# A CASCADE MULTI-STAGE ONE-TIME PASSWORD, TEXTUAL AND RECALL-BASED GRAPHICAL PASSWORD FOR ONLINE AUTHENTICATION

**BY**

**HARUNA, Adamu**
**MTech/SICT/2018/9195**

**A THESIS SUBMITTED TO THE POSTGRADUATE SCHOOL, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGER STATE IN PARTIAL FUFILMENT OF THE REQUIREMENTS FOR THE AWARD OF MASTER DEGREE IN COMPUTER SCIENCE**

**APRIL, 2023**

# ABSTRACT

The most widely used method of computer authentication is text passwords. This method has been discovered to have several drawbacks. Passwords that are easy to guess, for example, are frequently chosen by users. On the other hand, a difficult-to-guess password is also difficult-to-remember. Brute-force and keylogger attacks are also possible with textual passwords. Based on the notion that people recall visuals better than text, graphical passwords have been advocated in the literature as a potential replacement to alphanumerical passwords. However, existing graphical passwords are vulnerable to a shoulder surfing attack. This thesis presents an authentication system for online applications based on a combination of one-time passwords, textual, and graphical passwords to address these security weaknesses. Usability testing and security analysis procedures were used to confirm the suggested system's efficiency. Thirty people participated in the system evaluation. Twenty-seven out of the thirty gave the solution, scored 80% in terms of efficiency, security, reliability and seamless ability, while three out of the thirty have concern about network unavailability for one time password dropping. The findings of the security study revealed that the proposed system met its primary security requirements. The usability test revealed that the suggested system is simple to operate, friendly, and secure. This study demonstrated superior usability and security when compared to conventional authentication technologies.

# TABLE OF CONTENTS

# LIST OF FIGURES

**FIGURE** **PAGES**

# LIST OF TABLES

**TABLE** **PAGES**

# CHAPTER ONE

# 1.0 INTRODUCTION

## 1.1 Background to the Study

Passwords, which are the most important aspect of the authentication process, are crucial to information and computer security(Das *et al*., 2012). User Authentication is a process that allows a device to verify the identity of a person who connects to network resources. For all websites and applications, textual passwords are the most utilized form of authentication (Sun *et al*., 2012). Authentication refers to the act of only showing the belongings to their rightful owner (O'Gorman, 2003). Authentication is also the first line of defense in protecting any resource. A password is a type of secret authentication data used to control resource access. Those who are not given access are kept in the dark about the password, and those who have login details are tested to see if they know it before being granted or refused access. Passwords are required for a variety of tasks by a normal computer user, including login into accounts, getting email from servers, accessing files, databases, networks, and web sites, and even reading the morning newspaper online(Karode *et al*., 2013). Nowadays, a variety of user authentication mechanisms are accessible.

Textual passwords are made up of a series of letters and numbers that may or may not include special characters or numbers. In most circumstances, users can log into several accounts with just one username and password (Fulkar *et al*., 2012). However, they are not completely secure. As a result, users should choose strong passwords that include digits, uppercase, and lowercase letters. These textual passwords are then thought to be strong enough to withstand brute force attacks. A strong textual password, on the other hand, is difficult to remember and recall (Stobert, 2015). Along these lines clients will in general prefer passwords that are either brief or derived from the word reference, rather than irregular alphanumeric strings (Alt *et al*., 2016). The weakest link in the authentication chain is considered to be human actions such as choosing terrible passwords for new accounts and typing incorrect passwords in an insecure manner for later logins. Textual passwords are also vulnerable to password replay and key-logger attacks (Yenape & Waghmare, 2017).

A significant number of graphical password schemes have been developed and tested to overcome this issue with alphanumeric authentication (Jadhav *et al*., 2014). One explanation for the surge in popularity of graphical passwords is because visuals, as opposed to strings of characters, are thought to be more remembered. Using graphics or drawings as passwords is referred to as graphical passwords. Humans recall visuals better than text, therefore graphical passwords should be easier to remember (Karode *et al*., 2013). Furthermore, because the search space is nearly endless, they should be more resistant to brute-force attacks. In general, there are two types of graphical password procedures: recognition-based and recall-based graphical techniques. A user gets authenticated using recognition-based techniques by asking him or her to identify one or more photos at the registration stage. A user is requested to recreate something that he or she made or selected earlier during the registration process in recall-based approaches (Kenneth & Olujuwon, 2021).

One of the drawbacks of using a graphical password technique is the possibility of shoulder surfing (Kenneth & Olujuwon, 2021). A graphical password could be physically witnessed, especially in public locations, and if the attacker has a clear visual of the password being inserted several times, they could easily crack the password, which is a serious weakness (Ometov *et al*., 2018). Another disadvantage of a graphical password technique is that it is vulnerable to guessing. If the user only registered a short and predictable password, the odds of it being guessable would increase, just like with an alphanumeric password.

To overcome these potential limitations, Chuen *et al*. (2020) proposed that a shoulder surfing resistant technique be implemented, such as requiring at least 10 click points to make the graphical password stronger and implemented to the system to ensure that the user does not just enter a sloppy password, reducing the chances of an attacker guessing the user's password dramatically. A graphical password with a significant number of clicks is more difficult for the owner to remember and consumes more memory space, which is a disadvantage of this proposed approach (Fulkar *et al*., 2012; Jadhav *et al*., 2014).

There are several methods for avoiding keyloggers, shoulder surfing, and replay attacks, but none of them are sufficient on their own. To effectively eliminate the problem, a mix of techniques must be used (Santwana, 2014). This research attempts to solve the issues of shoulder-surfing, replay and key-logging attacks using a combination of one-time password, textual and graphical password.

## 1.2 Statement of the Research Problem

Many flaws exist in today's authentication schemes. Textual passwords have long been known to be vulnerable. The problem of textual passwords is largely due to human long-term memory limits (Afandi & Jali, 2017). After choosing and memorizing a password, the user must be able to recall it to log in. People, on the other hand, frequently forget their passwords (Chuen *et al*., 2020). Users are more likely to use short passwords or passwords that are easy to remember, leaving them vulnerable to attackers. Furthermore, textual passwords are subject to brute force attacks, dictionary attacks, keyloggers, social engineering, and shoulder surfing (Por *et al*., 2017; Salim Istyaq, 2018; Togookhuu & Zhang, 2017; Vaddeti *et al*., 2020). Graphic password authentication techniques have been presented in the literatures to overcome the constraints of text-based passwords. Graphical password strategies, on the other hand, are vulnerable to shoulder surfing. When a graphical password is physically witnessed, especially in public locations, and the attacker has a clear visual of the password being inserted several times, they can simply crack it, which is a serious weakness (Ometov *et al*., 2018). Another disadvantage of a graphical password technique is that it is vulnerable to guessing. Chuen *et al*. (2020) proposed that a graphical password with ten or more click points can resist guessing and shoulder surfing attacks. However, to store each click, this proposed approach necessitates additional memory space, making it more difficult for the user to remember (Fulkar *et al*., 2012). An authentication system based on one-time passwords, textual and graphical passwords is presented to address the highlighted limitations of shoulder surfing, guessing, key-logger attack, memorability, and huge memory space requirements.

## 1.3 Aim and Objectives of Study

The aim of this research is to develop a cascade multi-stage One-Time Password (OTP), textual and recall-based graphical password for an online authentication.

The objectives of this project work are:

i. To develop a cascade multi- stage authentication technique.

ii. To evaluate the performance of the proposed technique in (ii) using unit testing, usability testing and security analysis.

## 1.4 Scope of the Research

The primary goal of this research is to improve user authentication on online applications by employing a multi-stage authentication technique based on OTP, textual and cued clicked point recall-based graphical passwords, without taking into account recognition-based graphical passwords, visual cryptography, or advanced encryption standard cryptography.

## 1.5 Significance of the study

The significance of this study cannot be overstated owing to the numerous areas in which it can be applied. This research will contribute to the field of cyber-security by developing a secure multi-level security system that safeguards online user information. Furthermore, this study will help researchers better understand the design and execution of a graphic password authentication strategy, as well as existing user authentication techniques. Furthermore, this research will aid in preventing intruders from obtaining user passwords through shoulder surfing, guessing, dictionary, and key logger attacks. As a result, the user's privacy and integrity are safeguarded.

## 1.6 Organization of the Thesis

This research work consists of five chapters ranging from Chapter one to Chapter five. The background of the study is discussed in the first chapter. It consists of the problem statement, aim and objectives, the study's scope, and significance of the study. A review of the previous related literatures is presented in chapter two. The research methodology is presented in chapter three. This includes the techniques of authentication. Chapter four presents the details of the actual experimentation conducted and the results obtained compared to existing methods. Conclusions were drawn in chapter five, as well as suggestions for future research.

# CHAPTER TWO

# 2.0 LITERATURE REVIEW

## 2.1 Authentication

Authentication is the process or action of authenticating a user's or process' identity in computing. It's the process of connecting a set of identifying credentials with an incoming request. When you log in to a website, you typically provide credentials such as your username and password. On a local operating system or within an authentication server, the credentials provided are compared to those on a file in a database containing the authorized user's information (Lal *et al.*, 2016).

The authentication process always runs at the start of an application, before the permission and throttling checks occur, and before any other code is allowed to proceed (Peisert *et al.*, 2013). To verify a user's identity, different systems may require different sorts of credentials. The credential is frequently in the form of a password, which is kept private and only known by the user and the system. There are three ways to verify someone's authenticity (Ometov *et al.*, 2018): something the user knows, something the user is, and something the user has. Something the user knows mostly requires the individual to get access to the system by typing the username, and pin or password. Something the user has is where the user uses smart card for authentications.

## 2.2 Classification of Current Authentication Methods

Currently the authentication methods can be broadly divided into three main areas. Token-based, Biometric-based, and Knowledge-based authentication (Chiasson *et al*., 2012;

Ibrahim *et al.*, 2019; Masdari & Ahmadzadeh, 2016). This classification is shown in the Figure 2.1.

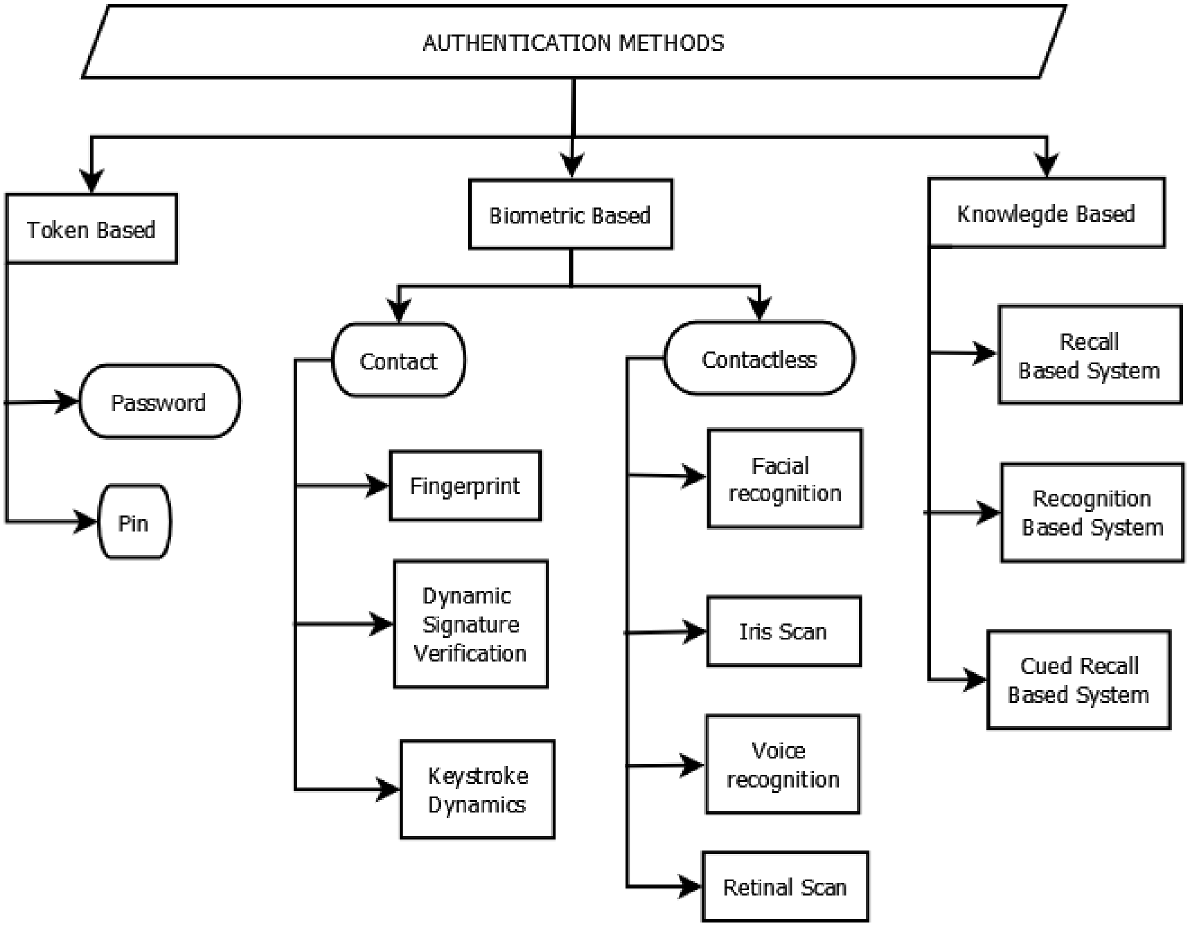


Figure 2. 1 Classification of Current Authentication Methods *(Mathuri & Valarmathi, 2013)*

In figure 2.1 token-based authentication consist of password and pin. The biometric-based authentication is dived into contact and contactless authentication. The contact biometric include fingerprint, dynamic signature, and keystroke dynamics. The contactless biometric consist of iris and retinal scan, voice, and face recognition. Lastly, the knowledge-based authentication consist of recall- recognition and cued recall-based authentication.

### 2.2.1 Token-Based Authentication

Token based authentication is based on “Something You Possess” like smart Cards, a driver’s license, credit card, and a national ID card. It allows users to input their login and password to get a token that allows them to access a specified resource without having to enter their username and password (O’Gorman, 2003). The user can then offer their token - which grants access to a specified resource for a set length of time - to the distant site (Ethelbert *et al*., 2017). To improve security, many token-based authentication systems include knowledge-based approaches (Jeong & Kim, 2015).

### 2.2.2 Biometric-Based Authentication

The study of automated systems for uniquely detecting persons based on one or more intrinsic physical or behavioral attributes is known as biometrics (Jin *et al*., 2004). It is based on “Something You Are” (Leng *et al*., 2014). It recognizes users based on physiological or behavioral factors such as fingerprint or facial scans, iris, or voice recognition. A biometric scanning device translates biometric data, such as an iris pattern or fingerprint scan, into digital information that can be interpreted and verified by a computer. Voice recognition, fingerprints, face recognition, iris scan, infrared facial and hand vein thermograms, retinal scan, hand and finger geometry, signature, gait, and keystroke dynamics are all biometric technologies that can be used in a biometric-based authentication system (Awasthi & Srivastava, 2013). To make a yes/no conclusion, biometric identification relies on computer algorithms. It improves user service by making identification quick and simple (Mishra *et al*., 2014).

