CLICK CARD- MAKING AN APPLICATION FOR GENERATING AND UTILIZING INFORMATION CARD USING AUGMENTED REALITY

A PROJECT REPORT

Submitted by

JOEL STANLY.A **(211417205066)**

VASANTHAKUMAR.R **(211417205175)**

SANJEEV KUMAR.S **(211417205142)**

In partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

PANIMALAR ENGINEERING COLLEGE, POONAMALLEE

ANNA UNIVERSITY: CHENNAI 600 025

MARCH 2021

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**CLICK CARD- MAKING AN APPLICATION FOR GENERATING AND UTILIZING INFORMATION CARD USING AUGMENTED REALITY"** is the Bonafide work of **"A.JOEL STANLY (211417205066), R.VASANTHAKUMAR (211417205175), S.SANJEEV KUMAR (211417205142)"** who carried out the project under my supervision.

SIGNATURE

Dr.M.HELDA MERCY,M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
Department of Information Technology
Panimalar Engineering College
Poonamallee, Chennai - 600 123

SIGNATURE

Dr.N.BALA SUNDARAGANAPATHY
ASSOCIATE PROFESSOR
Department of Information Technology
Panimalar Engineering College
Poonamallee, Chennai - 600 123

Submitted for the project and viva-voce examination held on **05-08-2021.**

SIGNATURE
INTERNAL EXAMINER

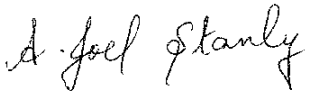
SIGNATURE
EXTERNAL EXAMINER

DECLARATION

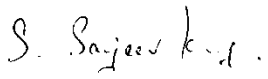
I hereby declare that the project report entitled "**CLICK CARD- MAKING AN APPLICATION FOR GENERATING AND UTILIZING INFORMATION CARD USING AUGMENTED REALITY**" which is being submitted in partial fulfillment of the requirement of the course leading to the award of the 'Bachelor of Technology in Information Technology' in **Panimalar Engineering College, Affiliated to Anna University – Chennai** is the result of the project carried out by me under the guidance and supervision of **Dr. N. BALA SUNDARA GANAPATHY, M.E., Ph.D., Associate Professor in the Department of Information Technology**. I further declare that I or any other person has not previously submitted this project report to any other institution/university for any other degree/diploma or any other person.

Signature of batch members

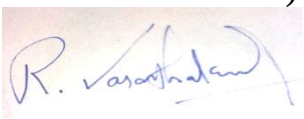
Date: **05-08-2021**



Place: Chennai (A.JOEL STANLY)



(S.SANEEV KUMAR)



(R.VASANTHAKUMAR)

It is certified that this project has been prepared and submitted under my guidance.

Date: **05-08-2021** (**Dr. N.BALA SUNDARA GANAPATHY, M.E.,Ph.D.,)**

Place: Chennai (Associate Professor/IT)

ACKNOWLEDGEMENT

A project of this magnitude and nature requires kind cooperation and support from many, for successful completion. We wish to express our sincere thanks to all those who were involved in the completion of this project.

We would like to express our deep gratitude to **Our Honourable Secretary and Correspondent, Dr. P. CHINNADURAI, M.A., Ph.D.,** for his kind words and enthusiastic motivation which inspired us a lot in completing this project.

We also express our sincere thanks to **Our Reputed Directors Mrs. C. VIJAYA RAJESHWARI, Dr. C. SAKTHI KUMAR M.E., Ph.D., and Mrs. SARANYA SREE SAKTHI KUMAR B.E.,** for providing us with the necessary facilities for completion of this project.

We also express our appreciation and gratefulness to **Our Principal Dr. K. MANI, M.E., Ph.D.,** who helped us in the completion of the project. We wish to convey our thanks and gratitude to our head of the department, **Dr. M. HELDA MERCY, M.E., Ph.D.,** Department of Information Technology, for her support and by providing us ample time to complete our project.

We express our indebtedness and gratitude to our staff in charge, **Dr. N.BALA SUNDARA GANAPATHY, M.E., Ph.D.,** Associate Professor, Department of Information Technology for his guidance throughout the course of our project.

Last but not least, we thank our parents and friends for providing their extensive moral support and encouragement during the course of the project.

ABSTRACT

Augmented Reality for business cards is a new way of introducing their product to others. Inviting others using AR is also an added advantage of this project. If we use a physical card, we are limited to giving information. This drawback is recovered in this project since we are using digital data to store the information. We can edit the information whenever we want and it is pretty simple rather than printing out a new set of physical cards for corrections and editing. We can use a variety of formats such as text, images, videos, and even 3D animations that can't be used in physical cards and the cost is very less. Augmented Reality for business cards is a new way of introducing their product to others. Online Website used to let customers upload all their text, Images, Videos, So that we can collect all the information about the company. Therefore, the user would have a more personalized way of exploring the details of the product and company – right out of your card. The pre-recorded video/image can be uploaded through the website which generates a unique QR code that can be printed and which can be scanned by the customers.

LIST OF FIGURES

LIST OF ABBREVIATIONS

SRS	Software Requirement Specification
FCL	Framework Class Library
CLR	Common Language Runtime
BCL	Base Class Library
CLI	Common Language Infrastructure
RDBMS	Relational Database Management System
ANSI	American National Standards Institute
BLO	Binary Large Objects
UWP	Universal Windows Platform
ERD	Entity-Relationship Diagram
DFD	Data Flow Diagram

LIST OF CONTENTS

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

1.1 INTRODUCTION TO PROJECT

Augmented reality for business cards is a new way of introducing their product to others. Inviting others using AR is also an added advantage of the project. If we use a physical card, we are limited to give information. This drawback is recovered in this project since we are using digital data to store the information. We can edit the information whenever we want and it is pretty simple rather than printing out a new set of physical cards for corrections and editing. We can use varieties of formats such as text, images, videos that can't be used in physical cards and these cost very less. Augmented Reality for business cards is a new way of introducing their product to others.

Website for customers:

Online Website used to let customers upload all their text, Images, and Videos, So that we can collect all the information about the company. Therefore the user would have a more personalized way of exploring the details of your product and company – right out of your card. The pre-recorded video/image can be uploaded through the website which generates a unique QR code that can be printed and which can be scanned by the customers

Application for End user:

Once the users use the free application to scan the card, it would instantly load a video/animation/image right on their screen, in a pop-up manner that is pre-recorded and designed in collaboration with you. Hence the application can be viewed with the help of Augmented Reality So that it will be more attractive to the user to use the application.

Uploading the Data:

The pre-recorded video/image can be uploaded through the website which generates a unique QR code that can be printed and which can be scanned by the customers. Therefore the user would have a more personalized way of exploring the details of your product and company – right out of your card.

Augmented Reality:

Click Card can make our information card unique by adding AR features to it and creating a personalized mobile app for us.

Click Card:

For business, the more creative card is, the more curious clients would be, and the more popularity the business would gain. Making card innovative would end up more people eager to scan the card through the branded app.

Edit the updated Information:

If we use a physical card, we are limited to giving information. We can edit the information whenever we want and it is pretty simple rather than printing out a new set of physical cards for corrections and editing.

1.2 SCOPE OF THE PROJECT

Augmented Reality for business cards is a new way of introducing their product to others. Inviting others using AR is also an added advantage of the project. If we use a physical card, we are limited to giving information. This drawback is recovered in this project since we are using digital data to store the information. We can edit the information whenever we want and it is pretty simple rather than printing out a new set of physical cards for corrections and editing. We can use variety of formats such as text, images, videos, and even 3D animations that

can’t be used in physical cards and these cost Augmented Reality for business cards is a new way of introducing their product to others.

Click card using augmented reality optimizes Click card operations, attractive processes and is cost-effective in the longer run.

- On a study of what would improve the Business card from using the technology we concluded at the following results which included online Advertisement, Augmented reality, Application for the end users, correcting the updated information, providing loyal and regular customer’s offers. We included some of the above services for our application this software solves the problem of optimizing the time in accessing the click card by providing facilities such as:

- We can scan the QR code using the mobile application itself.

CHAPTER 2
SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

2.1INTRODUCTION

After analyzing the requirements of the task to be performed, the next step is to analyze the problem and understand its context. The first activity in the phase is studying the existing system and another is to understand the requirements and domain of the new system. Both the activities are equally important, but the first activity serves as a basis of giving the functional specifications and then the successful design of the proposed system. Understanding the properties and requirements of a new system is more difficult and requires creative thinking and understanding of the existing running system is also difficult, improper understanding of the present system can lead to diversion from solution.

2.2 ANALYSIS MODEL

The model that is basically being followed is the Iterative Model. In the Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the model:

- Requirement Analysis
- Project Planning
- System design
- Detail design

- Coding
- Unit System integration & testing

One phase is the input of another phase. The output of each phase is to be consistent with the overall requirement of the system.

2.3 SYSTEM REQUIREMENTS

HARDWARE AND SOFTWARE SPECIFICATIONS

HARDWARE REQUIREMENTS

• PROCESSOR - Core I 7

• SPEED - 2.5 GHZ

• HARD DISK - 8 GB (MIN)

• KEY BOARD - STANDARD WINDOWS KEYBOARD

• MOUSE - TWO OR THREE BUTTON MOUSE

• MONITOR - SVGA

SOFTWARE REQUIREMENTS

. OPERATING SYSTEM - WINDOWS 7 / ABOVE

• APPLICATION - PYTHON

• TOOL – VISUAL STUDIO CODE

.

2.4 EXISTING SYSTEM

➢ Business cards have been widely used and a core part of the common business etiquette. It is arguably one of the few business communication tools that have not evolved since it began in the 17th century.

➢ Professional companies and business professionals use business card advertising to get a message across and make contact information available to potential customers.

➢ Business cards are cards bearing business information about a company or individual. They are shared during formal introductions as a convenience and a memory aid.

➢ Since the early 1990s there has been an exponential increase in the growth of online advertising, which has evolved into a standard for small and large organizations

➢ Online advertising is a marketing strategy that involves the use of the Internet as a medium to obtain website traffic and target and deliver marketing messages to the customers

DRAWBACK OF EXISTING SYSTEM:

➢ Consumers don’t usually look and click on the online advertisements.

➢ The printed cards are very cost expensive considering that it is limited to give the required information.

➢ The printed cards are not editable.

➢ The printed cards are digitalized which doesn’t reach across as expecting.

➢Online Advertisements cannot be used for personal purposes like invitation cards etc.,

2.5 PROPOSED SYSTEM

➢An augmented reality information card is an ideal solution to this problem, not just because of its "wow" effect, but also because of its utility and convenience. With an AR information card, you can stand out from other people and make it really easy for the other person to find out more about you. All they need to do is point their phone at it.

➢ Your AR business card acts as a virtual ad for yourself, bringing people additional valuable content about you, which you couldn't possibly fit onto a regular business card.

➢This can range from videos about your business and fun 3D models to links to your social media and website and call-to-actions.

➢By simply scanning your business card with mobile devices, your 3D product model will immediately appear.

➢ Privatized information can be only accessed by certain people specified which can be detected by their login information.

Advantages:

➢Impressive Rich Content

➢Share from anywhere & anytime

➢Easy and inexpensive.

➢Can be editable at anytime.

➢The amount of information can be shown is not limited.

2.6 STUDY OF THE PROJECT

Click Card can make your information card unique by adding AR features to it and creating a personalized mobile app for you. For business, the more creative your card is, the more curious your clients would be, and the more popularity you and your business would gain. Making your card innovative would end up in more people eager to scan the card through the branded app.

