SOFTWARE ENGINEERING AND PROJECT MANAGEMENT

ANAIAPPAN R (RA1911003020301)

JOEAL CHRISH H (RA1911003020335)

SELIN RIONA V (RA1911003020350)

TABLE OF CONTENT

5	**08/02/2021**	**PROJECT EFFORT BASED ON RESOURCES** 5.1.1 Work Breakdown structure 5.1.2 Risk Management		
6	**15/02/2021**	**ESTIMATION OF PROJECT METRICS** 6.1.1 Function Point Analysis 6.1.2 COCOMO Model		
7	**15/02/2021**	**DESIGN** 7.1.1 System Architecture		
8	**22/02/2021**	**MODELLING UML USE CASE DIAGRAMS & CAPTURING USE CASE SCENARIOS** 8.1.1 Use Case Diagram Description 8.1.2 Use Case Diagram		
9	**01/03/2021**	**ER MODELLING FROM THE PROBLEM STATEMENTS** 9.1.1 ER Diagram Description 9.1.2 ER Diagram		
10	**08/03/2021**	**IDENTIFYING DOMAIN CLASSES FROM THE PROBLEM STATEMENTS** 10.1.1 Domain Class Diagram Description 10.1.2 Domain Class Diagram		
11	**15/03/2021**	**STATECHART AND COMMUNICATION MODELLING** 11.1.1 State Chart Diagram Description 11.1.2 State Chart Diagram 11.1.3 Communication Diagram Description 11.1.4 Communication Diagram		

12	**22/03/2021**	**MODELLING UML CLASS DIAGRAM & SEQUENCE DIAGRAMS** 12.1.1 Class Diagram Description 12.1.2 Class Diagram 12.1.3 Sequence Diagram Description 12.1.4 Sequence Diagram		
13	**29/03/2021**	**MODELLING DATA FLOW DIAGRAM** 13.1.1 Data Flow Diagram (DFD) Description 13.1.2 Data Flow Diagram (DFD)		
14	**05/04/2021**	**IMPLEMENTATION** 14.1.1 Module Implementation		
15	**12/04/2021**	**ESTIMATION OF TEST COVERAGE METRICS & STRUCTURAL COMPLEXITY** 15.1.1 Control Flow Graph 15.1.2 McCabe's Cyclomatic Complexity 15.1.3 Optimum Value of Cyclomatic Complexity		
16	**19/04/2021**	**DESIGNING TEST SUITES** 16.1.1 Software Testing 16.1.2 Testing Frameworks 16.1.3 Master Test Plan 16.1.4 Manual Testing		
17	**19/04/2021**	**CONCLUSION**		
18	**19/04/2021**	**REFERENCES**		

PROJECT COMPLETED IN THE SESSION **2020 - 2021**

1.1 PROBLEM STATEMENTS

DATE	**25/01/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

1.1.1 PROBLEM STATEMENT

INTRODUCTION:

Passport Automation System is an interface between the Applicant and the Authority responsible for the issue of Passport. If the entire process of the issue of the Passport is done manually then it would take several months for the Passport to reach the applicant as the number of applicants for the Passport are increasing every year. Therefore, this system aims at improving the efficiency in the issue of the Passport, reduces the complexities involved in it and for the effective dispatch of the passport to the maximum possible extent.

PROBLEMS:

The following problems were faced during the manual process of Passport dispatch system:

- **TEDIOUS PROCESS**: Requires a large number of skilled individuals in various fields of issuing Passport. Chances of error and workload on each individuals is more.
- **TIME CONSUMPTION**: Takes a lot of time when it is done manually. Search for particular data also leads to unnecessary delays.
- **DATABASE MANAGEMENT**: Due to the increasing number of applicants, the management and the processing of data are difficult to update and maintain with time.
- **SECURITY:** More accessibility to data, which made tampering much easier.

PROPOSAL:

The above problem can be overcome by using Passport Automation System. Instead of using manual process, we can automate this process of issuing Passport which increases the efficiency and reduces the complexities.

1.1.2 ONE PAGE BUSINESS CASE TEMPLATE

THE PROJECT:

- ♦ Since, the number of applicants for the passport is increasing every year, manual process becomes very much complicated.
- ♦ The development of this **"Passport Automation system"** makes the entire process automated keeping in mind the view of database integration approach.
- ♦ Authentication is provided where only the registered users can access.

THE HISTORY:

- ♦ The manual system does not provide secured registration and profile management of the Applicants properly**.**
- ♦ In manual system, all details are taken in the form of documents, which can be misplaced anytime.
- ♦ Every individual had to stand in queue, which is more time consuming and involves more manpower.

CONSTRAINTS:

- ⊗ The Applicants require a computer to submit their information.
- ⊗ Prior knowledge of computers and English language should be known

⊗ Although the security is given high importance, there is always a chance of intrusion in the web world which requires constant monitoring.

⊗ The Applicant must be careful while submitting the information.

APPROACH:

- We approach this solution from the technical point of view. So, we are keen in maintaining an easy way to execute interface for the applicant as well as the back end maintenance.
- The development of this project will require front end and back end programming knowledge. Thus, it is important to equip with the same.

BENEFITS:

✓ Security is provided.
✓ Less time consumption.
✓ Improves the efficiency.
✓ Reduces the complexities.
✓ Elimination of human error.

Disclaimer

Any articles, templates, or any information provided by the Smart sheet on the website are for reference only. While we strive to keep the information up to date and correct, we make no representations or warranties of any kind, expressed or implied, about the completeness, accuracy, reliability, suitability, or availability with respect to the website or the information, articles, templates, or related graphics contained on the website. Any reliance you place on such information is therefore strictly at your own risk.

2.1 STAKEHOLDERS AND USER DESCRIPTION

DATE	**01/02/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

2.1.1 IDENTIFYING STAKEHOLDERS

1] USER:

- The persons who will use the project's service are known as users.
- In this project, the end user will be the Applicant of the Passport and the authority is the integrated user.

2] SPONSOR:

- Sponsor is the person or the organization that provides financial support for the project.
- As of now, this project is self-sponsored.
- Once we upgrade, we may require financial interference of a management or a crowd-sourcing initiative.

3] PORTFOLIO MANAGER:

- Portfolio Managers are primarily responsible for creating and managing investment allocations for private clients.

4] PORTFOLIO REVIEW BOARD:

- The Portfolio Review Board is the overarching decision-making governing body, guiding and prioritizing product investments for existing products, products in development and future product projects.

5] PROGRAM MANAGER:

- A Program Manager articulates a program's strategy and objectives and assesses how it will impact a business.

6] PROJECT MANAGEMENT OFFICE (PMO):

- A project management office (PMO) is a team or department that sets and maintains standards for project management throughout an organization.

7] PROJECT MANAGER:

- Project Managers (PMs) are responsible for planning, organizing, and directing the completion of specific projects for an organization while ensuring these projects are on time, on budget, and within scope.

8] PROJECT TEAM:

- A project team is comprised of the project manager, project management team and the other members who carry out work

9] FUNCTION MANAGER:

- Functional Managers are tasked with controlling the resources that will support a project, such as financial backing and skilled employees.

2.1.2 USER STORY

1] APPLICANTS FOR THE PASSPORT:

An Applicant here is the individual who applies for the Passport.

2] PASSPORT INFORMATION PROVIDERS:

Applicants can ask any kinds of doubt related to passport from the Passport information providers.

3] FINANCE DEPARTMENT:

The finance department is responsible for management of the cash flow when the processing fee is applied.

2.1.3 IDENTIFYING THE PROCESS MODULES

This project consists of 5 modules namely:

1] LOGIN:

In this module, the applicant can enter the user name and password. If the user name and password is correct then it can be entered into the specific web page. Otherwise, re-enter the user name and password after the particular time only.

2] FILL THE APPLICATION:

Before filling the application, select either apply for new passport or renewal the old passport. We can fill the application form with corresponding ID proof and address proof and then submit the form.

3] VERIFICATION OF THE FORM:

The responsible authority verifies the given information and then it is sent to local police. Local police enquiry that corresponding information is true or not.

4] VALIDATION CHECKING:

After verifying it will be forwarded to higher authorities there, the process of validation takes place. Every passport is valid for 5 years.

5] ISSUE THE PASSPORT:

After verification and validation, the passport can be issued to the applicant through the post. Then, the passport can be checked and sign the post record the accepted the passport.

2.1.4 ARRIVING AT THE PROBLEM STATEMENT

PROBLEM STATEMENT:

- Website loading issues.
- Online payment issues.
- Bank server issues.
- Refund sometimes take longer time after cancellation.

2.1.5 COMPARISION BETWEEN AGILE AND WATERFALL MODEL

- Agile model is an incremental delivery process whereas, the Waterfall model is highly structured and systematic.
- Agile model allows to change the requirements after the development process starts (i.e) it is more flexible whereas, the Waterfall model is rigid.
- Customer interaction is very high in Agile model and less in Waterfall model.
- In Agile model progress is measured in terms of the developed and delivered functionalities whereas, In Waterfall model progress is generally measured in terms of the number of completed and reviewed artifacts such as requirement specifications, design documents, test plans, code reviews, etc.

3.1 IDENTIFYING THE REQUIREMENTS FROM THE PROJECT STATEMENT

DATE	08/02/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

3.1.1 REQUIREMENTS

Project requirements are conditions or tasks that must be completed to ensure the success or completion of the project. They provide a clear picture of the work that needs to be done.

3.1.2 FUNCTIONAL REQUIREMENTS

It describes the behavior of a particular system. In this project, login is the functional requirement. After checking entity of login, we can show the detail based on:

- ♦ Analysis
- ♦ Design
- ♦ Implementation
- ♦ Maintenance

3.1.3 NON - FUNCTIONAL REQUIREMENTS

- The system can load at the speed of 2.4GHz-3.6GHz
- Memory 4GB RAM
- Transferring data speed 50 Mbps in time
- It is high portability, reliability, accepting failure rates and user friendly

3.1.4 HARDWARE REQUIREMENTS

- Processor: Pentium – IV
- Hard drive: 320 GB
- RAM: 4GB
- DVD-Drive: 1

3.1.5 SOFTWARE REQUIREMENTS

∇ Operating system: Windows XP
∇ Front-end:
∇ Back-end:

4.1 PROJECT PLAN AND PROJECT EFFORT BASED ON RESOURCES

DATE	08/02/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

4.1.1 PROJECT PLAN

PROJECT NAME:

"PASSPORT AUTOMATION SYSTEM"

PROJECT MEMBERS:

Our Project consists of 3 members:

1] ANAIAPPAN R - RA1911003020301

2] JOEAL CHRISH H - RA1911003020335

3] SELIN RIONA V - RA1911003020350

MODULES:

- Login

- Fill the Application
- Verification of the form
- Validation Checking
- Issue the Passport

SCHEDULING:

TASK	START DATE	END DATE
Business case development	**25/01/2021**	**27/01/2021**
Identifying Stakeholders, Process Modules and required Modules	**01/02/2021**	**03/02/2021**
Identifying requirements	**08/02/2021**	**09/02/2021**
Setting cost estimates and budget	**10/02/2021**	**11/02/2021**
UML Diagrams	**22/02/2021**	**08/03/2021**
Coding	**15/03/2021**	**05/04/2021**
Final Revisions	**21/04/2021**	**25/04/2021**

PROJECT COST:

- **Development of the project:**

RESOURCE REQUIREMENT	COST
Computer with core i7 8th gen processor, at least 8Gb of RAM, running on windows 10	Rs.50000/-
Code	Open source
Printing	Rs.500/-

- **Server-End:**

RESOURCE REQUIREMENT	COST
My SQL	Enterprise Edition Rs.14078/-
Azure Web Services	Std. edition Rs.24000/-
UPS	Rs.34000/-

