Graphical Password Authentication

Submitted in partial fulfillment of the

requirements of the degree of

Bachelor of
Engineering

Avani Deora, 60003115006
Amish Shah, 60003115042
Parth Ved, 60003115059

Supervisors:

Guide:
Prof. Arjun Jaiswal

Co-Guide:
Prof. Mitchell D’Silva

Information
Technology

Dwarkadas J. Sanghvi College of Engineering, University Of
Mumbai
2014-2015

SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
Approved by AICTE and Affiliated to the University of Mumbai

Certificate

This is to certify that the project entitled **“Graphical Password Authentication”** is a bonafide work of “**Amish Shah (60003115042), Avani Deora (60003115006), and Parth Ved (60003115059)”** submitted to the University of Mumbai in partial fulfilment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Information Technology”**.

Internal Guide
Prof. Arjun Jaiswal

Dr. A. R. Joshi
(HOD IT Dept.)

Dr. Hari Vasudevan
(Principal)

SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
Approved by AICTE and Affiliated to the University of Mumbai

Project Report Approval for B. E.

This project report entitled "***Graphical Password Authentication"*** by ***Avani Deora, Amish Shah and Parth Ved*** is approved for the degree of ***Information Technology***.

Examiners 1.---

2. ---

Date: 23/04/2015
Place: Mumbai

SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
Approved by AICTE and Affiliated to the University of Mumbai

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Avani Deora,
60003115006

-- --------------------------------

Amish Shah,
60003115042

-- --------------------------------

Parth Ved,
60003115059

-- --------------------------------

(Name of student and Roll No) (Signature)

Date:
23/04/2015

ACKNOWLEDGEMENTS

We are highly indebted to Dwarkadas J. Sanghvi College of Engineering for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our heartfelt gratitude towards our guide *Prof. Arjun Jaiswal* and our co-guide *Prof. Mitchell D'Silva* for their kind co-operation and encouragement which help us in completion of this Synopsis.

We would like to express our special gratitude and thanks to faculty of Information Technology Department for giving us such attention and time. Our thanks and appreciations also go to our colleagues in developing the project and people who have willingly helped us out with their abilities.

Avani Deora
Amish Shah
Parth Ved

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

Authentication is the act of confirming the truth of an attribute of a single piece of data. Nowadays username and text–based password are the most common and widely used technique in knowledge-based authentication methods. However, the vulnerabilities of this traditional technique are well known. One of the main problems is the difficulty of remembering passwords. Studies have shown that users tend to pick short passwords or passwords that are easily remembered for example pet's name, first name and street address.

The shoulder surfing attack can be performed by the antagonist to obtain the user's password by watching over the user's shoulder as he enters his password. Traditionally, shoulder surfing attacks also called "peeping attacks" concerns moved from telephone calling card fraud to automated teller machine (ATM) fraud, and more recently to mobile computer users. Shoulder surfing attacks also include the use of binoculars or more lately camera phones to record users entering their personal identification numbers (PINs).
This paper provides the analysis and presents the performance of a graphics oriented password entry system that greatly reduces the threat of shoulder surfing.

The main purpose of improving the existing user authentication technology is to make the method simple and usable which brings an ease of use to the user. Graphical authentication schemes have been proposed as a substitute to the traditional password techniques which are text-based, inspired predominantly by the fact that humans can remember images better than text. Images are by and large simpler to reminisced or recognized than text, especially photos, which are even easier to be remembered than random pictures. It has also been recommended that graphical passwords is more difficult to guess or broken by brute force. If the number of possible images is sufficiently large, the possible password space of a graphical password scheme may exceed that of text-based schemes which in turn reduces the possibility of dictionary attacks. The use of graphical passwords for authentication is on rise lately because of the few merits mentioned above. As a result graphical password find a lot of applications in systems which require high security. In addition, graphical passwords have also been implemented and applied to workstations, websites, ATM machines and mobile devices such as personal digital assistants (PDAs).

1.1 Problem Definition

Textual passwords are the most common method used for authentication. But textual passwords are vulnerable to eves dropping, dictionary attacks, social engineering and shoulder surfing. Graphical passwords are introduced as alternative techniques to textual passwords. Most of the graphical schemes are vulnerable to shoulder surfing. To address this problem, text can be combined with images or colors to generate session passwords for authentication. Shoulder surfing resistant authentication schemes which are less complex than the existing systems will provide optimum user comfort with security.

1.2 Aim and Scope of the project

This project will develop a picture-based authentication system which is a web based application. The type of algorithm used during the authentication process is a newly developed algorithm based on the research of multiple existing graphical password schemes. The picture-based authentication system also includes anti-shoulder surfing mechanism which prevents shoulder surfer from guessing the user’s password. In addition, the system will provide an optimal level of security and requires lesser effort for the user to remember his password. On top of that, the system developed will only require minimal of time for the user to finish the authentication process.

Chapter 2

Review of Literature

Here we give review of materials such as IEEE paper, existing Password Schemes, and different technologies to go about building the project.

- **IEEE papers: Authentication Schemes for Session Passwords using Color and Images**

In this paper, two new authentication schemes are proposed for PDAs. The proposed authentication schemes use text and colors for generating session passwords.

- **PHP**

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

- **jQuery**

jQuery is a cross-platform designed to simplify the client-side scripting of HTML. jQuery is free, open source software, licensed under the MIT License. jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and web applications.

- **JavaScript**

A scripting language developed by Netscape to enable Web authors to design interactive sites. Although it shares many of the features and structures of the full Java language, it was developed independently. JavaScript can interact with HTML source code, enabling Web authors to spice up their sites with dynamic content. JavaScript is endorsed by a number of software companies and is an open language that anyone can use without purchasing a license. It is supported by recent browsers from Netscape and Microsoft, though Internet Explorer supports only a subset, which Microsoft calls Jscript.

- **HTML and CSS**

HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>). The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language rather than a programming language. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of HTML web pages.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.

2.1 Existing System

Jensen et al. proposed a graphical password scheme based on “picture password”. This scheme was designed especially for mobile devices such as PDAs. Throughout the password creation, the user has to select the theme first (e.g. sea and shore, cat and dog and etc) which consists of thumbnail photos. The user then selects and registers a sequence of the selected thumbnail photo to form a password (Figure 1). The user needs to recognize and identify the previously seen photos and touch it using a stylus in the correct sequence in order to be authenticated. However, as the numbers of thumbnail photos are limited only to 30, the size of the password space is considered small. A numerical value is assigned for each thumbnail photo and the sequence of selection will produce a numerical password. This numerical password is shorter than the length of textual password. To overcome this problem a user can select one or two thumbnail photos as one single action in order to create and enlarge the size of the password space. However, this will make the memorability of the created password become more complex and difficult.

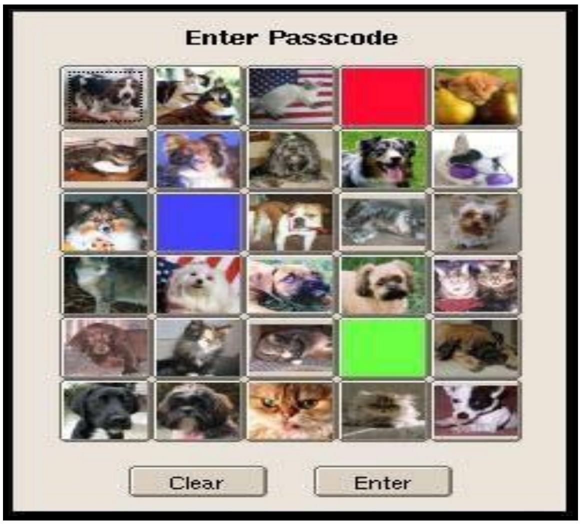


Figure 1.1: Cats and dog theme

Based on the assumption that human can recall human faces easier than other pictures, Real User Corporation has developed their own commercial product named Passfaces TM. Basically, Passfaces works as follows, users are required to select the previously seen human face from a grid of nine faces one of which is known while the rest are decoys. This step is continuously repeated until all the four faces are identified.

A comparative study conducted by Brostoff and Sasse in which 34 subjects involved in thetest showed that, the Passfaces password is easier to remember compared to textual passwords. Results also showed that Passfaces took a much longer login time than textual passwords. Empirical and comparative studies by Davis et al. showed that, in Passfaces the user's choice is highly affected by race, the gender of the user and the attractiveness of the faces. This will make the Passfaces password somewhat predictable.