### 2.2.3 Knowledge Based Authentication

The most widely used authentication approaches are knowledge-based strategies, which include both text-based and picture-based passwords (Chiasson *et al*., 2012). The knowledge-based authentication (KBA) verifies a user using "Something the User Know." A Personal Identification Number (PIN), password, or pass phrase, for example. It's a form of authentication that requires the user to answer at least one "secret" question (Hafiz *et al*., 2008). KBA is frequently used in multifactor authentication (MFA) and password retrieval via self-service. Knowledge-based authentication (KBA) has various advantages over classic (conventional) e-authentication methods such as passwords, public key infrastructure (PKI), and biometrics (Hafiz *et al*., 2008).

## 2.3 Classification of Graphical Password Based Systems

Humans can remember visuals better than words, hence graphical-based password strategies have been offered as a potential replacement to text-based password techniques. Using graphics or drawings as passwords is referred to as graphical passwords. Humans recall visuals better than text, therefore graphical passwords should be easier to remember (Yildirim, 2017). Furthermore, because the search space is nearly endless, they are more resistant to brute-force attacks. In general, graphical password approaches can be divided into two groups: Graphics approaches based on recognition and recall (Rajeh *et al.*, 2014).

### 2.3.1 Recognition-based Techniques

A user is authenticated using recognition-based techniques by motivating them to recognize one or more photos during the registration stage (Islam *et al*., 2020). In most recognition-based systems, users must memorize a portfolio of images during password formation, and then must recognize their own images among decoys to log in. Humans have a remarkable ability to recognize images they've seen before, even if they have just seen them briefly (Karode *et al*., 2013). Such methods are not appropriate substitutes for text password schemes from a security standpoint, as their password spaces are comparable in cardinality to merely 4- or 5-digit PINs (assuming a set of images whose cardinality remains reasonable, with respect to usability). Various forms of images, including people, random art, common objects, and icons, have been proposed for recognition-based systems (Afandi & Jali, 2017). In some graphical password schemes, the system must remember some shared secret facts, such as user-specific profile data. For example, in recognition schemes, the system must remember which photographs belong to a user's portfolio to display them. This information must be kept in such a way that it is accessible to the system in its original form (perhaps under reversible encryption), and thus to anyone who gains access to the stored data (Haque & Imam, 2014).

### 2.3.2 Recall-based Techniques

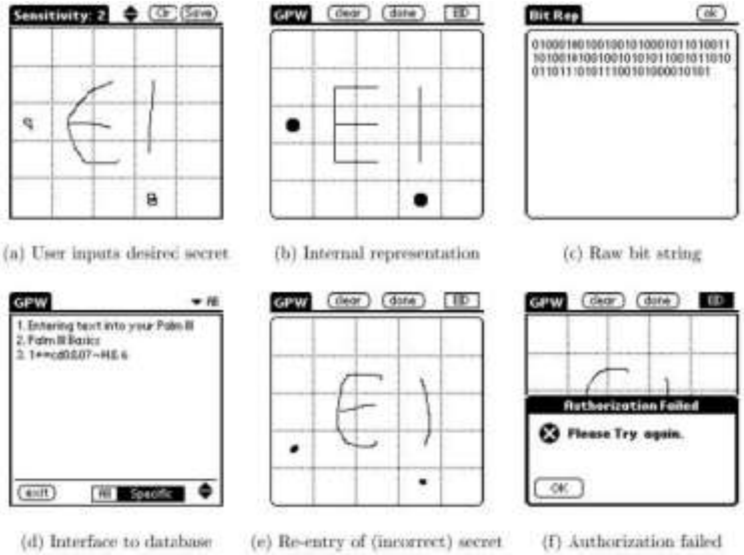
A user is requested to recreate anything that he or she made or selected earlier during the registration stage in recall-based approaches (Chuen *et al*., 2020). Recall-based graphical password systems are also referred to as draw-metric systems since users recall and replicate a secret drawing. Users often draw their passwords on a blank canvas or a grid (which may serve as a moderate memory aid) in these systems (Salim Istyaq, 2018). Recall is a difficult memory task since it is done without any memory prompts or clues. Users sometimes figure out how to utilize the interface as a cue even if it was not designed that way, turning the task into a cued-recall task with the same cue available to all users and attackers. There are two types of recall-based schemes: pure recall-based techniques and cued recall-based techniques (Ramalingam, 2017) .

### *2.3.2.1 Pure Recall-Based Techniques*

Users in this group must be able to reproduce passwords without the assistance or reminder of the system. Pure recall-based approaches include the Draw-A-Secret technique, Grid selection, and Pass doodle (Bhanushali *et al*., 2015).

#### a. Draw-A-Secret (DAS) Technique

As demonstrated in Figure 2.2, this method allows users to construct their own unique password by drawing a basic graphic on a 2D grid (Lin *et al*., 2010). In the sequence of the drawing, the coordinates of the grids occupied by the picture are saved. The user is asked to redraw the picture during authentication. The user is authenticated if the drawing touches the same grids in the same order. The complete password space of DAS is greater than the full text password space when given reasonable-length passwords in a 5 X 5 grid (Bhanushali *et al*., 2015).



(a) User inputs desired secret (b) Internal representation (c) Raw bit string

(d) Interface to database (e) Re-entry of (incorrect) secret (f) Authorization failed

Figure 2. 2 Draw-A-Secret Graphical Password (Lin *et al.,* 2010)

DAS, like other pure recall-based approaches, has a number of disadvantages. When it comes to DAS authentication, the majority of users forget their stroke order. In addition, users' passwords are subject to graphical dictionary attacks and replay assaults (Fulkar *et al*., 2012).

**b. Passdoodle Technique**

This is a graphical password that consists of handwritten drawings or text produced with a stylus onto a touch-sensitive screen. Cracking doodle passwords is more difficult, according to Goldberg *et al*. (2002), since they have a theoretically far larger number of possible doodle passwords than text passwords (Lashkari *et al*., 2010). Figure 2.3 shows an example of a Passdoodle password.

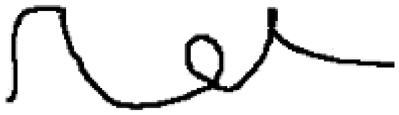


Figure 2. 3 Example of Passdoodle password (Lashkari *et al*., 2009)

Experiments with passdoodle revealed that people could recall whole doodle images with the same accuracy as alphanumeric passwords. Unfortunately, while the resulting artwork could be remembered entirely and perfectly, the order in which the doodles were drawn was highly unlikely to be remembered by the same users. The Passdoodle system is found to be vulnerable to a variety of assaults, including guessing, malware, keylogging, and shoulder surfing.

**c. Signature Technique**

This technique was first developed by Syukri *et al*. (1998), and it involves the user sketching their signature on a computer screen with a mouse. Registration and verification were the two stages of their method. During the registration stage, the user will be prompted to draw their signature with the mouse, after which the system will extract the signature region and increase or scale-down signatures, as well as rotate them if necessary (also known as normalizing). The information will be saved in a database afterwards (Gokhale & Waghmare, 2016). The verification stage receives the user input, performs another normalization, and then extracts the signature parameters. The system then performs verification using geometric average means and a database dynamic update. A sample of Syukri *et al*., (1998) signature authentication is shown in figure 2.4.

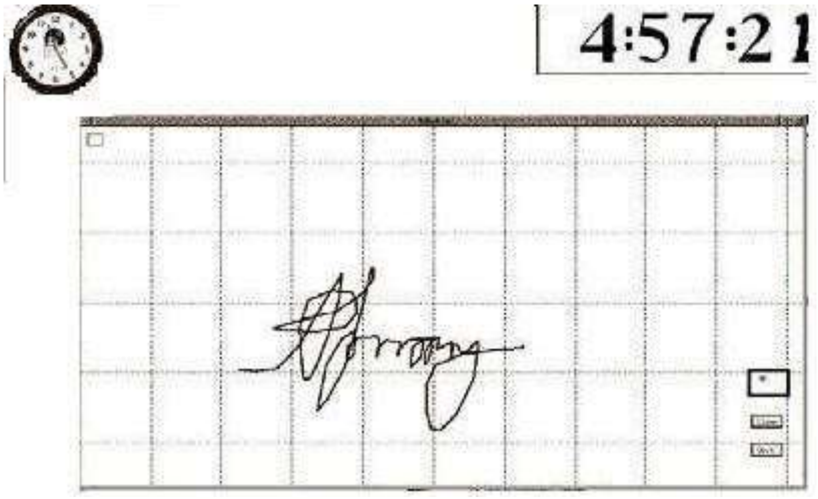


Figure 2. 4 Signature authentication(Syukri *et al.*, 1998)

The primary benefit of this method is that it eliminates the need to memorize one's signature, and signatures are difficult to forge (Navaz & Durairaj, 2016). However, not everyone is experienced with using a mouse as a writing tool, making the signature difficult to draw. A pen-like input device could be one answer to this problem, but such devices are not generally used, and adding additional hardware to an existing system can be costly.

#### *2.3.2.2 Cued Recall-Based Techniques*

In this strategy, the system provides certain tips that assist users in accurately reproducing their passwords (Nandgaonkar *et al*., 2019). Hot spots (regions) within an image will be used to provide these hints. To register as a password, the user must select one of these regions, and to log into the system, they must select the same region in the same order. The user must remember and keep the "selected click sites" secret. PassPoints and Cued Click Points, which will be described in the next sub-sections, are the most widely investigated graphical passwords in this category (Haque & Imam, 2014).

**a. PassPoints**

Any image can be utilized with the PassPoints approach. To generate a password, a user can click on points on an image. The graphic itself clearly serves as a reminder for the user's click-points. PassPoints is prone to "hotspots," according to security studies, and its users tend to utilize identical simple geometric patterns with photos (Haque & Imam, 2014). Hotspots are specific parts of an image that users choose as part of their passwords more frequently than others (Moraskar *et al*., 2014). Hotspots are a concern because attackers can forecast where hotspots will appear in an image based on the users' visual perceptions and image selection preferences. They can then create a password dictionary including combinations of these potential hotspots. This is the primary flaw in PassPoints, which is being addressed in a modified form known as the Cued Click Point Scheme to address the issue of hotspots (Patra *et al*., 2016). Figure 2.5 shows an example of passpoint authentication, where each clicked point on the image generates a password.

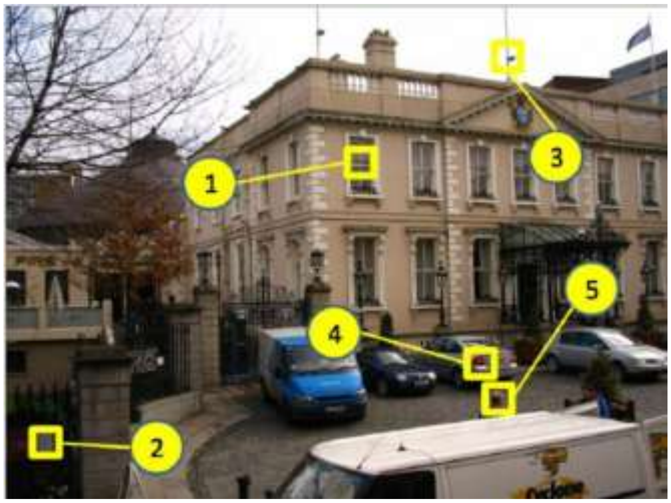


Figure 2. 5 Passpoint authentication (Lashkari *et al*., 2010)

**b. Cued Click Points**

Cued Click Points (CCP) approach mitigates some of the shortcomings of passpoints (Shantha *et al*., 2015). Users must click on one point per image on five separate images given in order in this manner (Sahu & Singh, 2014). After each click, a new image appears. The image that appears is decided by the previously specified click location (Nandgaonkar *et al*., 2019). To log on, legitimate users must click on the identical click locations in the image sequence. Authentication failure is reported at the end of the login procedure if users pick any incorrect click points (Swapnil *et al*., 2014). Figure 2.6 illustrates the implementation of CCP. In the illustration, a click on any point of the image lead to a new set of images to be click for authentication.

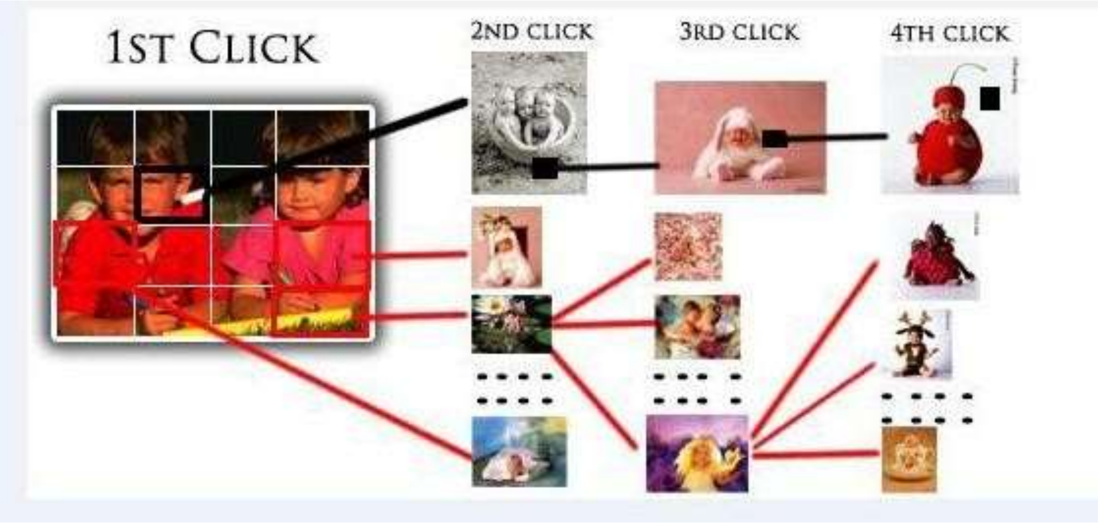


Figure 2. 6 Implementation of Cued Click Point (Moraskar *et al*., 2014)

## 2.4 Password Security Vulnerability

Passwords, both text-based and graphical, are subject to a variety of assaults. Shoulder surfing, credential stuffing, brute force, dictionary, spyware, and social engineering attacks are examples of these attacks.

### 2.4.1 Spyware Attack

Spyware is a program that collects statistical data from a computer and sends it over the internet without the user's permission. Spyware which is a computer virus allows hackers to access your personal information. Keylogger, mouse tracking, and key listening attacks are examples of spyware attacks. Key logging or key listening malware, with a few exceptions, cannot be used to crack graphical passwords (Fulkar *et al*., 2012). It's unclear whether "mouse tracking" spyware will be helpful in preventing graphical passwords. However, simply moving the mouse is insufficient to crack graphical passwords. Such data must be linked to application-specific information, such as window location and size, as well as time data (Bindu, 2015; Wazid *et al*., 2013).

### 2.4.2 Shoulder Surfing Attack

Shoulder surfing is a type of assault in which the attacker looks over the victim's shoulder to gather information such as personal identification numbers, passwords, and other sensitive information. When sharing personal information in a public area, a shoulder surfing attack can occur (Por *et al*., 2017). Shoulder surfing attacks are more vulnerable to text-based passwords than graphical-based passwords. Shoulder surfing can be divided into two categories.

The first sort of assault is when data is obtained through direct observation. When a person stares directly over the victim's shoulder while they are inputting data, such as their PIN at a checkout terminal, this is known as snooping (Li *et al*., 2005). The victim's behaviors are initially videotaped in the second type. Criminals can then go back and review these videos in depth to get the information they need (Lashkari *et al*., 2009).

