"Graphical Password Authentication implemented in Web based System"

A Project Report Submitted to
Rajiv Gandhi Proudyogiki Vishwavidyalaya

Towards Partial Fulfillment for the Award of
Bachelor of Technology in *Computer Science & Engineering*

Submitted by:

Abdul Rehman(0827CS201008)
Abhishek Sharma(0827CS201012)
Ankita Arya(0827CS201034)
Alokit Sharma(0827CS201023)

Guided by:

Prof. Priyanka Jangde

Acropolis Institute of Technology & Research, Indore
July – Dec 2022

EXAMINER APPROVAL

The Project entitled ***"Graphical Password Authentication using web based system"*** submitted by **Abdul Rehman (0827CS201008), Abhishek Sharma (0827CS201012), Ankita Arya (0827CS201034),Alokit Sharma(0827CS201023)** has been examined and is hereby approved towards partial fulfillment for the award of Bachelor of Technology degree in Computer Science & Engineering discipline, for which it has been submitted. It understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project only for the purpose for which it has been submitted.

(Internal Examiner)
Date:

(External Examiner)
Date:

GUIDE RECOMMENDATION

This is to certify that the work embodied in this project entitled **"Graphical Password Authentication for web based system"** submitted by **Abdul Rehman (0827CS201008), Abhishek Sharma (0827CS201012), Ankita Arya (0827CS201034),Alokit Sharma(0827CS201023)** is a satisfactory account of the bonafide work done under the supervision of **Prof. Priyanka Jangde and Prof. Narendra Pal Singh** are recommended towards partial fulfillment for the award of the Bachelor of Engineering (Computer Science & Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.

(Project Guide) **(Project Coordinator)**

STUDENTS UNDERTAKING

This is to certify that project entitled **"Graphical Password Authentication for web based system"** has developed by us under the supervision of **Prof. Priyanka Jangde** The whole responsibility of work done in this project is ours. The sole intension of this work is only for practical learning and research.

We further declare that to the best of our knowledge, this report does not contain any part of any work which has been submitted for the award of any degree either in this University or in any other University / Deemed University without proper citation and if the same work found then we are liable for explanation to this.

Abdul Rehman(0827CS201008)
Abhishek Sharma(0827CS201012)
Ankita Arya(0827CS201034)
Alokit Sharma(0827CS201023)

Acknowledgement

We thank the almighty Lord for giving me the strength and courage to sail out through the tough and reach on shore safely.

There are number of people without whom this projects work would not have been feasible. Their high academic standards and personal integrity provided me with continuous guidance and support.

We owe a debt of sincere gratitude, deep sense of reverence and respect to our guide and mentors **Prof. Priyanka Jangde**, Associate Professor, AITR, for their motivation, sagacious guidance, constant encouragement, vigilant supervision and valuable critical appreciation throughout this project work, which helped us to successfully complete the project on time.

We express profound gratitude and heartfelt thanks to **Dr Kamal Kumar Sethi**, HOD CSE, AITR Indore for his support, suggestion and inspiration for carrying out this project. I am very much thankful to other faculty and staff members of CSE Dept, AITR Indore for providing me all support, help and advice during the project. We would be failing in our duty if do not acknowledge the support and guidance received from **Dr S C Sharma**, Director, AITR, Indore whenever needed. We take opportunity to convey my regards to the management of Acropolis Institute, Indore for extending academic and administrative support and providing me all necessary facilities for project to achieve our objectives.

We are grateful to **our parent** and **family members** who have always loved and supported us unconditionally. To all of them, we want to say, “Thank you”, for being the best family that one could ever have and without whom none of this would have been possible.

Abdul Rehman (0827CS201008), Abhishek Sharma (0827CS201012), Ankita Arya (0827CS201034),Alokit Sharma (0827CS201023)

Executive Summary

Graphical Password Authentication implemented in web based system

This project is submitted to Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal(MP), India for partial fulfillment of Bachelor of Engineering in Computer Science & Engineering branch under the sagacious guidance and vigilant supervision of ***Prof. Priyanka Jangde***.

The project is based on Hashing, which is the process of transforming any given key or a string of characters into another value. In the project,ejs,Javascript is used for structuring and designing,Node js is used for connecting frontend to backend,MongoDB and MySql are used for the database The purpose of this project is to create a website which makes a question paper from submitted questions

Key words : Graphical passwords,hashing

“Success is the sum of small efforts,repeated day in and day out” - Robert collier

List of Abbreviations

Abbr1: GPAS:Graphical Password Authentication System

Abbr2: POI:Points of Interest

Abbr3: HTML:Hypertext Markup Language

Abbr4: CSS:Cascading Style Sheets

Abbr5: OTP:One Time Password

Abbr6: NPM:Node Package Manager

Abbr7: HTTP:Hyper Text Transfer Protocol

Abbr8: HTML:Hyper Text Markup Language

Table of Contents

Chapter 1

Introduction

Authentication is the process of determining that the person requesting a resource is the right person. Most of the authentication systems nowadays use an integration of username and password. The problem with the password is that it requires the user to remember it and it should be kept secret. Each authentication system has its own guidelines and limitations like password length, password must contain alphanumeric and special characters. These passwords are mostly text-based passwords. Either user use passwords that are easy to remember like license plate number, parent name, phone number, sometimes their own name which are very much predictable or complex passwords which they overlook so they might use the same password for different accounts or they jot down their password somewhere. Moreover, users are vulnerable to various attacks. Text-based passwords faces from security and usability matters.

1.1 Overview

Graphical passwords consist of choosing images or drawing symbols rather than entering textual characters. It was first described by Greg Blonder in 1996. Human brain is capable of processing and storing large volumes of graphical information with easiness. While it is very tough to recall a string of fifty characters, humans are capable of easily recalling faces of people, places we visited, and things. These graphical records characterize millions of bytes of facts and thus make them available to big password spaces. Thus, graphical password schemes deliver a way of creating more human-friendly passwords while growing the level of security.

