
Passport Visa Management System

“Application Integration Module

Functional Specification Document

Software Requirements Document

V 1.0

Table of Contents

1	Introduction	3
1.1	About this document	3
1.1.1	Purpose & Scope of the document	3
1.1.2	Intended Audience	3
1.2	About the Software System	3
1.2.1	Scope of the system	3
1.2.2	Exclusions	4
1.2.3	System Perspective	4
1.2.4	System diagram	4
1.2.5	Architecture diagram	5
1.2.6	System Environment	6
1.2.7	User Characteristics	6
1.2.8	Impact of the System	7
1.2.9	Assumptions, Risks / Constraints	7
1.2.10	Design Constraints	7
2	System Requirements	8
2.1	Functional Requirements	8
2.1.1	User Registration Module	9
2.1.2	Apply Passport Module	12
2.1.3	Apply Visa Module	16
2.1.4	Re-issue Passport Module	19
2.1.5	Visa Cancellation Module	22
	User Interface Requirements	23
2.2	Nonfunctional Requirements	25
2.2.1	UI Requirements:	25
2.2.2	Performance Requirements:	25
2.2.3	Installation, Deployment & Operational Requirements:	25
2.2.4	Maintainability & Portability Requirements:	26
2.2.5	Security Requirements :	26
3	Requirements / Use Case Summary Table	26
4	Make /Buy analysis	Error! Bookmark not defined.
4.1	Reusable components	Error! Bookmark not defined.
5	Annexure	Error! Bookmark not defined.
6	Change Log	Error! Bookmark not defined.

1 Introduction

1.1 About this document

1.1.1 Purpose & Scope of the document

The purpose of the software requirements document is to systematically capture requirements for the project and the system “Passport Visa Management System” to be developed. Both functional and non-functional requirements of this system are captured in this document. It also serves as the input for the project scoping.

The scope of this document is limited to addressing the requirements from a user, quality, and non-functional perspective. It is recommended that design aspects are not added in this document

1.1.2 Intended Audience

Project Team

1.2 About the Software System

The following section will cover aspects related to

Passport Visa Management System (PVMS) application. PVMS is a system used for registering Passport and Visa applications, Apply for Passport and Visa also for renewal of Passport, and cancelation of Visa.

The following are the modules in this proposed system

- a) User Registration (Registration type: passport /visa)
- b) Apply Passport
- c) Apply Visa
- d) Passport Renewal
- e) Visa Cancellation

1.2.1 Scope of the system

The scope of the system is explained through its modules as follows

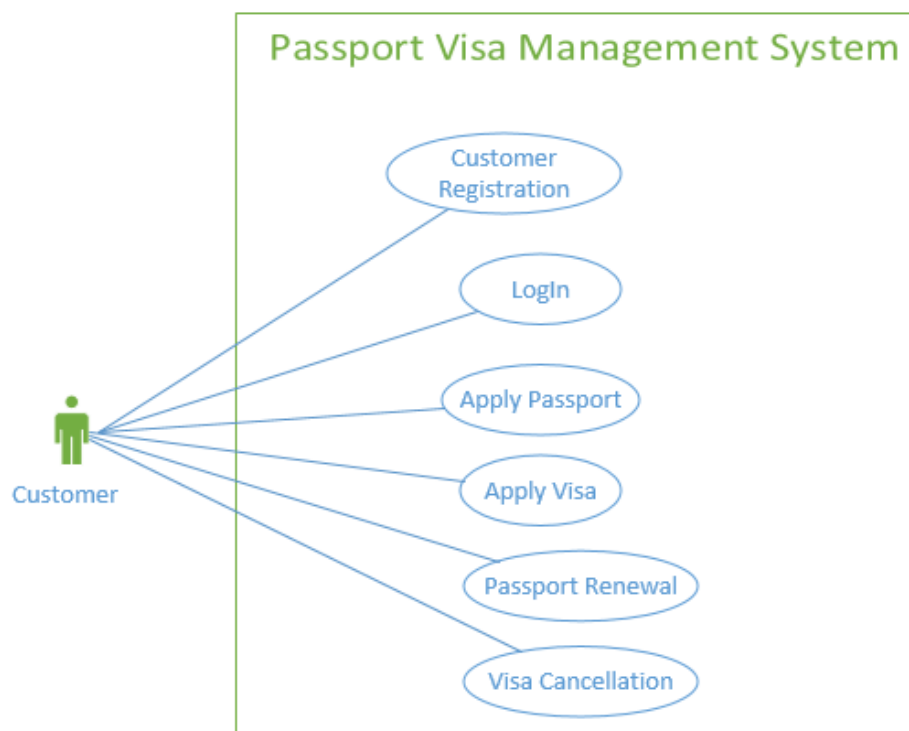
- **User Registration(Registration type: passport /visa):**This use case will be used by the user to register the service based on different parameters. The system will calculate the User id, find the Citizen type and store the details in the database.
- **Apply Passport:** This use case will be used by the user to apply the passport based on different parameters. The system should generate the passport id and expiry date. Update the details into the database.