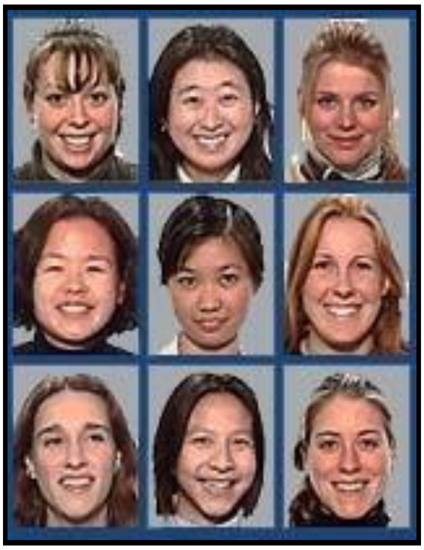


Fig. 1.2: Passfaces

Dhamija and Perrig proposed a scheme using a hash visualization technique on the abstract images. The scheme is called “Déjà vu” (Figure 7). 20 participants were involved in this study. The participants were asked to create the Déjà vu password by selecting 5 images from a challenge set of 25 images. At the same time, the participants were required to create the text-based password which is at least 6 characters long. After the password creations were finished, participants need to authenticate themselves using both techniques. According to their studies, the result showed that it took more time to create a graphical password compared to traditional approach. Besides that, 90% of the authentication using Déjà vu succeeded compared to 70% using the traditional approach. However, due to the larger amount of pictures stored on the server side, the authentication process can be slow due to network traffic delay. Even though the size of the password space of Déjà vu is much smaller compared to text based password, it cannot be concluded that Déjà vu scheme is easy to remember.

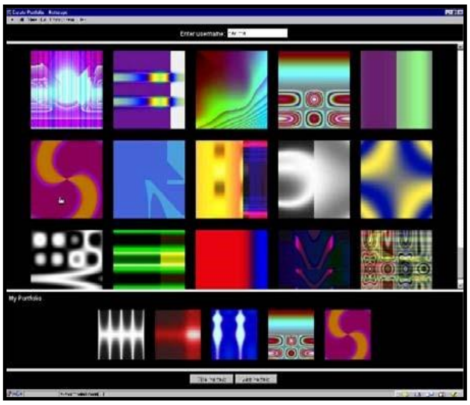


Fig 1.3: DejaVu Scheme

Sobrado and Birget proposed graphical passwords schemes that overcome shoulder-surfing attacks. In their first scheme which they called “triangle scheme”, a user needs to select their pass-object among many displayed object. To be authenticated, a user needs to recognize all the pre-selected pass-object which was selected during the registration phase. Users are required to click inside the convex-hull which is formed by the pass-object. To make the password space large enough and difficult to guess, Sobrado and Birget suggested using 1000 objects on the login process. However, by increasing the number objects, the display becomes more crowded and making it difficult to find the pass-object. On the other hand if the number of objects is reduced, the size of password space will become smaller thus making it easier to crack and guess.

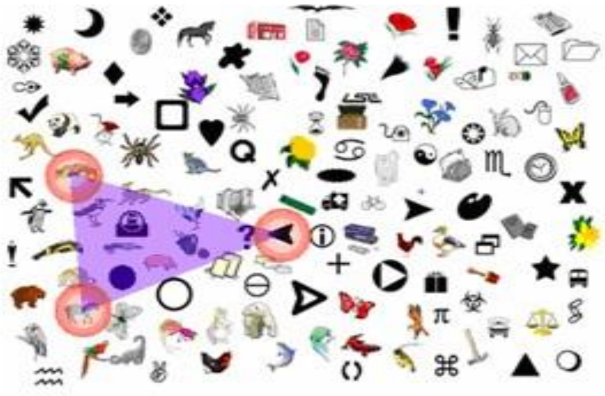


Fig 1.4: Convex Hull shoulder-surfing resistant

2.2 Pitfalls in Existing System

The current scenario of authenticating into any system suffers from following pitfalls:

Shoulder Surfing: In normal session based password, the password entered by the user by the means of images can be seen by a person standing beside the user. The attacker may then try different possibilities and break into the system, etc.

Guessing: When the images are arranged in a pattern, the images may be guessed by the attacker and he may try different combinations and gain illegal access.

Brute Force Attacks: Brute force attacks are the most common in text based passwords. Graphical passwords are generally resistant to this attack.

Complexity: The Graphical passwords which are difficult to guess require a complex sequence of actions that the user needs to do. This compromises the ease of use of the authentication method.

Chapter 3

Project Management

Project management is the discipline of planning, organizing, securing and managing resources to achieve specific goals.

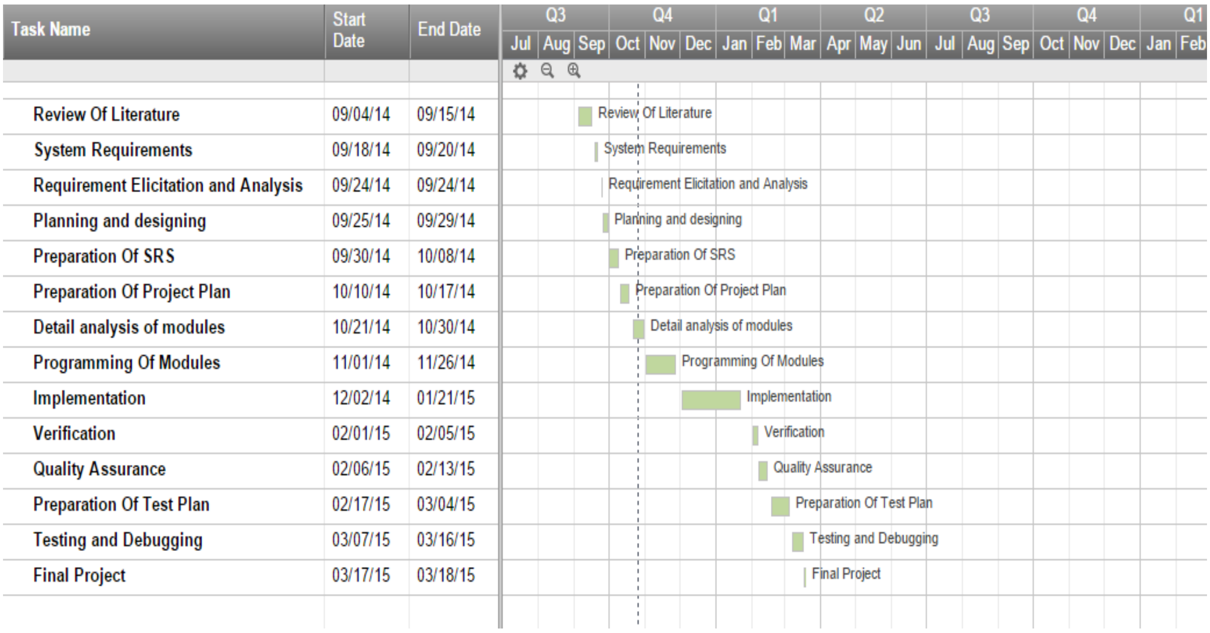
3.1 Schedule

Schedule helped us to know project's milestones, activities, and deliverables, with start and finish dates.

3.1.1 Gantt Chart

Project scope is defined and the appropriate methods for completing the project are determined. The durations for the various tasks necessary to complete the work are listed.

Gantt chart for our project is as follows:



Task Name	Start Date	End Date
Review Of Literature	09/04/14	09/15/14
System Requirements	09/18/14	09/20/14
Requirement Elicitation and Analysis	09/24/14	09/24/14
Planning and designing	09/25/14	09/29/14
Preparation Of SRS	09/30/14	10/08/14
Preparation Of Project Plan	10/10/14	10/17/14
Detail analysis of modules	10/21/14	10/30/14
Programming Of Modules	11/01/14	11/26/14
Implementation	12/02/14	01/21/15
Verification	02/01/15	02/05/15
Quality Assurance	02/06/15	02/13/15
Preparation Of Test Plan	02/17/15	03/04/15
Testing and Debugging	03/07/15	03/16/15
Final Project	03/17/15	03/18/15

Figure 3.1: Gantt chart

3.1.2 Task network

Task Network is an approach in which, dependency among actions can be given in the form of networks. Following is the task network of our project:

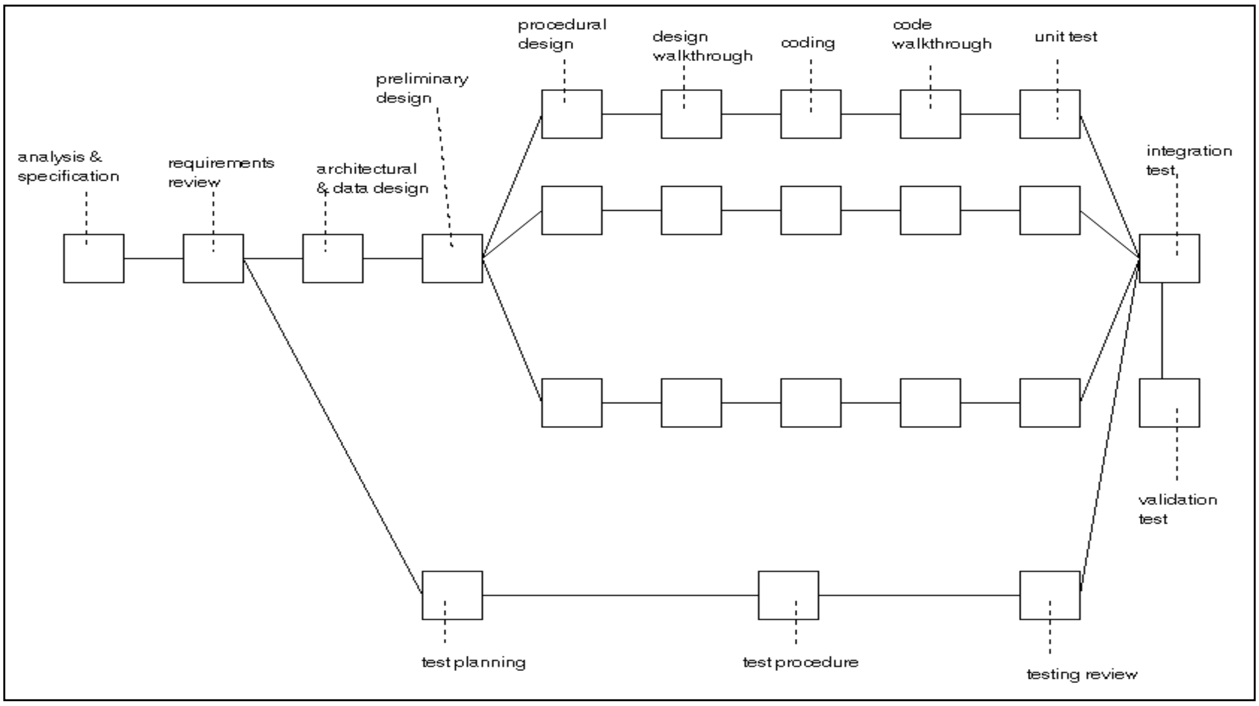


Figure 3.2: Task network diagram

The project begins with the primitive tasks such as study of existing system, followed by defining problem definition and scope, gathering of requirements. Compound tasks involve algorithm design, UML diagrams, implementation of modules and integration. Goal tasks involve testing, development and documentation.

3.2 Project Resources

3.2.1 Hardware Requirements

Processor: minimum Pentium IV

System Bus: 32- Bit

RAM: 1GB

Hard drive: 160GB

Display: SVGA Colour

Key board: Windows compatible

3.2.2 Software Requirements

IDE: Microsoft Visual Studio for web/WebMatrix

3.2.3 Operating Environment

Windows XP/Vista/7/8, Mac OSX, Linux.

3.3 Estimation

Estimation is basically identifying and acquiring necessary resources such as equipment’s, materials, man-power etc. required for accomplishing the project successfully. Estimation techniques used for our project are as follows:

Lines of Code

Lines of code (LOC) is a software metric used to measure the size of a computer program by counting the number of lines in the text of the program's source code. It is typically used to predict the amount of effort that will be required to develop a program, as well as to estimate programming productivity or maintainability once the software is produced.

For our project, the estimated lines of code = 6.5 K

The above mentioned Lines of Code (LOC) include the following:

- Authentication to access the tool
- Code for adding/deleting questions and updating keywords of the questions
- Logic for highlighting the important sentences in the document
- Frequency calculation of the keywords in the document for report preparation
- Graph generation code to display the overall performance of the class

3.3.1 COCOMO Estimation Model

The Constructive Cost Model (COCOMO) is an algorithmic software cost estimation model that computes software development effort and cost as a function of program size. Program size is expressed in estimated thousands of source lines of code (SLOC). Basic COCOMO is good for quick estimate of software cost. COCOMO applies to three classes of software projects:

- *Organic projects*: "small" teams with "good" experience working with "less than rigid" requirements
- *Semi-detached projects*: "medium" teams with mixed experience working with a mix of rigid and less than rigid requirements
- *Embedded projects*: developed within a set of "tight" constraints. It is also combination of organic and semi-detached projects.

The basic COCOMO equations take the form:

- Effort applied=a*(KLOC)^{b} [man-months]
- Development time=c*(effort applied)^{d} [months]
- People required = $\frac{\text{Effort applied}}{\text{Development time}}$ [count]

where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project.

The co-efficient a, b, c, d is given in the following table:

Software Project	A	B	C	D
Organic	**2.4**	**1.05**	**2.5**	**0.38**
Semi-detached	**3.0**	**1.12**	**2.5**	**0.35**
Embedded	**3.6**	**1.20**	**2.5**	**0.32**

Table 3.1: COCOMO Model

3.3.2 Estimates of Effort, Cost, Duration:

- **$E = a*(KLOC)^{b}$**
  $= 3.0*(6.5)^{1.12}$
  $= 24.411$

- **$D = c*(E)^{d}$**
  $= 2.5 *(24.411)^{0.35}$
  $= 7.65$

- **$P = E/D$**
  $=24.411/7.65$
  $=3.19$

Thus according to COCOMO model we get the following result:

Effort applied	24 man-months
Development time	7 months
People required	3

3.3.3 Risk Mitigation Strategy

RISK	CATEGORY	PROBABILITY	IMPACT
User does not understand the system	BU	Medium	2
Database is accessible to unauthorized user	TU	Low	1

Table 3.2: Risk Mitigation Strategy

Impact Values :

1-catastropic	2-critical
3-marginal	4-negligible

A project team begins by listing all risks in first column of table. Each risk is categorized in the second column (PS-Project Risk, DE-Development Risk, BU-Business Risk, and TE- Technical Risk). The probability of occurrence of each risk is entered in the next column of the table.

RMMM Plan for each risk :

Risk Information Sheet			
Project Name : Graphical Password Authentication			
Risk Id:- 001	Date :- 4/8/2014	Probability :- Medium	Impact :- Critical
Origin :- Avani Deora		Assigned To :- Parth Ved	
Description :- During registration the user might understand something incorrectly and then when he logs in the next time he maybe be choosing some wrong pattern which he feels is correct.			
Mitigation/Monitoring :- The user will be provided by step by step instructions during registration.			
Management :- Once the risk becomes active the user will have to reset the password.			
Status :- Still left to implement.			
Approval :- (Arjun Jaiswal)		Closing Date :-	

Table 3.3: Risk Sheet 1

Risk Information Sheet			
Project Name : Graphical Password Authentication			
Risk Id:- 002	Date :- 15/8/2013	Probability :- Low	Impact :-catastrophic
Origin :- Parth Ved		Assigned To :- Amish Shah	
Description :- Database is accessible to unauthorized individual. The database contains all the information about every user: user-id and images or the text passwords. Hence anyone who gets his hands on the database can log-in as any user he wishes to.			
Mitigation/Monitoring :- The developers should make sure that only the admin, or in some case only a authorized user is able to read or query the database.			
Management :- Once risk becomes active then we will ask the user to reset their passwords.			
Status :- Still left to implement.			
Approval :- (Arjun Jaiwal)		Closing Date :-	

Table 3.4: Risk Sheet 2

Chapter 4

Project Design

Project design includes an array of activities from generating ideas to planning how these ideas could become a realisable project. It basically represents a blueprint for the project.