1.2 Background and Motivation

Today's world of networked computing can be a frightening and dangerous place with attackers, hackers, crackers, scammers, and spammers at work. Computer security, which was relatively simple in prenetwork days, is now a major and expensive problem for organizations and individuals. Constant attention to security is needed to protect against damage or theft of one's electronic assets.

According to a recent computer world news article, the security team at a large company ran a network password cracker and within 30 seconds, they identified about 80% of the passwords. On the other hand, passwords that are difficult to guess or break are often difficult to remember. Studies showed that since user can only remember a limited number of

passwords, they tend to write them down or will use the same passwords for different accounts. To address the problems with traditional username password authentication, alternative authentication methods, such as biometrics have been used. In this paper, however, we will focus on another alternative: using pictures as password

1.3 Problem Statement and Objectives

Background: Passwords are ubiquitous today on any platform, on possibly any website. But to remember so difficult passwords and that too on numerous websites seems daunting and therefore you can devise a project illustrating graphical password strategy. This will allow the user to set passwords in the form of graphical presentation in a certain pattern and later use that pattern to login to the system. Summary: Remembering numerous passwords from various different sites can be difficult for a user. So to provide some flexibility we can provide users a graphical password authentication system where instead of creating a password a user has to select graphical objects in a particular order to keep it as their password. Objective: : In this method, the user is required to select some images (let's say different chocolates) in a specific pattern (for example dairy milk is followed by 5 stars which is in turn followed by KitKat and so on). Next time the user tries to log in, the images would have been shuffled, but the user will be required to follow the same pattern which was used initially. Every time the user will have to use the same sequence while the images are placed in different ways. This type of authentication is difficult to break since neither brute force nor dictionary attacks could breach it. We need techniques that can be easily implemented and provide better results to this process.

1.4 Scope of the Project

The scope for this project is identified which to make the web system process easier. This project concentrates more on the security of the system. i) Scope of User - Enter username, password, email during registration and login phase. - Select an image during registration phase and login phase. - Click five points during registration phase and login phase. ii) Scope of System - Sign up – the authentication system lets the user select pictures and click points in a correct number of clicks. - Log in – check either the user username, password, image and clicked points are valid and exist in the data store.

1.5 Team Organization

Abhishek Sharma:

I investigated and found the right technology and studied deep about it. I worked at the backend of the project. For implementation i used nodeJS and expressJS. I worked on the algorithms of our project. I also helped in the integration of the frontend with backend.

Abdul Rehman:

I studied about ejs,nodejs and javascript deeply.I made connections between javascript node js and mysql,and worked on nodejs scripts in the backend.I did all the documentation work of this project.

Ankita Arya:

I worked mainly on the backend of the project.I studied mysql and created the database of the project using it.I worked on making various tables of the database.I researched the existing system and found the flaws and worked on them.I helped in presentation and project reports.

Alokit Sharma:

I studied ejs,nodejs,HTML,JS,CSS deeply. I worked mainly on the frontend of the project designing login page, forgot password page.I also helped in connecting the frontend to backend using javascript

1.6 Report Structure

The project Graphical Password Authentication for web based system is primarily concerned with examination assessment and the whole project report is categorized into five chapters.

Chapter 1: Introduction- introduces the background of the problem followed by rationale for the project undertaken. The chapter describes the objectives, scope and applications of the project. Further, the chapter gives the details of team members and their contribution in development of the project which is then subsequently ended with a report outline.

Chapter 2: Review of Literature- explores the work done in the area of Project undertaken and discusses the limitations of the existing system and highlights the issues and challenges of the project area. The chapter finally ends up with the requirement identification for present project work based on findings drawn from reviewed literature and end user interactions.

Chapter 3: Proposed System - starts with the project proposal based on requirement identified, followed by benefits of the project. The chapter also illustrates the software engineering paradigm used along with different design representations. The chapter also includes details of major modules of the project. Chapter also gives insights of different types

of feasibility study carried out for the project undertaken. Later it gives details of the different deployment requirements for the developed project.

Chapter 4: Implementation - includes the details of different Technology/ Techniques/ Tools/ Programming Languages used in developing the Project.

The chapter also includes the different user interfaces designed in the project along with their functionality. Further it discusses the experiment results along with testing of the project. The chapter ends with evaluation of the project on different parameters like accuracy and efficiency.

Chapter 5: Conclusion - Concludes with objective wise analysis of results and limitation of present work which is then followed by suggestions and recommendations for further improvement.

Chapter 2

Review of Literature

2.1 Preliminary Investigation

Passwords have been used since ancient times by the people all around the world as a means for security.It has obtained many changes since then and it was adapted in computers and it is evolving now too,and graphical Password authentication Has been Introduced recently and it is developing too.

2.1.1 Current System

The current system of GPAS is dependent on different types of authentication method such as color authentication and image authentication were used while making the system.Malicious attacks such as brute force attacks and dictionary attacks were hard to use on these system were hard to use on them.But these system are very large and hard to make,passwords were also very large.

This system is having the problems of shoulder surfing attacks in which a hacker can easily see the password of the user.The hacker can also hack the system and the database as the password storage is not secure in these systems .Better systems for storing and transmitting these passwords are required as a extra layer of security.

2.2 Limitations of Current System

The limitations of these are as follows :

1. The system has problems dealing with shoulder surfing attacks.
2. Inputting data by using a keyboard is often inconvenient for systems using graphical passwords.
3. Passwords can be shared, guessed or stolen, which means they aren't secure.
4. Shoulder navigation attacks are effective for stealing passwords.
5. It is hard to login in multilayered security.

