

Integrating Multiple Security Features in an Online Banking Platform

A major project report submitted in partial fulfilment of the requirement
for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

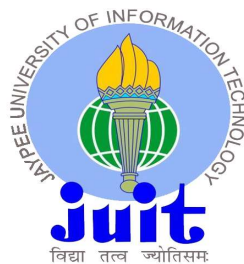
Submitted by

Harshit Upadhyay (201436)

Rushil Wadhawan (201233)

Under the guidance & supervision of

Dr. Deepak Gupta



**Department of Computer Science & Engineering and
Information Technology**

Jaypee University of Information Technology,

Waknaghat, Solan - 173234 (India)

CERTIFICATE

This is to certify that the work which is being presented in the project report titled “Integrating multiple security features in an Online Banking Platform” in partial fulfilment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Wagnaghat is an authentic record of work carried out by “Harshit Upadhyay, 201436” and “Rushil, 201233” during the period from August 2023 to May 2024 under the supervision of Dr. Deepak Gupta, Department of Computer Science and Engineering, Jaypee University of Information Technology, Wagnaghat.

Harshit Upadhyay

(201436)

Rushil Wadhawan

(201233)

The above statement made is correct to the best of my knowledge.

Dr. Deepak Gupta

Assistant Professor (Senior Grade)

Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Wagnaghat

CANDIDATE'S DECLARATION

We hereby declare that the work presented in this report entitled '**Integrating multiple security features in an Online Banking Platform**' in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Wagnaghat is an authentic record of our own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Deepak Gupta** (Assistant Professor (Senior Grade), Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature with Date)

Harshit Upadhyay

201436

(Student Signature with Date)

Rushil Wadhawan

201233

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature with Date)

Dr. Deepak Gupta

Assistant Professor (Senior Grade)

Computer Science & Engineering and Information Technology

Dated:

ACKNOWLEDGEMENT

Firstly, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the project work successfully.

We are really grateful and wish our profound indebtedness to Supervisor **Dr. Deepak Gupta, Assistant Professor (Senior Grade)**, Department of CSE Jaypee University of Information Technology, Wanknaghat. Deep Knowledge & keen interest of our supervisor in the field of “**Information Security**” has helped us to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism and valuable advice along with reading many inferior drafts and correcting them at all stage have made it possible for us to complete this project.

We would like to express our heartiest gratitude to **Dr. Deepak Gupta**, Department of CSE, for his kind help to finish our project.

We would also generously welcome each one of those individuals who have helped us straight forwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated our undertaking.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

Harshit Upadhyay (201436)

Rushil Wadhawan (201233)

TABLE OF CONTENTS

CERTIFICATE	i
CANDIDATE’S DECLARATION	ii
ACKNOWLEDGEMENT	iii
LIST OF ABBREVIATIONS	vi
LIST OF FIGURES	vii-viii
LIST OF TABLES	ix
ABSTRACT	x
1 INTRODUCTION	1-8
1.1 Introduction	1
1.2 Problem Statement	2-3
1.3 Objectives	4
1.4 Significance and motivation of the project report	5-6
1.4.1 Significance	5-6
1.4.2 Motivation	6
1.5 Organization of project report	6-8
2 LITERATURE SURVEY	9-14
2.1 Overview of relevant literature	9-12
2.1.1 Introduction	9
2.1.2 A summary of the relevant papers	9-12
2.2 Key gaps in the literature	12-14
3 System Development	15-40
3.1 Requirements and Analysis	15-19
3.1.1 Functional Requirements	15
3.1.2 Non-Functional Requirements	16
3.1.3 Hardware Requirements	16

3.1.4	Software Requirements	17-19
3.1.4.1	Languages used	17
3.1.4.2	Libraries used	18
3.1.4.3	Tools used	18-19
3.2	Project Design and Architecture	19-22
3.2.1	Methodology	19-22
3.3	Data Preparation.....	22-23
3.4	Implementation	23-39
3.4.1	Downloading Essential Software	23-24
3.4.2	OTP Authentication using Twilio	24-25
3.4.3	AES-128 Algorithm	26-28
3.4.4	Visual Cryptography	28-33
3.4.5	Cryptographic Hashing	33-36
3.4.6	Google Recaptcha	36-37
3.4.7	Resend OTP Feature	37-38
3.4.8	Sending Email Confirmation	39
3.4.9	Added MPIN to view Account Balance	39
3.5	Key Challenges	40
4	Testing	41-46
4.1	Testing Strategy	41
4.2	Test Cases and Outcomes	42-46
5	Results and Evaluation	47-50
5.1	Results	47-50
6	Conclusions and Future Scope	51-52
6.1	Conclusion	51
6.2	Future Scope	52
	REFERENCES	53-55
	APPENDIX.....	56

LIST OF ABBREVIATIONS

AES	Advanced Encryption Standard
DES	Data Encryption Standard
OTP	One-time Password
UML	Unified Modeling Language
XAMPP	X-operating system, Apache, MySQL, PHP, Perl
XML	Extensible Markup Language
MySQL	My Structured Query Language
GD	Graphic Draw
HTML.....	Hypertext Markup Language
CSS.....	Cascading Style Sheets
PHP.....	Hypertext Preprocessor
API	Application Program Interface
SMTP	Simple Mail Transfer Protocol
RDBMS	Relational Database Management System
REST	Representational State Transfer
TLS	Transport Layer Security
2FA.....	Two Factor Authentication
VC.....	Visual Cryptography

LIST OF FIGURES

1.1	Online Banking Vulnerabilities	2
3.1	Flow Chart of the Project	21
3.2	Class Diagram..	22
3.3	XAMPP Control Panel	24
3.4	Different columns of online banking platform table	24
3.5	Twilio API Dashboard.....	25
3.6	Code to send OTP using Twilio	25
3.7	Structure of AES algorithm.....	27
3.8	AES-128 encryption	27
3.9	AES-128 decryption	28
3.10	Representation of Shares Generation	29
3.11	Visual Cryptography Method of black and white Pixels	29
3.12	Initial Captcha	30
3.13	Shares in which captcha is divided	30
3.14	Captcha displayed after Decryption	30
3.15	Code for generating captcha image for Encryption	31
3.16	Code for breaking original image into two shares that is encryption	32
3.17	Code for combining the two shares to decrypt the message	33
3.18	Cryptographic Hash Function	34
3.19	Attack against Hash Function	34
3.20	Bcrypt Hashing flow diagram	35
3.21	Code for Bcrypt Hashing Algorithm	36
3.22	ReCAPTCHA being used in the login page	37
3.23	Code for the ReCAPTCHA authentication	37
3.24	Resend OTP Button	38
3.25	Code for Resend OTP	38
3.26	Notification received when user receives the money	39
3.27	Code for MPIN	39
4.1	Registration Page of Online Banking Platform	42

4.2	Error displayed as same phone number is registered twice	42
4.3	Login Page of the Online Banking Platform	43
4.4	Error Displayed as wrong OTP entered	43
4.5	Details are stored in the Database after successful Registration	43
4.6	Money transfer authentication area to upload share	44
4.7	To enter details to send money after uploading the share	44
4.8	Transaction history page showing transaction details	45
4.9	To enter MPIN	45
4.10	Email to notify Login Information	45
4.11	Credit email sent to receiver	46
5.1	OTP received on the registered mobile number	47
5.2	Encrypted value Captcha Text stored in the Database	48
5.3	Share 1 received on the Email of the user	48
5.4	Password hashed through Bcrypt	49
5.5	Confirmation Email	49
5.6	Showing Balance Correctly	49
5.7	Login Email	50
5.8	Transaction details stored in database	50

LIST OF TABLES

2.1 Summary of the Relevant Literature	10
--	----

ABSTRACT

Banking refers to the industry and financial activities that are related to the storage, management and utilization of funds. Banks serve as a bridge in order to facilitate the flow of funds between the customers, thus providing a range of services such as deposits, loans, investments and payment processing. It plays a crucial role in the economic development of a country by supporting businesses, enabling individuals to save and borrow, and contributing to overall financial stability. Banking has a very rich history, the earliest known banking practices can be traced back to ancient Mesopotamia, where temples served as financial institutions, storing valuables and offering loans. Ancient Greece and Rome also had early banking systems, with moneylenders and currency exchange services.

The advancement of technology has led to changes in traditional banking system, now most of the banks have gone online and their customers are able to enjoy the majority of the banking services at the comfort of their homes, and they no more need to face the rush and hustle for even doing a small payment but this comfort does not come free of cost, as the attacks on these online banking systems have also increased tremendously. According to a data submitted by the Central Government in the Indian Parliament during the period of June 2018 and March 2022 the Indian banks were hit by 248 successful data breaches that were undertaken by hackers and criminals [1]. This is the official figure, there are many instances of phishing and spoofing attacks which are not reported anywhere. Although the online banking has helped the users in saving their time and efforts but the number of attacks faced on such system is a matter of grave concern.

In order to address these issues, we aim to create a web application for online banking, which will be completely safe from any sort of attacks as multiple security features have been implemented in it.

CHAPTER 1: INTRODUCTION

1.1 INTRODUCTION

In today's world everything is advancing and with this the one thing that is growing rapidly is technology. Keeping this in mind as we consider the growth, we see many problems evolving around this which can be solved by using technology. One such issue is the process of traditional offline banking. Online banking platforms have proved themselves as an essential part of today's financial transactions, providing customers with simplicity and access.

In the traditional way of banking, that is offline banking there are many problems that customers face. The problems that these customers face is firstly, the ease of use, the customer has to be physically present in the bank to perform any operation such as transactions, account opening etc. Secondly offline banking is a very slow process, customers have to wait in queues for a very long time to get their work done which is very tiring. To add on, another problem is that each transaction takes a very long time to process as it takes one to two days. Lastly, offline banking platforms provide only a simple structure that we are following form a long time and there are very less changes to it but this is not the case with online banking.

So, online banking platforms are very feasible for the customers as they can perform the transactions with their own electronic gadgets in their homes. There are many tailored financial products that fulfill customer's wants, preferences and quality expectations. In the past year's the online banking has gained popularity because of transaction security, transaction accuracy and user friendliness. In the online banking platforms, apart from the normal banking environment there are many other operations that the user can perform such stock market investments, shopping from e-commerce websites, all these through an integrated online banking platform.

There are some risks involved in online banking such as phishing, internet scams, malware, pharming, virus etc., which need to be taken care of in order to create a robust online banking platform. In conclusion there are many benefits of using an online banking platform instead of the traditional offline banking as it is more accessible, convenient, and fast. But for the trustworthiness of the online banking platform, it needs to be secure, reliable and user friendly in order to become a strong system without any shortcomings.

1.2 PROBLEM STATEMENT

The growth and advancements in the digital infrastructure has led to the conversion of various traditional systems into online processes and systems, one such traditional system that has been in practice from many years is the banking system. Now it has also been converted into an online platform allowing the customers the feasibility to perform transactions easily on their electronic gadgets.

Nevertheless, it also leads to a whole lot of security concerns, that can become a major problem with the online banking. It can lead to leak of confidential information, integrity of the online banking platform. Some fraudulent activities such as phishing and pharming pose a serious threat to the online banking which results in decrease of trust of users, ultimately resulting to financial losses.

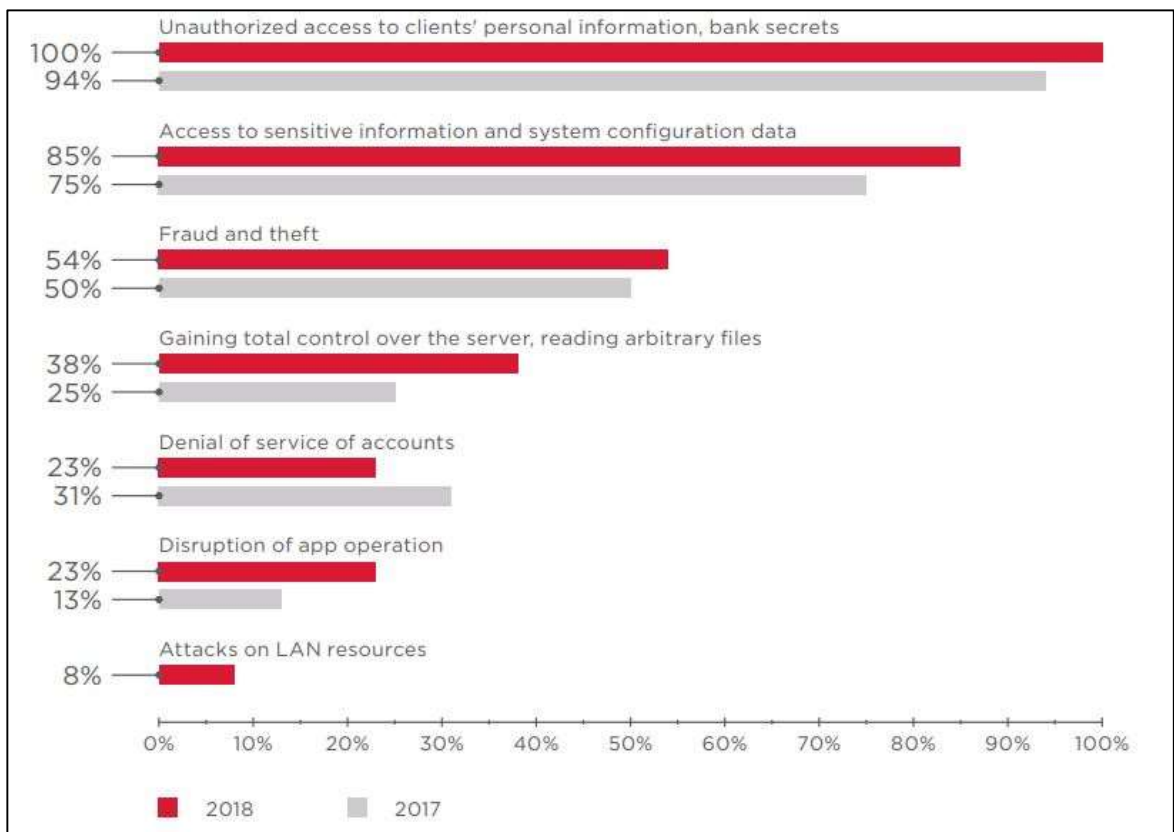


Figure 1.1: Online Banking Vulnerabilities

It affects the privacy of users by leaking their personal details such as name, mobile number, account number, transactions, pin and it further puts the financial integrity of transactions at risks. Due to the weak customer authentication, the risks increase on a higher scale as the hacker gains unauthorized access of the account.