2.6.1 LITERATURE SURVEY

Review and analysis of augmented reality literature for construction industry

AUTHORS: Sara Rankohi and Lloyd Waugh

DESCRIPTION: Research has identified various beneficial capabilities for augmented reality technologies in the AEC industry such as virtual site visits, comparing as-built and as-planned status of projects, pre-empting schedule disputes, enhancing collaboration opportunities, and planning/training for similar projects. This paper provides an expanded foundation for future research by presenting a statistical review of augmented reality technology in the AEC industry. The review is based on articles found within eight well-known journals in architecture, engineering, construction, and facility management (AEC/FM) until the end of the year 2012. The review further narrows the literature within these journals by considering only those 133 articles found through a key word search for "augmented reality." The selected journal articles are classified within the following dimensions: improvement focus, industry sector, target audience, project phase, stage of technology maturity, application area, comparison role, and technology. The number of articles within these dimensions are used to identify maturing and emerging trends in the literature as well as to synthesize the current state-of-the-art of augmented reality research in the AEC industry.

Augmented Reality and Television: Dimensions and Themes

AUTHORS: pejman Saeghe,Gavin Abercrombie,Bruce Weir,Sarah Clinch

DESCRIPTION: Commercialization of augmented reality (AR) devices has led to their growing application in domestic environments and leisure activities. One such domain is that of television, where AR is one of several technologies driving innovation (c.f. Internet broadcasting, second screen devices). We conduct a systematic literature review to quantify research at the intersection of AR and broadcast television. We identify six common themes and a set of cross-cutting design decisions. We distill this information into a design space incorporating six dimensions: abstraction, interaction, time, display, context and editorial control. We provide methods to operationalize the dimensions to enable research and development of novel concepts, and through this generate six design guidelines to shape future activity at the intersection of AR and television.

An augmented reality interface for visualizing and interacting with virtual content

AUTHORS: Fotis Liarokapis

DESCRIPTION: In this paper, a novel AR interface is proposed that provides generic solutions to the tasks involved in augmenting simultaneously different types of virtual information and processing of tracking data for natural interaction. Participants within the system can experience a real-time mixture of 3D objects, static video, images, textual information and 3D sound with the real environment. The user-friendly AR interface can achieve maximum interaction using simple but effective forms of collaboration based on the combinations of human–computer interaction techniques. To prove the feasibility of the interface, the use of indoor AR

techniques are employed to construct innovative applications and demonstrate examples from heritage to learning systems. Finally, an initial evaluation of the AR interface including some initial results is presented.

Teleoperators and virtual environment

AUTHORS: Bilal Mirza, Wei Wang, Jie Wang, Howard C

DESCRIPTION: This paper surveys the field of augmented the field of augmented reality, in which 3D virtual objects are integrated into a 3D real environment in real time. Detailed discussion of the tradeoffs between optical and video blending approaches. Registration and sensing are two biggest problems in building effective augmented reality systems

Human interface technology laboratory

AUTHORS: Craig E. Wheelock, Victoria M. Goss, David Balgoma

DESCRIPTION: The objective is to observe the trend and the importance of mobile augmented reality. The result indicates that mobile AR is a potential tool to assist a user in many tasks. Registration and sensing are two biggest problems in building effective augumented reality systems

2.7 CONCLUSION

By analyzing the existing system and literature survey, we have concluded that the Click card using Augmented reality are those most informative features that can be used to provide your Information unique by adding AR feature to it and hence by creating an application for the end user.

CHAPTER 3
FEASIBILITY REPORT

3. FEASIBILITY REPORT

The preliminary investigation examines project feasibility; the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging the oldest running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility

3.1. Technical Feasibility

Visual Studio is an open source environment is used in our project. The front end used in our project is Visual Studio to run the coding part. The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Does the proposed equipment have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide an adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of the ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible.

It is a browser-based user interface for audit workflow. Thus, it provides easy access to users.

The purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the rules specified. Therefore, it provides a technical guarantee of accuracy, reliability, and security. The software and hard requirements for the development of this project are not many and are already available or are available as free as open source.

The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing fast feedback to the users irrespective of the number of users using the system.

3.2. Operational Feasibility

The analyst considers the extent the proposed system will fulfill his departments. That is, whether the proposed system covers all aspects of the working system and whether it has considerable improvements. We have found that the proposed “Secure transaction” will certainly have considerable improvements over the existing system.

3.3. Economic Feasibility

The proposed system is economically feasible because the cost involved in purchasing the hardware and the software is approachable. Working in this system need not require a highly qualified professional. The operating-environment costs are marginal. The less time involved also helped in its economic feasibility.

CHAPTER 4
SYSTEM DEVELOPMENT ENVIRONMENT

4. SYSTEM DEVELOPMENT ENVIRONMENT

4.1 INTRODUCTION TO VISUAL STUDIO CODE:

Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

Microsoft has released most of Visual Studio Code's source code on the Microsoft/vscode (**Code – OSS**) repository of GitHub, under the permissive MIT License, while the releases by Microsoft are proprietary freeware.

In the Stack Overflow 2019 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents reporting that they use it.

Visual Studio Code was first announced on April 29, 2015, by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter.

On November 18, 2015, Visual Studio Code was released under the MIT License, having its source code available on GitHub. Extension support was also announced. On April 14, 2016, Visual Studio Code graduated from the public preview stage and was released to the Web.

FEATURES:

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python and C++. It is based on the Electron framework, which is used to develop Node.js Web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure

DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language. It supports a number of programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface but can be accessed via the command palette.

Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, and debuggers, perform static code analysis, and add code linters using the Language Server Protocol.

Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

Visual Studio Code allows users to set the code page in which the active document is saved, the newline character, and the programming language of the active document. This allows it to be used on any platform, in any locale, and for any given programming language.

Language support:

Out-of-the-box, Visual Studio Code includes basic support for most common programming languages. This basic support includes syntax highlighting, bracket matching, code folding, and configurable snippets. Visual Studio Code also ships with IntelliSense for JavaScript, TypeScript, JSON, CSS, and HTML,

as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace.

Data collection:

Visual Studio Code collects usage data and sends it to Microsoft, although this can be disabled. In addition, because of the open-source nature of the application, the telemetry code is accessible to the public, who can see exactly what is collected. According to Microsoft, the data is shared with Microsoft-controlled affiliates and subsidiaries, although law enforcement may request it as part of a legal process.

Version control

Source control is a built-in feature of Visual Studio Code. It has a dedicated tab inside of the menu bar where you can access version control settings and view changes made to the current project. To use the feature, you must link Visual Studio Code to any supported version control system (Git, Subversion, Perforce, etc.). This allows you to create repositories as well as make push and pull requests directly from the Visual Studio Code program.

4.2 SOFTWARE REQUIREMENT SPECIFICATION

The purpose of this document is to present a detailed description of the Web application system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the Regional Historical Society for its approval.

The purpose of this Software Requirement Specification (SRS) is to help the project. It is provided with some requirements which are used in the Transaction Mercator System. All parts design, coding, and testing will be

prepared with the helping of SRS. The purpose of this document is to detail the requirements placed on the Transaction Mercator System and serves as a contract between the customer and the developers as to what is to be expected of the stock exchange, and how the components of the system are working with each other with external systems.

DEVELOPERS RESPONSIBILITIES OVERVIEW

The developer is responsible for: Developing the system, which meets the SRS and solving all the requirements of the system?

- Demonstrating the system and installing the system at the client's location after the acceptance testing is successful.

Submitting the required user manual describing the system interfaces to work on it and also the documents of the system

- Conducting any user training that might be needed for using the system.
- Maintaining the system for a period of one year after installation.

4.3 FUNCTIONAL REQUIREMENTS

Following is a list of functionalities of the browsing enabled system.

- An Activity with a UI that allows you to browser settings. Provide a second Activity that allows users to access the share with permission from the administrator. Handle the activity lifecycle appropriately. A precondition for any points in this part of the grade is code that compiles and runs.

- Your application should allow a user to browse the shares, buy and sell the shares with specific metadata. The assignment requires you to create a UI for browsing and a UI for integrating the two.

4.4 NON-FUNCTIONAL REQUIREMENTS

Each member should have a separate system. The system should ask the username and password to open the application. It doesn't permit to unregistered user to access the System. The system should have Role-based System functions access. Approval Process has to be defined. The system should have Modular customization components so that they can be reused across the implementation.

These are mainly the following:

- Secure access to confidential data (employee's details). 24 X 7 availability
- Better component design to get better performance at peak time
- Flexible service based architecture will be highly desirable for future extension.

4.5 PERFORMANCE REQUIREMENTS

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the required specifications are properly given, it is possible to design a system, which will fit into the required environment. It rests largely on the part of the users of the existing system to give the required specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

CHAPTER 5
SYSTEM DESIGN

5. SYSTEM DESIGN

5.1 INTRODUCTION

Software design sits in the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement has been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. The design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design, we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either a technical or project management perspective. From the technical point of view, the design is comprised of four activities – architectural design, data structure design, interface design, and procedural design.

5.2 NORMALIZATION

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy, i.e. repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insert, Update, deletes anomalies.

Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relations.

Insertion anomaly: Inability to add data to the database due to the absence of other data.

Deletion anomaly: Unintended loss of data due to the deletion of other data.

Update anomaly: Data inconsistency resulting from data redundancy and partial update

Normal Forms: These are the rules for structuring relations that eliminate anomalies.

FIRST NORMAL FORM

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this, we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

SECOND NORMAL FORM

A relation is said to be in the second normal form is it is in first normal form and it should satisfy any one of the following rules.

- The primary key is not a composite primary key
- No, no key attributes are present

- Every now key attribute is fully functionally dependent on a full set of the primary key.

THIRD NORMAL FORM

A relation is said to be in third normal form if their exits no transitive dependencies.

Transitive Dependency:

If two on key attributes depend on each other as well as on the primary key, then they are said to be transitively dependent.

The above normalization principles were applied to decompose the data in multiple tables, thereby making the data to be maintained in a consistent state.

5.3 SYSTEM ARCHITECTURE

- An "architecture" can be defined as an abstract description of entities in a system and the relationships between them. It involves a series of decision-making processes.
- The architecture is a structure and a vision.
- A "system architecture" is the embodiment of concepts and the distribution of the correspondences between the functions of things or information and formal elements. It defines the relationships among elements as well as between elements and the surrounding environment.
- Building a sound architecture is a complex task and a great topic for us to discuss here. After you build an architecture, relevant parties must understand it and follow its dictates
- An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool

as it provides an overall view of the physical deployment of the software system and its evolution roadmap.