2.3 Requirement Identification and Analysis for Project

Significant work has been done in the field of Graphical Password authentication ; however, it is not easy to achieve desired results. The review of literature leads to draw certain major findings which are as under :

In this research paper since the password space is very large, it offers security against brute force attack. It’s easy to use. Passwords can be easily created and recall.Shoulder navigation attack is subject to safety precaution. [1]

In the proposed system, a user freely chooses a picture, POIs and corresponding words. The order and number of POIs can be enforced for stronger authentication. Together, these parameters allow for a very large password space. [2]

Although the main use for graphical passwords is that people are better at memorizing graphical passwords than text-based passwords, the existing user studies are very limited and there is not yet convincing evidence to support this argument. Our preliminary analysis suggests that it is more difficult to break graphical passwords using the traditional attack methods such as brute force search, dictionary attack, or spyware.[3]

This paper introduced various algorithms for graphical password authentication and showed different ways in which our project can be made.this gave gave 12 different algorithms to create graphical password authentication.[4]

This Paper introduced the amazing idea of pointer images to prevent shoulder surfing attacks by using shoulder surfing resistant shields which may be adapted in
future as a major authentication system.[5]

This Paper Showed different ways of implementing our project with their specifications such as advantages and disadvantages for each of them.[6]

In this paper it is an extended Blonder's idea by eliminating the predefined boundaries and allowing arbitrary images to be used. The image could be any natural picture or painting then it contains several possible clicks points. As a result, a user can picture or painting then it contains several possible clicks points.[7]

According to the paper,Graphical authentication may offer greater resistance to guessing and capture attacks but there are other attacks against graphical authentication including brute force attacks, intercepted communication and spyware which might be threats to the security breach. Authentication mechanism that is often being used is the combination of usernames and passwords which is based on textual-based password. Nevertheless, this traditional approach has shown 13 disadvantages. The significant consequences of the approach are the user might choose simple password for authentication process or the user can create a strong password however it is hard to be remembered by the user itself.[8]

Sobrado and Birget developed a graphical password technique that deals with the shoulder-surfing problem. In the first scheme, the system will display a number of pass-objects (pre-selected by user) among many other objects. To be authenticated, a user needs to recognize pass-objects and click inside the convex hull formed by all the pass-objects.[9]

2.3.1 Conclusion

This chapter reviews the literature surveys that have been done during the research work. The related work that has been proposed by many researchers has been discussed. The research papers related to question banks which discussed about different methods to create a question bank.

Chapter 3

Proposed System

3.1 The Proposal

We have built a powerful Graphical Password authentication system. The security methods of this system are very good. This system increases the security of the system.In this proposed project, graphical password authentication by using passpoints scheme can give many benefits to users in many aspects. It will secure the users to make an authentication process in spite of the in fact it takes users longer time to access into a system.

3.2 Benefits of the Proposed System

The current system had a lot of challenges that are overcome by this system :

1. This System uses color and image authentication along with password authentication.
2. It also provides multi-factor authentication in a friendly intuitive system.
3. Graphical passwords schemes provide a way of making more human friendly passwords.
4. Here the security of the system is very high. Dictionary attacks and brute force search are infeasible