A method for safeguarding the transparency and privacy of the auction system is Visual Cryptography. Visual cryptography includes dividing an image into several shares, each of which only holds a portion of the original image's information. The original image can only be recreated by combining a particular combination of shares, which can then be distributed to various parties. We may make sure that only authorized people have access to sensitive information by utilizing visual cryptography.

There are three types of visual cryptography, first one is (n,n) method where an image broken down into n shares and n shares are needed to be uploaded for successful authorization. Second one is (k,n) method in which an image is broken down into n shares and we need k number of shares for successful authorization. Third is $(2,2)$ where an image is broken down into two shares and the user need one share for successful authorization. And, therefore we will be using $(2,2)$ method as it is more convenient and feasible for the user and it is also secure. AES-128 algorithm has also been used to encrypt the database so that it is not hacked by the hackers and the customers' information remains safe.

At the time of registration, we will be generating an image with unique captcha and then that image will be broken into two shares, one share will be sent on the user's email address and the other will be saved in our database. And also, during login after entering correct password an OTP will be sent to user's registered mobile number after which the user will be able to login.

After the login the user would be able to perform transactions by uploading the share that the user has received on the email and it will be checked with the share on our database and after that the user will be authorized to perform transaction. Also, if any hacker gets the one share of the image, the hacker will not be able to perform transaction as he would not have access to the other share.

So, an online banking platform that has visual cryptography along with AES-128 encryption along with other security features can offer a robust online banking platform which smoothly carries out the day-to-day transactions safely and securely, keeping users' information confidential and maintaining integrity.

1.3 OBJECTIVES

This project aims to enhance the security and privacy of an online banking platform by using various encryption and cryptographic algorithms. The main objectives are as follows:

1. To ensure users' confidentiality, this online platform protects user's identity and data such as personal details, account number, transaction details etc. from attackers trying to get illegal access to the system. For this we use various encryption algorithms in order to keep our user details confidential.
2. To log onto the dashboard, only after successful authentication the user will be authorized to use the account. We have added 2FA in which firstly the user has to enter the password and if the password is correct, an OTP will be sent to his registered mobile number, upon entering which the user will be able to view the dashboard. So, this is an additional security feature which adds to overall robustness of the system.
3. To detect and prevent the fraudulent activities such as phishing attacks, strong cryptographic algorithms such as visual cryptography have been used, AES-128 encryption algorithm has also been used to protect database from being hacked. And, password has also been encrypted through Bcrypt hashing.
4. To secure online transactions and preventing suspicious activities and unauthorized transactions by adding a security layer in which the user has to enter his part of share and if that part matches with the other part a captcha will be displayed upon entering which the user will be allowed to carry out the transaction.
5. To handle user loads ensuring security and performance.
6. To educate users about online banking platforms, how to manage passwords, how to be safe from phishing type of attacks and safeguard themselves from any kind of personal and financial loss.
7. To comply with regulation, the online banking platform needs to adhere to relevant banking regulations and data protection laws, thus ensuring a secure and reliable platform safeguarding customer data and privacy.

1.4 SIGNIFICANCE AND MOTIVATION OF THE PROJECT REPORT

Here we will get to know about the significance of our project, integrating multiple security features in an online banking platform, and the motivation behind making this project:

1.4.1 SIGNIFICANCE

This secure online banking platform holds immense significance which are :

1. Protection of Sensitive Data: As being an online banking platform, it handles a lot of sensitive information such as personal details, financial details which when leaked can lead to financial losses. So, a secure platform protects sensitive data from being hacked thereby saving the customer from kind of theft.

2. Customer Trust: By protecting the customers' personal and financial details, thus making our system secure, we gain the trust and confidence of our customer, which in turn makes the customer use the platform to perform operations.

3. Adaptation to Digital Transformation: As we know that technology is advancing and so is digital banking, so this online banking platform makes the user to be a step ahead from the traditional offline banking and use a secure online banking platform which is more convenient to use.

4. Enhanced Security Features: This online banking platform is integrated with multiple security features such as visual cryptography, AES-128 encryption, two factor authentication, which overall helps to create a robust and secure platform.

5. Continuity of Operations: A significance of this proposed online banking platform is that it reduces the risk of security features and ensures the continuity of banking operations providing uninterrupted services to customers.

6. Innovation: This online banking platform encourages innovation in cybersecurity technologies and practices. It not only drives the development of cutting-edge security solutions but also benefitting the banking sector and contributing to advancements in cybersecurity overall.

7. Societal and Economic Development: This platform overall plays a role in societal and economic development by providing a secure financial environment. This leads to investment, economic growth and overall social progress.

1.4.2 MOTIVATION

The motivation behind developing an integrated online banking platform with multiple security features are:

- 1. Prevention of Financial Loss:** By applying encryption algorithms and security measures we prevent unauthorized access and transactions and cyber threats. By applying the techniques, the platform save itself and the customer from any kind of financial loss.
- 2. Competitive Edge:** In today's fast-growing world, many financial institutions fight to differentiate themselves in a crowded market. Therefore, by offering a secure and reliable online banking platform, it can have a competitive edge which can attract customers.
- 3. Mitigating Financial Risks:** Addressing security concerns and vulnerabilities helps mitigate financial risks associated with data breaches and fraud.
- 4. Ethical Responsibility:** The main responsibility of our platform is to protect our customers data and personal information and therefore we prioritize security measures to safeguard our customers information.
- 5. Customer Expectations:** Our customers expect a convenient yet secure and reliable online banking platform and meeting these expectations is a driving force behind the development of this platform.

1.5 ORGANIZATION OF PROJECT REPORT

Chapter 1: Introduction

In this chapter we get to know about traditional offline banking and how is it not convenient to use it. And with increasing technology the idea of online banking arrives. But with this comes many risks such as attackers trying to steal data and hack the account resulting in loss for our user. There are many other attacks associated with it such as phishing, pharming etc. Therefore, we describe a problem statement and how to deal with it with the help of encryption algorithms and we have also defined a set of objectives and motivation for this project.

Chapter 2: Literature Review

This chapter aims to present various literature review of existing research on the need of security in the online banking platforms. It tells us about various encryption algorithms and compares which one is one secure and reliable to use. Here we also address the major key gaps present in those literature.

Chapter 3: System Development

The chapter explains the various stages of the project and also describes the various security features and encryption that have been used. The chapter also describes the requirements, methodology and the problems faced in the project. The working of the visual cryptography algorithm and the AES-128 algorithm have also been described in detail. Along with this we have also talked about various requirements for the projects, and each and every stage of implementation and all the encryption algorithms used in the project are explained in detail over here.

Chapter 4: Testing

This chapter presents a very detailed analysis of the testing strategy that has been used in the project, and also talks about the various checks employed at different stages of the project to protect our user's privacy. Some of these include prompting the user at every instance of new login, and notifying the user through email when amount is credit in their account.

Chapter 5: Result and Evaluation

This chapter showcases the result obtained in the previous chapter, it also showcases various important portions of the project, such as OTP authentication after entering correct password, AES-128 algorithm and visual cryptography. Along with this we have also attached screenshots of the database after the applying the hashing algorithm, and also the messages and mails received by the user.

Chapter 6: Conclusion and Future Scope

This is the final chapter and there is a summary of the project and its limitations, also suggesting improvements in the field of online banking. This chapter also talks about the future scope of the project that what all changes in future can be further made in order to make the system more secure thus enhancing overall performance of the project. Some of

the work that is to be done in the future is that we can build an administrator panel, work on scalability and build a more interactive platform.

CHAPTER 2: LITERATURE SURVEY

2.1 OVERVIEW OF RELEVANT LITERATURE

In this chapter, we will study about the various literature that we have read during the term of this project and what all information we could gather from it.

2.1.1 INTRODUCTION

In the recent, online banking platforms have been coming into usage as they have gained a lot of popularity. Online banking platforms are an essential part of today's financial transactions as they are accessible and simple to use. They are in greater use, so it is important to protect the security and privacy of customer's information.

Frauds and cyberthreats pose a serious risk. In order to protect the privacy and integrity of the user, their data and their transactions, this project aims to build and implement an online banking platform that is integrated with multiple security features. In the past few years, many research papers have been published, in which the main study area is focused on developing more effective algorithms for strengthening the security of online banking systems. The authors of these papers are from different backgrounds. The diverse backgrounds of these authors have helped in developing efficient techniques to fight the problem of various attacks on the online banking systems and to build such algorithms that develop a system capable to fight these problems.

Therefore, this chapter aims to discuss these papers and their methodologies. It will help in providing an overview of the latest advancements in increasing the security of the online banking platforms, which will be helpful for other researchers working in this particular field.

2.1.2 A SUMMARY OF THE RELEVANT PAPERS

Here we have provided a summary of some research papers that we have gone through over the past months, firstly in the form of a table and elaborated those in brief afterwards.

Below we have the table:

Table 2.1: Summary of the Relevant Literature

S.No.	Paper Title	Journal/Conference	Tools/Techniques	Result	Limitations
1.	A.G. Patil et al., "Secure E-Banking application using visual cryptography" [2]	International Research Journal of Engineering and Technology (2023)	An architecture of e-banking system in which user firstly registers himself by providing and in transaction phase visual cryptography is applied.	This method employs Color Image Visual Cryptography to secure passwords, and it is currently technologically hard to defeat this security.	No algorithm for database security is provided which can lead to hacking of database.
2.	Y. Shah et al., "Analysis of AES and DES Algorithm" [3]	International Journal of Trend in Research and Development (2020)	Comparing AES and DES algorithm on basis of parameters such as avalanche effect.	The avalanche effect is more in AES so it is concluded that AES is more secure than DES.	Only comparing AES and DES algorithm and not any other as other algorithms may have more avalanche effect.
3.	K.Dheeraj, "A study on secure system in online banking system" [4]	Journal of Emerging Technologies and Innovative Research (2018)	This paper tries to explore several technologies and securities standards in order to protect the data.	This study indicates that online banking allows customers to conduct transactions at any time and reduce cost per transaction.	The paper only tells the security issues and existing methods and does not specify algorithms to how to secure the system.
4.	Chandrasekhara et al., "A Novel approach of secure banking application using visual cryptography against fake website authenticity theft" [5]	International Journal of Engineering Research and Technology (2013)	For secure authentication of users, using encryption techniques such as steganography and visual cryptography to implement it.	An algorithm is developed which can be used to authenticate users and authorize them to use internet banking.	Single algorithm that is only visual cryptography is used.
5.	D.R. Moscato et al., "Internationals Perceptions of Online Banking Security Concerns" [6]	Communications of the IIMA (2012)	A group of banks was selected and reporting form was completed for each bank's website and after this data on security features were gathered.	The investigation of these security features has shown how banks have been tailoring their security concerns and also help customers.	Does not tell how to tackle all the security issues.

A.G. Patil et al. [2], proposed a secure e-banking system which was different from the other e-banking systems as the other ones had textual passwords and biometrics. In this literature the type of security that was used was visual cryptography in which an image was broken into shares and a share had to be given in order to complete the process completely. This paper also told us about various type of visual cryptography such as (2,2), (n,n) and (k,n) types. Also, color visual cryptography is used in this literature to secure passwords.

Y. Shah et al. [3], proposed a study which is a comparison of AES (Advanced Encryption Standard) and DES (Data Encryption Standard) as these two are very popularly used. In DES encryption there is a 64-bit block size with 64 bits key. Out of 64 bit, 56 bits are directly used by algorithm and remaining 8 bits are being used for error detection. DES performs 16 rounds of permutations and combinations.

AES algorithm has 128-bit block size which handles three different key sizes such as 128,192 and 256 bits. On the basis of the key size the number of rounds are 10,12 and 14. To compare the both, concept of avalanche affect has been used. Avalanche affect is the number of flipped bits in cipher text by number of bits in cipher text. The avalanche effect of AES was significantly more than DES. Therefore, AES is more secure encryption algorithm than DES.

K. Dheeraj et al. [4], proposed a literature which tries to explore several technologies and security standards. The banks should use the best of methods to secure their system and ought to define best security design and practice. Some security issues in online banking are phishing attacks, internet scams, malware, identity theft, lottery fraud, pharming attack, spyware, trojan, virus. Some existing security systems for online banking are user id and password, OTP, biometric, e-token and SMS banking. We have to protect ourselves online by making sure we have up to date security updates, install effective antivirus software, utilize an individual firewall, be aware to potential fraud and keep our passwords secure. Therefore, this study indicates that online banking allows customers to conduct transactions at any time and reduce cost per transactions.

Chandrasekhara et al. [5], proposed an online banking system as in traditional offline banking there is a chance of encountering forged signature for transaction and in net banking passwords can be hacked. As technology is advancing there has been a significant increase in online attacks and phishing is one of them. Phishing is a type of attack in which the attacker tries to gain access to personal information and perform fraudulent activities which may

result in financial loss. So, in this study visual cryptography is used in order to create a strong system which is not vulnerable to these types of attacks.

D.R. Moscato et al. [6], proposed a paper which is mostly concerned about security of customers. The security policy is illustrated as a tool for banks to use to manage their user's perception. It tells us about what different security measures different banks in different region applies to its systems.

Therefore, in this paper in total of nine questions were gathered and compared. Some of these questions were, was encryption used in transmission of e-commerce data? Was encryption used in the storage of banking data? Was there any statement on the use of 128-bit encryption? Did the site discussed the use of network security or firewalls. The answer to these questions were, 40% banks used them while 60% doesn't, 15% of them use it while 85% doesn't, 36% of them say that they use 128-bit encryption while 64% does not use, 43% say that there exists a firewall but 53% does not use them. So, these all-security features state what percentage of banks use them and what not. Therefore, if all the banks used them and make their system more trustworthy for their customers then their customers will also feel secure while using their services.