- **Other cost:**

Employee Salary	--
Maintenance Cost	Rs.2500/- per month

4.1.2 IDENTIFYING JOB ROLES AND RESPONSIBILITIES

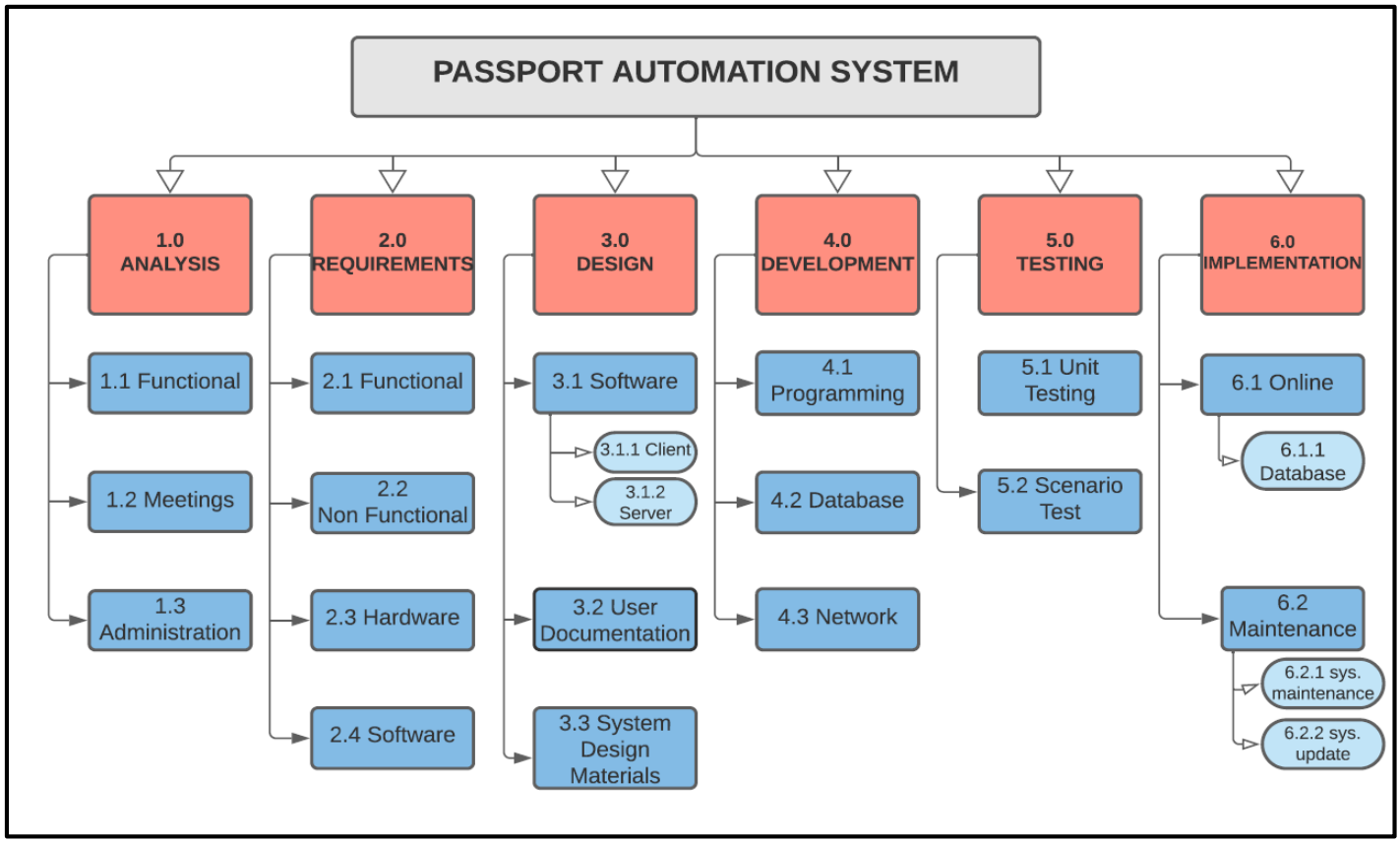
MEMBERS	ROLES AND RESPONSIBILITIES
SELIN RIONA V (RA1911003020350)	• **TEAM LEADER:** Responsibility of coordination of the team, checking for errors, updating for the current status of the project, guiding the team. • **DEVELOPER:** Responsibility of coding, compiling and debugging of the modules. • **SOFTWARE ARCHITECT:** Design of UML Diagrams and other blueprint.
ANAIAPPAN R (RA1911003020301)	• **WEB DEVELOPER:** Responsibility of designing the website and interfacing with the server. • **DEVELOPER:** Responsibility of coding compiling and debugging of the modules. • **MANUAL TESTER:** Responsibility of testing the project at a smaller level and reporting the errors.
JOEAL CHRISH H (RA1911003020335)	• **WEB DEVELOPER:** Responsibility of designing the website and interfacing with the website. • **TESTER:** Responsibility of testing the project at a vast level and variety of ways. • **DESIGNER:** Identifying areas for modification in existing programs and developing the modification.

5.1 PROJECT EFFORT BASED ON RESOURCE

DATE	08/02/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

5.1.1 WORK BREAKDOWN STRUCTURE

A Work Breakdown structure is a deliverable – oriented hierarchical decomposition of the work to be executed by the project team to accomplish the project objectives and create the required deliverables. A WBS is the cornerstone of effective project planning, execution, controlling, monitoring and reporting. All the work contained within WBS is to be identified, estimated, scheduled and budgeted.



5.1.2 RISK MANAGEMENT

DESCRIPTION:

In the modern world, risk management refers to the practice of identifying potential risks in advance by analyzing them and taking precautionary steps to curb the risk**.**

Risk management is the identification, evaluation, and prioritization of risks, controlling the probability or impact of unfortunate events.

RISKS TO BE HANDLED:

- Hacker's intent to get user data.
- Low website speed.
- Improper internet connection.
- Maintaining Database.

MANAGING RISKS:

- ✓ Performing periodic maintenance of the server.
- ✓ Using of Captcha and other security protection things to protect from bot attack.
- ✓ The bugs must be removed and the code must pass as many test cases as possible.

6.1 ESTIMATION OF PROJECT METRICES

DATE	**15/02/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

Estimation of various project parameters is a basic project planning activity. The important project parameters that are estimated include:

- ✓ Project size
- ✓ Cost
- ✓ Duration
- ✓ Effort

6.1.1 FUNCTION POINT ANALYSIS

Function Point Analysis (FPA) refers to the practice of using function points to size and estimate the cost of work on systems. Function Points are a normalized unit of measure used to: Quantify the amount of business functionality a system provides business users.

TYPES OF FP ATTRIBUTES:

SI.NO	MEASUREMENT PARAMETERS	EXAMPLES
1.	Number of External Inputs (EI)	Input screen and Tables
2.	Number of External Outputs (EO)	Output screen and Reports
3.	Number of External Inquiries(EQ)	Prompts and Interrupts
4.	Number of Internal Files (ILF)	Databases and Directories
5.	Number of External Interfaces (EIF)	Shared Databases and routines

6.1.2 COCOMO MODEL

The **COCOMO Model** estimates the cost for software product development in terms of effort (resources required to complete the project work) and schedule (time required to complete the project work) based on the size of the software product. According to COCOMO, there are three modes of software development projects that depend on complexity. Such as:

1.ORGANIC PROJECT:

It belongs to small & simple software projects which are handled by a small team with good domain knowledge and few rigid requirements.

2.SEMIDETACHED PROJECT:

It is an intermediate (in terms of size and complexity) project, where the team having mixed experience (both experience & inexperience resources) to deals with rigid/non-rigid requirements.

3.EMBEDDED PROJECT:

This project having a high level of complexity with a large team size by considering all sets of parameters (software, hardware and operational).

- **Effort (E) = $a_b(KLOC)^{b}{}_b$**
- **Deployment (D) = $c_b(E)_b$**
- **Productivity (P) = E/D**

Where, a_b, b_b, c_b, d_b are called as coefficients

PROJECT	a_b	b_b	c_b	d_b
Organic Mode	2.4	1.05	2.5	0.38
Semidetached Mode	3.0	1.12	2.5	0.35
Embedded Mode	3.6	1.20	2.5	0.32

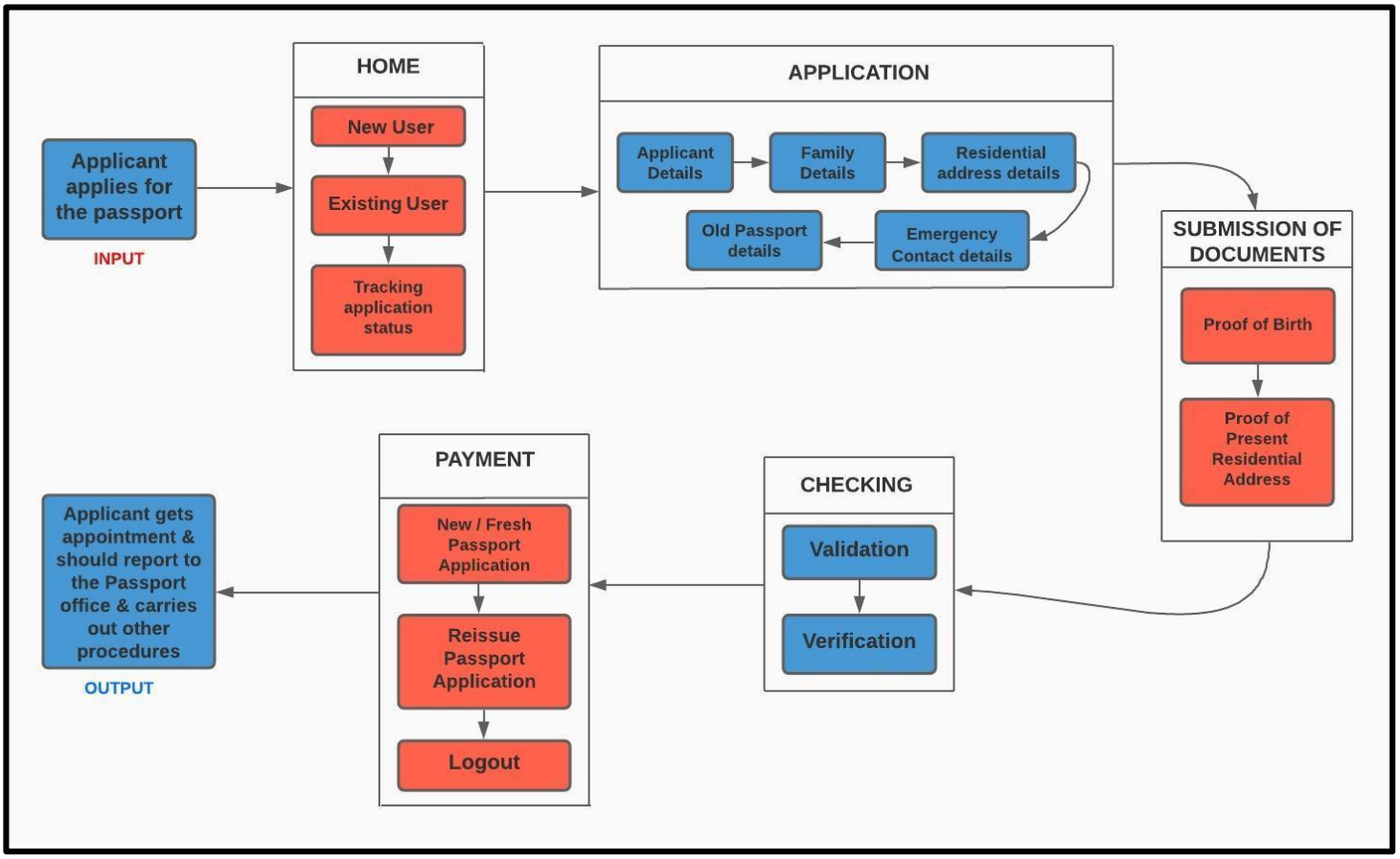
7.1 DESIGN

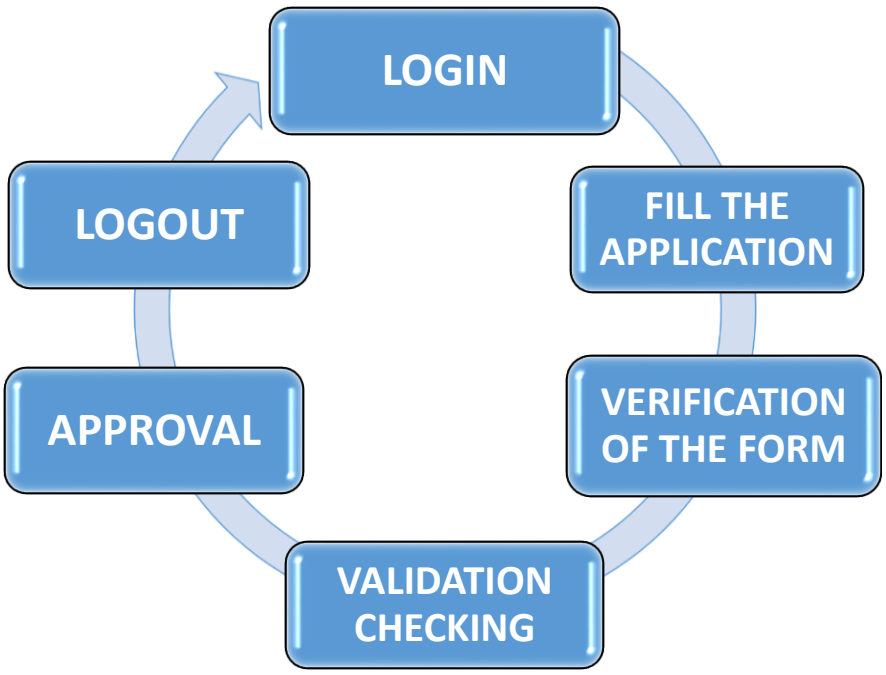
DATE	15/02/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