3.3 Block Diagram

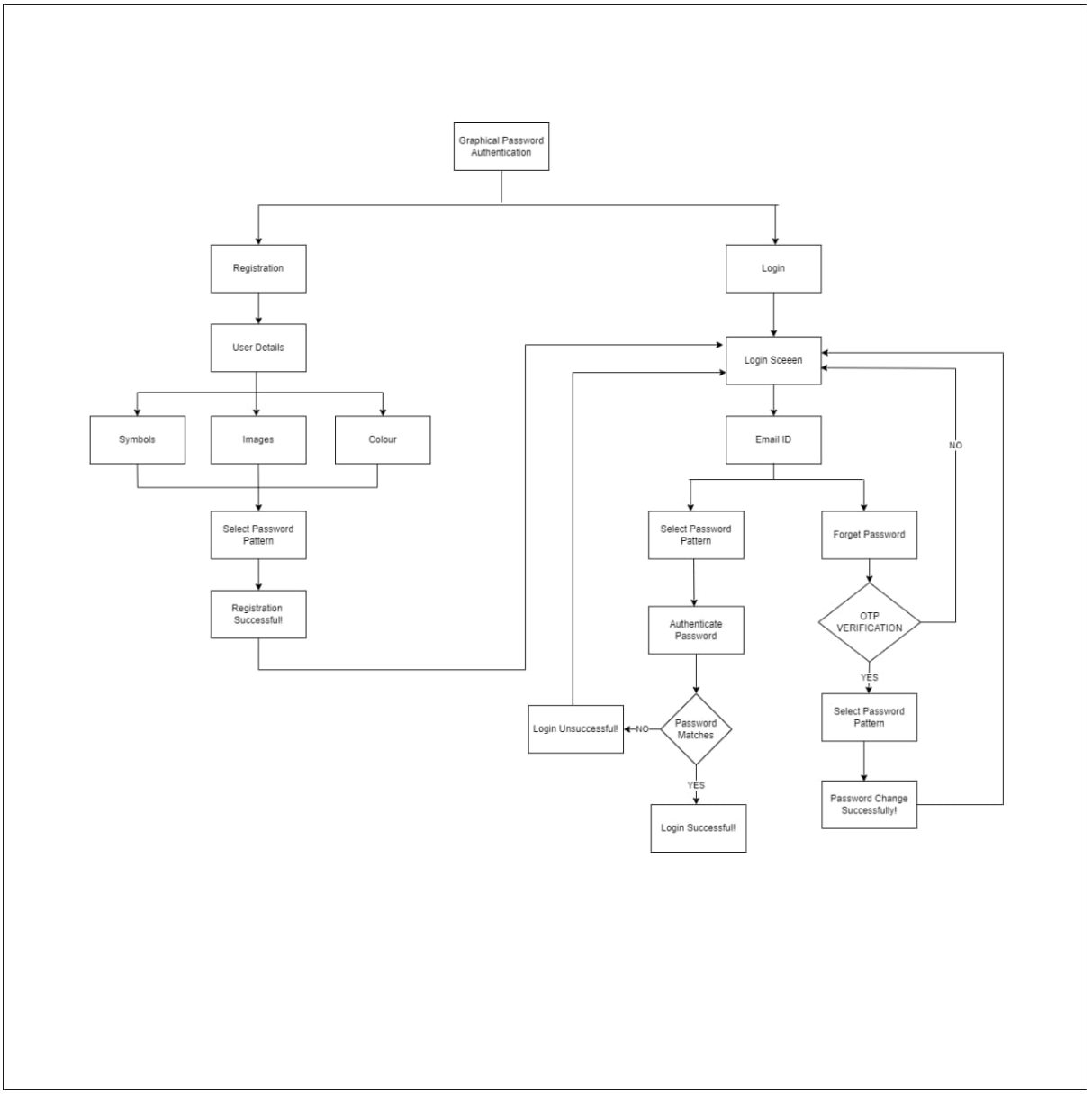

Figure 3.1 : Block Diagram

.

3.4 Feasibility Study

A feasibility study is an analysis of how successfully a system can be implemented, accounting for factors that affect it such as economic, technical and operational factors to determine its potential positive and negative outcomes before investing a considerable amount of time and money into it.

3.4.1 Technical

For our Examination Assessment system,a web browser supporting HTML 5 is required to run the site.HTML,CSS and JavaScript are used to create front end of the web page and node js,MySQL are used for the back end

3.4.2 Economical

For any graphical password authentication system, there is a need for a browser running device.

3.4.3 Operational

The main motto of our system is to increase the security of a system.

The system is able to do that accurately and efficiently making the system operationally feasible