4.1 System Architecture:

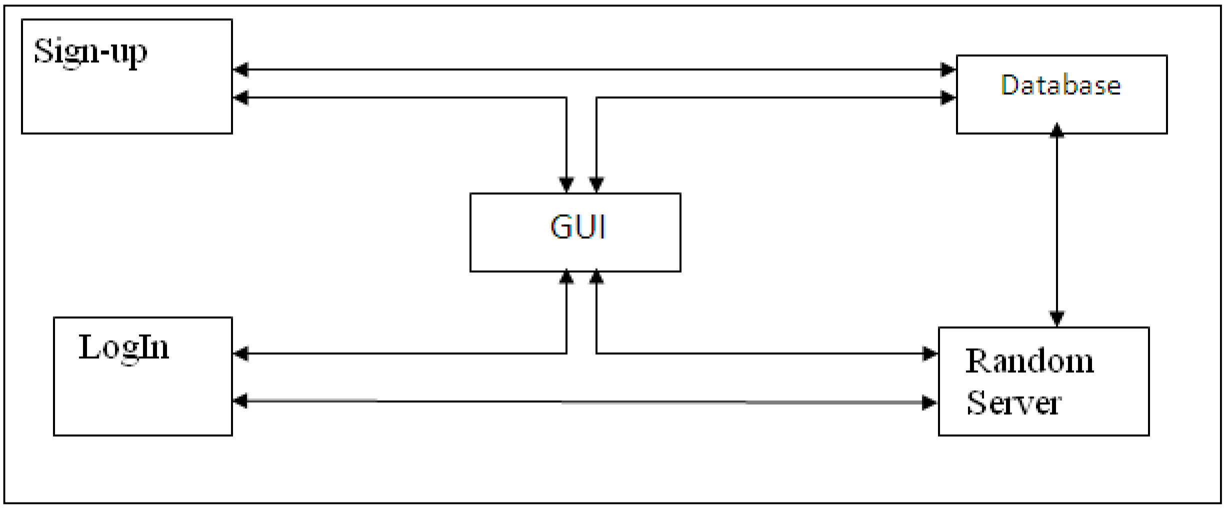


Figure 4.1: System Architecture

4.2 Modules Description:

Registration:

This module allows the user to register itself with the system. The username and the password of the user is passed to the database.

Authentication Technique 1: Movable frame scheme

This module is one of the two techniques for Authentication scheme. At the time of registration the user chooses three images and an alignment. At the time of login, the user has to align the three images in the alignment chosen by him/her initially by moving the rows of the grid. Once this is done, if the alignment and the pass images are correct, the user is successfully authenticated.

Authentication Technique 2: Text based graphical password

At the time of login the user selects a 8-15 character password along with a color. During login the circle should be rotated clockwise or anti-clockwise so as to get the character of the password and the color in the same sector. Every character has to be confirmed by the user and once all the characters are entered the user clicks on login.

Database:

It contains all the information about the user i.e. the username and the corresponding 3 passwords. For safety reasons the passwords are encrypted using MD5.

4.3 Design Diagrams

UML is de facto standard notation for software design. It can be used for drawing diagrams and also to generate codes, apply design patterns, mine requirements and perform impact analysis. UML is flexible and UML models are portable. UML is well known visual language that can capture much of the information that one needs to communicate about the system.

Use Case Diagram: A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

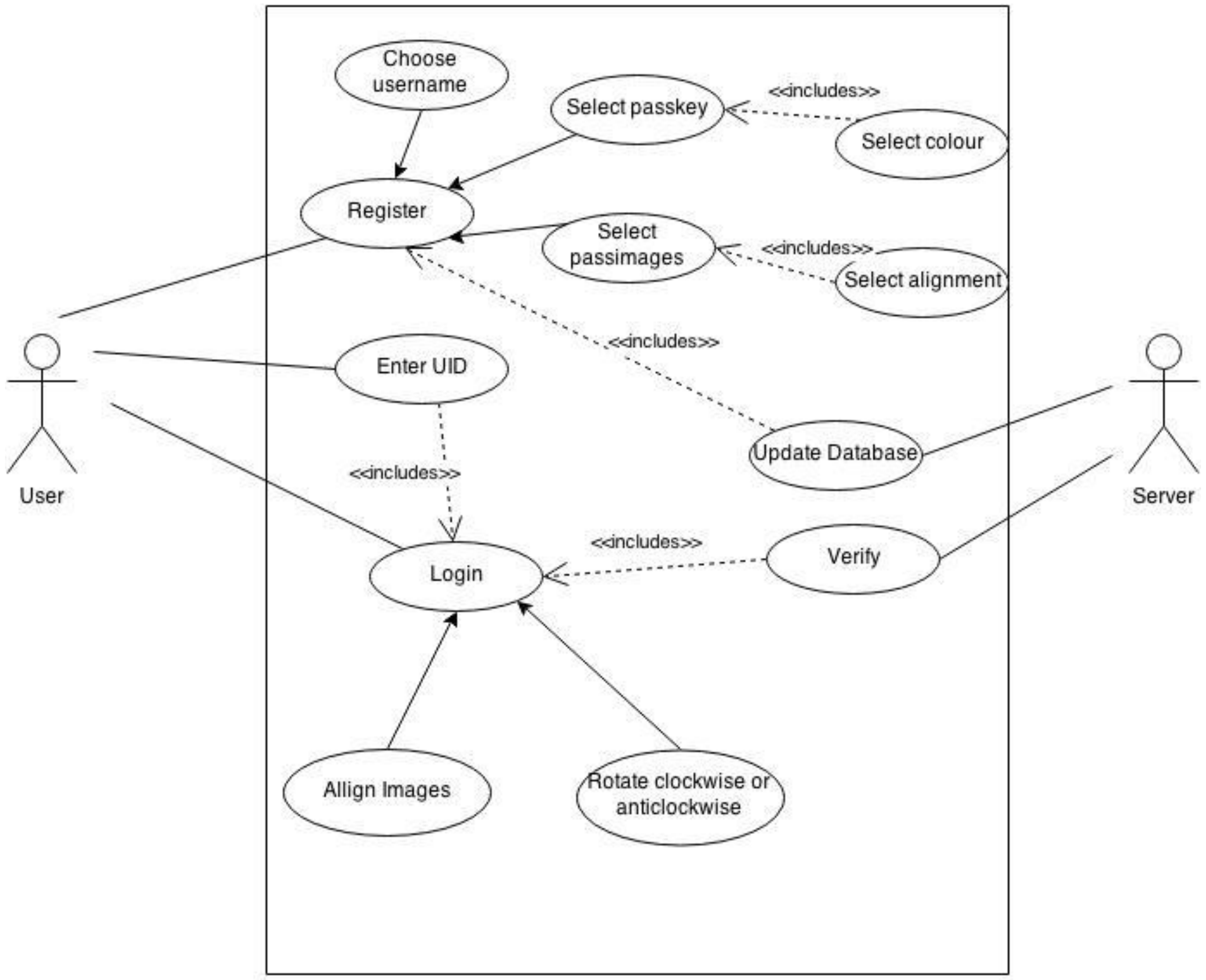


Fig 4.2: Use Case Diagram

Activity Diagram: Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organisational processes (i.e. workflows). Activity diagrams show the overall flow of control.

