Improve Accountability**
  Introduce clear ownership of tickets through automated assignments and escalation mechanisms.
- **Boost Operational Efficiency**
  Automate routine processes such as ticket routing and SLA monitoring to reduce manual intervention and errors.
- **Enable Data-Driven Decisions**
  Generate actionable insights through reporting and analytics to identify patterns, improve service quality, and optimize resources.

.

4.2 Business Objectives

1. **Enhanced Customer Satisfaction:** By reducing resolution time and ensuring timely support through SLA compliance.
2. **Optimized Support Team Performance:** By providing tools for assigning tickets to the appropriate agents, managing workloads, and tracking performance.
3. **Better Decision-Making:** By offering detailed reports on ticket trends, SLA performance, and agent efficiency.
4. **Cost Efficiency:** By automating routine support processes, reducing human error, and speeding up the resolution process, leading to a reduction in operational costs.

4.3 Business Rules

🎬 Tickets must be assigned to agents within 15 minutes of creation.
🎬 Tickets cannot be closed until all required information is provided, and the issue is fully resolved.
🎬 Support agents must adhere to the SLAs defined for each ticket type.

🎬 A ticket must be reopened if a customer reports the issue again within 30 days of closure.

4.4 Background

- The company's current ticketing system is inefficient, causing slow response times, missed SLAs, and manual processing. To address these issues, the company will implement a new Ticketing System that automates ticket management, tracks SLAs, integrates a knowledge base, and provides detailed reporting. This system will streamline support processes, improve customer satisfaction, and ensure compliance with SLAs.
- The project aims to enhance efficiency, reduce costs, and provide better data insights for decision-making. Key stakeholders include customer support, IT, and business leadership. The project will be completed in 6 months, with phased rollout and ongoing support.

4.5 Scope of the System

In-Scope:

- User portal for raising tickets.
- Admin/Support team portal for ticket management.
- Automated ticket assignment and prioritization based on predefined rules.
- SLA tracking and notification alerts for pending tickets.
- Reports and dashboards for performance analysis.

Out-of-Scope:

- Integration with third-party tools (to be planned in future phases).
- Hardware procurement and setup.

---

5. Assumptions

- All users will have access to a computer or mobile device with an internet connection.
- Support teams will adhere to SLA policies as defined by the organization.

---

6. Constraints:

- The system must be developed and implemented within a 6-month timeline.
- The system should be scalable to handle up to 10,000 tickets per day without performance degradation.
- It must comply with applicable data privacy regulations (e.g., GDPR).

7. Risk

1. Technical Risk:

- **Risk:** Integration and compatibility issues with existing systems (CRM, email, chat, etc.) could lead to delays or functionality problems, causing disruptions in the ticket management process.
- **Mitigation:** Conduct detailed technical assessments and integration testing to ensure compatibility with current systems. Allocate time for troubleshooting and ensure robust APIs for seamless data exchange.

2. Political Risk:

- **Risk:** Internal organizational changes, such as shifts in key personnel or management priorities, could affect the project's support or direction, causing delays or shifting project goals.

- **Mitigation:** Maintain regular communication with key stakeholders and senior leadership to ensure alignment. Create clear documentation to keep all parties informed, regardless of organizational changes.

3. **Requirement Risk:**

- **Risk:** The requirements for the ticketing system might not be fully understood or documented, leading to misalignment between business needs and the delivered solution.
- **Mitigation:** Engage stakeholders early in the project to define detailed requirements. Use iterative feedback and validation (e.g., user stories and prototypes) to refine the system according to actual needs.

4. **Business Risk:**

- **Risk:** The new ticketing system might not achieve the expected improvements in customer satisfaction or operational efficiency, leading to a lack of return on investment (ROI).
- **Mitigation:** Set clear, measurable business objectives before the project starts (e.g., reduced response time, SLA compliance). Regularly assess the system post-implementation and adjust processes as needed based on feedback and performance metrics.

8. Business Process Overview

1. Legacy System (AS-IS)

The current ticketing process is largely manual, involving several disconnected systems and processes that result in inefficiencies and delays. Below are the key elements of the legacy system:

- **Ticket Creation:** Customers submit support tickets via email or phone, which are manually entered into the system by agents. This often leads to errors in ticket categorization and delays in assignment.
- **Ticket Assignment:** Tickets are manually assigned to support agents based on availability or expertise. This process is time-consuming and lacks prioritization, leading to unequal workload distribution among agents.
- **SLA Management:** SLA compliance is monitored manually through spreadsheets or ad hoc tracking, which is prone to human error. Tickets often exceed SLA timelines, leading to customer dissatisfaction.
- **Resolution & Closure:** Agents resolve tickets based on available information, but often have limited access to knowledge resources, requiring them to solve recurring issues from scratch. The closure process is also manual and lacks consistency, leading to reopened tickets.

- **Reporting:** Reporting is done manually through ad hoc data collection and spreadsheets, making it difficult to track key performance metrics (KPIs) like ticket resolution time, agent performance, and SLA adherence.
- **Knowledge Sharing:** Information related to past issues and resolutions is stored across different systems, making it challenging for agents to quickly access relevant solutions.

2. Proposed Recommendations (TO-BE)

The proposed ticketing system aims to address the inefficiencies of the legacy system by automating and streamlining key processes. The following changes will improve the overall ticketing process:

- **Ticket Creation:** The new system will allow customers to create tickets through multiple channels (email, web portal, chat, etc.). The system will automatically capture key details such as issue type, urgency, and customer information, eliminating manual data entry and reducing errors.
- **Ticket Assignment:** The system will automatically categorize and prioritize tickets based on predefined rules (e.g., urgency, issue type). Tickets will be assigned to the most appropriate agent based on their expertise and availability, ensuring a more efficient workload distribution.
- **SLA Management:** The new system will include automated SLA tracking, with real-time alerts and escalations for tickets nearing or exceeding their resolution deadlines. This will help ensure that tickets are resolved on time and SLAs are met.
- **Resolution & Closure:** The system will integrate a knowledge base, allowing agents to quickly access solutions for common issues. Automated workflows will guide agents through the ticket resolution process, ensuring consistency and reducing resolution time. Tickets will be automatically closed once all resolution steps are completed, and customers are satisfied.
- **Reporting:** The system will include built-in reporting features, providing real-time dashboards and detailed reports on KPIs such as ticket volume, resolution time, agent performance, and SLA compliance. These reports will be customizable, enabling managers to monitor performance and identify areas for improvement.
- **Knowledge Sharing:** A centralized knowledge base will be integrated into the system, allowing agents to search for and contribute solutions to recurring issues. This will help reduce resolution times, improve consistency, and enable new agents to ramp up more quickly.

9. Business Requirements

Ticket Creation and Submission

The system must allow customers to submit tickets through multiple channels, including email, web portal, and chat, ensuring seamless ticket creation across various platforms.

Automated Ticket Categorization and Prioritization

The system must automatically categorize and prioritize tickets based on predefined criteria such as issue type, urgency, and customer profile.

SLA Management and Alerts

The system must track and monitor ticket SLAs automatically, providing real-time alerts and escalation notifications for tickets approaching or breaching their SLA deadlines.

Ticket Assignment and Routing

The system must automatically assign tickets to the most appropriate agent based on factors such as expertise, workload, and availability.

Knowledge Base Integration

The system must integrate a centralized knowledge base that allows agents to access solutions for recurring issues. The knowledge base must be easily searchable and allow agents to contribute new solutions.

Ticket Resolution Workflow

The system must provide an automated workflow for ticket resolution that guides agents through the necessary steps, ensuring consistency in the resolution process.

Ticket History and Audit Trail

The system must maintain a complete history of all interactions related to each ticket, including agent notes, customer updates, and actions taken.

Reporting and Dashboards

The system must include customizable reporting features and real-time dashboards to track KPIs such as ticket volume, resolution time, SLA compliance, and agent performance.

User Roles and Permissions

The system must have configurable user roles and permissions to restrict access to sensitive data and ensure that agents and managers can only view or edit information relevant to their roles.

Multi-Language Support

The system must support multiple languages to cater to a diverse customer base, allowing customers and agents to interact in their preferred language.

.

Mobile Access for Support Agents

The system must provide mobile access for support agents, allowing them to view and manage tickets from anywhere.

Ticket Escalation Process

The system must include an automated ticket escalation process that triggers based on predefined conditions (e.g., unresolved ticket for a certain period or SLA breach).

Customer Feedback and Satisfaction Tracking

The system must allow customers to provide feedback on ticket resolution and agent performance, and this data must be captured and analyzed for continuous improvement.

Security and Data Privacy Compliance

The system must comply with relevant data privacy regulations (e.g., GDPR) and ensure secure handling of sensitive customer data.

10. Appendices

10.1 List of Acronyms

- ➢ **AI** - Artificial Intelligence
- ➢ **GDPR** - General Data Protection Regulation
- ➢ **UAT** - User Acceptance Testing
- ➢ **D&I** - Diversity and Inclusion
- ➢ **ROI** - Return on Investment
- ➢ **AS-IS** - Current State of the Process/System
- ➢ **TO-BE** - Future State of the Process/System
- ➢ **KPI** - Key Performance Indicator
- ➢ **TAT** - Turnaround Time
- ➢ **SLA** - Service Level Agreement
- ➢ **API** - Application Programming Interface
- ➢ **UI** - User Interface
- ➢ **UX** - User Experience
- ➢ **BRD** - Business Requirements Document
- ➢ **RTM** - Requirement Traceability Matrix
- ➢ **DPA** - Data Protection Act (often used with GDPR)

1. **Ticket**

A record of a customer's issue, question, or request that is tracked and managed through the ticketing system. Tickets are created by customers or automatically generated by the system, and they represent an individual support case to be resolved by an agent.

2. **SLA (Service Level Agreement)**

   A formal agreement between the company and the customer that defines the expected timeframes for responding to and resolving tickets. SLAs outline the maximum time allowed for each stage of the ticket lifecycle.

3. **Knowledge Base**

   A centralized repository of articles, solutions, FAQs, and troubleshooting guides accessible to support agents to assist in resolving tickets more efficiently.

4. **Ticket Assignment**

   The process of routing a support ticket to the appropriate agent or team for resolution, based on factors such as expertise, availability, and workload..

5. **Ticket Lifecycle**

   The various stages a ticket goes through from creation to closure. These stages typically include ticket creation, categorization, assignment, resolution, and closure.

6. **Escalation**

   The process of transferring a ticket to a higher level of support or management when it cannot be resolved at the current level within the defined SLA or requires additional expertise.

7. **Agent**

   A support team member responsible for handling and resolving customer tickets within the ticketing system.

8. **Customer Feedback**

   Feedback provided by customers after the resolution of their support tickets, typically through surveys or ratings.