3.5 Design Representation

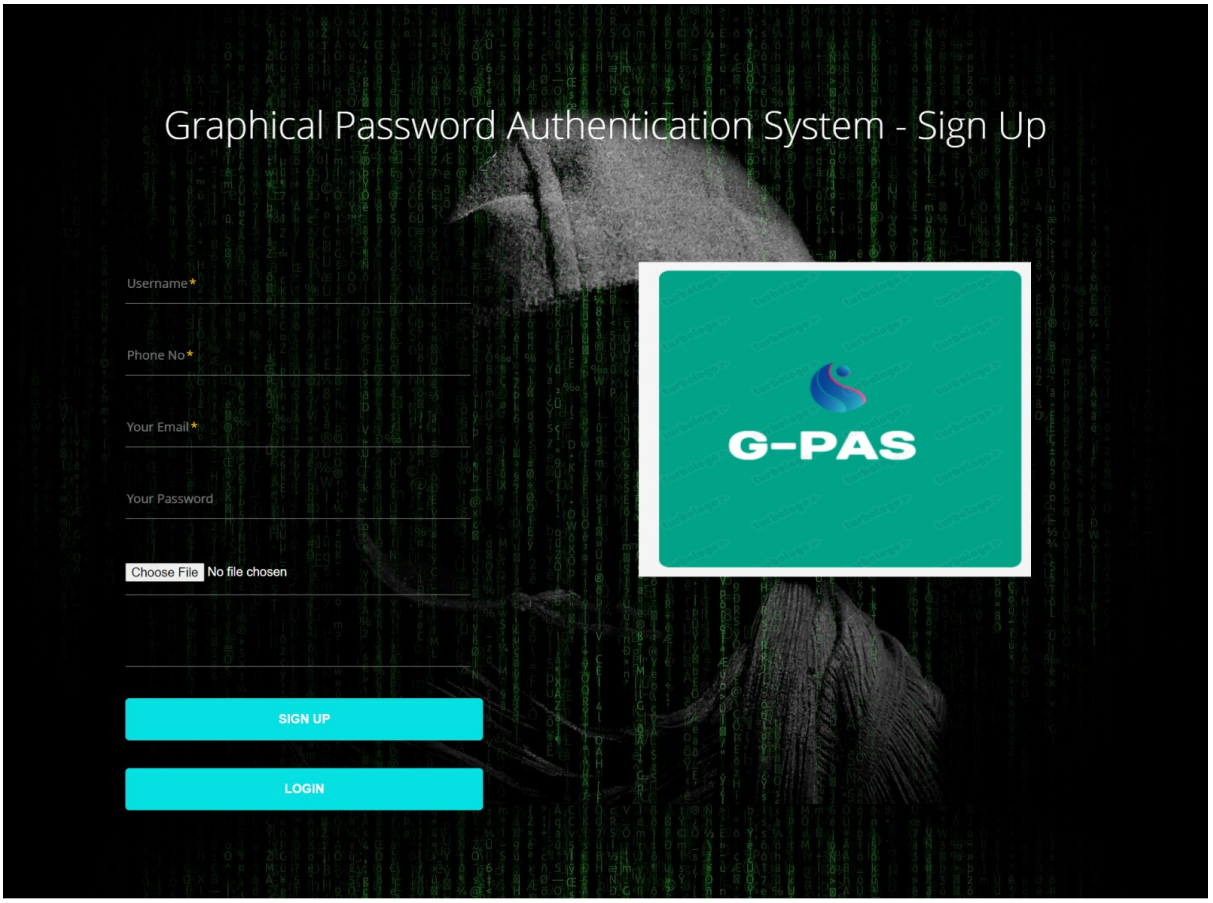

Figure 3.5.1 Sign up page

Figure 3.5.2 GPAS Login Page

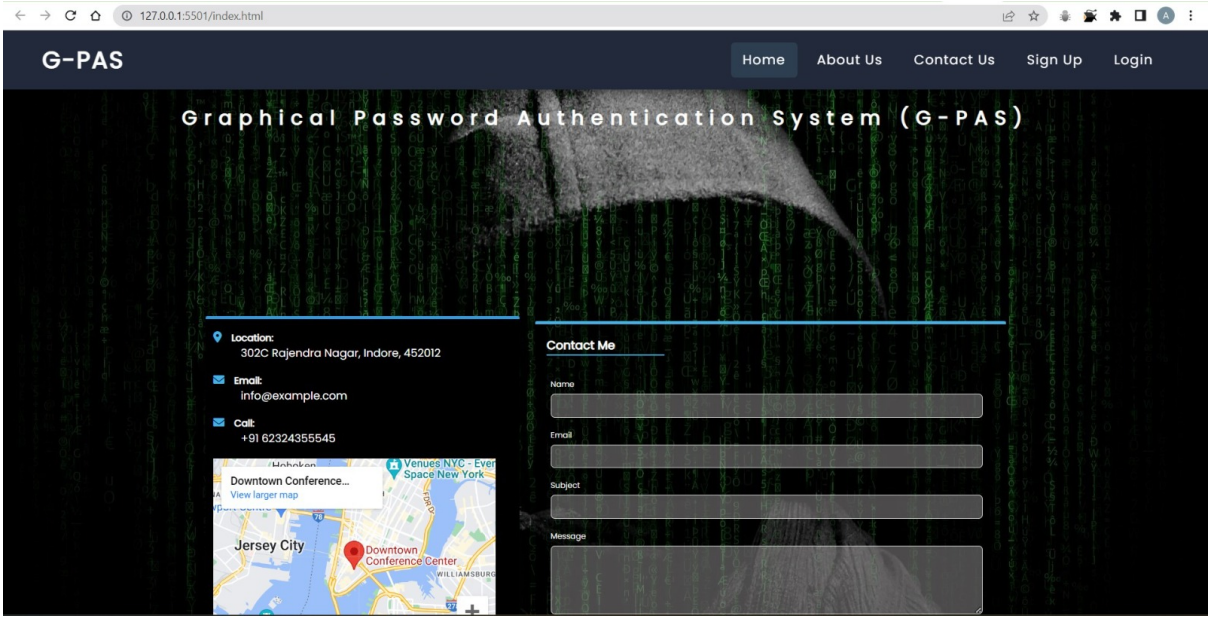

Figure 3.5.3 Contact Us page

3.5.1 Data Flow Diagrams

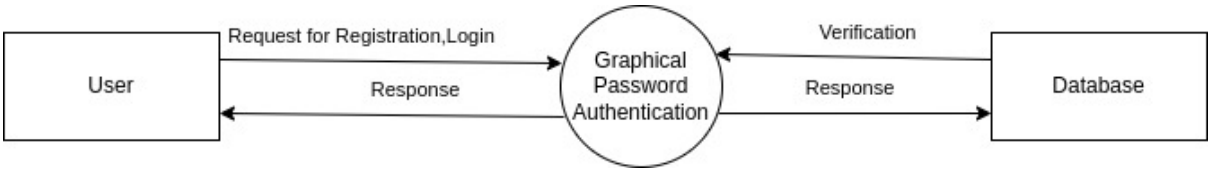

Figure 3.5.1 Data Flow Diagram Level 0

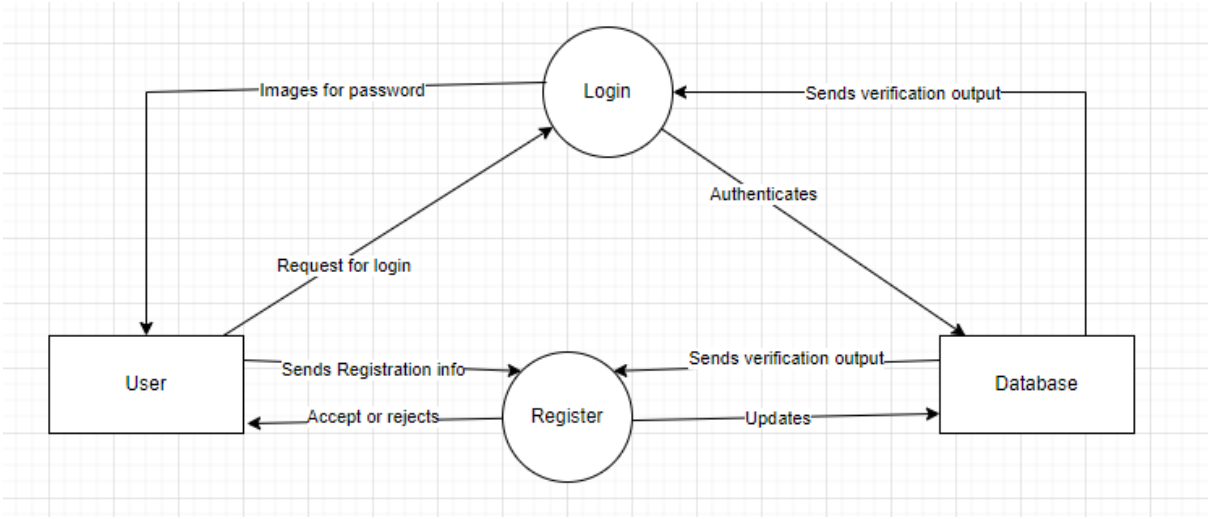

Figure 3.5.2 Data Flow Diagram Level 1

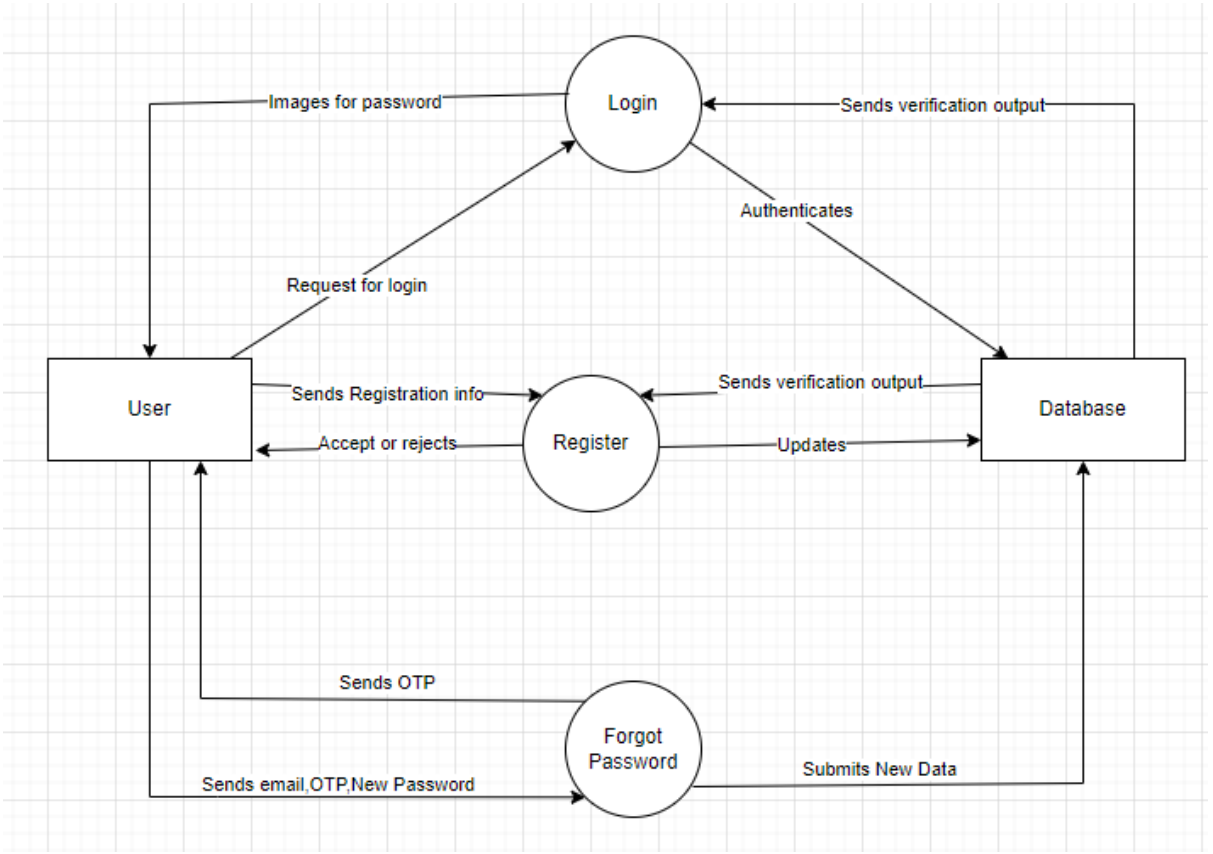

Figure 3.5.3 Data Flow Diagram Level 2

3.5.2 E-R diagram

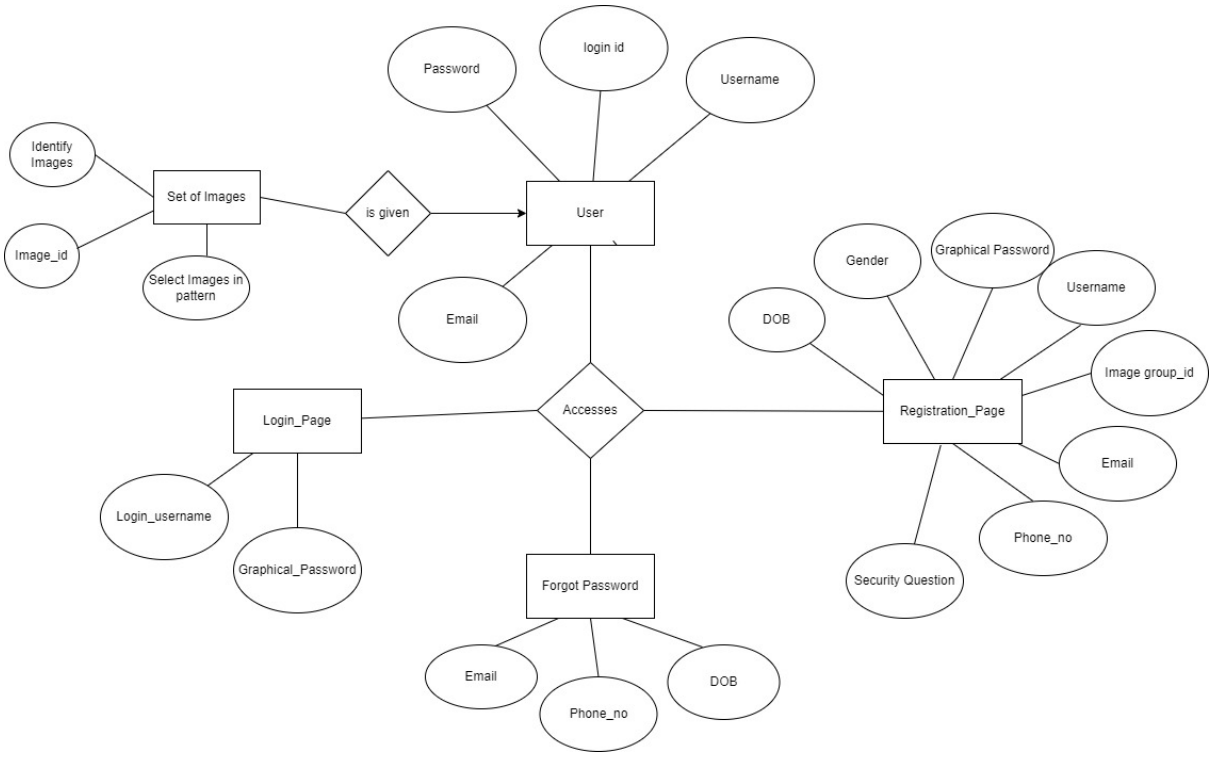

Figure 3.5.2 E-R diagram

3.5.3 Use case diagram

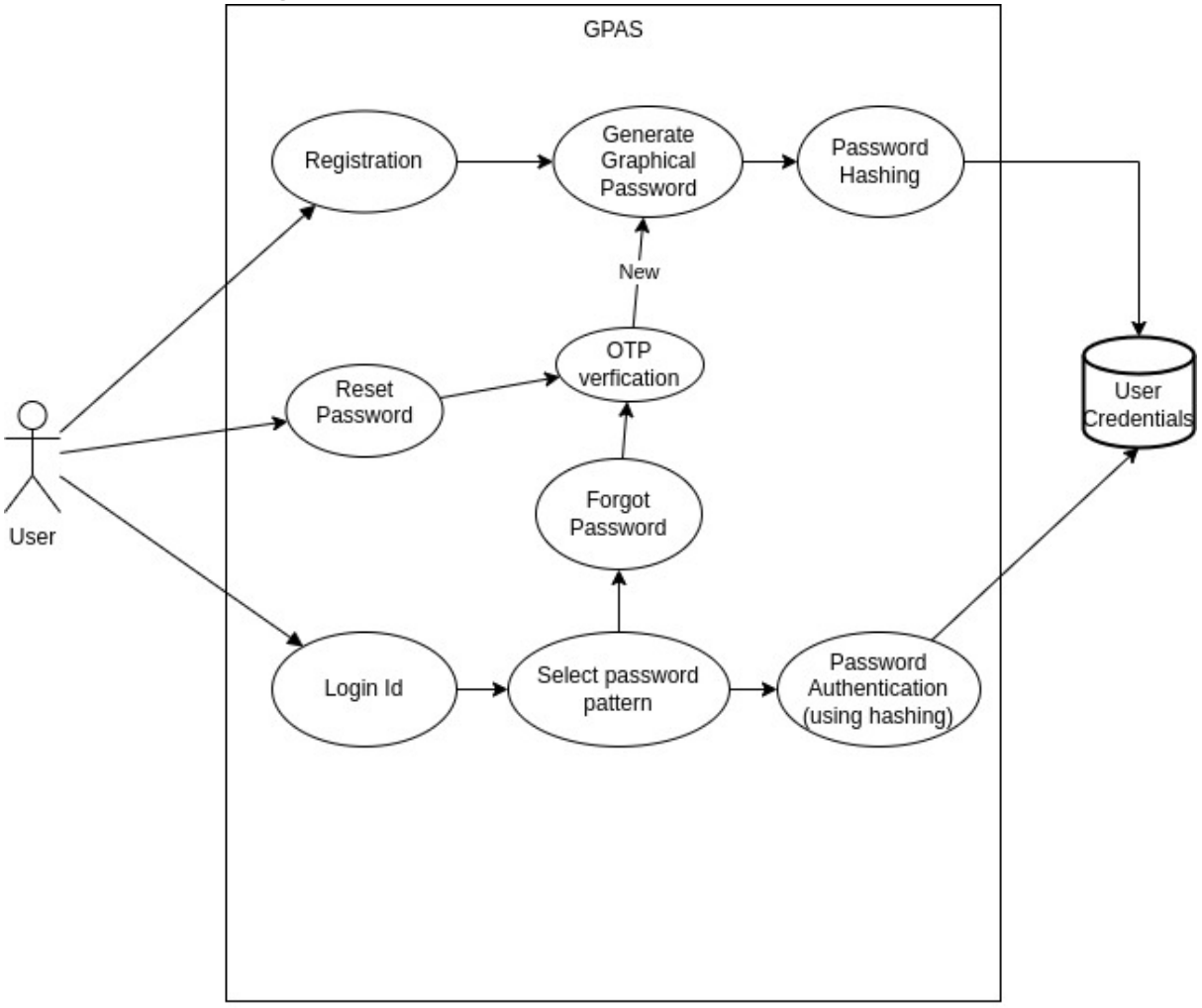

Figure 3.5.2 Use Case Diagram

3.5.4 Database Structure

The GPAS database has the following schema :

```
//defining schema
const imageSchema = new mongoose.Schema({
    image:{type:Buffer,contentType:String, required:true},
    imageName:{type:String}
})

const ImageModel = mongoose.model("image",imageSchema )
