

A Software Requirements Specification

On

Verification of Passengers at Airport Entry
Gate

Submitted in Partial Fulfillment for the Award of Degree of Bachelor of
Technology in Information Technology from Rajasthan Technical University,
Kota



PROJECT MENTOR:

Dr. Mehul Mahrishi

Associate Professor

SUBMITTED BY:

Shubham Agarwal(19ESKIT089)

Tanmay Shah (19ESKIT094)

Vimal Sharma (19ESKIT097)

COORDINATOR:

Dr. Sushila Vishnoi

Associate Professor

DEPARTMENT OF INFORMATION TECHNOLOGY
SWAMI KESHWANAND INSTITUTE OF TECHNOLOGY,
MANAGEMENT & GRAMOTHAN
Ramnagar (Jagatpura), Jaipur – 302017

SESSION 2022-23

Table of Contents

1. Introduction.....	3
1.1. Methodology.....	3
1.2. Purpose.....	5
1.3. Scope.....	5
1.4. Definitions, Acronyms and Abbreviations.....	5
1.5. Tools Used.....	5
1.6. Technologies to be used.....	6
1.7. Overview.....	7
2. Overall Description.....	8
2.1. Product Perspective.....	8
2.2. Software Interface.....	8
2.3. Hardware Interface.....	9
2.4. Communication Interface.....	9
2.5. Constraints.....	9
2.6. E-R Diagram.....	9
2.7. Use-Case Model Survey.....	12
2.8. Database Design.....	12
3. Specific Requirements.....	14
3.1. Activity Diagrams.....	15
3.2. Sequence Diagram.....	16
References.....	18

1.Introduction:

1.1 Methodology:

1.1.1 Agile Methodology:

The Agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, team cycle through a process of planning, executing, and evaluating. Continuous collaboration is vital, both with team members and project stakeholders. Agile's four main values are:

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation.
- Responding to change over following a plan.



Fig 1.1: Phases of Agile Methodology

1.1.2 The different phases in Agile Methodology are:

- **Concept**

A product owner will determine the scope of their project. The product owner will discuss key requirements with a client and prepare documentation to outline them, including what features will be supported and the proposed end results. In the concept stage, the product owner will also estimate the time and cost of potential projects. This detailed analysis will help them to decide whether or not a project is feasible before commencing work.

- **Inception**

Once the concept is outlined. A product owner will pick the best people for the project while also providing them with the necessary tools and resources. They can then start the design process. The team will create a mock-up of the user interface and build the project architecture. The inception stage involves further input from stakeholders to fully flesh out the requirements on a diagram and determine the product functionality. Regular check-ins will help to ensure that all requirements are built into the design process.

- **Iteration**

Next up is the iteration phase, also referred to as construction. It tends to be the longest phase as the bulk of the work is carried out here. The developers will work with UX designers to combine all product requirements and customer feedback, turning the design into code. The goal is to build the bare functionality of the product by the end of the first iteration. This stage is a cornerstone of Agile software development, enabling developers to create working software quickly and make improvements to satisfy the client.

- **Release**

In this phase, the quality assurance team needs to perform some tests to ensure the software is fully functional. These Agile team members will test the system to ensure the code is clean — if potential bugs or defects are detected, the developers will address them swiftly. User training will also take place during this phase, which will require more documentation. When all of this is complete, the product's final iteration can then be released into production.

- **Maintenance**

The software will now be fully deployed and made available to customers. This action moves it into the maintenance phase. During this phase, the software development team will provide ongoing support to keep the system running smoothly and resolve any new bugs. They will also be on hand to offer additional training to users and ensure they know how to use the product. Over time, new iterations can take place to refresh the

existing product with upgrades and additional features.

●Retirement

There are two reasons why a product will enter the retirement phase: either it is being replaced with new software, or the system itself has become obsolete or incompatible with the organization over time. The software development team will first notify users that the software is being retired. If there is a replacement, the users will be migrated to the new system. Finally, the developers will carry out any remaining end-of-life activities and remove support for the existing software.