### 2.4.3 Brute Force Attack

A Brute Force attack is a type of password guessing attack in which the attacker attempts every conceivable code, combination, or password until the correct one is found (Bosnjak *et al*., 2018; Dave, 2013). This type of assault could take a long time to complete. Brute-forcing a password with a complex password can take a long time. By trial and error, a brute force attack attempts to guess login information, encryption keys, or the location of a hidden web page (Fulkar *et al*., 2012). Hackers experiment with every possible combination in the hopes of making the correct guess. These attacks use 'brute force,' which means they try to 'force' their way into your personal accounts by employing overwhelming force (Vaithyasubramanian *et al*., 2014). The brute force attack on graphical passwords is more difficult than the brute force attack on text-based passwords. To duplicate human input in a brute force attack, an artificially generated correct mouse movement is required, which is particularly difficult in the case of recall-based graphical passwords (Fulkar *et al*., 2012).

### 2.4.4 Man-in-the-middle attack

A man-in-the-middle attack is a form of eavesdropping attack in which an attacker intercepts a conversation or data transfer in progress (Callegati *et al*., 2009). The attackers pose as both genuine parties after inserting themselves in the "middle" of the transfer. This allows an attacker to intercept data and information from both parties while also providing malicious links or other information to both genuine users in a way that may not be discovered until it is too late (Bhushan *et al*., 2018).

### 2.4.5 Social Engineering

An attacker may employ social engineering to persuade a person to give personal information such as their username, credit card number, and passcode, or to provide the hacker more access (Santwana, 2014). The following are some examples of social engineering attacks: To alter an employee's passcode, impersonate that staff at the IT Help Desk. This exploit is more prone to alphanumeric or text-based passwords than graphical passwords. Giving out graphical passwords over the phone, for example, is challenging. It would also take longer to set up a malicious website to get graphical passwords (Krombholz *et al*., 2015; Salahdine & Kaabouch, 2019).

### 2.4.6 Credential stuffing

Credential stuffing is a cyber-attack in which details obtained from one service's security breach are used to log in to another unrelated service (Li *et al*., 2019). For instance, an adversary may take a list of usernames and passwords acquired from a compromise at a major department store and try to get in to a national bank's website using the same login details.

The adversary is hoping that some of those department store clients also have a bank account there, and that they used the same usernames and passwords for both (Pal *et al*., 2019).

### 2.4.7 Dictionary Attack

A dictionary attack is an attempt of breaking into a passcode computer or server by methodically inputting each word from a dictionary as a password (Fulkar *et al*., 2012; Wang & Wang, 2015). A dictionary attack could also be used to find out what key is required to decrypt an encrypted messages or document (Bosnjak *et al*., 2018). Dictionary attacks against recognition-based graphical passwords will be impractical because they demand mouse input rather than keystrokes. On some recall-based graphical passwords, a dictionary attack is possible, but an automated dictionary attack will be much more difficult than a text-based dictionary attack. Dictionary attacks are more vulnerable to text-based passwords than graphical passwords (Bosnjak *et al*., 2018).

## 2.5 Related Works

Abhijith *et al*. (2021) recommended employing a graphical password authentication system to prevent shoulder surfing attacks. The proposed method used both textual and graphical passwords, eliminating the need for complex textual passwords that could be difficult for users to remember. Users can use any textual password with the graphical password in place. Nonetheless, the type of graphical password mechanism employed in this study was not specified. Furthermore, the proposed system's usefulness was not evaluated.

Vaddeti *et al*. (2020) suggested a graphical password authentication scheme based on the best existing features like distorted images, hash index, and loci metrics, as well as visual cryptosystems and additional naive features, to defend against well-known attacks like brute force, educated guessing, sniffing, hidden camera, shoulder surfing, and phishing. The paper's flaw is that no assessment metric was utilized to assess the system's performance.

Yang *et al*. (2017) developed a visual cryptography method that allows visual information to be encoded in a manner that can be decrypted simply by sight-reading. The encryption of the original image is derived into two images in this method of cryptography by changing every pixel into a structure that looks like gray or noise. The User ID was extracted from the server share using an optical character recognition method. As a result, a user's identity is verified by matching the retrieved and preserved IDs. Using optical character recognition raises the computational complexity of the procedures, which is a disadvantage.

The limits of graphical and alphanumeric passwords were identified by Chuen *et al*. (2020). One of the drawbacks of using a graphical password technique is the possibility of shoulder surfing. A graphical password could be witnessed physically, giving the attacker a clear view of the password being entered. Another disadvantage of a graphical password technique is that it is vulnerable to guessing. If the person only registered a short and regular password, the odds of it being guessable would increase, just like with an alphanumeric password. To conquer these potential drawbacks, a shoulder surfing-resistant method could be put in place, such as including numerous mouse cursors when users log in to their accounts, which would make it difficult for the intruder to determine which mouse cursors are valid and which click points the user has clicked.

Shnain and Shaheed (2018) employed a graphical password to improve E-commerce authentication problems. A modified Inkblot authentication mechanism was proposed in this work. Images are used as a trigger for text passcode entry in the Inkblot authenticator. Users are given the option of selecting a series of inkblots and typing in the first and last letter of the word/phrase that best represents the inkblot during password creation. The user's password is made up of these pairs of letters. The inkblot is a useful tool for users to create their login. The problem with this inkblot authentication method is that users are only given a limited number of password options.

Por *et al*. (2017) suggested a new technique based on digraph substitution rules to hide the procedure or activity necessary to generate password-images. In the suggested method, a user only needs to click on one of the pass-images in each challenge set for three consecutive sets, rather than both pass-images. While this activity is simple enough to reduce login time, the photos clicked appear to be random and can only be retrieved by knowing the registration password as well as the activity rules. As a result, shoulder-surfing attackers will be unable to get information about which password images and pass-images the user uses.

Togookhuu and Zhang, (2017) suggested a three-layer verification recall graphical password technique. The suggested recall-based authentication technique was an upgrade to the Pass-Go scheme, which included secret questions, answers, and backdrop imagery. The suggested system called Questions-Background Image-Pattern (QBP), is made up of three pieces that work together to ensure password security. The initial portion of the verification process is concerned with the secret question and the text-based answer. The second section is about selecting a picture based on recognition, and the third section is about constructing a password using a drawing that is easy to remember. The disadvantage of this method is that it is easy for people to forget their stroke order while using drawing to create a password.

Ahsan and Li (2017) suggested a graphical password authentication using an image sequence. In this manner, the user uploads photographs from his or her own directory for password selection, and the images supplied by one user are not visible to the other. The planned system is divided into four phases. The legitimate email address phase is the first step. The user will submit a genuine email address during registration, which will be used throughout the login phase. The system will redirect the visitor to the next page after inputting a valid email address, which will provide photographs for selection. The second stage is the picture selection phase, in which a user can choose between a limit of six and a minimum of four photographs to finish the registration process. A user will be asked to pick the number of photographs that were uploaded during the registration step after logging in. Users upload their chosen photographs based on the prior number of images picked in the third level. In the last phase, the picked photos are stored in order. When logging in, the user selects uploaded photographs in the same order as they were picked during the registration step. The user will not be able to login if the sequence of selected photos is incorrect. The proposed technique is vulnerable to shoulder surfing attack as an attack can easily obtain the image sequence during registration or login.

Mackie and Yıldırım (2018) Mark presents a hybrid authentication technique that combines text and graphical passwords based on recognition. Since adversaries do not know the users' image choices, this authentication technique can limit phishing assaults because even if users are duped into sharing their key passwords, there is still a possibility to save the entire password because adversaries do not know the users' image priorities. Aside from security, the proposed authentication technique improves memorability by eliminating the need for users to recall long and difficult passwords. To prevent others from looking over the user's shoulder while selecting the photos and inputting the relevant characters, the input is given through the keyboard rather than clicking on the images. However, because users must select images and enter the textual password via keystroke, this system is subject to keylogger attacks.

The use of fingerprint authentication was suggested by Chakraborty *et al.* (2019). The suggested method incorporates both biometrics and a password-based security scheme. More than one fingerprint can be stored by the user. The user can simply verify their identity using any of the fingerprints they gave throughout the registration process. The reading of fingerprints begins with the left hand's pinkie finger, which will be saved as "1," and progresses to the thumb, which will be kept as "5". The position of the thumb in the right hand will be "6," and so on until we reach the pinkie finger in the right hand, which will be "10." For this scheme, users must supply at least two finger fingerprints or all ten finger fingerprints. After the fingerprints have been submitted, the user is needed to input the fingerprints passcode, which must be entered twice for confirmation. One of the disadvantages of this authentication mechanism is that it is difficult to recall. If one of the fingers is used as a passcode, for example, once it is compromised, it can never be used again because changing a fingerprint is nearly hard, so it is compromised permanently.

Patel *et al*. (2021) suggested a color login solution for graphical password authentication. The method is divided into two stages: enrollment and login. The enrollment phase includes an eight-color selection section as well as a textual password element. Only 16 characters are allowed in the password: 8 alphabetic characters (a-f), and 8 numeric digits (1-8) For added protection, the graphical password is coupled with the textual password. However, one disadvantage of this strategy is that the enrolment step must take place in an atmosphere free of shoulder surfing because it is subject to shoulder-surfing attacks. The login time, login success rate, and vulnerabilities to keylogger and guessing attacks were all measured.

A hybrid solution for graphical password authentication was presented by (Sepideh, 2019) to prevent shoulder surfing, smearing, and brute force assaults. This technique combines two types of graphical passwords: recognition-based and cued recall-based graphical passwords. To begin, a 5 by 5 grid with 25 cells is presented, each containing 25 randomly selected photographs. The user is given three photos from which to choose. The password is made up of these three photos. Following the selection of the three photos, the user is presented with a new 5 by 5 grids on which to draw a pattern that will be used during the login step. Because the password is entered with a mouse click, this method is vulnerable to a mouse tracking attack.

| **S/No** | **Author/year** | **Title of work** | **Strength** | **Weakness** |
|---|---|---|---|---|
| 1. | Chiasson *et al*., 2012 | Knowledged base authentication | Easy to remember | Shoulder surfing, brute force |
| 2. | Peisert *et al*., 2013 | Knowledged base and Graphical based authentication | Double authentication method | Shoulder surfing, keylogger attack |
| 3. | Ometov *et al*., 2018 | Biometric based authentication | Requires individuals' presence or something owned | Can be manipulated |

Figure 2.6: Tabular expressions of literatures reviewed.

# CHAPTER THREE

# 3.0 RESEARCH METHODOLOGY

## 3.1 Analysis of Existing System

The primary goal of the system analysis is to examine the requirements of the cascade multi-stage-based password system while taking into account what the present authentication system does, its faults, and potential solutions. Text-based password authentication currently employs alphanumeric to generate passwords. Users always create passwords that are easy to remember, but attackers can easily crack these passwords. Users utilize strong system-assigned passwords that are typically difficult to remember for increased security.

Existing graphical-based authentication systems are vulnerable to close range or far range (use of camera) shoulder surfing attack.

### 3.1.1 Limitations of the Existing Systems

The Limitations of the existing systems are:

1. The existing text-based password system is vulnerable to brute force attack.
2. Some existing graphical systems are vulnerable to shoulder surfing attack as the entire image and drawing is visible on the screen as it is being entered, and thus an attacker need accurately observe or record only one login for the entire password to be revealed.
3. Existing text-based system is vulnerable to dictionary attack.
4. Text-based passwords must be easy to remember while yet being difficult to guess. If a textual password is difficult to guess, it is also difficult to remember.
5. This single level security of system makes the existing system easy to compromise by attacks.

## 3.2 Justification for the Proposed System

The new system is designed to solve the problems affecting the existing textual and graphical password system in use. This system was designed to improve password usability and security. Authentication reliability is improved as One-time password (OTP), graphical and textual passwords are implemented. Firstly, a user is authenticated using OTP which only the user can have access to. Secondly, the user is authenticated with textual password and lastly the user is authenticated with recall-based graphical password. In addition, the user do not have to create long and difficult textual and graphical passwords, given that the proposed system makes use of OTP as an additional layer of security.

### 3.2.1 Advantages of the New System

The advantages of the new system are:

1. More human friendly password system.
2. Integration of OTP, textual and graphical password makes security of the new system very high.
3. Dictionary attack is infeasible.
4. It's impossible to launch a guessing attack, because OTP can only be used once.
5. The system is not vulnerable to shoulder surfing attack with the Implementation of OTP as most graphical password are vulnerable to shoulder surfing attack.
6. Key-logging attack is infeasible.

## 3.3 Design of the New System

The important stage of development is the main focus of this design phase. The users or stakeholders involved, as well as the complexity of their occupations, were considered; it's more of a network relational model that identifies relationships and links between one worker's job and another's.

### 3.3.1 Sequence Diagram

A sequence diagram depicts item interactions in chronological order. The sequence diagram shows the objects and classes involved in the scenario, as well as the sequence of messages sent between them to carry out the scenario's functionality (Sarma *et al.*, 2007). Figures 3.1 and 3.2 show the sequence diagram for the proposed Registration and login module.

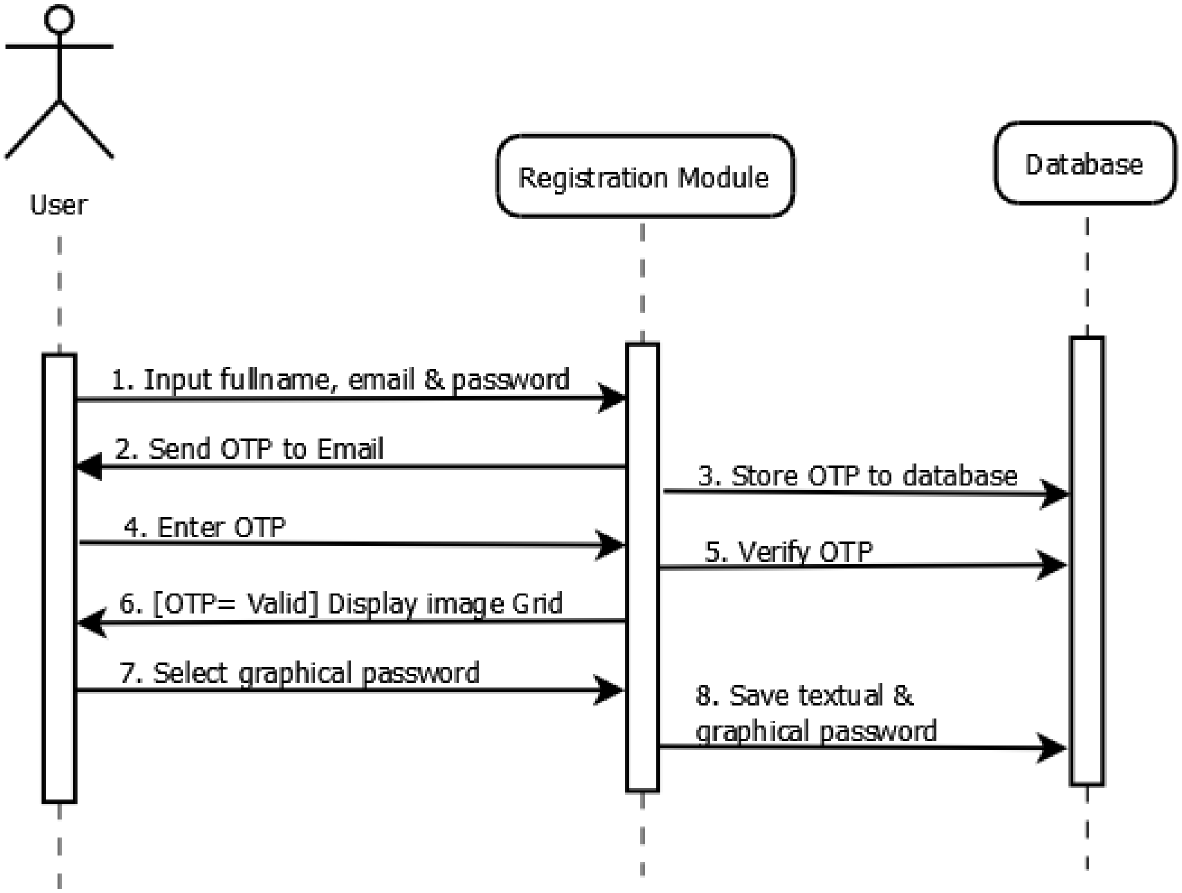

User
Registration Module
Database
1. Input fullname, email & password
2. Send OTP to Email
3. Store OTP to database
4. Enter OTP
5. Verify OTP
6. [OTP= Valid] Display image Grid
7. Select graphical password
8. Save textual & graphical password

Figure 3. 1 Sequence Diagram for Registration Stage

In figure 3.1 the first step of registration is to enter user’s full name, email, and password and the second step is that the system sends an OTP to your previously supplied email and then the user is asked to enter the OTP sent to their email address. After entering the correct OTP, the user is given access to select the graphical password. The user is allowed to select three different graphical passwords in sequence and lastly in the last step the user selected graphical password and supplied textual password is stored in the database.