7.1.1 SYSTEM ARCHITECTURE

System Architecture is an abstract, conceptualization-oriented, global, and focused to achieve the mission and life cycle concepts of the system.

The purpose of system architecture is to define a comprehensive solution based on principles, concepts, and properties logically related to and consistent with each other.





SYSTEM DESIGN

8.1 MODELLING UML USE CASE DIAGRAM & CAPTURING USE CASE SCENARIOS

DATE	**22/02/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

8.1.1 USE CASE DESCRIPTION

USE CASE DIAGRAM:

The purpose of a use case diagram in UML is to demonstrate the different ways that a user might interact with a system.

USE CASE SYMBOLS AND NOTATION:

The notation for a use case diagram is pretty straightforward and doesn't involve as many types of symbols as other UML diagrams.

1] SYSTEM:

A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.

2] USE CASES:

Horizontally shaped ovals that represents an action which accomplishes some sort of task within the system.

3] ACTORS:

Stick figures that represent the people actually employing the use cases. It should be placed outside the system.

There are two types of Actors namely:

- **PRIMARY ACTOR:** Initiates the use of the system. It should be placed on the left side of the system.
- **SECONDARY ACTOR:** It is more reactionary and should be placed on the right side of the system.

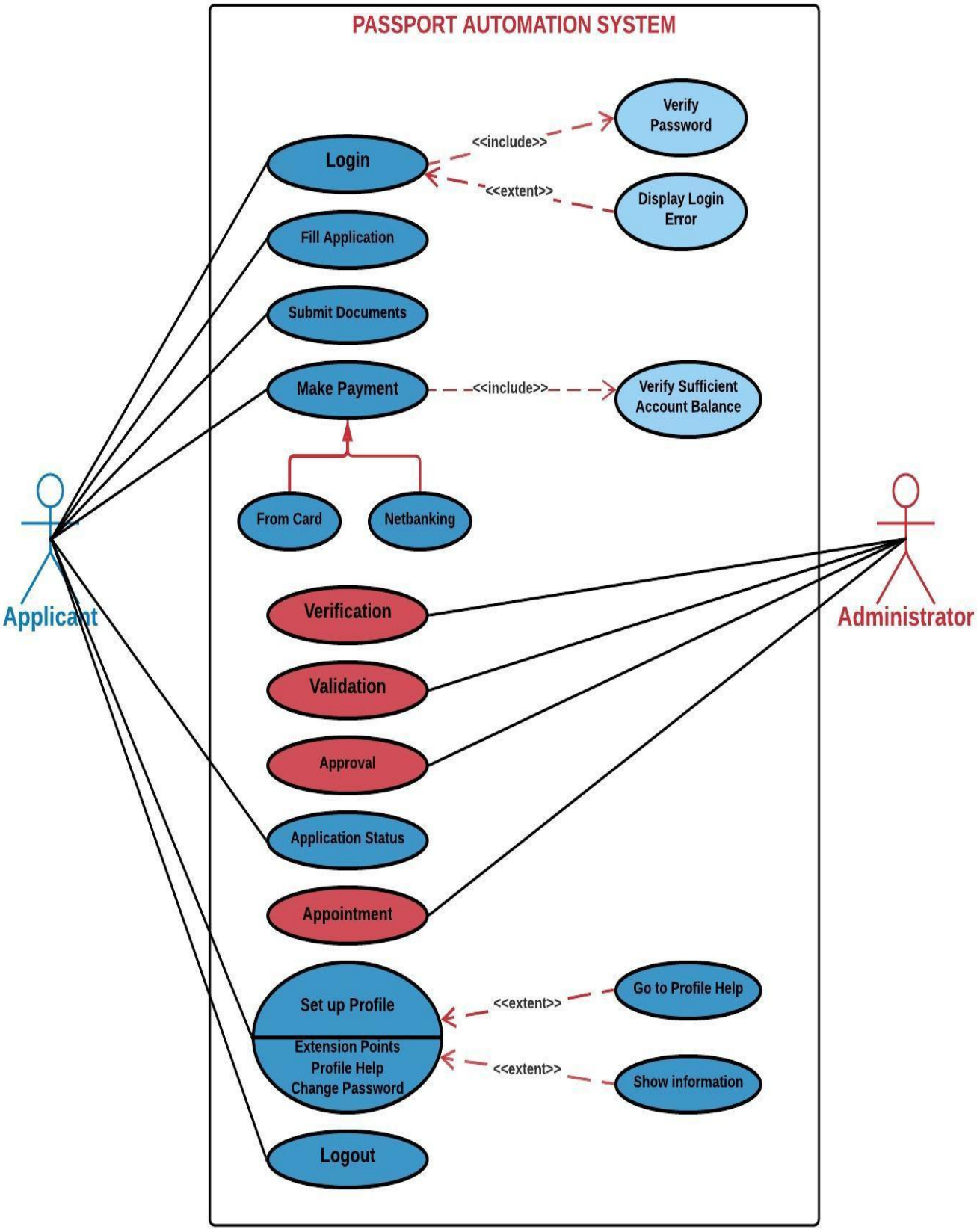
4] RELATIONSHIPS:

- **INCLUDE:** This shows the dependency between base and included use case (it happens every time).
- **EXTENT:** This happens only when certain criteria is met.

5] ASSOCIATION:

A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.

8.1.2 USE CASE DIAGRAM



9.1 ER MODELLING FROM THE PROBLEM STATEMENT

DATE	**01/03/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

9.1.1 ER MODELLING DESCRIPTION

ENTITY RELATIONSHIP DIAGRAM:

- An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.
- ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research.

USES OF ER DIAGRAM:

- ✓ Database design
- ✓ Database troubleshooting
- ✓ Business information systems
- ✓ Business process re-engineering (BPR)
- ✓ Education
- ✓ Research

COMPONENTS OF ER DIAGRAM:

ER Diagrams are composed of entities, relationships (Cardinality) and attributes. They also depict cardinality, which defines relationships in terms of numbers.

1] ENTITY:

A definable thing—such as a person, object, concept or event—that can have data stored in it.

2] ATTRIBUTES:

A property or characteristic of an entity.

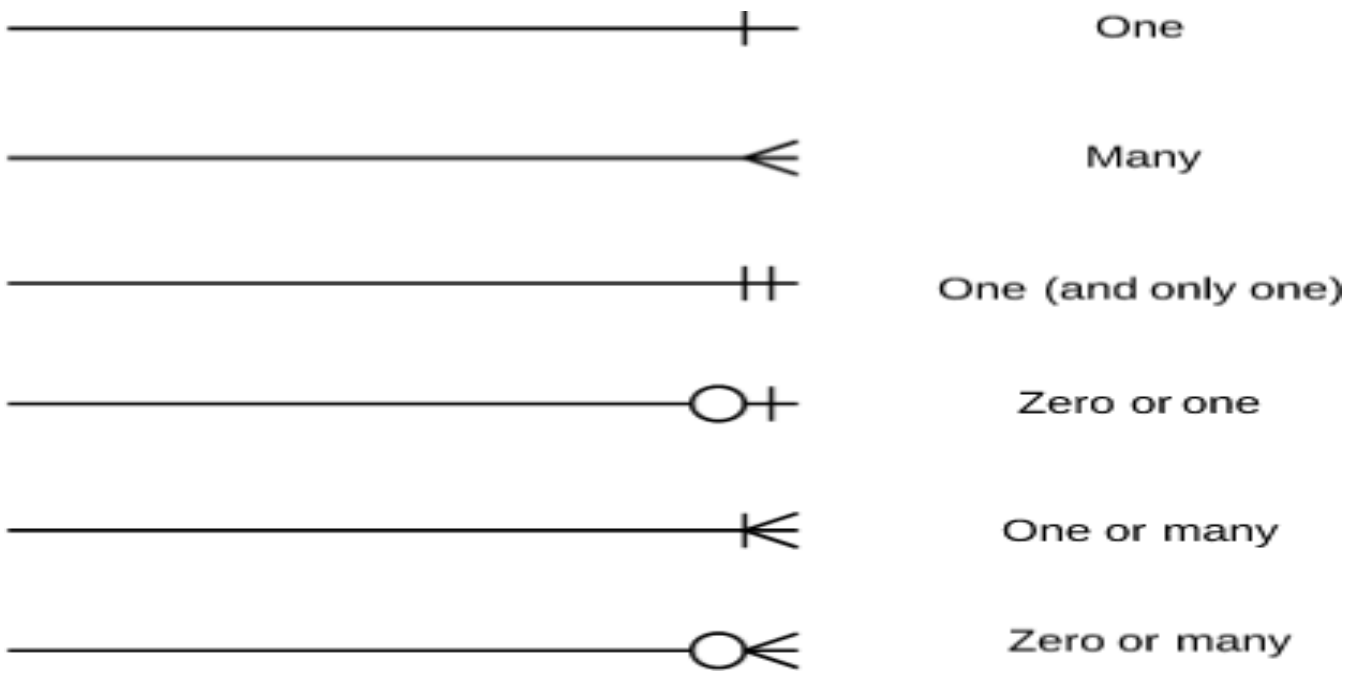
3] KEYS:

PRIMARY KEY(PK): It is unique, cannot be repeated and never null.