1.2 Purpose:

Verification of passengers at airport entry gate is a project that aims to automate the passenger verification process in order to make it a contact less and time saving experience for the passenger by deploying an application that allows a passenger to log in using a user ID and password and then scan a QR code provided on the e ticket issued by the airline. This allows the passenger to verify himself to the airport security agencies. Through this project we intend to develop a passenger verification system that is contact less, efficient, saves the time of a passenger and permits the airport authorities to utilize its manpower and resources for better tasks.

1.3 Scope:

- The Project use Computer Vision (OpenCV), NumPy and Pyzbar libraries of the python programming language to scan a QR code.
- The QR codes contain a unique ticket ID that is used to verify the identity of a passenger at the entry gate itself.
- Project can be used for enhancing the security infrastructure at an airport and creating a one-step security check-in system for a passenger which is time saving and contact less.
- The Project uses a central database created using MySQL to store User IDs and passwords which will be maintained by the Airports Authority of India.
- The project enables security agencies such as CISF to ensure that only authorized passengers are granted access to the airport terminal.
- It can also be used to prevent the entry of any hazardous object into the airport terminal building.

1.4 Definitions, Acronyms and Abbreviations:

●Computer Vision ()

Computer vision is a field of artificial intelligence that trains computers to interpret and understand the visual world. Using digital images from cameras and videos and deep learning models, machines can accurately identify and classify objects — and then react to what they “see.”

●NumPy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high – level mathematical functions to operate on these arrays.

•MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of “My”, the name of co-founder Michael Widener’s daughter My, and “SQL”, the abbreviation for Structured Query Language

•QR Code

A machine-readable code consisting of an array of black and white squares, typically used for storing URLs or other information for reading by the camera

1.5 Tools Used:

Application architecture – Python

•Python

Python is a high- level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically- typed and garbage-collected. It supports Multiple programming paradigms, including structured, object-oriented, and functional programming.

•Pyzbar

It is a pure Python library that reads one-dimensional barcodes and QR codes using the zbar library, an open-source software suite for reading bar codes from various sources, such as video streams, image files and raw intensity sensors.

Development tool – PyCharm IDE

PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, web, and data science development.

1.6 Technologies to be used:

•Python: Programming Language.

•Pyzbar: Python Library for reading 1D barcodes and QR codes.

•MySQL: Relational Database Management System.

1.7 Overview:

Existing System:

•Currently, passenger verification is done at the entry gate by CISF personnel and then again at the check-in counters.

Drawbacks:

• The process is very time consuming for the passenger.

- As a result, the turnaround time for flights is very high
- It is not contactless in anyway.
- Forged tickets and travel documents can be used to enter the airport illegally.

Proposed System:

- A Kiosk based system to enable a passenger to log in to register his presence at airport.
- A webcam mounted on the kiosk to help the passenger scan the QR code on e-ticket.

Our Plan:

- Register passengers on the AAI website and store their unique User IDs and passwords in a central database managed by AAI and available at every airport.
- Collect the User IDs and passwords using the User Interface running on kiosk for the passenger to log in.
- Allow the logged in user(passenger) to scan his QR code using the webcam mounted on Kiosk.
- Display “Access Granted” or “Access Denied” once a QR code is scanned.

