



FUNCTIONAL SPECIFICATION

Project Code:	e-Passport-01
Project Name:	e-Passport

Prepared by/Date

Reviewed by/Date

Approved by/Date

**Revision History**

Version (x.yy)	Date of Revision	Description of Change	Reason for Change	Affected Sections	Approved By
1.00	31-May-2019	Initial Draft			

Affected Groups

Development Engineering
Quality Assurance
XYZ Passport Ltd

List of Reference Documents

Name	Version No.
1. Request For Proposal	1.2
2.	
3.	
4.	



Table of Contents

1. INTRODUCTION	4
2. SYSTEMS OVERVIEW	4
3. SUB-SYSTEM DETAILS	5
4. DATA ORGANIZATION	7
5. ASSUMPTIONS	8
6. EXPECTATIONS	8
7. ACCEPTANCE CRITERIA	9
8. TRACEABILITY TO REQUIREMENTS	9
9. ACRONYMS AND GLOSSARY	9



1. Introduction

1.1 Background

XYZ Passport Ltd provides passport applying facilities to Indian citizens.

1.2 Purpose

XYZ Passport Ltd plans to develop “e-Passport” - a web application wherein a user can apply for passport online.

1.3 Scope

The scope of the e-passport will be to provide the functionality as described in Functional Requirements below. The system will be developed on a Windows 10 or Windows Server machine using ASP.NET, C#, SQL Server etc.

2. Systems Overview

2.1 System Description

The e-Passport should support the following users.

- Administrator (A)
- User (U)

The common functional requirements are explained below.

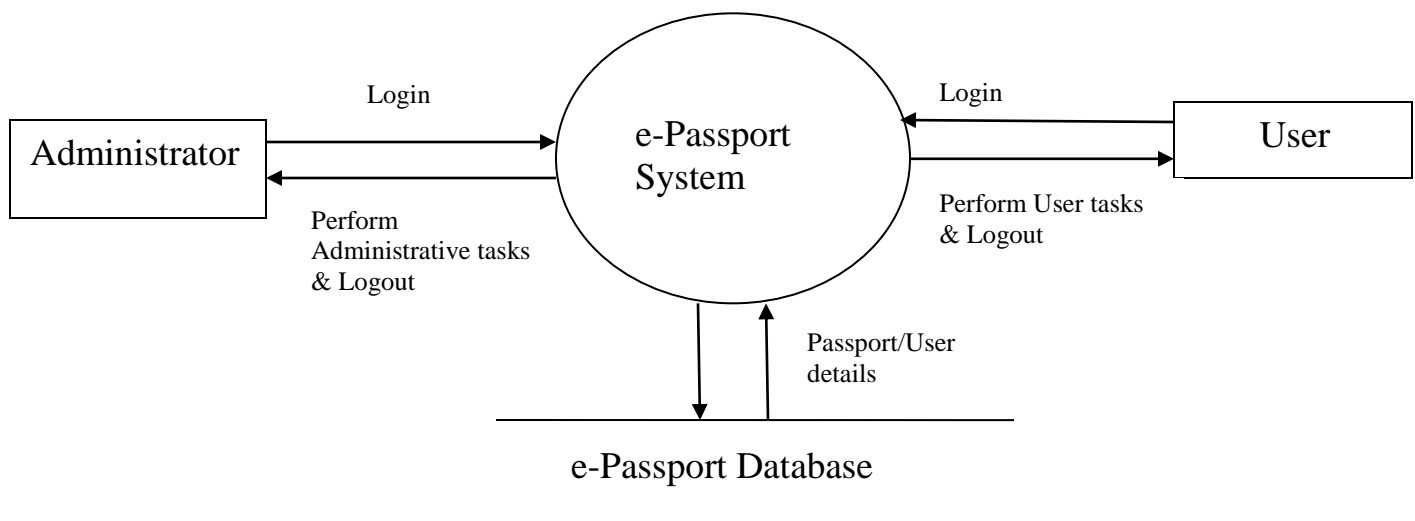
2.1.1 Authentication

Any end-user should be authenticated using a unique login ID and password.

2.1.2 Authorization

The operations supported and allowed would be based on the user type. For example, Administrator has the rights to add or modify appointment schedule details.

The functional flow of the messages across different application components is shown below.



2.2 Environment



FUNCTIONAL SPECIFICATION

e-Passport-01/FS-1.00

The system will be developed on a Windows 10 machine using ASP.NET, C# and SQL Server 2014.

- Intel hardware machine (PC i5, 4 GB RAM, 500 GB HDD)
- Windows 10
- .NET Framework 4.6 or higher
- IIS 7.0 or higher
- Visual Studio 2017 or higher
- Database – SQL Server 2014 or higher

3. Sub-system Details

3.1 Administrator

The Administrative User should be able to do the following operations once he has logged in with his unique user id and password

- Add Passport Office Details
- Add Appointment Schedule Details
- Modify Appointment Schedule Details
- View User Information
- View User Documents
- Update Passport Status (Issue/Pending/Cancel upon verification)
- Quit Session

Add Passport Office Details: As part of this operation, the admin should have the ability to add passport office details into the system and get confirmation on successful entry. OfficeID is auto-generated.

Passport Office Details include:

- OfficeID
- OfficeName
- Jurisdiction
- Address
- PhoneNumber

NOTE: More fields for passport office details can be added depending on requirement.