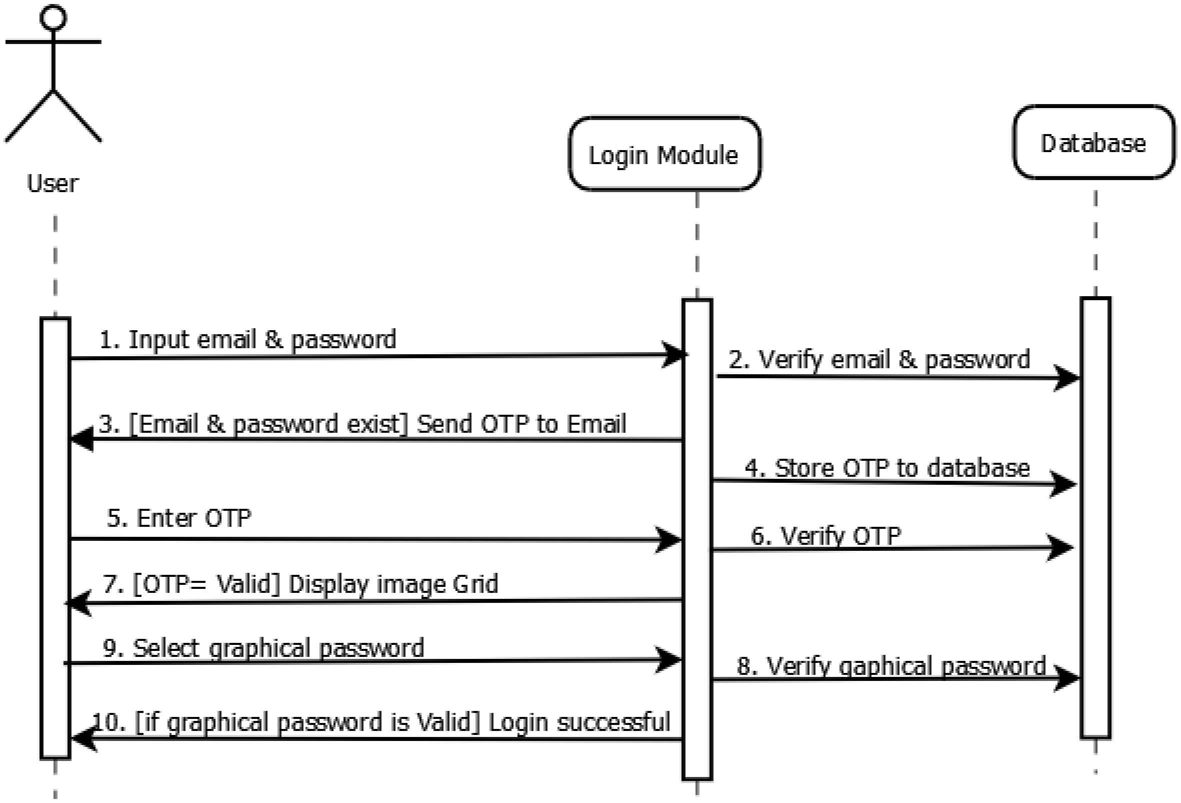

User
Login Module
Database
1. Input email & password
2. Verify email & password
3. [Email & password exist] Send OTP to Email
4. Store OTP to database
5. Enter OTP
6. Verify OTP
7. [OTP= Valid] Display image Grid
9. Select graphical password
8. Verify gaphical password
10. [if graphical password is Valid] Login successful

Figure 3. 2 Sequence Diagram for Login Module

The sequence activity for the login module is represented in figure 3.2. The login procedure begins when the user enters their email address and password, which grants them access to the next stage of the OTP authentication process. After the OTP has been verified, the user is prompted to choose their registered graphical password. The verification of the chosen graphical password is the final stage in the login process.

### 3.3.2 Database Design

After a thorough investigation of the entities and their relationships, the database was constructed with a total of two tables. The schema diagram in Figure 3.3 depicts a summary of the entities and their interrelationships.

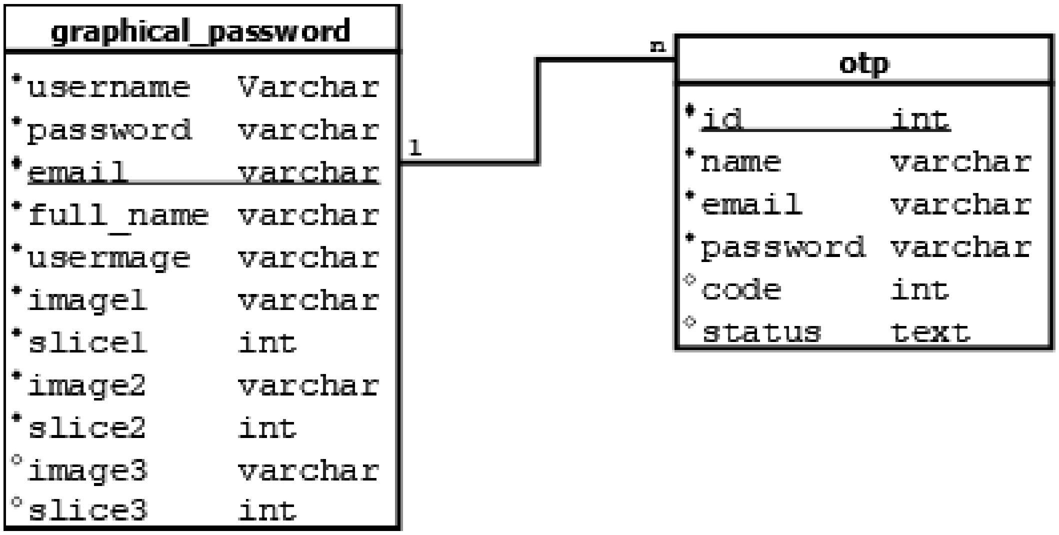


Figure 3. 3 Database Design for the Proposed System

The proposed system database design in figure 3.6 consist of just two tables, which are: graphical_password and otp table. The graphical_password table is used to save the graphical passwords, textual password, and the form values. While the otp table is used to save the generated OTP with name, email and password as foreign keys linking it to the graphical_password table. The two tables have a one-to-many relationship because a single graphical password can be related to several OTPs.

## 3.4 Proposed Method

The proposed system consists of mainly of three authentication techniques in sequential order: textual, one-time password and graphical password. The architectural diagram of the proposed system is given in figure 3.4.

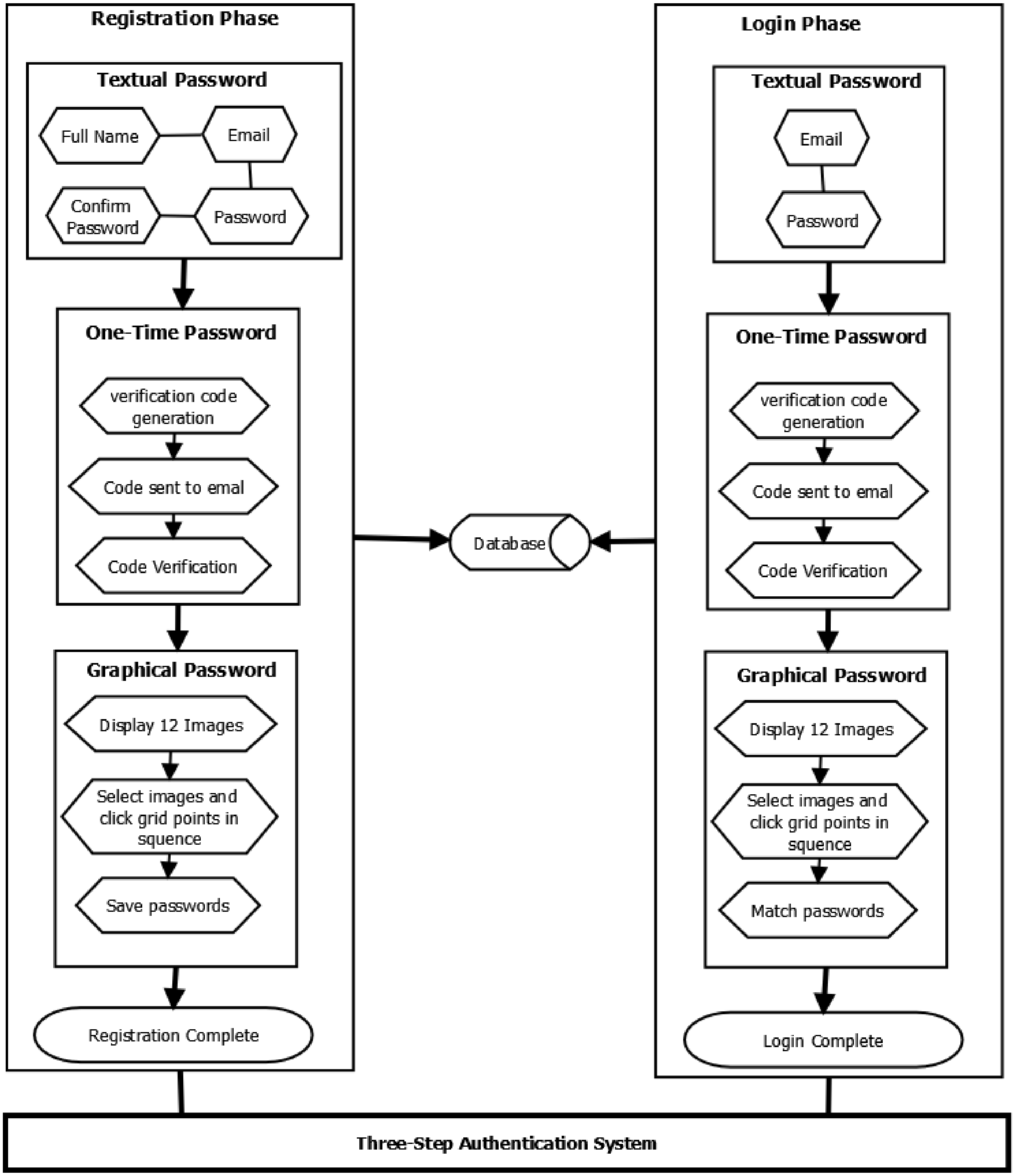


Figure 3. 4 Proposed System Architecture

Figure 3.4 depicts the process function, system database, and external entities that will interact with the system. Figure 3.4 demonstrates a two-phase authentication system:

registration and login. The registration phase includes email verification, alphanumeric password registration, and graphical password registration. The login stage includes OTP, textual password verification, and graphical password validation.

### 3.4.1 Registration phase

The registration phase consists of three main processes: textual password, One-Time Password (OTP) and graphical password implementation. A flowchart of the registration phase is presented is figure 3.5.

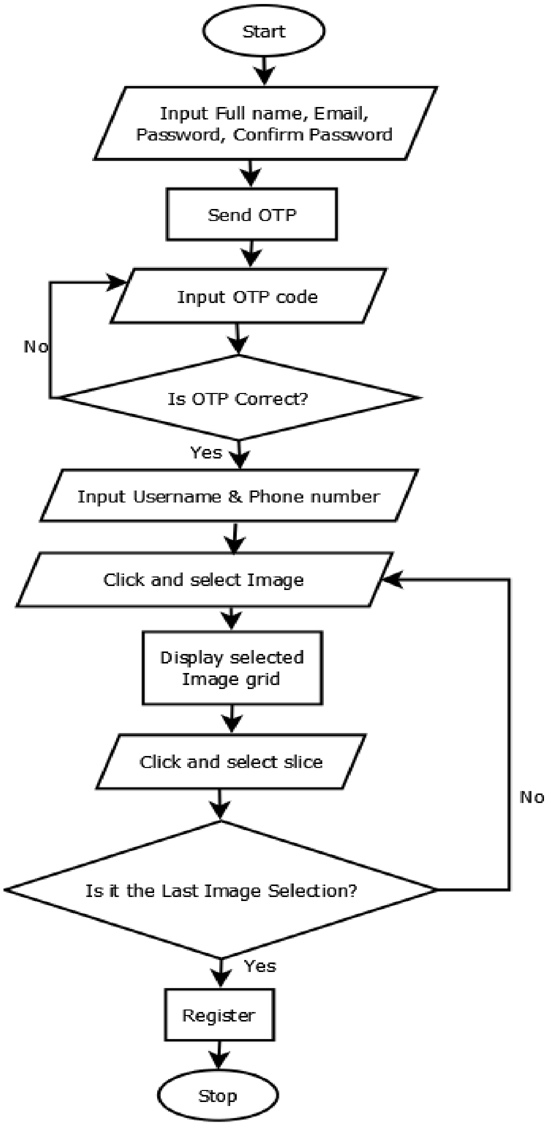
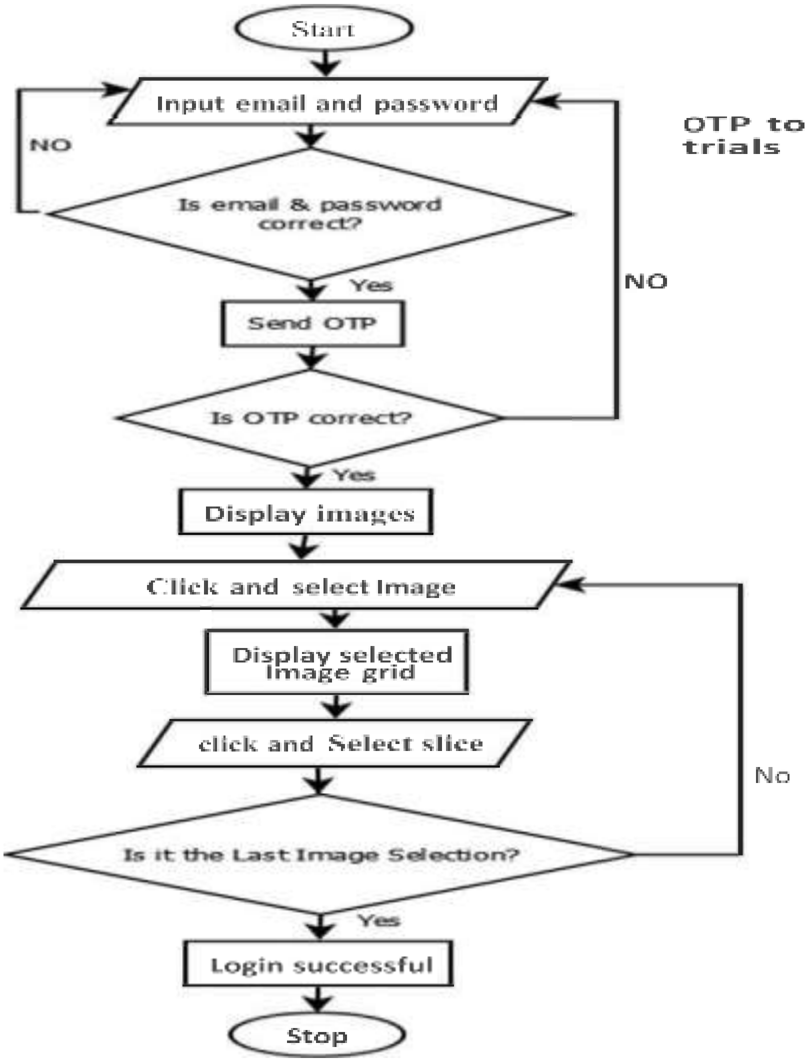

Start
Input Full name, Email, Password, Confirm Password
Send OTP
Input OTP code
No
Is OTP Correct?
Yes
Input Username & Phone number
Click and select Image
Display selected Image grid
Click and select slice
No
Is it the Last Image Selection?
Yes
Register
Stop

Figure 3. 5 Registration Flowchart

- **Process 1: Textual Password Registration-** In this phase the user is asked to input their email, full name, password and confirm password.
- **Process 2: OTP Authentication-** The OTP authentication phase deals with the generation of OTP by the system. This generated OTP is sent the inputted email from the textual password registration phase. Then, the user is asked to input the OTP for verification. If the OTP is wrong user is denied access to the next phase, otherwise the user is granted access to the graphical password phase.
- **Process 3: Graphical Password implementation-** A 2 × 2-image grid is now displayed to the user from which the user clicks on one point of the image. After which, the user is to select another image and click on the generated 2 × 2-image grid.