- **Apply Visa** :This use case will be used by the user to apply visa based on different parameters.
- **Passport Re-Issue** :This use case will be used by the user to request renewal of passport.
- **Visa Cancellation**: This use case will be used by the user to perform visa cancellation. Update all the details into the system.

1.2.2 Exclusions

1. The system will operate only on the modules discussed above and will not include any additional functionality.

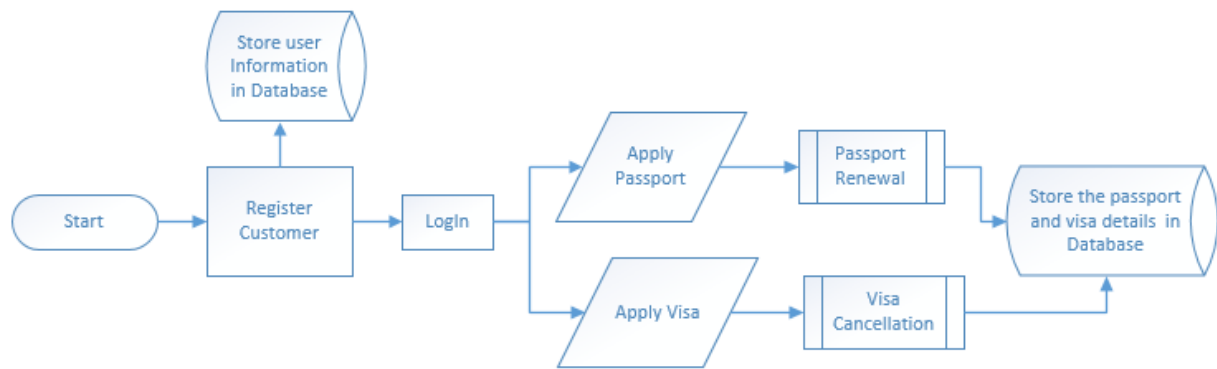
Use Case Diagram



1.2.3 System Perspective

The PVMS is an independent software system developed to manage the activities like Registering New Medicines, Ordering New Medicines, etc. using the Java/Dotnet architecture.

1.2.4 System diagram



1.2.5 Architecture diagram

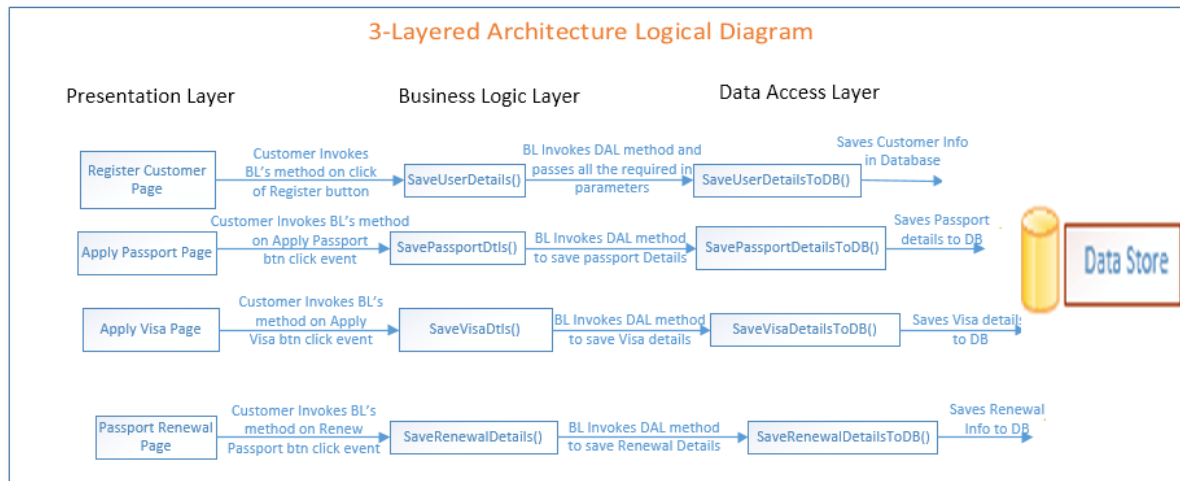
1.2.5.1 Physical Architecture:

A physical architecture is an arrangement of physical elements, (system elements and physical interfaces) that provides the designed solution for a product, service, or enterprise. It is intended to satisfy logical architecture elements and system requirements. Passport Visa Management Process follows a three layered architecture namely presentation layer, business logic layer and data access layer.