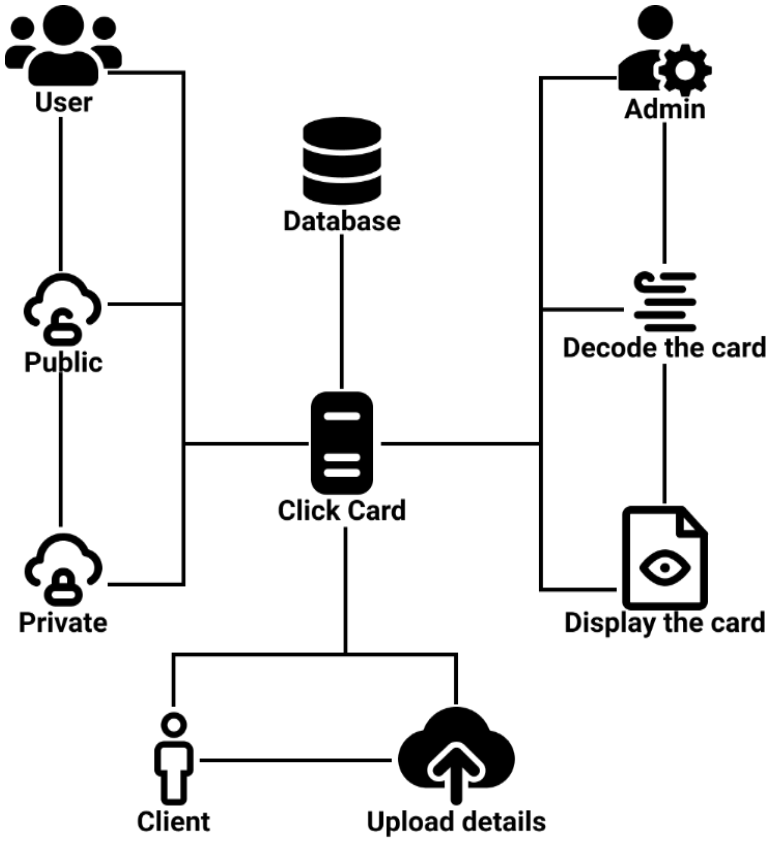


Fig.5.3.1 system architecture

There are three accessors, user, client and admin. Here the user can post their data onto the server to be stored and which can be accessed by the client using either the QR code or their id of the data. The admin then decodes the data asked by the client and displays the card.

User:

At first, the user has to enter the basic user details and then their data have to entered by the user. The details entered by the user will be sent to the client and will be stored in the database. The user can either store their data in public so that anyone can view the data or in private so that those who have the access can only view the data.

Client:

The client will analyze the details from the database and uses the client information that is stored in the database. If the user provided the data as private, then client need the auto generated password to access the data of the user.

Admin:

Finally, the admin who is referred to us the higher of this entire process will manage the storage of the user data and provide access to those who are authorized. The admin also displays the data of the user to the client in a certain way.

Database:

The database is the place where all the contents provided are stored. When the user uploads the data, their password, unique id, QR data and the data provided by user is generated and stored into the database.

Display the card:

When a client wants to access the specific data, first we need to decode the data from the QR then we need to check whether they are authorized to view the data. If they are authorized then, the data given is displayed to the user using augmentation.

5.4 DATA FLOW DIAGRAM

Definition:

A Data Flow Diagram (DFD) is a graphical tool used to describe and analyse the movement of data through the system. It is a graphical representation of the “flow” of data through a computer system or a data or it looks at how data flows through a system. These are a central tool and basic

from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. The development of DFD is done at several levels. The flow diagram describes the boxes that describe computations, decisions, interactions & loops. It is important to keep in mind that the flow diagrams are not flowcharts and should not include control elements.

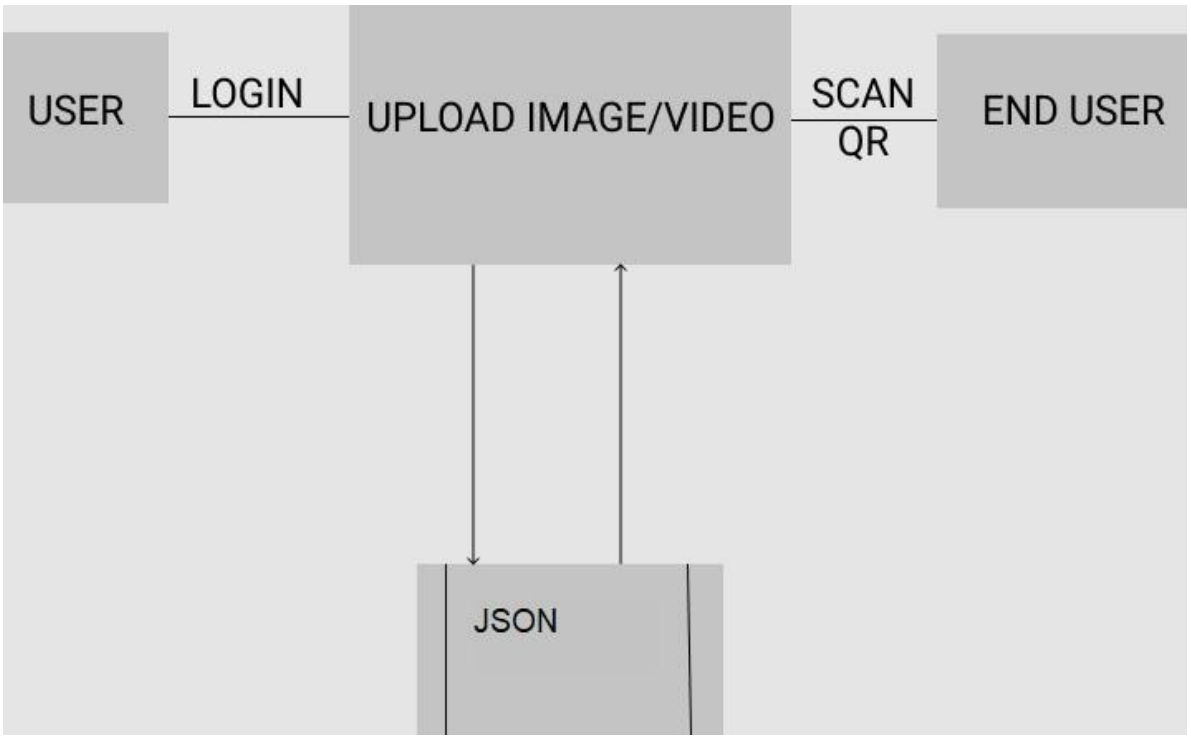


Fig.5.4.1 Dataflow Diagram 0

5.5 USE CASE DIAGRAM

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

A use case diagram is usually simple. It does not show the detail of the use cases:

It only summarizes some of the relationships between use cases, actors, and systems.

It does not show the order in which steps are performed to achieve the goals of each use case.

As said, a use case diagram should be simple and contains only a few shapes. If yours contain more than 20 use cases, you are probably misusing use case diagram**.**

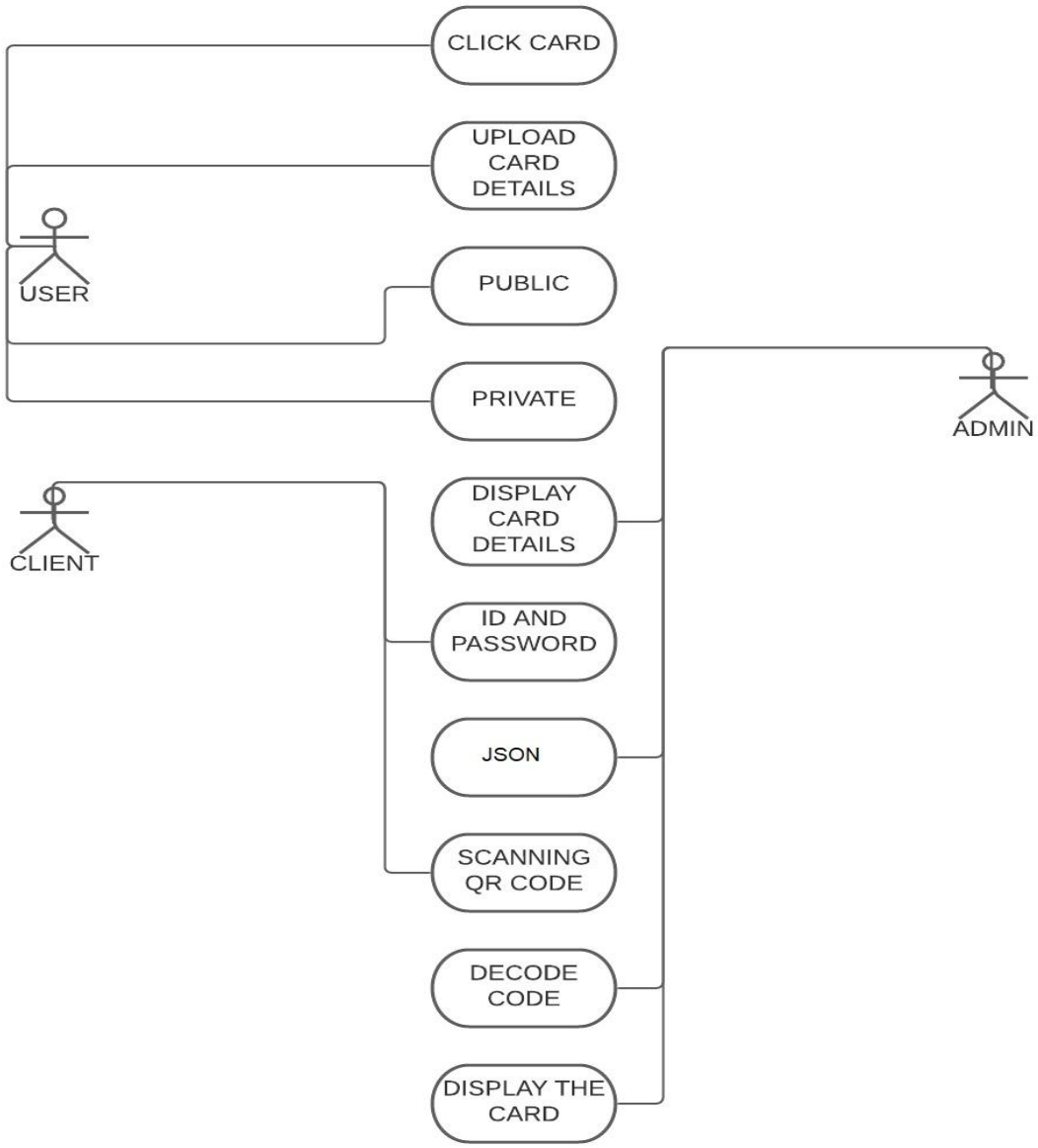


Fig.5.5.1 Use Case Diagram

5.6 CLASS DIAGRAM

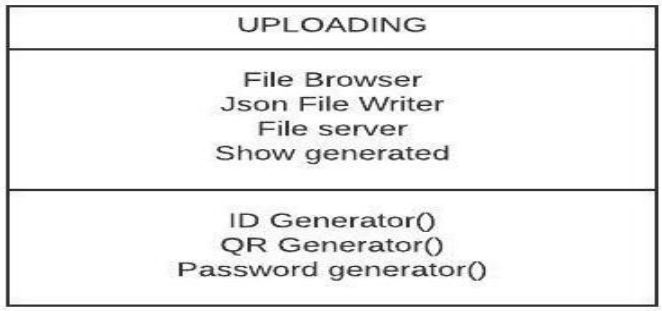
The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling the classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

In the diagram, classes are represented with boxes that contain three compartments:

- The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
- The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase.
- The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase.

A class with three compartments:

In the design of a system, a number of classes are identified and grouped together in a class diagram that helps to determine the static relations between them. With detailed modeling, the classes of the conceptual design are often split into a number of subclasses



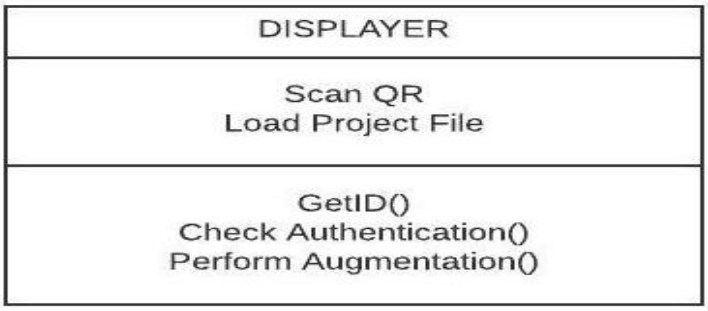


Fig.5.6.1 Class Diagram

5.7 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time. The four major components of a collaboration diagram are:

1. Objects- Objects are shown as rectangles with naming labels inside. The naming label follows the convention of object name: class name. If an object has a property or state that specifically influences the collaboration, this should also be noted.
2. Actors- Actors are instances that invoke the interaction in the diagram. Each actor has a name and a role, with one actor initiating the entire use case.

