**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**

**ASSIGNMENT 1:**

**BUSINESS REQUIREMENT DOCUMENT (BRD)**

**Project Name:** Inventory and Delivery Management System for Ice Cream & Milk Products

**Business Analyst:** NACHIKET HEMLANI

**Company Overview:** The client is a manufacturer of ice cream and milk products with multiple manufacturing plants and warehouses across the country. As the Business Analyst for this project, I have identified the need to develop software that will help manage inventory efficiently and ensure the quickest delivery to customers.

## ****Business Objectives****

* Optimize inventory management to minimize wastage and ensure product availability.
* Enhance the supply chain to facilitate the quickest possible delivery.
* Automate order processing and tracking.
* Improve real-time visibility into stock levels across different locations.
* Integrate with logistics partners for efficient route planning and dispatch.

## ****Project Scope****

### ****In-Scope****

* Centralized inventory management system.
* Order processing and tracking.
* Automated stock replenishment.
* Warehouse and manufacturing plant integration.
* Logistics and route optimization.
* Role-based access control for different users (Admin, Warehouse Manager, Logistics Team, Customer Support, etc.).
* Notifications and alerts for low stock, order status updates, and delivery tracking.

### ****Out-of-Scope****

* Manufacturing process automation.
* Physical warehouse infrastructure changes.
* Hardware procurement (barcode scanners, etc.).

## ****Key Features & Functional Requirements****

### ****Inventory Management****

* Track stock levels in real time.
* Forecast demand based on sales trends.
* Automated restocking notifications.
* Categorization of products by type, batch, and expiration date.

### ****Order Management****

* Online order placement for distributors and retailers.
* Order status tracking (Received, Processing, Dispatched, and Delivered).
* Integration with billing and invoicing systems.

### ****Delivery Optimization****

* Route planning for fastest delivery.
* GPS tracking and live order updates.
* Auto-assignment of delivery partners.
* Integration with third-party logistics (if applicable).

### ****User Management****

* Role-based access control.
* Audit logs for system transactions.

### ****Reports & Analytics****

* Inventory movement reports.
* Order fulfilment performance.
* Delivery time analysis.

## ****Development & Resource Plan****

### ****Development Phases****

1. **Requirement Gathering & Analysis (Week 1-2)** – Conduct stakeholder interviews, define business requirements, and finalize scope.
2. **System Design & Architecture (Week 3-4)** – Develop system workflows, UI/UX wireframes, and database design.
3. **Prototype Development (Week 5-6)** – Build a clickable prototype for client validation.
4. **Core Module Development (Week 7-12)**
   * Inventory Management
   * Order Processing
   * Delivery Optimization
5. **Testing (Week 13-15)** – Conduct unit testing, system integration testing, and user acceptance testing (UAT).
6. **Deployment & Training (Week 16-18)** – Deploy the system in a production environment and conduct user training sessions.

### ****Resource Allocation****

* **Business Analyst (1) – NACHIKET HEMLANI** – Requirement gathering, documentation, stakeholder communication.
* **Project Manager (1)** – Overall project planning, risk management, coordination.
* **UI/UX Designer (1)** – Design user-friendly interfaces.
* **Developers (3-4)** – Backend and frontend development.
* **QA Engineers (2)** – Functional, performance, and security testing.
* **DevOps Engineer (1)** – Deployment.
* **Support Team (2)** – Post-deployment assistance and maintenance.

## ****Risks & Mitigation Strategies****

|  |  |
| --- | --- |
| **Risk** | **Mitigation Strategy** |
| Data inconsistency | Implement real-time synchronization across locations |
| Delivery delays | AI-based route optimization & tracking |
| System downtime | Cloud-based architecture with failover backup |
| User adoption challenges | Training sessions & user-friendly interface |

#### ****Assumptions****

1. The client will provide accurate and up-to-date inventory and order data for system implementation.
2. All warehouses and manufacturing plants will have internet connectivity to enable real-time data synchronization.
3. The system will integrate with existing ERP and third-party logistics systems as required.
4. Users will undergo training to ensure smooth adoption of the new system.
5. The software will be cloud-based, ensuring high availability and scalability.
6. The project will follow Agile methodology, allowing for iterative development and feedback.
7. The budget and resources required for development, testing, and deployment will be approved and allocated.
8. The system will support multiple user roles, each with specific permissions.
9. Mobile and web applications will be developed to support inventory tracking and order processing.
10. External logistics partners will have API-based integration for order fulfilment and tracking.