2 Overall Description:

2.1 Product Perspective:

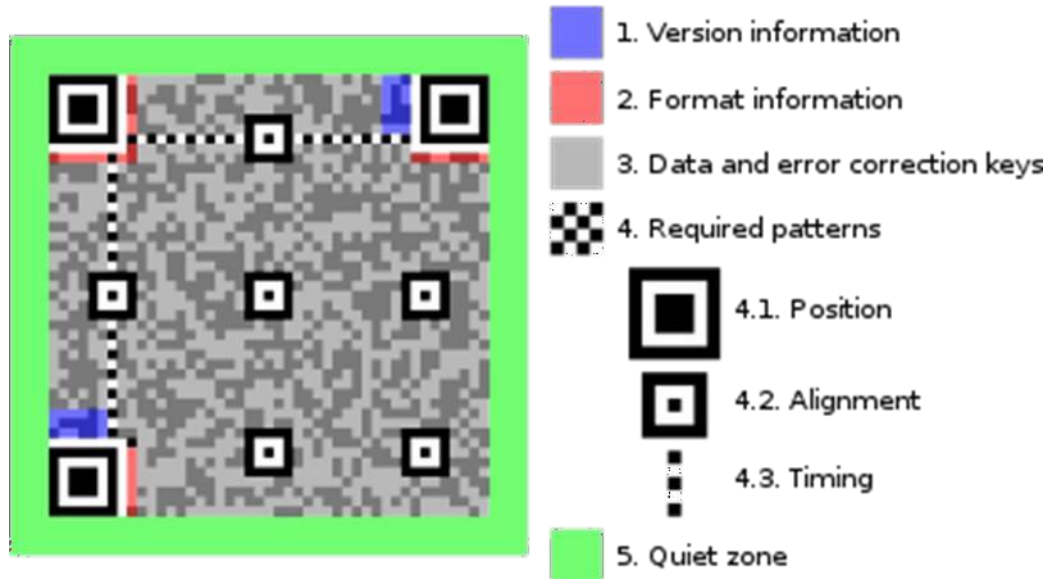


Fig 2.1: Product Perspective

2.2 Software Interface:

- Client on Intranet
User Interface, Windows OS
- Development End
Python, Pyzbar, MySQL, Windows (OS)

2.3 Hardware Interface:

Minimum Requirements:

Client Side			
	Processor	RAM	Disk Space
User Interface	SSE3 Capable Intel Pentium 4	4 GB	10 GB

Server Side			
	Processor	RAM	Disk Space
MySQL	x86 64-bit CPU	4 GB	5 GB
Python Model	x86 64-bit CPU	8 GB	5 GB (Excluding Data Size)

Recommended Requirements:

Client Side			
	Processor	RAM	Disk Space
User Interface	All Intel or AMD – 2.5 GHZ	8 MB	5 GB

Server Side			
	Processor	RAM	Disk Space
MySQL	All Intel or AMD – 2.5 GHZ	8 GB	3.5 GB
Python Model	All Intel or AMD – 2.5 GHZ	8 GB	5 GB (Excluding Data Size)

2.4 Communication Interface:

- Client (User) on Intranet will be using TCP/IP protocol.

2.5 Constraints:

- GUI is only in English.
- No restriction on number/type of characters in User ID and passwords.
- Processing is constraint to Server's Capability.
- Limited to TCP/IP.
- This system is working for single server.

2.6 E-R Diagram:

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

ER Model stands for Entity Relationship Model is a high-level conceptual data model diagram. ER model helps to systematically analyze data requirements to produce a well-designed database. The ER Model represents real-world entities and the relationships between them. Creating an ER Model in DBMS is considered as a best practice before implementing your database.

ER Modeling helps you to analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modeling before implementing your database.

Uses of ER Diagrams:

- Helps you to define terms related to entity relationship modeling
- Provide a preview of how all your tables should connect, what fields are going to be on each table
- Helps to describe entities, attributes, relationships
- ER diagrams are translatable into relational tables which allows you to build databases quickly
- ER diagrams can be used by database designers as a blueprint for implementing data in specific software applications
- The database designer gains a better understanding of the information to be contained in the database with the help of ERP diagram
- ERD Diagram allows you to communicate with the logical structure of the database to users

Facts about ER Diagram Model:

- ER model allows you to draw Database Design.
- It is an easy-to-use graphical tool for modeling data.
- Widely used in Database Design.
- It is a GUI representation of the logical structure of a Database.
- It helps you to identifies the entities which exist in a system and the relationships between those Entities.

