Software Requirements Specification

for

Airline Ticket Reservation

Version 1.0 approved

Prepared by Mustafa Zahid(20k-1045) and Munir Abbasi(20k-0128)
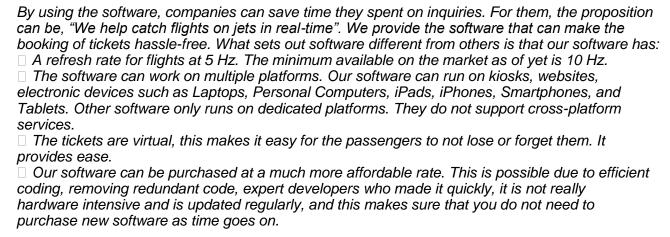
23rd December 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

By using the software, companies can save time they spent on inquiries. For them, the proposition can be, "We help catch flights on jets in real-time". We provide the software that can make the booking of tickets hassle-free. What sets out software different from others is that our software has:
☐ A refresh rate for flights at 5 Hz. The minimum available on the market as of yet is 10 Hz.
☐ The software can work on multiple platforms. Our software can run on kiosks, websites, electronic devices such as Laptops, Personal Computers, iPads, iPhones, Smartphones, and Tablets. Other software only runs on dedicated platforms. They do not support cross-platform services.
☐ The tickets are virtual, this makes it easy for the passengers to not lose or forget them. It provides ease.
☐ Our software can be purchased at a much more affordable rate. This is possible due to efficient coding, removing redundant code, expert developers who made it quickly, it is not really hardware intensive and is updated regularly, and this makes sure that you do not need to purchase new software as time goes on.

We offer the cheapest tickets as compared to others. This will help our clients attract their customers
and increase the return on investment. We give customers the advantage of canceling their bookings
and even rescheduling them in the same application within 24 hours. This attracts customers and always
leaves room for emergency flights that may occur at the last instant. Another special offer we provide
for our users is that our software can also handle payment processing. We provide buy now pay later
services to clients. This means flights can be booked on credit and paid later on. This makes the software
owner have a payment processing not readily present in other software. The personal information of
the details for the customer is encrypted and the company has full responsibility for protecting customer details. We accept responsibility as the company has made use of Secure Socket layer

connections while following transport layered protocols. By using digital certificates and asymmetric keys, we bolster the security of the data. Multiple hardware servers store the data and make the data
safe.

1.2 Document Conventions

The document is formatted according to IEEE standard

1.3 Intended Audience and Reading Suggestions

The intended audience for this document consists of requirements engineers, software developers, software designers, testers and project manager.

1.4 Product Scope

Subject facet: The User Interface (UI), When looking for one-way flights, When looking for round-trip flights, When looking for numerous destinations, Reservations for flights, Cancellation of reservations, Payment over the internet, Request for cancellation of a reservation and response Warning notifications are displayed.

IT Facet: Database, Web-based software system, AAS for logins, Performance maintenance.

Development Facet: The airline company's internal policy and culture should be taken into account

Usage facet: Flight searching and sorting Ticket reservations, Managing an existing reservation, flight details, etc. Keeping track of the flights.
.

1.5 References

1.www.wikipedia.com
2. Luke Paireepinart, David Keyes, Jingtao Liu, Frank Medjo and Seth Orell, Software Requirements Specification for Airline Flight Booking System, February 2009 (http://www.academia.edu/23567842/Software_Requirements_Specification_for_Airline_Flight_ Booking_System_Software_Requirements_Specification_for_Airline_flight_booking_system)

2. Overall Description

2.1 Product Perspective

ATRS is a digital version of the airline office's traditional manual reservation system. Customers must visit the airline office to make a reservation under the current manual system. booking. Apart from the fact that not all clients have the time to visit the office, existing customers have the opportunity to do so. Long lines form at the office as a result of the system. Some clients become bored while waiting. As a result of the line, the airline loses potential customers. In addition, while making a reservation at the office, a hard copy of the passport is necessary. Customers who are unable to present their passports at the office for whatever reason will be denied service.
The new system tries to address the above-mentioned shortcomings of the old one. It is going to allow people to make bookings based on their needs from all around the world without having to travel leave their places. It will also lessen the workload of personnel who are in charge of making reservations at the office.
Customers can use the system to check flight availability for specific dates and routes, as well as learn about the lengths of available flights. Customers can also check pricing and see what is

included in the ticket, such as baggage allowance, dinner, and so on, as well as book the ticket. The administrator has the ability to edit, delete, and add new flights to the system. Furthermore, the administrator can view client cancellation requests and determine whether or not to accept or reject them.