H.O. Alanazi et al. [7], proposed a literature which tells us about Unified Modelling Language (UML). As we know, nowadays, the internet banking system is widely used and the banks are looking to provide best quality systems with fast response, secure and safe to use. The unifies modelling language is the unique language which is used to analyze and design any system.

In this paper the UML diagrams have been proposed to illustrate the design phase for any banking system. UML sequence diagrams model flow of logic within your system in a visual manner enabling you both to document and validate your logic and is commonly used for both analysis and design purposes. Similarly, there are class diagrams, data flow diagrams and architectural design. The layered approach will help in achieving a concrete system with less vulnerabilities.

2.2 KEY GAPS IN THE LITERATURE

After going through several relevant literature on the secure online banking platform, the main problem that was coming out was that not a single system was fully secure and were

vulnerable to the risks. Also, we got to know how many percent of banking institutions had these encryption algorithms applied to their systems and how many did not. One more problem was that every study included only one algorithm which was not safe enough in case if of an attack. None of the literature focused on more than one algorithm as we have to make a robust platform to avoid any financial losses.

In A.G. Patil et al.'s work, we were introduced to three types of visual cryptography out of which one was used in this study to make a secure e-banking platform. There was no information about any authentication procedure and also about the safety of the information stored in the database.

In Y. Shah et al.'s work, we got to know about two popular algorithms AES and DES, and by the avalanche affect, AES encryption algorithm came out to be more efficient than DES. But the problem in this study was that by only using AES one cannot build a strong and secure online banking platform.

In K. Dheeraj et al.'s work, we were told about various security issues in online banking and after this what security systems at present we have in online banking. At last, we got to know about some prevention steps that would keep us safe from attacks. The major problem in this study was that it did not tell any kind of encryption and cryptographic algorithm which would keep us safe from attacks.

In Chandrasekhara et al.'s work, we were introduced to type of attack that happens in online banking that is phishing attack. To be safe from this attack, only visual cryptography was being used. So the problem was that apart from visual cryptography no other algorithm was used due to which a strong system could not be built.

In D.R. Moscato et al.'s study, there were many questions involved around the security of online banking as in recent years online banking have gained a lot of popularity. We got to know that how many of the banks really apply these algorithms to keep their customer's data safe and protect them from financial loss. The key gap here was that that the study was not educating about how to keep the data safe, it did not tell us anything about encryption algorithms.

In H.O. Alanazi et al.'s study, we got to know about very interesting field that is unified modelling diagrams as these diagrams such as sequence diagram, class diagram, data flow diagram and the architectural design helps us in following a step-by-step process which can

create a robust system. The major gap in this paper is same as above as it did not tell us about any security algorithms to keep our information private and safe.

CHAPTER 3: SYSTEM DEVELOPMENT

3.1 REQUIREMENTS AND ANALYSIS

The proposed online banking platform needs to be designed in such a way that it is not vulnerable to any security risks. So, we have to contemplate various requirements, and we have to make sure that these requirements are met completely, and apart from this we have to scan thoroughly each and every potential risks involved. Some key requirements and analysis that we must need to consider are:

3.1.1 FUNCTIONAL REQUIREMENTS

The following is included in the functional requirements:

- 1. Customer Authentication and Authorization:** Users will be able to register by using valid credentials and after registering successfully they will be asked for two factor authentication and only after successful authentication they will be authorized to view the account.
- 2. Manage Accounts:** To allow users to view their account details, balance, and facilitate them in checking their transaction history and in transferring funds securely.
- 3. Transaction security:** To facilitate secure transactions between two accounts by using cryptographic algorithms.
- 4. Visual Cryptography Algorithm:** The system is also equipped with an efficient and secure visual cryptography algorithm which is capable of creating unique pair of shares for each user and these shares can then be used to provide an extra layer of security at the time of auction.
- 5. Accessibility and Reliability:** To ensure that the online banking platform is easily accessible to all the customers and is also reliable for them in a long-term usage.
- 6. Customer Trust:** To make such a robust online banking platform which is capable of saving the users from harmful attacks such as phishing and pharming, and in the process gain their trust and confidence enabling them to use our system efficiently.

3.1.2 NON-FUNCTIONAL REQUIREMENTS

The following have been included in the non-functional requirements:

- 1. Security:** In order to safeguard the data being entered into the database the AES-128 encryption algorithm has been used and apart from this visual cryptography, mobile OTP and alphanumeric password have also been used.
- 2. Performance:** Keeping in mind the overall security the performance also needs to be taken care of. We have to ensure that the time taken by transactions is less and the system is not laggy and works smoothly. Along with this we also need to maintain system reliability in order to ensure continuous banking services.
- 3. Usability:** Bootstrap has been used to ensure that the website is responsive and user friendly, by doing this we have ensured that there are no limitations on the device on which it can be used.
- 4. Scalability:** The application has been designed in such a way that sudden increase in the user loads does not affect it and the time taken to fetch data from the database is also not compromised in such an untimely situation.
- 5. Consistency:** To maintain consistency in design for a cohesive user experience.

3.1.3 HARDWARE REQUIREMENTS

The hardware requirements are as follows:

- 1. Server:** In order to scale the application XAMPP has been used as a local server, by doing this we were able to see the changes made in the code being updated in real time on the website and it also never faced any issue due to the increased traffic and load on the system.
- 2. Storage:** In order to make sure that the rendering and loading time is less we have used the local storage, it also provided us with extra storage and there was no burden present for the storage level.
- 3. Network:** A reliable internet connection with sufficient bandwidth is needed in order to handle the incoming and outgoing data.

3.1.4 SOFTWARE REQUIREMENTS

The software requirements needed for the completion of the project are as follows:

3.1.4.1 LANGUAGES USED

1. HTML: HTML stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is a combination of Hypertext and Markup language. It uses tags to organize different elements like text, images, links and multimedia. HTML defines the layout, headings, paragraphs, lists and forms. It is the foundation for building web content and works with CSS for styling and JavaScript for interactivity [19].

2. CSS: CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It defines how HTML elements look on a web page. It controls the appearance, design and layout aspects such as colors, fonts, spacing and positioning. CSS helps developers to style elements precisely and create visually appealing and responsive web pages across different devices [20].

3. JavaScript: JavaScript is a versatile programming language, used for enhancing user interfaces and enabling dynamic, interactive elements within web browsers. It primarily serves as a client-side scripting language, it gets executed within the browsers of the user thus allowing for real time updates, dynamic content modifications, and improved user experiences without requiring full page reloads. It is responsive to the actions of the user such as clicks and keypresses, giving the developers freedom to create engaging and responsive web pages [21].

4. PHP: PHP stands for Hypertext Preprocessor, it is a scripting language which is freely available and is very widely used for web development, it is primarily used for server-side scripting, but can also be used for command line scripting. It is integrated with a number of popular databases, including MySQL, PostgreSQL, etc. The MySQL server after its execution is capable of handling complex queries with huge result sets in no time. It is used for developing dynamic web pages, it is executed on the server, and the results are sent to the client as plain HTML code [22].

3.1.4.2 LIBRARIES USED

1. Bootstrap: It is an open-source front-end framework widely recognized for simplifying the creation of responsive and mobile-first web pages. It was originally developed by Twitter, but now maintained by the open-source community. It offers a comprehensive toolkit which comprises of HTML, CSS and JavaScript components. Its main feature is the responsive grid system which allows the developers to create adaptable layouts across various screen sizes from desktops to mobile devices. It includes an array of pre-built UI components such as navigation bars, buttons, forms and models, which helps in providing consistency in design and saving time and effort which are normally required for styling common elements [23].

2. PHPMailer: It is very useful library in PHP which is used to send email directly from the code itself, it has an integrated SMTP support due to which it does not require any local mail server. We can send emails to multiple people and also include CC, BCC, and Reply-to addresses, it also enables us to add attachments, due to which we were able to send a share of image to the email id of the users. It validates email addresses automatically and also protects against header injection attacks [25].

3. Twilio: Twilio is a cloud communications platform that provides APIs and services for adding various communication features, such as SMS, voice, video, and more. In order to facilitate the integration of the Twilio services into PHP, Twilio provides an official PHP library. This library allows the developers to interact with the Twilio REST API, thus making it easier to handle and manage the communication related tasks [27].

4. GD: It is a library in PHP which is used for the creation and manipulation of images. GD stands for Graphic Draw, it can be used for the resizing, rotation and cropping of images, it is also used to write text onto the image, this functionality of GD was utilized while applying the visual cryptography algorithm [28].

3.1.4.3 TOOLS USED

1. Visual Studio Code (VS Code): Visual Studio Code, commonly known as VS Code, is a very renowned code editor developed by Microsoft. It offers a strong code editing environment, with features such as syntax highlighting, autocompletion, and IntelliSense, thus making it adaptable to a lot of programming languages. There are various extensions available on it, which help the developers in various ways. It also has an integrated terminal within the editor which eases the process of command execution and task automation.

Notably, VS Code seamlessly integrates Git, easing version control operations without leaving the editor. Live Share facility provided on it is very useful in web development.

2. XAMPP: XAMPP is an open-source web server solution package, it is mainly used for web application testing on a local host webserver. XAMPP integrates the Apache HTTP Server with MySQL and PHP, this enables the developers to use a pre-configured environment for testing and deploying web applications on their own system. Apart from the core components mentioned above, it also includes additional tools and interpreters like OpenSSL and phpMyAdmin facilitating secure connection and database management.

3. MySQL: MySQL is a relational database management system (RDBMS), it was developed by Oracle Corporation, it is widely acknowledged for its efficiency in handling structured data. It operates on basis of the relational model by organizing data into tables with rows and columns. It also supports various data types, transactions, and complex queries, which makes it suitable for variety of applications ranging from small-scale projects to enterprise-level solutions.

3.2 PROJECT DESIGN AND ARCHITECTURE

In this chapter we will get to know about the idea of the project by understanding the methodology behind it, also, we will understand it the help of flow chart and class diagram.

3.2.1 METHODOLOGY

The methodology is that firstly the user will register himself with valid credentials and only after successful registration the user will be allowed to login. During the registration process the user will be asked to enter his personal details such as name, mobile number, date of birth, email, address, city, district, Aadhar card number, pan number, Aadhar card photo, pan card photo and personal photo. After entering all the details correctly, the user will be registered.

Now if the registration is successful, the user will receive one share of the image as we are using visual cryptography in which an image is broken down into two shares, and one share is stored in the database and the other share is emailed to the user. So, the user will receive one share on the email, which has to be kept safely as it will be used in future, when the user will perform any transaction.

After this, the user can login using his email and phone number as the password. If the password entered by the user is correct then only, he will receive OTP on the registered mobile number. After entering the OTP, the user will be able to view the dashboard.

On the dashboard, several sections would be there such as view account details, view account balance, transfer money and transaction history. In account details section the user will be able to see his details. In view balance section, the user will be able to view the balance. In the transfer money section, the user can make payments by entering the receiver's account number and amount. And, in the transaction history section, the user will be able to see all the payment history.

In the transfer money section, after entering all the relevant details of the receiver's account and the amount the user will be asked to upload the share of image that he got through email while registering. And in the backend, we will match both the shares by superimposing them on each other and checking the captcha value. If the image uploaded by the user is correct then the captcha value will be displayed on the screen, after which by entering that correct captcha value the transfer will be completed. If any attacker trying to commit fraudulent activity tries to continue with the transfer by uploading the wrong share, the attacker will not be allowed to continue the transfer as the share uploaded by the attacker will not match with the one saved in our database, so the attacker will not be able to get away with the transaction and our customer will be safe from any kind of financial loss.

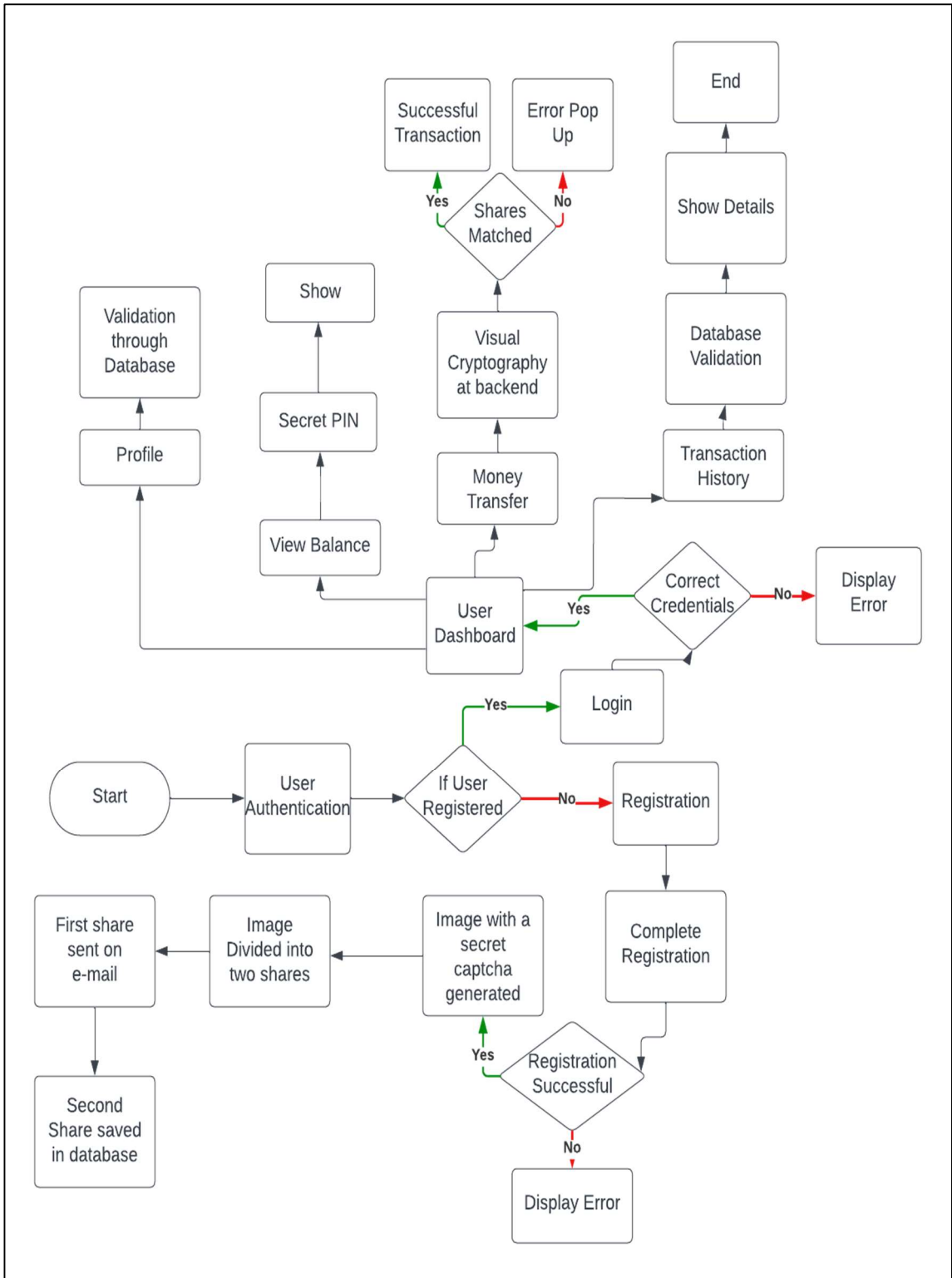


Figure 3.1: Flow Chart of the Project [26]