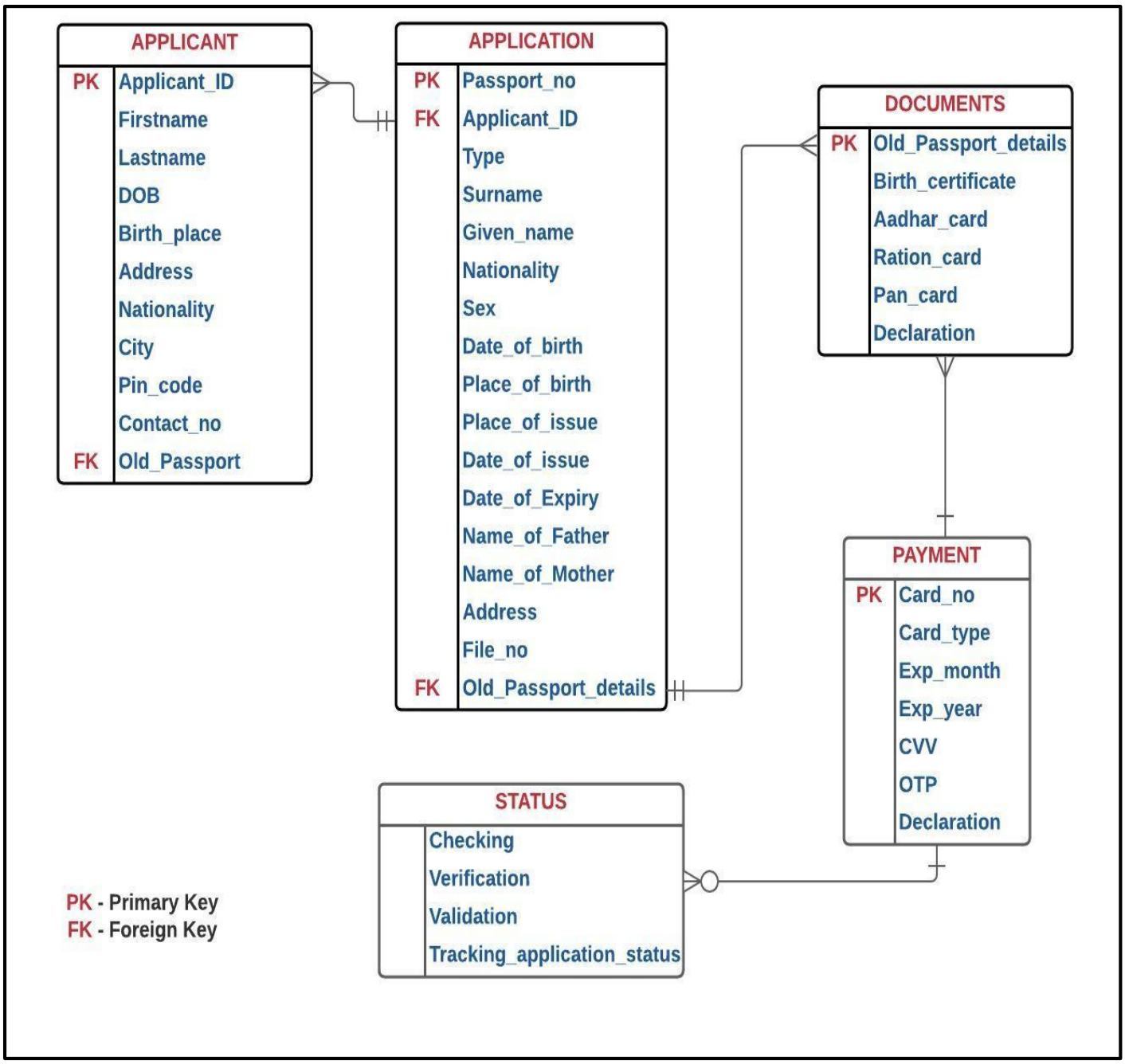
FOREIGN KEY(FK): It is not unique and can be repeated.

4] CARDINALITY:

Defines the numerical attributes of the relationship between two entities.



9.1.2 ER DIAGRAM



In this diagram,

Applicant, Application, Documents, Payment, Status are the **Entities.**

10.1 IDENTIFYING DOMAIN CLASSES FROM THE PROBLEM STATEMENTS

DATE	08/03/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

10.1.1 DOMAIN CLASS DIAGRAM DESCRIPTION

DOMAIN CLASS DIAGRAM:

Domain modeling is a technique used to understand the project problem description and to translate the requirements of that project into software components of a solution.

DOMAIN CLASSES AND OBJECTS:

Objects that represent domain entities are called **entities or domain objects**. The classes they instantiate are called **domain classes**.

IDENTIFYING THE STEREOTYPES:

<<thing>>: An entity that has mass and volume

<<event>>: An entity that has a start time and duration

<<role>>: An entity that executes tasks.

<<type>>: an entity that describes other entities.

IDENTIFYING ATTRIBUTES:

A class contains definitions of all of the attributes its instances will contain.

An attribute has four attributes:

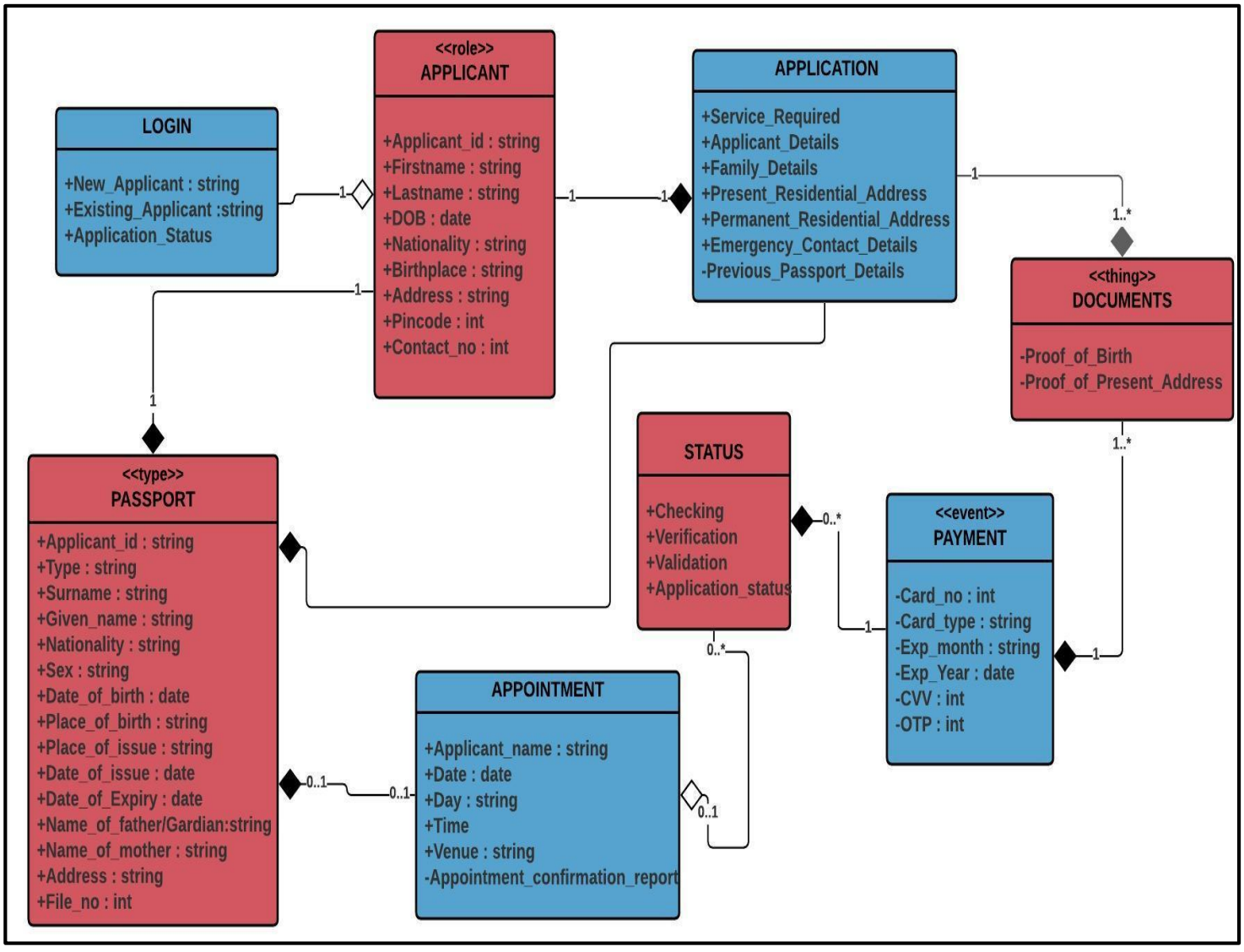
1] Name

2] Type

3] Visibility

- Private (-)
- Public (+)
- Protected (#)
- Package/Default (~)

4] Initial value (optional)



11.1 STATECHART & COMMUNICATION MODELLING

DATE	15/03/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

11.1.1 STATECHART DIAGRAM DESCRIPTION

STATECHART DIAGRAM:

A state diagram, sometimes known as a state machine diagram, is a type of behavioral diagram in the Unified Modeling Language (UML) that shows the transitions between various objects.

COMPONENTS OF STATECHART DIAGRAM:

We can include many different shapes in a state diagram, particularly if we choose to combine it with another diagram. This list summarizes the most common shapes we may encounter.

1] START STATE:

It is denoted by black filled circle and represents the beginning of the state.

2] STATE:

We use a rounded rectangle to represent a state. A state represents the conditions or circumstances of an object of a class at an instant of time.

3] TRANSITION:

We use a solid arrow to represent the transition or change of control from one state to another. The arrow is labelled with the event which causes the change in state.

4] FORK:

We use a rounded solid rectangular bar to represent a Fork notation with incoming arrow from the parent state and outgoing arrows towards the newly created states. We use the fork notation to represent a state splitting into two or more concurrent states.

5] JOIN:

We use a rounded solid rectangular bar to represent a Join notation with incoming arrows from the joining states and outgoing arrow towards the common goal state. We use the join notation when two or more states concurrently converge into one on the occurrence of an event or events.

6] DECISION:

We use a diamond symbol to apply a condition wherever necessary.

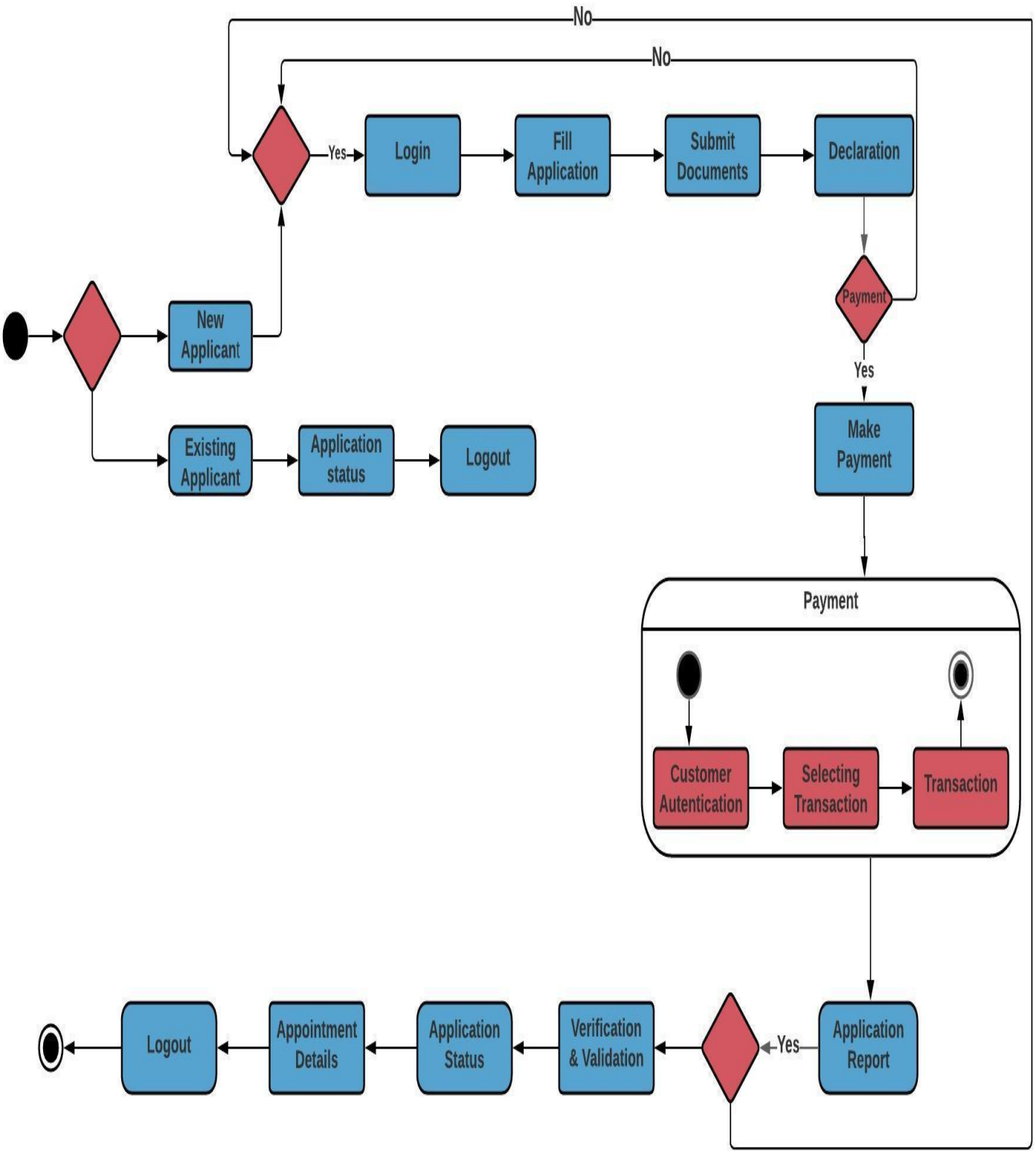
7] COMPOSITE STATE:

We use a rounded rectangle to represent a composite state also. We represent a state with internal activities using a composite state.