- **Presentation Tier** is the tier in which the users interact with an application. Presentation Tier contents Shared UI code, Code Behind and Designers used to represent information to user.
- **Business Tier** is mainly working as the bridge between Data Tier and Presentation Tier. All the Data passes through the Business Tier before passing to the presentation Tier. Business Tier is the sum of Business Logic Layer, Data Access Layer and Value Object and other components used to add business logic.
- **Data Tier** is basically the server which stores all the application's data. Data tier contents Database Tables, XML Files and other means of storing Application Data.

Logical Architecture:

The Logical Architecture defines the Processes (the activities and functions) that are required to provide the required User Services. Many different Processes must work together and share information to provide a User Service. The Processes can be implemented via software, hardware, or firmware. The Logical Architecture is independent of technologies and implementations.



Frameworks

- **SpringBoot Rest API** for connecting to database
- **Client App using Spring Framework/ Spring Boot**
- **Use RabbitMQ** for sending the messages

1.2.6 System Environment

The “PVMS” application will be operated from the client server with parallel processor support. When a user connects to the Web Server, the Web Server will interact with the Database after processing the business logic to transfer data to and from a database.

1. .NET Framework application deployment.
2. SQL server database for data storage

1.2.7 User Characteristics

The application should be user friendly; in such a way that the user would not need any software and hardware knowledge. The user may be new to the client products having never used them before.

The types of users likely to use the system are as follows,

1. Client
2. Delivery Assurance Group, Process Engineering group
3. Developers, testers, other associates in the project

1.2.8 Impact of the System

This is a new product which is developed for internal users. Expected impact of the product is to automate existing manual processes in order to make them more efficient and cost effective.

1.2.9 Assumptions, Risks / Constraints

Assumptions:

1. The processing admin must possess prior knowledge on **PVMS** operations undertaken offline.
2. The admin should have the details of the Medicines to be registered updated into the system.

Risks:

3. If there is any failure in stock management processes, the system should be able to handle such a financial risk.

1.2.10 Design Constraints

1. The Admin registration module and the Medicine registration modules should remain independent, without affecting **Passport Visa Management System**.
2. Application should have single login feature. Login functionality should be the Welcome Page for the application.
3. Only Main Administrator and registered Branch Administrator can access the application.
4. None of the application features can be accessed without login.

2 System Requirements

2.1 Functional Requirements

Passport Visa Management Process Req-2.1.1	Home Page
Functional Requirements	<p>When the User enter the following link http://localhost:8080/PassportManagementSystem in the browser, Home page appears which will have a short welcome message; a brief introduction to the purpose of the Passport Visa Management Process, the User can choose appropriate options.</p> <p>Under home page, we need below interface.</p>
Home page Interface Requirements	<ol style="list-style-type: none">1. Login/Signup2. About Us3. Contact Us

2.1.1 User Registration Module

2.1.1.1 User Registration Page

Passport Visa Management System Req-2.1.1.1	User Registration Page
Functional Requirements	This module will be used by the user to register the service based on different parameters. The system will calculate the User id, find the Citizen type and store the details in the database.
User Interface Requirements	<ol style="list-style-type: none">1. First name: Name of the user2. Sur name: Name of the father3. Date Of Birth: User DOB in DD/MM/YYYY format4. Address: Address of the User5. Contact no: Contact number of the Applicant6. Email Address: Email Address of the User7. Qualification: Educational qualification of the user8. Gender: Gender of the user9. Apply Type: Apply type (Passport r Visa)10. Hint Question: Hint question for recovery of password11. Hint Answer: Answer for the question <p>After successfully validation of the details, System should generate the User id, password and citizen type.</p> <p>Update all the details in the database.</p>
Trigger	User triggers the functionality once he/she submits the details into the system.
Pre-Conditions	The system should collect all the required details for User Registration.
Post Conditions	System should generate the user id and decide citizen type based on date of birth.
Success End Condition	<p>On successfully saving the details the message should be displayed in this format:</p> <p>Dear User,</p> <p>Your User Id is <userId> and your password is <password>.</p> <p>You are planning for <apply Type> and your citizen type is <citizen type>.</p>