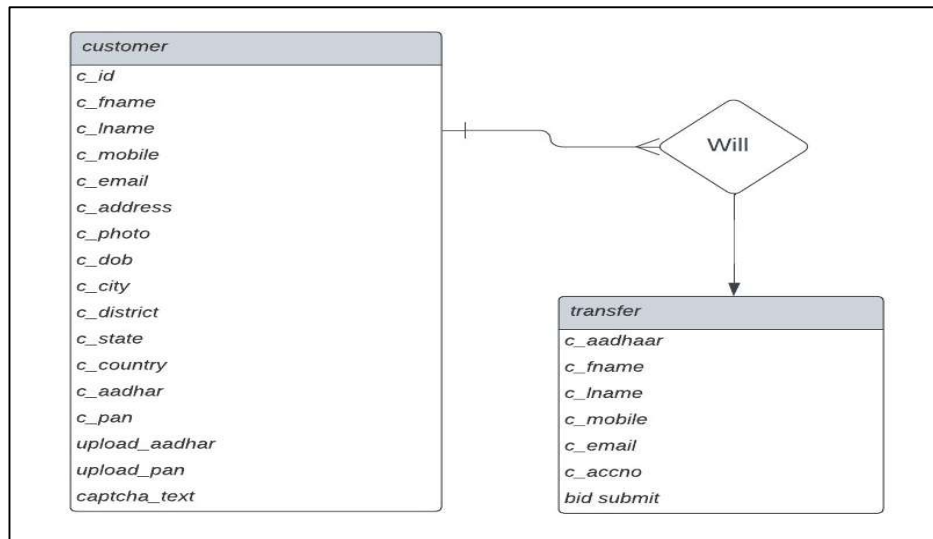


Figure 3.2: Class Diagram

3.3 DATA PREPARATION

We have created two tables in our database one is customer table and second is transfer table.

The entities of our customer table are:

1. `c_id`: This is our primary key.
2. `c_fname`: To store first name of the user.
3. `c_lname`: To store last name of the user.
4. `c_mobile`: To store mobile number of the user.
5. `c_email`: To store email of the user.
6. `c_address`: To store address of the user.
7. `c_photo`: To upload photo of the user.
8. `c_dob`: To store date of birth of the user.
9. `c_city`: To store city of the user.
10. `c_district`: To store district of the user.
11. `c_state`: To store state of the user.
12. `c_country`: To store country of the user.

13. c_aadhar: To store Aadhar number of the user.
14. c_pan: To store first name of the user.
15. upload_aadhar: To upload Aadhar card of the user
16. upload_pan: To upload pan card of the user.
17. captcha_text: Captcha text is stored in the database in encrypted form and will be used later.

The entities of our transfer table are:

1. c_aadhar: For user to enter Aadhar number.
2. c_fname: For user to enter first name of receiver.
3. c_lname: For user to enter last name of receiver.
4. c_mobile: For user to enter mobile number of the receiver.
5. c_email: For user to enter email of receiver.
6. c_accno: For user to enter account number of receiver.
7. amount: For user to enter the amount, he wants to transfer.

3.4 IMPLEMENTATION

The following is the step-by-step plan in which the online banking platform is implemented:

3.4.1 DOWNLOADING ESSENTIAL SOFTWARE

The initial step to develop the project was downloading essential tools that would be further required in the making of the project. So, we installed the XAMPP software which is a web server used for hosting the websites locally on the system. After setting up the XAMPP software on the system, we set up the database on the MySQL software which is a relational database management system. Lastly, we created all the necessary tables in the database that will be required in our project.

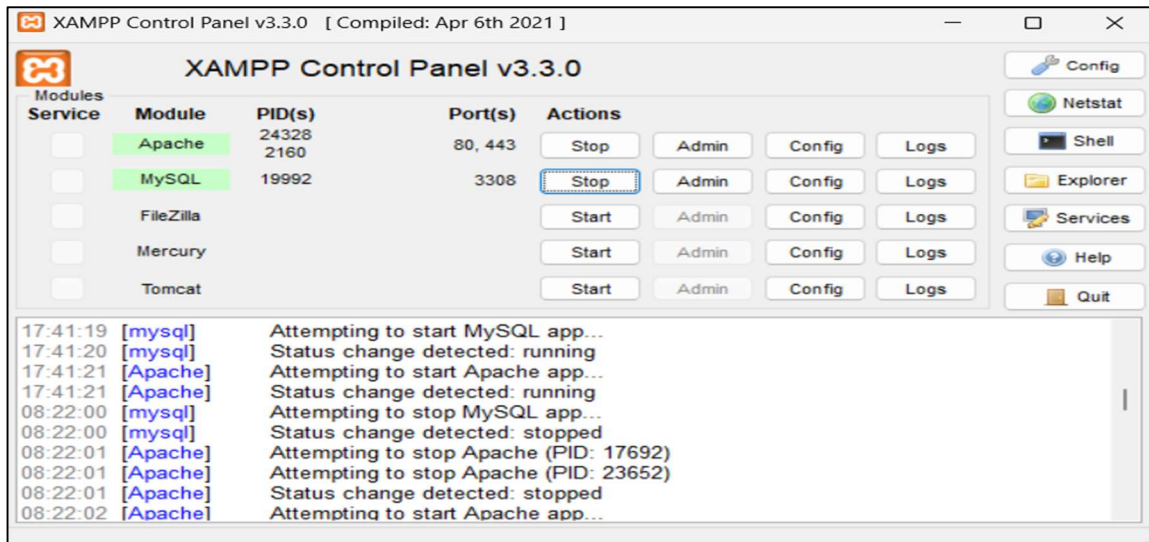


Figure 3.3: XAMPP Control Panel

Table	Action	Rows	Type	Collation	Size	Overhead
customer	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
transfer	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
2 tables	Sum	5	InnoDB	utf8mb4_0900_ai_ci	32.0 KiB	0 B

Figure 3.4: Different Columns of Online Banking Platform Table

3.4.2 OTP AUTHENTICATION USING TWILIO

Twilio is a cloud communications platform that provides APIs and services for adding various communication features, such as SMS, voice, video, and more. In order to ease the integration of the Twilio services into PHP, Twilio provides an official PHP library. This library allows the developers to interact with the Twilio REST API, thus making it easier to handle and manage the communication related tasks.

In order to integrate Twilio to our website we first of all obtained API credentials from Twilio by signing up for a free account, then we imported the Twilio PHP library in our own program.



Figure 3.5: Twilio API Dashboard

After this on the registered mobile number the user will get four-digit OTP which will be a random number between 1000 to 9999.

```

if (($username == $user) && ($password == $pass)) {
    $otp_code = rand(1000, 9999);
    $_SESSION['otp_code'] = $otp_code;

    $to = '+91' . $pass;

    $sid = "ACcf35b617cbe3cc34de807c560c02e2c5";
    $token = "4ca3a1a69551a2fb9b45aac02b53ea8c";
    $client = new Twilio\Rest\Client($sid, $token);

    $client->messages->create(
        $to,
        array(
            'body' => 'Your OTP code is: ' . $otp_code,
            'from' => '+14147518834'
        )
    );

    echo "OTP has been sent";
} else {
    echo "<script>alert('Invalid credentials')</script>";
}
else {
    echo "<script>alert('User not found')</script>";
}

```

Figure 3.6: Code to send OTP using Twilio

3.4.3 AES-128 ALGORITHM

To encrypt the captcha text that is being stored in the database we have used AES-128 encryption algorithm. AES stands for Advanced Encryption Standard and the 128 is the key size here. [10] The block size is also of 128 bits in this encryption algorithm, there are two more variants of this algorithm where the key size is 192 and 256 bits respectively. The number of rounds also vary depending on the key size for 128,192 and 256 it is 10,12 and 14 rounds respectively. We are specifically using AES-128 algorithm instead of DES algorithm because the avalanche effect [3] of AES is more than DES and to hack DES it takes 1142 years and for AES-128 it is 5.4×10^{24} years which is significantly more than the DES. Therefore, AES is widely used and is considered as one of the strongest encryption algorithms. We can also use AES-192 or AES-256 but there can be some change in the overall efficiency so keeping in mind all the factors, AES-128 is used.

In AES-128, there are 10 rounds and each round consists of four operations:

- 1. SubBytes:** In this round each byte in the block is substituted with a corresponding value from a lookup table.
- 2. ShiftRows:** In this operation the rows of the blocks are shifted by a certain number of bytes in this operation.
- 3. MixColumns:** The mixcolumn operation mixes the columns of the block by multiplying them with a fixed polynomial.
- 4. AddRoundKey:** This operation is used to XOR the block with a portion of the secret key.

These four operations occur in all the rounds except in the last round, MixColumn operation does not occur as the omission of the last round MixColumn has no security implications. After the last round, the ciphertext block is obtained which can be decrypted using the same secret key. The decryption process is just the reverse of the encryption process in which all the operations occur in the reverse order.

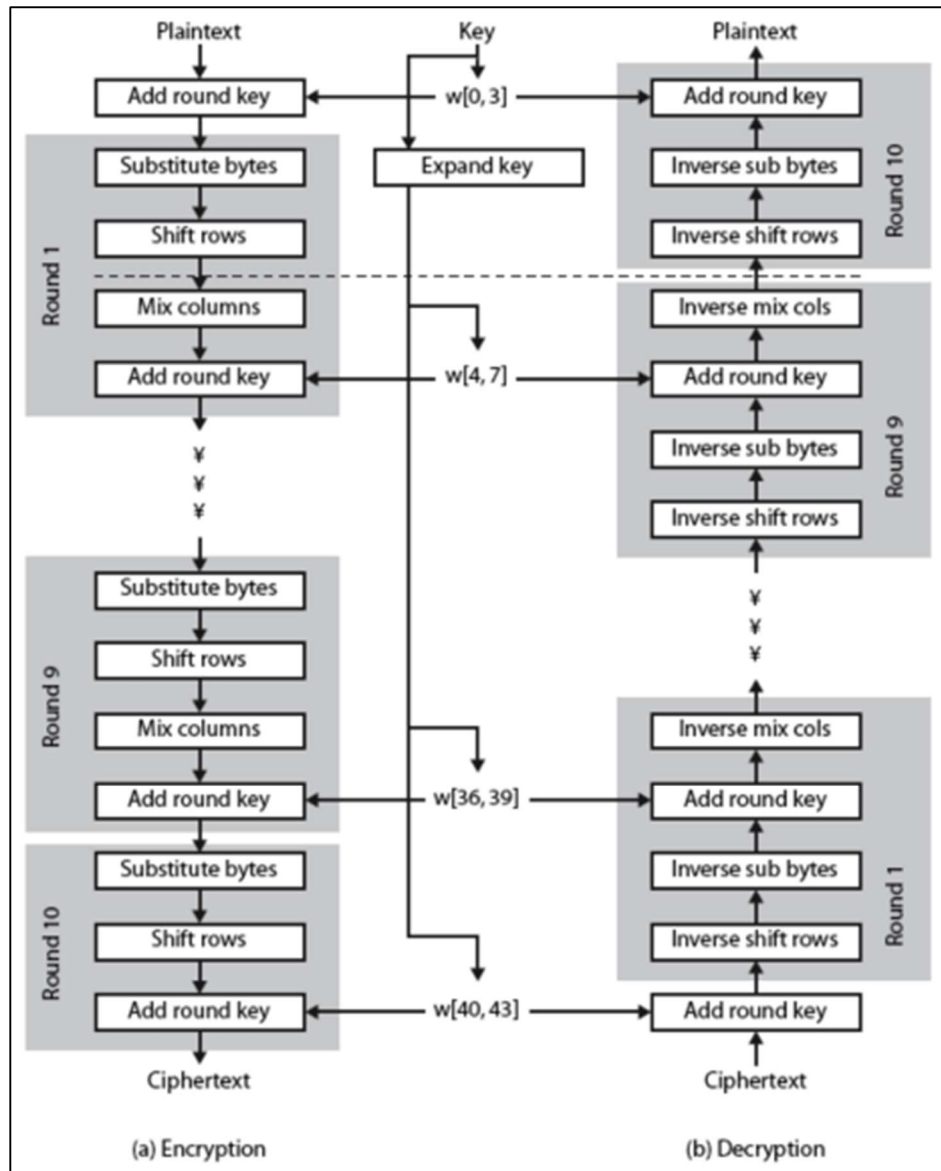


Figure 3.7: Structure of AES Algorithm

```

$original_string = $captcha;
$cipher_algo = "AES-128-CTR";
$iv_length = openssl_cipher_iv_length($cipher_algo);
$option = 0;
$encrypt_iv = $phone;
$encrypt_iv = str_pad($encrypt_iv, 16, "\0");
$encrypt_key = $name;
$encrypted_string = openssl_encrypt($original_string, $cipher_algo, $encrypt_key, $option, $encrypt_iv);
$captcha = $encrypted_string;

```

Figure 3.8: AES-128 Encryption

```

$cap_t = $row['captcha_text'];
$encrypted_string = $cap_t;
$cipher_algo = "AES-128-CTR";
$iv_length = openssl_cipher_iv_length($cipher_algo);
$option = 0;
$decrypt_iv = $row['phone'];
$encrypt_iv = str_pad($decrypt_iv, 16, "\0");
$decrypt_key = $row['key'];
$decrypted_string=openssl_decrypt ($encrypted_string, $cipher_algo,$decrypt_key, $option, $decrypt_iv);
$cap_t=$decrypted_string;

```

Figure 3.9: AES-128 Decryption

3.4.4 VISUAL CRYPTOGRAPHY

There are many types of cryptography techniques in which we can use to encrypt our data. Here we will be using visual cryptography which is a method of secure communication that uses images to encrypt secret messages [14]. The methodology of visual cryptography is that, an image is divided into multiple shares and the original message that was present behind the image is also broken into that many shares, now in order to decrypt the message all the shares in which the original image was broken into need to be superimposed on each other and after that we will get the original message.