### 3.4.2 Login Phase

Figure 3.6 depicts the login process. The processes involved in the login phase are detailed further down.



OTP to be blocked after trials

Figure 3. 6 Login Phase

- **Process 1: Textual Password Authentication:** During the login phase, the user registered password and email must be submitted which is compared with the email and password stored in the database. If email and password match, then the user is allowed to move to the next step.
- **Process 2: OTP Authentication:** in this step the user is asked to supply the OTP that was generated and sent to their registered email address. If a wrong OTP is supplied then, access is denied. However, if the OTP is correct the user is given access to the next authentication process.
- **Step 3: Graphical Password Authentication -** After authenticating the OTP dose, twelve images are displayed. The user is prompted to select one of the displayed images. On click of an image a 2 x 2 grid containing parts of the selected image is displayed user is expected to click on the grid in image for successful authentication. If the first attempt fails, the user is asked to login from the beginning.

## 3.5 Textual Password Authentication

A textual password is a piece of encrypted data, usually a string of letters, numerals, or other symbols, that is used to verify a user's identity. Passwords were once expected to be memorized, however given the enormous number of password-protected services that the average person uses, memorizing unique passwords for each site is impracticable (Mackie & Yıldırım, 2018b).

Textual passwords are the most often used authentication passwords. Users tend to choose passwords with short lengths that are easier to remember. Textual passwords, on the other hand, are subject to a variety of attacks, including shoulder-surfing, brute-force attacks, hidden camera attacks, and spyware attacks (Shen *et al*., 2017).

## 3.6 One-Time Password (OTP)

A one-time password (OTP) is a number or alphanumeric string of characters that is created automatically and used to authenticate a user for a single transaction or login session. A user-created password, especially one that is weak and/or reused across several accounts, is less secure than an OTP. OTPs can be used in place of or in addition to authentication login information to add an extra degree of protection.

OTP generation methods frequently employ pseudo-randomness or randomness, as well as cryptographic hash functions, which may be used to extract a value but are difficult to reverse, making it hard for an attacker to gain access to the data used for the hash. This is crucial since it would otherwise be possible to predict future OTPs simply by looking at recent ones. Password replay attacks are prevented by the pseudo-random value's unpredictable and unique nature.

## 3.7 Recall-Based Graphical Password (Cued Click Point)

In this strategy, the system provides certain tips that assist users in accurately reproducing their passwords. Hot spots (regions) within an image will be used to provide these hints (Panduranga Rao, 2013). The user must choose one of these regions to register as a password, and they must choose the same region in the same order to log into the system. This study used Cued Click Points (CCP), a recall-based strategy for user authentication. Cued Click Points (CCP) have been proposed as an alternative to Pass-Points (Islam *et al*., 2020). In CCP, users click one point on each image rather than on many points on one image. It offers cued-recall and introduces visual cues that instantly alert valid users if they have made a mistake when entering their latest click-point. It also makes attacks based on hotspot analysis more challenging (Moraskar *et al*., 2014).

A wrong click leads down the wrong path, with identification failure being explicitly indicated only after the final click. Users can only choose their images to the extent that the next image is dictated by their click-point. If they don't like the images that come up, they can make a new password with different click-points to achieve other results. CCP works in the same way as Pass-Points when it comes to implementation. A discretization approach is utilized to identify a click-tolerance point's square and associated grid during password formation. This grid is collected for each click-point in a future login attempt and used to determine whether the click-point falls within the tolerance of the originating point (Moraskar *et al*., 2014). A usability improvement of CCP is that being cued to recall one point on each of the three images appears easier than remembering an ordered sequence of three points on one image.

## 3.8 Implementation Language and Tools used

The materials and tools used to implement this new system are:

### 3.8.1 PHP (Hypertext Preprocessing)

PHP is a popular general-purpose scripting language with a focus on web development (Prokofyeva & Boltunova, 2017). PHP is an HTML-enabled server-side programming language. It's used to manage dynamic content, databases, and session monitoring, as well as to create full e-commerce websites. MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server are just a few of the databases it supports (Lamsal, 2020).

### 3.8.2 HTML (Hyper Text Markup Language)

HTML is a markup language that is used to create web pages. The structure of a Web page is described in HTML. HTML is a client-side markup scripting format that allows web browsers to render interactive pages for users (Peroni *et al*., 2017).

### 3.8.3 CSS (Cascading Style Sheets)

CSS is the stylesheet language for HTML documents. CSS specifies how HTML elements should appear on a screen, in print, or in other media. CSS is a client-side scripting syntax for making websites more attractive, fashionable, and interactive (Ndia *et al*., 2019).

### 3.8.4 Xamp Server:

XAMPP is a free and open-source cross-platform web server solution stack bundle built by Apache Friends, which consists mostly of the Apache HTTP Server, MariaDB database, and interpreters for PHP and Perl scripts. This application server was utilized to host the web application in this investigation. It was also used to create the application database and all of its tables, which house all of the data handled by the system.

### 3.8.5 JavaScript:

JavaScript is a full-featured dynamic programming language that may be used to make a website more interactive. JavaScript is largely used on the client side to enhance user interfaces and dynamic webpages. It is implemented as part of a web browser. This offers access to computational objects within a host environment via a programmatic interface (Antal & Heged, 2018).

### 3.8.6 Web browser

A web browser is application software for accessing the World Wide Web. When a user requests a web page from a particular website, the web browser retrieves the necessary content from a web server and then displays the page on the user's device (Gnana & Kamalanaban, 2016). Example of web browsers used are UC browser, Mozilla Firefox, Safari and Google Chrome.

## 3.7 Evaluation Metrics

### 3.7.1 Unit Testing

In unit testing each unit of the program were tested to ensure that the program performs its functions as defined in the program specification. A unit is a single testable part of a software system. The aim of unit testing is to validate unit components with its performance (Anwar & Kar, 2019).

### 3.7.2 Usability testing

Usability testing refers to evaluating a software by testing it with representative users. This was done by the users to check that the system meets its supposed requirements(Elsafi *et al*., 2015). Under the usability metric the login success rate and login time is evaluated. Login time is the time taken to login using a particular authentication system.

### 3.7.3 Security Analysis

1. **Hidden Camera attack:** occurs when cameras hidden to capture the authentication process of a user (Gokhale & Waghmare, 2016; Sepideh, 2019).
2. **Shoulder surfing**: occurs when someone watches over your shoulder to nab valuable information such as your password, ATM PIN, or credit card number, as you key it into an electronic device (Lashkari *et al*., 2009; Shah *et al*., 2015).
3. **Guessing:** Guessing is a notion in which an attacker guesses a password based on the user's proclivity. Typically, users select passwords such as birthdates, pet names, and account numbers. It has become guessable, and an attacker may be able to compromise the system (Ma *et al*., 2020; Noroozi & Eslami, 2019).

4. **Keylogging**: Keylogger is a type of malicious software that records keystrokes on a keyboard without the user's awareness (M. K. Shah *et al*., 2020; Shinde & Wanaskar, 2016).

# CHAPTER FOUR

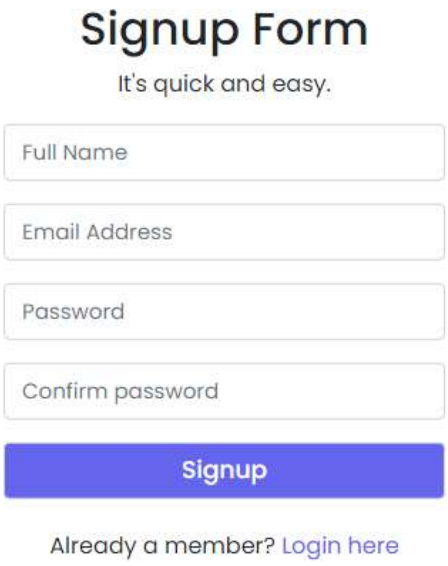
# 4.0 RESULTS AND DISCUSSION

## 4.1 Introduction

This chapter provides the proposed system implementation with screenshots for the registration and login process. The system provides easy to use graphics user interface. It also presents all the experiments conducted to evaluate the proposed system and results of the evaluation obtained from the research.

## 4.2 Textual Password Authentication Stage

The first step of authentication, that is textual password is shown in figure 4.1 and 4.2. Figure 4.1 is the signup page where the user registers their full name, email address and password. Figure 4.2 is the first login page where the user inputs their registered email and password. On clicking on the login button, the supplied email and password is verified with the ones stored in the database.

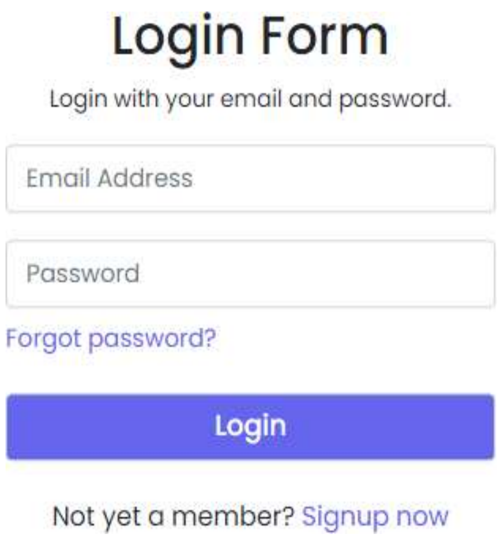


Signup Form

It's quick and easy.

Already a member? Login here

Figure 4. 1 Textual Password Registration Page



Login Form

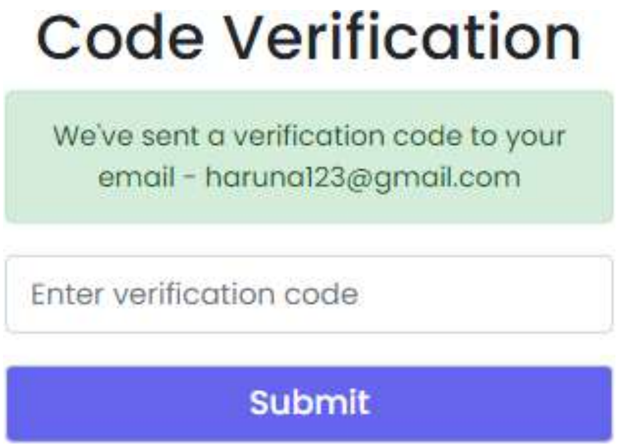
Login with your email and password.

Not yet a member? Signup now

Figure 4. 2 Textual Password Login Page

### 4.3 One-Time Password (OTP) Authentication Stage

The second step which is OTP authentication is presented in figure 4.3. The user is required to input the OTP code sent to their registered email. If the OTP code matches the sent OTP, then the user is allowed access to the last authentication phase displayed in figure 4.4.



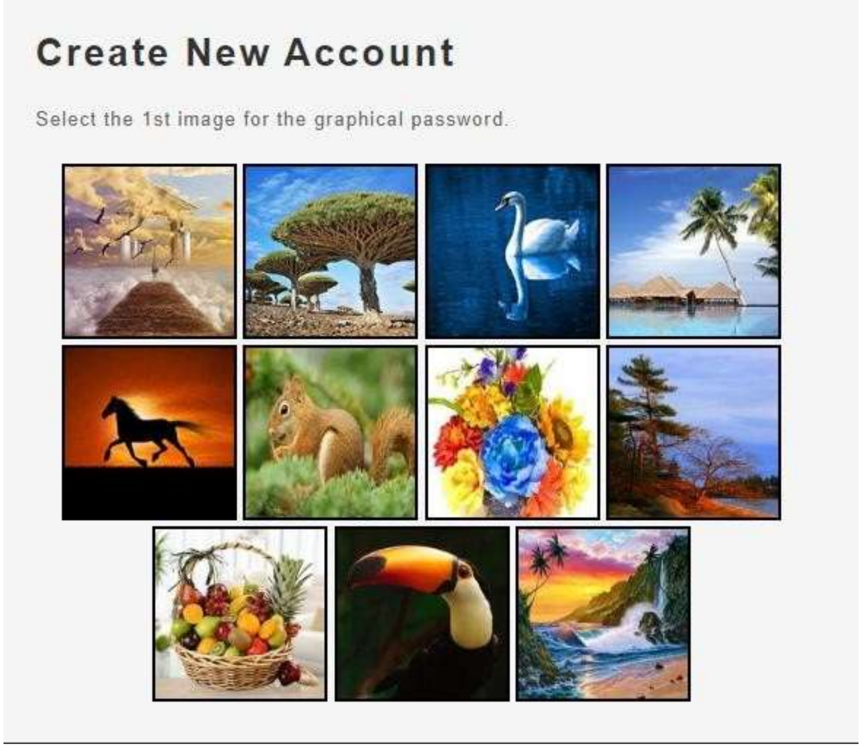
Code Verification

We've sent a verification code to your email - haruna123@gmail.com

Figure 4. 3 OTP Verification Page

## 4.4 Graphical Password Authentication Stage

After the OTP authentication stage, the user is given access to the last authentication stage, which is the graphical password phase. Figure 4.4 is the graphical password authentication page, which displays 11 images for users to choose from. After selecting an image, that image is then divided into four parts as shown in figure 4.5.