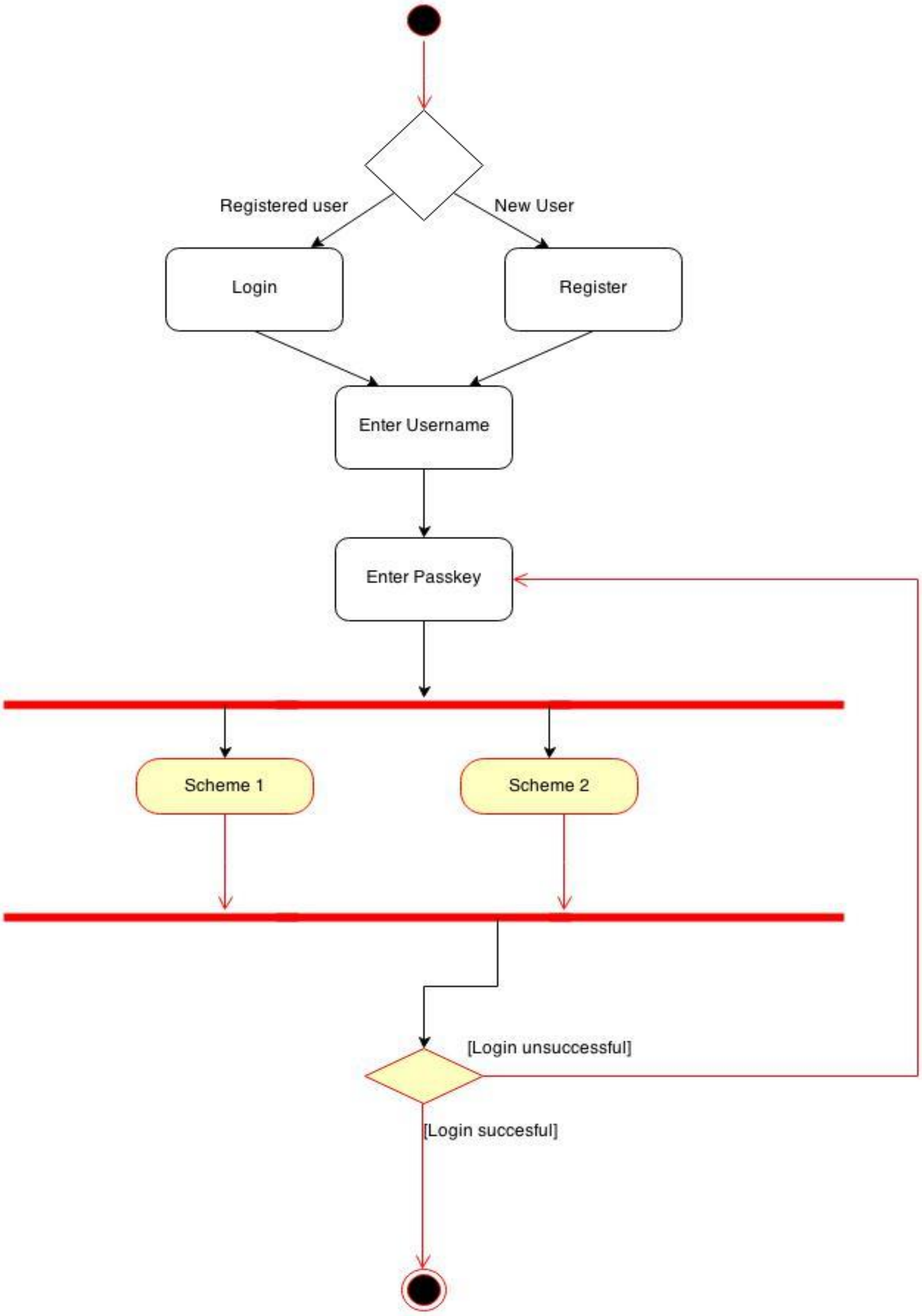


Fig 4.3: Activity Diagram

Component Diagram: Component diagrams are different in terms of nature and behaviour. Component diagrams are used to model physical aspects of a system. Now the question is what are these physical aspects? Physical aspects are the elements like executables, libraries, files, documents etc which resides in a node. So component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

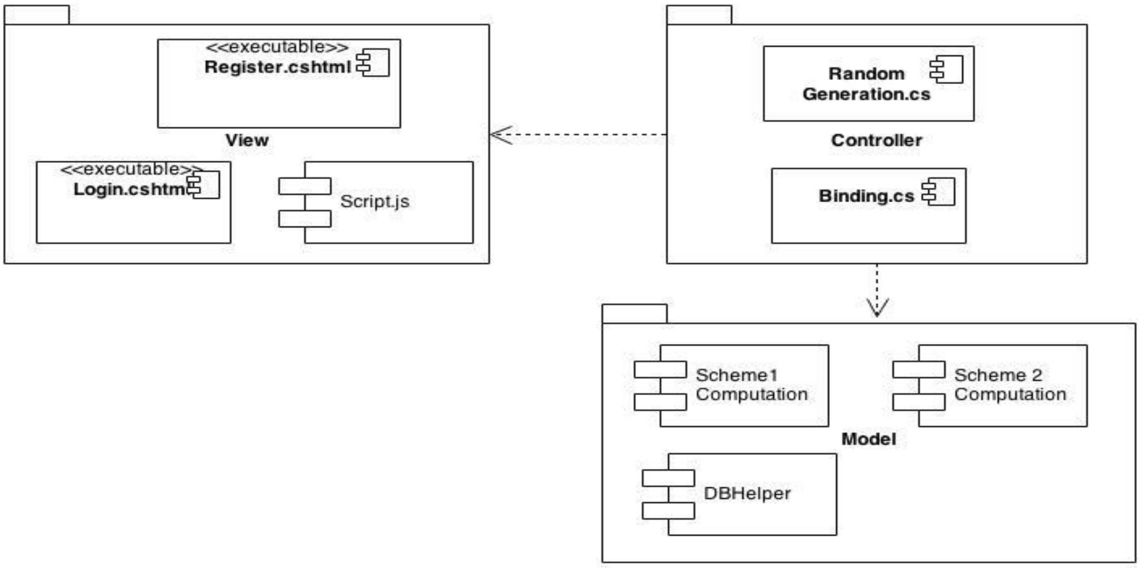


Fig 4.4: Component Diagram

Deployment Diagram: Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed.So deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