8] FINAL STATE:

We use a filled circle within a circle notation to represent the final state in a state machine diagram.

11.1.2 STARTCHART DIAGRAM



11.1.3 COMMUNICATION DIAGRAM DESCRIPTION

COMMUNICATION DIAGRAM:

Communication diagrams, formerly known as collaboration diagrams, are almost identical to sequence diagrams in UML, but they focus more on the relationships of objects—how they associate and connect through messages in a sequence rather than interactions.

COMPONENTS OF COMMUNICATION DIAGRAM:

1] OBJECTS:

Objects can be classed as either a supplier or a client. Suppliers call the function that supplies the message. Clients send the message to the supplier, who receives it. It is represented by rounded rectangle.

2] ACTORS:

Stick figure represents the actor. It is the instances that invokes the interaction. Each actor has a specific name and a role.

3] LINKS:

A straight line connecting two objects indicates a relationship between them. The two objects are able to send messages to each other.

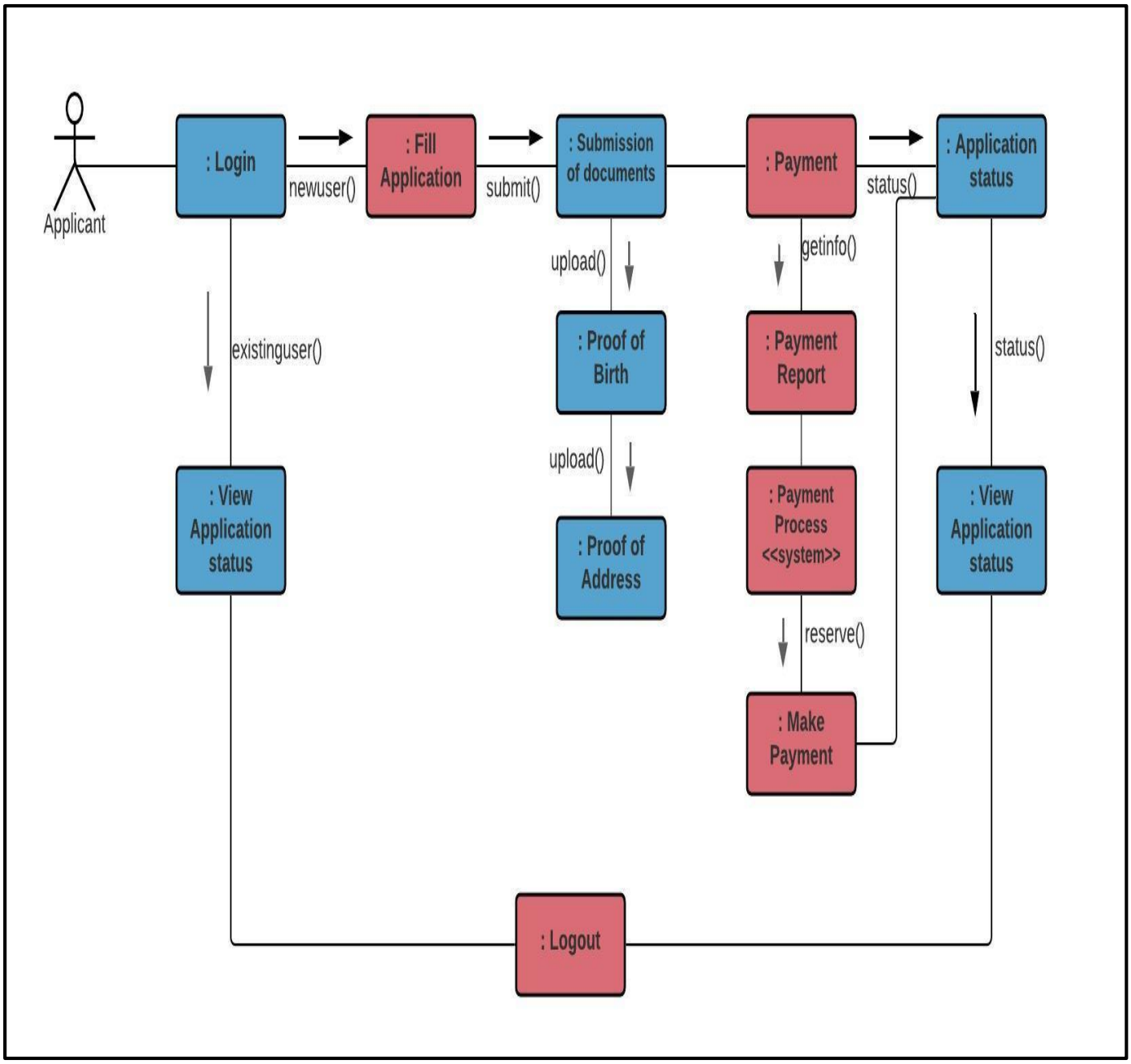
4] MESSAGES:

Typically, messages will have a number and description next to them. The number determines the order in which messages should be read.

NOTE:

Communication diagrams shows much more than just a sequence of events. Using a communication diagram to model your system allows you to breakdown a series of complex interactions in said system.

11.1.4 COMMUNICATION DIAGRAM



12.1 MODELLING UML CLASS AND SEQUENCE DIAGRAM

DATE	**22/03/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

12.1.1 CLASS DIAGRAM DESCRIPTION

CLASS DIAGRAM:

Class diagrams are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modeling its classes, attributes, operations, and relationships between objects.

COMPONENTS OF CLASS DIAGRAM:

The standard class diagram is composed of three sections:

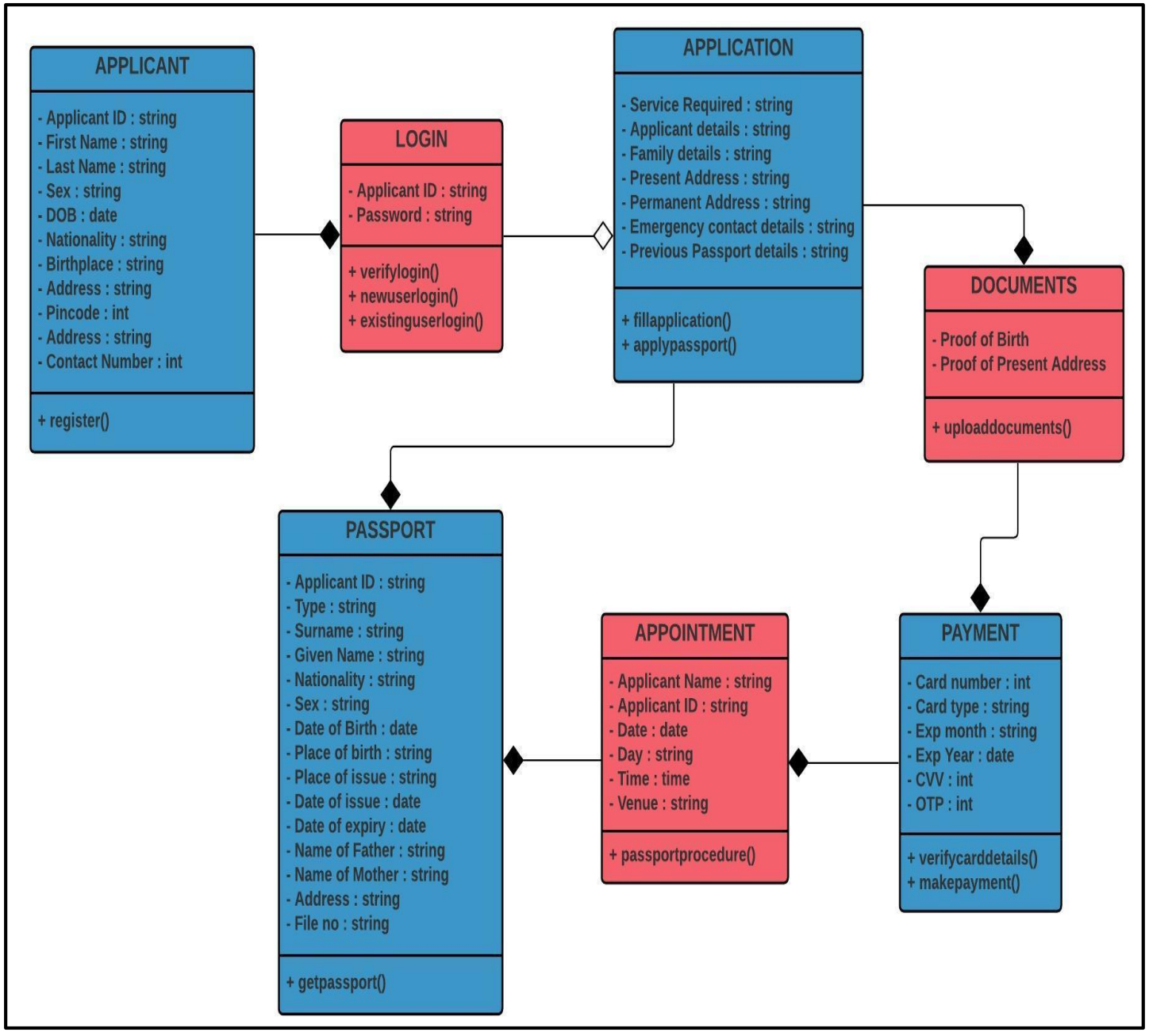
- **UPPER SECTION:** Contains the name of the class. This section is always required, whether you are talking about the classifier or an object.
- **MIDDLE SECTION:** Contains the attributes of the class. Use this section to describe the qualities of the class. This is only required when describing a specific instance of a class.
- **BOTTOM SECTION:** Includes class operations (methods). Displayed in list format, each operation takes up its own line. The operations describe how a class interacts with data.

MEMBER ACCESS SPECIFIERS:

All classes have different access levels depending on the access modifier (visibility). Here are the access levels with their corresponding symbols.

- Public (+)
- Private (-)
- Protected (#)
- Package (~)

12.1.2 CLASS DIAGRAM



12.1.3 SEQUENCE DIAGRAM DESCRIPTION

SEQUENCE DIAGRAM:

Sequence diagram are a popular dynamic modeling solution in UML because they specifically focus on lifelines, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends.

COMPONENTS IN SEQUENCE DIAGRAM:

1] ACTOR:

Stick figure represents the actor. Shows entities that interact the external objects of the system.

2] OBJECTS:

Rectangular boxes represent the object, demonstrates how an object will behave in the context of the system.

3] ACTIVATION BOXES:

Represents the time needed for an object to complete a task. The longer the task will take, the longer the activation box becomes.

