Software Requirement Specifications Document

For

Mytrip.com

Author: Maitri Chattopadhyay
(mach5953@colorado.edu)

1 Introduction

This section provides a brief description of the purpose of this SRS document built to elaborate the project Mytrip.com.

1.1 Purpose

The purpose of this document is to give a detailed description of the Mytrip.com web application. The document contains the work flow, system constrains, requirements and assumption made while designing the application. The intended reader of this document are the stakeholders, system engineers and managers.

1.2 Scope

Mytrip.com is a web based application that can be used by multiple users to book flights across the world. There are no advertisement displays on the site. The application is designed with security as its primary focus. It is heavily protected against attacks by providing multiple layers of security.

1.3 Acronyms and Definitions

Table 1: Terms used in the document and their meanings

Term used	**Definitions**
User	Refers to anyone accessing the Website
Customer	A user who access the website to make a booking or search for flights
Admin	A user with special administrative privileges
GUI	Graphical user interface
Flights	Name of the database for the application
SQLi	SQL injection
PHP	The application programing language
MariaDB	Database Server

1.4 References

IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.

1.5 Overview

The rest of the document provides an overall description of the application. It contains product perspective, product function, user characteristics, constraints, and assumptions. It also contains a detailed description of the specific requirements concerning the input/output and the storage of data of the web application.

2 Overall Description

This section describes how the different sub systems combine and interact with each other to produce the whole functionality of the system. The background for the requirements are also explained through functions, constrains characteristics and assumptions made while developing the system.

This section consists of five subsections listed as follows:

a) Product perspective
b) Product functions
c) User characteristics
d) Constraints
e) Assumptions and dependencies

2.1 Product Perspective

The application is a stand-alone product.

a. **Servers Used:**
Database server: 100.66.2.7
Web Server: 100.66.1.7

b. **User interfaces:** Any modern web browser can be used for user interface. Older browzers might not display the HTML features of the page appropriately, but the functionality of the application will not be affected.

Author: Maitri Chattopadhyay
(mach5953@colorado.edu)

c. **Hardware interfaces**: The system makes use of two Fedora Linux machines. One of them is a web server and the other one is a data base server.

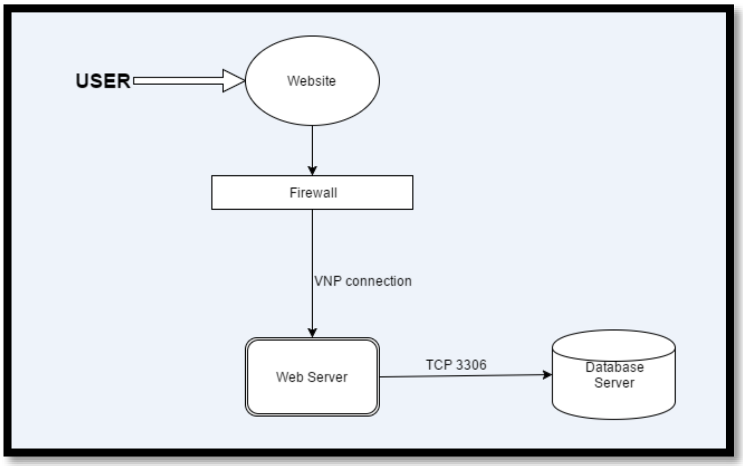

Diagram 1: Block Diagram of the major components of the system

d. **Software interfaces**: The application makes use of PHP, HTML and CSS. The database server uses MariaDB to create and query databases.

e. **Memory**: The database server acts as the memory for the application.

The user's browser connects to the web server through a secure VPN. The web server communicates with the database server via TCP 3306. The web server needs continuous connection to the database to log relevant data. The database can only be accessed by Admin. Multiple layers of protection are provided to protect the application against attacks.

2.2 Product Functions

The functionality of the application Is divided based of the type of user: Admin or Customer

Admin user:

- Can add flight schedules
- Can check the failed login logs
- Can block/unblock a user based on IP of the machine
-

Customer User:

- Can search for a flight
- Can login or create new account.
- Can select a flight and make payment for it
- Can check his order list of flight bookings.

2.3 Users Characteristics

a. **Customer:** A person who visits the website with the intensions to search for a flight and may or may not make a final booking.
b. **Admin:** A person with special privileges of modifying the database and blocking/unblocking customers for the safety of the website

2.4 Constrains

Some of the obstacles that would hinder the development of the software are the values being passed from one page to another in order to perform a whole unit functionality. Due to the presence of multiple files and more than one access point to a particular page, maintaining data authenticity is crucial.

2.5 Assumptions

The various assumptions made while designing the system are as follows:

a) The servers can only be connected through the VPN.
b) Must use fedora25 operating system.
c) There is only one Web server and only database server.
d) The website is protected enough to brace against any attack.

3 Specific Requirements

This section provides the functional and quality requirements of the system.

In the flights database, the requirements are:

a) Store the customer details.
b) Store the schedules of the flights.
c) Store the failed login attempts and the IP of the machine being used.
d) Store the order history of customers.
e) Store the IPs of the machines that have been blocked permanently by the admin.
f) Normalize all data possible. Use foreign keys for this.

In the webserver, the requirements are:

a) Has a PHP file called header.php containing the heading of the page. It is displayed on all the pages.
b) Has a PHP file called login.php containing the GUI of the page for user login.
c) Has a PHP file called register.php containing the GUI of the page for users to create new accounts.
d) Has a PHP file called index.php that contains the functionality of the website.
e) Has a PHP file called project-lib.php that contains all the isset statements and the stand-alone functions.

3.1 Input and Output

The Application has to be connected to both the Web Server as well as the Database Server for proper functioning. Data can be entered into the database through the web application or through the command line. Only the Admins will have access to the database through command line. All users can access the web application which intern inserts or queries from the database. The database executes the queries and displays the results on the web page. These results can be seen be seen by the users.

3.2 Functional Requirements

Multiple layers of protection are provided to protect the application against attacks.

3.3 Reliability

There are few external factors governing the reliability of the application. The most important is a continues internet connection. Next, is a stable VPN connection. VPN connection depends on the internet.

3.4 Availability

The web application can only be accessed when the VPN connection has been successfully established and both the database and the web server are properly functioning. If the VPN connection is disrupted in between, the application will not work until the connection is reformed. For all the connections to sustain, a stable internet connection is essential. Hence, if there is internet, VPN connection and both the servers are working, the application s available for use.

3.5 Security

Multiple layers of protection are provided to protect the application against attacks. Some of the protections uses are as follows.

- Htmlspecialchars: to sanitize the database outputs
- Prepared statements: To protect against SQL injection
- Strip tags: To sanitize the user inputs
- Situational awareness through failed login attempts
- Logout feature: To protect against brute force attack by blocking user to login for one hour after 5 simultaneous failed login attempts.
- Permanent blocking of an IP
- Checks for SQLi
- Check for integers
- Require Authentication to be logged in etc.