9. **Report/Reporting Dashboard**

   A tool within the ticketing system that generates visual or written reports based on key performance metrics such as ticket volume, SLA compliance, resolution times, and agent performance.

10. **Multi-channel Support**

    The ability for customers to submit tickets through various communication channels, such as email, web forms, live chat, or social media.

11. **Ticket Categorization**

    The process of assigning tickets to specific categories or types, such as technical support, billing inquiries, or product issues, to streamline the resolution process.

12. **Ticket Status**

    The current state of a ticket, indicating its progress in the resolution process (e.g., New, In Progress, Awaiting Customer Response, Resolved, Closed).

13. **Automated Workflow**

    A set of predefined rules or processes that automatically guide tickets through their lifecycle, such as ticket assignment, escalations, and reminders for overdue tickets.

14. **Ticket Closure**

    The final stage in the ticket lifecycle, where a ticket is marked as resolved and closed after the customer's issue has been addressed and no further action is needed.

15. **User Role/Permissions**

    The access level and permissions assigned to different users within the ticketing system, such as agents, managers, and administrators, to ensure appropriate access to data and functions.

10.3 Related Documents

- ➢ Business Case Document
- ➢ Business Requirement Document

SRS

1. Purpose

The purpose of the Ticketing Life Cycle System is to streamline the process of issue reporting, tracking, and resolution within an organization. It provides users with a platform to create tickets for their concerns or inquiries, assigns these tickets to appropriate agents, and ensures timely updates on their progress. The system fosters effective communication between users and support teams, enhancing customer satisfaction while improving the efficiency of ticket management processes.

2. Scope

The Ticketing Life Cycle System is designed to cater to organizations of varying sizes, supporting multiple roles such as users, agents, and administrators. Key functionalities include:

- **Ticket Creation and Management:** Users can report issues, which are categorized and prioritized for resolution.
- **Agent Assignment and Resolution:** Tickets are assigned to agents based on predefined rules and availability.
- **Role-Based Dashboards:** Users, agents, and administrators have tailored views to manage their specific tasks efficiently.
- **Tracking and Notifications:** The system keeps stakeholders informed through real-time updates and alerts.
- **Analytics and Reporting:** Administrators can analyze trends, monitor performance, and generate reports to optimize operations.

The system supports scalability, security, and compliance with industry standards, making it suitable for technical support, customer service, and other business functions requiring issue resolution.

3. Overview

The Ticketing Life Cycle System is a web-based application offering a user-friendly interface accessible across devices. It incorporates modules for:

- **User Management:** Facilitates user registration, login, and profile management.
- **Ticket Management:** Covers the full ticketing workflow, including creation, assignment, status updates, and resolution.
- **Agent Management:** Allows administrators to assign roles, monitor workloads, and evaluate agent performance.
- **Notification System:** Ensures users and agents receive timely updates about ticket status and escalations.
- **Integration and Extensibility:** The system supports integration with third-party tools like Slack or Microsoft Teams and offers APIs for additional customization.

5. Software Interfaces

- Operating Systems:
  - The system shall be compatible with Windows, macOS, and Linux for on-premises deployments and accessible on any OS via a browser for cloud-based deployments.

- Web Browsers:
  - Supports modern web browsers like Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari (latest versions).

- Database Management System
  - Utilizes relational databases such as MySQL, PostgreSQL, or cloud-based alternatives like AWS RDS or Azure SQL Database.

- Notification Services
  - Integrates with email systems (e.g., SMTP) and SMS gateways (e.g., Twilio) for sending ticket status notifications.

6. Hardware Interfaces

- User devise
  - Users and agents require devices like desktops, laptops, tablets, or smartphones with internet connectivity to access the system.

- Network Infrastructure
  - Requires a reliable internet connection with a minimum bandwidth of 10 Mbps for smooth access and operation.

Functional Requirement

Req ID	Requirement Name	Requirement Description	Priority
FR-001	User Registration	The system shall allow users to register by providing their name, email, and password.	High
FR-002	User Login	The system shall allow registered users to log in using their email and password.	High
FR-003	Ticket Creation	Users shall be able to create new tickets by entering a title, description, category, and priority.	High
FR-004	Ticket Assignment	The system shall automatically assign a ticket to an available agent based on the category.	High
FR-005	Ticket Status Update	Agents shall update the status of tickets to Open, In Progress, or Closed.	High
FR-006	Ticket Priority Update	The system shall allow users or agents to update the ticket's priority.	Medium
FR-007	View Ticket Details	Users and agents shall be able to view all ticket details, including status, category, and assigned agent.	High
FR-008	User Dashboard	Users shall have a dashboard displaying their open and	Medium

		closed tickets.	
FR-009	Agent Dashboard	Agents shall have a dashboard displaying assigned tickets with filters for status and priority.	Medium
FR-010	Ticket Search	Users and agents shall be able to search for tickets using keywords or filters like status and priority.	High
FR-011	Add Ticket Comments	Users and agents shall add comments to tickets for better collaboration.	Medium
FR-012	Email Notifications	The system shall send email notifications for ticket updates, such as status changes or new comments.	Medium
FR-013	Auto-assign Ticket to Agent	The system shall use predefined rules to assign tickets automatically to agents based on their availability and category.	High
FR-014	Ticket Escalation	Tickets not resolved within a specific timeframe shall be automatically escalated to a higher authority.	Medium
FR-015	Role-Based Access Control	The system shall provide role-based access control, restricting features for users, agents, and admins.	Medium
FR-016	View Ticket History	Users and agents shall view the history of changes made to a ticket, including status and priority updates.	High
FR-017	Add Attachments to Tickets	Users and agents shall attach files to tickets for additional context or support.	Medium
FR-018	View Agent Performance	Admins shall view performance metrics for agents, such as the number of resolved tickets.	Medium
FR-019	Define Ticket Categories	Admins shall create and manage ticket categories, such as Technical or Billing.	Low
FR-020	Filter Tickets	Users and agents shall filter tickets by status, priority, or category in their dashboards.	Medium
FR-021	SLA Configuration	The system shall allow admins to configure Service Level Agreements (SLAs) for ticket resolution.	Medium
FR-022	Audit Trail	The system shall maintain an audit trail of all ticket updates for compliance and troubleshooting.	Medium
FR-023	Mobile-Friendly Interface	The system shall provide a mobile-friendly interface for creating and managing tickets.	High
FR-024	Priority-Based Alerts	The system shall notify agents about high-priority tickets through pop-up alerts or email.	Medium
FR-025	Auto-Status Transition	The system shall automatically transition a ticket's status to "In Progress" when an agent starts working on it.	Medium
FR-026	Mobile-Friendly Interface	The system shall provide a mobile-friendly interface for creating and managing tickets.	Medium
FR-027	Export Ticket Data	Users, agents, and admins shall export ticket data in CSV or Excel format.	Low
FR-028	Multi-Language Support	The system shall support multiple languages for users in different regions.	Low
FR-029	Customize Ticket Fields	Admins shall customize ticket fields, adding new ones if required.	Low
FR-030	Agent	Admins shall reassign tickets from one agent to another.	Medium

	Reassignment		
FR-031	Ticket Merging	The system shall allow agents to merge duplicate tickets.	Medium
FR-032	Report Generation	Admins shall generate reports on ticket trends, resolution times, and category-wise breakdowns.	Medium
FR-033	Feedback Collection	Users shall provide feedback on ticket resolution.	Medium
FR-034	Archive Closed Tickets	The system shall archive closed tickets after a specified period.	Low
FR-035	Integration with Third-Party Tools	The system shall integrate with third-party tools like Slack or Microsoft Teams for ticket updates.	Low
FR-036	Ticket Duplication Check	The system shall check for duplicate tickets based on title and description.	Medium
FR-037	Visual Ticket Status Indicators	The system shall provide visual indicators (e.g., color codes) for ticket status and priority.	Low
FR-038	SLA Violation Alerts	The system shall alert agents and admins when a ticket is at risk of breaching SLA deadlines.	Medium
FR-039	Delete User Account	Users shall delete their accounts and associated data, adhering to data protection regulations.	Low
FR-040	Accessibility Compliance	The system shall comply with accessibility standards like WCAG 2.1 for users with disabilities.	High

Non-Functional Requirements

Req ID	Requirement Name	Requirement Description
NFR001	System Availability	The system shall maintain an uptime of 99.9% to ensure uninterrupted access to users and agents.
NFR002	Performance	The system shall handle up to 1,000 concurrent users without degradation in performance.
NFR003	Scalability	The system shall scale horizontally to accommodate up to 10,000 users and 50,000 tickets per month.
NFR004	Security	The system shall comply with industry security standards, such as OWASP guidelines, to prevent unauthorized access.
NFR005	Data Encryption	All sensitive data, including passwords and ticket information, shall be encrypted at rest and in transit.
NFR006	Response Time	The system shall provide responses to user actions, such as ticket creation, within 2 seconds under normal load.
NFR007	Browser Compatibility	The system shall support all major browsers, including Chrome, Firefox, Safari, and Edge.
NFR008	Mobile Compatibility	The system shall provide a responsive design for seamless operation on mobile devices.
NFR009	Accessibility Compliance	The system shall adhere to WCAG 2.1 Level AA standards to ensure accessibility for users with disabilities.
NFR010	Maintainability	The system shall allow developers to update or extend features

		with minimal impact on existing functionality.
NFR011	Backup and Recovery	The system shall perform daily backups and provide data recovery within 2 hours in case of failure.
NFR012	Logging and Monitoring	The system shall log all critical events and provide real-time monitoring for troubleshooting and performance analysis.
NFR013	Usability	The system shall provide an intuitive user interface, requiring no more than 30 minutes of training for basic operations.
NFR014	Localization	The system shall support localization for at least 5 languages, including English, Spanish, French, German, and Chinese.
NFR015	Data Retention	Closed tickets and related data shall be retained for a minimum of 5 years.
NFR016	Integration	The system shall integrate with third-party tools such as Slack, Microsoft Teams, and email systems.
NFR017	Fault Tolerance	The system shall automatically recover from a single point of failure within 30 seconds.
NFR018	Auditability	All user and system actions shall be auditable for compliance and troubleshooting purposes.
NFR019	Cost Efficiency	The system shall operate within an annual maintenance budget of $50,000.
NFR020	API Response Time	The system's APIs shall respond to requests within 500 milliseconds under normal load.

Use Case Specifications

Use Case 1: Raise a New Ticket

1. Use Case Name:
Raise a New Ticket

2. Description:
A user creates a ticket to report an issue, provide details, and request assistance.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the ticketing system.
2. User navigates to the "Raise a Ticket" section.
3. User selects the issue type.