3. Links- Links connect objects with actors and are depicted using a solid line between two elements. Each link is an instance where messages can be sent

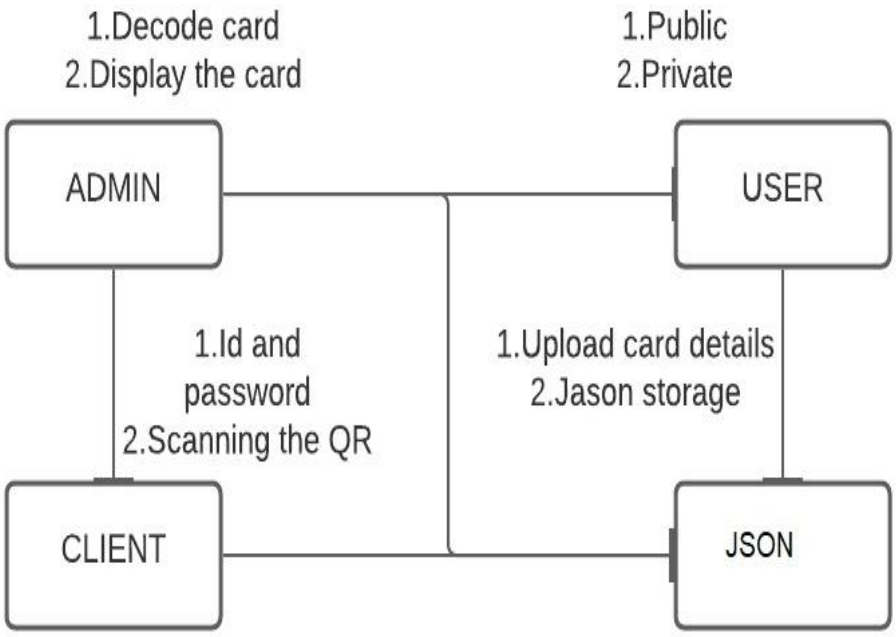


Fig.5.7.1 Collaboration Diagram

5.8 COMPONENT DIAGRAM

Component diagrams are different in terms of nature and behavior. Component diagrams are used to model the physical aspects of a system. Physical aspects are the elements such as executables, libraries, files, documents, etc. which reside in a node.

Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

Thus, from that point of view, component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc.

Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment.

A single component diagram cannot represent the entire system but a collection of diagrams is used to represent the whole.

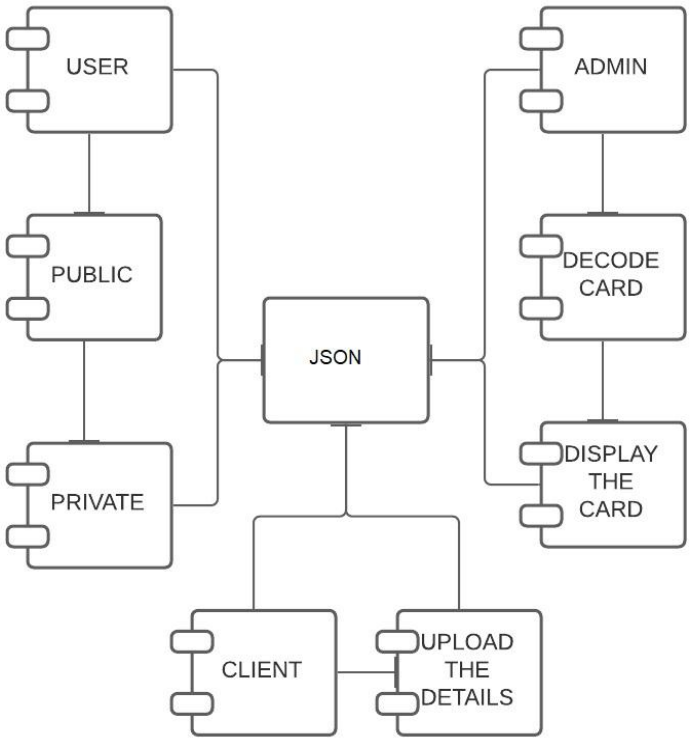


Fig.5.8.1 Component Diagram

5.9 DEPLOYMENT DIAGRAM

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

Deployment diagrams are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware.

Deployment diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system.

Follow the simple steps below to draw a deployment diagram. You can either use the deployment diagram examples below to get a head start or use our UML diagram tool to start from the beginning.

Step 1: Identify the purpose of your deployment diagram. And to do so, you need to identify the nodes and devices within the system you'll be visualizing with the diagram.

Step 2: Figure out the relationships between the nodes and devices. Once you know how they are connected, proceed to add the communication associations to the diagram.

Step 3: Identify what other elements like components, active objects you need to add to complete the diagram.

Step 4: Add dependencies between components and objects as required

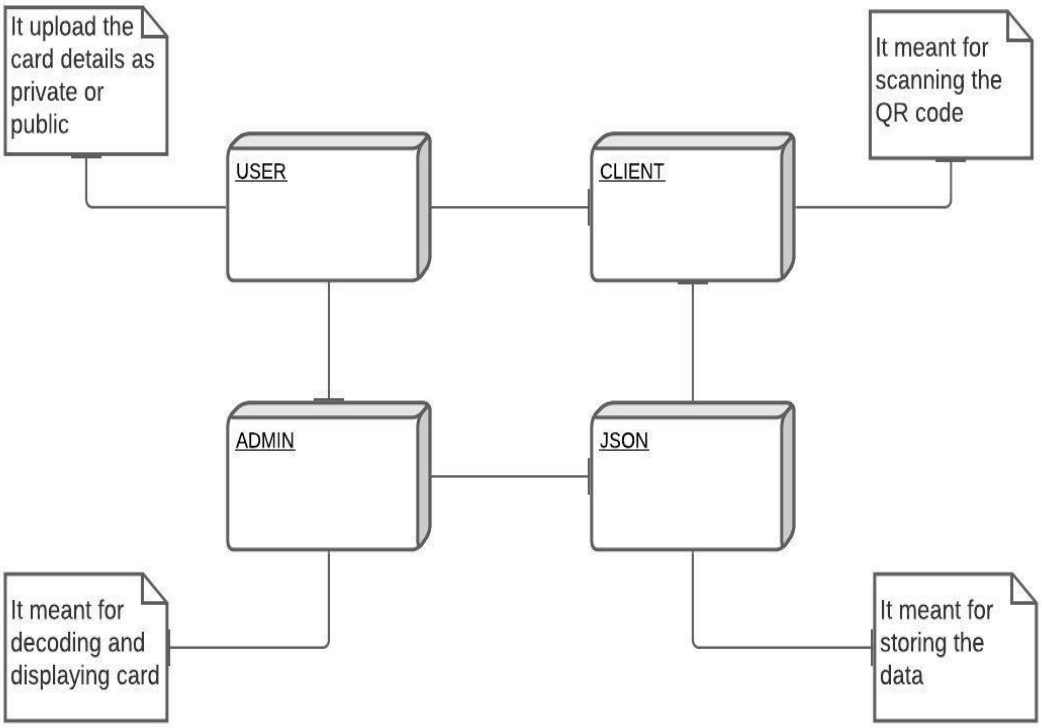


Fig.5.9.1 Deployment Diagram

CHAPTER 6
SYSTEM MODULES

6. SYSTEM MODULES

There are four modules in the system. They are as follows:

- **DISPLAY THE DETAILS**
- **HOME LOGIN**
- **UPLOAD FILES**
- **RESULT**

6.1 DISPLAY THE DETAILS:

In this module we have to display the details of the card that is accessed. If the card is accessed using the QR, we have to decode the data from the QR and the data is used to look for the result data that is provided by the user. If the card is not accessed by the QR, then it can be accessed by the id given to the result data. Some people need privacy for their data, so they would privatize it. If so, they need the password to access the result data. If the result data is not privatized then anyone can access the data. The result data will be augmented to the QR code by the system or it will be displayed by the system if it cannot be augmented.

ALGORITHM:

Step 1: Start.

Step 2: Get the information whether the user wants to access the card using id or QR code.

Step 3: If the user chooses to do with QR code, then get the password provided and get the data from the QR code.

Step 4: If the password matches for the data from the QR code scanned in the database, then provide access to the data from the database to be augmented.

Step 5: If the user chooses to do with Id, then get the id and password provided.

Step 6: If the password matches for the id in the database, the provide access to the data from the database to be displayed.

Step 7: Stop.

CODE:

```
def idgetter(id_value,password_value,*stuffs):
    for stuff in stuffs:
        stuff.destroy()
    ids = id_value.get() + ".jpg"
    password = password_value.get()
    with open('data.json') as json_file:
        data = json.load(json_file)
        temp = data['details']
        for values in temp:
            if(values["id"] == ids and values["password"] == password):
                id_value.destroy()
                password_value.destroy()
                project = cv2.imread("Projected/" + ids)
                cv2.imshow("ClickCard Result" , project)
                window.destroy()
                cv2.waitKey(0)
                cv2.destroyAllWindows()
```

This sample code gets the id and password from the user and checks whether the data on the server are the same and the authenticate for the data to be shown in the result.

Fig 6.1 ENTER THE DETAILS

6.2 HOME LOGIN:

It is a page where we can get accessed with upload credentials and display card feature. If we want to upload the details then we can go on with the upload credentials or else we can use the display card feature where we can get the click card easily.

ALGORITHM:

Step 1: Start

Step 2: Create a window to get the user to choose the module they want to use.

Step 3: Display the buttons ‘Upload Card Details’ and ‘Display Card’.

Step 4: Link the buttons to their functions.

Step 5: Stop.

CODE:

```
def home(*stuffs):
    for stuff in stuffs:
        stuff.destroy()


    upload = Button(window, text="Upload Card Details", command=lambda:
                uploader(upload,display),height = 2, width = 20)
    upload.grid(column=1, row=0,padx = 100,pady = 60)


    display = Button(window, text="Display Card", command=lambda:
                displayer(upload,display),height = 2, width = 20)
    display.grid(column=1, row=1,padx = 100,pady = 20)
```

The sample code provides the first and foremost path to other modules using the Tkinter python module.

Fig 6.2 Home Login

6.3 UPLOAD FILES:

Here the major part is that the uploading of the files like images, videos and much more can be uploaded to the website. We have made the card as public or as private by giving the password. If we want to make the card as public then there is no password is required. Then the QR code generator will be generate a specific QR code for the user so that the QR code can be get accessed by the customer easily. At first it takes to the page where we can browse the file which may be in computer or some online method. Hence after that we have to browse and upload the file to browser.Then it takes to the page where it shows that the opened file is to be made as a click card as a public or private mode. And after that a specific QR code is generated for the login credentials. Hence the ID, specific password, the QR data is generated for the particular user. Then all the generated data is stored and the data uploaded is stored.

ALGORITHM:

Step 1: Start

Step 2: Generate the QR code and store it in the server.

Step 3: Get the information to be displayed from the user.