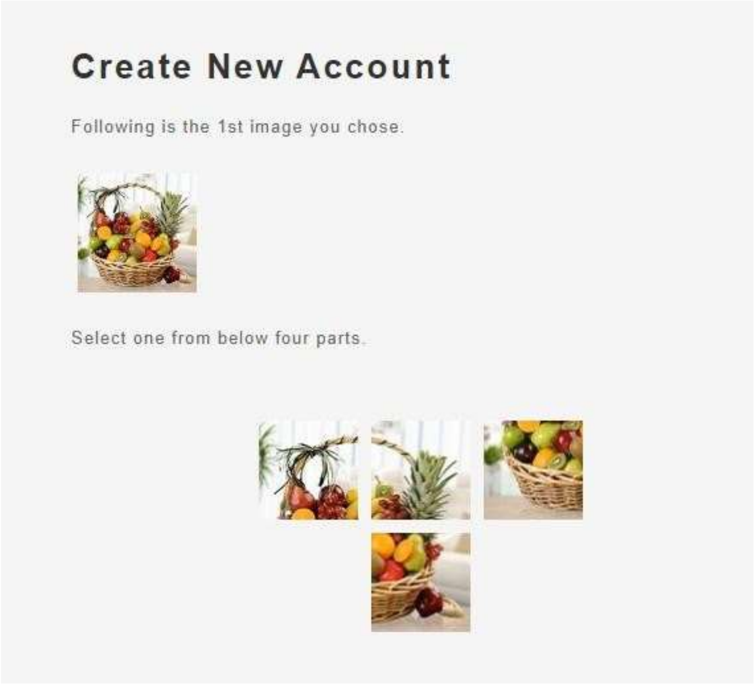


Create New Account

Select the 1st image for the graphical password.

Figure 4. 4 Graphical Password Page

Figure 4.5 shows four sub-images of the selected image. The user is required to select one of these four sub-images. After clicking on one of the sub-images, the user is asked to select another image from the eleven initial images. The second selected image is then divided into four sub-images and the user is prompted to select from these sub-images.



Create New Account

Following is the 1st image you chose.

Select one from below four parts.

Figure 4. 5 Grid of selected image

During the registration process the selected images and sub-images are stored in the database, while during login phase the stored images are used to validate the selected images for login.

## 4.5 System Evaluation

Three different forms of software testing were used in this study. Unit testing, usability testing, and security analysis are examples of these tests. In terms of functionality and usability, the system was put to the test by a variety of people. The developer performed unit and system testing to confirm that all the program's features and components are fully functional. Thirty people tried out the system. Tables 4.1 provide the results of the system's unit testing, which show that it meets its requirements.

### 4.5.1 Unit Testing

Unit testing deals with testing each software unit to ensure that it performed the functions specified in the program specification. The unit testing result is presented in table 4.1.

**Table 4. 1 Unit Testing**

| SECTION | INPUT | EXPECTED RESULT | ACTUAL RESULT | COMMENT |
|---|---|---|---|---|
| REGISTRATION | Fill Sign up form | Allows user to input their details | Correct | Passed |
| | Generate and Send OTP | System generating and sending OTP to Users email address | Correct | Passed |
| | Enter OTP | Allows user to input OTP | Correct | Passed |
| | Authenticate OTP | System verifies the OTP | Correct | Passed |
| | Click on image to register as password | Allows user click on image to register as password | Correct | Passed |
| | Save password | System sends user details and passwords to database | Correct | Passed |
| LOGIN | Enter email address and password | Allows user to input their email and password | Correct | Passed |
| | Authenticate username and password | System verifies the username and password | Correct | Passed |
| | Generate and Send OTP | System generating and sending OTP to Users email address | Correct | Passed |
| | Enter OTP | Allows user to input OTP | Correct | Passed |
| | Authenticate OTP | System verifies the OTP | Correct | Passed |
| | Select images in sequence | Allow users select images in sequence | Correct | Passed |
| | Authenticate select images | System authenticates the graphical password | Correct | Passed |

### 4.5.2 Usability Testing and Security Analysis

The degree to which a product allows individual users to fulfill their specified goals efficiently, effectively, and satisfactorily in the given context is referred to as usability. When developing a good graphical password strategy that meets the demands and requirements of its users, usability is a crucial thing to consider. This section defines and describes the primary usability aspects utilized in graphical passwords. These characteristics of usability are discussed in further depth farther down.

1. **Easy to remember:** This means that the system should allow for easy-to-remember passwords.

2. **Easy to Use:** This refers to the system's ability to provide a good platform for password creation.

3. **Easy to Create:** Means users can create their graphical passwords easily when the registration steps are simple.

4. **Easily Executed:** When the registration and login process is broken down into basic steps, people can easily perform the algorithm.

5. **Nice and Simple Interface**: Apart from making the interface appealing, it focuses on the user's interactions. The goal of a nice and simple interface is to make user interactions as efficient and straightforward as possible.

6. **Creation Time:** What is the average time it takes for a user to complete the registration process?

7. **Login Time**: What is the average time it takes for a user to complete the login process?

8. **Login Succes Rate**: the percentage of users who were successful in completing the login task.

The system’s usability testing based on the eight defined features and security analysis based on four common attacks are presented in table 4.2.

**Table 4. 2 Usability Testing and System Analysis**

| | Attributes | Shnain and Shaheed (2018) | Mackie and Yıldırım (2018a) | Proposed System |
|---|---|---|---|---|
| **Security Analysis** | Prone to hidden camera attacks | Yes | No | Yes |
| | Prone to shoulder surfing attacks | Yes | No | Yes |
| | Prone to guessing attacks | No | No | Yes |
| | Prone to key-logger attacks | No | Yes | Yes |
| **Usability features** | Easy to remember | Yes | No | Yes |
| | Easy to Use | Yes | Yes | Yes |
| | Easy to Create | Yes | No | Yes |
| | Easily Executed | Yes | Yes | Yes |
| | Nice and Simple Interface | Yes | Yes | Yes |
| | Creation Time | - | 94.08 Seconds | 111 Seconds |
| | Login Time | - | 57.40 seconds | 82 Seconds |
| | Login Success rate | - | 90.38% | 90% |

The suggested system is immune to hidden camera, shoulder surfing, guessing, and keylogger attacks, as shown in Table 4.2, whereas Shnain and Shaheed (2018) is vulnerable to hidden camera, shoulder surfing, guessing, and keylogger assaults. Hidden camera, shoulder surfing, and guessing are all immune to Mackie and Yıldırım (2018a), but keylogger attack is vulnerable. According to the usability features, the suggested system takes longer to register and login than Mackie and Yıldırım (2018a). The proposed solution is more user-friendly and secure than earlier authentication methods.

## 4.7 Users Feedback

Based on an interview with some of the users, the feedback summary is shown in Table 4.3.

**Table 4. 3 Feedback Summary**

| S/N | QUESTIONS | FEEDBACKS |
|---|---|---|
| 1 | How user-friendly is the system? | According to the respondents when fully understood, the method is very simple to use. |
| 2 | What are your thoughts on the user interface and layout? | The system's design is simple and efficient, according to all respondents. |
| 3 | Please score the software interfaces' performance. | All the respondents confirmed that the loading time is sufficient and still reasonable. |
| 4 | What are your thoughts on the system's content? | According to the respondents, the system's content covered the basics of authentication operations. |
| 5 | Please offer your opinion on the system's overall appearance. | The overall framework was scored as good by all respondents. |
| 6 | Do you think this system will increase the efficiency of information security? | All the respondents agreed that the system would significantly increase information security performance. |

# CHAPTER FIVE

## 5.0 CONCLUSION AND RCOMMENDATION

In this study the OTP, textual and cued click point recall-based graphical password techniques were used to perform user's authentication for access to web application. The user authentication system consists of the registration and login phase. The registration phase captures the user's graphical password in sequence, textual password and validate the user's email using OTP. The login phase gives user access to a web application by verifying the authenticity of the user via the submitted OTP, username, textual password and the submitted sequence of the graphical password. The combination of OTP, textual and graphical password made the proposed system immune to shoulder surfing attack, guessing, dictionary attack and key-logger attack.

### 5.1 Conclusion

A method for authentication of users for web application was proposed based on OTP, textual and cued click point recall-based graphical password techniques. In this research, authentication using these combined techniques achieved a stronger and reliably security than the existing textual and graphical password systems which are vulnerable to shoulder surfing attack. The first objective of identifying the problems in existing authentication techniques was achieved in chapter two. The identified issues include vulnerability to shoulder surfing attack, keylogger attack, and guessing attack.

The second objective of developing a secure cascade multi-stage authentication system was achieved using the combination of OTP, textual password and cued click point recall-based graphical password technique. For simplicity in this technique the user is allowed to select up to three images and click on one of the grids of each of the selected images.

Lastly the performance of the muti-stage authentication system was validated using unit, testing, usability testing, and security analysis. This system validation is an accomplishment of the third objective. From the security analysis the system is secure against shoulder surfing, guessing, hidden camera and keylogger attack. In conclusion a reliable and robust authentication system has been achieved in this study.

## 5.2 Recommendations

The proposed system can be used with any web application for access control. However, the study made use of the cued click point recall-based graphical password technique for authentication. For future work other graphical password methods such as recognition-based authentication can be used in combination with text, and OTP password.

## 5.3 Contribution to Knowledge

Improved the security of web applications through the development of a cascade muti-stage authentication system which is immune to several common security attacks. Secures the environment, the people in it and the online site they are using without requiring cumbersome resets or complicated policies. It will also secure identity theft via stolen password online.

# REFERENCES

Abhijith, S., Sam, S., Sreelekshmi K U, & Samjeevan, T. T. (2021). Web based Graphical Password Authentication System. *International Journal of Engineering Research & Technology* , *9*(7), 29–32. www.ijert.org

Afandi, R. R., & Jali, M. Z. (2017). ChoCD: Usable and Secure Graphical Password Authentication Scheme. *Indian Journal of Science and Technology*, *10*(4), 4–9. https://doi.org/10.17485/ijst/2017/v10i4/110885

Ahsan, M., & Li, Y. (2017). *Graphical Password Authentication using Images Sequence*. *December*, 1824–1832. https://www.irjet.net/archives/V4/i11/IRJET-V4I11330.pdf

Alt, F., Mikusz, M., Schneegass, S., & Bulling, A. (2016). Memorability of cued-recall graphical passwords with saliency masks. *ACM International Conference Proceeding Series*, 191–200. https://doi.org/10.1145/3012709.3012730

Antal, G., & Heged, P. (2018). Static JavaScript Call Graphs : a Comparative Study. *Journal of Advances in Information Technology*, *7*(1). https://doi.org/10.1109/SCAM.2018.00028

Anwar, N., & Kar, S. (2019). Review Paper on Various Software Testing Techniques & Strategies. *Global Journal of Computer Science and Technology: C Software & Data Engineering*, *19*(2).

Awasthi, A. K., & Srivastava, K. (2013). A biometric authentication scheme for telecare medicine information systems with nonce. *J. Med. Syst.*, *37*(5), 1–4. https://doi.org/10.1007/s10916-013-9964-1

Bhanushali, A., Mange, B., Vyas, H., Bhanushali, H., & Bhogle, P. (2015). Comparison of Graphical Password Authentication Techniques. *International Journal of Computer Applications*, *116*(1), 11–14. https://doi.org/10.5120/20299-2332

Bhushan, B., Sahoo, G., & Rai, A. K. (2018). Man-in-the-middle attack in wireless and computer networking - A review. *Proceedings - 2017 3rd International Conference on Advances in Computing, Communication and Automation (Fall), ICACCA 2017*, *2018-January*, 1–6. https://doi.org/10.1109/ICACCAF.2017.8344724

Bindu, C. S. (2015). Click based Graphical CAPTCHA to thwart spyware attack. *Souvenir of the 2015 IEEE International Advance Computing Conference, IACC 2015*, 324–328. https://doi.org/10.1109/IADCC.2015.7154723

Bosnjak, L., Sres, J., & Brumen, B. (2018). Brute-force and dictionary attack on hashed real-world passwords. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings, May 2018*, 1161–1166. https://doi.org/10.23919/MIPRO.2018.8400211

Callegati, F., Cerroni, W., & Ramilli, M. (2009). Man-in-the-middle attack to the HTTPS protocol. *IEEE Security and Privacy*, *7*(1), 78–81. https://doi.org/10.1109/MSP.2009.12

Chakraborty, A., Pathan, S., Kabir, M., & Thakur, K. (2019). Fingerprint Authentication Security: An Improved 2-Step Authentication Method with Flexibility. *International Journal of Scientific and Engineering Research*, *10*(February), 438–442.

Chiasson, S., Stobert, E., Forget, A., Biddle, R., & van Oorschot, P. C. (2012). Persuasive cued click-points: Design, implementation, and evaluation of a knowledge-based authentication mechanism. *IEEE Transactions on Dependable and Secure Computing*, *9*(2), 222–235. https://doi.org/10.1109/TDSC.2011.55

Chuen, Y. S., Al-Rashdan, M., & Al-Maatouk, Q. (2020). Graphical password strategy. *Journal of Critical Reviews*, *7*(3), 102–104. https://doi.org/10.31838/jcr.07.03.19

Das, A. K., Sharma, P., Chatterjee, S., & Sing, J. K. (2012). A dynamic password-based user authentication scheme for hierarchical wireless sensor networks. *Journal of Network and Computer Applications*, *35*(5), 1646–1656. https://doi.org/10.1016/J.JNCA.2012.03.011

Dave, K. T. (2013). Brute-Force Attack “Seeking but Distressing.” *Internation Journal of Innovation in Engineering and Technology (IJIET)*, *2*(3), 75–77.

Elsafi, A., Jawawi, D. N. A., Abdelmaboud, A., & Ali, A. (2015). A comparative evaluation of state-of-the-art integration testing techniques of component-based software. *Journal of Theoretical and Applied Information Technology*, *71*(2), 257–267.

Ethelbert, O., Moghaddam, F. F., Wieder, P., & Yahyapour, R. (2017). A JSON token-based authentication and access management schema for cloud SaaS applications. *Proceedings - 2017 IEEE 5th International Conference on Future Internet of Things and Cloud, FiCloud 2017*, *2017-January*, 47–53. https://doi.org/10.1109/FICLOUD.2017.29

Fulkar, A., Sawla, S., Khan, Z., & Solanki, S. (2012). a Study of Graphical Passwords and Various Graphical Password Authentication Schemes. *World Research Journal of Human-Computer Interaction ISSN: 2278-8476*, *1*(1), 4–8.

Gnana Sambandam, K., & Kamalanaban, E. (2016). A Systematic Review of Security Measures for Web Browser Extension Vulnerabilities. *Proceedings of the International Conference on Soft Computing Systems*, *398*(February), 319–329. https://doi.org/10.1007/978-81-322-2674-1

Gokhale, M. A. S., & Waghmare, V. S. (2016). The Shoulder Surfing Resistant Graphical Password Authentication Technique. *Procedia Computer Science*, *79*, 490–498. https://doi.org/10.1016/j.procs.2016.03.063

Goldberg, J., Hagman, J., & Sazawal, V. (2002). Doodling our way to better authentication. *Conference on Human Factors in Computing Systems - Proceedings*, *January 2002*, 868–869. https://doi.org/10.1145/506443.506639

Hafiz, M. D., Abdullah, A. H., Ithnin, N., & Mammi, H. K. (2008). Towards identifying usability and security features of graphical password in knowledge based authentication technique.

*Proceedings - 2nd Asia International Conference on Modelling and Simulation, AMS 2008*, 396–403. https://doi.org/10.1109/AMS.2008.136

Haque, A., & Imam, B. (2014). A New Graphical Password : Combination of Recall & Recognition Based Approach. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, *8*(2), 310–314.