Failed End Condition	The system should throw an error message if any one or more of the validations fail. The error messages and error codes need to be identified during design. The validation and business rules are mentioned in the business rules section.												
Steps & Actions	<ol style="list-style-type: none"> 1. System should get the all the required information from the user. 2. Identify the citizen type based on the age. Age can be calculated with the help of Date of Birth. 3. Identify the citizen type as follows <table border="1"> <thead> <tr> <th><i>Age</i></th><th><i>Citizen type</i></th></tr> </thead> <tbody> <tr> <td>0-1</td><td>infant</td></tr> <tr> <td>1-10</td><td>Children</td></tr> <tr> <td>10-20</td><td>Teen</td></tr> <tr> <td>20-50</td><td>Adult</td></tr> <tr> <td>50-till date</td><td>Senior Citizen</td></tr> </tbody> </table> 4. Generate the User id from apply type as follows. <p>If (apply type is passport) then User id is PASS-XXXX</p> <p>If (apply type is visa) then User id is VISA-XXXX.XXXX</p> <p>Fetch the last generated user id to identify the new number. But the last generated user id has to be retrieved separately from the database for VISA and PASSPORT.</p> <p>(Example: if the last value for Passport is</p> 	<i>Age</i>	<i>Citizen type</i>	0-1	infant	1-10	Children	10-20	Teen	20-50	Adult	50-till date	Senior Citizen
<i>Age</i>	<i>Citizen type</i>												
0-1	infant												
1-10	Children												
10-20	Teen												
20-50	Adult												
50-till date	Senior Citizen												

	<p>PASS-1234 in USER_REGISTRATION table, then 1234 will be taken and 1 will be added. So the next record for Passport will be PASS-1235.)</p> <p>5. Generate the password as follows.</p> <p>Password = (current date (dd))+(current month (mmm)) + (# or @ or \$) +unique random 3 digit number</p> <p>Example: 02apr@234, 03may\$452</p> <p>Note: Special character should be any of this (# or @ or \$) that can be decided randomly.</p> <p>Make sure the last random 3 digits number should not be duplicated.</p>
Business Rules & Validation	<ul style="list-style-type: none"> • All the information's are mandatory. • User Name should be alphabets. • Email id should have special character @ and dot (.) • Contact should contain number. • Gender needs to be selected. • Date of birth cannot be after the current date. • Email id should not exist in the system.

2.1.1.2 Authenticate User Page

Passport Visa Management System Req-2.1.4.2	Authenticate User Page
Functional Requirements	This module will be used by customers to login into the system.
User Interface Requirements	<ol style="list-style-type: none"> 1. Customer ID: The Customer name should be filled by the user 2. Contact Number: Displays the Contact Number of the customer 3. Password: The Customer Password should be filled by the user

Trigger	Customer triggers the functionality to login to the system.
Pre-Conditions / Assumptions	The customer details must already exist in the system.
Post Condition	The customer can login into the system.
Success End Condition	The customer can book the hotels or cancel/edit the reservation and can make the payment.
Failed End Condition	The system should throw an error message if the validation is wrong. The error message and error code needs to be identified during design. The validation and business rules are mentioned in the business rules section.
Steps & Actions	System should authenticate the user details.
Business Rules & Validations	<ol style="list-style-type: none"> 1. Customer Id and Old password must match with the database. If not, throw an error message. 2. New password must not be the same as the old password. 3. New password and confirm password must be the same. 4. Password must be alphanumeric and must contain at least one symbol and it should not contain the customer name. Password should not allow any blank spaces.

2.1.2 Apply Passport Module

2.1.2.1 Apply Passport Page

Passport Visa Management System Req-2.1.2.1	Apply Passport Page
--	---------------------

Functional Requirements	This module will be used by the user to apply the passport based on different parameters. The system should generate the passport id and expiry date. Update the details into the database.
User Interface Requirements	<ol style="list-style-type: none"> 1) User ID: Login Id entered by the user 2) Country: Represents which country the user belongs to. 3) State: Represents the State of the user. 4) City: Represents the city user belongs. 5) Pin: Security Pin Number 6) Type of Service: Service type of the application 7) Booklet type: Booklet type of the passport 8) Issue Date: Date of issue passport <p>After successful validation of the details, the system generate the Temporary ID, the expiry Date of the passport and the amount to be paid.</p>
Trigger	User triggers the functionality once he/she submits the details into the system
Pre-Conditions / Assumptions	<ul style="list-style-type: none"> • Up on valid login all the details to apply the passport should be given.
Post Conditions	<ul style="list-style-type: none"> • System should generate the temporary id and passport expiry date based on date of issue. • The user passport data should be stored into the system.
Success End Condition	<p>On successfully saving the details, message will be displayed as</p> <p>Need the passport number while giving payment? Please note down your Id</p> <p>"<Passport Id >". Passport application cost is Rs. <Amount></p>
Failed End Condition	The system should throw an error message if one of the validations is wrong. The error message and error code needs to be identified during design. The validation and business rules are mentioned in the business rules section.