4. User provides a detailed description of the issue.
5. User attaches files, if necessary.
6. User sets the ticket priority.
7. User submits the ticket.
8. System generates a unique ticket ID and confirms submission.

5. Alternate Flow:

- **5a.** User saves the ticket as a draft instead of submitting it.
    1. System stores the ticket as a draft for later editing.

6. Exceptional Flow:

- **6a.** Required fields are incomplete:
    1. System highlights missing fields and prevents submission.
- **6b.** File exceeds the allowed size limit:
    1. System displays an error and prevents file upload.

7. Pre-Conditions:

- User must have an active account.

8. Post-Conditions:

- Ticket is logged in the system and assigned a unique ID.
- Notification is sent to the user.

9. Assumptions:

- Users know how to navigate the ticketing system interface.

10. Constraints:

- Attachments must not exceed 5MB per file.
- Submission allowed only during system operational hours.

11. Dependencies:

- Database and notification systems must be operational.

12. Inputs and Outputs:

- **Inputs:** Issue type, description, priority, optional attachments.
- **Outputs:** Ticket ID, confirmation notification.

13. Business Rules:

- Tickets must include all mandatory fields before submission.
- Duplicate tickets should be flagged.

14. Miscellaneous Information:

- Interface should support mobile and desktop views.

Use Case 2: View Ticket Details

1. Use Case Name:
View Ticket Details

2. Description:
A user views the details of a previously submitted ticket.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to "My Tickets."
3. User selects a ticket to view.
4. System displays ticket details: ID, issue type, description, status, and history.

5. Alternate Flow:

- **4a.** If the ticket is closed:
    - System displays resolution details.

6. Exceptional Flow:

- **6a.** Ticket is not accessible:
    1. System displays "Ticket not found" error.

7. Pre-Conditions:

- User must have access rights to the ticket.

8. Post-Conditions:

- Ticket details are displayed to the user.

9. Assumptions:

- System’s database is up-to-date.

10. Constraints:

- Users can view only their own tickets.

11. Dependencies:

- Database must store ticket details accurately.

12. Inputs and Outputs:

- **Inputs:** Ticket ID or selection from list.
- **Outputs:** Detailed ticket information.

13. Business Rules:

- Closed tickets should display resolution summaries.

14. Miscellaneous Information:

- Interface may limit details for mobile views.

Use Case 3: Edit Draft Ticket

1. Use Case Name:
Edit Draft Ticket

2. Description:
User updates the details of a saved draft ticket.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to the "My Drafts" section.
3. User selects a draft ticket.
4. User updates fields and saves changes or submits the ticket.

5. Alternate Flow:

- **4a.** If the user decides not to submit the ticket:
    1. User saves changes and exits without submitting.

6. Exceptional Flow:

- **6a.** Draft is not found:
    1. System displays "Draft not found" error.

7. Pre-Conditions:

- A draft ticket must exist in the system.

8. Post-Conditions:

- Draft is updated, or ticket is submitted.

9. Assumptions:

- User has edit rights for the draft.

10. Constraints:

- Drafts expire after 30 days.

11. Dependencies:

- Drafts must be stored in the system.

12. Inputs and Outputs:

- **Inputs:** Updated ticket details.
- **Outputs:** Updated draft or new ticket ID.

13. Business Rules:

- Mandatory fields must be completed for submission.

14. Miscellaneous Information:

- Drafts support multiple edits before submission.

Use Case 4: Delete Draft Ticket

1. Use Case Name:
Delete Draft Ticket

2. Description:
User deletes a saved draft ticket they no longer wish to submit.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to the "My Drafts" section.
3. User selects a draft ticket.
4. User chooses the "Delete" option.
5. System confirms deletion.

5. Alternate Flow:

- **5a.** If the user cancels the deletion:
  - Draft remains in the system.

6. Exceptional Flow:

- Draft not found:
  - System displays "Draft not found" error.

7. Pre-Conditions:

- Draft ticket exists in the system.

8. Post-Conditions:

- Draft is removed from the system.

9. Assumptions:

- User has the right to delete drafts.

10. Constraints:

- Drafts cannot be recovered after deletion.

11. Dependencies:

- Draft tickets must be stored in the system.

12. Inputs and Outputs:

- **Inputs:** Draft ticket ID.
- **Outputs:** Deletion confirmation.

13. Business Rules:

- Only the ticket owner can delete their draft.

14. Miscellaneous Information:

- Deletion requires user confirmation.

Use Case 5: Submit Feedback for a Resolved Ticket

1. Use Case Name:
Submit Feedback for a Resolved Ticket

2. Description:
A user provides feedback on the resolution of a ticket.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to "Resolved Tickets."
3. User selects a ticket.
4. System displays the resolution details.
5. User submits feedback, including ratings and optional comments.
6. System saves the feedback and updates the ticket record.

5. Alternate Flow:

- **5a.** User opts to skip providing feedback.

6. Exceptional Flow:

- **6a.** Feedback submission fails due to system error:
    1. System notifies the user and prompts them to retry later.

7. Pre-Conditions:

- Ticket must have a "Resolved" status.

8. Post-Conditions:

- Feedback is successfully recorded in the system.

9. Assumptions:

- User understands how to provide feedback.

10. Constraints:

- Feedback must be submitted within 30 days of ticket resolution.

11. Dependencies:

- Ticket details and resolution history must be accessible.

12. Inputs and Outputs:

- **Inputs:** Rating, comments.
- **Outputs:** Confirmation of feedback submission.

13. Business Rules:

- Users can submit feedback only once per ticket.

14. Miscellaneous Information:

- Feedback may influence team performance metrics.

Use Case 6: Reopen a Closed Ticket

1. Use Case Name:
Reopen a Closed Ticket

2. Description:
Users can reopen a ticket if the issue persists after it was marked resolved.

3. Actors:

- **Primary:** End-User

- **Secondary:** Support Team, System

4. Basic Flow:

1. User logs into the system.
2. User navigates to "Closed Tickets."
3. User selects a ticket and clicks "Reopen."
4. System changes the ticket status to "Reopened" and notifies the support team.

5. Alternate Flow:

- **5a.** User provides additional information while reopening the ticket.

6. Exceptional Flow:

- **6a.** Reopening is not allowed due to policy:
    1. System displays a message stating, "Reopening is not permitted for this ticket."

7. Pre-Conditions:

- Ticket must have a "Closed" status.

8. Post-Conditions:

- Ticket status changes to "Reopened."
- Notification is sent to the support team.

9. Assumptions:

- Users understand when reopening a ticket is appropriate.

10. Constraints:

- Tickets can only be reopened within 15 days of closure.

11. Dependencies:

- Notification system must be functional.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, additional details (optional).
- **Outputs:** Updated ticket status, notification.

13. Business Rules:

- Tickets closed due to policy violations cannot be reopened.

14. Miscellaneous Information:

- Reopening requests are logged for audit purposes.

Use Case 7: Attach Files to a Ticket

1. Use Case Name:
Attach Files to a Ticket

2. Description:
Users can upload files to provide additional context for a ticket.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User navigates to the ticket submission or update page.
2. User clicks "Attach File."
3. User selects a file from their device.
4. System uploads and attaches the file to the ticket.

5. Alternate Flow:

- User removes the attachment before submitting the ticket.

6. Exceptional Flow:

- **6a.** File exceeds size limit:
    1. System displays an error message and cancels the upload.

7. Pre-Conditions:

- Ticket must be in draft or active status.

8. Post-Conditions:

- File is successfully attached to the ticket.

9. Assumptions:

- User has the required file ready to upload.

10. Constraints:

- Maximum file size is 5MB.
- Only specific file types (e.g., PDF, PNG) are allowed.

11. Dependencies:

- Storage system must be operational.

12. Inputs and Outputs:

- **Inputs:** File.
- **Outputs:** Confirmation of successful upload.

13. Business Rules:

- Sensitive files must be flagged for restricted access.

14. Miscellaneous Information:

- Attachments are virus-scanned before being uploaded.

Use Case 8: Set Priority for a Ticket

1. Use Case Name:
Set Priority for a Ticket

2. Description:
Users set the priority level for a ticket during submission.

3. Actors:

- **Primary:** End-User
- **Secondary:** System

4. Basic Flow:

1. User selects the priority level (Low, Medium, High) from a dropdown menu.
2. System associates the priority with the ticket.

5. Alternate Flow:

- **5a.** Priority is automatically assigned based on issue type.

6. Exceptional Flow:

- **6a.** Priority selection is missing:
    1. System prompts the user to select a priority before proceeding.

7. Pre-Conditions:

- User must be submitting a new ticket.

8. Post-Conditions:

- Ticket is created with a defined priority level.

9. Assumptions:

- Users understand the importance of priority levels.

10. Constraints:

- System defaults to "Medium" priority if no selection is made.

11. Dependencies:

- Issue type must be mapped to priority rules.

12. Inputs and Outputs:

- **Inputs:** Priority level.
- **Outputs:** Confirmation of priority assignment.

13. Business Rules:

- Only "Critical" priority requires manager approval.

14. Miscellaneous Information:

- Priority levels influence SLA timelines.

Use Case 9: Receive Confirmation for Ticket Submission

1. Use Case Name:
Receive Confirmation for Ticket Submission

2. Description:
Users receive a confirmation notification after submitting a ticket.

3. Actors:

- **Primary:** System
- **Secondary:** End-User

4. Basic Flow:

1. User submits a ticket.
2. System generates a unique ticket ID.
3. System sends a confirmation email to the user.

5. Alternate Flow:

- **5a.** Confirmation is sent via SMS instead of email.

6. Exceptional Flow:

- **6a.** Email server is unavailable:
    1. System queues the email for later delivery.

7. Pre-Conditions:

- Ticket submission process is completed successfully.

8. Post-Conditions:

- User is notified of ticket submission.

9. Assumptions:

- User’s contact information is accurate.

10. Constraints:

- Email notifications are sent within 5 minutes of submission.

11. Dependencies:

- Notification system and email server must be operational.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, user email.

- **Outputs:** Confirmation email.

13. Business Rules:

- Confirmation must include ticket details for user reference.

14. Miscellaneous Information:

- Notification preferences determine the delivery channel.

Use Case 10: Update Ticket Status

1. Use Case Name:
Update Ticket Status

2. Description:
Support staff updates the status of a ticket as it progresses through different stages (e.g., Open, In Progress, Resolved).

3. Actors:

- **Primary:** Support Staff
- **Secondary:** System

4. Basic Flow:

1. Support staff logs into the system.
2. Support staff navigates to the assigned ticket list.
3. Support staff selects a ticket and clicks "Update Status."
4. Support staff chooses the new status from the dropdown menu.
5. System updates the status and logs the action in the ticket history.

5. Alternate Flow:

- **5a.** Support staff adds comments while updating the status:
  - System logs the comments along with the status change.