#### ****Constraints****

1. The system must comply with industry regulations related to food safety, storage, and distribution.
2. Any third-party API integration (logistics, payment gateway, etc.) will depend on their availability and technical limitations.
3. Data security and compliance with relevant laws (such as GDPR for personal data) must be ensured.
4. The software should be accessible via web browsers and mobile devices with minimal performance degradation.
5. The budget and timeline are fixed, requiring efficient resource utilization.
6. The software should be scalable to handle increased demand during peak seasons.
7. The system should support multiple warehouses and distribution centres with location-based tracking.
8. Downtime for system maintenance should be minimal and planned in advance.
9. User authentication and role-based access must be implemented to prevent unauthorized access.
10. The company’s existing IT infrastructure should support the new system without major upgrades.

**PROCESS FLOW DIAGRAM**



**Assignment 2:**

**Subject:** Introduction as Business Analyst for Project Collaboration

Dear ABC LTD,

I hope this email finds you well. My name is NACHIKET HEMLANI, and I am pleased to introduce myself as the Business Analyst assigned to collaborate with you and your team on this project. I am excited about the opportunity to work closely with you in understanding your business requirements and ensuring the successful development of a solution tailored to your needs.

As a Business Analyst, my primary role is to bridge the gap between your business vision and the technical development team. I will be responsible for gathering and analysing requirements, identifying key challenges, and ensuring that the final product aligns with your expectations. Throughout this process, I will facilitate discussions, document specifications, and provide insights to enhance the project's efficiency.

To initiate our engagement, I would love to schedule a meeting at your convenience. This will allow us to discuss your business goals, pain points, and expectations in detail. Your input will be invaluable in shaping the project's direction and ensuring we deliver a solution that meets your needs effectively.

Please let me know your availability for an introductory call, or feel free to share any initial thoughts or documents that would help us get started. Looking forward to collaborating with you on this exciting journey.

**Best Regards,**   
NACHIKET HEMLANI  
Business Analyst   
[+91 0000000000]   
[APT IT SOLUTIONS]

**1. BUSINESS REQUIREMENT DOCUMENT (BRD)**

**Project Overview**

The Online Ticketing System is designed to provide a seamless ticket booking experience for users across various events, concerts, and travel services. The system will allow users to browse available tickets, book seats, and receive e-tickets instantly.

**Business Objectives**

* Enable users to search, book, and manage tickets online.
* Provide real-time availability and pricing updates.
* Ensure a secure and smooth payment process.
* Allow event organizers and transport providers to manage ticket sales.
* Generate reports for sales analysis and customer trends.

**Scope**

**In-Scope:**

* User registration and login.
* Event and transport ticket booking.
* Payment gateway integration.
* QR code-based e-ticket generation.
* Admin panel for managing ticket inventory.
* Reports and analytics module.

**Out-of-Scope:**

* Physical ticket printing and delivery.
* Third-party vendor management beyond ticket sales.

**Stakeholders**

* Customers (end-users booking tickets)
* Event Organizers and Transport Providers
* System Administrators
* Payment Gateway Providers

### ****Risks & Mitigation****

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk Category** | **Risk** | **Likelihood** | **Impact** |
| Performance & Scalability | High traffic load | High | High |
| Security | Fraudulent transactions | Medium | High |
| Security | Data breaches | Medium | High |
| Availability | Server downtime | Medium | High |
| Scalability | Scalability issues | Medium | High |
| Security | Ticket duplication | Medium | High |
| Financial | Refund and cancellation disputes | High | Medium |
| Compliance | Compliance with local laws | Medium | High |
| Usability | Poor user experience | High | Medium |
| Operational | Payment failures | High | High |

#### ****Assumptions:****

* Users have stable internet connectivity for booking and payments.
* Payment gateways will be third-party services like PayPal, Stripe, or Razor pay.
* Ticket inventory updates will be handled in real-time through API integrations.
* The system will support multiple currencies for international users.