2.2 Product Functions

Customers and administrators will have access to ten features, which are described below

2.2.1 Search for flights

Customers can use this option to search for one-way, round-trip, and multiple-destination flights by entering specified dates and destinations.

2.2.2 Sort flights.

In terms of these features, customers can sort flights by price or flight duration.

2.2.3 Specify passengers

This feature allows the user to specify the number of passengers and their type (adult, child, or infant).

2.2.4 Book flights

Customers can use this feature to book flights by selecting ticket kinds and making an online payment.

2.2.5 Add new flights

The function grants administrator the privilege of adding new flights to the system

2.2.6 Request cancellation

This function shows that the customer can request that the reserved ticket be cancelled.

2.2.7 Remove flight

This feature allows the administrator to delete flights from the system that have been cancelled for whatever reason.

2.2.8 Modify flight details

Using this functionality administrator can modify the details of the existing flights

2.2.9 See booking details

This functionality enables administrator to view the customers' booking details.

2.2.10 See cancellation requests

This feature allows administrators to see a list of client cancellation requests and accept or reject them.

2.3 User Classes and Characteristics

Administrators, consumers, and support personnel are the three types of users in the system. Administrators should be taught and knowledgeable on how to use this software. Customers, on the other hand, do not require any training or prior expertise. Specialists with good analytical and problem-solving skills, up-to-date technical knowledge, and good interpersonal and customer-service skills make up the support personnel.

2.4 Operating Environment

The designed system is envisioned as a website that will be accessible through any web browser. It will not be reliant on the user's device's technological capability or operating system.

2.5 Design and Implementation Constraints

Flight dates and times should be presented in accordance with the time zones of the departure and destination cities, as well as the daylight saving time settings for each country. Furthermore, information concerning any database changes should be displayed as soon as possible.

2.6 User Documentation

The instructions on how to book a flight will be provided on the website for novice users

2.7 Assumptions and Dependencies

It is assumed that the user has access to the internet and is capable of making online payments. The quality and speed of the internet connection affect ATRS' performance.

3. External Interface Requirements

3.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

3.2 Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

3.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

3.4 Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

4. System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

4.1 System Feature 1

<Don't really say "System Feature 1." State the feature name in just a few words.>

    4.1.1   Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

    4.1.2   Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

    4.1.3   Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use "TBD" as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:
REQ-2:

4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 System should be able to handle 1000 transactions per second.

5.2 Safety Requirements

5.2.1 System should include restore and recover functions in order to prevent data loss.
5.2.2 System should assure data integrity.

5.3 Security Requirements

5.3.1: For logins, the system should have an Authentication and Authorization System (AAS).
5.3.2: The system should allow the administrator to customize his or her system-generated password.
5.3.3: Only those who login with a predetermined administrator username and password should be granted administrative privileges.
5.3.4: System should allow administrator to login with password.
5.3.5: Access should be blocked by the system to one who fails to login three time in a row.

5.3.6: Customized password should be at least 8 characters and should contain both letters and numbers.
5.3.7: The security of the communication channel should be guaranteed by the system.

5.4 Software Quality Attributes

5.4.1 Usability
A user's non-technical background should not be a barrier to understanding and using the system..

5.4.2 Consistency
Once a transaction for the flight ticket is completed, the number of available seats for that flight should be reduced by one unit.

5.4.3 Robustness
In the event of a page refresh after a sudden connection loss, the system should be able to display the user's most recent enquiry.

5.5 Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

6. Other Requirements

6.1: System ought to show a warning message if the client input for full name label is out of outlined commonplace
6..2: If the client input for travel document number falls outside of the stated standard, the system should display a warning message..
6.3: If a customer wants to book flights for more than 6 people, the system should display a warning message.
6.4: If a customer wants to book flights for more than 6 people, the system should display a warning message

Appendix A: Glossary

IEEE: The Institute of Electrical and Electronics Engineers

HTTP: Hypertext Transfer Protocol

ATRS: Airline Ticket Reservation System

Authentication: The process of identifying an individual, usually based on username and password

Database: An organized collection of data, stored and accessed Electronically.

Standby database: Database replica created from a backup of a primary database

Server: Computer program that provides functionality to other programs such as clients

World Wide Web: Combination of all resources and users on the Internet that are using HTTP.

Web-browser: Software application to access information on World Wide Web

Authorization: The process of granting defined privileges to successfully authenticated individuals

AAS: Authentication and Authorization System

Client: Computer application, such as web-browser, that runs on a computer and connects to server as necessary

Service Level Agreement (SLA) States agreed level of availability

VIP: Very Important Person

Adult: 12+ years old individuals

Children: 0-12 years old individuals

Infant: To 2 years

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>