Ibrahim, T. M., Abdulhamid, S. M., Alarood, A. A., Chiroma, H., Al-garadi, M. A., Rana, N., Muhammad, A. N., Abubakar, A., Haruna, K., & Gabralla, L. A. (2019). Recent advances in mobile touch screen security authentication methods: A systematic literature review. *Computers & Security*, *85*, 1–24. https://doi.org/10.1016/J.COSE.2019.04.008

Islam, A., Por, Y. L., & Othman, F. (2020). *A review of the recognition-based graphical password A review of the recognition-based graphical password*. *July*.

Jadhav, R. S., Chandole, D. K., Wani, M. D., Kuslkar, S. R., Shinde, K. G., & Dighe, M. S. (2014). Graphical Password Authentication System. *International Journal of Latest Technology in Engineering, Management & Applied Science*, *3*(3), 64–68.

Jeong, Y. S., & Kim, Y. T. (2015). A token-based authentication security scheme for Hadoop distributed file system using elliptic curve cryptography. *Journal of Computer Virology and Hacking Techniques*, *11*(3), 137–142. https://doi.org/10.1007/S11416-014-0236-5/FIGURES/4

Jin, A. T. B., Ling, D. N. C., & Goh, A. (2004). Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recognit.*, *37*(11), 2245–2255. https://doi.org/10.1016/j.patcog.2004.04.011

Karode, A., Mistry, S. & Chavan, S. (2013). Graphical password authentication system. *International Journal of Engineering Research & Technology (IJERT)*, 2(9), 2557-2560.

Kenneth, M. O., & Olujuwon, S. M. (2021). Web Application Authentication Using Visual Cryptography and Cued Clicked Point Recall-based Graphical Password. *Journal of Computer Science Research*, *3*(3). https://doi.org/10.30564/jcsr.v3i3.3535

Krombholz, K., Hobel, H., Huber, M., & Weippl, E. (2015). Advanced social engineering attacks. *Journal of Information Security and Applications*, *22*, 113–122. https://doi.org/10.1016/J.JISA.2014.09.005

Lamsal, K. (2020). *Designing and Developing a dynamic website using PHP Designing and Developing a dynamic website using PHP*.

Lashkari, A. H., Farmand, S., Zakaria, Dr. O. bin, & Saleh, Dr. R. (2009). Shoulder Surfing attack in graphical password authentication. *IJCSIS) International Journal of Computer Science and Information Security*, *6*(2). https://arxiv.org/abs/0912.0951v1

Lashkari, A. H., Gani, A., Sabet, L. G., & Farmand, S. (2010). A new algorithm on Graphical User Authentication (GUA) based on multi-line grids. *Scientific Research and Essays*, *5*(24), 3865–3875.

Leng, L., Teoh, A. B. J., Li, M., & Khan, M. K. (2014). A remote cancelable palmprint authentication protocol based on multi-directional two-dimensional PalmPhasor-fusion. *Security and Communication Networks*, *7*(11), 1860–1871. https://doi.org/10.1002/SEC.900

Li, L., Ali, J., Sullivan, N., Chatterjee, R., Tech, C., & Pal, B. (2019). Protocols for Checking Compromised Credentials. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 17. https://doi.org/10.1145/3319535

Li, Z., Sun, Q., Lian, Y., & Giusto, D. D. (2005). An association-based graphical password design resistant to shoulder-surfing attack. *IEEE International Conference on Multimedia and Expo, ICME 2005*, *2005*, 245–248. https://doi.org/10.1109/ICME.2005.1521406

Lin, T. H., Lee, C. C., Tsai, C. S., & Guo, S. D. (2010). A tabular steganography scheme for graphical password authentication. *Computer Science and Information Systems*, *7*(4), 823–841. https://doi.org/10.2298/CSIS081223028L

Ma, M., He, D., Fan, S., & Feng, D. (2020). Certificateless searchable public key encryption scheme secure against keyword guessing attacks for smart healthcare. *Journal of Information Security and Applications*, *50*, 102429. https://doi.org/10.1016/J.JISA.2019.102429

Mackie, I., & Yıldırım, M. (2018a). A Novel Hybrid Password Authentication Scheme Based on Text and Image. In *32nd Annual IFIP WG 11.3 Conference, DBSec 2018* (pp. 1–16).

Mackie, I., & Yıldırım, M. (2018b). A Novel Hybrid Password Authentication Scheme Based on Text and Image. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10980 LNCS*, 182–197. https://doi.org/10.1007/978-3-319-95729-6_12

Madhuravani, B., Reddy, Dr. P. B., & Lalithsamanthreddy, P. (2013). A Comprehensive Study on Different Authentication Factors. *International Journal of Engineering Research & Technology*, *2*(10). www.ijert.org

Masdari, M., & Ahmadzadeh, S. (2016). Comprehensive analysis of the authentication methods in wireless body area networks. *Security and Communication Networks*, *9*(17), 4777–4803. https://doi.org/10.1002/SEC.1642

Mathuri, M., & Valarmathi, P. A. (2013). A Secured Graphical Password Authentication System. *International Journal of Engineering Research & Technology*, *2*(5), 13–19. www.ijert.org

Mishra, D., Mukhopadhyay, S., Kumari, S., Khan, M. K., & Chaturvedi, A. (2014). Security Enhancement of a Biometric based Authentication Scheme for Telecare Medicine Information Systems with Nonce. *Journal of Medical Systems 2014 38:5*, *38*(5), 1–11. https://doi.org/10.1007/S10916-014-0041-1

Moraskar, V., Jaikalyani, S., Saiyyed, M., Gurnani, J., & Pendke, K. (2014). *Cued Click Point Technique for Graphical Password Authentication*. *3*(1), 166–172.

Nandgaonkar, Mr. V., Dongre, N., & Kashid, P. G. | D. H. | A. (2019). QR Code Based Secure Billing System for Shops using Cued Click Points. *International Journal of Trend in Scientific Research and Development*, *Volume-3*(Issue-4), 834–836. https://doi.org/10.31142/ijtsrd23946

Navaz, A. S. S., & Durairaj, K. (2016). Signature Authentication Using Biometric Methods. *International Journal of Science and Research (IJSR)*, *5*(1), 1581–1584. https://doi.org/10.21275/v5i1.nov153159

Ndia, J., Muketha, G., & Omieno, K. (2019). A Survey of Cascading Style Sheets Complexity Metrics. *International Journal of Software Engineering & Applications (IJSEA)*, *3*(10), 21–33. https://doi.org/10.2139/ssrn.3405783

Noroozi, M., & Eslami, Z. (2019). Public-key encryption with keyword search: a generic construction secure against online and offline keyword guessing attacks. *Journal of Ambient Intelligence and Humanized Computing 2019 11:2*, *11*(2), 879–890. https://doi.org/10.1007/S12652-019-01254-W

O'Gorman, L. (2003). Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, *91*(12), 2021–2040. https://doi.org/10.1109/JPROC.2003.819611

Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T., & Koucheryavy, Y. (2018). Multi-factor authentication: A survey. *Cryptography*, *2*(1), 1–31. https://doi.org/10.3390/cryptography2010001

Pal, B., Daniel, T., Chatterjee, R., & Ristenpart, T. (2019). Beyond credential stuffing: Password similarity models using neural networks. *Proceedings - IEEE Symposium on Security and Privacy*, *2019-May*, 417–434. https://doi.org/10.1109/SP.2019.00056

Panduranga Rao, P. G. (2013). A Study of Various Graphical Passwords Authentication Schemes Using Ai Hans Peter Wickelgren Approach. *IOSR Journal of Computer Engineering*, *10*(6), 14–20. https://doi.org/10.9790/0661-1061420

Patel, B., Sarwar, A., & Chavan, S. (2021). Graphical Password Authentication Using Colour Login Technique. *International Research Journal of Engineering and Technology*. www.irjet.net

Patra, K., Nemade, B., Mishra, D. P., & Satapathy, P. P. (2016). Cued-Click Point Graphical Password Using Circular Tolerance to Increase Password Space and Persuasive Features. *Procedia Computer Science*, *79*, 561–568. https://doi.org/10.1016/j.procs.2016.03.071

Peroni, S., Osborne, F., Iorio, A. di, Giovanni, A., Poggi, F., Vitali, F., & Motta, E. (2017). Research Articles in Simplified HTML : a Web-first format for HTML-based scholarly articles. *PeerJ Computer Science*, *2*(3), 1–35. https://doi.org/10.7717/peerj-cs.132

Por, L. Y., Ku, C. S., Islam, A., & Ang, T. F. (2017). Graphical password: prevent shoulder-surfing attack using digraph substitution rules. *Frontiers of Computer Science*, *11*(6), 1098–1108. https://doi.org/10.1007/s11704-016-5472-z

Prokofyeva, N., & Boltunova, V. (2017). Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems. *Procedia - Procedia Computer Science*, *104*(December 2016), 51–56. https://doi.org/10.1016/j.procs.2017.01.059

Ramalingam, R. (2017). Password-based Authentication in Computer Security: Why is it still there? *The SIJ Transactions on Computer Science Engineering & Its Applications (CSEA)*, *5*(2). https://www.researchgate.net/publication/316350564

Sahu, S. B., & Singh, A. (2014). Secure User Authentication & Graphical Password using Cued Click-Points. *International Journal of Computer Trends and Technology*, *18*(4), 156–160. https://doi.org/10.14445/22312803/ijctt-v18p137

Salahdine, F., & Kaabouch, N. (2019). Social Engineering Attacks: A Survey. *Future Internet 2019, Vol. 11, Page 89*, *11*(4), 89. https://doi.org/10.3390/FI11040089

Salim Istyaq, M. S. U. (2018). Hybrid Authentication Scheme for Graphical Password Using QR Code and Integrated Sound Signature. *International Journal of Computer and Information Engineering*, *12*(2), 111–115. https://waset.org/publications/10009015/hybrid-authentication-scheme-for-graphical-password-using-qr-code-and-integrated-sound-signature

Santwana, C. (2014). Hypervisor based Mitigation Technique for Keylogger Spyware Attacks. *International Journal of Computer Science and Information Technologies*, *5*(2), 1867–1870.

Sarma, M., Kundu, D., & Mall, R. (2007). Automatic test case generation from UML sequence diagrams. *Proceedings of the 15th International Conference on Advanced Computing and Communications, ADCOM 2007*, 60–65. https://doi.org/10.1109/ADCOM.2007.68

Sepideh, F. (2019). Providing a Secure Hybrid Method for Graphical Password Authentication to Prevent Shoulder Surfing, Smudge and Brute Force Attack. *International Journal of Computer and Information Engineering*, *13*(12), 624–628.

Shah, A., Ved, P., Deora, A., Jaiswal, A., & D'Silva, M. (2015). Shoulder-surfing resistant graphical password system. *Procedia Computer Science*, *45*(C), 477–484. https://doi.org/10.1016/j.procs.2015.03.084

Shah, M. K., Kataria, D., Raj, S. B., & Priya, G. (2020). *Real Time Working of Keylogger Malware Analysis*. *9*(10), 569–573.

Shantha, R., Kumari, S., & Viji, S. (2015). Cued Click Points Password Authentication using Picture Grids. *IJCSN International Journal of Computer Science and Network*, *4*(6), 2277–5420. www.IJCSN.org

Shen, S. S., Kang, T. H., Lin, S. H., & Chien, W. (2017). Random graphic user password authentication scheme in mobile devices. *Proceedings of the 2017 IEEE International Conference on Applied System Innovation: Applied System Innovation for Modern Technology, ICASI 2017*, 1251–1254. https://doi.org/10.1109/ICASI.2017.7988123

Shinde, S., & Wanaskar, U. H. (2016). *Keylogging : A Malicious Attack*. *5*(6), 285–289. https://doi.org/10.17148/IJARCCE.2016.5661

Shnain, A. H., & Shaheed, S. H. (2018). The use of graphical password to improve authentication problems in e-commerce. *AIP Conference Proceedings*, *2016*(September). https://doi.org/10.1063/1.5055535

Stobert, E. (2015). *Graphical Passwords and Practical Password Management*. 235. https://curve.carleton.ca/system/files/etd/a117ba9b-3e10-4156-bc09-3119aad500a5/etd_pdf/c22d4a17276e71f36c49281c838c7e36/stobert-graphicalpasswordsandpracticalpasswordmanagement.pdf

Sun, H. M., Chen, Y. H., & Lin, Y. H. (2012). oPass: A user authentication protocol resistant to password stealing and password reuse attacks. *IEEE Transactions on Information Forensics and Security*, *7*(2), 651–663. https://doi.org/10.1109/TIFS.2011.2169958

Swapnil Sunil, S., Prakash, D., & Ramesh Shivaji, Y. (2014). Cued Click Points: Graphical Password Authentication Technique for Security. *(IJCSIT) International Journal of Computer Science and Information Technologies*, *5*(2), 1073–1075.

Syukri, A. F., Okamoto, E., & Mambo, M. (1998). A user identification system using signature written with mouse. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *1438*, 403–414. https://doi.org/10.1007/BFB0053751

Togookhuu, B., & Zhang, J. (2017). New Graphic Password Scheme Containing Questions-Background-Pattern and Implementation. *Procedia Computer Science*, *107*, 148–156. https://doi.org/10.1016/j.procs.2017.03.071

Vaddeti, A., Vidiyala, D., Puritipati, V., Ponnuru, R. B., Shin, J. S., & Alavalapati, G. R. (2020). Graphical passwords: Behind the attainment of goals. *Security and Privacy*, *3*(6). https://doi.org/10.1002/spy2.125

Vaithyasubramanian, S., Christy, A., & Saravanan, D. (2014). An analysis of Markov password against brute force attack for effective web applications. *Applied Mathematical Sciences*, *8*(117–120), 5823–5830. https://doi.org/10.12988/ams.2014.47579

Wang, D., & Wang, P. (2015). Offline dictionary attack on password authentication schemes using smart cards. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *7807*, 221–237. https://doi.org/10.1007/978-3-319-27659-5_16

Wazid, M., Katal, A., Goudar, R. H., Singh, D. P., Tyagi, A., Sharma, R., & Bhakuni, P. (2013). A framework for detection and prevention of novel keylogger spyware attacks. *7th International Conference on Intelligent Systems and Control, ISCO 2013*, 433–438. https://doi.org/10.1109/ISCO.2013.6481194

Yang, D., Doh, I., & Chae, K. (2017). Enhanced password processing scheme based on visual cryptography and OCR. *International Conference on Information Networking,* 254–258. https://doi.org/10.1109/ICOIN.2017.7899514

Yenape, R. S., & Waghmare, A. (2017). Three Way Graphical Password Authentication. *International Advanced Research Journal in Science, Engineering and Technology* (*IARJSET), 4*(4), 155–157. https://doi.org/10.17148/iarjset/nciarcse.2017.45

Yildirim, M. (2017). Security and Usability in Password Authentication. Published Ph.D Thesis, School of Engineering and Informatics of the University of Sussex http://sro.sussex.ac.uk/id/eprint/71873/1/Yildirim%2C Merve.pdf