Following are the main components and its symbols in ER Diagrams:

- Rectangles: This Entity Relationship Diagram symbol represents entity types.
- Ellipses: Symbol represent attributes.
- Diamonds: This symbol represents relationship types.
- Lines: It links attributes to entity types and entity types with other relationship types.
- Primary key: attributes are underlined.
- Double Ellipses: Represent multi-valued attributes.

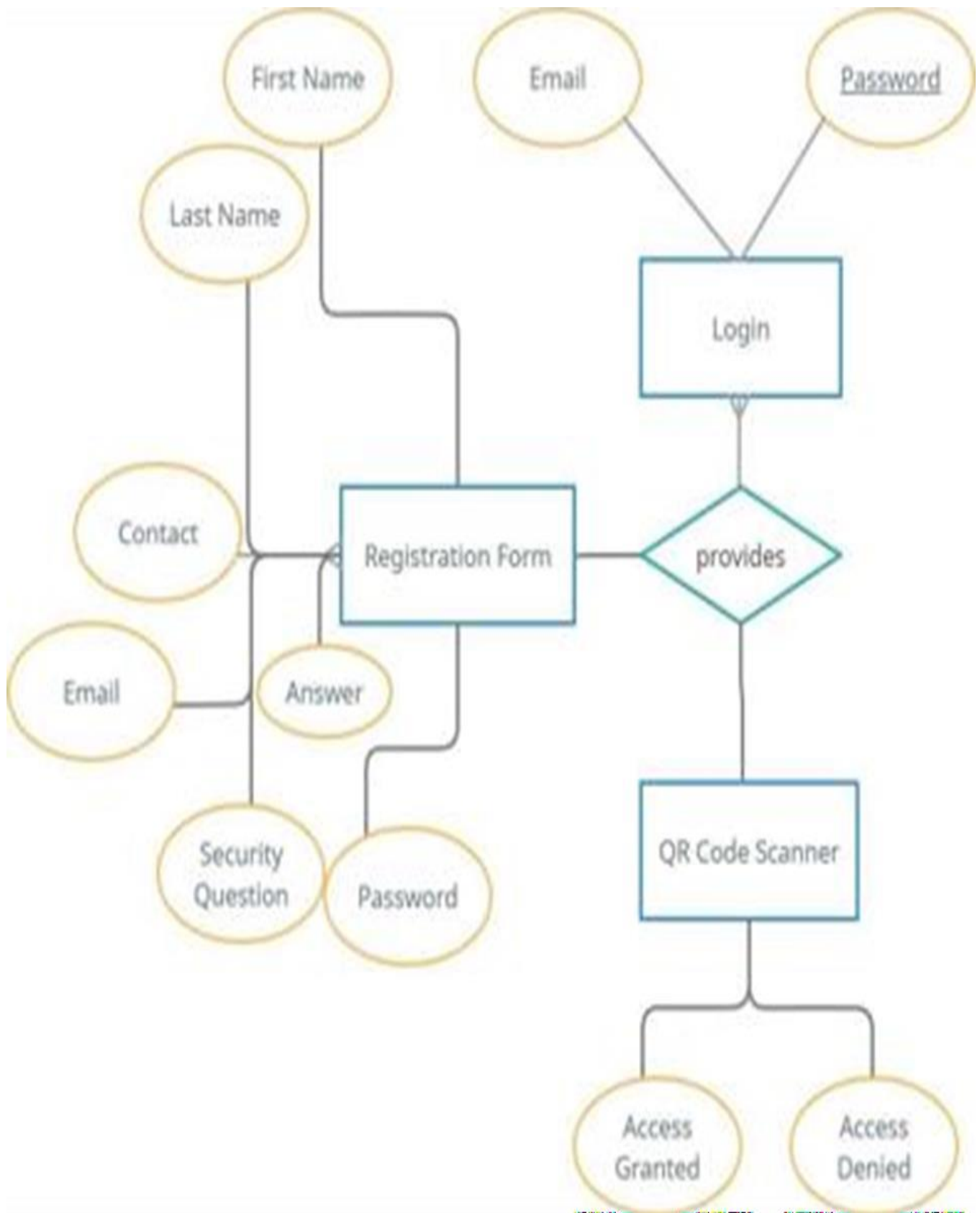


Fig:2.2 ER Diagram

2.7 Use Case Model Survey:

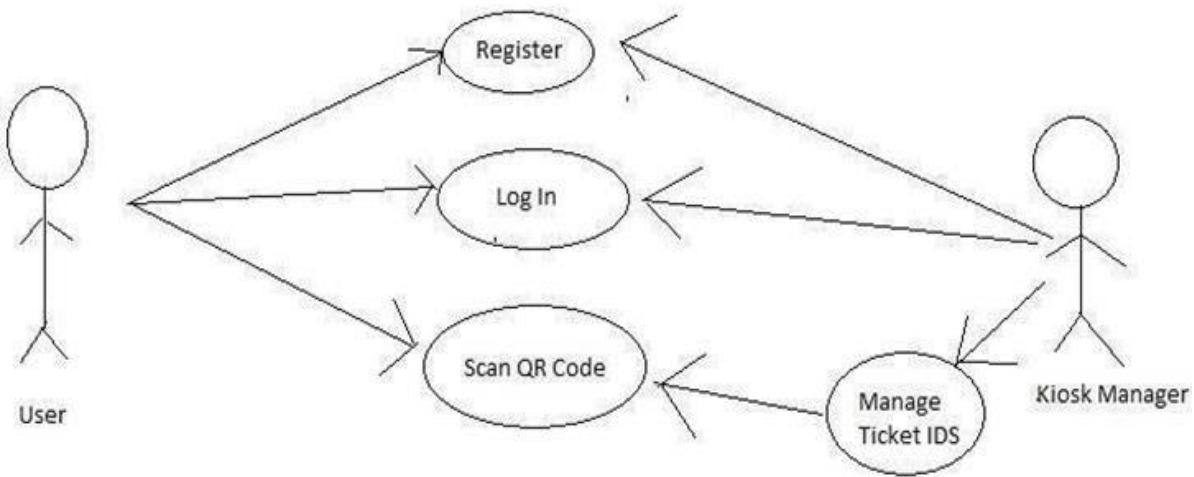


Fig2.3: Use Case Model Survey

• User:

User(passenger) can register itself with the airport, log in at the kiosk and scan the QR code printed on the e ticket

• Kiosk Manager:

The kiosk manager will be able to keep track of the passengers that have registered, logged in or have entered the airport by scanning the QR code. He will also be able to manage the ticket IDs that have been granted access to the airport on a particular day

2.8 Database Design:

Passengers
<ul style="list-style-type: none">• First_Name VARCHAR (50)• Last_Name VARCHAR (50)• Contact_Number INT• Email_ID VARCHAR (50)• Security_question VARCHAR (80)• Answer VARCHAR (50)• Key - Password VARCHAR (50)