Steps & Actions

1. System should collect all the necessary information.
2. Get the username, password for the login. After successful login authentication collect the required details for apply passport.
3. **Generate the passport id.**

Step - 1: Decide the passport Id format with the help of following details

<i>TYPE_OF_BOOKLET</i>	<i>ID Generation Format</i>
<i>30 Pages</i>	<i>FPS-30XXXX</i>
<i>60 Pages</i>	<i>FPS-60XXXX</i>

Step - 2: Get the maximum passport number from the database table.

Step - 3: Next XXXX = XXXX + 1

4. Generate the passport expiry date.

Expiry Date = issue Date + 10 Years.

5. Get the passport application cost as follows

<i>TYPE_OF_SERVICE</i>	<i>AMOUNT</i>	<i>PASSPORT_TYPE</i>
Normal	2500	NEW
Tatkal	5000	NEW

6. Identify the state and city information as follows.(Needs to be maintained in DB)

		<table><tr><th>STATE_ID</th><th>STATE_NAME</th></tr><tr><td>S_O001</td><td>Orissa</td></tr><tr><td>S_W002</td><td>West Bengal</td></tr><tr><td>S_B003</td><td>Bihar</td></tr><tr><td>S_K004</td><td>Kerala</td></tr><tr><td>S_A005</td><td>Andhra Pradesh</td></tr></table>	STATE_ID	STATE_NAME	S_O001	Orissa	S_W002	West Bengal	S_B003	Bihar	S_K004	Kerala	S_A005	Andhra Pradesh
	STATE_ID	STATE_NAME												
	S_O001	Orissa												
	S_W002	West Bengal												
	S_B003	Bihar												
	S_K004	Kerala												
	S_A005	Andhra Pradesh												

STATE_ID	CITY_NAME
S_O001	Bhubaneswar
S_O001	Cuttack
S_O001	Jajpur
S_W002	Kolkata
S_W002	Howrah
S_B003	Patna
S_B003	Gaya
S_K004	Kochi
S_K004	Kayamkulam
S_A005	Hyderabad
S_A005	Bellampalle

Business Rules & Validations	<ul style="list-style-type: none"> • User Id entered should be in valid id (PASS-XXXX) format. XXXX should be a 4 digit number. Otherwise system should display appropriate error message. • Pin has to be numeric and must be of six digits otherwise system should display appropriate error message. • Booklet type has to be chosen should be either ("30 Pages" or "60 Pages"). • Issue date entered should be of valid date format (dd/mm/yy) and must not be less than current date otherwise system should display appropriate error message. • User Id can't be left blank. • Country can't be left blank. • State can't be left blank. • City can't be left blank. • Pin can't be left blank. • Booklet Type can't be left blank. • Issue date can't be left blank.
---	--

2.1.3 Apply Visa Module

2.1.3.1 Apply Visa Page

Passport Visa Management System Req-2.1.3.1	Apply Visa Page
Functional Requirements	This module will be used by the user to apply visa based on different parameters.
User Interface Requirements	User ID - User id used while Login. Country - To place which your applying for Visa. Occupation- Occupation of the User. Date of Application - Represent the date of applying for visa
Trigger	User triggers the functionality once he/she submits the details into the system

Pre-Conditions / Assumptions	The system needs to collect the required information for applying Visa.												
Post Conditions	The system needs to generate the date of expiry and Visa number.												
Success End Condition	<p>On successfully saving the details the information should be displayed in the below format:</p> <p>Dear User your Visa request has been accepted successfully with id <VISA_ID>, User Id <userId> Destination <destination> Employee Occupation <occupation> Date of Application <date Of Application></p> <p>Date of Issue <date Of Issue></p> <p>Date of Expiry <date Of Expiry></p> <p>Registration Cost <registration Cost></p>												
Failed End Condition	The system should throw an error message if any one or more of the validations fail. The error messages and error codes need to be identified during design. The validation and business rules are mentioned in the business rules section.												
Steps & Actions	<ol style="list-style-type: none"> 1. System should collect all the necessary information for applying visa. 2. Get the Country where he/she needs the visa. <p>Decide visa Registration cost is based on place and applicant's occupation</p> <table> <tr> <th><i>APPLICANT'S OCCUPATION</i></th><th><i>PLACE</i></th></tr> <tr> <td><i>Student</i></td><td><i>USA</i></td></tr> <tr> <td><i>Private Employee</i></td><td><i>USA</i></td></tr> <tr> <td><i>Government Employee</i></td><td><i>USA</i></td></tr> <tr> <td><i>Self Employed</i></td><td><i>USA</i></td></tr> <tr> <td><i>Retired Eemployee</i></td><td><i>USA</i></td></tr> </table>	<i>APPLICANT'S OCCUPATION</i>	<i>PLACE</i>	<i>Student</i>	<i>USA</i>	<i>Private Employee</i>	<i>USA</i>	<i>Government Employee</i>	<i>USA</i>	<i>Self Employed</i>	<i>USA</i>	<i>Retired Eemployee</i>	<i>USA</i>
<i>APPLICANT'S OCCUPATION</i>	<i>PLACE</i>												
<i>Student</i>	<i>USA</i>												
<i>Private Employee</i>	<i>USA</i>												
<i>Government Employee</i>	<i>USA</i>												
<i>Self Employed</i>	<i>USA</i>												
<i>Retired Eemployee</i>	<i>USA</i>												