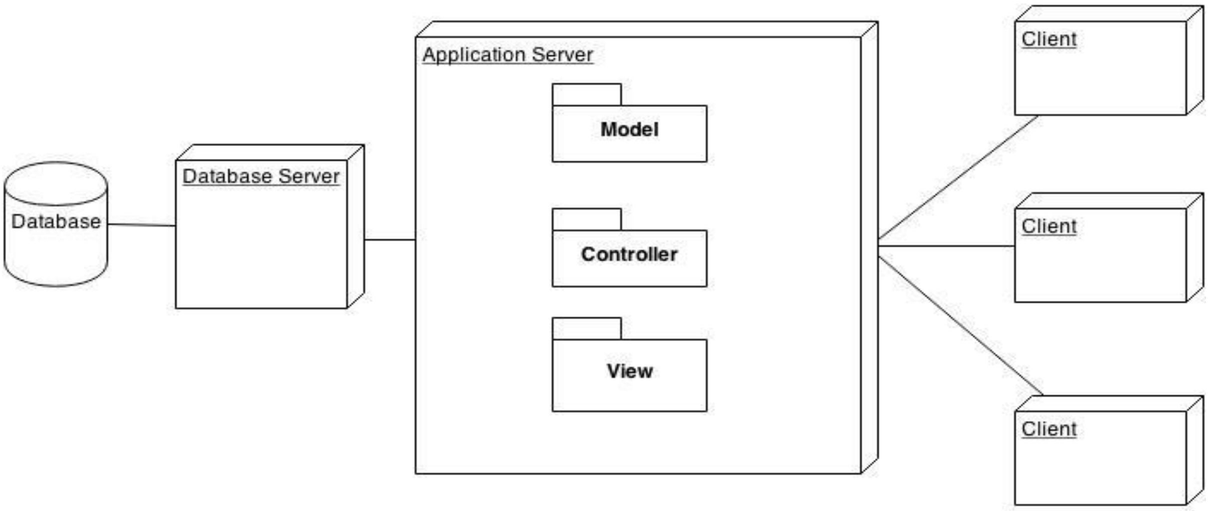


Fig 4.5: Deployment Diagram

Data Flow Diagram: A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

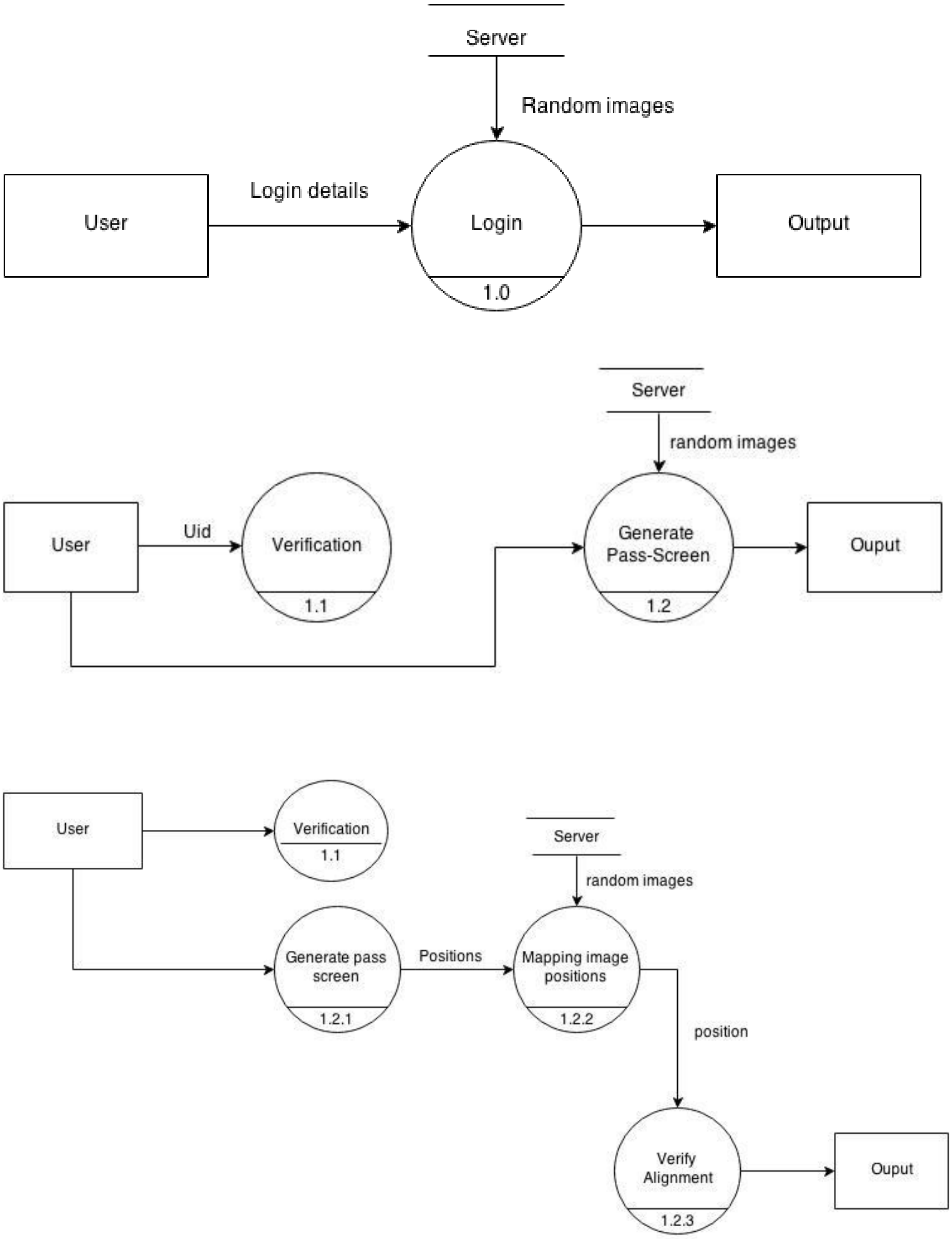


Fig 4.6: Data Flow Diagram

4.4 Data Design

ER Diagram

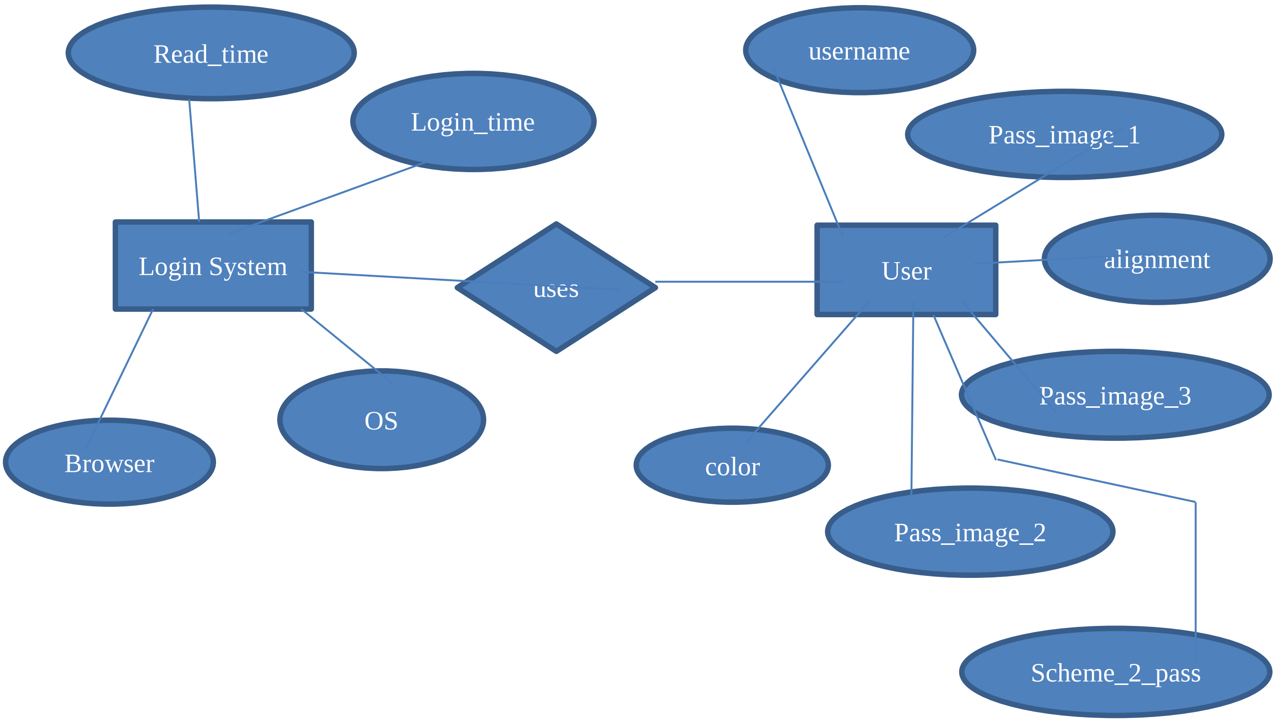
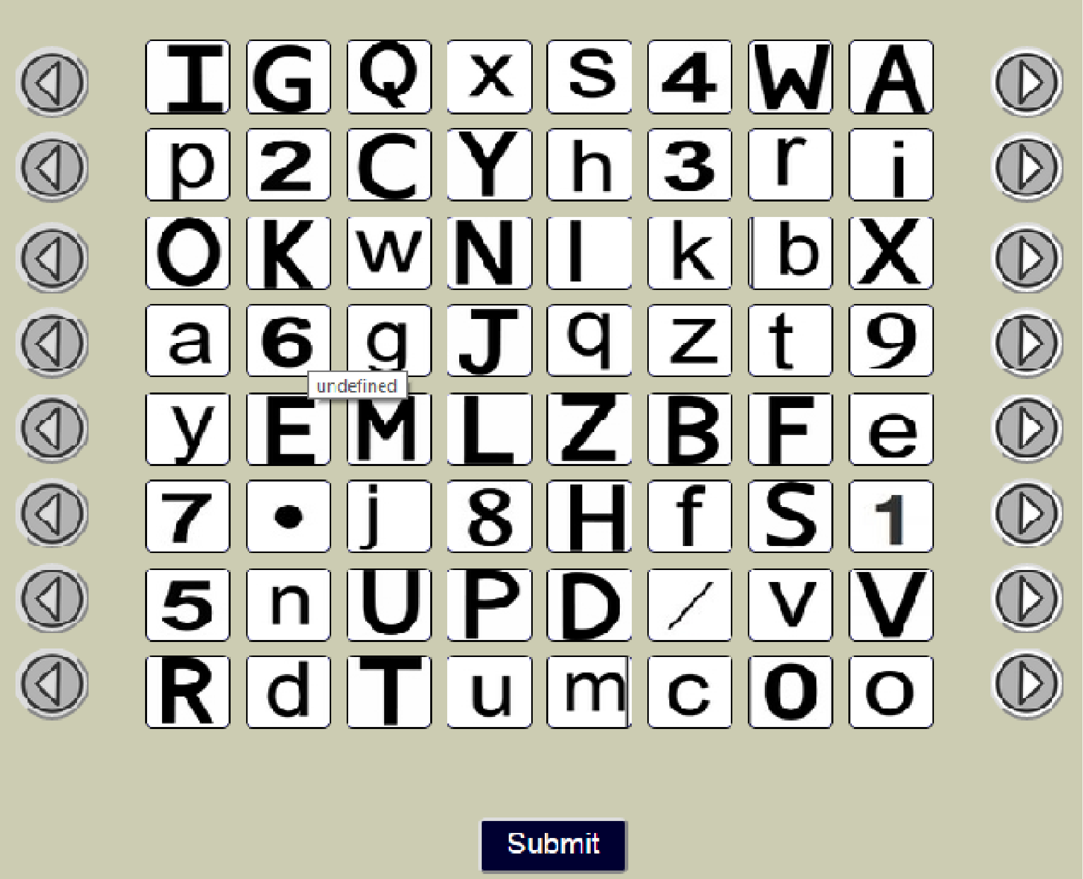