6. Exceptional Flow:

- **6a.** Attempt to update status fails due to a system error:
  - System displays an error message and prevents the update.

7. Pre-Conditions:

- Ticket must exist in the system.
- Support staff must have appropriate permissions.

8. Post-Conditions:

- Ticket status is updated and logged in the history.

9. Assumptions:

- Support staff is trained on status workflows.

10. Constraints:

- Certain statuses (e.g., "Closed") can only be applied after manager approval.

11. Dependencies:

- Status workflow must be predefined in the system.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, new status, optional comments.
- **Outputs:** Updated ticket status, logged history entry.

13. Business Rules:

- Only authorized users can update ticket statuses.

14. Miscellaneous Information:

- Status changes trigger notifications to relevant users.

Use Case 11: Assign a Ticket to a Support Team

1. Use Case Name:
Assign a Ticket to a Support Team

2. Description:
The system or a support manager assigns a ticket to the appropriate support team for resolution.

3. Actors:

- **Primary:** Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support manager logs into the system.
2. Manager navigates to unassigned tickets.
3. Manager selects a ticket and clicks "Assign."
4. Manager selects the appropriate support team from a dropdown menu.
5. System assigns the ticket to the selected team and notifies them.

5. Alternate Flow:

- **5a.** Ticket is automatically assigned by the system based on issue type.

6. Exceptional Flow:

- **6a.** No team is available for assignment:
  - System flags the ticket as "Unassigned" and notifies the manager.

7. Pre-Conditions:

- Ticket must exist in the system.

8. Post-Conditions:

- Ticket is assigned to a support team.
- Notification is sent to the team.

9. Assumptions:

- Support teams are predefined in the system.

10. Constraints:

- Tickets can only be assigned to one team at a time.

11. Dependencies:

- System must have an updated list of available teams.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, team selection.
- **Outputs:** Notification of assignment.

13. Business Rules:

- High-priority tickets must be assigned within 1 hour.

14. Miscellaneous Information:

- Assignment logs are tracked for audit purposes.

Use Case 12: Add Comments to a Ticket

1. Use Case Name:
Add Comments to a Ticket

2. Description:
Support staff or users can add comments to a ticket to clarify issues or provide updates.

3. Actors:

- **Primary:** End-User, Support Staff
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to a ticket.
2. User clicks "Add Comment."
3. User enters a comment in the text box.
4. System saves the comment and adds it to the ticket history.

5. Alternate Flow:

- **5a.** User attaches a file along with the comment.

6. Exceptional Flow:

- **6a.** System fails to save the comment due to a network error:
    - User is prompted to retry.

7. Pre-Conditions:

- Ticket must exist in the system.

8. Post-Conditions:

- Comment is added to the ticket.

9. Assumptions:

- Users have permissions to add comments.

10. Constraints:

- Comments are limited to 1,000 characters.

11. Dependencies:

- Database must store comment history.

12. Inputs and Outputs:

- **Inputs:** Text comment, optional attachment.
- **Outputs:** Confirmation of comment addition.

13. Business Rules:

- Comments cannot be edited after submission.

14. Miscellaneous Information:

- Comments are timestamped for tracking.

Use Case 13: Escalate a Ticket

1. Use Case Name:
Escalate a Ticket

2. Description:
Support staff or users escalate a ticket when the resolution is delayed or inadequate.

3. Actors:

- **Primary:** End-User, Support Staff
- **Secondary:** Manager, System

4. Basic Flow:

1. User navigates to the ticket and selects "Escalate."
2. User provides a reason for escalation.
3. System flags the ticket as "Escalated" and notifies the manager.

5. Alternate Flow:

- **5a.** Escalation is automatically triggered if SLA is breached.

6. Exceptional Flow:

- **6a.** Escalation fails due to missing manager assignment:
    - System notifies the user of the error.

7. Pre-Conditions:

- Ticket must have an "Active" status.

8. Post-Conditions:

- Ticket is marked as "Escalated."

9. Assumptions:

- Users understand escalation policies.

10. Constraints:

- Escalation is limited to once per ticket.

11. Dependencies:

- Manager roles must be defined in the system.

12. Inputs and Outputs:

- **Inputs:** Reason for escalation.
- **Outputs:** Escalation notification.

13. Business Rules:

- High-priority tickets are escalated automatically after 12 hours of inactivity.

14. Miscellaneous Information:

- Escalated tickets require immediate attention.

Use Case 14: Generate Reports on Ticket Status

1. Use Case Name:
Generate Reports on Ticket Status

2. Description:
Managers generate reports to analyze ticket statuses, including open, closed, and escalated tickets.

3. Actors:

- **Primary:** Manager
- **Secondary:** System

4. Basic Flow:

1. Manager logs into the system.
2. Manager navigates to the "Reports" section.
3. Manager selects a report type and timeframe.
4. System generates the report and displays it.

5. Alternate Flow:

- **5a.** Report is downloaded as a file instead of being displayed.

6. Exceptional Flow:

- **6a.** Report generation fails due to missing data:
    - System displays an error message.

7. Pre-Conditions:

- Manager must have reporting permissions.

8. Post-Conditions:

- Report is generated and available for viewing or download.

9. Assumptions:

- Required data exists in the system.

10. Constraints:

- Reports can only cover up to 1 year of data.

11. Dependencies:

- Reporting module must be functional.

12. Inputs and Outputs:

- **Inputs:** Report type, timeframe.
- **Outputs:** Generated report.

13. Business Rules:

- Reports must be generated within 5 minutes.

14. Miscellaneous Information:

- Reports are exportable in multiple formats (PDF, Excel).

Use Case 15: Assign a Ticket to a Specific Support Agent

1. Use Case Name:
Assign a Ticket to a Specific Support Agent

2. Description:
Support managers assign tickets to specific agents based on their expertise and workload.

3. Actors:

- **Primary:** Support Manager
- **Secondary:** Support Agent, System

4. Basic Flow:

1. Manager logs into the system.
2. Manager navigates to unassigned tickets or reassigns an existing ticket.
3. Manager selects a ticket and clicks "Assign to Agent."
4. Manager selects an agent from a dropdown list.
5. System assigns the ticket to the selected agent and notifies them.

5. Alternate Flow:

- **5a.** Manager uses an auto-suggest feature to filter agents by expertise or workload.

6. Exceptional Flow:

- **6a.** Selected agent is unavailable:
    1. System notifies the manager to select another agent.

7. Pre-Conditions:

- Ticket exists in the system.
- Manager and agents must have valid system accounts.

8. Post-Conditions:

- Ticket is assigned to the specified agent, and the agent is notified.

9. Assumptions:

- Manager is aware of agent availability and skills.

10. Constraints:

- A ticket can only have one assigned agent at a time.

11. Dependencies:

- Agent profiles and workload data must be up-to-date.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, selected agent.
- **Outputs:** Assignment confirmation and notification.

13. Business Rules:

- High-priority tickets must only be assigned to senior agents.

14. Miscellaneous Information:

- Assignment logs include timestamps and manager details.

Use Case 16: Send Automated Ticket Reminders

1. Use Case Name:
Send Automated Ticket Reminders

2. Description:
The system sends automated reminders for unresolved tickets approaching their SLA deadlines.

3. Actors:

- **Primary:** System
- **Secondary:** Support Staff, Managers

4. Basic Flow:

1. System periodically checks ticket statuses.
2. System identifies tickets nearing their SLA deadlines.
3. System sends reminder notifications to assigned staff and managers.

5. Alternate Flow:

- **5a.** User configures custom reminder intervals for specific tickets.

6. Exceptional Flow:

- **6a.** Notification fails to send:
  - System retries sending the notification later.

7. Pre-Conditions:

- Ticket SLA and reminder intervals are configured.

8. Post-Conditions:

- Reminder notifications are sent to relevant users.

9. Assumptions:

- Notification systems (email/SMS) are operational.

10. Constraints:

- Reminders cannot be disabled for high-priority tickets.

11. Dependencies:

- SLA tracking and notification modules must be functional.

12. Inputs and Outputs:

- **Inputs:** Ticket details, SLA data.
- **Outputs:** Reminder notifications.

13. Business Rules:

- Reminders are sent 24 hours and 1 hour before SLA breaches.

14. Miscellaneous Information:

- Reminders are logged for audit purposes.

Use Case 17: View Ticket History

1. Use Case Name:
View Ticket History

2. Description:
Users and support staff can view the complete history of a ticket, including status changes, comments, and updates.

3. Actors:

- **Primary:** End-User, Support Staff
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to a ticket.
2. User clicks "View History."
3. System displays a chronological list of all actions performed on the ticket.

5. Alternate Flow:

- **5a.** User filters the history by specific action types (e.g., comments, status updates).

6. Exceptional Flow:

- **6a.** History data fails to load due to a server error:
    - System displays an error message and prompts the user to retry.

7. Pre-Conditions:

- Ticket must exist in the system.

8. Post-Conditions:

- Ticket history is displayed successfully.

9. Assumptions:

- Users have appropriate permissions to view ticket history.

10. Constraints:

- History logs cannot be edited or deleted.

11. Dependencies:

- Database must store comprehensive ticket logs.

12. Inputs and Outputs:

- **Inputs:** Ticket ID, optional filters.
- **Outputs:** Ticket history.

13. Business Rules:

- Sensitive comments are marked as "Restricted" and only visible to authorized users.

14. Miscellaneous Information:

- History includes timestamps for all actions.

Use Case 18: Manage SLA Configurations

1. Use Case Name:
Manage SLA Configurations

2. Description:
Managers define and update Service Level Agreements (SLAs) for different ticket types.

3. Actors:

- **Primary:** Manager
- **Secondary:** System

4. Basic Flow:

1. Manager logs into the system.
2. Manager navigates to the SLA configuration section.
3. Manager selects a ticket type and modifies the SLA settings (e.g., response time, resolution time).
4. System saves the updated SLA configuration.

5. Alternate Flow:

- **5a.** Manager creates a new SLA configuration for a custom ticket type.

6. Exceptional Flow:

- **6a.** SLA update fails due to validation errors:
    - System notifies the manager of the issue.

7. Pre-Conditions:

- SLA module must be enabled in the system.

8. Post-Conditions:

- SLA configurations are updated successfully.

9. Assumptions:

- Managers understand SLA policies and ticket workflows.

10. Constraints:

- SLA settings cannot conflict with existing policies.

11. Dependencies:

- Ticket types must be predefined in the system.

12. Inputs and Outputs:

- **Inputs:** SLA parameters, ticket type.
- **Outputs:** Confirmation of SLA update.

13. Business Rules:

- Critical tickets must have a resolution time of under 4 hours.

14. Miscellaneous Information:

- SLA changes are logged for compliance audits.

