**Waterfall Project Part 2**

**Submitted by: Bhavana Padol**

**Batch: 10th Aug 2024**

**Railway Reservation System**

Railway Reservation System is a system used for booking tickets over internet. Any Customer Can book tickets for different trains. Customer can book a ticket only if the tickets are available. Customer searches for the availability of tickets then if the tickets are available, he books the tickets by initially filling details in a form. Tickets can be booked in two ways by i-ticket or by e-ticket booking.

In case of i-ticket booking customer can book the tickets online and the tickets are couriered to Particular customer at their address. But in case of e-ticket booking and cancelling tickets are booked and cancelled online sitting at the home and customer himself has to take print of the ticket but in both the cases amount for tickets are deducted from customer account.

For cancellation of i-ticket the customer has to go at reservation office then fill cancellation form and request the clerk to cancel the ticket. Then the refund is transferred to customer account.

Inclusion –

Clerk can be considered as agent in this scenario for cancellation of I tickets & e-tickets. Refund request is initiates by system automatically after ticket cancelation

**Document 6-** Please prepare a use case diagram, activity diagram and a use case specification document.

**Document 7-** Screens and pages

**Document 8-** Tools-Visio and Axure

**Document 9-** BA experience

**My experience as BA in following phases:**

1. Requirement gathering:

2. Requirement Analysis:

3. Design:

4. Development:

5. Testing:

6. Deployment:

**Document 6-** Please prepare a use case diagram, activity diagram and a use case specification

document.

**Answer:**

1. **Use case diagram:**



1. **Activity Diagram:**

* **User Registration/Login**



* **Check Ticket Availability**



* **Book Ticket**



* **Print Ticket**



* **Cancel Ticket**



* **Refund Money**



1. **Use case Specification Document:**

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US001 | | |
| Use case Name | User Registration/Login | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer | | |
| Description | The user can register or log into the system by providing valid credentials such as Name, Email ID, Phone Number, and Password. | | |
| Pre-condition | * The system must be online and accessible. * User must have a valid email ID and phone number. | | |
| Post- condition | * The user is successfully registered or logged in. * The user can access the system’s functionalities. | | |
| Basic flow | * The user opens the system’s login/registration page. * The user provides their credentials (Name, Email ID, Phone Number, and Password). * The system validates the credentials. * The user is logged into the system. | | |
| Alternative flow | * If the user forgets the password, the system allows resetting it via email or phone verification. | | |
| Exceptions | * If credentials are invalid, the system displays an error message. * If the email or phone number is already registered, the system prompts the user. | | |
| Frequency of use | High | | |
| Assumptions | * The user has access to a stable internet connection. * The system’s authentication process is operational. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US002 | | |
| Use case Name | Check Ticket Availability | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer, Clerk | | |
| Description | The user can check the availability of tickets for a specific destination, date, and time. | | |
| Pre-condition | * The system is online and accessible. * Travel details (Destination, Date, Time) are available. | | |
| Post- condition | * The user receives information about ticket availability. | | |
| Basic flow | 1. The user logs into the system. 2. The user provides travel details (Destination, Date, Time). 3. The system checks ticket availability. 4. The system displays the availability status to the user. | | |
| Alternative flow | If no tickets are available, the system suggests alternative dates or times. | | |
| Exceptions | If the system is down, ticket availability cannot be checked. | | |
| Frequency of use | High | | |
| Assumptions | Travel details are accurate and match the system database. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US003 | | |
| Use case Name | Book Ticket | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer, Clerk | | |
| Description | The user can book a ticket by selecting travel details and completing the payment process. | | |
| Pre-condition | * The user must be logged in. * Tickets must be available for the desired travel details. | | |
| Post- condition | The ticket is successfully booked, and a confirmation is sent to the user. | | |
| Basic flow | 1. The user logs into the system. 2. The user selects travel details and checks ticket availability. 3. The user confirms the booking and proceeds to payment. 4. The system processes the payment. 5. The system generates and sends the ticket confirmation. | | |
| Alternative flow | If the payment fails, the user is prompted to retry or choose another payment method. | | |
| Exceptions | If the system is down, the booking cannot be completed. | | |
| Frequency of use | High | | |
| Assumptions | Payment gateways are operational. | | |