```

```
//defining schema
const userSchema = new mongoose.Schema({
    name:{type:String, required:true, trim:true},
    email:{type:String, required:true, trim:true},
    mobileNo:{type:Number, required:true, trim:true},
    password:{type:String, required:true, trim:true},
    imageSetId:{type:mongoose.Types.ObjectId, ref:'imageSet'},
})

const UserModel = mongoose.model("user",userSchema )
```

```
const tempUserSchema = new mongoose.Schema({
    name:{type:String, required:true, trim:true},
    email:{type:String, required:true, trim:true},
    mobileNo:{type:Number, required:true, trim:true},
    password:{type:String},
    confirmPassword:{
    imageSetId:{type:mongoose.Types.ObjectId, ref:'imageSet'},
    isRegistered:{type:Boolean, default:false}
})

const TempUserModel = mongoose.model("tempUser",userSchema )
```

(alias) module "mongoose"
import mongoose

```
const imageSetSchema = new mongoose.Schema({
    imagesSet:[{type:Schema.Types.ObjectId, ref:'image'}],//check t
    allotedUserCount: {type:Number, default:0},
    isUserProvided:{type:Boolean, default:false},
    setCategory:{type:String, default:'all'}

})

const ImageSetModel = mongoose.model("imageSet",imageSetSchema )
```

3.6 Deployment Requirements

There are various requirements to successfully deploy the system. These are mentioned below :

3.6.1 Hardware

System Type:- 32-bit or 64-bit Operating System

OS:-Windows xp or better

3.6.2 Software

Web Browser(Chrome,Firefox)
Node js

Chapter 4

Implementation

For the problem of making a graphical password authentication,we have designed a website which has a login page which allows users only if they put the right password.This system takes passwords in a graphical way and only accepts the login of a user if he selects the right images.The admin controls the database for this system.