	Student	CHINA	1500											
	Private Employee	CHINA	2000											
	Government Employee	CHINA	3000											
	Self Employeed	CHINA	4000											
	Retired Eemployee	CHINA	2000											
	Student	JAPAN	3500											
	Private Employee	JAPAN	4000											
	Government Employee	JAPAN	4500											
	Self Employeed	JAPAN	9000											
	Retired Employee	JAPAN	1000											
	<div>6. Date of Issue for the visa is calculated as 10 days after date of application.</div> <div>7. Date of expiry for the visa (in years) is based on the applicant’s occupation.</div> <div>8. Check the date of expiry of the passport is after the date of expiry of the Visa. If not give expiry date of visa as expiry date of passport.</div> <div>9. Visa number is generated based on applicant’s occupation as follows</div>													
	<table><tr><th>APPLICANT’S_OCCUPATION</th><th>VISA PERMIT(IN YEARS)</th></tr><tr><td>Student</td><td>2</td></tr><tr><td>Private Employee</td><td>3</td></tr><tr><td>Government Employee</td><td>4</td></tr><tr><td>Self Employed</td><td>1</td></tr><tr><td>Retire Employee</td><td>1.5</td></tr></table>			APPLICANT’S_OCCUPATION	VISA PERMIT(IN YEARS)	Student	2	Private Employee	3	Government Employee	4	Self Employed	1	Retire Employee
APPLICANT’S_OCCUPATION	VISA PERMIT(IN YEARS)													
Student	2													
Private Employee	3													
Government Employee	4													
Self Employed	1													
Retire Employee	1.5													
Note: The (XXXX) value is separate for each category.														

Business Rules & Validations	<ul style="list-style-type: none"> • All fields are mandatory. • Passport Number should exist in the system. • Date of Application cannot be before the current date. • Check the date of expiry of the passport is after the date of expiry of the Visa. If not give expiry date of visa as expiry date of passport.

2.1.4 Re-issue Passport Module

2.1.4.1 Passport Re Issue Page

Passport Visa Management System Req-2.1.4.1	Passport Re-Issue Page
Functional Requirements	This module will be used by the user to request renewal of passport.
User Interface Requirements	<ol style="list-style-type: none"> 1. Reason for re issue: Reason for applying the passport again 2. User ID: Login Id entered by the user 3. Country: Represents which country the user belongs to. 4. State: Represents the State of the user. 5. City: Represents the city user belongs. 6. Type of Service: Service type of the application 7. Booklet type: Booklet type of the passport 8. Issue Date: Date of issue passport <p>After successful validation of the details, the system generates the Temporary ID, the expiry Date of the passport and the amount to be paid. Update the new details in the database for the existing user.</p>
Trigger	User triggers the functionality once he/she submits the details into the system.

Pre-Conditions / Assumptions	<ul style="list-style-type: none"> In the User Login screen all fields should be displayed properly. 	
Post Condition	<ul style="list-style-type: none"> System should check reason for the re issue, collect all the required information. 	
Success End Condition	<ul style="list-style-type: none"> On Successful saving the details the message display is <p>Passport re issue is successfully done.</p> <p>Amount to be paid is Rs. <amount></p> <p>Passport issue date is <issue date> and expiry date is <expiry date></p>	
Failed End Conditions	<p>The system should throw an error message if any one or more of the validations fail. The error messages and error codes need to be identified during design. The validation and business rules are mentioned in the business rules section.</p>	
Steps & Actions	<ol style="list-style-type: none"> System should collect all the necessary information. Get the username, password for the login. After successful login authentication collect the required details for apply passport. Update the passport information like new passport number, Type of booklet selected, expiry date of the passport for that user. The old passport details also to be maintained in the database. Generate the passport id. <p>Step - 1: Decide the passport Id format with the help of following details</p>	
	TYPE_OF_BOOKLET	ID Generation Form
	30 Pages	FPS-30XXXX
	60 Pages	FPS-60XXXX

Step - 2: Get the maximum passport number from the database table.

Step - 3: Next XXXX = XXXX + 1

12. Generate the passport expiry date.

expiry Date = issue Date + 10 Years.