Step 4: Store it in the server for retrieval of the information.

Step 5: Stop

CODE:

```
def browseFiles(label_file_explorer,button_explore):

   filename = filedialog.askopenfilename(initialdir = "/",title = "Select a
        File",filetypes = (("Image files","*.jpg,*.png"),("Video
files","*.mp4*"),("all files","*.*")))

        label_file_explorer.configure(text="File Opened: "+filename)

        prepare_public = Button(window, text="Make Card public",
        command=lambda:
        make(filename,0,button_explore,prepare_public, prepare_private,
        label_file_explorer),height = 2, width = 20)
```

```
prepare_public.grid(column=1, row=2,padx = 100,pady = 20)


prepare_private = Button(window, text="Make Card private",
command=lambda:
make(filename,1,button_explore,prepare_public, prepare_private,
label_file_explorer),height = 2, width = 20)

prepare_private.grid(column=1, row=3,padx = 100)
```

As here, the sample code uses the file exploration to browse and select the file that is to be stored in the database and retrieved to others using the Clickcard system.

Fig 6.6.3 MAKE AS PUBLIC OR PRIVATE

6.4 RESULT:

Hence the Module of the project click card can be given with three Modules and with the specific activities of each module can be given.

ALGORITHM:

Step 1: Start.

Step 2: Define the QR scanner through which the information can be retrieved.

Step 3: Then Resize the QR code within the bound box.

Step 4: Retrieve the content stored in the QR code.

Step 5: Then Resize the stored information (Image) and then to be Displayed.

Step 6: Stop.

CODE:

```
def performAugmentation(id,qrcode,img):
        pts = np.array([qrcode.polygon],np.int32)
        pts = pts.reshape(-1,1,2)
        cv2.polylines(img,[pts],True,(255,0,255),5)
        cv2.imshow("Bounding Box", img)
        pts2 = qrcode.rect

        project = cv2.imread("Projected/" + id)
        project = cv2.resize(project, (pts2.width, pts2.height))
        cv2.imshow("Resized Image",project)
        try:
                final = img.copy()
                final[pts2.top:pts2.top+pts2.height,pts2.left:pts2.left+pts2.width] =
```

```
            project
            cv2.imshow("ClickCard Result",final)
        except:
            pass
```

As here, the image that is provided by the user is resized based on the live capture and the QR is replaced by the image and shown as the result of the project.

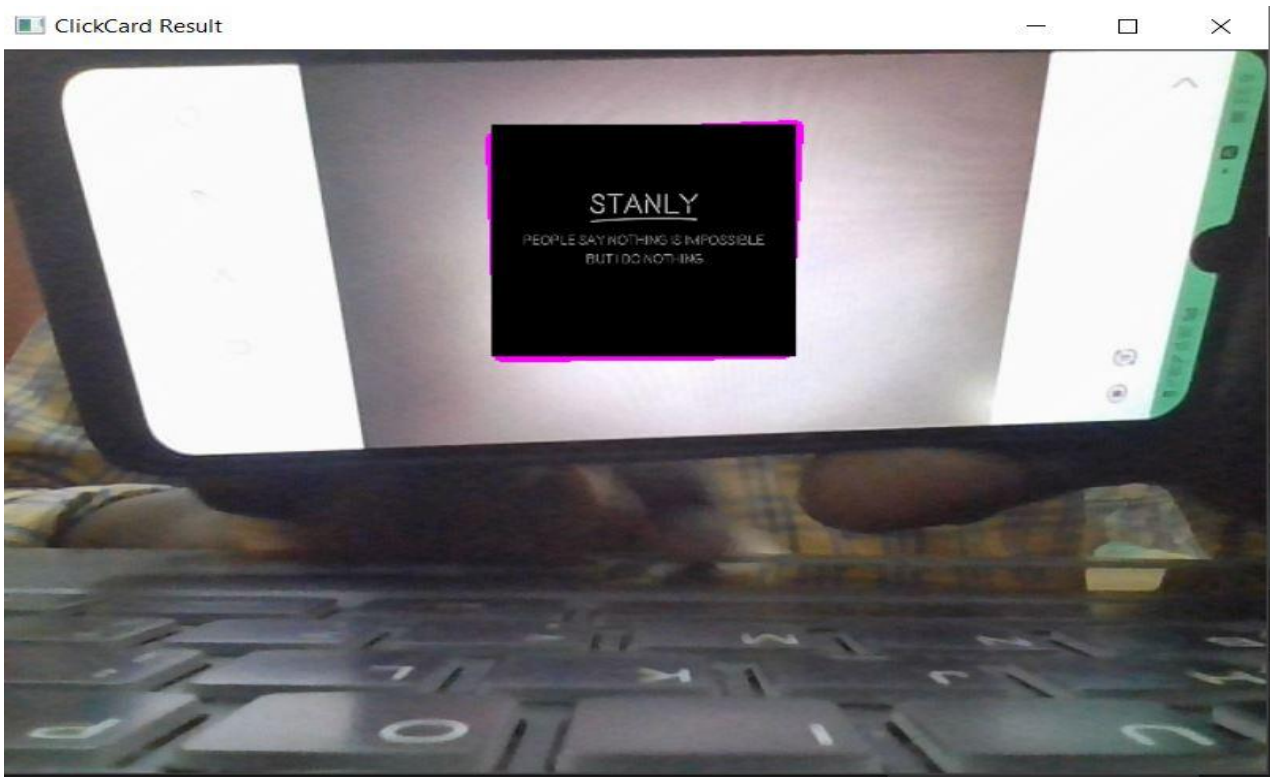


Fig 6.5.4 Viewing the uploaded image

CHAPTER 7
IMPLEMENTATION

7. IMPLEMENTATION

7.1 SAMPLESCREENSHOTS:

HOME LOGIN:

Fig 7.1 HOME LOGIN

DISPLAY THE DETAILS:

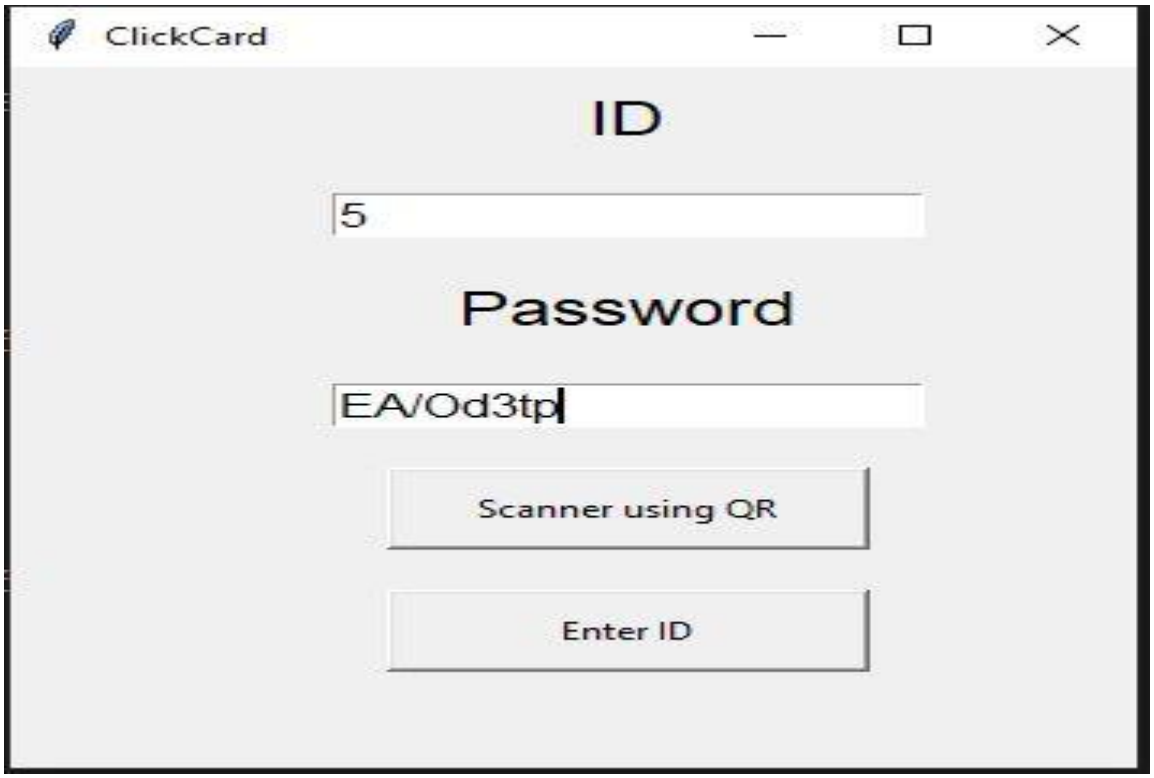


Fig 7.2.1 ENTERING THE PRIVATE DETAILS

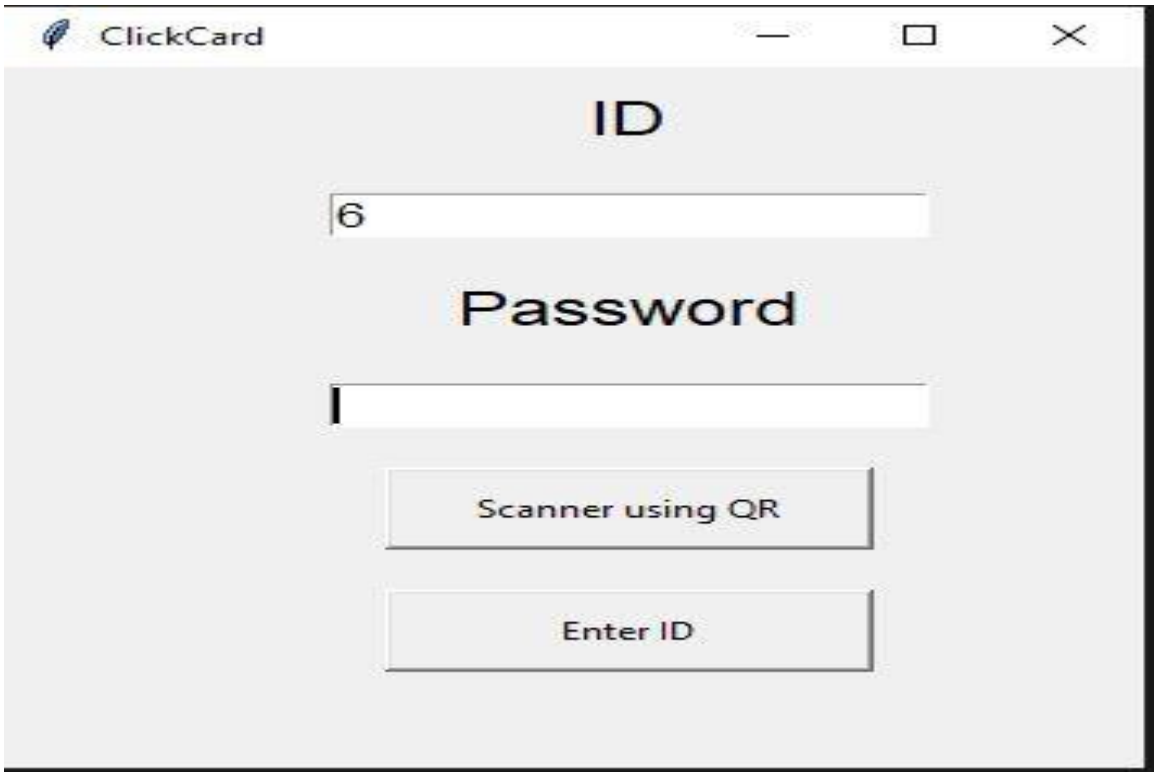


Fig 7.2.2 ENTERING THE PUBLIC DETAILS

Fig 7.3 ID RESULT

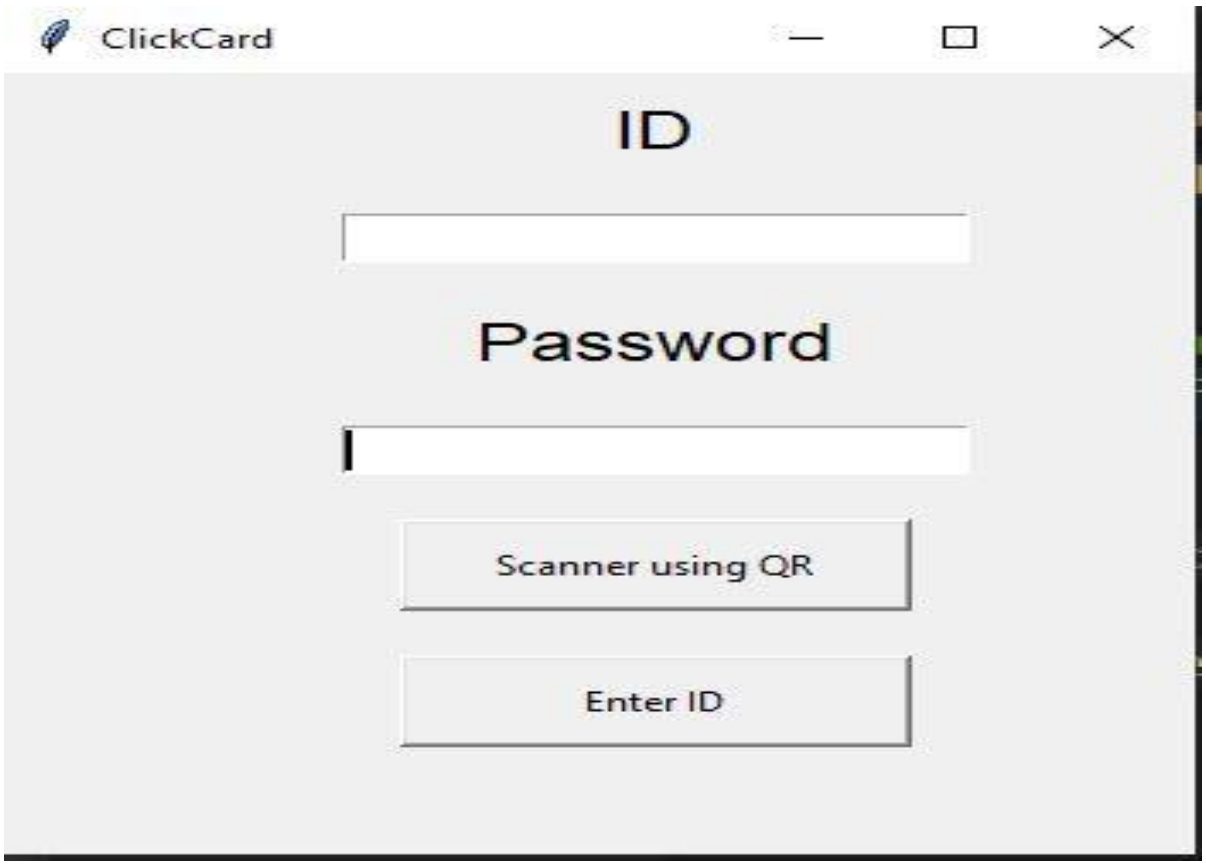


Fig 7.4.1 QR SCANNER ENTERING PUBLIC

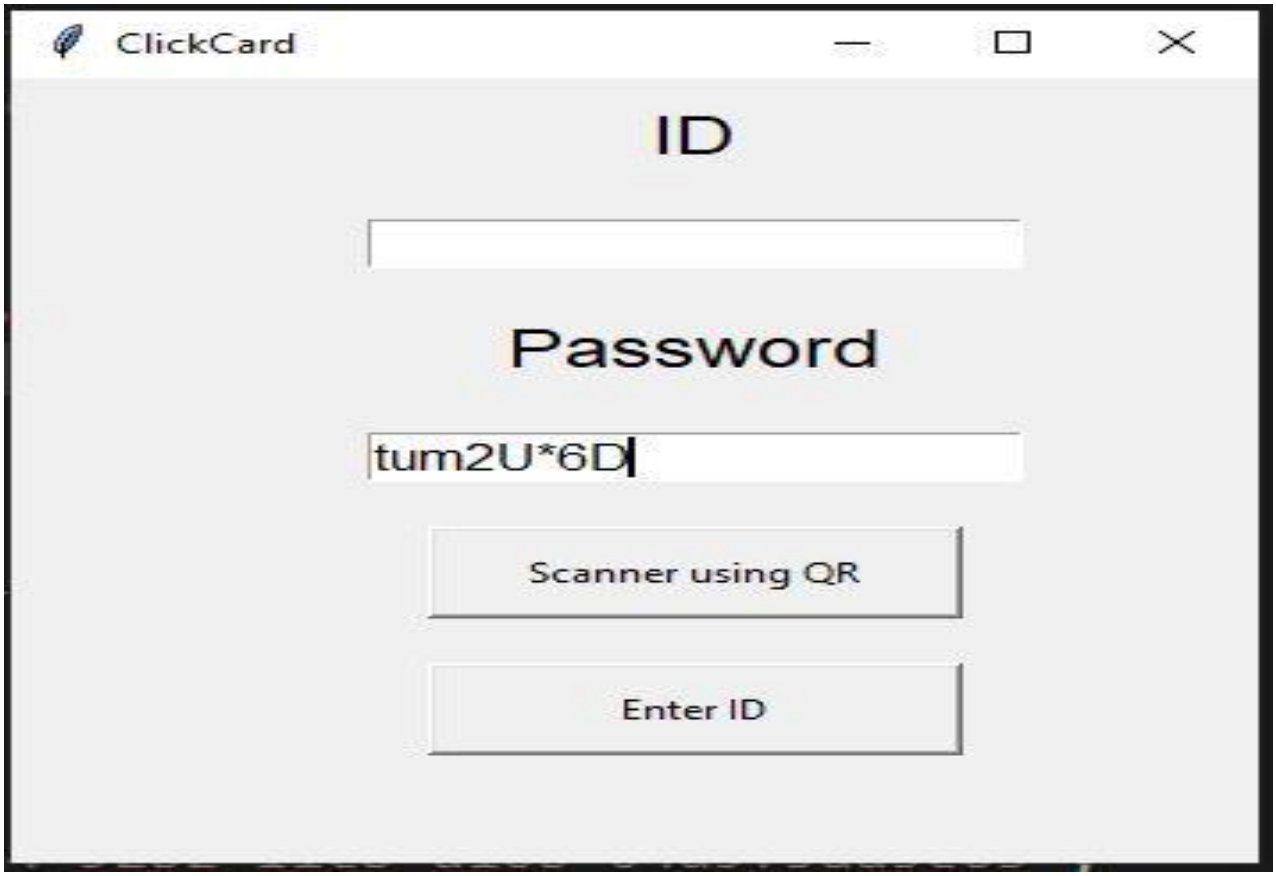


Fig 7.4.2 QR SCANNER ENTERING PRIVATE

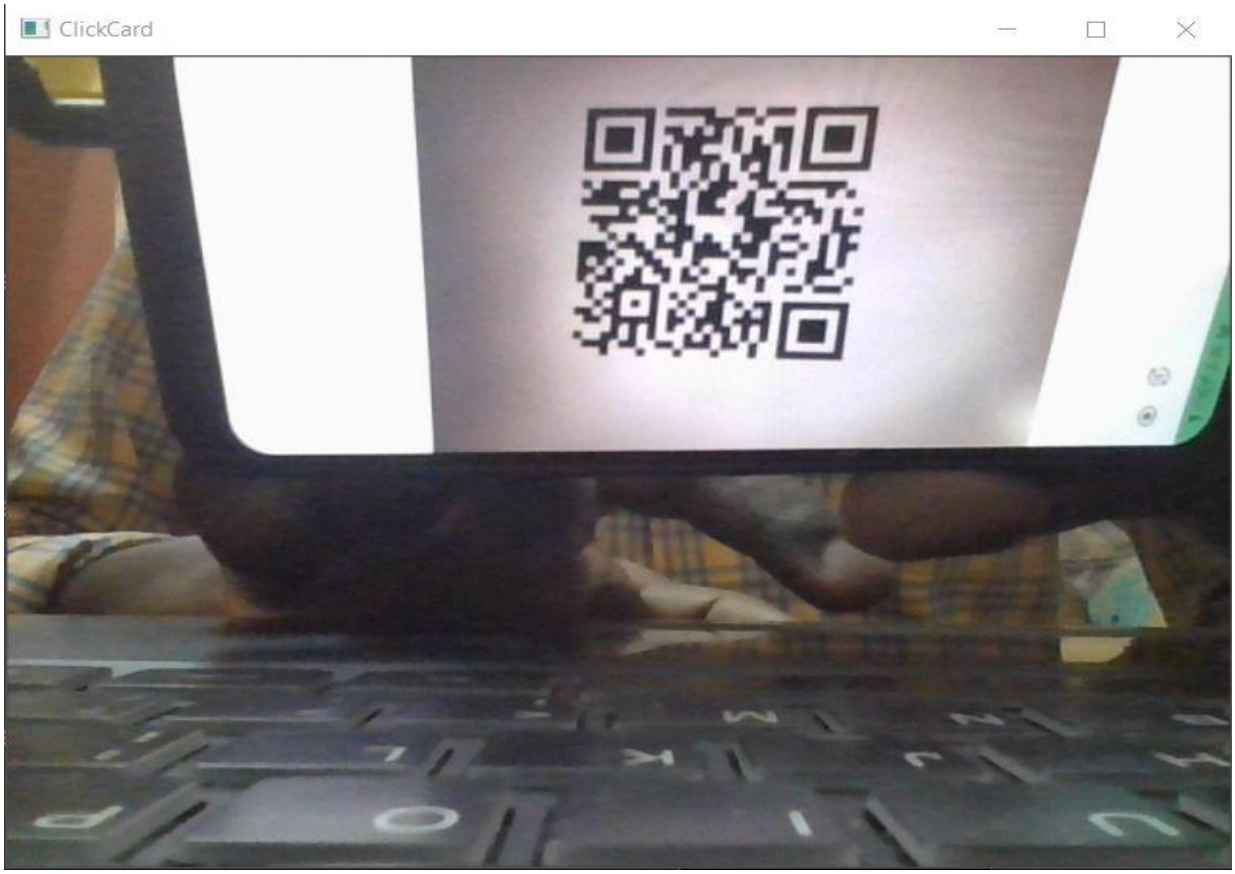


Fig 7.5.1QR CODE SCANNING

Fig 7.5.2POINTING THE QR CODE

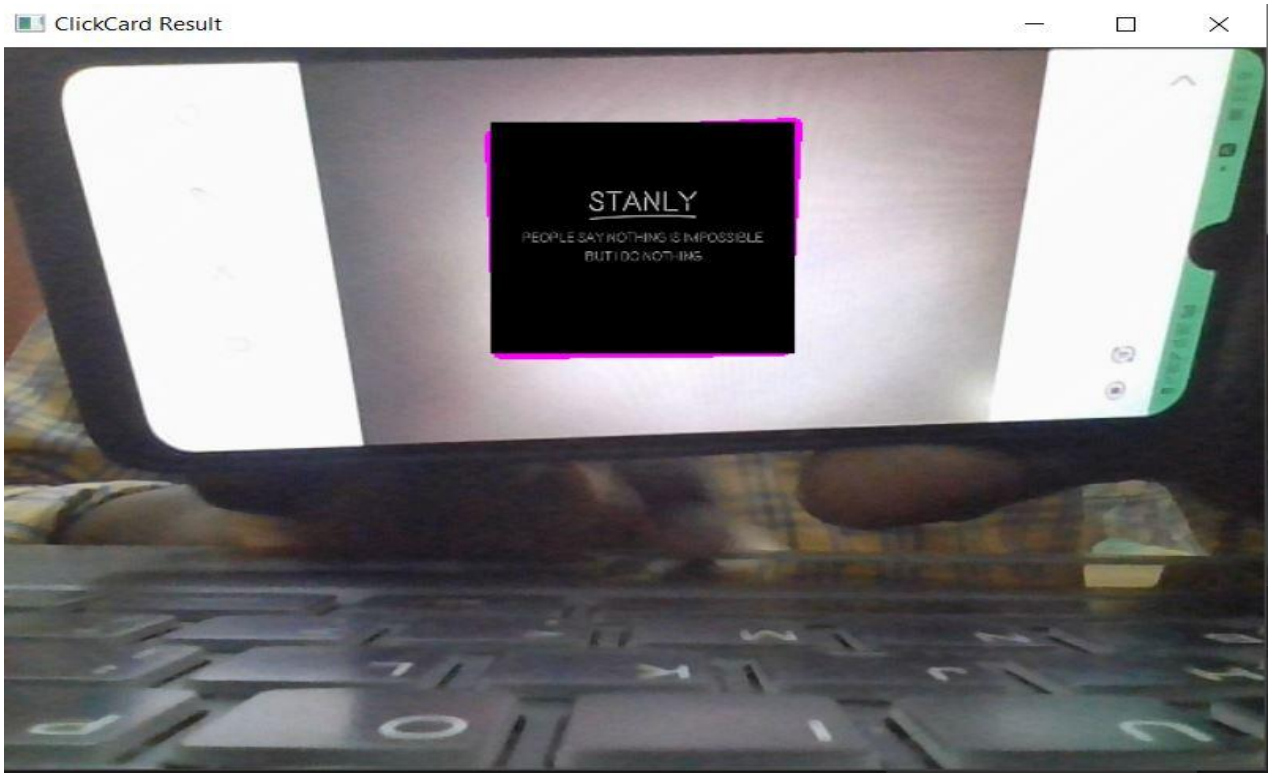


Fig 7.5.3VIEWING THE UPLOADED IMAGE

UPLOAD FILES:

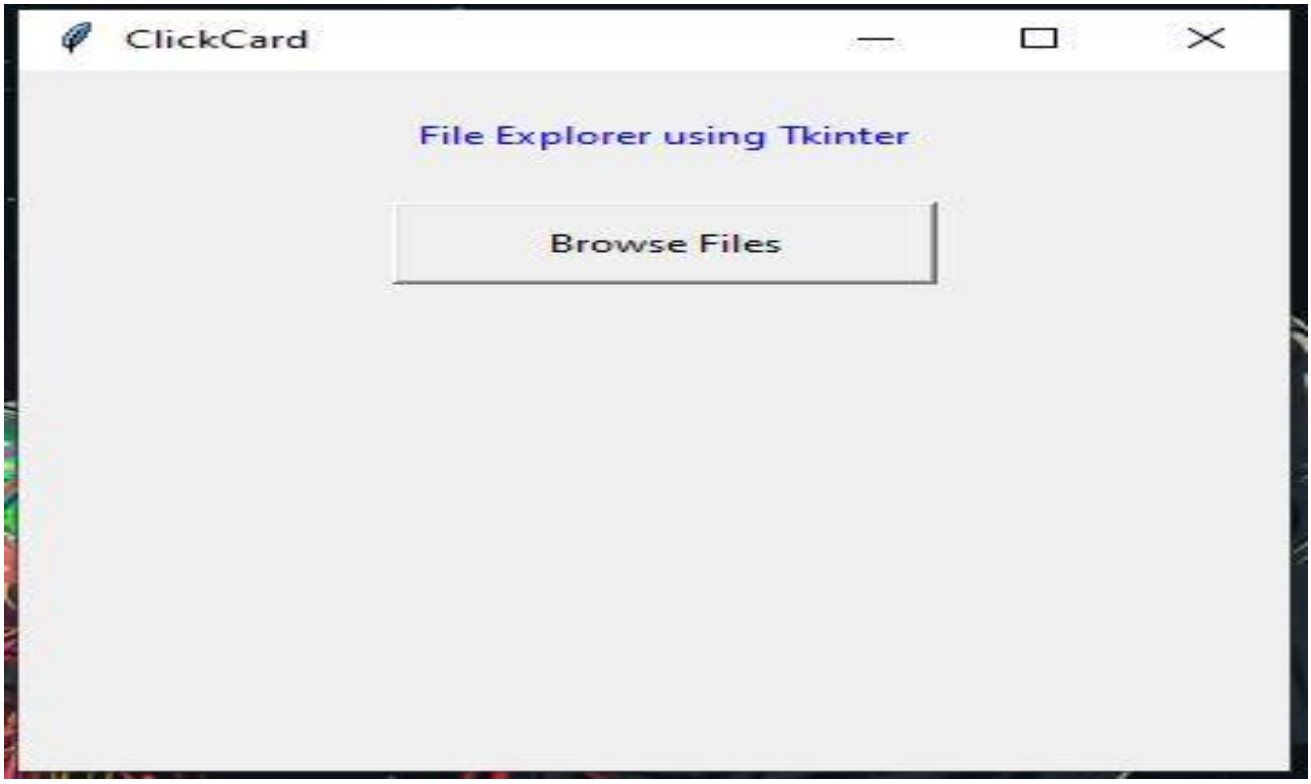


Fig 7.6.1 UPLOAD THE FILES

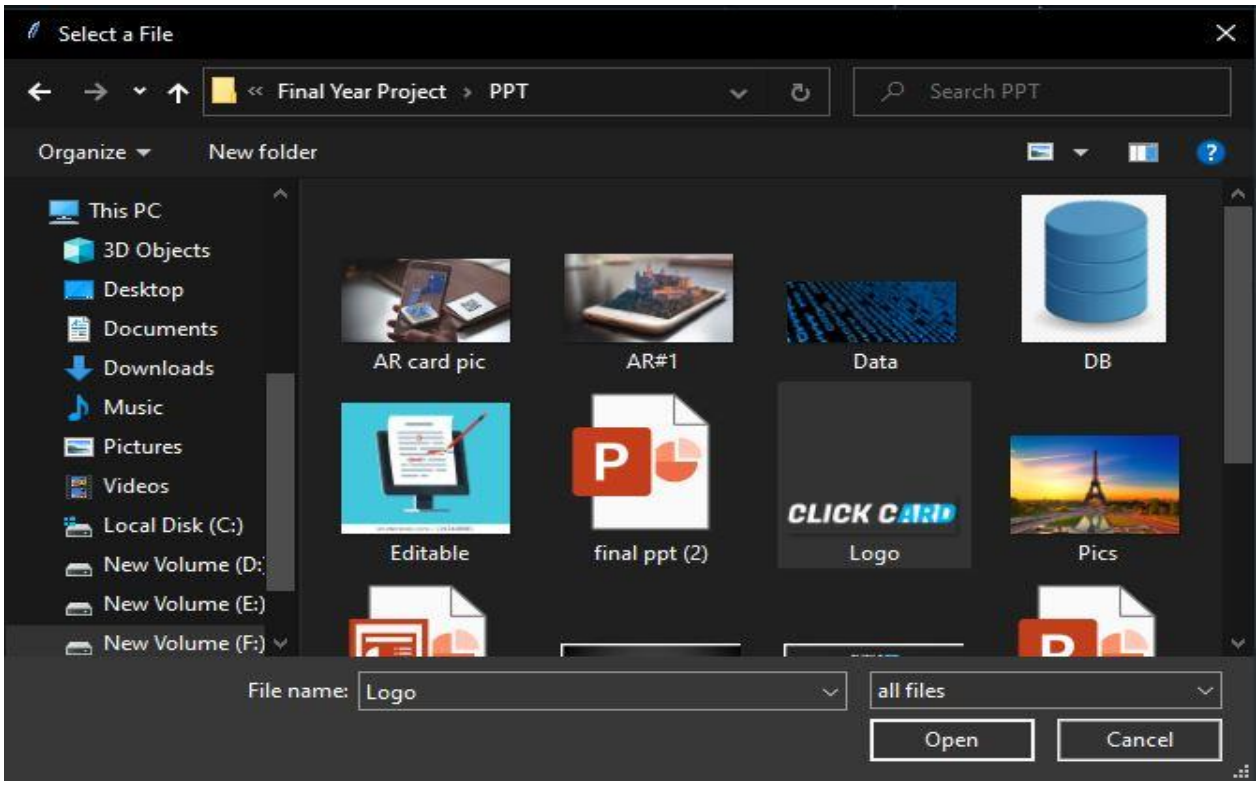


Fig 7.6.2 BROWSING THE FILE FROM FILE MANAGER

Fig 7.6.3 UPLOADED IMAGE

Fig 7.6.4 MAKE AS PUBLIC OR PRIVATE

Fig 7.6.5 QR CODE GENERATED

Fig 7.7 GENERATE ID AND PASSWORD FOR PRIVATE

Fig 7.8 GENERATE ID AND PASSWORD FOR PUBLIC

7.2 SAMPLE CODING

```
from tkinter import *
from tkinter import filedialog
import shutil
import qrcode
import json
import random
import array
from PIL import ImageTk,Image
import numpy as np
import cv2
from pyzbar.pyzbar import decode
import uuid
import time

# Displayer Page
def displayer(*stuffs):
    for stuff in stuffs:
        stuff.destroy()

    id = Label(window,text = "ID",font=("Arial", 18))
    id.grid(column=1, row=0,padx = 100,pady = 10)
```

```
    id_value = Entry(window,font= ("Arial",12))
    id_value.grid(column=1, row=1,padx = 100,pady = 10)

    password = Label(window,text = "Password",font=("Arial", 18))
    password.grid(column=1, row=2,padx = 100,pady = 10)

    password_value = Entry(window,font= ("Arial",12))
    password_value.grid(column=1, row=3,padx = 100,pady = 10)

    qr_scan = Button(window, text= "Scanner using QR",
command=lambda:
qrscanner(password_value,id_value,id,password,qr_scan,enter_id),hei
ght = 2, width = 20)
    qr_scan.grid(column=1, row=4,padx = 100,pady = 10)

    enter_id = Button(window, text= "Enter ID", command=lambda:
idgetter(id_value,password_value,id,password,qr_scan,enter_id),heig
ht = 2, width = 20)
    enter_id.grid(column=1, row=5,padx = 100,pady = 10)

def loadProjected_qr(qr_value,password):
    with open('data.json') as json_file:
        data = json.load(json_file)
        temp = data['details']
        for values in temp:
```

```
            if(values["qr_data"] == qr_value and values["password"] ==
password):
                return values["id"]
    return -1
```