Use Case 19: View Dashboard Analytics

1. Use Case Name:
View Dashboard Analytics

2. Description:
Managers and support staff view analytics on ticket trends, resolutions, and team performance.

3. Actors:

- **Primary:** Manager, Support Staff
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to the dashboard.
2. System displays analytics, including charts and key performance indicators (KPIs).

5. Alternate Flow:

- **5a.** User customizes the dashboard to display specific metrics.

6. Exceptional Flow:

- **6a.** Dashboard fails to load due to a system error:
  - System displays an error message and suggests retrying later.

7. Pre-Conditions:

- Data must exist for the selected timeframe.

8. Post-Conditions:

- Dashboard is displayed with up-to-date analytics.

9. Assumptions:

- Users know how to interpret the analytics provided.

10. Constraints:

- Data refreshes occur every 15 minutes.

11. Dependencies:

- Analytics engine and database must be operational.

12. Inputs and Outputs:

- **Inputs:** Timeframe, metrics selection.
- **Outputs:** Dashboard visualizations.

13. Business Rules:

- Only authorized users can access detailed team performance metrics.

14. Miscellaneous Information:

- Dashboard data is exportable as reports.

Use Case 20: Close a Ticket

1. Use Case Name:
Close a Ticket

2. Description:
A support agent or manager closes a ticket once the issue has been resolved and verified.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager navigates to the ticket and clicks "Close Ticket."
3. System prompts for confirmation of ticket closure.
4. Support agent or manager confirms the closure.
5. System updates the ticket status to "Closed" and logs the closure details.

5. Alternate Flow:

- **5a.** Support agent or manager adds a resolution note before closing the ticket.

6. Exceptional Flow:

- **6a.** Ticket cannot be closed due to incomplete information or unresolved issues:
  - System displays an error message and prevents the closure.

7. Pre-Conditions:

- Ticket status must be "Resolved" or similar.

8. Post-Conditions:

- Ticket is marked as "Closed" in the system.
- Closure details are logged.

9. Assumptions:

- The ticket is fully resolved and ready for closure.

10. Constraints:

- Only agents with "Manager" or "Senior Agent" roles can close tickets.

11. Dependencies:

- The ticket must not have any open dependencies (e.g., unresolved escalations).

12. Inputs and Outputs:

- **Inputs:** Confirmation of ticket closure, optional resolution note.
- **Outputs:** Updated ticket status and logged closure details.

13. Business Rules:

- Closed tickets cannot be reopened unless explicitly flagged for review.

14. Miscellaneous Information:

- Ticket closure triggers a feedback survey to be sent to the customer (if applicable).

Use Case 21: Search Tickets by Criteria

1. Use Case Name:
Search Tickets by Criteria

2. Description:
Users search for tickets in the system using various criteria such as status, priority, date, and agent.

3. Actors:

- **Primary:** End-User, Support Staff, Support Manager
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to the "Search Tickets" page.
3. User enters search criteria (e.g., ticket status, priority, assigned agent).
4. System displays a list of tickets that match the criteria.

5. Alternate Flow:

- **5a.** User applies multiple filters to narrow down the search results.

6. Exceptional Flow:

- **6a.** Search results fail to load due to a database error:
  - System displays an error message and suggests retrying.

7. Pre-Conditions:

- User must be logged in with appropriate search permissions.

8. Post-Conditions:

- Search results are displayed, showing relevant tickets.

9. Assumptions:

- The system has a sufficiently large and well-maintained dataset of tickets.

10. Constraints:

- Search results may be limited to a maximum of 500 tickets.

11. Dependencies:

- The system must have a functional search engine and updated ticket data.

12. Inputs and Outputs:

- **Inputs:** Search criteria (e.g., ticket status, priority, agent).
- **Outputs:** List of matching tickets.

13. Business Rules:

- Search results are sorted by ticket creation date by default.

14. Miscellaneous Information:

- User can save frequently used search queries for future use.

Use Case 22: Reopen a Closed Ticket

1. Use Case Name:
Reopen a Closed Ticket

2. Description:
A support agent or manager reopens a closed ticket if the issue is not fully resolved or if new information arises.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager navigates to the closed ticket.
3. Support agent or manager selects "Reopen Ticket."
4. System prompts for a reason to reopen the ticket.
5. Support agent or manager provides the reason and confirms the action.
6. System updates the ticket status to "Open" and logs the reason for reopening.

5. Alternate Flow:

- **5a.** Support agent adds comments or updates to the ticket before reopening.

6. Exceptional Flow:

- **6a.** The ticket cannot be reopened due to restrictions (e.g., closure was final):
  - System notifies the user that reopening is not allowed.

7. Pre-Conditions:

- Ticket must be in the "Closed" status.
- User must have appropriate permissions to reopen the ticket.

8. Post-Conditions:

- Ticket status is changed back to "Open."

9. Assumptions:

- The issue is still unresolved or a new issue has been identified.

10. Constraints:

- Only tickets closed within the last 30 days can be reopened.

11. Dependencies:

- The system must track the ticket's closure date.

12. Inputs and Outputs:

- **Inputs:** Reason for reopening the ticket.
- **Outputs:** Updated ticket status, history log.

13. Business Rules:

- Reopened tickets must be reviewed by a manager.

14. Miscellaneous Information:

- Reopening a ticket triggers a notification to the assigned agent.

Use Case 23: Send Ticket Notifications

1. Use Case Name:
Send Ticket Notifications

2. Description:
The system sends notifications (e.g., email, SMS) to users and support staff about ticket updates, status changes, or assigned actions.

3. Actors:

- **Primary:** System
- **Secondary:** End-User, Support Staff, Support Manager

4. Basic Flow:

1. The system detects a ticket status change, comment update, or other relevant event.
2. The system identifies the appropriate recipients (e.g., assignee, reporter).
3. The system sends notifications to the recipients.

5. Alternate Flow:

- **5a.** User customizes notification preferences to receive specific updates (e.g., only critical updates).

6. Exceptional Flow:

- **6a.** Notification fails to send due to server error:
  - System retries sending the notification after 30 minutes.

7. Pre-Conditions:

- The notification system must be configured and operational.

8. Post-Conditions:

- Recipients receive the appropriate notifications.

9. Assumptions:

- Users have valid contact details (email/SMS) configured in the system.

10. Constraints:

- Notifications are sent within 5 minutes of an event.

11. Dependencies:

- Email and SMS service providers must be integrated with the system.

12. Inputs and Outputs:

- **Inputs:** Ticket event (status change, comment update, etc.).
- **Outputs:** Notification message.

13. Business Rules:

- High-priority tickets trigger immediate notifications to the assigned team and manager.

14. Miscellaneous Information:

- Notifications are logged in the system for audit purposes.

Use Case 24: Merge Duplicate Tickets

1. Use Case Name:
Merge Duplicate Tickets

2. Description:
Support agents or managers merge tickets that are identified as duplicates to prevent redundant work.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager identifies two tickets as duplicates.

2. Support agent or manager selects both tickets and clicks "Merge."
3. System prompts the user to confirm the merge.
4. Support agent or manager confirms the merge.
5. System merges the tickets, keeping the most relevant data and closing the duplicate ticket.

5. Alternate Flow:

- **5a.** Support agent adds comments to explain the reason for the merge.

6. Exceptional Flow:

- **6a.** Tickets cannot be merged due to missing or conflicting information:
  - System displays an error message and prevents the merge.

7. Pre-Conditions:

- Tickets to be merged must be open or unresolved.

8. Post-Conditions:

- Duplicate ticket is closed, and the information is merged into the primary ticket.

9. Assumptions:

- The system can accurately detect duplicate tickets based on certain criteria (e.g., issue type, description).

10. Constraints:

- Only tickets with the same issue type can be merged.

11. Dependencies:

- The system must be able to identify and compare ticket data.

12. Inputs and Outputs:

- **Inputs:** Selected duplicate tickets.
- **Outputs:** Merged ticket details and status update.

13. Business Rules:

- Merged tickets retain the most recent updates.

14. Miscellaneous Information:

- Merged tickets are logged for auditing

Use Case 25: Assign Priority to a Ticket

1. Use Case Name:
Assign Priority to a Ticket

2. Description:
A support agent or manager assigns a priority level (e.g., Low, Medium, High, Critical) to a ticket based on its urgency and impact.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager navigates to a ticket.
3. Support agent or manager selects "Assign Priority."
4. System presents priority options (e.g., Low, Medium, High, Critical).
5. Support agent or manager selects the appropriate priority.
6. System updates the ticket with the selected priority and logs the change.

5. Alternate Flow:

- **5a.** Support agent or manager uses an auto-suggest feature to assign priority based on predefined criteria (e.g., ticket type, urgency).

6. Exceptional Flow:

- **6a.** Invalid priority selected:
    - System notifies the user and reverts to the last valid priority.

7. Pre-Conditions:

- The ticket exists and is in an unresolved status.

8. Post-Conditions:

- Ticket priority is updated successfully.

9. Assumptions:

- The support agent or manager has the knowledge to assess ticket priority.

10. Constraints:

- Only authorized users (e.g., managers) can assign "Critical" priority.

11. Dependencies:

- The ticket priority options must be predefined in the system.

12. Inputs and Outputs:

- **Inputs:** Priority level (Low, Medium, High, Critical).
- **Outputs:** Updated ticket priority.

13. Business Rules:

- Critical tickets must be resolved within 4 hours.

14. Miscellaneous Information:

- Changes to ticket priority are logged for auditing purposes.

Use Case 26: Send Ticket Escalation Notifications

1. Use Case Name:
Send Ticket Escalation Notifications

2. Description:
The system sends automatic escalation notifications when a ticket exceeds predefined SLA limits or is not resolved within a specified time frame.

3. Actors:

- **Primary:** System
- **Secondary:** Support Agent, Support Manager

4. Basic Flow:

1. The system monitors ticket progress against SLA limits.
2. The system identifies tickets approaching or breaching their SLA limits.
3. The system sends escalation notifications to the assigned support agent and manager.

5. Alternate Flow:

- **5a.** User manually triggers an escalation for high-priority tickets.

6. Exceptional Flow:

- **6a.** Notification fails to send:
    - The system retries sending the notification after a set period.

7. Pre-Conditions:

- SLA limits and escalation rules are configured.

8. Post-Conditions:

- Escalation notifications are sent.

9. Assumptions:

- SLA rules are well-defined and accurately configured.

10. Constraints:

- Escalation notifications cannot be disabled for high-priority tickets.

11. Dependencies:

- The system must be integrated with a notification service (email/SMS).

12. Inputs and Outputs:

- **Inputs:** Ticket SLA, time exceeded.
- **Outputs:** Escalation notification.