There are three types of visual cryptography:

- 1. (2,2) Visual Cryptography:** In this method, the original image is divided into two shares, these shares appear as random patterns or noise when these two shares are overlapped on each other then only the original message is revealed.
- 2. (k, n) Visual Cryptography:** In this method, the original image is divided into more than n number of shares and any k number of shares among the n shares can be overlapped to reveal the original message inside the image.
- 3. (n, n) Visual Cryptography:** In this method, the original image is divided into n number of shares and all the n number of shares from the n shares are needed to be overlapped to reveal the original message inside the image.

Here we will be using (1,2) Visual Cryptography method, as it more convenient and feasible for users to upload one image rather k number or n number of images as it is time consuming task and will lead to have a effect on the efficiency so (1,2) method is used.

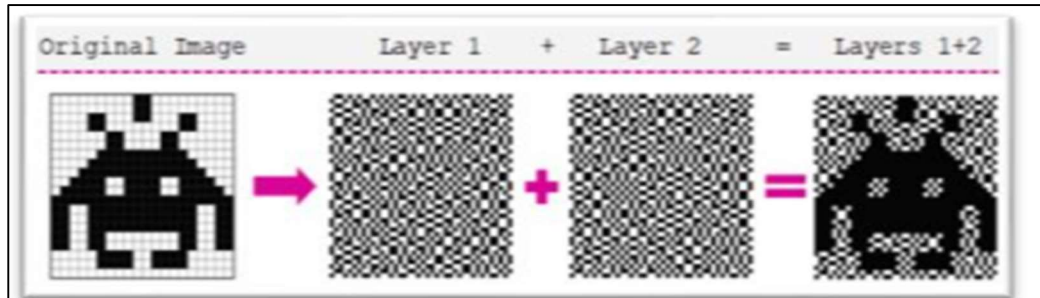


Figure 3.10: Representation of Share Generation [7]

Secret image	Share1	Share2	Stacked image
□			
■			

Figure 3.11: Visual Cryptography Method of Black and White Pixels

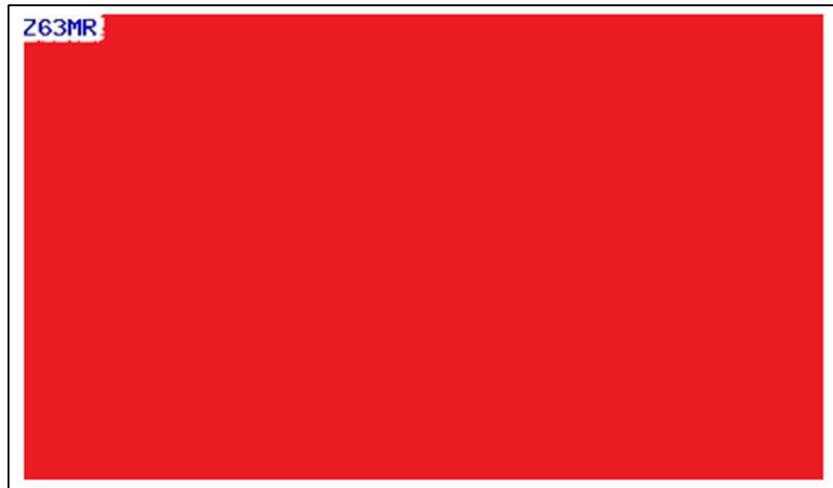


Figure 3.12: Initial Captcha

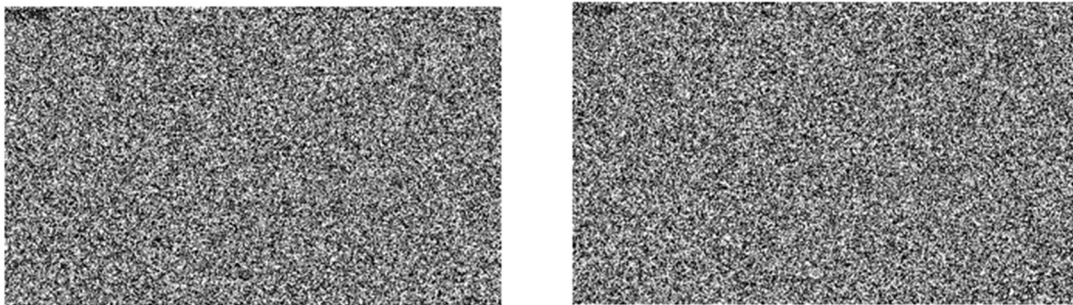


Figure 3.13: Shares in which Captcha is Divided



Figure 3.14: Captcha displayed after Decryption

To implement the Visual Cryptography algorithm, we have used the GD library of PHP, which is used for image generation and manipulation. Firstly, we have generated a random captcha text and then used the GD library to enshrine it on the image.

```
$captcha="";
$captcha = substr(str_shuffle('ABCDEFGHIJKLMNOPQRSTUVWXYZ123456789'), 0, 5);

$img = imagecreate(500, 300);
$white = imagecolorallocate($img, 255, 255, 255);
$blue = imagecolorallocate($img, 0, 0, 255);
imagefilledrectangle($img, 0, 0, 500, 300, $white);
imagestring($img, 5, 0, 0, $captcha, $blue);
imagejpeg($img, "IMAGE.jpg");
imagedestroy($img);
$original_image = imagecreatefromjpeg('IMAGE.jpg');
```

Figure 3.15: Code for generating Captcha image for Encryption

Now we have broken the image into two shares by iterating over each pixel position in the original image and then extracting the color present at that particular pixel position, then we have broken the color into their RGB components and after this a random value has been generated between 0 and 255, this value will determine which share the pixel will be assigned to. Now we will assign pixel to share in following way, if random value is greater than or equal to 128, in share 1 the color is set to the same color as original pixel that is RGB and in share 2 the same pixel color is set as black. If random value is less than 128, in share 1 the pixel color is set as black and in share 2 the pixel color is set to same color as the original pixel. This process ensures that each share individually appears as a noisy image, but when combined, the original image can be reconstructed.

```

$share1 = imagecreatetruecolor(imagesx($original_image), imagesy($original_image));
$share2 = imagecreatetruecolor(imagesx($original_image), imagesy($original_image));
for ($x = 0; $x < imagesx($original_image); $x++) {
    for ($y = 0; $y < imagesy($original_image); $y++) {
        $color = imagecolorat($original_image, $x, $y);
        $r = ($color >> 16) & 0xFF;
        $g = ($color >> 8) & 0xFF;
        $b = $color & 0xFF;
        $rand = rand(0, 255);
        if ($rand >= 128) {
            imagesetpixel($share1, $x, $y, imagecolorallocate($share1, $r, $g, $b));
            imagesetpixel($share2, $x, $y, imagecolorallocate($share2, 0, 0, 0));
        } else {
            imagesetpixel($share1, $x, $y, imagecolorallocate($share1, 0, 0, 0));
            imagesetpixel($share2, $x, $y, imagecolorallocate($share2, $r, $g, $b));
        }
    }
}
imagejpeg($share1, "share1.jpg");
$filename = "shares/" . $aadhar . "_share2.jpg";
imagejpeg($share2, $filename);

```

Figure 3.16: Code for breaking the Original Image into two shares that is Encryption

For the decryption part we will be combining both of the shares of image that were obtained during encryption, and for this we will check if both the shares have black pixels at the current position by assigning all RGB values zero. If condition is true, reconstruction image pixel is set to black color otherwise, the pixel is reconstructed by adding corresponding color components from both the shares. The component of reconstructed pixel is calculated as sum of RGB. If any component exceeds 255, they are clamped to 255 using if condition and if any component is less than zero then it is clamped to zero. Therefore, we have our reconstructed image.

```

$width = imagesx($share1);
$height = imagesy($share1);
$reconstructed_image = imagecreatetruecolor($width, $height);

if (!$reconstructed_image) {
    die('Failed to create new image');
}

for ($x = 0; $x < $width; $x++) {
    for ($y = 0; $y < $height; $y++) {
        $color1 = imagecolorat($share1, $x, $y);
        $color2 = imagecolorat($share2, $x, $y);
        $r1 = ($color1 >> 16) & 0xFF;
        $g1 = ($color1 >> 8) & 0xFF;
        $b1 = $color1 & 0xFF;
        $r2 = ($color2 >> 16) & 0xFF;
        $g2 = ($color2 >> 8) & 0xFF;
        $b2 = $color2 & 0xFF;
        if ($r1 == 0 && $g1 == 0 && $b1 == 0 && $r2 == 0 && $g2 == 0 && $b2 == 0) {
            imagesetpixel($reconstructed_image, $x, $y, imagecolorallocate($reconstructed_image, 0, 0, 0));
        } else {
            $r=$r1 + $r2;
            $g=$g1 + $g2;
            $b=$b1 + $b2;
            if($r > 255) { $r = 255; }
            if($g > 255) { $g = 255; }
            if($b > 255) { $b = 255; }
            if($r < 0) { $r = 0; }
            if($g < 0) { $g = 0; }
            if($b < 0) { $b = 0; }
            imagesetpixel($reconstructed_image, $x, $y, imagecolorallocate($reconstructed_image, $r, $g, $b));
        }
    }
}

```

Figure 3.17: Code for combining the two shares to Decrypt the Message

3.4.5 CRYPTOGRAPHIC HASHING

A variable-length block of data M is fed into a hash function H , which outputs a fixed-size hash value, $h = H(M)$. The feature of a "good" hash function is that it will yield outputs that are uniformly distributed and appear to be random when applied to a wide set of inputs. Data integrity is, generally speaking, the main goal of a hash function. There is a strong likelihood that any modification to any bit or bits in M will alter the hash value.

A cryptographic hash function is the type of hash function required for security applications. A cryptographic hash function is an algorithm for which finding either (a) a data object that maps to a pre-specified hash result (the one-way property) or (b) two data objects that map to the same hash result (the collision-free property) is computationally impossible (because

no attack is significantly more efficient than brute force). Hashing functions are useful for determining whether or not data has changed because of these features.

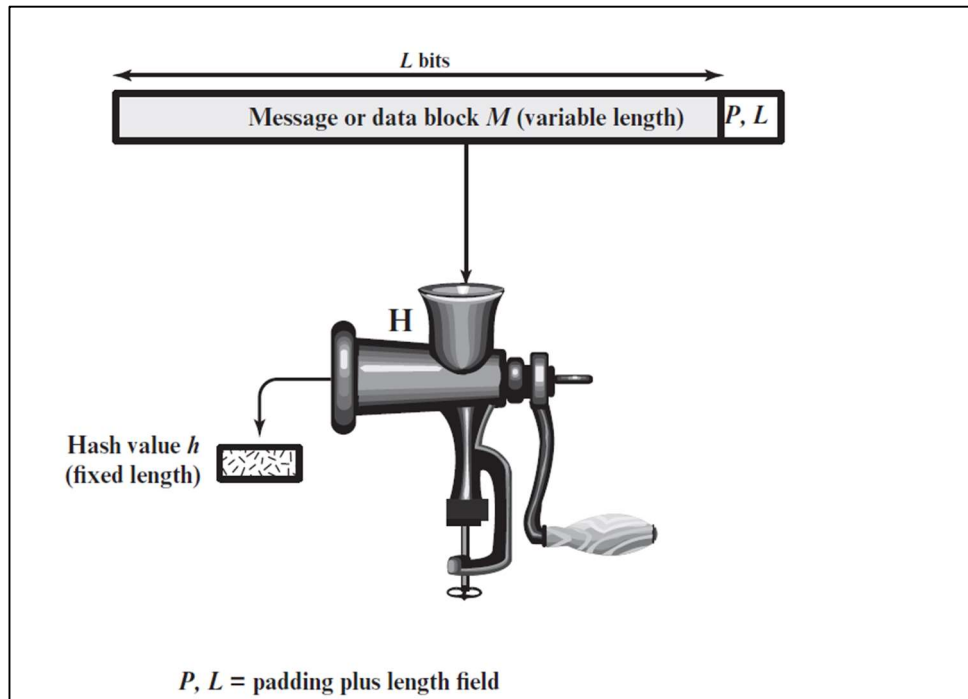


Figure 3.18: Cryptographic Hash Function; $h = H(M)$

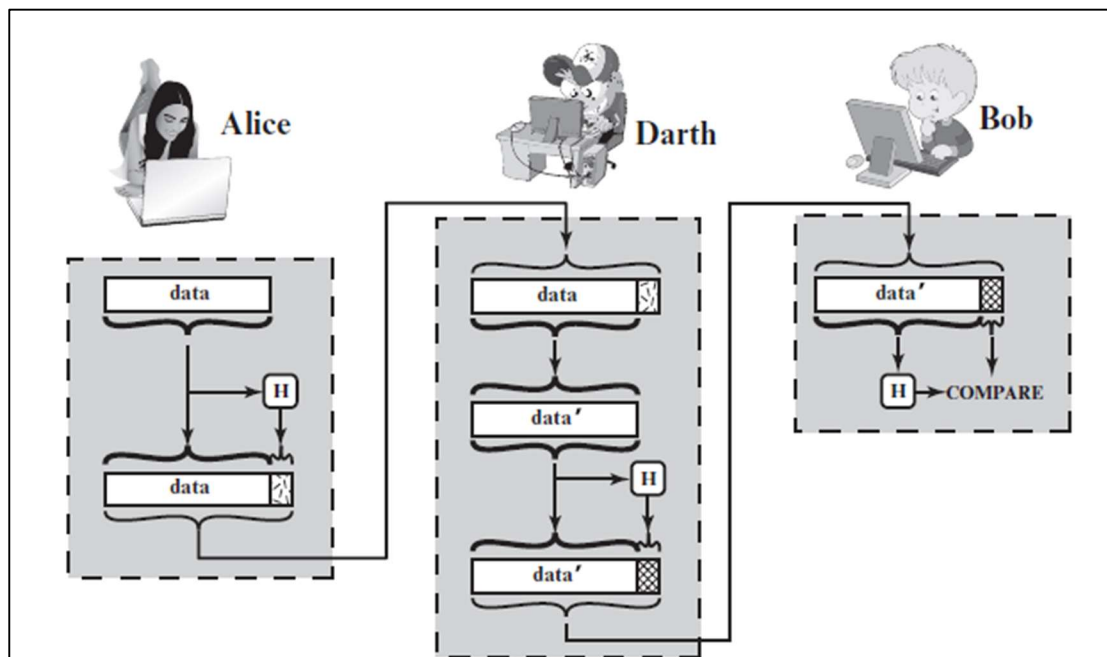


Figure 3.19: Attack against Hash Function

In order to enhance the security, we have used bcrypt hashing algorithm to store the password of user in form of a cryptographic hash in the database. Bcrypt is a cryptographic hash function which is designed for password hashing and safe storing in the backend of applications in a way that is less susceptible to dictionary-based cyberattacks. Bcrypt is better than SHA 256 algorithm as it contains a salt element which is not present in SHA due to which it becomes more susceptible to dictionary-based cyberattacks, therefore bcrypt is a better solution for safely storing passwords.

