Software Requirements Specification (SRS) for Online Flight Ticketing Platform

Student Name: Mutale Bwalya
Student ID: SE2023011
College: Lusaka Institute of Technology (LIT)
Date: June 20, 2025

1. Introduction – What We're Doing Here

This document is basically the blueprint, or the "what-it-needs-to-do" list, for a new online system that handles flight bookings. We're calling it the "Zambian Air Travel System" (ZATS). The main point is to build a user-friendly website where folks can easily search for flights, book their seats, and make payments online. It also has to include some tools for the airline's own team to manage flights and all the customer bookings efficiently. This whole project aims to bring flight booking in Zambia into the modern era, making it much more convenient for everyone.

2. General System Description

2.1 The System at a Glance

The ZATS will be a web application, working independently for now. But we're designing it to be ready to link up with any existing airline databases in the future, especially for real-time flight data. The big idea is to move away from old, slow booking methods and make everything faster and more accurate for both passengers and the airline.

2.2 Who Will Be Using This?

We have two main types of users for this system:

- **Passengers:** These are the regular people looking to book flights. They'll use the public side of the website to search, choose flights, make bookings, and check their reservations. We figure they'll have a basic understanding of how to use websites.
- **Airline Administrative Personnel:** This group includes staff from operations, customer service, and ticketing. They'll use a secure, restricted part of the system to handle flight schedules, assist passengers with requests, and access various reports. Their specific access will depend on their job roles.

2.3 Where It Will Operate

The ZATS will run on standard web browsers (like Chrome, Firefox, Edge, or Safari). It's crucial that it adapts and looks good on all devices – from large desktop screens to smaller mobile phones. We’ll host it on a reliable cloud service to ensure it’s always available and performs well.

2.4 Strict Requirements (Constraints)

- **Top-Tier Security:** This is non-negotiable. Since we’re dealing with sensitive personal details and payment information, the system absolutely *must* be highly secure. This means all data, whether it’s being sent or stored, needs to be encrypted. We need strong defenses against any hacking attempts or data breaches.
- **Legal Compliance:** We are obligated to adhere to all relevant aviation laws in Zambia and international data privacy regulations concerning passenger information. There's no compromise on this.
- **Payment Gateway Standards:** The system can only integrate with payment gateways that meet strict security certifications (like PCI DSS compliance).

2.5 What We’re Relying On (Assumptions)

We are making a few key assumptions: that most users will have a stable internet connection; that the payment services we choose will provide robust and reliable API connections; and, fundamentally, that a strong, scalable database system will be available for storing all our flight and user data.

3. Specific Features Required

3.1 User Account Setup & Access

- **F.1:** New users must be able to register for an account. They will provide a unique email address, a password that meets predefined strength criteria, and their essential contact information.
- **F.2:** Existing registered users shall be able to log into their accounts securely using their credentials.
- **F.3:** Once logged in, users must have the capability to view and update their personal profile details.
- **F.4:** A secure mechanism shall be in place for users to reset their forgotten passwords via their registered email address.

3.2 Flight Search & Display

- **F.5:** Users shall be able to conduct flight searches by specifying: departure airport, destination airport, desired travel dates (supporting both one-way and round-trip options), and the number of passengers (categorized, e.g., adult, child, infant).
- **F.6:** The system's search results must clearly present all pertinent flight

information, including: the airline's name, specific flight number, exact departure and arrival times, total flight duration, details regarding any layovers, and the corresponding ticket price.

- **F.7:** Users shall be provided with filtering options for search results, allowing them to refine choices based on criteria such as price range, specific airlines, number of stops, and preferred time slots for departure or arrival.

3.3 The Booking & Reservation Process

- **F.8:** Users must be able to select their preferred flight(s) from the search results to initiate the booking procedure.
- **F.9:** The system shall guide the user through entering all necessary passenger details for each traveler included in the booking (e.g., full legal name, date of birth, and passport information for international travel, if applicable).
- **F.10:** Prior to final payment, a comprehensive summary of the selected flights, entered passenger details, and the total cost shall be displayed for the user’s review and explicit confirmation.
- **F.11:** Upon successful reservation, the system shall generate and prominently display a unique booking reference number.