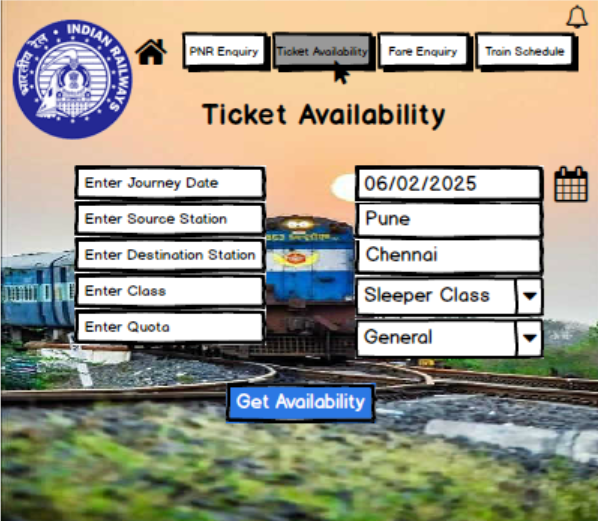
|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US004 | | |
| Use case Name | Print Ticket | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer, Clerk | | |
| Description | The user can print a booked ticket for offline use. | | |
| Pre-condition | The user must have a booked ticket. | | |
| Post- condition | * The ticket is printed successfully. | | |
| Basic flow | 1. The user logs into the system. 2. The user navigates to the ticket history section. 3. The user selects a booked ticket and clicks on the print option. 4. The system generates a printable format of the ticket. 5. The user prints the ticket. | | |
| Alternative flow | If the user cannot print the ticket, they can download it for later use. | | |
| Exceptions | * If the printer is unavailable, printing cannot be completed. | | |
| Frequency of use | Medium | | |
| Assumptions | * The user has access to a printer. * The booked ticket is stored in the system. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US005 | | |
| Use case Name | Cancel Ticket | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer, Clerk | | |
| Description | The user can cancel a previously booked ticket and receive a refund. | | |
| Pre-condition | * The user must have a valid ticket. * The ticket must be eligible for cancellation based on the system’s policy. | | |
| Post- condition | The ticket is cancelled successfully, and the refund is processed. | | |
| Basic flow | 1. The user logs into the system. 2. The user navigates to the ticket history section. 3. The user selects a ticket and clicks on the cancel option. 4. The system processes the cancellation and confirms it. 5. The system initiates a refund as per policy. | | |
| Alternative flow | If the ticket is non-refundable, the system informs the user. | | |
| Exceptions | If the system is down, the ticket cannot be cancelled. | | |
| Frequency of use | Medium | | |
| Assumptions | * The user is aware of the cancellation policy. * Refund mechanisms are operational. | | |

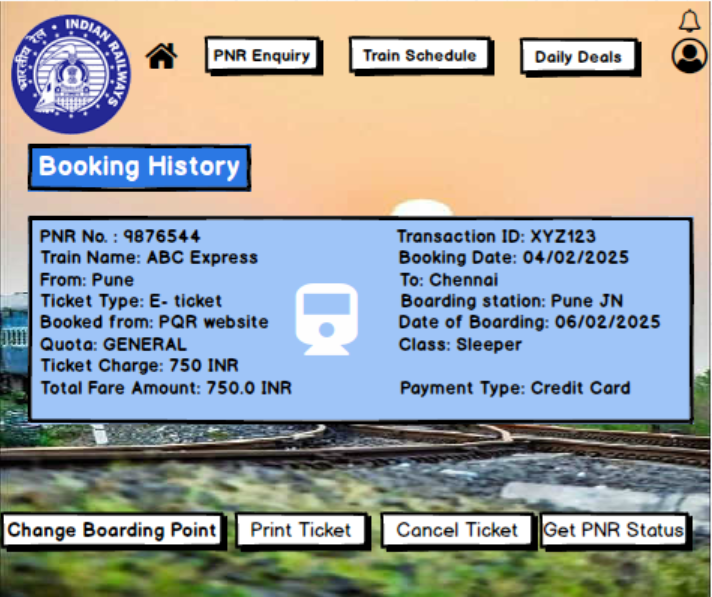
|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | US006 | | |
| Use case Name | Refund Money | | |
| Created by | Ms. Bhavana | Last Updated |  |
| Date created |  | Last Revision Date |  |
| Actor | Customer, Clerk | | |
| Description | The system processes the refund after a ticket cancellation, as per the policy. | | |
| Pre-condition | * The ticket cancellation process must be completed. * Refund policy must allow for the ticket to be refunded. | | |
| Post- condition | The refund is successfully processed, and the user is notified. | | |
| Basic flow | 1. The user cancels a ticket. 2. The system validates the refund eligibility. 3. The system processes the refund. 4. The refund amount is transferred to the user’s account. 5. The user receives a confirmation of the refund. | | |
| Alternative flow | If the refund process fails, the system notifies the user and retries or provides escalation steps. | | |
| Exceptions | If the payment gateway is down, the refund cannot be processed. | | |
| Frequency of use | Medium | | |
| Assumptions | Refund policies and payment gateways are operational. | | |