Bcrypt runs a complex hashing process, during which a user's password is transformed into a fixed-length thread of characters. It uses a one-way hash function, meaning that once the password is hashed, it cannot be reversed to its original form. Every time the user logs into their account, bcrypt hashes their password anew and compares the new hash value to the version stored in the system's memory to check if the passwords match [24].

Instead of simply hashing the given password, bcrypt adds a random piece of data, called salt, to create a unique hash that is almost impossible to break with automated guesses during hash dictionary and brute force attacks.

Bcrypt also stands out among other hashing algorithms because it uses a cost factor. With the help of this, we can determine the number of password iterations and hashing rounds to be performed, increasing the amount of time, effort, and computational resources needed to calculate the final hash value. The cost factor makes bcrypt a slow algorithm that takes significantly more time to produce a hash key, turning it into a safe password-storing tool.

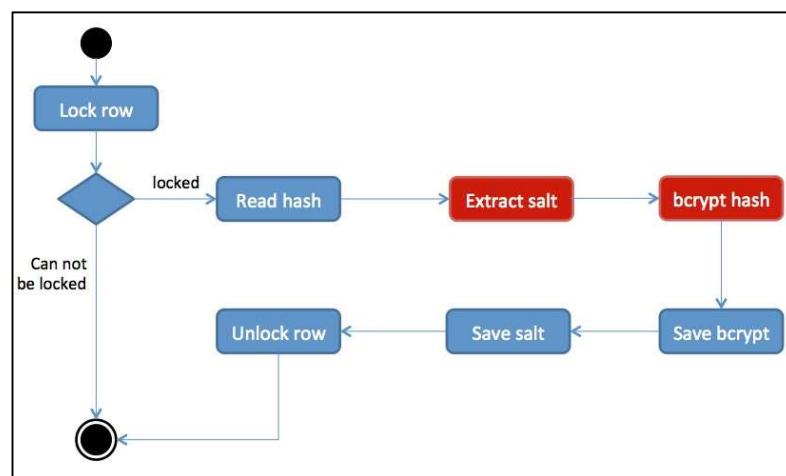


Figure 3.20: Bcrypt Hashing flow diagram

```
if(isset($_POST['submit'])) {
    $fname = $_POST['fname'];
    $lname = $_POST['lname'];
    $dob = $_POST['dob'];
    $mobile = $_POST['mobile'];
    $mobilec = $mobile;
    $mobile = password_hash($mobilec, PASSWORD_BCRYPT);
    $email = $_POST['email'];
    $address = $_POST['address'];
    $city = $_POST['city'];
    $district = $_POST['district'];
    $state = $_POST['state'];
    $country = $_POST['country'];
    $aadhar = $_POST['aadhar'];
    $pan = $_POST['pan'];
}
```

Figure 3.21: Code for Bcrypt Hashing algorithm

3.4.6 GOOGLE RECAPTCHA

Google reCAPTCHA has been used in our system, in order to protect it from the automated bots and spam. It is a technology that is developed by Google, which presents users with challenges, such as identifying objects in images or solving puzzles, to verify if they are human. It helps to make sure that the information submitted is coming from real users rather than automated scripts or bots. We have used this feature for the login of the user.

The reCAPTCHA service generates a unique token for the response of each user, which is then sent to the server for verification. The server, in turn, communicates with the reCAPTCHA's API to confirm the validity of the response of the user. If the user verification is successful, then form submission is allowed to proceed.

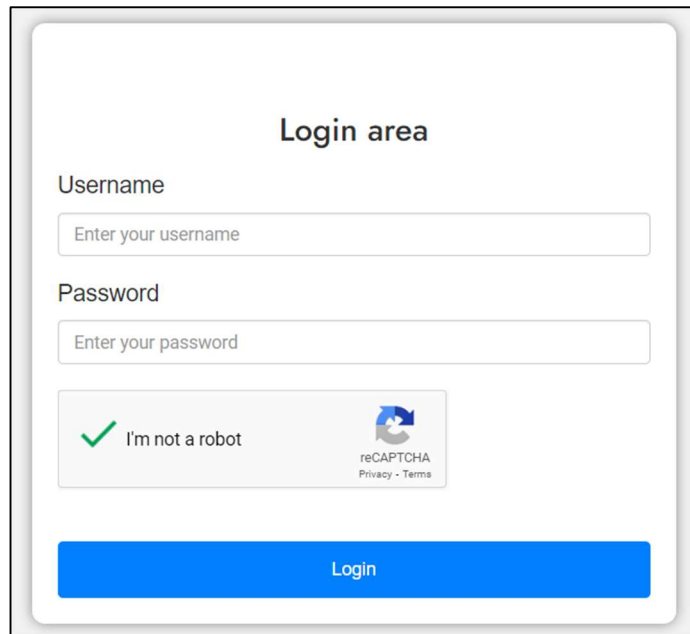


Figure 3.22: ReCAPTCHA being used in the login page

```

if(isset($_POST['g-recaptcha-response']))
{
    $secretkey = "6Lc8fBg1AAAA0z5vqCJ9S11Y0g2KcNTlwCp0fgY";
    $ip = $_SERVER['REMOTE_ADDR'];
    $response = $_POST['g-recaptcha-response'];
    $url = "https://www.google.com/recaptcha/api/siteverify?secret=$secretkey&response=$response&remoteip=$ip";
    $fire = file_get_contents($url);
    $data = json_decode($fire);
    if($data->success==true){
        if($Rows!=Null && $Rows['c_email']==$username && password_verify($password,$Rows['c_mobile']))
        {
            session_start();
            $_SESSION['username'] = $username;
            $_SESSION['password'] = $password;
            $_SESSION['aadhar_number'] = $Rows['c_aadhar'];
        }
    }
}

```

Figure 3.23: Code for the reCAPTCHA authentication

3.4.7 RESEND OTP FEATURE

We have added an extra feature of resend OTP option for the user. If by some mistake the user loses the OTP sent to the registered mobile number then there is a resend OTP button by which the user will obtain another OTP on the registered mobile number. And after using this OTP user can view the dashboard and perform banking operations.

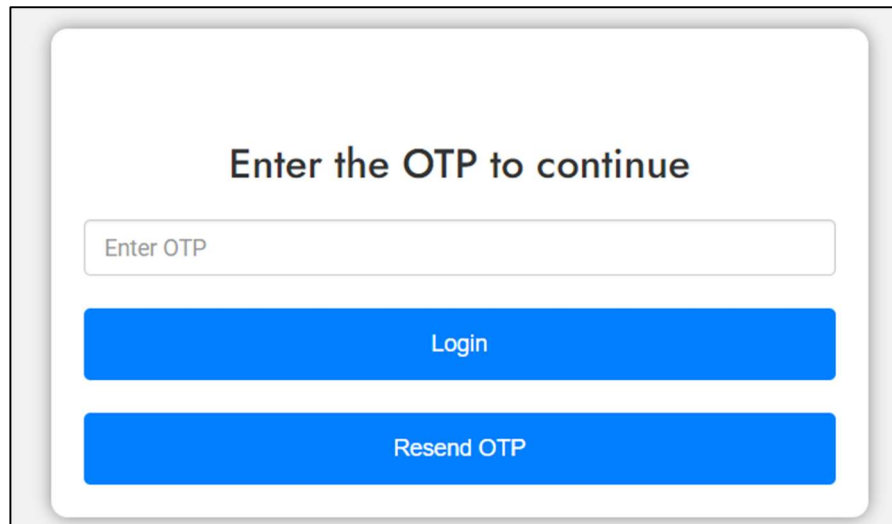


Figure 3.24: Resend OTP Button

```
<?php
require __DIR__ . '/vendor/autoload.php';
use Twilio\Rest\Client;
session_start();
$password = $_SESSION['password'];
$otp_code = rand(1000, 9999);
$_SESSION['otp_code'] = $otp_code;
$to = '+91' . $password;

$sid = "ACcf35b617cbe3cc34de807c560c02e2c5";
$token = "4ca3a1a69551a2fb9b45aac02b53ea8c";
$client = new Twilio\Rest\Client($sid, $token);

$client->messages->create(

    $to,
    array(
        'body' => 'Your OTP code is: ' . $otp_code,
        'from' => '+14147518834'
    )
);

echo "OTP has been sent";

header("Location:otp_verification.php");
exit();
?>
```

Figure 3.25: Code for Resend OTP

3.4.8 SENDING EMAIL CONFIRMATION

We have used the PHP Mailer library, in order to send mail to the users and notify them when someone has sent them money so that they are updated with the process of the transaction.

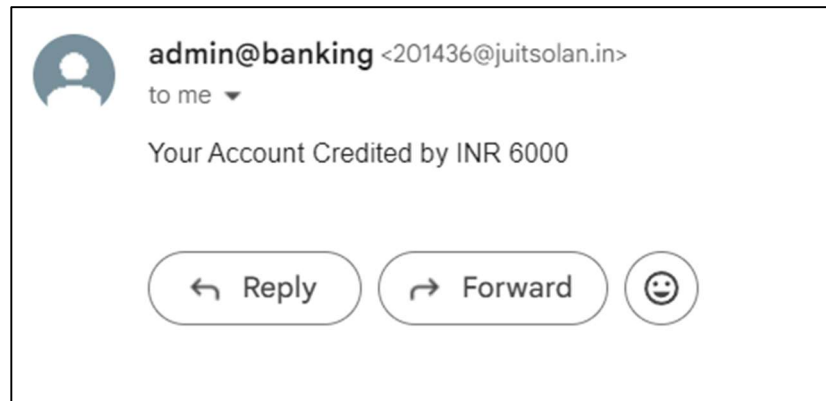


Figure 3.26: Notification received when user receives the money

3.4.9 ADDED MPIN TO VIEW ACCOUNT BALANCE

We have added the MPIN feature as the user will have to choose 4-digit MPIN during registration and after entering that 4-digit MPIN, the user will be allowed to view the balance.

```
if(isset($_POST['submit'])) {  
  
    $pin = $_POST['mpin'];  
    $aad_num = $aadhar_number;  
    $sql = mysqli_query($con,"SELECT * FROM `customer` WHERE `c_aadhar`='".$aad_num'");  
    $row = mysqli_fetch_array($sql);  
    if (password_verify($pin,$row['c_pin']))  
    {  
        session_start();  
        $_SESSION['aadhar_number'] = $row['c_aadhar'];  
        header("Location:money.php");  
        exit();  
    }  
}
```

Figure 3.27: Code for MPIN

3.5 KEY CHALLENGES

Some key challenges that we faced while developing this project are:

- 1. Security Threats and Risks:** One major challenge was to apply the algorithm so efficiently that it does not lead to any kind of inaccuracy as a little bit of error would in result loss of personal information of user which could ultimately result in financial loss.
- 2. Separate formats of the image:** At the time of encryption, we had saved the image in form of jpeg but when we were superimposing the image then we were creating the resulted image in the form of png this resulted in format mismatch and the hidden text was not being displayed.
- 3. Use of API for sending Email:** At the beginning we had used Mailgun API for sending the email to the users, but it proved to be very time consuming, therefore after doing some research we came to the conclusion of using the inbuilt PHPMailer which has the TLS encryption for security and also there is no limit in it for sending the emails.