#### ****Constraints:****

* Compliance with regional taxation and financial regulations.
* The system should operate within the defined performance benchmarks (e.g., page load = 2 sec).
* Payment processing must comply with PCI-DSS standards.
* Initial deployment will support web-based applications, with mobile apps introduced later.

## ****2. SOLUTION REQUIREMENT SPECIFICATION (SRS)****

### ****Functional Requirements****

|  |  |  |  |
| --- | --- | --- | --- |
| **Project name** | Online Ticketing | **Customer name** | ABC LTD |
| **Project Version** | 1.001 | **Project Sponsor** | James Smith |
| **Project Manager** | Avinash Singh | **Project Initiation date** | 25-01-2025 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement Name** | **Req Description** | **Priority** |
| FR-01 | User Registration | Users should be able to register and log in securely. | High |
| FR-02 | Search Functionality | Users can search for tickets using filters (event type, date, location). | High |
| FR-03 | Seat Selection | Users can select seats for applicable events before booking. | Medium |
| FR-04 | Real-time Availability | The system should update ticket availability in real time. | High |
| FR-05 | Secure Payments | The system must integrate with payment gateways for secure transactions. | High |
| FR-06 | E-Ticket Generation | The system should generate QR-based e-tickets. | High |
| FR-07 | Notifications | Users should receive email and SMS confirmations after booking. | Medium |
| FR-08 | Admin Panel | Admins should be able to manage events and ticket inventory. | High |
| FR-09 | Refund and Cancellation | Users should be able to request refunds or cancel bookings. | Medium |
| FR-10 | Sales Reports | Admins should be able to generate sales and customer trend reports. | Medium |

### ****Non-Functional Requirements****

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement Name** | **Req Description** | **Priority** |
| NFR-01 | System Availability | The system should ensure 99.9% uptime. | High |
| NFR-02 | Performance | The system should handle up to 10,000 concurrent users. | High |
| NFR-03 | Security | The system must use SSL encryption and secure authentication. | High |
| NFR-04 | Scalability | The system should be able to scale based on demand. | Medium |
| NFR-05 | Response Time | Page load time should be less than 2 seconds. | High |
| NFR-06 | Backup & Recovery | Data should be backed up daily with a recovery option. | High |
| NFR-07 | Compliance | The system must adhere to legal and financial regulations. | High |
| NFR-08 | Multi-Device Compatibility | The system should work on mobile, tablet, and desktop. | Medium |
| NFR-09 | Logging & Monitoring | The system should log all transactions and provide monitoring alerts. | Medium |
| NFR-10 | Usability | The system should be user-friendly with an intuitive UI. | High |

**ERD FOR A SUPPORT TICKET/TICKETING LIFE CYCLE:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **USERS** |  | **Support\_Tickets** |  | **Status** |  | **Attachments** |
| user\_id (PK) |  | ticket\_id (PK) |  | status\_id (PK) |  | attachment\_id (PK) |
| name |  | user\_id (FK) |  | status\_name |  | ticket\_id (FK) |
| email |  | agent\_id (FK) |  | created\_at |  | file\_path |
| phone |  | status\_id (FK) |  | modified\_at |  | file\_type |
| created\_at |  | created\_at |  | deleted\_at |  | created\_at |
| modified\_at |  | updated\_at |  |  |  | modified\_at |
| deleted\_at |  | subject |  | **Responses** |  | deleted\_at |
|  |  | description |  | response\_id (PK) |  |  |
|  |  | created\_at |  | ticket\_id (FK) |  |  |
|  |  | modified\_at |  | user\_id (FK) |  |  |
|  |  | deleted\_at |  | agent\_id (FK) |  |  |
|  |  |  |  | response\_time |  |  |
|  |  | **Agents** |  | message |  |  |
|  |  | agent\_id (PK) |  | created\_at |  |  |
|  |  | name |  | modified\_at |  |  |
|  |  | email |  | deleted\_at |  |  |
|  |  | phone |  |  |  |  |
|  |  | created\_at |  |  |  |  |
|  |  | modified\_at |  |  |  |  |
|  |  | deleted\_at |  |  |  |  |