Add Appointment Schedule Details: As part of this operation, the admin should have the ability to add appointment schedule details into the system and get confirmation on successful entry. MonthID is auto-generated.

Appointment Schedule Details include:

- MonthID
- MonthName
- AvailableDays
- TimeSlots

Modify Appointment Schedule Details: As part of this operation, the admin should have the ability to edit appointment schedule details in the system by picking MonthID from the list.

Appointment Schedule Details include:

- AvailableDays
- TimeSlots



View User Information: As part of this operation, the admin should have the ability to view user information who has registered to the system.

View Documents: As part of this operation, the admin should have the ability to view user documents in the system by picking up ApplicantID.

Update Passport Status: As part of this operation, the admin should have the ability to update the status of user passport upon verification. Status can be issued/pending/cancel.

Quit Session: As part of this operation, admin should have the ability to quit the session completely.

3.2 User

The User should be able to do the following operations once he has logged in with his unique LoginID and password.

- Fill and Submit Application Form
- View/Print Application Form
- Upload Supporting Documents
- Request for Appointment
- Change Appointment
- Quit Session

User Registration

An unregistered user should be able to register into the system by selecting the passport office that falls under his/her jurisdiction with the following details.

Registration details include (all mandatory):

- FirstName
- LastName
- DOB
- Gender
- EmailID
- Password
- PhoneNo

When the registration is complete, a LoginID (numeric) will be auto-generated and displayed to the user.

(Refer to section 4. Data Organization → Table 1: RegistrationDetails)

Fill Application Form

A registered user should be able to fill and submit the application form into the system with the following details.

Application Form details include (all mandatory):

Applicants Details:

- FirstName
- LastName
- DOB
- PlaceOfBirth
- Gender



FUNCTIONAL SPECIFICATION

e-Passport-01/FS-1.00

- State
- District
- MaritalStatus
- PAN
- EmploymentType
- EducationalQualification

Family Details:

- Fathers Name
- Mothers Name

Address Details:

- HouseNo
- Street Name
- City
- State
- District
- PINCode
- TelephoneNumber
- EmailID

Reference Details:

- Reference Name
- Address
- Telephone Number

Note: Application Form details provided here are indicative. Extra details can be added if required.

When the application form filling is complete, an ApplicantID (numeric) will be auto-generated and displayed to the user.

View/Print Application Form: As part of this operation, the user should have the ability to get all application form details by using ApplicantID and print the same.

Upload Supporting Documents: As part of this operation, the user should have the ability to upload all supporting documents by using ApplicantID.

Request for Appointment: As part of this operation, the user should be able to take online appointment based on date and timeslots available.

Change Appointment: As part of this operation, the user should be able to change appointment schedule.

Quit Session: As part of this operation, the user should have the ability to quit the session completely.

4. Data Organization

This section explains the data storage requirements of the e-Passport application and **indicative** table (database) structure. The following sections explain few of the tables required for the application and the other tables will have to be designed accordingly.

**4.1 Table: User Information**

The user specific details such as name, address, authentication and authorization / privileges should be kept in one or more tables, as necessary and applicable.

Table 1: RegistrationDetails

Field Name	Data Type	Description
LoginID	VARCHAR(9)	It will be a numeric value sequentially generated by the system and need not be passed from the client. It will be added by the system whenever a new user gets registered.
Password	VARCHAR(9)	Encrypted Password
First Name	VARCHAR(15)	First Name of the user
Last Name	VARCHAR(15)	Last Name of the user
DOB	DATE	DOB of the user
Gender	VARCHAR(7)	Gender of the user
EmailID	VARCHAR(30)	EmailID of the user
PhoneNo	VARCHAR(10)	Phone No of the user

4.2 Table: Authentication Information

The table contains Authentication Information for Administrator and User

Table 2: LoginCredentials

Field Name	Data Type	Description
UserType	VARCHAR(1)	A-Administrator, U-User
LoginID	VARCHAR(9)	User Identification
Password	VARCHAR(9)	Password

Note: Similarly, create other necessary normalized tables.

5. Assumptions

- User Interface: The type of client interface (front-end) to be supported is – GUI based
- The scope of the application is limited to only one country

6. Expectations

- The application should be designed using client server architecture
- The server should be a concurrent server servicing multiple clients
- Database can be implemented using SQL server 2014
- To begin with, the application should support at least 1 admin
- Compilation and Build should be done using Visual Studio 2017 or higher
- Source-code and all documents must be maintained (checked-in) in configuration management system (subversion)
- Coding standards should be followed
- All requirements are to be implemented using MVC 5, Entity Framework 6 & Web API (Optional)
- Regular Expressions should be implemented at appropriate places to validate the data within the application
- Deliverables should include compiled and tested source code.



7. Acceptance Criteria

All P1 requirements have to be mandatorily implemented

8. Traceability to Requirements

Document Reference ID & Description: (Doc ID from which this document is derived)		
Sl. No.	Reference document: RS Requirement/Feature (Section ID/Name)	Current document: FS Location (Section ID/Name)
1.	Requirements as mentioned in Section 2.1	Section 3.1
2.	Requirements as mentioned in Section 2.2	Section 3.2,4.1
3.	Requirements as mentioned in Section 2.3	Section 2.1.1, 2.1.2, 4.2
4.	Requirements as mentioned in Section 2.5	Section 3.1, 3.2

9. Acronyms and Glossary

Abbreviation	Remark
e-Passport	e-Passport
RS	Requirement Specification
FS	Functional Specification