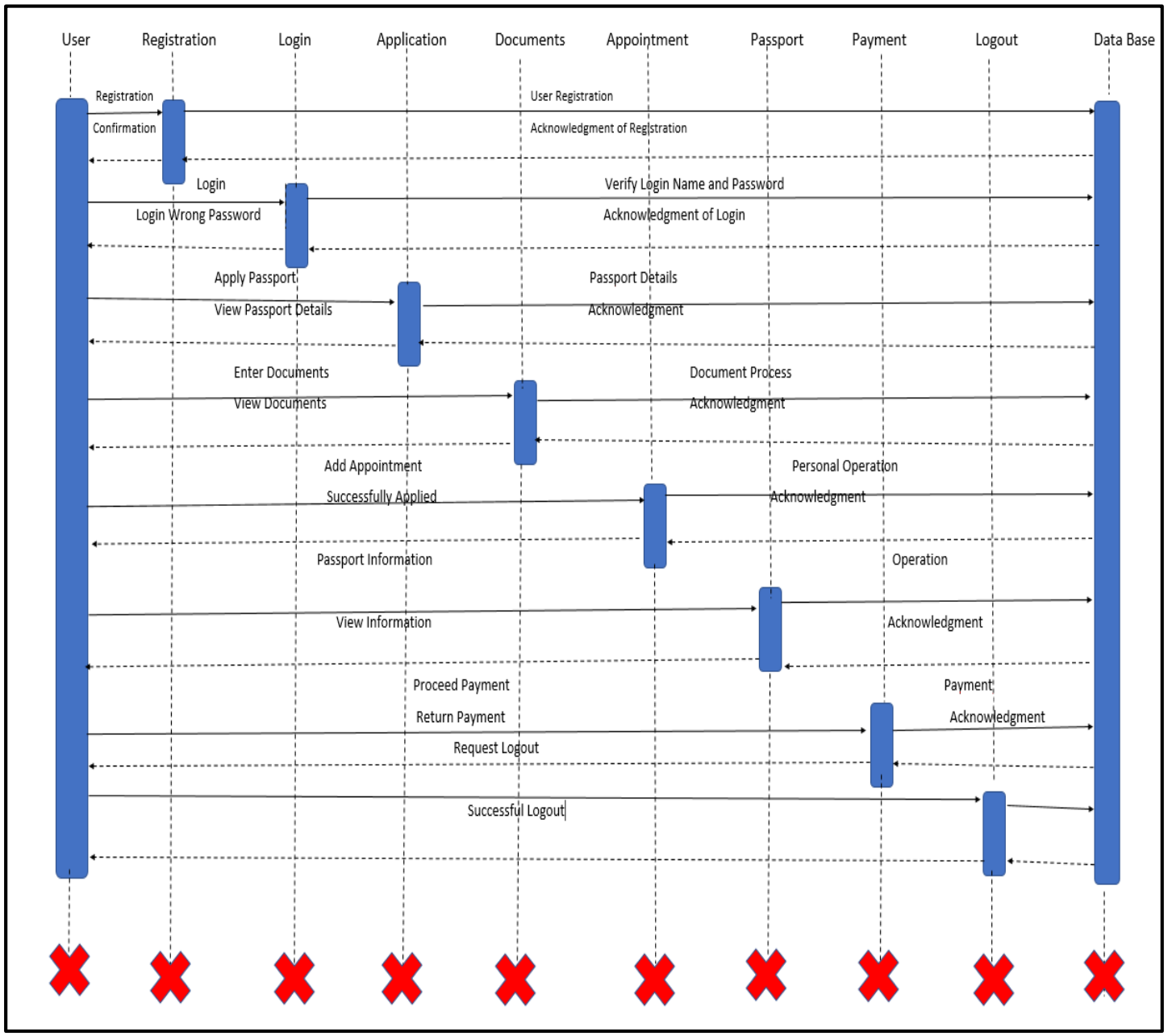
4] MESSAGE SYMBOLS:

We use the following arrows and message symbols to show how information is transmitted between objects. These symbols may reflect the start and execution of an operation or the sending and reception of a signal.

- **SYNCHRONOUS MESSAGE:** Represented by a solid line with a solid arrowhead. This symbol is used when a sender must wait for a response to a message before it continues. The diagram should show both the call and the reply.

- **ASYNCHRONOUS MESSAGE:** Represented by a solid line with a lined arrowhead. Asynchronous messages don't require a response before the sender continues. Only the call should be included in the diagram.
- **REPLY MESSAGE:** Represented by a dashed line with a lined arrowhead, these messages are replies to calls.
- **DELETE MESSAGE:** Represented by a solid line with a solid arrowhead, followed by an X. This message destroys an object.

12.1.4 SEQUENCE DIAGRAM



13.1 MODELLING DATA FLOW DIAGRAM

DATE	**29/03/2021**
SUBMITTED BY	**Anaiappan R (301)** **Joeal Chrish H(335)** **Selin Riona V (350)**
TITLE/ROLE	**PASSPORT AUTOMATION SYSTEM**

13.1.1 DATA FLOW DIAGRAM DESCRIPTION

DATA FLOW DIAGRAM:

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.

COMPONENTS OF DATA FLOW:

There are 4 basic symbols that are used to represent a data-flow diagram.

1] PROCESS:

Rounded rectangle represents the process, which receives input data and produces output with a different content or form. Processes can be as simple as collecting input data and saving in the database, or it can be complex as producing a report. Every process has a name that identifies the function it performs.

2] DATA FLOW:

A data-flow is a path for data to move from one part of the information system to another.

- Straight lines with incoming arrows are input data flow.
- Straight lines with outgoing arrows are output data flow.

3] DATA STORE:

A data store or data repository is used in a data-flow diagram to represent a situation when the system must retain data because one or more processes need to use the stored data in a later time.

- Data can be written into the data store, which is depicted by an outgoing arrow.
- Data can be read from a data store, which is depicted by an incoming arrow.

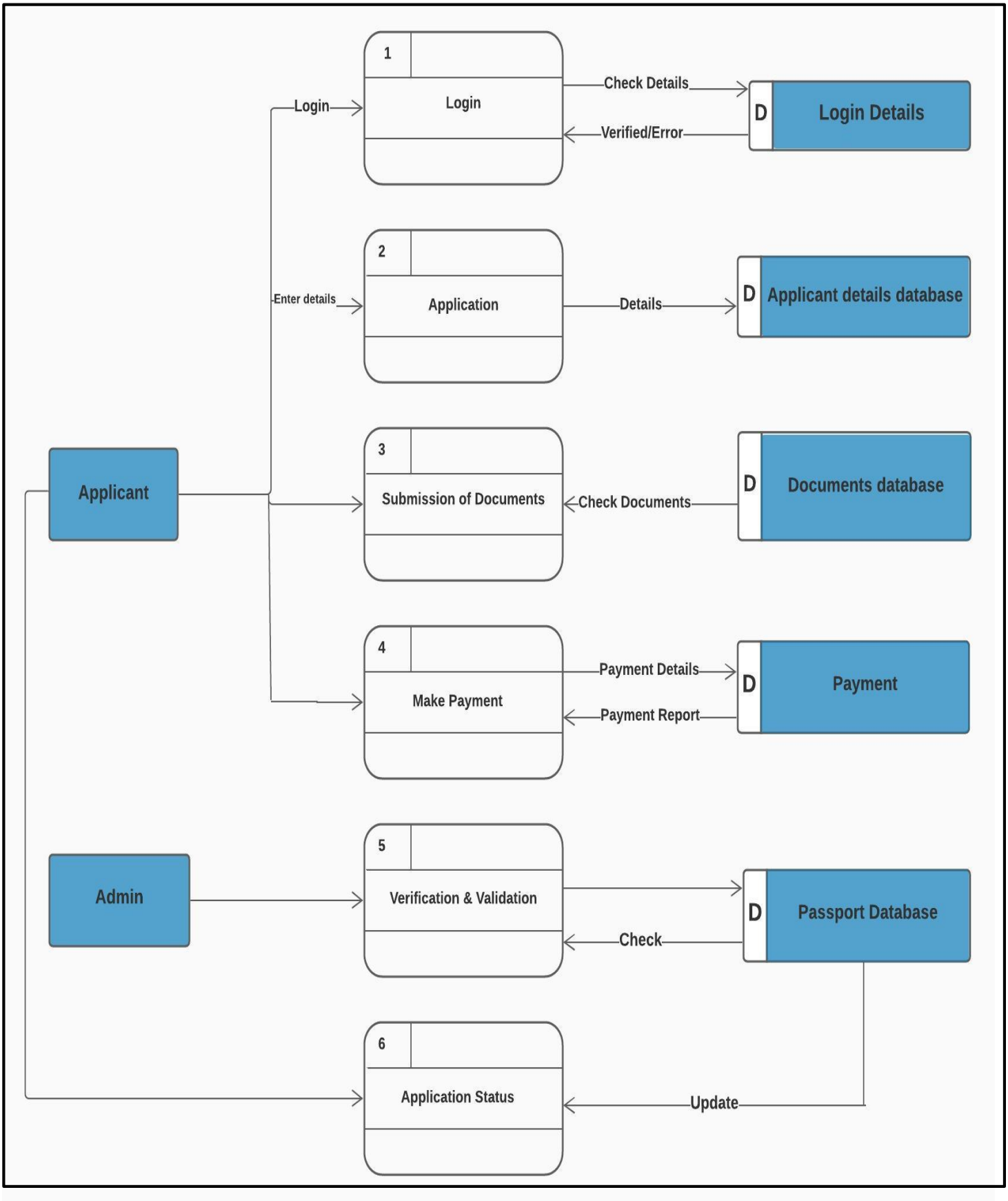
4] EXTERNAL ENTITY:

A rectangle represents an external entity. They are components outside of the boundaries of the information systems which either supply or receive data but does not process data. They represent how the information system interacts with the outside world.

External entities also are called **terminators** because they are data origins or final destinations.

An external entity must be connected to a process through a data-flow.

13.1.2 DATA FLOW DIAGRAM

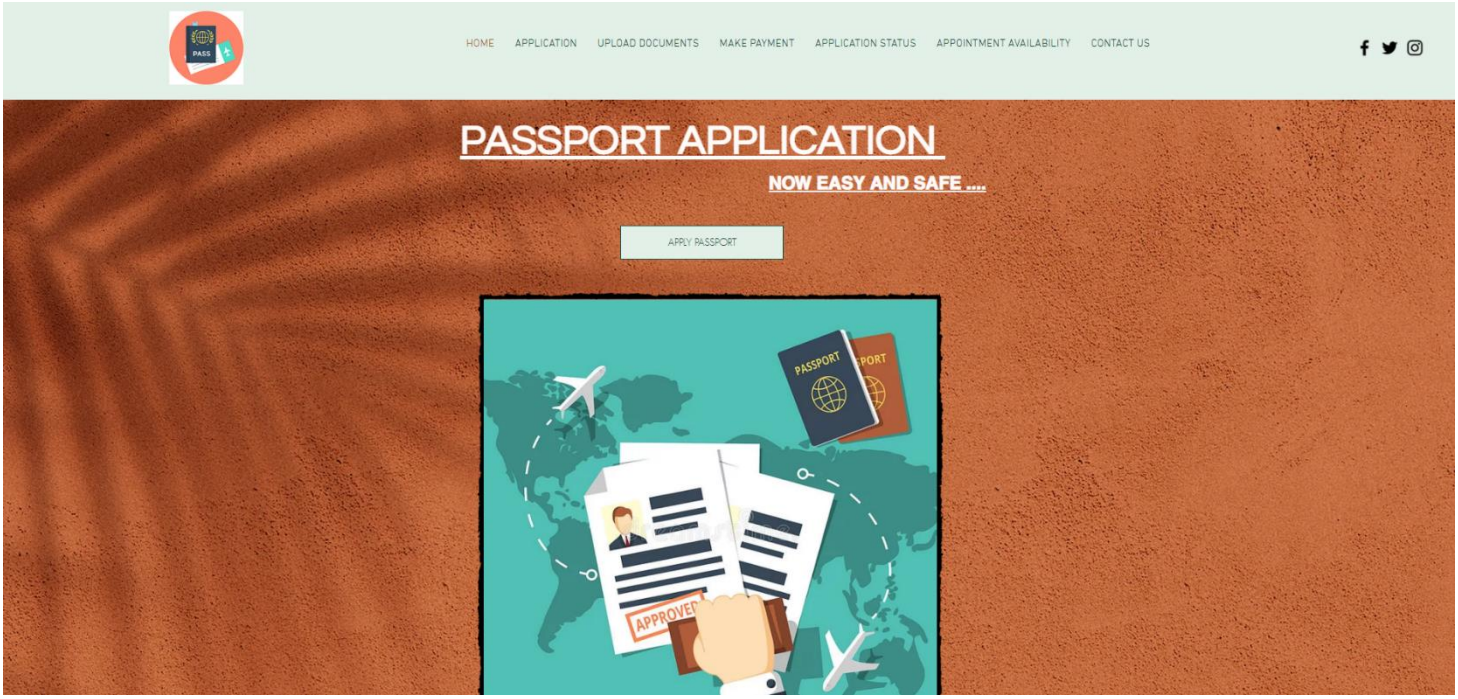


14.1 IMPLEMENTATION

DATE	05/04/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

14.1.1 MODULE IMPLEMENTATION

1] HOME PAGE:



2] APPLICATION:

SERVICE REQUIRED

* Applying for
- Select -

* Type of Application
- Select -

* Type of Passport Booklet
- Select -

SUBMIT

FAMILY AND CONTACT DETAILS

Father's Name

Mother's Name

Legal Guardian's name

* Applicant's Mail ID

* Applicant's Mobile number

SUBMIT

Emergency Contact Details

* Name

* Address

* Street Address

Address line 2

* City

* State

* Postal / Zip Code

* Country
- *Country -

* Email ID

* Mobile Number

SUBMIT

APPLICANT DETAILS

* First Name

* Last Name

* Surname

* Gender
Male Female Other

* DOB

* Nationality

SUBMIT

PERMANENT RESIDENTIAL ADDRESS

* Is your present address out of India ?
Yes No

* Present Address

* Street Address

Address line 2

* City

* State

* Postal / Zip Code

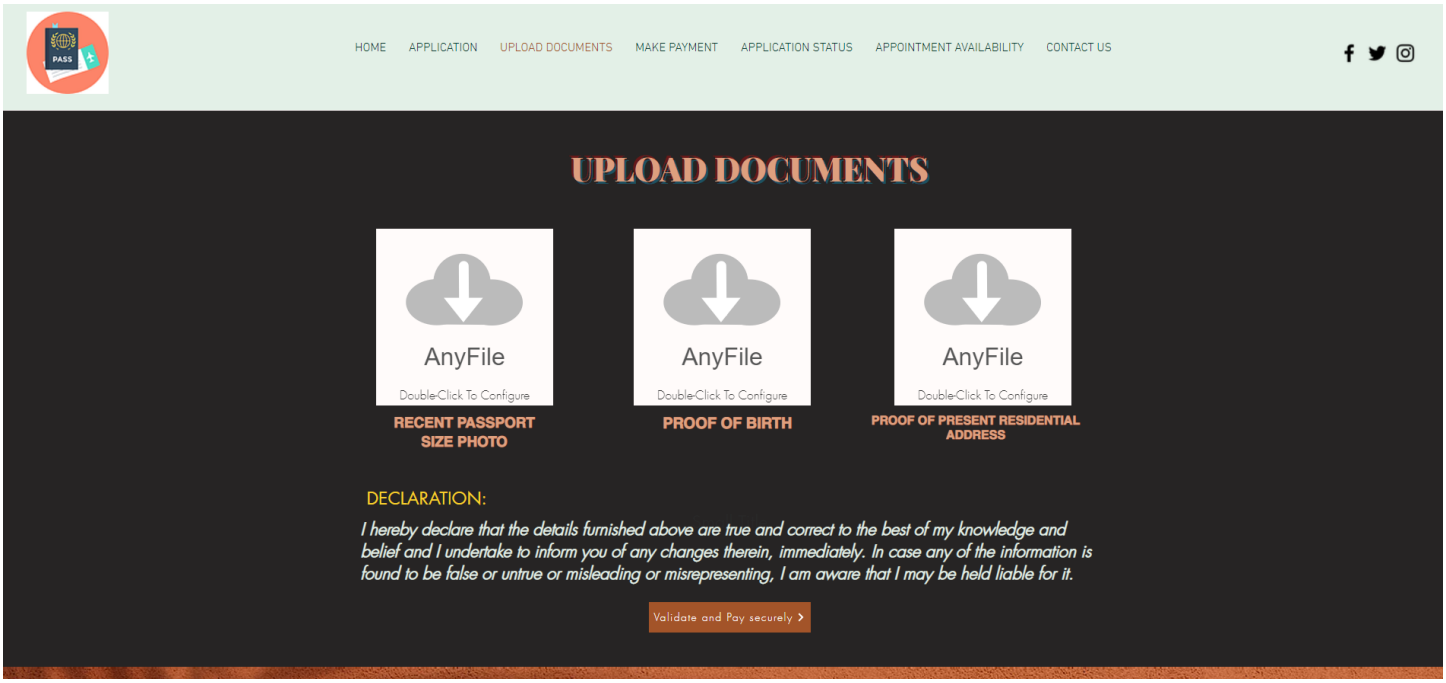
* Country
- *Country -

* Police Station

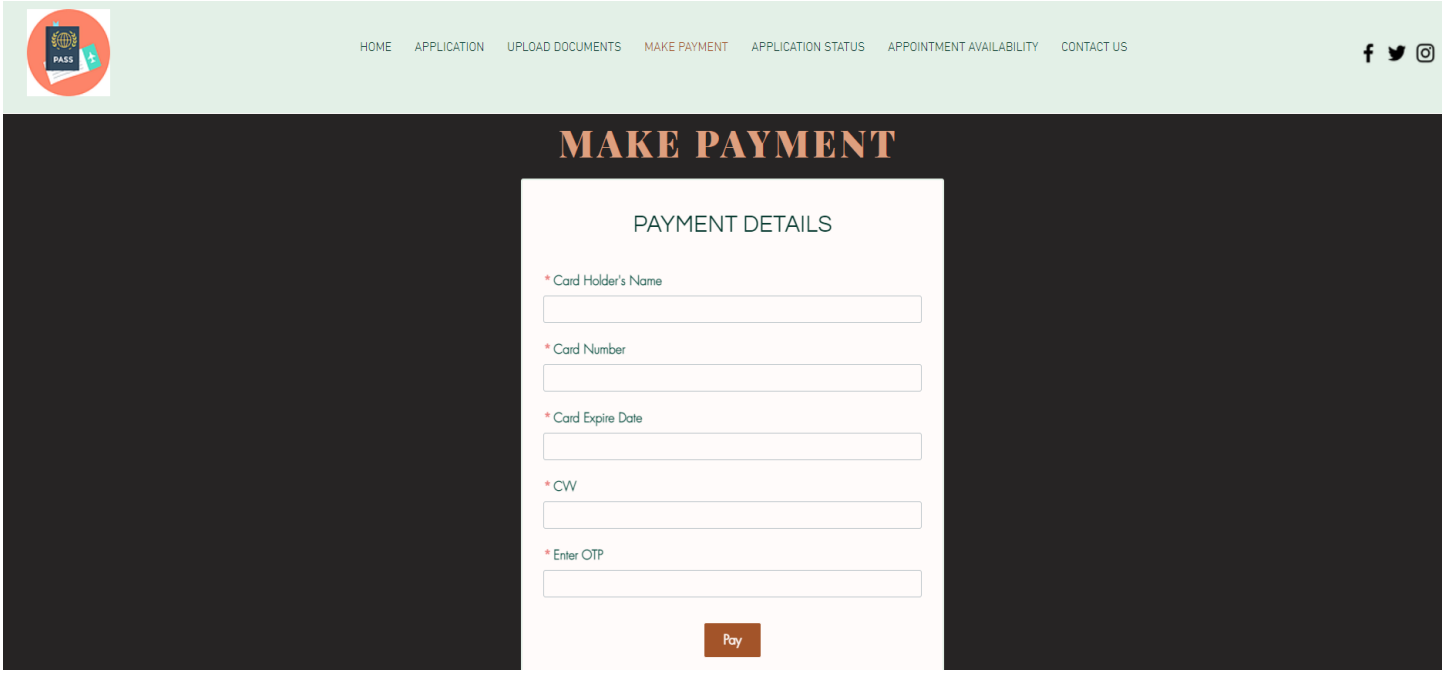
* Do you have a Permanent Address ?
- Select -

SUBMIT

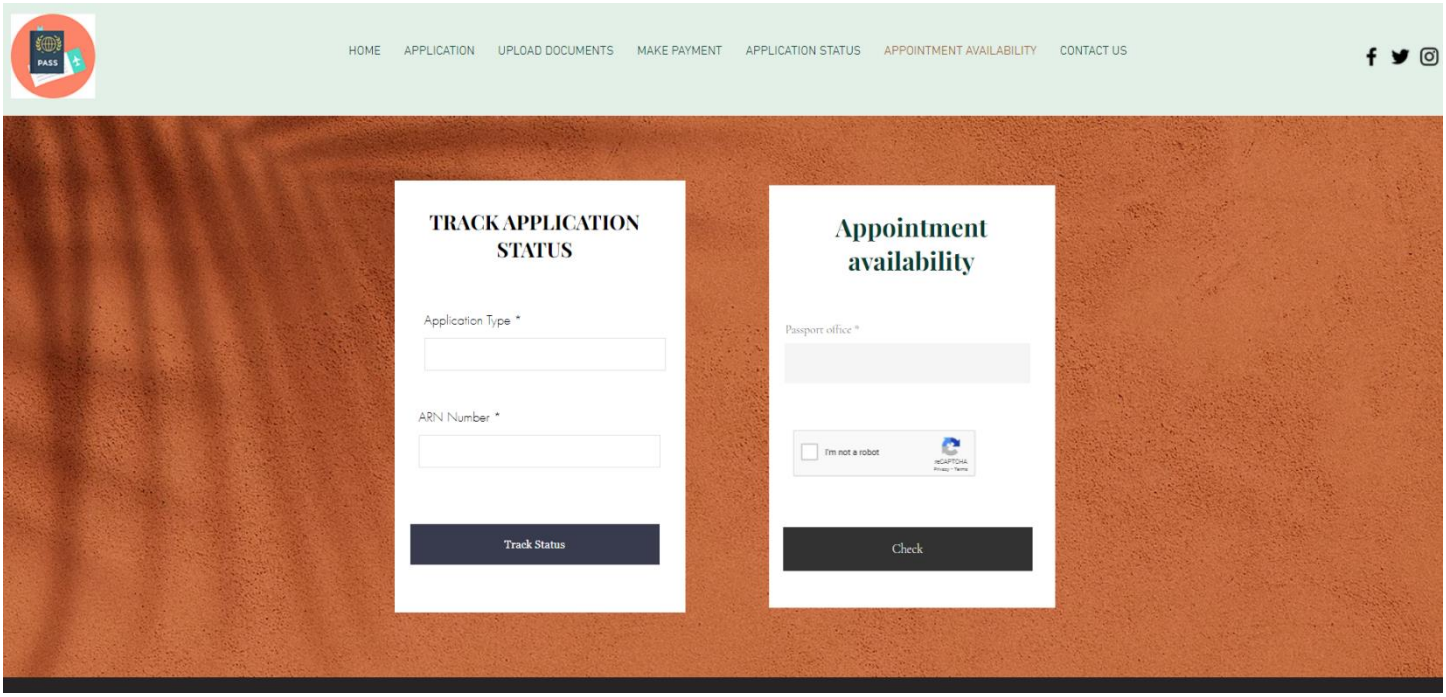
3] UPLOAD DOCUMENTS:



4] MAKE PAYMENT:



5] APPLICATION STATUS & APPOINTMENT AVAILABILITY:



6] CONTACT US:

15.1 ESTIMATION OF TEST COVERAGE METRICES & STRUCTURAL COMPLEXITY

DATE	12/04/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

15.1.1 CONTROL FLOW GRAPH

A Control Flow Graph (CFG) is the graphical representation of control flow or computation during the execution of programs or applications.

There exist 2 designated blocks in Control Flow Graph:

1] Entry Block: It allows the control to enter into the control flow graph.