Fig 2.4 Data base design

2.9 Architecture Diagram:

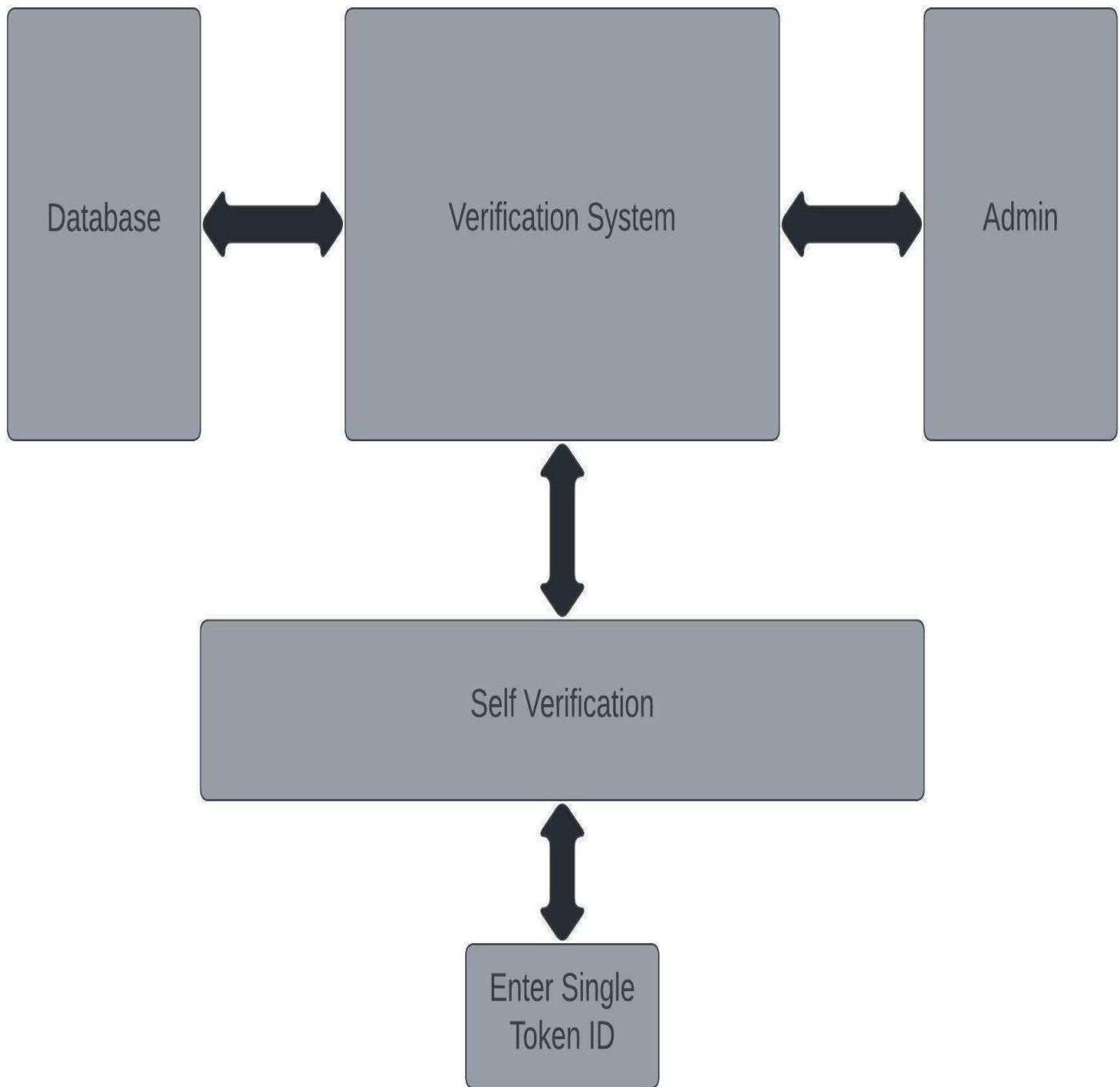


Fig 2.5 Architecture Diagram

- **3. Specific Requirements:**

- **3.1 Activity Diagrams:**

- **Log In and QR Code verification Activity**

Initially, a passenger arrives at the Log In kiosk and uses his user ID and password to log into the kiosk. After the log in is successful, the passenger then proceeds to scan the QR code printed on the e ticket which contains the ticket ID. The QR code is scanned and based on whether the ticket ID is valid or not, the passenger is either granted or denied access.