13. Get the passport application cost as follows

<i>TYPE_OF_SERVICE</i>	<i>AMOUNT</i>	<i>PASSPORT_TYPE</i>
Normal	1500	RENEWAL
Tatkal	3000	RENEWAL

14. Identify the state and city information as follows.(Needs to be maintained in DB)

<i>STATE_ID</i>	<i>STATE_NAME</i>
<i>S_O001</i>	<i>Orissa</i>
<i>S_W002</i>	<i>West Bengal</i>
<i>S_B003</i>	<i>Bihar</i>
<i>S_K004</i>	<i>Kerala</i>
<i>S_A005</i>	<i>Andhra Pradesh</i>

<i>STATE_ID</i>	
<i>S_O001</i>	
<i>S_O001</i>	
<i>S_O001</i>	

		<i>S_W002</i>		<i>Kolkata</i>
		<i>S_W002</i>		<i>Howrah</i>
		<i>S_B003</i>		<i>Patna</i>
		<i>S_B003</i>		<i>Gaya</i>
		<i>S_K004</i>		<i>Kochi</i>
		<i>S_K004</i>		<i>Kayamkulam</i>
		<i>S_A005</i>		<i>Hyderabad</i>
		<i>S_A005</i>		<i>Bellampalle</i>
Business Rules & Validations	<ul style="list-style-type: none"> • User Id entered should be in valid id (PASS-XXXX) format. XXXX should be a 4 digit number. Otherwise system should display appropriate error message. • Pin has to be numeric and must be of six digits otherwise system should display appropriate error message. • Booklet type has to be chosen should be either ("30 Pages" or "60 Pages"). • Issue date entered should be of valid date format (dd/mm/yy) and must not be less than current date otherwise system should display appropriate error message. • User Id can't be left blank. • Country can't be left blank. • State can't be left blank. • City can't be left blank. • Pin can't be left blank. • Booklet Type can't be left blank. • Issue date can't be left blank. 			

2.1.5 Visa Cancellation Module

2.1.5.1 Visa Cancellation Page

Passport Visa Management System	Visa Cancellation Page
---------------------------------	------------------------

Req-2.1.5.1	
Functional Requirements	This module will be used by the user to perform visa cancellation. Update all the details into the system.
User Interface Requirements	<ol style="list-style-type: none"> 1. User ID: User ID of the user who is currently logged in. 2. Passport Number: Passport number of the user 3. Visa Number: Visa number to cancel 4. Date of Issue: Date of issue of visa <p>After getting the valid details, System needs to calculate the amount to be paid. Update the status as cancelled.</p>
Trigger	None
Pre-Conditions / Assumptions	It is assumed that at the time of commencement of this module the user is logged in. So that the user ID is available module-wide.
Post Conditions	The system will calculate the cancellation charges and update the same in database.
Success End Condition	<p>After the VISA Cancellation request is submitted into the system, a message should be displayed saying</p> <p>“Your request has been submitted successfully. Please pay \$\$\$ to complete the cancellation process”</p> <p>Where \$\$\$ is the cancellation charge.</p>
Failed End Condition	The system should throw an error message if one of the validations is wrong. The error message and error code needs to be identified during design. The validation and business rules are mentioned in the business rules section.