13. Business Rules:

- High-priority tickets must trigger an escalation after 2 hours if unresolved.

14. Miscellaneous Information:

- Escalation notification logs are stored for compliance purposes.

Use Case 27: Attach Files to a Ticket

1. Use Case Name:
Attach Files to a Ticket

2. Description:
Users can attach files (e.g., screenshots, logs, documents) to a ticket to provide additional context or information for the support agent.

3. Actors:

- **Primary:** End-User, Support Agent
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to the ticket.
2. User clicks on "Attach File."
3. System prompts the user to browse and select a file.
4. User selects the file and clicks "Upload."
5. The system uploads the file and associates it with the ticket.

5. Alternate Flow:

- **5a.** User attaches multiple files at once.

6. Exceptional Flow:

- **6a.** File upload fails due to size limitations:
    - System notifies the user and prevents the upload.

7. Pre-Conditions:

- Ticket exists and the user has proper permissions to attach files.

8. Post-Conditions:

- The selected file(s) are uploaded and attached to the ticket.

9. Assumptions:

- The system supports the file types being uploaded.

10. Constraints:

- File size cannot exceed 25MB.

11. Dependencies:

- File storage and handling modules must be functional.

12. Inputs and Outputs:

- **Inputs:** Selected file(s).
- **Outputs:** File attachment confirmation.

13. Business Rules:

- Only certain file types (e.g., PDF, JPG, PNG) can be uploaded.

14. Miscellaneous Information:

- Files are stored in the system for future reference and auditing.

Use Case 28: Assign Ticket to a Queue

1. Use Case Name:
Assign Ticket to a Queue

2. Description:
Tickets are assigned to specific queues (e.g., technical support, billing issues) for appropriate handling based on the issue type.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager selects a ticket and clicks "Assign to Queue."
3. System displays a list of available queues.
4. Support agent or manager selects the appropriate queue.
5. System assigns the ticket to the selected queue and notifies the relevant support team.

5. Alternate Flow:

- **5a.** System automatically assigns tickets to predefined queues based on the issue type.

6. Exceptional Flow:

- **6a.** Queue is full or unavailable:
  - System alerts the user to select another queue or reassign the ticket manually.

7. Pre-Conditions:

- Ticket must be open and ready to be assigned to a queue.

8. Post-Conditions:

- Ticket is assigned to the appropriate queue.

9. Assumptions:

- Support teams are aware of which queues handle specific types of tickets.

10. Constraints:

- Tickets can only belong to one queue at a time.

11. Dependencies:

- Queue management and classification modules must be operational.

12. Inputs and Outputs:

- **Inputs:** Selected queue.
- **Outputs:** Ticket assigned to the queue.

13. Business Rules:

- Tickets related to billing issues are automatically assigned to the billing queue.

14. Miscellaneous Information:

- Queue assignment is logged for tracking purposes.

Use Case 29: View Ticket Statistics

1. Use Case Name:
View Ticket Statistics

2. Description:
Managers and support staff view statistical data related to tickets, such as the number of open tickets, average resolution time, etc.

3. Actors:

- **Primary:** Manager, Support Staff
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to the "Ticket Statistics" section.
2. System displays various statistical metrics (e.g., open tickets, resolved tickets, average resolution time).

5. Alternate Flow:

- **5a.** User filters the statistics by date, ticket type, or priority.

6. Exceptional Flow:

- **6a.** Data fails to load due to an error:
  - System displays an error message and suggests retrying.

7. Pre-Conditions:

- Data must exist in the system for the selected timeframe.

8. Post-Conditions:

- Ticket statistics are displayed successfully.

9. Assumptions:

- The system accurately tracks and processes ticket statistics.

10. Constraints:

- Statistical data may be delayed by up to 30 minutes.

11. Dependencies:

- The system must have access to ticket data and analytics functionality.

12. Inputs and Outputs:

- **Inputs:** Timeframe and filter parameters.
- **Outputs:** Ticket statistics dashboard.

13. Business Rules:

- Statistics are updated every 15 minutes to reflect real-time ticket data.

14. Miscellaneous Information:

- Statistics can be exported as CSV or PDF reports.

Use Case 30: Log Ticket Comments

1. Use Case Name:
Log Ticket Comments

2. Description:
Support agents and users can log comments on a ticket to communicate updates, clarifications, or additional information.

3. Actors:

- **Primary:** Support Agent, End-User
- **Secondary:** System

4. Basic Flow:

1. User logs into the system and navigates to a ticket.
2. User selects "Add Comment."
3. User types the comment and clicks "Submit."
4. System saves the comment and associates it with the ticket.

5. Alternate Flow:

- **5a.** User attaches files to the comment before submitting.

6. Exceptional Flow:

- **6a.** Comment submission fails due to connectivity issues:
  - System displays an error message and prompts the user to retry.

7. Pre-Conditions:

- Ticket exists, and the user has the necessary permissions to comment.

8. Post-Conditions:

- Comment is added to the ticket history.

9. Assumptions:

- Users have the knowledge to provide relevant and useful comments.

10. Constraints:

- Comments must be under 500 characters.

11. Dependencies:

- The system must support comment logging and ticket history tracking.

12. Inputs and Outputs:

- **Inputs:** Text comment, optional file attachment.
- **Outputs:** Updated ticket with the new comment.

13. Business Rules:

- Comments should be respectful and relevant to the ticket.

14. Miscellaneous Information:

- Comments are time-stamped and attributed to the author.

Use Case 31: Assign Ticket to an Agent

1. Use Case Name:
Assign Ticket to an Agent

2. Description:
Support managers or agents can assign a ticket to a specific support agent for resolution.

3. Actors:

- **Primary:** Support Manager, Support Agent
- **Secondary:** System

4. Basic Flow:

1. Support manager or agent logs into the system.
2. Support manager or agent opens the ticket that needs to be assigned.
3. System displays available agents.
4. Support manager or agent selects an agent and assigns the ticket.
5. System updates the ticket with the new assigned agent and sends a notification to the agent.

5. Alternate Flow:

- **5a.** System automatically assigns the ticket to the most available agent based on workload.

6. Exceptional Flow:

- **6a.** No agents are available for assignment:
    - System notifies the user and suggests retrying later.

7. Pre-Conditions:

- Ticket must be unresolved and need assignment.

8. Post-Conditions:

- Ticket is assigned to the selected agent.

9. Assumptions:

- Agents are available for assignment and have the necessary skills to resolve the issue.

10. Constraints:

- Tickets cannot be assigned to agents with ongoing tickets exceeding their maximum workload.

11. Dependencies:

- System must have real-time agent availability data.

12. Inputs and Outputs:

- **Inputs:** Selected agent.
- **Outputs:** Updated ticket with assigned agent.

13. Business Rules:

- Tickets must be assigned within 30 minutes of creation to meet SLA targets.

14. Miscellaneous Information:

- Ticket assignment logs are stored for audit purposes.

Use Case 32: View Ticket History

1. Use Case Name:
View Ticket History

2. Description:
Users can view the history of a ticket, including comments, status changes, and assigned agents.

3. Actors:

- **Primary:** End-User, Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to a ticket and clicks on "View History."
3. System displays a detailed timeline of the ticket, including status updates, comments, and assignments.

5. Alternate Flow:

- **5a.** User filters the history by date or event type (e.g., only status changes or only comments).

6. Exceptional Flow:

- **6a.** History data is not available:
  - System displays an error message indicating that no history is available for the selected ticket.

7. Pre-Conditions:

- Ticket must exist in the system.

8. Post-Conditions:

- History information is displayed to the user.

9. Assumptions:

- The system maintains an accurate record of ticket history.

10. Constraints:

- History data may be limited to a specific period (e.g., the last 6 months).

11. Dependencies:

- Ticket history logging functionality must be operational.

12. Inputs and Outputs:

- **Inputs:** Ticket selection.
- **Outputs:** Displayed ticket history.

13. Business Rules:

- Only authorized users can view full ticket history.

14. Miscellaneous Information:

- History can be exported as a CSV or PDF.

Use Case 33: Add Ticket Tags

1. Use Case Name:
Add Ticket Tags

2. Description:
Users can add tags to tickets to categorize them based on issue type or other relevant attributes.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User opens a ticket and clicks on "Add Tags."
3. System prompts the user to enter tags.
4. User enters tags and submits.
5. System associates the tags with the ticket and saves them.

5. Alternate Flow:

- **5a.** System suggests pre-configured tags based on the ticket's category.

6. Exceptional Flow:

- **6a.** Tags cannot be added due to system restrictions:
  - System displays an error message and prevents the action.

7. Pre-Conditions:

- The user must have the necessary permissions to add tags.

8. Post-Conditions:

- Tags are added to the ticket and saved in the system.

9. Assumptions:

- Tags are predefined, or users can create new ones.

10. Constraints:

- A ticket can have up to 5 tags.

11. Dependencies:

- Tagging functionality must be integrated with the ticketing system.

12. Inputs and Outputs:

- **Inputs:** Tags (text).
- **Outputs:** Updated ticket with new tags.

13. Business Rules:

- Tags must be relevant and not exceed 50 characters.

14. Miscellaneous Information:

- Tags are searchable, making it easier to filter tickets later.

Use Case 34: Create Knowledge Base Article

1. Use Case Name:
Create Knowledge Base Article

2. Description:
Support agents or managers can create knowledge base articles to assist in resolving recurring issues.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager navigates to the knowledge base section.
3. Support agent or manager clicks on "Create Article."
4. System prompts the user to enter article details (e.g., title, description, steps).
5. Support agent or manager submits the article.
6. System saves the article and makes it available for future reference.

5. Alternate Flow:

- **5a.** User attaches relevant files or screenshots to the article.

6. Exceptional Flow:

- **6a.** Article submission fails due to missing mandatory fields:
  - System displays an error message and prevents submission.

7. Pre-Conditions:

- The user must have knowledge base creation permissions.

8. Post-Conditions:

- The article is added to the knowledge base and accessible to support agents.

9. Assumptions:

- Support agents have the necessary knowledge to create useful articles.

10. Constraints:

- Articles cannot exceed 1000 words.

11. Dependencies:

- The knowledge base system must be operational.

12. Inputs and Outputs:

- **Inputs:** Article title, description, steps, optional file attachments.
- **Outputs:** New article added to the knowledge base.

13. Business Rules:

- Articles must be reviewed and approved by a manager before being published.

14. Miscellaneous Information:

- Articles can be categorized for easier searching.

Use Case 35: Search Knowledge Base

1. Use Case Name:
Search Knowledge Base

2. Description:
Users can search for existing knowledge base articles to find solutions to recurring issues.

3. Actors:

- **Primary:** End-User, Support Agent
- **Secondary:** System

4. Basic Flow:

1. User logs into the system.
2. User navigates to the knowledge base section.
3. User enters a search term or topic.
4. System displays a list of matching articles.
5. User selects an article to view.