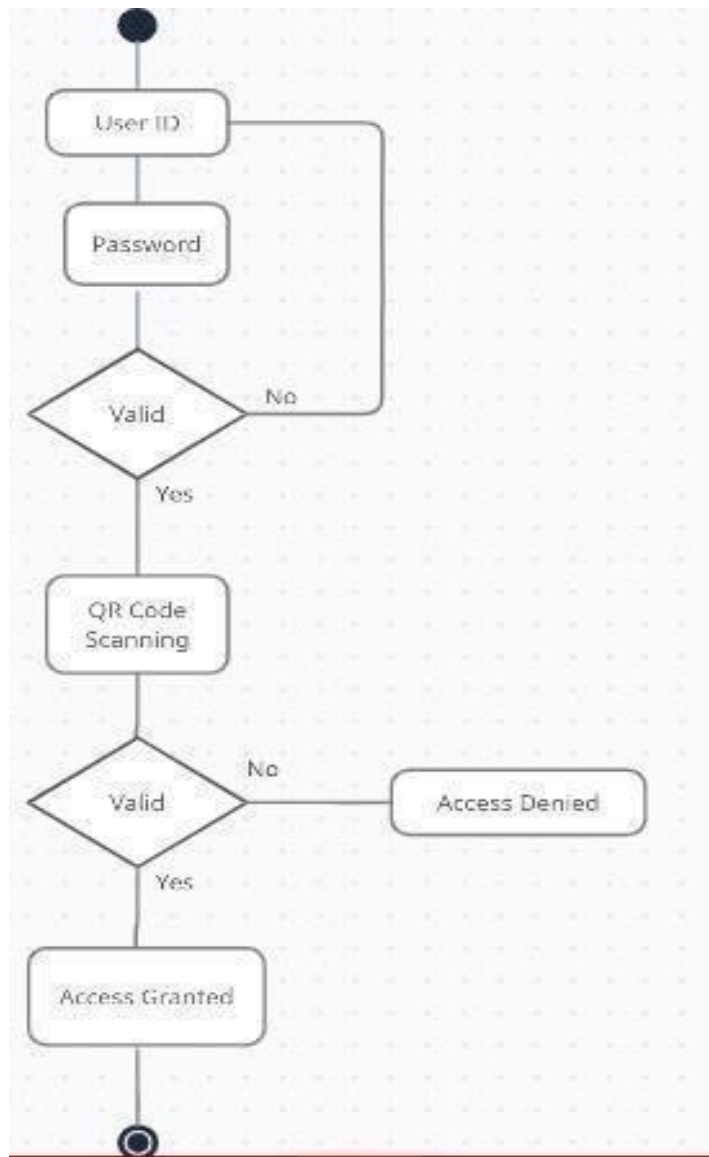


Fig 3.1: Login and QR Code verification Activity

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

• **Purpose of Activity Diagrams:**

The purpose of an activity diagram can be described as –

Draw the activity flow of a system.

Describe the sequence from one activity to another.

Describe the parallel, branched and concurrent flow of the system.

• **Uses of Activity Diagrams:**

The basic usage of activity diagram is similar to other four UML diagrams. The specific usage is to model the control flow from one activity to another. This control flow does not include messages.

Activity diagram is suitable for modeling the activity flow of the system. An application can have multiple systems. Activity diagram also captures these systems and describes the flow from one system to another. This specific usage is not available in other diagrams. These systems can be database, external queues, or any other system.

So, it gives high level view of a system. This high-level view is mainly for business users or any other person who is not a technical person.

This diagram is used to model the activities which are nothing but business requirements. The diagram has more impact on business understanding rather than on implementation details.

Activity diagram can be used for – Modeling work flow by using activities. Modeling business requirements. High level understanding of the system's functionalities.