CHAPTER 4: TESTING

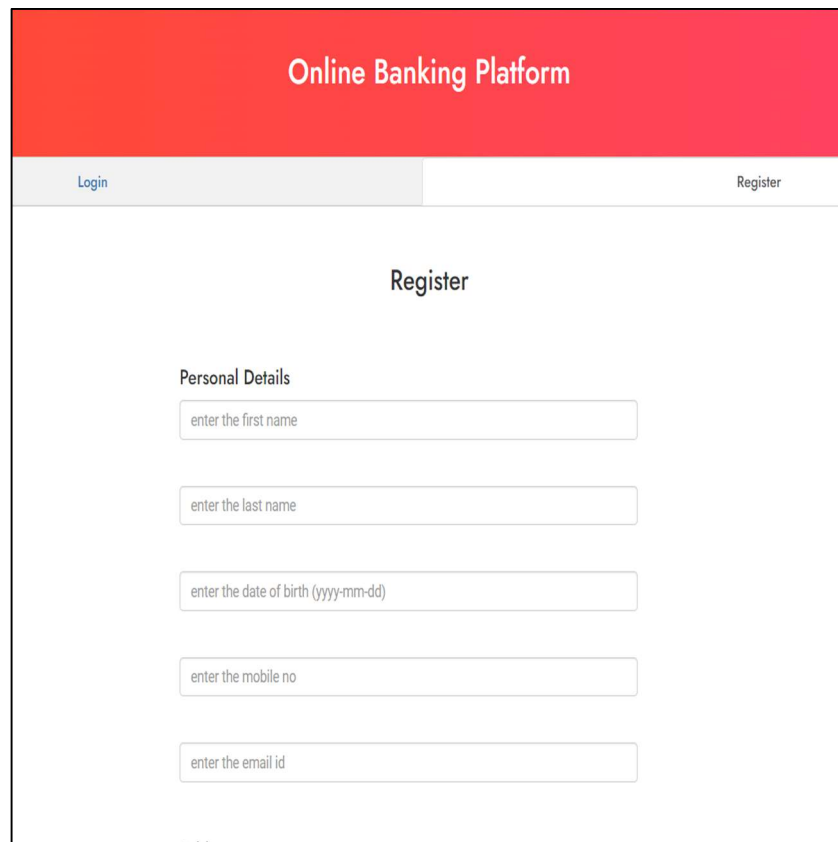
4.1 TESTING STRATEGY

The main aim of the testing strategy is to make our project online banking platform a very secure and reliable platform. We will test the system by making sure that the system is working efficiently with the security measures such as login through OTP, AES-128 encryption, visual cryptography and overall system working. So, by this, we ensure that we protect our customer's personal information and prevent phishing attacks which could in turn lead to financial loss. Therefore, we have developed the strategies keeping in mind all the risks involved which could pose a potential harm to our customers.

Various types of testing techniques have been deployed such as firstly during registering we have added checks to see whether any specific detail such as email, phone number, Aadhar number, pan card number is not being repeated. If any user enters the same details which matches with the data stored in database, then there will be error in registering and user will be asked to enter correct details. After this during login, user will be required to complete two factor authentication, after entering correct password and OTP will be sent on the user's registered mobile number and then only after successful authentication user will be authorized to the dashboard of the account. Then to view balance user will be required to enter correct pin which will be chosen by the user. After this, during money transfer, the user will be required to upload the correct share of image which the user already received during registration and after superimposing that image with the image stored in our database and if it matches the captcha will be displayed after which the user can complete transfer of money.

All these testing strategies are applied in order to make sure that our customers personal and sensitive data is safe from attacks such as phishing and pharming, internet scams and from hackers trying to steal financial information of users. Ultimately, this strategic testing aims to identify issues and correct them, ensuring the security of this online banking platform. And lastly it will make the users feel confident while users using our platform as they know that their data is safe from any kind of malicious activity, no one can hack into their account and they will not face any kind of financial loss.

4.2 TEST CASES AND OUTCOMES



The screenshot shows the 'Online Banking Platform' registration page. At the top, there is a red header with the text 'Online Banking Platform'. Below the header, there are two tabs: 'Login' and 'Register', with 'Register' being the active tab. The main content area is titled 'Register' and contains a section for 'Personal Details'. This section includes five input fields: 'enter the first name', 'enter the last name', 'enter the date of birth (yyyy-mm-dd)', 'enter the mobile no', and 'enter the email id'.

Figure 4.1: Registration Page of the Online Banking Platform

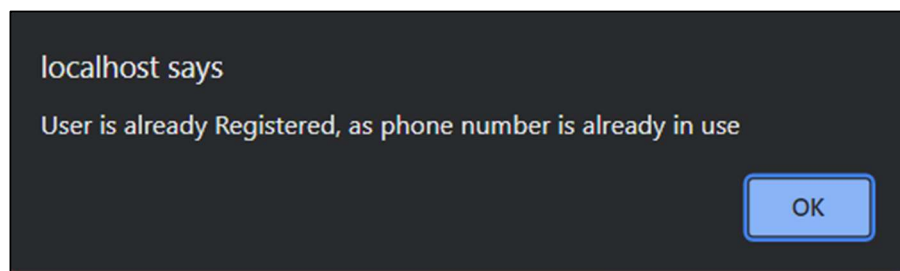


Figure 4.2: Error displayed as same Phone Number is registered twice

Figure 4.3: Login Page of the Online Banking Platform

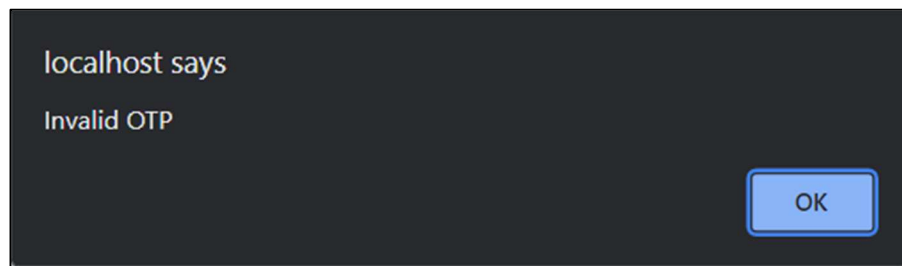


Figure 4.4: Error displayed as wrong OTP entered

	c_id	c_fname	c_mobile	c_email	c_address	c_photo	c_lname	c_dob	c_city
	26	Satvik	8299013641	tripathisid24@gmail.com	H.No 3/13 Vishnupuri Colony Kursi Road Aliganj	passport_picture_1691693876_72749.jpg	Tripathi	2002-01-24	Lucknow
	27	Harshit	7376921711	harshitupadhyay1919@gmail.com	Lucknow	harshit.JPG	Upadhyay	2001-12-19	Lucknow

Figure 4.5: Details are stored in the Database after successful Registration

Authentication area

Upload First Share

No file chosen

Captcha

Figure 4.6: Money Transfer authentication area to upload share

Details of the Receiver

Figure 4.7: To enter details to send money after uploading the share

Sr no	First Name	Last Name	Mobile no	Email id	Account no	Amount
1	Rushil	Wadhavan	7376921711	201436@juitsolan.in	1254789630	6000
2	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	7463528374	6000
3	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	1254789630	200
4	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	1254789630	6000

Figure 4.8: Transaction history page showing transaction details

Figure 4.9: To enter MPIN

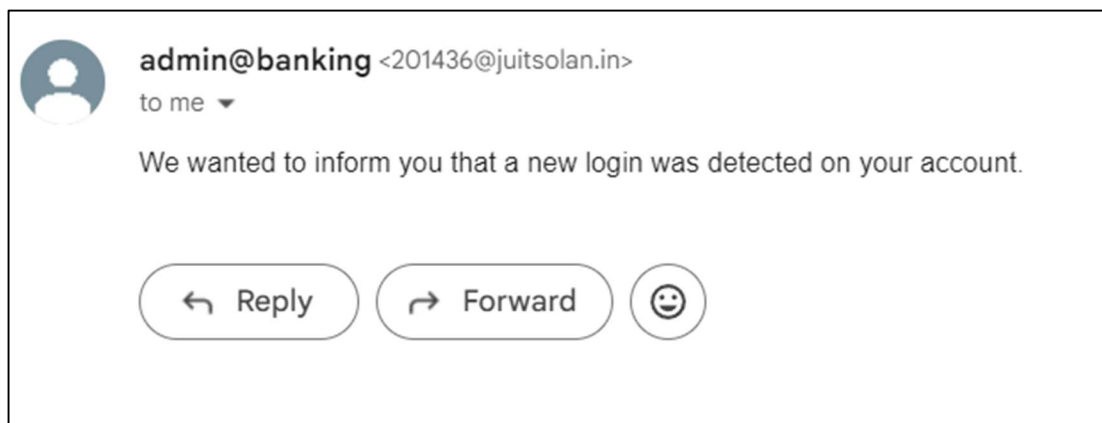


Figure 4.10: Email to notify Login Information

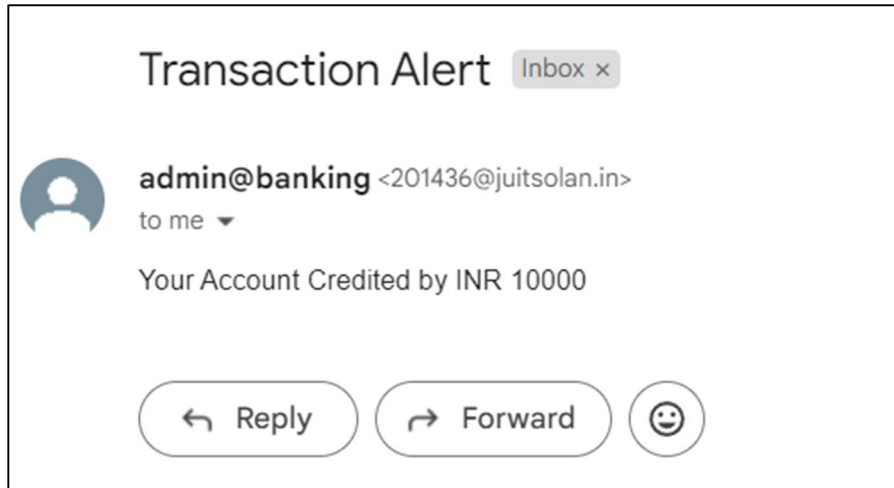


Figure 4.11: Credit Email sent to receiver

CHAPTER 5: RESULTS AND EVALUATION

5.1 RESULTS

In our project integrating multiple security features in an online banking platform, we added many encryption algorithms and security features which will overall result in the robust security of the overall online banking platform. These features are:

- 1. Two Factor Authentication:** During the login phase we have used the 2FA, firstly the user will enter password and if that entered password is correct then only OTP will be sent to the registered mobile number. We have used Twilio API for this. This will prevent hackers from gaining any unauthorized access to the account.

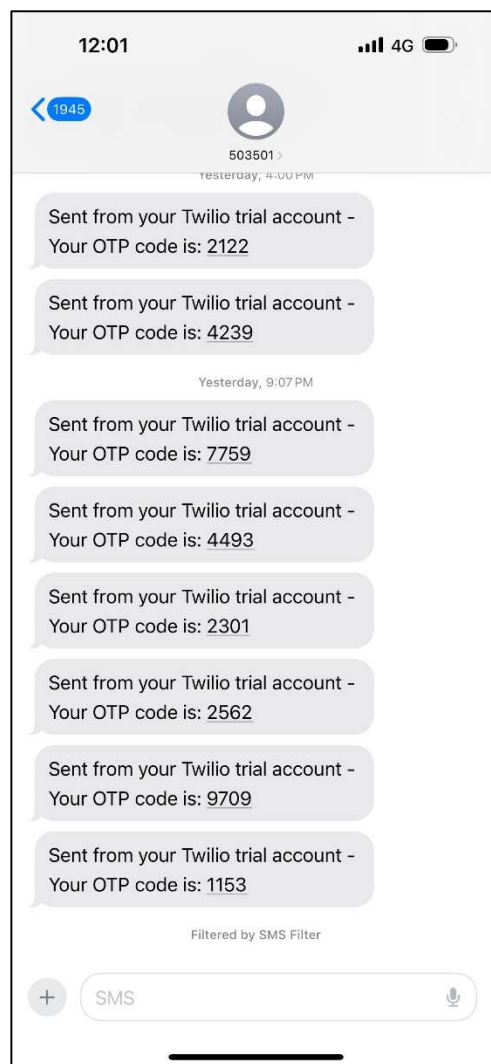


Figure 5.1: OTP received on the Registered Mobile Number.

- AES-128 Encryption:** We have applied the AES-128 encryption algorithm to secure the database and to encrypt the captcha text value before sending it to the database, in order to secure the original text and to make sure it is not leaked or hacked.

c_address	c_photo	c_name	c_dob	c_city	c_district	c_state	c_country	c_aadhar	c_pan	upload_aadhar	upload_pan	captcha_text
H.No 3/13 Vishnupuri Colony Kursi Road Aliganj Lucknow	passport_picture_1691693876_72749.jpg	Tripathi	2002-01-24	Lucknow	Lucknow	Uttar Pradesh	India	313131313131	12345ABCDf	WhatsApp Image 2023-11-17 at 22.43.14.jpeg	photo_2023- 04-04_23-17- 14 (2).jpg	M2K02x4= =
	harshit.JPG	Upadhyay	2001-12-19	Lucknow	Lucknow	Uttar Pradesh	India	895710744949	ALRPU5531G	WhatsApp Image 2023-11-17 at 22.38.00.jpeg	WhatsApp Image 2023- 11-17 at 22.37.48.jpeg	WnMUmpg= =

Figure 5.2: Encrypted Value Captcha Text stored in the Database

- Visual Cryptography:** The most important is the VC algorithm which mostly resulted in security and privacy of our online banking platform. It basically sends a first share of an image to the email of the user which will be further used by the customer to transfer the money.