Steps & Actions	<ol style="list-style-type: none"> 1. System should collect all the required details from the user. 2. After login get the security question and answer from the user. 3. Once the user had been authenticated, the system will ask for the Passport Number, VISA Number and VISA Issue Date. 4. The system should calculate the cancellation charges as follows.. 5. Get the difference between the date of expiry of VISA and the date of VISA-cancellation request should be calculated in months (say Diff_mon). <table border="1" data-bbox="667 792 1409 1666"> <tr> <td data-bbox="667 792 853 952" rowspan="2">Student</td><td data-bbox="853 792 1409 871"><i>Diff_mon</i> < 6 months</td></tr> <tr> <td data-bbox="853 871 1409 952"><i>Diff_mon</i> >= 6 months</td></tr> <tr> <td data-bbox="667 952 853 1171" rowspan="3">Private Employee</td><td data-bbox="853 952 1409 1030"><i>Diff_mon</i> < 6 months</td></tr> <tr> <td data-bbox="853 1030 1409 1108">6 <=<i>Diff_mon</i> < 12 months</td></tr> <tr> <td data-bbox="853 1108 1409 1171"><i>Diff_mon</i> >= 12 months.</td></tr> <tr> <td data-bbox="667 1171 853 1377" rowspan="3">Govt. Employee</td><td data-bbox="853 1171 1409 1249"><i>Diff_mon</i> < 6 months</td></tr> <tr> <td data-bbox="853 1249 1409 1328">6 <=<i>Diff_mon</i> < 12 months</td></tr> <tr> <td data-bbox="853 1328 1409 1377"><i>Diff_mon</i> >= 12 months.</td></tr> <tr> <td data-bbox="667 1377 853 1523" rowspan="2">Self Employed</td><td data-bbox="853 1377 1409 1456"><i>Diff_mon</i> < 6 months</td></tr> <tr> <td data-bbox="853 1456 1409 1523"><i>Diff_mon</i> >= 6 months</td></tr> <tr> <td data-bbox="667 1523 853 1666" rowspan="2">Retired Employee</td><td data-bbox="853 1523 1409 1583"><i>Diff_mon</i> < 6 months</td></tr> <tr> <td data-bbox="853 1583 1409 1666"><i>Diff_mon</i> >= 6 months</td></tr> </table> <p>10. Update the status of the visa as cancelled in the database.</p>	Student	<i>Diff_mon</i> < 6 months	<i>Diff_mon</i> >= 6 months	Private Employee	<i>Diff_mon</i> < 6 months	6 <= <i>Diff_mon</i> < 12 months	<i>Diff_mon</i> >= 12 months.	Govt. Employee	<i>Diff_mon</i> < 6 months	6 <= <i>Diff_mon</i> < 12 months	<i>Diff_mon</i> >= 12 months.	Self Employed	<i>Diff_mon</i> < 6 months	<i>Diff_mon</i> >= 6 months	Retired Employee	<i>Diff_mon</i> < 6 months	<i>Diff_mon</i> >= 6 months
Student	<i>Diff_mon</i> < 6 months																	
	<i>Diff_mon</i> >= 6 months																	
Private Employee	<i>Diff_mon</i> < 6 months																	
	6 <= <i>Diff_mon</i> < 12 months																	
	<i>Diff_mon</i> >= 12 months.																	
Govt. Employee	<i>Diff_mon</i> < 6 months																	
	6 <= <i>Diff_mon</i> < 12 months																	
	<i>Diff_mon</i> >= 12 months.																	
Self Employed	<i>Diff_mon</i> < 6 months																	
	<i>Diff_mon</i> >= 6 months																	
Retired Employee	<i>Diff_mon</i> < 6 months																	
	<i>Diff_mon</i> >= 6 months																	
Business Rules & Validations	<ul style="list-style-type: none"> ▪ The selected security question should be same as the Hint question. ▪ The security question and given answer should be a valid pair. ▪ The visa number field in the Visa Cancellation 																	

	<p>Screen should not be left blank.</p> <ul style="list-style-type: none"> ▪ The passport number field in the Visa Cancellation Screen should not be left blank. ▪ Neither of the visa number nor the passport number should accept any value which is not numeric or alphabet ▪ The given passport number should correspond to the User ID. ▪ The visa for which the cancellation is requested should not have expired already.
--	--

2.2 Nonfunctional Requirements

2.2.1 UI Requirements:

1. The front end should be user-friendly and pleasant
2. Any error message or exception displayed to the user should be user-readable (and not technical)
3. All entered values should be validated
4. The UI screens are designed with any theme color along with the logo of the company.
5. Exceptions should not be shown on console.
6. No debug message should come on console.
7. All the errors and exceptions should be logged to log file.
8. Log file path should be configurable.
9. Database connections should be configurable.
10. It should not be hard coded.
11. Requirements involving UI features should be taken care in console accordingly

2.2.2 Performance Requirements:

1. All front-end pages should be served up in less than 5 seconds post click when up to 1000 users are on the application concurrently.
2. Since it's a financial operations application, transactions should be safe and fast.

2.2.3 Installation, Deployment & Operational Requirements:

1. Application Java and J2EE deployment with My SQL database or SQL Server database for persistence

Or

Application .NET Framework deployment with SQL Server database for persistence

2. This system must be up 24X7

2.2.4 Maintainability & Portability Requirements:

1. If the allocated memory space for the existing application needs to be increased then it should be possible without any impact to performance

2.2.5 Security Requirements :

1. No unauthorized users should be able to log on to the system
2. User Id/Login Id should have combination of letter and numeric values, none of the special characters are allowed (#,@,\$,%,&).
3. Validate the user id/login id and password, accordingly to the user it will redirect to the appropriate page.

3 Requirements / Use Case Summary Table

Business Requirement ID	Software Requirements ID	Short Description	Requirement Provider (Originator)	Priority	Complexity	Requirement type	Remarks
BR001	SR001	User Registration	Client	High	Medium	UI screen & backend processing	-
BR002	SR002	Apply Passport	Client	Medium	Medium	UI screen & backend processing	-
BR003	SR003	Apply Visa	Client	High	Medium	UI screen & backend processing	-
BR004	SR004	Re-Issue Passport	Client	Medium	Medium	UI screen & backend processing	-
BR005	SR005	Visa Cancellation	Client	High	High	UI screen & backend	-