Investigating business requirements at a later stage

3.2 Sequence Diagram:

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

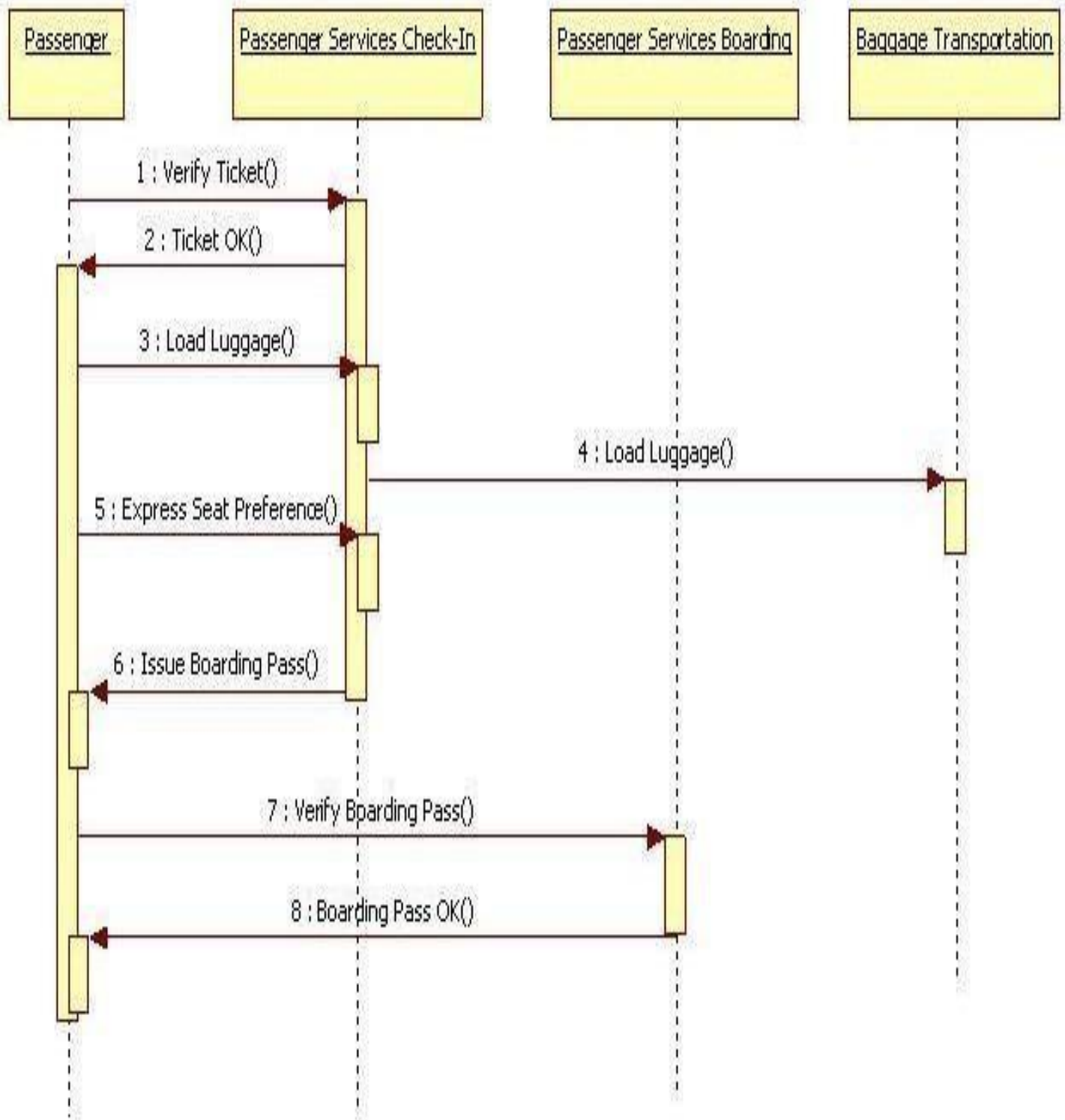


Fig 3.2: Sequence Diagram for Passenger Verification System

References:

- <https://www.jetbrains.com/pycharm/>
- <https://amadeus.com/en/portfolio/airports/amadeus-passenger-verification>
- [https://www.provab.com/travel-apis.](https://www.provab.com/travel-apis)
- <https://dev.mysql.com/doc/airportdb/en/airportdb>
- [https://docs.python.org/3/library/tkinter.](https://docs.python.org/3/library/tkinter)
- <https://www.lucidchart.com/>