 # Showing Augmentation

```
def performAugmentation(id,qrcode,img):
    pts = np.array([qrcode.polygon],np.int32)
    pts = pts.reshape(-1,1,2)
    cv2.polylines(img,[pts],True,(255,0,255),5)
    cv2.imshow("Bounding Box", img)
    pts2 = qrcode.rect

    project = cv2.imread("Projected/" + id)
    project = cv2.resize(project, (pts2.width, pts2.height))
    cv2.imshow("Resized Image",project)
    try:
        final = img.copy()
        final[pts2.top:pts2.top+pts2.height,pts2.left:pts2.left+pts2.width]
= project
        cv2.imshow("ClickCard Result",final)
    except:
        pass
```

```
def qrscanner(value,*stuffs):
    for stuff in stuffs:
        stuff.destroy()
    password = value.get()
    value.destroy()

    vid = cv2.VideoCapture(0)
    while(True):
        ret, frame = vid.read()
        cv2.imshow('ClickCard', frame)
        decodedObjects = decode(frame)
        for obj in decodedObjects:
            id = loadProjected_qr(obj.data.decode('utf-8'),password)
            if(id != -1):
                performAugmentation(id,obj,frame.copy())
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    vid.release()
    cv2.destroyAllWindows()
    window.destroy()
```

```
def idgetter(id_value,password_value,*stuffs):
    for stuff in stuffs:
        stuff.destroy()
    ids = id_value.get() + ".jpg"
    password = password_value.get()
    with open('data.json') as json_file:
        data = json.load(json_file)
        temp = data['details']
        for values in temp:
            if(values["id"] == ids and values["password"] == password):
                id_value.destroy()
                password_value.destroy()
                project = cv2.imread("Projected/" + ids)
                cv2.imshow("ClickCard Result" , project)
                window.destroy()
                cv2.waitKey(0)
                cv2.destroyAllWindows()
```

 # Main Home page

```
def home(*stuffs):
```

```
    for stuff in stuffs:
        stuff.destroy()
    upload = Button(window, text="Upload Card Details",
command=lambda: uploader(upload,display),height = 2, width = 20)
    upload.grid(column=1, row=0,padx = 100,pady = 60)


    display = Button(window, text="Display Card", command=lambda:
displayer(upload,display),height = 2, width = 20)
    display.grid(column=1, row=1,padx = 100,pady = 20)
```

 # Upload Page

```
def uploader(*stuffs):
    for stuff in stuffs:
        stuff.destroy()
    label_file_explorer = Label(window,text = "File Explorer using
Tkinter",width = 50, height = 4,fg = "blue")
    label_file_explorer.grid(column = 1, row = 0)
```

```
    button_explore = Button(window,text = "Browse Files",command
=lambda: browseFiles(label_file_explorer,button_explore),height = 2,
width = 20)

    button_explore.grid(column = 1, row = 1)
```

 # File Browsing Selector

```
def browseFiles(label_file_explorer,button_explore):

    filename = filedialog.askopenfilename(initialdir = "/",title = "Select
a File",filetypes = (("Image files","*.jpg,*.png"),("Video
files","*.mp4*"),("all files","*.*")))

    label_file_explorer.configure(text="File Opened: "+filename)


    prepare_public = Button(window, text="Make Card public",
command=lambda:
make(filename,0,button_explore,prepare_public,prepare_private,label
_file_explorer),height = 2, width = 20)

    prepare_public.grid(column=1, row=2,padx = 100,pady = 20)


    prepare_private = Button(window, text="Make Card private",
command=lambda:
make(filename,1,button_explore,prepare_public,prepare_private,label
_file_explorer),height = 2, width = 20)

    prepare_private.grid(column=1, row=3,padx = 100)
```

 # Making the card

```
def make(filename,privatize,*stuffs):
    for stuff in stuffs:
        stuff.destroy()
    id = getId(filename)
    password = ""
    shutil.copyfile(filename,"Projected/" + id)

    if privatize:
        password = generatePassword()
    data = generateQR(id)
    filesaver(data, password)
    show(id,password)
```

 # Generating ID for the card

```
def getId(filename):
    with open('data.json') as json_file:
        data = json.load(json_file)
        temp = data['details']
        id = int(temp[len(temp)-1]["id"][:-4]) + 1
    return str(id) + ".jpg"
```

 # Generating Password for the Private cards

```
def generatePassword():
    MAX_LEN = 8

    DIGITS = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

    LOCASE_CHARACTERS = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h',
                         'i', 'j', 'k', 'm', 'n', 'o', 'p', 'q',
                         'r', 's', 't', 'u', 'v', 'w', 'x', 'y',
                         'z']

    UPCASE_CHARACTERS = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H',
                         'I', 'J', 'K', 'M', 'N', 'O', 'p', 'Q',
                         'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y',
                         'Z']

    SYMBOLS = ['@', '#', '$', '%', '=', ':', '?', '.', '/', '|', '~', '>',
           '*', '(', ')', '<']

    COMBINED_LIST = DIGITS + UPCASE_CHARACTERS +
LOCASE_CHARACTERS + SYMBOLS

    rand_digit = random.choice(DIGITS)
```

```
    rand_upper = random.choice(UPCASE_CHARACTERS)
    rand_lower = random.choice(LOCASE_CHARACTERS)
    rand_symbol = random.choice(SYMBOLS)

    temp_pass = rand_digit + rand_upper + rand_lower + rand_symbol


    for x in range(MAX_LEN - 4):
        temp_pass = temp_pass + random.choice(COMBINED_LIST)

        temp_pass_list = array.array('u', temp_pass)
        random.shuffle(temp_pass_list)

    password = ""
    for x in temp_pass_list:
            password = password + x
    return password
```

 # Generate QR code for the card

```
def generateQR(id):
    input_data = generateData()
    qr = qrcode.QRCode(
            version=1,
```

```
        box_size=10,
        border=5)
    qr.add_data(input_data)
    qr.make(fit=True)
    img = qr.make_image(fill='black', back_color='white')
    img.save("QR/"+ id)
    return input_data


# Generating Unique Data for the QR code for the Card
def generateData():
    return str(uuid.uuid1())

# Saving the data that are generated for the Card
def filesaver(qr_data, password):
    with open('data.json') as json_file:
        data = json.load(json_file)

        temp = data['details']
        id = len(temp)+1
        y = {"id": str(id) + ".jpg", "qr_data":qr_data ,"password":
password}
        temp.append(y)
```

```
        write_json(data)
        return id


# Writing the Json File
def write_json(data, filename='data.json'):
    with open(filename,'w') as f:
        json.dump(data, f, indent=4)


# Showing the QR Generated and password and id
def show(id,password):
    card_id = Label(window,text = "ID",font=("Arial", 18)).place(x = 40,y = 60)
    card_id_value = Label(window,text = id[:-4],font=("Arial", 18)).place(x = 140,y = 60)
    card_password = Label(window,text = "Password",font=("Arial", 18)).place(x = 120,y = 100)
    card_password_value = Label(window,text = password,font=("Arial", 18)).place(x = 120,y = 140)
    img = cv2.imread("QR/" + id,0)
    cv2.imshow('QR Code',img)
    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

 # Main Running Function

```
if __name__ == "__main__":
    window = Tk()
    window.geometry('350x350')
    window.title("ClickCard")

    home()

    window.mainloop()
```

CHAPTER 8

SYSTEM TESTING AND IMPLEMENTATION

8. SYSTEM TESTING AND IMPLEMENTATION

8.1 INTRODUCTION

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design, and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply both strategic to both large and small-scale systems.

8.2 STRATEGIC APPROACH TO SOFTWARE TESTING

The software engineering process can be viewed as a spiral. Initially, system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behaviour, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software we spiral in along streamlines that decrease the level of abstraction at each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in the source code. Testing progress is done by moving outward along the spiral to integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally, we arrive at system testing, where the software and other system elements are tested as a whole.

8.3 ALPHA TESTING

This is a form of internal acceptance testing performed mainly by the in-house software QA and testing teams. Alpha testing is the last testing done by the test teams at the development site after the acceptance testing and before releasing the software for the beta test. Alpha testing can also be done by the potential users or customers of the application. But still, this is a form of in-house acceptance testing.

8.4 BETA TESTING

This is a testing stage followed by the internal full alpha test cycle. This is the final testing phase where the companies release the software to a few external user groups outside the company test teams or employees. This initial software version is known as the beta version. Most companies gather user feedback in this release.

8.5 BLACK BOX TESTING

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code

structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

8.6 WHITE BOX TESTING

White box testing techniques analyze the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing.

Working process of white box testing:

- **Input:** Requirements, Functional specifications, design documents, source code.
- **Processing:** Performing risk analysis for guiding through the entire process.
- **Proper test planning:** Designing test cases so as to cover entire code. Execute rinse-repeat until error-free software is reached. Also, the results are communicated.
- **Output:** Preparing final report of the entire testing process.

CHAPTER 9

CONCLUSION AND FUTURE ENCHANCEMENT

9. CONCLUSION AND FUTURE ENCHANCEMENT

9.1 CONCLUSION

In this project I have dynamically used Augmented reality for the better viewing and surfing experience. Uploading the file by the customer and which is viewed by the end user is an innovative idea. Images and Videos can be viewed by the user which is uploaded by the customer. Our main aim is to provide the user a great experience of buying new things from home with the help of internet. And that to be viewed by the augmented reality manner is more beautiful.

FUTURE ENHANCEMENT:

Our Future plan is to add the video augmentation and audio augmentation. Which will be more attractive and will improve the advertisement to be viewed by the people. Hence the business will also be improved in a faster rate and at a easy mode of purchasing new things.

10. REFERENCES

1) Trudy Kawal Ravindra Sahadev Arun K Srinivasan Aseem R Shukla, 'Robotic surgery in infants and children: an argument for smaller and fewer incisions', [Aug 2020]

2) Cristina Zirafa,Gaetano Romano,Teresa Hung Key,Franca Melfi, 'The evolution of robotic thoracic surgery using Augumented reality', [Mar 2019]

3) Ignacio Cifuentes,Bruno Dagnino,María Carolina Salisbury,Daniela Maldonado, 'Augmented reality and dynamic infrared thermography for perforator mapping in the anterolateral thigh', [May 2018]

4) Max J Greenfield,Joshua Luck,Michael L Billingsley,Nadine Hachach-Haram, 'Demonstration of the Effectiveness of Augmented Reality Telesurgery in Complex Hand Reconstruction in Gaza' [Mar 2018]

5) Philip Pratt,Matthew Ives,Graham Lawton,Dimitri Amiras, 'Through the HoloLens™ looking glass: augmented reality for extremity reconstruction surgery using 3D vascular models with perforating vessels' [Dec 2018]

6) Petr Vávra,Jan Roman,Pavel Zonca,Ahmed El-Gendi, 'Recent Development of Augmented Reality in Surgery: A Review, Article' [Aug 2017]

7) Daisuke Mitsuno,Koichi Ueda,Tomoki Itamiya,Yuki Otsuki'Intraoperative Evaluation of Body Surface Improvement by an Augmented Reality System That a Clinician Can Modify' [Aug 2017]

8) Youngjun Kim,Hannah Kim,Yong Oock Kim 'Virtual Reality and Augmented Reality in Plastic Surgery: A Review' [May 2017]

9) Soh Nishimoto,Maki Tonooka,Kazutoshi Fujita,Masao Kakibuchi 'An augmented reality system in lymphatico-venous anastomosis surgery' [May 2017]

10) Li Lin,Yunyong Shi,Andy Tan,Gang Chai, 'Mandibular angle split osteotomy based on a novel augmented reality navigation using specialized robot-assisted arms—A feasibility study' [Nov 2016]