5. Alternate Flow:

- **5a.** User refines the search using filters (e.g., date, category).

6. Exceptional Flow:

- **6a.** No results found:
  - System displays a message indicating that no articles match the search.

7. Pre-Conditions:

- The knowledge base must be populated with articles.

8. Post-Conditions:

- The user is presented with search results.

9. Assumptions:

- The knowledge base is regularly updated with new articles.

10. Constraints:

- Search results may be limited to a certain number (e.g., 20 articles).

11. Dependencies:

- The knowledge base search engine must be functional.

12. Inputs and Outputs:

- **Inputs:** Search terms, optional filters.
- **Outputs:** Search results (articles).

13. Business Rules:

- Knowledge base articles are indexed for quick searching.

14. Miscellaneous Information:

- Users can rate articles for usefulness.

Use Case 36: Monitor Ticket SLA Compliance

1. Use Case Name:
Monitor Ticket SLA Compliance

2. Description:
Managers monitor whether tickets meet their SLA targets for timely resolution.

3. Actors:

- **Primary:** Support Manager
- **Secondary:** System

4. Basic Flow:

1. Manager logs into the system and navigates to the "SLA Compliance" dashboard.
2. System displays the current status of all open tickets in relation to their SLA.

3. Manager reviews the SLA compliance for each ticket.

5. Alternate Flow:

- **5a.** Manager applies filters to view only high-priority tickets or tickets approaching SLA breaches.

6. Exceptional Flow:

- **6a.** System cannot retrieve SLA data due to an error:
  - System displays an error message and prompts the user to retry.

7. Pre-Conditions:

- Tickets must have defined SLAs and be actively tracked.

8. Post-Conditions:

- SLA compliance data is displayed for review.

9. Assumptions:

- SLAs are configured for each ticket type.

10. Constraints:

- SLA data may not update in real-time (may have a 5-minute delay).

11. Dependencies:

- The SLA tracking system must be fully integrated with the ticketing system.

12. Inputs and Outputs:

- **Inputs:** SLA configurations.
- **Outputs:** SLA compliance status.

13. Business Rules:

- Tickets must be resolved within the time frame defined by their SLA.

14. Miscellaneous Information:

- SLA compliance data can be exported for reporting.

Use Case 37: Create Custom Ticket Fields

1. Use Case Name:
Create Custom Ticket Fields

2. Description:
Administrators can create custom fields to capture additional information when tickets are created.

3. Actors:

- **Primary:** Administrator
- **Secondary:** System

4. Basic Flow:

1. Administrator logs into the system and navigates to the ticket configuration settings.
2. Administrator selects "Create Custom Field."
3. System prompts the administrator to define the field type (e.g., text, dropdown, date).
4. Administrator defines the custom field details (name, type, options).
5. System saves the custom field and makes it available for ticket creation.

5. Alternate Flow:

- **5a.** Administrator creates multiple custom fields in a batch.

6. Exceptional Flow:

- **6a.** Invalid custom field type selected:
  - System displays an error message and prompts for valid input.

7. Pre-Conditions:

- Administrator has necessary permissions to configure the system.

8. Post-Conditions:

- Custom fields are added to the ticket creation form.

9. Assumptions:

- Custom fields are used to gather relevant data for specific business processes.

10. Constraints:

- Each custom field can only have a maximum of 255 characters for text-based fields.

11. Dependencies:

- Custom fields must be integrated with the ticket creation process.

12. Inputs and Outputs:

- **Inputs:** Field name, type, and options.
- **Outputs:** Updated ticket creation form with new custom fields.

13. Business Rules:

- Custom fields must be clearly defined and documented for support agents.

14. Miscellaneous Information:

- Custom fields are available for reporting and searching.

Use Case 38: Close Ticket

1. Use Case Name:
Close Ticket

2. Description:
Support agents or managers can close a ticket when the issue is resolved and no further action is required.

3. Actors:

- **Primary:** Support Agent, Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support agent or manager logs into the system.
2. Support agent or manager reviews the ticket details to ensure the issue is resolved.
3. Support agent or manager clicks "Close Ticket."
4. System prompts for confirmation.
5. Support agent or manager confirms.
6. System updates the ticket status to "Closed" and notifies the user.

5. Alternate Flow:

- **5a.** Ticket is closed automatically after all required fields are filled out.

6. Exceptional Flow:

- **6a.** Ticket cannot be closed due to incomplete information:
  - System alerts the user and prevents closure until all required data is provided.

7. Pre-Conditions:

- Ticket must be in a resolved status.

8. Post-Conditions:

- Ticket is marked as "Closed" in the system.

9. Assumptions:

- The ticket is fully resolved and does not require additional follow-up.

10. Constraints:

- Tickets cannot be closed if they are pending customer confirmation.

11. Dependencies:

- The system must support ticket closure functionality.

12. Inputs and Outputs:

- **Inputs:** Confirmation to close.
- **Outputs:** Ticket marked as closed.

13. Business Rules:

- Closed tickets are archived after 30 days.

14. Miscellaneous Information:

- Ticket closure is logged for reporting and auditing purposes.

Use Case 39: Reopen Ticket

1. Use Case Name:
Reopen Ticket

2. Description:
Support agents can reopen a closed or resolved ticket if the customer reports an ongoing issue.

3. Actors:

- **Primary:** Support Agent
- **Secondary:** System

4. Basic Flow:

1. Support agent logs into the system and navigates to the closed ticket.
2. Support agent reviews the ticket details.
3. Support agent clicks "Reopen Ticket."
4. System changes the ticket status to "Open" and notifies the assigned agent.

5. Alternate Flow:

- **5a.** System automatically reopens tickets if a new comment is added within 48 hours of closure.

6. Exceptional Flow:

- **6a.** Ticket cannot be reopened due to system restrictions:
    - System displays an error message and provides options for support.

7. Pre-Conditions:

- The ticket must be in a closed or resolved status.

8. Post-Conditions:

- Ticket is reopened and becomes active for further resolution.

9. Assumptions:

- The customer has reported an ongoing issue or dissatisfaction.

10. Constraints:

- A ticket can be reopened only once after closure.

11. Dependencies:

- The ticketing system must allow for status changes after closure.

12. Inputs and Outputs:

- **Inputs:** Request to reopen ticket.
- **Outputs:** Updated ticket status.

13. Business Rules:

- Tickets can only be reopened within 30 days of closure.

14. Miscellaneous Information:

- Reopened tickets are tracked for performance and SLA adherence.

Use Case 40: Generate Ticket Reports

1. Use Case Name:
Generate Ticket Reports

2. Description:
Support managers can generate reports on ticket performance, including resolution times, SLA compliance, and ticket volumes.

3. Actors:

- **Primary:** Support Manager
- **Secondary:** System

4. Basic Flow:

1. Support manager logs into the system and navigates to the reports section.
2. Support manager selects the type of report (e.g., resolution time, SLA compliance).
3. System generates the requested report and displays it.
4. Support manager reviews the report and exports it if necessary.

5. Alternate Flow:

- **5a.** System allows filtering the data based on date ranges or other criteria.

6. Exceptional Flow:

- **6a.** Report generation fails due to system overload:
  - System notifies the user and prompts for retry.

7. Pre-Conditions:

- The system must have sufficient ticket data for reporting.

8. Post-Conditions:

- Report is generated and available for review or export.

9. Assumptions:

- Reports are accurate and reflect real-time ticketing data.

10. Constraints:

- Reports may have a delay of up to 15 minutes due to data processing.

11. Dependencies:

- Reporting functionality must be integrated with ticket data.

12. Inputs and Outputs:

- **Inputs:** Report type, date range, filters.
- **Outputs:** Generated report.

13. Business Rules:

- Reports are subject to access control based on user roles.

14. Miscellaneous Information:

- Reports can be scheduled for periodic generation.

5. User story of shopping from ecommerce.

User Story 1

Story No: 1
Task: Allow users to browse products by category.
Priority: High
Value Statement: As a customer, I want to browse products by category so that I can quickly find items of interest.
BV: 100
CP: 5
Acceptance Criteria:

- Categories are displayed on the homepage and the product listing page.

- Users can filter products within a selected category.

User Story 2

Story No: 2
Task: Enable a product search bar on the homepage.
Priority: High
Value Statement: As a customer, I want to search for products using a search bar so that I can find specific items quickly.
BV: 100
CP: 6
Acceptance Criteria:

- The search bar is visible and functional on the homepage.
- Search results are displayed within 2 seconds after submission.

User Story 3

Story No: 3
Task: Provide product sorting options (e.g., price, popularity).
Priority: High
Value Statement: As a customer, I want to sort products based on price or popularity so that I can make an informed choice.
BV: 800
CP: 8
Acceptance Criteria:

- Sorting options are available on product listing pages.
- Products are sorted correctly based on the selected criteria.

User Story 4

Story No: 4
Task: Implement a user-friendly product details page.
Priority: High
Value Statement: As a customer, I want to see detailed information about a product so that I can decide whether to buy it.
BV: 100
CP: 6
cceptance Criteria:

- The product page includes images, price, description, and reviews.
- "Add to Cart" and "Buy Now" buttons are visible and functional.

User Story 5

Story No: 5
Task: Add a feature for customers to leave reviews and ratings.
Priority: Medium
Value Statement: As a customer, I want to leave a review for a product so that I can share my feedback with other users.
BV: 700
CP: 5
Acceptance Criteria:

- Customers can submit reviews and ratings after purchasing a product.
- Reviews are displayed on the product details page.

User Story 6

Story No: 6
Task: Enable customers to add items to a shopping cart.
Priority: High
Value Statement: As a customer, I want to add products to my shopping cart so that I can purchase multiple items at once.
BV: 100
CP: 9
Acceptance Criteria:

- Items are added to the cart with correct quantity and pricing.
- Users can view, update, or remove items from the cart.

User Story 7

Story No: 7
Task: Create a secure user authentication system.
Priority: High
Value Statement: As a customer, I want to log in securely so that my personal information is protected.
BV: 100

CP: 6
Acceptance Criteria:

- Users can sign up, log in, and reset passwords securely.
- Passwords are encrypted, and two-factor authentication is available.

User Story 8

Story No: 8
Task: Develop a wishlist feature for saving favorite products.
Priority: Medium
Value Statement: As a customer, I want to save items to a wishlist so that I can purchase them later.
BV: 700
CP: 7
Acceptance Criteria:

- Users can add, view, and remove items from the wishlist.
- Wishlist items persist even after logout.