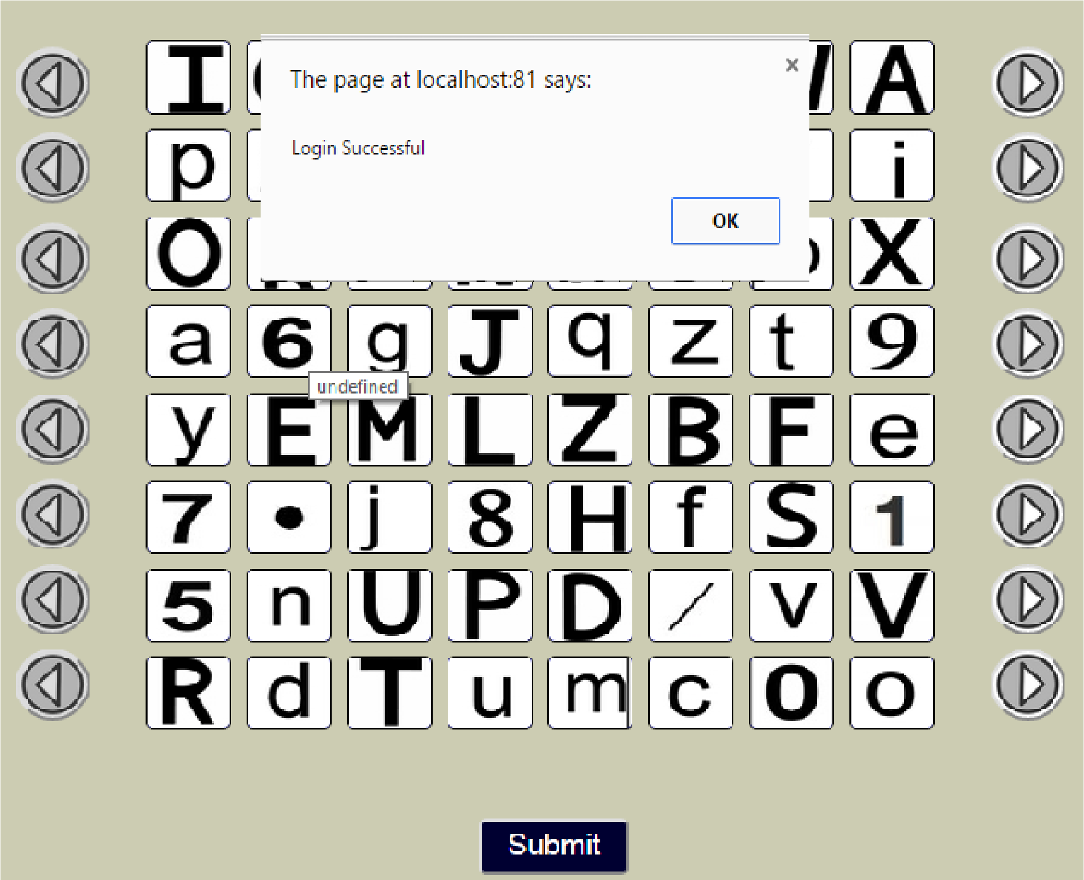
Figure 4.7: ER Diagram

4.5 User Interface Design

Scheme 1:



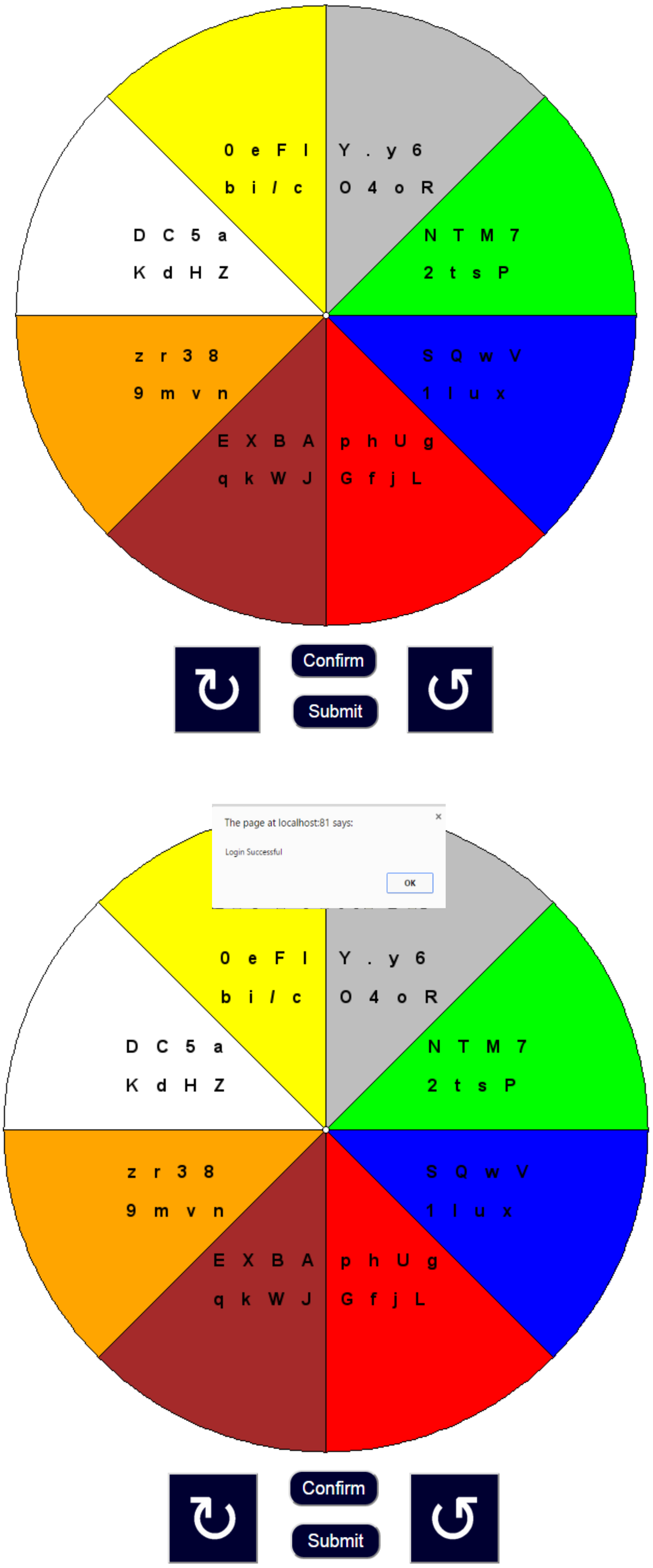
I	G	Q	x	S	4	W	A
p	2	C	Y	h	3	r	i
O	K	w	N	l	k	b	X
a	6	g	J	q	z	t	9
y	E	M	L	Z	B	F	e
7	•	j	8	H	f	S	1
5	n	U	P	D	/	v	V
R	d	T	u	m	c	O	o



The page at localhost:81 says:

Login Successful

Scheme 2:



The page at localhost:81 says:

Login Successful

Figure 4.8: User Interface design

Chapter 5

Implementation

5.1 Modules/ Component Description:

Scheme 1:

1. Retrieval of images

Images are to be arranged randomly on the login screen. Each image is assigned with IDs. So we shuffle the IDs and then retrieve images from the database. In this module, user's pass IDs are retrieved. Then these pass IDs are sent to random arrangement function that will allow the corresponding pass images to be placed in different rows. The images will be encoded and sent to the login screen.

2. Login UI

The images sent from the database are pushed into an array using jQuery. This is done dynamically so that they can be used as carousel. Trackers are implemented in this module so that it is possible to keep a track of state of these arrays of images. Multiple arrays of images form a matrix structure. So we kept a track of movement of images in the columns for each rows. Hence, once the user submits, new image positions are stacked upon in array and sent for authentication.

3. Authentication

Our self – designed algorithm is executed. This algorithm checks the registered pattern and runs the respective algorithm on the new image positions sent. The algorithm is robust and also handles the extreme cases well.

Scheme 2:

1. Piechart

A pieplot module was included which was apt for the graphical scheme we designed. We created a variance with this pie plot module. In this module, we incorporated randomly arranged elements into the sectors of the pie graph. Also, form an array of colors such that they could be shifted clockwise or anticlockwise as defined by the user with clicks. At first when the Pie Graph is formed it is loaded from an image. When colored-sectors are rotated a new image is formed and is overwritten on the previously formed image. Since the process happens before the blink of an eye one can see that the pie chart is rotates.

2. Login UI

The above Piechart module is the heart of the login UI. This Piechart is then bind with the controls on the login page using jQuery. At each confirm, the characters present in the user registered color within the sector are sent to server. This registered color is obtained from the database. Also, the user is acknowledged that a password is being entered by displaying a textbox that is filled up as he/she hits ‘confirm’.

3. Authentication

The characters sent from the login page as an array are all stored into a single array. Then once the user hits the login button our authentication algorithm is executed over this array.

5.2 Module-wise Algorithm:

Scheme 1

1. Login page is generated

a. As per the username, pass image IDs are retrieved from the database.

b. Since there are only horizontal translations possible, there can be only 1 pass image in each row. So a recursive function is called to check the position of pass image IDs in the array. If 2 or more IDs are present in the same array than random_arrangement function is called again.

2. After the user performs translations to arrange images, the state of image array is send to authentication file.

3. The registered pattern of the user is retrieved. The pass images are checked whether they are arranged as per the alignment.

Scheme 2

1. Login page is generated.

2. All the characters that fall into the registered colour are sent to the authentication file

3. Authentication:

a. First count the number of arrays that contain characters to the server sent. If it matches the length of the password go further.

b. Slice the password and look for each character in the retrieved array. If each retrieved array contain corresponding pass-word character than allow access.

Chapter 6

Results and discussions

Testing is the process of validating and verifying that an application:

- Meets the requirements that guided its design and development,
- Works as expected,
- Can be implemented with the same characteristics,
- Satisfies the needs of stakeholders.