# APPENDIX A

# SOURCE CODE

## Textual Authentication Registration Page

```
<?php require_once "controllerUserData.php"; ?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Signup Form</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="container">
        <div class="row">
            <div class="col-md-4 offset-md-4 form">
                <form action="index.php" method="POST" autocomplete="">
                    <h2 class="text-center">Signup Form</h2>
                    <p class="text-center">It's quick and easy.</p>
                    <?php
                    if(count($errors) == 1){
                        ?>
                        <div class="alert alert-danger text-center">
                            <?php
                            foreach($errors as $showerror){
                                echo $showerror;
                            }
                            ?>
                        </div>
                        <?php
                    }elseif(count($errors) > 1){
                        ?>
                        <div class="alert alert-danger">
                            <?php
                            foreach($errors as $showerror){
                                ?>
                                <li><?php echo $showerror; ?></li>
                                <?php
                            }
                            ?>
                        </div>
                        <?php
                    }
                    ?>
                    <div class="form-group">
                        <input class="form-control" type="text" name="name" placeholder="Full Name" required value="<?php echo $name ?>">
                    </div>
                    <div class="form-group">
                        <input class="form-control" type="email" name="email" placeholder="Email Address" required value="<?php echo $email ?>">
                    </div>
                    <div class="form-group">
                        <input class="form-control" type="password" name="password" placeholder="Password" required>
                    </div>
                    <div class="form-group">
                        <input class="form-control" type="password" name="cpassword" placeholder="Confirm password" required>
                    </div>
                    <div class="form-group">
                        <input class="form-control button" type="submit" name="signup" value="Signup">
                    </div>
                    <div class="link login-link text-center">Already a member? <a href="login-user.php">Login here</a></div>
                </form>
            </div>
        </div>
    </div>

</body>
</html>
```

### OTP Generation and Verification / Form submission and validation Code

```
<?php
session_start();
```

```
require "connection.php";
$email = "";
$name = "";
$errors = array();

//if user signup button
if(isset($_POST['signup'])){
    $name = mysqli_real_escape_string($con, $_POST['name']);
    $email = mysqli_real_escape_string($con, $_POST['email']);
    $password = mysqli_real_escape_string($con, $_POST['password']);
    $cpassword = mysqli_real_escape_string($con, $_POST['cpassword']);
    if($password !== $cpassword){
        $errors['password'] = "Confirm password not matched!";
    }
    $email_check = "SELECT * FROM usertable WHERE email = '$email'";
    $res = mysqli_query($con, $email_check);
    if(mysqli_num_rows($res) > 0){
        $errors['email'] = "Email that you have entered is already exist!";
    }
    if(count($errors) === 0){
        $encpass = password_hash($password, PASSWORD_BCRYPT);
        $code = rand(999999, 111111);
        $status = "notverified";
        $insert_data = "INSERT INTO usertable (name, email, password, code, status)
                        values('$name', '$email', '$encpass', '$code', '$status')";
        $data_check = mysqli_query($con, $insert_data);
        if($data_check){
            $subject = "Email Verification Code";
            $message = "Your verification code is $code";
            $sender = "From: shahiprem7890@gmail.com";
            if(mail($email, $subject, $message, $sender)){
                $info = "We've sent a verification code to your email - $email";
                $_SESSION['info'] = $info;
                $_SESSION['email'] = $email;
                $_SESSION['password'] = $password;
                header('location: user-otp.php');
                exit();
            }else{
                $errors['otp-error'] = "Failed while sending code!";
            }
        }else{
            $errors['db-error'] = "Failed while inserting data into database!";
        }
    }

}
    //if user click verification code submit button
    if(isset($_POST['check'])){
        $_SESSION['info'] = "";
        $otp_code = mysqli_real_escape_string($con, $_POST['otp']);
        $check_code = "SELECT * FROM usertable WHERE code = $otp_code";
        $code_res = mysqli_query($con, $check_code);
        if(mysqli_num_rows($code_res) > 0){
            $fetch_data = mysqli_fetch_assoc($code_res);
            $fetch_code = $fetch_data['code'];
            $email = $fetch_data['email'];
            $code = 0;
            $status = 'verified';
            $update_otp = "UPDATE usertable SET code = $code, status = '$status' WHERE
code = $fetch_code";
            $update_res = mysqli_query($con, $update_otp);
            if($update_res){
                $_SESSION['name'] = $name;
                $_SESSION['email'] = $email;
                header('location: registration/signupnew.php');
                exit();
            }else{
                $errors['otp-error'] = "Failed while updating code!";
            }
        }else{
            $errors['otp-error'] = "You've entered incorrect code!";
        }
    }

    //if user click login button
    if(isset($_POST['login'])){
        $email = mysqli_real_escape_string($con, $_POST['email']);
        $password = mysqli_real_escape_string($con, $_POST['password']);
        $check_email = "SELECT * FROM usertable WHERE email = '$email'";
        $res = mysqli_query($con, $check_email);
        if(mysqli_num_rows($res) > 0){
            $fetch = mysqli_fetch_assoc($res);
            $fetch_pass = $fetch['password'];
            if(password_verify($password, $fetch_pass)){
                $_SESSION['email'] = $email;
                $status = $fetch['status'];
                if($status == 'verified'){
                  $_SESSION['email'] = $email;
                  $_SESSION['password'] = $password;
                    //header('location: home.php');
                    $code = rand(999999, 111111);
                    $insert_code = "UPDATE usertable SET code = $code WHERE email = '$email'";
                    $run_query =  mysqli_query($con, $insert_code);
                    if($run_query){
                        $subject = "Password Reset Code";
                        $message = "Your password reset code is $code";
                        $sender = "From: shahiprem7890@gmail.com";
                        if(mail($email, $subject, $message, $sender)){
                            $info = "We've sent a verification code to your email - $email";
                            $_SESSION['info'] = $info;
                            $_SESSION['email'] = $email;
                            $_SESSION['password'] = $password;
                            header('location: verify-login.php');
                            exit();
                        }else{
                            $errors['otp-error'] = "Failed while sending code!";
                        }
                    }else{
                        $errors['db-error'] = "Something went wrong!";
                    }
                    header('location: user-otp.php');
                }else{
                    $info = "It's look like you haven't still verify your email - $email";
                    $_SESSION['info'] = $info;
                    header('location: user-otp.php');
                }
            }else{
                $errors['email'] = "Incorrect email or password!";
            }
        }else{
            $errors['email'] = "It's look like you're not yet a member! Click on the bottom link to signup.";
        }
    }

    //if user click continue button in forgot password form
    if(isset($_POST['check-email'])){
        $email = mysqli_real_escape_string($con, $_POST['email']);
        $check_email = "SELECT * FROM usertable WHERE email='$email'";
        $run_sql = mysqli_query($con, $check_email);
        if(mysqli_num_rows($run_sql) > 0){
            $code = rand(999999, 111111);
            $insert_code = "UPDATE usertable SET code = $code WHERE email = '$email'";
            $run_query =  mysqli_query($con, $insert_code);
            if($run_query){
                $subject = "Password Reset Code";
                $message = "Your password reset code is $code";
                $sender = "From: shahiprem7890@gmail.com";
                if(mail($email, $subject, $message, $sender)){
                    $info = "We've sent a passwrod reset otp to your email - $email";
                    $_SESSION['info'] = $info;
                    $_SESSION['email'] = $email;
                    header('location: reset-code.php');
                    exit();
                }else{
                    $errors['otp-error'] = "Failed while sending code!";
                }
            }else{
                $errors['db-error'] = "Something went wrong!";
            }
        }else{
            $errors['email'] = "This email address does not exist!";
        }
    }

    //if user click check reset otp button
    if(isset($_POST['check-reset-otp'])){
        $_SESSION['info'] = "";
        $otp_code = mysqli_real_escape_string($con, $_POST['otp']);
        $check_code = "SELECT * FROM usertable WHERE code = $otp_code";
        $code_res = mysqli_query($con, $check_code);
        if(mysqli_num_rows($code_res) > 0){
            $fetch_data = mysqli_fetch_assoc($code_res);
```

```
            $email = $fetch_data['email'];
            $_SESSION['email'] = $email;
            $info = "Please create a new password that you don't use on any other site.";
            $_SESSION['info'] = $info;
            header('location: new-password.php');
            exit();
        }else{
            $errors['otp-error'] = "You've entered incorrect code!";
        }
    }

    //if user click check verify login otp button
    if(isset($_POST['verifyotp'])){
        $_SESSION['info'] = "";
        $otp_code = mysqli_real_escape_string($con, $_POST['otp']);
        $check_code = "SELECT * FROM usertable WHERE code = $otp_code";
        $code_res = mysqli_query($con, $check_code);
        if(mysqli_num_rows($code_res) > 0){
            $fetch_data = mysqli_fetch_assoc($code_res);
            $fetch_code = $fetch_data['code'];
            $email = $fetch_data['email'];
            $code = 0;
            $status = 'verified';
            $update_otp = "UPDATE usertable SET code = $code, status = '$status' WHERE
code = $fetch_code";
            $update_res = mysqli_query($con, $update_otp);
            if($update_res){
                $_SESSION['name'] = $name;
                $_SESSION['email'] = $email;
                header('location: log_in/login.php');
                exit();
            }else{
                $errors['otp-error'] = "Failed while updating code!";
            }
        }else{
            $errors['otp-error'] = "You've entered incorrect code!";
        }
    }

    //if user click change password button
    if(isset($_POST['change-password'])){
        $_SESSION['info'] = "";
        $password = mysqli_real_escape_string($con, $_POST['password']);
        $cpassword = mysqli_real_escape_string($con, $_POST['cpassword']);
        if($password !== $cpassword){
            $errors['password'] = "Confirm password not matched!";
        }else{
            $code = 0;
            $email = $_SESSION['email']; //getting this email using session
            $encpass = password_hash($password, PASSWORD_BCRYPT);
            $update_pass = "UPDATE usertable SET code = $code, password = '$encpass'
WHERE email = '$email'";
            $run_query = mysqli_query($con, $update_pass);
            if($run_query){
                $info = "Your password changed. Now you can login with your new password.";
                $_SESSION['info'] = $info;
                header('Location: password-changed.php');
            }else{
                $errors['db-error'] = "Failed to change your password!";
            }
        }
    }

   //if login now button click
    if(isset($_POST['login-now'])){
        header('Location: login-user.php');
    }
?>
```

### Graphical Password Page

```
<!DOCTYPE html>
<html>
<head>
      <title>Register</title>
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <meta charset="utf-8">
      <link rel="stylesheet" href="css/style-footer.css">
            <link href="css/style1.css" rel="stylesheet">
      <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/font-
awesome/4.2.0/css/font-awesome.min.css">
```

```
        <link href="css/font-awesome.min.css" rel="stylesheet" type="text/css"
media="all">
    <link href="css/style-body.css" rel="stylesheet" type="text/css" media="all"/>

    <script>
    // passing the selected image reference to slice the image
    function changeIt(img)
    {
        var name = img.src;
        console.log(name);
        window.location.href = "reg_slice1.php?var="+name;
    }
    </script>

</head>

<body>
<!--Main Header-->
<nav class="navbar navbar-default">
     <div class="container">
         <!-- Brand and toggle get grouped for better mobile display -->
         <div class="navbar-header">
             <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1"
                 aria-expanded="false">
                 <span class="sr-only">Toggle navigation</span>
                 <span class="icon-bar"></span>
                 <span class="icon-bar"></span>
                 <span class="icon-bar"></span>
             </button>
         </div>
         <!-- Collect the nav links, forms, and other content for toggling -->
         <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
             <ul class="nav navbar-nav">
                 <li class="active">
                     <a href="../index.html">Home</a>
                 </li>
                 <li>
                     <a href="../about.html">About Us</a>
                 </li>
                 <li>
```

```
                     <a href="#">Service</a>
                 </li>
                 <li>
                     <a href="#">Gallery</a>
                 </li>
                 <li>
                     <a href="#">Blog</a>
                 </li>
                 <li>
                     <a href="../contact.html">Contact Us</a>
                 </li>
             </ul>
         </div>
         <!-- /.navbar-collapse -->
     </div>
     <!-- /.container-fluid -->
   </nav>
   <!--End Main Header -->

<!-- signup form -->
<div class="signupform">
        <div class="container">
                <div class="agile_info">
                        <div class="login_form">
                                <div class="left_grid_info">
                                        <h1>Manage Your User Account</h1>
                                        <p>This system provides high security to your
account through the graphical password.</p><br>
                                        <img class="im1" src="../images/cover.jpg"
height="270" width="370">
                                </div>
                        </div>
                        <div class="login_info">
                                <h2>Create New Account</h2>
                                <p class="account1">Select the 1st image for the graphical
password.</p>
                                <center>
                                <img class="im" src="..\images\pw\image1.jpg"
onclick="changeIt(this)" height="120" width="120">
                                <img class="im" src="..\images\pw\image2.jpg"
onclick="changeIt(this)" height="120" width="120">
```

```
                <img class="im" src="..\images\pw\image3.jpg"
onclick="changeIt(this)" height="120" width="120">
                <img class="im" src="..\images\pw\image4.jpg"
onclick="changeIt(this)" height="120" width="120">
                <img class="im" src="..\images\pw\image5.jpg"
onclick="changeIt(this)" height="120" width="120">
        <img class="im" src="..\images\pw\image6.jpg" onclick="changeIt(this)"
height="120" width="120">
        <img class="im" src="..\images\pw\image7.jpg" onclick="changeIt(this)"
height="120" width="120">
        <img class="im" src="..\images\pw\image8.jpg" onclick="changeIt(this)"
height="120" width="120">
        <img class="im" src="..\images\pw\image9.jpg" onclick="changeIt(this)"
height="120" width="120">
        <img class="im" src="..\images\pw\image10.jpg" onclick="changeIt(this)"
height="120" width="120">
        <img class="im" src="..\images\pw\image11.jpg" onclick="changeIt(this)"
height="120" width="120">
                </center>
            </div>
        </div>
    </div>
</div>

<!-- footer -->
<footer class="footer-distributed">
  <!-- footer left -->
  <div class="footer-left">
    <h3>Company<span>logo</span></h3>
    <p class="footer-links">
      <a href="#">Home</a>
      ·
      <a href="#">Blog</a>
      ·
      <a href="#">About</a>
      ·
      <a href="#">Faq</a>
      ·
      <a href="#">Contact</a>
    </p>
    <p class="footer-company-name">Company Name &copy; 2019</p>
    </div>
    <!-- footer center-->
    <div class="footer-center">
        <div>
            <i class="fa fa-map-marker"></i>
            <p><span>21 Bandaranaike Mawatha</span> Katubedda, Sri Lanka</p>
        </div>

        <div>
            <i class="fa fa-phone"></i>
            <p>+ (9471) - 123 - 4567</p>
        </div>

        <div>
            <i class="fa fa-envelope"></i>
            <p><a href="mailto:support@company.com">support@company.com</a></p>
        </div>
    </div>
    <!-- footer right-->
    <div class="footer-right">
        <p class="footer-company-about">
            <span>About the company</span>
            Lorem ipsum dolor sit amet, consectateur adispicing elit. Fusce euismod
convallis velit, eu auctor lacus vehicula sit amet.
        </p>

        <div class="footer-icons">

            <a href="#"><i class="fa fa-facebook"></i></a>
            <a href="#"><i class="fa fa-twitter"></i></a>
            <a href="#"><i class="fa fa-linkedin"></i></a>
            <a href="#"><i class="fa fa-github"></i></a>

        </div>

    </div>

</footer>

<script src="plugins/jquery.js"></script>
<script src="plugins/bootstrap.min.js"></script>
```

```
<script src="plugins/bootstrap-select.min.js"></script>

<script src="plugins/validate.js"></script>
<script src="plugins/wow.js"></script>
<script src="plugins/jquery-ui.js"></script>
<script src="js/script.js"></script>

</body>
</html>
```