User Story 9

Story No: 9
Task: Implement a secure checkout process.
Priority: High
Value Statement: As a customer, I want a secure and easy checkout process so that I can complete my purchase confidently.
BV: 100
CP: 9
Acceptance Criteria:

- Users can enter billing and shipping details securely.
- Payment is processed securely using industry standards.

User Story 10

Story No: 10
Task: Offer multiple payment methods (credit card, PayPal, etc.).
Priority: High
Value Statement: As a customer, I want to choose my preferred payment method so that I can complete my purchase conveniently.
BV: 900
CP: 9
Acceptance Criteria:

- Payment gateway supports credit card, PayPal, and digital wallets.
- Transactions are processed within 5 seconds.

User Story 11

Story No: 11
Task: Provide a product comparison feature.
Priority: Medium
Value Statement: As a customer, I want to compare products side-by-side so that I can choose the best option.
BV: 700
CP: 5
Acceptance Criteria:

- Users can select up to three products to compare.
- Key attributes are displayed for easy comparison.

User Story 12

Story No: 12
Task: Enable guest checkout without account creation.
Priority: High
Value Statement: As a customer, I want to check out as a guest so that I can make a quick purchase without creating an account.
BV: 800
CP: 7
Acceptance Criteria:

- Users can complete the checkout process without signing in.

- Guest users can track orders using email and order ID.

User Story 13

Story No: 13
Task: Develop an order tracking feature.
Priority: High
Value Statement: As a customer, I want to track my orders so that I can know when they will arrive.
BV: 900
CP: 4
Acceptance Criteria:

- Users can view the status of their orders in real-time.
- Notifications are sent for key updates (e.g., shipped, out for delivery).

User Story 14

Story No: 14
Task: Display related products on product pages.
Priority: Medium
Value Statement: As a customer, I want to see related products so that I can discover complementary items.
BV: 700
CP: 8
Acceptance Criteria:

- Related products are displayed dynamically based on user behavior.
- Clicking on a related product redirects to its details page.

User Story 15

Story No: 15
Task: Add an email and SMS notification system for orders.
Priority: Medium
Value Statement: As a customer, I want to receive notifications about my order status so that I stay informed.
BV: 800
CP: 5
Acceptance Criteria:

- Notifications are sent for order confirmation, shipping, and delivery.
- Users can opt-in or opt-out of notifications.

User Story 16

Story No: 16
Task: Implement advanced search with filters (e.g., price range, brand).
Priority: Medium
Value Statement: As a customer, I want to filter my search results so that I can refine my product search efficiently.
BV: 800
CP: 8
Acceptance Criteria:

- Filters for price range, brand, and ratings are available.
- Results update dynamically based on selected filters.

User Story 17

Story No: 17
Task: Allow users to view their order history.
Priority: Medium
Value Statement: As a customer, I want to view my past orders so that I can reorder items easily.
BV: 700
CP: 7
Acceptance Criteria:

- Users can view a list of their completed orders.
- Clicking on an order shows detailed information, including items and total cost.

User Story 18

Story No: 18
Task: Develop a loyalty rewards program.
Priority: Medium
Value Statement: As a customer, I want to earn rewards for my purchases so that I feel valued.
BV: 700
CP: 9
Acceptance Criteria:

- Users earn points for each purchase.
- Points can be redeemed for discounts during checkout.

User Story 19

Story No: 19
Task: Support multi-language functionality.
Priority: Low
Value Statement: As a global customer, I want to view the site in my preferred language so that I can shop comfortably.
BV: 600
CP: 7
Acceptance Criteria:

- Users can select their preferred language from a dropdown menu.
- All text content updates dynamically based on the selected language.

User Story 20

Story No: 20
Task: Implement a discount and promo code feature.
Priority: High
Value Statement: As a customer, I want to apply promo codes during checkout so that I can save money.
BV: 900
CP: 5
Acceptance Criteria:

- Users can enter promo codes at checkout.
- Discounts are applied correctly, and the updated total is displayed.

User Story 21

Story No: 21
Task: Add a "frequently bought together" section on product pages.
Priority: Medium
Value Statement: As a customer, I want to see related items frequently purchased with the product so that I can bundle my purchase.
BV: 700
CP: 7
Acceptance Criteria:

- Frequently bought together items are displayed dynamically on the product page.
- Users can add all items to the cart with one click.

User Story 22

Story No: 22
Task: Allow customers to schedule delivery times.
Priority: Medium
Value Statement: As a customer, I want to schedule a delivery time so that I can receive my order at a convenient time.
BV: 700
CP: 8
Acceptance Criteria:

- Users can select preferred delivery dates and time slots during checkout.
- Delivery preferences are confirmed in the order summary.

User Story 23

Story No: 23
Task: Enable product availability notifications.
Priority: Medium
Value Statement: As a customer, I want to receive notifications when a product is back in stock so that I don’t miss out.
BV: 800
CP: 8
Acceptance Criteria:

- Users can subscribe to back-in-stock alerts.
- Notifications are sent via email or SMS when the product becomes available.

User Story 24

Story No: 24
Task: Create a user profile page with editable details.
Priority: Medium
Value Statement: As a customer, I want to update my personal information so that my profile is accurate.
BV: 700
CP: 7
Acceptance Criteria:

- Users can update their name, address, and contact details.
- Changes are saved and reflected immediately.

User Story 25

Story No: 25
Task: Add a feature for users to track shipping costs.
Priority: Medium
Value Statement: As a customer, I want to see the shipping cost before completing my purchase so that I know the total expense.
BV: 800
CP: 5
Acceptance Criteria:

- Shipping costs are calculated dynamically based on the delivery location.
- The total cost, including shipping, is displayed in the cart.

User Story 26

Story No: 26
Task: Provide support for multiple currencies.
Priority: Medium
Value Statement: As an international customer, I want to see prices in my local currency so that I can understand costs better.
BV: 600
CP: 6
Acceptance Criteria:

- Users can select their preferred currency from a dropdown menu.
- Prices and totals update dynamically based on the selected currency.

User Story 27

Story No: 27
Task: Add social media sharing for products.
Priority: Low
Value Statement: As a customer, I want to share products on social media so that I can recommend them to others.
BV: 600
CP: 6
Acceptance Criteria:

- Product pages have social media sharing buttons (e.g., Facebook, Twitter, Instagram).
- Clicking a button opens a pre-filled post with the product link.

User Story 28

Story No: 28
Task: Develop a referral program for customers.
Priority: Medium
Value Statement: As a customer, I want to refer my friends and earn rewards so that I feel incentivized to promote the site.
BV: 700
CP: 7
Acceptance Criteria:

- Users receive unique referral codes.
- Referrals result in discounts or rewards for both parties.

User Story 29

Story No: 29
Task: Introduce AI-based personalized product recommendations.
Priority: Medium
Value Statement: As a customer, I want personalized recommendations so that I can discover products I like.
BV: 800
CP: 9
Acceptance Criteria:

- Recommendations are displayed based on browsing and purchase history.

- Recommendations refresh dynamically as users interact with the site.

User Story 30

Story No: 30
Task: Add a blog or resource section for product guides and tips.
Priority: Low
Value Statement: As a customer, I want to read product-related guides so that I can make informed decisions.
BV: 600
CP: 9
Acceptance Criteria:

- Blogs are categorized and accessible from the homepage.
- Users can search for specific topics within the blog section.

User Story 31

Story No: 31
Task: Enable bulk ordering for businesses or large customers.
Priority: Medium
Value Statement: As a bulk buyer, I want to order large quantities of items easily so that I can manage my purchases efficiently.
BV: 800
CP: 6
Acceptance Criteria:

- Users can specify bulk quantities for products.
- Discounts for bulk orders are applied automatically at checkout.

User Story 32

Story No: 32
Task: Add a real-time chat support feature.
Priority: Medium
Value Statement: As a customer, I want instant help through live chat so that I can resolve my queries quickly.
BV: 900
CP: 6
Acceptance Criteria:

- Live chat is accessible from the footer or help section.

- Responses are sent within 30 seconds of initiating the chat.

User Story 33

Story No: 33
Task: Implement a "recently viewed products" section.
Priority: Medium
Value Statement: As a customer, I want to revisit products I recently viewed so that I can make decisions more easily.
BV: 700
CP: 7
Acceptance Criteria:

- Recently viewed products are displayed dynamically on the homepage or product listing page.
- Items persist for the session or up to 24 hours.

User Story 34

Story No: 34
Task: Add a filter for eco-friendly or sustainable products.
Priority: Low
Value Statement: As a customer, I want to filter for eco-friendly products so that I can make environmentally conscious purchases.
BV: 600
CP: 6
Acceptance Criteria:

- An "eco-friendly" filter is available in product search.
- Only products tagged as sustainable appear when the filter is applied.

User Story 35

Story No: 35
Task: Create an admin panel for managing inventory.
Priority: High
Value Statement: As an admin, I want to manage product inventory efficiently so that stock levels are accurate.
BV: 100

CP: 6
Acceptance Criteria:

- Admins can view, add, update, or remove products.
- Low-stock alerts are generated automatically.

User Story 36

Story No: 36
Task: Provide an option for gift wrapping and personalized messages.
Priority: Low
Value Statement: As a customer, I want to add gift-wrapping and a message to my order so that I can send personalized gifts.
BV: 600
CP: 6
Acceptance Criteria:

- Users can select gift wrapping during checkout.
- Users can enter a personalized message up to 250 characters.

User Story 37

Story No: 37
Task: Add subscription-based purchasing for certain products.
Priority: Medium
Value Statement: As a customer, I want to subscribe to recurring purchases so that I don't have to reorder frequently used items manually.
BV: 800
CP: 8
Acceptance Criteria:

- Users can select subscription options (e.g., weekly, monthly).
- Subscriptions can be paused, modified, or canceled at any time.

User Story 38

Story No: 38
Task: Integrate product recommendations based on geographic location.
Priority: Low
Value Statement: As a customer, I want location-specific product recommendations so that I

can discover items relevant to my area.
BV: 700
CP: 8
Acceptance Criteria:

- Recommendations are tailored based on the user's location.
- Location data is obtained with consent and is updateable.

User Story 39

Story No: 39
Task: Develop a feature to calculate estimated delivery times.
Priority: Medium
Value Statement: As a customer, I want to see an estimated delivery date so that I can plan accordingly.
BV: 900
CP: 9
Acceptance Criteria:

- Delivery dates are displayed based on the shipping method and location.
- Estimates are updated dynamically for different delivery options.

User Story 40

Story No: 40
Task: Include accessibility features for differently-abled users.
Priority: High
Value Statement: As a customer with accessibility needs, I want to use an inclusive interface so that I can shop comfortably.
BV: 100
CP: 7
Acceptance Criteria:

- Features include screen reader support, high-contrast mode, and keyboard navigation.

4. Make an ERD of creating a Ticketing life cycle.