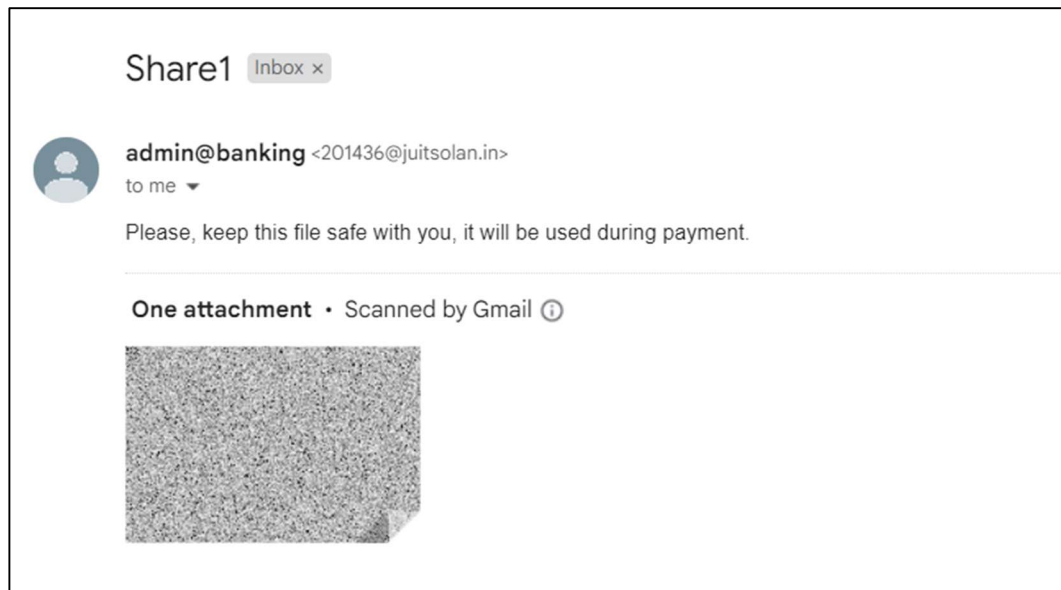


Figure 5.3: Share 1 received on the Email of the User

- Bcrypt Hashing:** We used this cryptographic hashing algorithm in order to convert the passwords in the form of a hash and then save them into the database, this helped us to stop any unauthorized user from entering our system and also in preventing any form of data breach. Bcrypt uses an adaptive hashing algorithm due to which it takes

a larger amount of time in generating hash and thus hackers find it difficult to hack the passwords using brute force attacks.

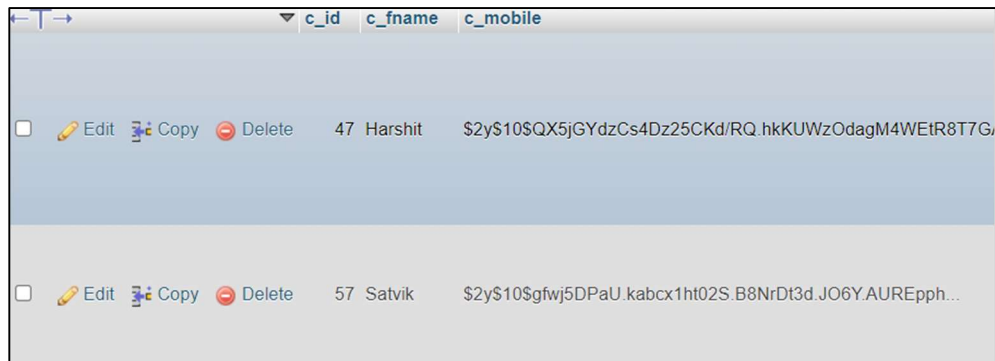


Figure 5.4: Password hashed through Bcrypt

- 5. Email Notification:** We are sending a confirmation email to the recipient that the transaction is successful and what amount of money the user has received.



Figure 5.5: Confirmation Email

- 6. MPIN Feature:** After entering correct MPIN user will be allowed to view the balance.

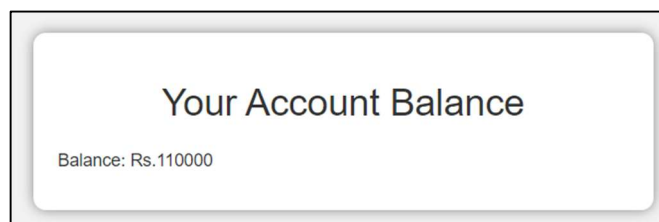


Figure 5.6: Showing Balance Correctly

- 7. Login Email:** An email is sent just after the login to notify the user that there is some activity happening.

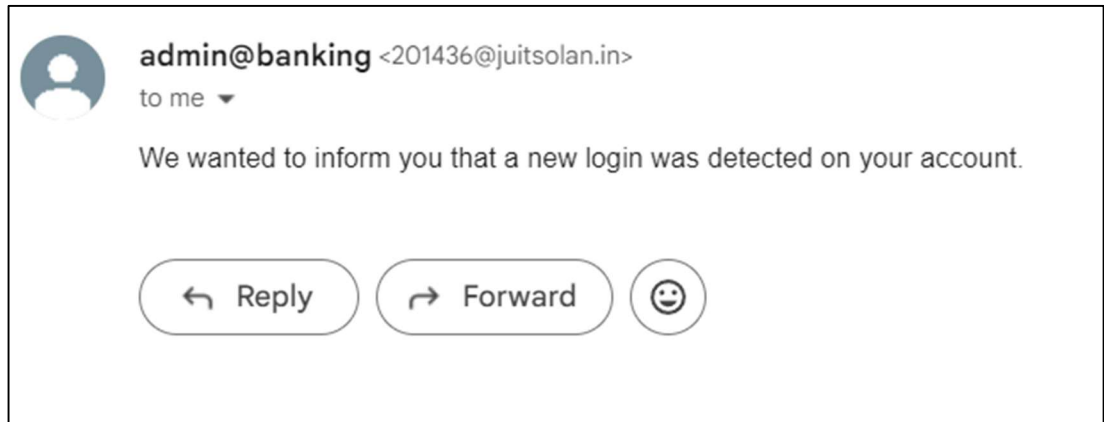


Figure 5.7: Login Email

- 8. Transaction details stored in the database:** After transferring the money, the details of the transaction are stored in the database.

	transaction_id	c_aadhar	c_fname	c_lname	c_mobile	c_email	c_accno	amount
<input type="checkbox"/> Edit Copy Delete	1	313131313131	Satvik	Tripathi	8091521753	tripathisid24@gmail.com	1324569870	50000
<input type="checkbox"/> Edit Copy Delete	2	313131313131	Harshit	Upadhyay	7376921711	201436@juitsolan.in	2154918764	10000
<input type="checkbox"/> Edit Copy Delete	3	895710744949	Rushil	Wadhavan	7376921711	201436@juitsolan.in	1254789630	6000
<input type="checkbox"/> Edit Copy Delete	4	895710744949	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	7463528374	6000
<input type="checkbox"/> Edit Copy Delete	5	895710744949	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	1254789630	200
<input type="checkbox"/> Edit Copy Delete	6	895710744949	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	1254789630	6000
<input type="checkbox"/> Edit Copy Delete	7	895710744949	Rushil	Wadhavan	1273635271	harshitupadhyay1919@gmail.com	1254789630	6000
<input type="checkbox"/> Edit Copy Delete	8	313131313131	Satvik	Sharma	8922341657	201239@juitsolan.in	1547896235	5000

Figure 5.8: Transaction details stored in database

CHAPTER 6: CONCLUSIONS AND FUTURE SCOPE

6.1 CONCLUSION

To conclude, the implementation of our project integrating multiple security features in an online banking platform, was a hard task because any breach of security at any point of time would result in financial loss. So, to add encryption and cryptographic algorithms such as AES-128 encryption algorithm and visual cryptography along with two factor authentication helped us to protect the privacy of our users and in turn saving their sensitive information from getting leaked.

So firstly, let us talk about 2FA, the initial step is to register and after successful registration the user will be able to login. During login phase, if the user enters the correct password, an OTP will be sent to the registered mobile number, and only after entering correct OTP user will be able to login. By implementing 2FA, we are having an additional layer of security beyond passwords, and by this we prevent unauthorized access and breaches.

By implementing AES-128 encryption, we are able to secure the randomly generated captcha text by storing it in the database in an encrypted format. After this we implemented visual cryptography through which were able to encrypt the captcha text without any problems and were able to differentiate legitimate user from the illegitimate ones.

Using the above encryption algorithms, we built a strong and secure online banking platform, with enhanced security features and performance, which would result in building of trust of user towards our platform and a sense of confidence while using it with an assurance in mind that the details are safe and secure. Also, our aim is to educate user about best security practices and how to use this system and strengthen the system's overall resilience against potential threats.

The project's conclusion signifies a substantial pace towards establishing a secure online banking system, meeting very high security standards and assuring users of a safer and more trustworthy banking experience.

6.2 FUTURE SCOPE

The future scope of this project which is integrated with highly advanced security features demands of continuous improvements, innovation and adaptation to emerging technologies. Some areas for future development where we will work are:

- 1. Making an Administrator panel:** We will make an interactive admin panel where the administrator will be able to see the overview of all the things that are happening such as all the transactions taking place, any unusual activity in any of the accounts, will be able to remove or block any account trying to perform malicious activities.
- 2. Building a complete ecosystem:** The work that we have done till now is that we have added multiple security features in order to perform a transaction but that transaction is only happening from one account. We can build an overall system where one person sends money and the other one receives it and this is visible in their transaction history section. Also, we can make an integrated third-party website where the user can pay bills or an e-commerce website or a stock investing platform.
- 3. Aadhaar based authentication:** Although we have done an Aadhaar check at the time of registration we can add additional feature in which the OTP is sent to the mobile number which is registered with the Aadhaar.
- 4. User Experience and Scalability:** We can also use some additional features to make the user experience more enhanced and to scale the overall system so that it can handle large load of users.

REFERENCES

1. “Digital India Under Cyber Attack,” [indiandefenceoverview.com http://www.Indiandefencereview.com/spotlights/digital-india-under-cyber-attack/](http://www.Indiandefencereview.com/spotlights/digital-india-under-cyber-attack/)(accessed Nov. 30,2023)
2. A.G. Patil, A. Hasbe, Y. Kore, A. Sakale, “Secure E-Banking application using visual cryptography”, International Research Journal of Engineering and Technology, Vol.1, March 2023.
3. Y. Shah, R. Rane, S. Kharade, R. Patil, “Analysis of AES and DES algorithm”, International Journal of Trend in Research and Development, Vol. 2 April 2020.
4. K.Dheeraj, “A study on secure system in online banking system”, Journal of Emerging Technologies and Innovative Research, Vol. 5, October 2018.
5. Chandrasekhara, Roopalakshmi R., “A Novel approach of secure banking application using visual cryptography against fake website authenticity theft”, International Journal of Engineering Research and Technology, Vol. 2, April 2013.
6. D.R. Moscato, S. Altschuller, “International Perception of Online Banking security Concerns”, Communications Of the IIMA, Vol. 12, May 2012.
7. H.O. Alanazi, R. Alnaqeib, A.K. Hmood, M.A. Zaidan, “On the module of internet banking system”, Journal of Computing, Vol. 2, May 2010.
8. P.Rajguru, J.Dhomse, P.Pawar, “Securing online transaction using visual cryptography”, Journal of Telecommunication System and Management, Vol.2, May 2018
9. V. Vadgam, B. Tanti, C. Modi, N. Doshi, “A novel approach for E-Payment using Virtual Password System”, International Journal on Cryptography and Information Security, Vol. 1, December 2011.
10. B. Balakumar, R. Sivakumar, V.A. Pandeewaran, “A study of encryption algorithms (DES, 3DES and AES) for Information Security”, International Research Journal of Engineering and Technology, Vol. 5, April 2018.
11. M. Tan, T.S.H.Teo, “Factors influencing the adoption of internet banking”, Journal of the Association for Information Systems, Vol. 1, January 2000.

12. R.O. Akinyede O.A. Esese, "Development of a secure mobile e-banking system", International Journal of Computer, Vol. 26, September 2017.
13. N.P. Vishnu, T.M. Kishor, A.Raza, N. Gayatri, "Implementing securing online banking using visual cryptography schemes using QR code.", Journal of Emerging Technologies and Innovative Research, Vol. 7, August 2020.
14. "Visual Cryptography," geeksforgeeks.org. <https://www.geeksforgeeks.org/visual-cryptography-introduction/> (accessed Aug. 20,2023)
15. "Visual Cryptography." cs.jhu.edu. <https://www.cs.jhu.edu/~fabian/courses/CS600.624/NaorShamir-VisualCryptography.pdf> (accessed Aug. 20,2023)
16. "Advanced encryption standard." wikipedia.org. https://en.wikipedia.org/wiki/Advanced_Encryption_Standard (accessed Aug. 22,2023)
17. "AES vs DES." javatpoint.com. <https://www.javatpoint.com/des-vs-aes> (accessed Aug. 22,2023)
18. "IRJET." irjet.net. <https://www.irjet.net/> (accessed Sep. 1,2023)
19. "HTML." w3schools.com. <https://www.w3schools.com/html/> (accessed Sep. 13,2023)
20. "CSS Tutorial." tutorialspoint.com. <https://www.tutorialspoint.com/css/index.htm> (accessed Sep. 15,2023)
21. "Learn JavaScript Tutorial." javatpoint.com. <https://www.javatpoint.com/javascript-tutorial> (accessed Sep. 18,2023)
22. "PHP Tutorial." geeksforgeeks.org. <https://www.geeksforgeeks.org/php-full-form/> (accessed Sep. 20,2023)
23. "Bootstrap Tutorial" w3schools.com. <https://www.w3schools.com/bootstrap5/> (accessed Oct. 12,2023)
24. "SQL Tutorial." w3schools.com. <https://www.w3schools.com/sql/> (accessed Oct. 13,2023)
25. "How to send Emails in PHP Using PHPMailer Library" Cloudways.com. <https://www.cloudways.com/blog/send-emails-in-php-using-phpmailer/> (accessed

Oct. 23, 2023)

26. "Lucidchart." lucidchart.com <https://www.lucidchart.com/pages/examples/flow-chart-maker> (accessed Oct. 24,2023)

27. "Twilio." twilio.com. <https://www.twilio.com/blog/what-does-twilio-do> (accessed Oct. 25,2023)

28. "GD - Manual" Php.net. <https://www.php.net/manual/en/book.image.php> (accessed Nov. 1, 2023)

APPENDIX

Here is the plagiarism certificate of the report:

G30

ORIGINALITY REPORT

12% SIMILARITY INDEX	9% INTERNET SOURCES	2% PUBLICATIONS	7% STUDENT PAPERS
--------------------------------	-------------------------------	---------------------------	-----------------------------

PRIMARY SOURCES

1	nordvpn.com Internet Source	2%
2	scholarworks.lib.csusb.edu Internet Source	1%
3	Submitted to Colorado Technical University Online Student Paper	1%
4	iitmjanakpuri.com Internet Source	<1%
5	ia804506.us.archive.org Internet Source	<1%
6	www.irjmets.com Internet Source	<1%
7	fastercapital.com Internet Source	<1%
8	www.coursehero.com Internet Source	<1%
9	www.geeksforgeeks.org Internet Source	<1%