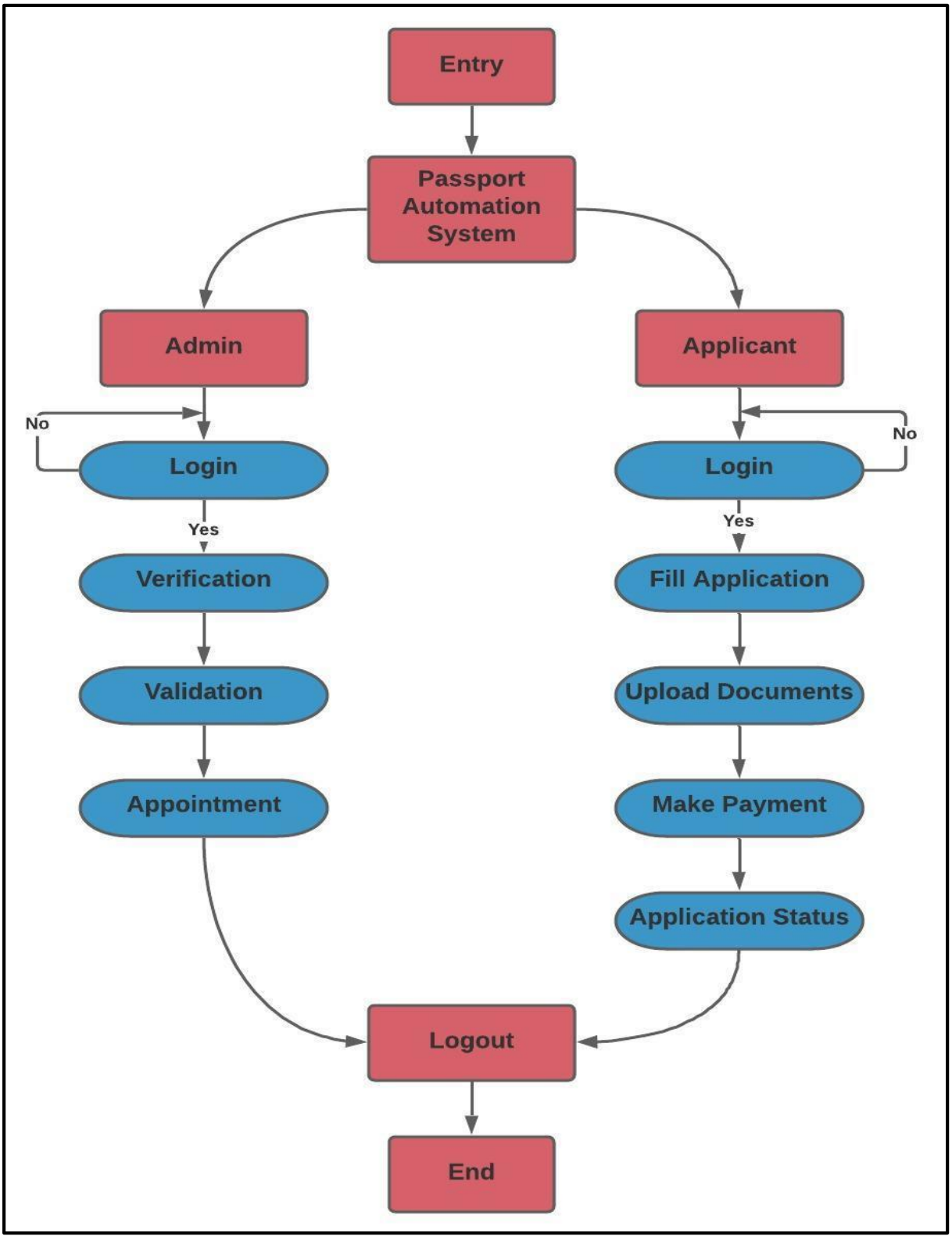
2] Exit Block: Control flow leaves through the exit block.

CHARACTERISTICS OF CFG:

- Control flow graph is process oriented.
- Control flow graph shows all the paths that can be traversed during a program execution.
- Control flow graph is a directed graph.
- Edges in CFG portray control flow paths and the nodes in CFG portray basic blocks.

ADVANTAGE OF CFG:

- It can easily encapsulate the information per each basic block.
- It can easily locate inaccessible codes of a program and syntactic structures such as loops are easy to find in a control flow graph.



15.1.2 McCABE's CYCLOMATIC COMPLEXITY

- McCabe's Cyclomatic Complexity is a software quality metric that quantifies the complexity of a software program.
- Complexity is inferred by measuring the number of linearly independent paths through the program.
- The higher the number the more complex the code.

SIGNIFICANCE OF McCABE'S NUMBER:

- Measurement of McCabe's Cyclomatic Complexity metric ensures that developers are sensitive to the fact that programs with high McCabe numbers (e.g. > 10) are likely to be difficult to understand and therefore have a higher probability of containing defects.
- The Cyclomatic Complexity number also indicates the number of test cases that would have to be written to execute all paths in a program.

MATHEMATICAL REPRESENTATION:

Cyclomatic complexity is derived from the control flow graph of a program. Mathematically, it is set of independent paths through the graph diagram. The Code complexity of the program can be defined using the formula:

$$V(G) = E - N + 2$$

Where, E - Number of edges; N - Number of Nodes

$$V(G) = P + 1$$

Where, P = Number of predicate nodes (node that contains condition)

PROPERTIES OF CYCLOMATIC COMPLEXITY:

- V (G) is the maximum number of independent paths in the graph
- V (G) >=1
- G will have one path if V (G) = 1
- Minimize complexity to 10

15.1.3 OPTIMUM VALUE OF CYCLOMATIC COMPLEXITY

There are several metrics for software quality and cyclomatic complexity is an important one.

WAYS TO USE CYCLOMATIC COMPLEXITY:

Cyclomatic complexity can be used in two ways:

- ✓ To limit code complexity.
- ✓ To determine the number of test cases required.

Cyclomatic complexity can be one of the most difficult software quality metrics to understand. And that makes it difficult to calculate.

That's why it is important to learn about software quality metrics — like cyclomatic complexity — and how to accurately measure them.

Complexity(M) = Edges(E) – Nodes(N) + Exit Nodes(P)

OPTIMUM VALUES:

COMPLEXITY	TYPE OF MODULE	RISK LEVEL
1 - 4	Simple	Low
5 - 10	Marginal	Low
11 - 20	Complex	Moderate
21 - 50	Complex	High
>50	Untestable	Very High

16.1 DESIGN TESTING SUITES

DATE	19/04/2021
SUBMITTED BY	Anaiappan R (301) Joeal Chrish H(335) Selin Riona V (350)
TITLE/ROLE	PASSPORT AUTOMATION SYSTEM

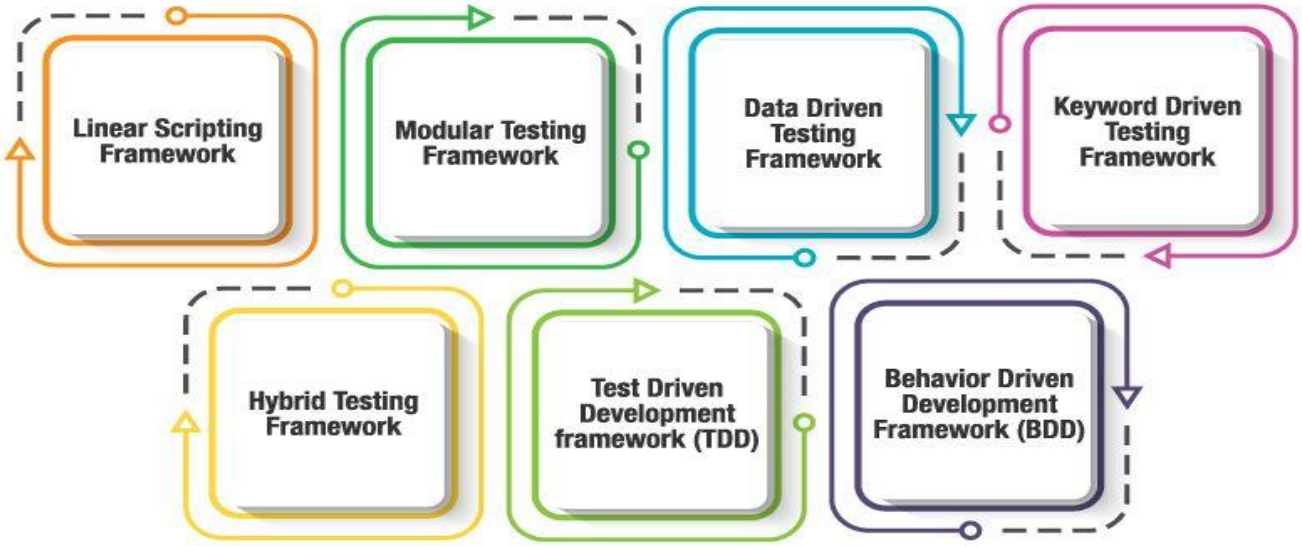
16.1.1 SOFTWARE TESTING

Software testing can be stated as the process of verifying and validating that a software or application is bug free, meets the technical requirements as guided by its design and development and meets the user requirements effectively and efficiently with handling all the exceptional and boundary cases.

TYPES OF SOFTWARE TESTING:

- **MANUAL TESTING:** It includes testing a software manually. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug.
- **AUTOMATION TESTING:** It is when the tester writes scripts and uses another software to test the product. It is used to re-run the test scenarios that were performed manually, quickly, and repeatedly.

16.1.2 TESTING FRAMEWORKS



MODULE BASED TESTING FRAMEWORK:

- Abstraction is the concept on which this framework is built. Based on the modules, independent test scripts are developed to test the software.
- Specifically, an abstraction layer is built for the components to be hidden from the application under test.
- This sort of abstraction concept ensures that changes made to the other part of the application does not affect the underlying components.

KEYWORD DRIVEN TESTING FRAMEWORK:

- It is an application independent framework and uses data tables and keywords to explain the actions to be performed on the application under test.
- This is more so called as keyword driven test automation framework for web based applications and can be stated as an extension of data driven testing framework.

16.1.3 MASTER TEST PLAN

TESTING OBJECTIVE	FOCUSING ON PERFORMANCE ISSUE
Test Items	Login system, Registration system, Uploading documents, Payment System
Features to be tested	Login verification, Registration feature, Uploading documents feature, Payment feature
Features not to be tested	Database Connectivity, Payment verifier, Two way pinging tool
Approach	Method – Manual Testing
Required Hardware/Software	A PC with 8 GB RAM, Internet Connectivity
Risks	Instability of the product
Testers & Schedule	Tester: SELIN RIONA V Scheduling Information : 25th April 2021, 3:00 PM
Estimate	Rs500/- (Excluding Tax and other charges)

16.1.4 MANUAL TESTING

TEST AREA	INPUT	TEST DESCRIPTION	OUTPUT/RESULT
Login Module	User Name and Password	Permits the user to enter into the application	tested
Application Module	Fill the Application form	Allows the user to apply Passport	tested
Upload Documents Module	Upload Documents required	Allows the user to upload documents	tested
Payment Module	Click validate and pay option	Checks whether the payment feature is functioning and secure	tested

17.1 CONCLUSION

Thus, the above project for the **"Passport Automation System"** has been successfully created.

BENEFITS OF PASSPORT AUTOMATION SYSTEM:

- ✓ The System provides an online interface to the user where they can fill in their personal details.
- ✓ The authority concerned with the issue of passport can use this system to reduce his workload and process the application in a speedy manner.
- ✓ Provide a communication platform between the applicant and the administrator.
- ✓ This system aims at improving the efficiency in the issue of the Passport, reduces the complexities involved in it and for the effective dispatch of the passport to the maximum possible extent.

FUTURE SCOPE:

Enter details and submit the necessary documents (may be by scanning). workload and process the application in a speedy manner. verification of applicant's information.

We finally conclude this project by conveying our sincere thanks to everyone who has supported us during the course of this project.

18.1 REFERENCES

- www.google.com
- www.wikipedia.com
- https://www.passportindia.gov.in/AppOnlineProject/welcomeLink#
- https://passportsindiaonline.com/reissue-passport.php
- www.youtube.com
- https://www.vidyarthiplus.com/vp/attachment.php?aid=24303