**Document 7-** Screens and pages

****

****

****

****

**Document 8-** Tools-Visio and Axure

**Document 9-** BA experience

**My experience as BA in following phases:**

1. Requirement gathering:

* **Role & Activities**: As a Business Analyst, my primary role during the requirement gathering phase was to engage with key stakeholders to understand their needs and business goals. I conducted interviews, workshops, and surveys to capture detailed requirements, facilitated discussions to align on objectives, and clarified expectations with end-users.
* **Challenges**: A major challenge was managing conflicting priorities among stakeholders. I addressed this by conducting prioritization sessions and using a MoSCoW (Must have, Should have, Could have, Won't have) technique to reach a consensus on essential requirements.
* **Outcome**: The outcome was a comprehensive set of well-documented, prioritized requirements that provided a solid foundation for the project. I also ensured stakeholder alignment and buy-in, which helped prevent misunderstandings later in the project.

2. Requirement Analysis:

* **Role & Activities**: In the analysis phase, I worked closely with both business and technical teams to analyse requirements for feasibility and alignment with business goals. I created detailed requirement specifications, user stories, and use cases, and worked on creating process flow diagrams to visualize how the solution would meet business needs.
* **Challenges**: Some requirements were initially vague or over-scoped, which created potential issues for implementation. To mitigate this, I conducted follow-up sessions with stakeholders to refine these requirements and worked with the technical team to assess feasibility and potential constraints.
* **Outcome**: I produced a detailed requirements document, including clear user stories and acceptance criteria, which enabled developers and testers to better understand the expectations. This also provided a structured reference for the rest of the project phases.

3. Design:

* **Role & Activities**: In the design phase, I collaborated with UI/UX designers to create wireframes and prototypes that translated the requirements into a visual representation of the solution. I provided feedback on design choices and ensured they aligned with user expectations and business goals.
* **Challenges**: Balancing user needs with technical constraints was a key challenge. I facilitated design review meetings between designers and developers to address any potential issues and ensure designs were technically feasible without compromising user experience.
* **Outcome**: The designs created were user-friendly and met the requirements outlined by stakeholders. These prototypes were presented to stakeholders for feedback, which helped in gaining early approval and reducing potential design-related changes during development.

4. Development:

* **Role & Activities**: During the development phase, I acted as the bridge between developers and stakeholders. I clarified requirements, addressed any questions from developers, and ensured alignment with the initial goals. I participated in sprint planning and review meetings, helping prioritize development tasks and resolve issues as they arose.
* **Challenges**: Occasionally, developers encountered challenges in implementing certain functionalities as specified in the requirements. To address this, I facilitated discussions to explore alternative solutions that could satisfy both business needs and technical limitations.
* **Outcome**: By maintaining close communication with the development team, I ensured that the functionality developed aligned closely with business objectives. This proactive collaboration helped minimize the need for extensive rework and kept the project on track.

5. Testing:

* **Role & Activities**: In the testing phase, I supported the creation of test cases and participated in User Acceptance Testing (UAT) to verify that the solution met business requirements. I reviewed test results, identified gaps, and worked with the team to ensure that all critical issues were resolved.
* **Challenges**: One challenge was ensuring that all edge cases and user scenarios were thoroughly tested. I worked closely with the QA team to develop comprehensive test cases and reviewed feedback from testers to ensure all aspects of the requirements were covered.
* **Outcome**: The testing phase was successful, with a high level of defect resolution and satisfaction from stakeholders. The rigorous UAT process ensured that the product was well-aligned with business needs and ready for deployment.

1. Deployment:

* **Role & Activities**: During deployment, I coordinated with stakeholders to prepare for go-live, which included creating user guides, conducting training sessions, and ensuring that all necessary documentation was ready. I also collaborated with the technical team to ensure a smooth transition to the live environment.
* **Challenges**: Some users were unfamiliar with the new system, which created a potential risk for user adoption. To address this, I organized training workshops and provided support during the initial rollout.
* **Outcome**: The deployment was successful, with minimal issues. My efforts in preparing users and providing adequate support helped ensure a smooth transition and high user satisfaction with the new system.