**USER STORIES:**

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 1:** | Tasks: 1 | | Priority: HIGH |
| **As a user**,  I want to create a support ticket  So that I can report an issue to the support team. | | | |
| BV: 500 | | CP: 03 | |
| **ACCEPTANCE CRITERIA**  User can input issue details, category, and priority while creating a ticket.  A unique ticket ID is generated upon submission.  The ticket is stored securely in the database. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 2:** | Tasks: 1 | | Priority: HIGH |
| **As a user**,  I want to track the status of my support ticket  So that I know its resolution progress. | | | |
| BV: 500 | | CP: 05 | |
| **ACCEPTANCE CRITERIA**  User can view the current status of their submitted ticket.  Status updates are reflected in real time.  Ticket resolution progress is visible on the dashboard. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 3:** | Tasks: 2 | | Priority: MEDIUM |
| **As a user**,  I want to receive a confirmation email after submitting a support ticket  So that I know it was successfully created. | | | |
| BV: 200 | | CP: 03 | |
| **ACCEPTANCE CRITERIA**  User receives an email with ticket details after submission.  Email contains the ticket ID, issue summary, and support contact information.  The email is sent within 1 minute of ticket creation. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 4:** | Tasks: 2 | | Priority: HIGH |
| **As a user**,  I want to view all open support tickets  So that I can prioritize and manage them efficiently. | | | |
| BV: 500 | | CP: 05 | |
| **ACCEPTANCE CRITERIA**  Support agents can access a list of open tickets.  Tickets can be filtered by priority, issue type, and creation date. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 5:** | Tasks: 2 | | Priority: MEDIUM |
| **As a user**,  I want to assign tickets to specific team members  So that they can work on resolving them. | | | |
| BV: 100 | | CP: 01 | |
| **ACCEPTANCE CRITERIA**  Support agents can assign or reassign tickets to team members.  Assigned agents receive a notification. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 6:** | Tasks: 2 | | Priority: HIGH |
| **As a user**,  I want to attach files to my support ticket  So that I can provide more context about the issue. | | | |
| BV: 500 | | CP: 01 | |
| **ACCEPTANCE CRITERIA**  Users can upload files while submitting or updating a ticket.  Accepted formats include images, PDFs, and text files (max 10MB). | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 7:** | Tasks: 1 | | Priority: HIGH |
| **As a user**,  I want to receive notifications when my ticket status changes  So that I stay updated on progress. | | | |
| BV: 200 | | CP: 03 | |
| **ACCEPTANCE CRITERIA**  Users receive an email/SMS when their ticket is updated.  Notifications include status changes (e.g., "In Progress," "Resolved"). | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 8:** | Tasks: 3 | | Priority: HIGH |
| **As a user**,  I want to comment on my support ticket  So that I can provide additional information or clarification. | | | |
| BV: 500 | | CP: 01 | |
| **ACCEPTANCE CRITERIA**  Users can add comments to open tickets.  Support agents are notified when a comment is added. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 9:** | Tasks: 1 | | Priority: MEDIUM |
| **As a user**,  I want to add internal notes to a ticket  So that I can document troubleshooting steps without the user seeing them. | | | |
| BV: 200 | | CP: 03 | |
| **ACCEPTANCE CRITERIA**  Support agents can add internal notes to tickets.  Notes are visible only to the support team. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **USER STORY 10:** | Tasks: 3 | | Priority: MEDIUM |
| **As a user**,  I want to view the response from the support team  So that I can understand how to resolve my issue. | | | |
| BV: 200 | | CP: 05 | |
| **ACCEPTANCE CRITERIA**  Users receive a notification when a response is added.  Responses are visible in the ticket details. | | | |