6.1 Test plan

A test plan is a systematic approach to testing a system. The test plan approach that has been used in our project includes the following:

1. Design verification or Compliance test: This testing was performed during the development and approval stage of the product on each of the module.

2. Acceptance or Commissioning test: This testing was performed at the time of delivery and installation of the software.

3. Test Coverage: The design verification tests were performed at the point of reaching every milestone. Test areas included testing of various modules such as highlight, add question, update keywords, statistics calculation, graph generation etc.

4. Test Methods: For each module, corresponding outputs were checked in the view of Browser Output Console. For testing highlight module, the output produced from running the PHP code was checked with the test data set. The username and the password were verified with the database.

5. Test Responsibility: Team members working on their respective modules performed the testing of those modules. Test responsibilities also included, the collection of data, and decision on how that data should be stored, used and reported.

6.2 Test cases

A test case is a set of conditions or variables under which we will determine whether the Smart Snap application is working correctly or not. We have used many test cases to determine that the system is sufficiently scrutinized.

Test Case ID	Case Description	Expected Result	Actual Result	Pass/Fail
1	On login screen of scheme1, check whether any of the pass images are not on the same row.	Pass-images should never appear on the same row.	Pass-images never appears on the same row.	Pass
2	For scheme1, check the authentication algorithm such that the system allows access for correct arrangement	If the arrangement is correct, the system should allow user to login.	If the arrangement is correct, the system should allow user to login.	Pass
3	For scheme1, check the authentication algorithm such that the system does not allow access for incorrect arrangement.	If the arrangement is incorrect, the system should not allow user to login.	If the arrangement is incorrect, the system notifies that login is unsuccessful.	Pass
4	For scheme1 UI, check rotation of images on click of left or right arrow buttons.	On a single click of left button, images for that respective frame must move towards left and the first image to the left becomes the last image in that frame and vice versa on click of right arrow button.	Images translate properly as expected like carousel.	Pass

5	For scheme 2 UI, check whether colored-sectors rotate on click of clockwise and anticlockwise buttons.	Clicking on anticlockwise button must rotate the sectors anticlockwise and vice-versa for clockwise buttons.	Sectors are rotating clockwise and anticlockwise by clicking clockwise and anticlockwise buttons.	Pass
6	For scheme 2, check whether the characters present in the sector (that is colored with the color registered) are sent to server.	Server must receive array of characters that belong to correct sector.	Server receives an array of characters that fall into the correct sector.	Pass
7	For scheme 2, check the number of the arrays of characters sent by the user at the time of login.	User should not be allowed to access if the length of the password is different from the number of arrays received.	User is not allowed access if the length of the password is different from the number of arrays.	Pass
8	For scheme 2, check whether the user logins in if the password entered is correct.	The authentication algorithm must work well and the user must be allowed access.	If the order in which the character arrays are sent is correct, then the user is allowed access	Pass

Table 6.1: Test Case

6.3 Methods Used:

We used the following methods for testing:

Unit Testing: Unit testing is a method by which individual units of source code, sets of one or more program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use. In our tool, we considered each module as one unit and tested these units with help of test cases and test plan developed. Unit testing was carried out on each module and on every function within the module. Output of each unit was assessed for accuracy and if found incorrect, appropriate corrections were made.

Integration Testing: Integration testing is the phase in software testing in which individual software modules are combined and tested as a group .The modules of our tool were integrated together in order to verify that they provide the required functionalities appropriately. The various modules were tested together to check for their accuracy and compatibility.

System Testing: System testing of software is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. In this testing, we tested the system as whole to ensure that it provides the appropriate output as stated in the requirements. Overall performance of the system was also tested simultaneously.

Validation Testing: Validation testing is the assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. Here, we gave the system various possible inputs that the user might give to the system and tested if it provides correct and expected outputs. In case of any deviation from the expected output, corrective action was taken.

Chapter 7

Maintenance

7.1 User Manual

Prerequisites

- The user must have a working pc with a browser installed.
- The user must have an active data plan or any sort of internet connectivity

How to use

- The user must open the browser.
- A new user will first have to register.
- For registration, in the movable frame scheme the user has to select three pass images and an alignment among vertical, diagonal (45) or diagonal (135).
- During login the user has to use the arrow keys on the right and left of the grid to move the rows of the grid as to align the pass images selected initially in the alignment chosen during login.
- Once the correct pass images are aligned in the required pattern, the user has to click on submit so as to authenticate himself.
- For the text based authentication scheme, at the time of registration the user has to choose a 8-15 character password using any combination of 64 characters (26 capital alphabets, 26 small alphabets, 10 digits and . and /). There will be 8 color options available to him to choose from. The user also has to select a color during registration and remember it for any further login.
- During login in this scheme, the user will be given a circle with 8 sectors of different colors. The 64 characters will be randomly scattered in the 8 sectors of the circle. Every sector will be having a different color.
- The user can rotate the circle clockwise or anti clockwise. With every rotation, the color moves either clockwise or anti clockwise. The user has to get each character of the password in the color chosen initially and click confirm. Once all the characters are entered, click submit.

7.2 Constraints

- The user must have internet connectivity.

Chapter 8

Conclusion and Future Scope

In this project, two authentication techniques based on text and colors are proposed. These techniques are resistant to dictionary attack, brute force attack and shoulder-surfing.

To make the system even more usable the following developments can be made:

In Scheme 1, vertical translations can also be implemented along with the horizontal rotations so as to increase the difficulty in guessing the password by the peeking attacks.

In Scheme 1, better patterns to be used to arrange images so that recognition gets easier for the user and eventually reduce the login time. Both the schemes can be up-scaled to make it more resistant to shoulder surfing attacks.

These schemes are completely new to the users and the proposed authentication techniques should be verified extensively for usability and effectiveness. These techniques can also be developed as ATM application where the person standing outside the transparent door could guess the ATM pin with the same Debit card or can be used against the key loggers in cyber café or windows application such as a folder locker or an external gateway authentication to connect the application to a database or an external embedded device.

References and Bibliography

A

Applications of Information Security Technology, Dec.2010, pp. 204-210.

B

B. R. Cheng, W. C. Ku, and W. P. Chen, “An efficient login-recording attack resistant graphical password scheme －SectorLogin,” Proc. of 2010 Conf. on Innovative.

Sobrado, L and Birget, J. “Graphical Passwords”, The Rutgers Scholar, Ruthgers University, New Jersey, Vol.4, ’04

Shoulder Shuffling Free Graphical Locker for Android Graphical Pattern Lock with Text Support for Android Devices (International Journal of Advanced Research in Computer Science)

D

Dhamiga and Perig Method R. Dhamija and A. Perrig. ‘Déjà vu: A User Study Using Images for Authentication’, USENIX Security Symposium, 2000.

J

Jensen et al. Method, ‘Picture Password- A Visual Login Technique for Mobile Devices’, NISTt NISTIR 7030, 2003

M

M. Sreelatha, M. Shashi, M. Anirudh, Md. Sultan Ahamer, and V. Manoj Kumar. “Authentication schemes for session passwords using color and images,” *International Journal*of Network Security & Its Applications, vol. 3, no. 3, May2011.

R

Real User Corporation, PassfacesTM, www.realuser.com, Accessed on January’07.
“Recognition memory for words, sentences, and pictures”, *Journal of Verbal Learning and Verbal Behaviour*, vol. 6, 1967, pp. 156-163.
Recognition memory for words, sentences, and pictures”, *Journal of Verbal Learning and Verbal Behaviour*, vol. 6, 1967, pp. 156-163.

S

S. H. Kim, J. W. Kim, S. Y. Kim, and H.G. Cho. “A new shoulder-surfing resistant password for mobile environments,” Proc. of 5th Int. Conf. on Ubiquitous Information Management and Communication, Feb. 2011.

Z

Z. Imran and R. Nizami, “Advance secure login,”International Journal of Scientific and Research *Publications*, vol. 1, Dec. 2011. M. K. Rao and S. Yalamanchili. “Novel shoulder-surfing resistant authentication schemes using text-graphicalpasswords,” International Journal of Information & *Network Security*, vol. 1, no. 3, pp. 163-170, Aug. 2012 . Network Working Group of the IETF, “The Secure Sockets Layer (SSL) Protocol Version 3.0,” *RFC 6101*,2011. Network Working Group of the IETF, “The Transport Layer Security (TLS) Protocol Version 1.2,” *RFC 5246*, 2008.

Links:
1.http://clam.rutgers.edu/~birget/grPssw/
2. http://draw.io/
3.http:// www.smartsheet.com