3.4 Payment Handling

- **F.12:** The system is required to securely integrate with at least one major online payment gateway for processing credit/debit card transactions (e.g., Visa/MasterCard). The feasibility of integrating popular local Zambian mobile money options will also be thoroughly explored.
- **F.13:** The booking status shall automatically transition to "Confirmed" immediately following successful payment processing.
- **F.14:** An electronic ticket (e-ticket) and a detailed booking confirmation email shall be automatically dispatched to the user's registered email address upon successful payment.

3.5 Administrative Tasks for Airline Staff

- **F.15:** Authorized administrative staff shall be capable of managing flight schedules, which includes adding new flights, modifying details of existing flights (e.g., timings, aircraft assignments, seat capacities), and cancelling flights.
- **F.16:** Administrators shall also possess the ability to search for, view details of, modify, or cancel passenger bookings as required.
- **F.17:** The system must generate various reports essential for administrative analysis, such as flight occupancy reports, revenue summaries, and passenger manifests for specific flights.

4. External Interface Requirements

4.1 User Interface (What People See)

The ZATS will feature an intuitive, web-based graphical user interface (GUI). Its design will be fully responsive, ensuring an optimal and consistent user experience across desktops, tablets, and mobile phones, with clear navigation and visual cues.

4.2 Software Interfaces (Talking to Other Systems)

- **Payment Gateway APIs:** These are critical for handling all secure financial transactions.
- **Email Service API:** For automated sending of transactional emails like booking confirmations, e-tickets, and password reset links.
- **Airline Internal Systems API (Future Consideration):** Potential future integration with the airline's own core operational databases for real-time inventory and flight data synchronization.

4.3 Communication Interfaces (Secure Connections)

All data communication between the client (user’s browser) and our servers, especially sensitive personal or payment information, *must* be secured using industry-standard HTTPS protocols (TLS 1.2 or higher).

5. Non-Functional Requirements (The "How Well" Factors)

5.1 Performance – How Quick It Is

- **NFR.P1 (Search Response Time):** Flight search results should consistently display within 2.9 seconds for 90% of requests, even during periods of high system load.
- **NFR.P2 (User Concurrency):** The system shall be able to support up to 68 concurrent users actively performing searches and booking transactions without experiencing any significant slowdown.

5.2 Security – Keeping It Safe and Sound

- **NFR.S1 (Data Encryption):** All sensitive data, including personally identifiable information (PII) and payment details, must be encrypted both during transmission (using robust TLS encryption) and when stored within the database.
- **NFR.S2 (Access Control):** A strict role-based access control (RBAC) system shall be implemented, ensuring that administrative staff can only access functionalities commensurate with their assigned roles.
- **NFR.S3 (Authentication Strength):** User authentication mechanisms will enforce strong password policies, implement account lockout after multiple failed

attempts, and include robust measures against brute-force attacks.

- **NFR.S4 (Session Security):** Robust session management techniques shall be employed to prevent session hijacking and unauthorized access to user accounts.

5.3 Reliability – Always There When You Need It

- **NFR.R1 (System Availability):** The ZATS shall maintain an operational uptime of 99.75%, excluding pre-scheduled maintenance windows, ensuring very high availability of service.
- **NFR.R2 (Data Integrity & Recovery):** Comprehensive mechanisms for data integrity (e.g., transactional consistency) and a robust backup and disaster recovery plan shall be in place to prevent data loss for all booking and payment records.

5.4 Usability – Easy to Use and Understand

- **NFR.U1 (Learning Curve):** A new user, even with only basic web navigation skills, should be able to successfully complete a flight booking (from initial search to final confirmation) within a maximum of 6 intuitive steps.
- **NFR.U2 (Error Feedback):** The system shall provide clear, concise, and helpful error messages for any invalid inputs or system issues, guiding the user toward a quick resolution.

5.5 Scalability – Ready for the Future

The architectural design of the ZATS shall explicitly support horizontal scaling, enabling it to accommodate substantial future growth in user base, flight inventory, and transaction volume without requiring a major re-architecture.

5.6 Maintainability – Easy to Manage and Update

- **NFR.M1 (Code Quality & Documentation):** The codebase shall adhere to high coding standards, be extensively commented, and follow a modular design pattern to facilitate ease of understanding, debugging, and future modifications by development teams.
- **NFR.M2 (Deployment Efficiency):** Updates, bug fixes, and new feature deployments shall be achievable with minimal system downtime, ideally leveraging automated Continuous Integration/Continuous Deployment (CI/CD) pipelines.