4.1 Technique Used

4.1.1 Hashing

Hashing is the process of transforming any given key or a string of characters into another value. This is usually represented by a shorter, fixed-length value or key that represents and makes it easier to find or employ the original string.

The most popular use for hashing is the implementation of hash tables. A hash table stores key and value pairs in a list that is accessible through its index. Because key and value pairs are unlimited, the hash function will map the keys to the table size. A hash value then becomes the index for a specific element.

A hash function generates new values according to a mathematical hashing algorithm, known as a hash value or simply a hash. To prevent the conversion of hash back into the original key, a good hash always uses a one-way hashing algorithm.

Hashing is relevant to -- but not limited to -- data indexing and retrieval, digital signatures, cybersecurity and cryptography.

Hashing is most commonly used to implement *hash tables*. A hash table stores key/value pairs in the form of a list where any element can be accessed using its index.

Since there is no limit to the number of key/value pairs, we can use a hash function to map the keys to the size of the table; the hash value becomes the index for a given element.

4.2 Tools Used

Npm package,body parser,express,ejs view engine

4.2.1 npm package

npm (originally short for Node Package Manager) is a package manager for the JavaScript programming language maintained by npm, Inc. npm is the default package manager for the JavaScript runtime environment Node.js. It consists of a command line client, also called npm, and an online database of public and paid-for private packages, called the npm registry. The registry is accessed via the client, and the available packages can be browsed and searched via the npm website. The package manager and the registry are managed by npm, Inc.

4.2.2 body parser

Express body-parser is an npm module used to process data sent in an HTTP request body. It provides four express middleware for parsing JSON, Text, URL-encoded, and raw data sets over an HTTP request body. Before the target controller receives an incoming request, these middleware routines handle it.

4.2.3 express

Express.js, or simply **Express**, is a back end web application framework for building RESTful APIs with Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js.

4.2.4 ejs view engine

EJS or Embedded Javascript Templating is a templating engine used by Node.js. Template engine helps to create an HTML template with minimal code. Also, it can inject data into HTML template at the client side and produce the final HTML. EJS is a simple templating language which is used to generate HTML markup with plain JavaScript. It also helps to embed JavaScript to HTML pages.

4.3 Language Used

Html,CSS,Javascript,node js, mysql

4.3.1 HTML

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and other items. HTML elements are delineated by *tags*, written using angle brackets.

4.3.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of content and presentation, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

4.3.3 Javascript

JavaScript ,often abbreviated as JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for webpage behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices.

JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O.

JavaScript engines were originally used only in web browsers, but are now core components of some servers and a variety of applications. The most popular runtime system for this usage is Node.js.

4.3.4 Node js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser, which was designed to build scalable network applications. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games).

The Node.js distributed development project was previously governed by the Node.js Foundation, and has now merged with the JS Foundation to form the OpenJS Foundation, which is facilitated by the Linux Foundation's Collaborative Projects program.

4.3.5 MySQL

MySQL is an open-source relational database management system (RDBMS).Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

4.4 Testing

Testing is the process of evaluation of a system to detect differences between given input and expected output and also to assess the features of the system. Testing assesses the quality of the product. It is a process that is done during the development process.

4.4.1 Strategy Used

Tests can be conducted based on two approaches –

Functionality testing

Implementation testing

The texting method used here is Black Box Testing. It is carried out to test functionality of the program. It is also called ‘Behavioral’ testing. The tester in this case, has a set of input values and respective desired results. On providing input, if the output matches with the desired results, the program is tested ‘ok’, and problematic otherwise.

4.4.2 Test Case and Analysis

Test Cases for Registration Page

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
1	TC-001	User Interface	Check all the text boxes, radio buttons, buttons, etc	1. Click on Radio buttons, buttons and dropdowns	N/a	UI works perfectly
2	TC-002	Required fields	Check the required fields by not filling any data	1. Do not enter any value in the field. 2. Click on the Register button.	N/a	It should show a mandatory symbol (*) on mandatory fields.
3	TC-003	Required fields	Check user should Register by filling all the required fields	1. Enter valid values in the required fields. 2. Click the Register button.	N/a	1. Users should be registered successfully 2. A successful registration message should show. 3. Mail should send to the user
4	TC-004	Optional Fields	Check all the optional fields when do not fill data	1. Do not enter any detail in optional fields 2. Enter valid data in required fields 3. Click on the Signup button	N/a	1. It should not ask to fill the optional fields 2. User should be registered successfully 3. A successful registration message should show 4. Mail should send to the user
5	TC-005	Email validation	• Check the Email text field that has an Email address without @ symbol. • Check the Email text field that has a random string instead of a real email. • Check the Email text field that has @ symbol written in words. • Check the Email text field that has a missing dot in	1. Enter Invalid Emails 2. Click on the Register Button.	1.testAtgmail.com 2.test@gmailcom 3.test@gmail 4.@gmail	It should show the validation message for valid email

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
			the email address.			
6	TC-006	Email validation	Check all the valid emails	1. Enter valid Emails 2. Click on the Register Button.	1.test.22@gmail.com 2.test@gmail.com	It should not show any validation message
7	TC-007	Phone Number validation	Check the phone number when passing alphanumeric data	1. Enter alphanumeric data in phone field 2. Click on Register button	1. dada5$7567#7	It should show the validation message 8 for Phone Number
8	TC-008	Phone Number validation	Check the phone number when not pass country code	1. Enter valid phone number without country code 2.Click on Register button	1. 9012078654	It should show the validation message for country code is required
9	TC-009	Phone Number validation	Check the phone number when passing country code	1. Enter valid phone number with country code 2.Click on Register button	1. +9190112244	It should not show any validation message
10	TC-010	Password Validation	Check the password limit when enter value less than min	1. Enter value which is alphanumeric but less than 8. 2.Click on Register button	1. Password	It should show validation message
11	TC-011	Password Validation	Check the password limit when enter value greater than max	1. Enter alphanumeric value but more than 32. 2.Click on Register button	Any Random string with numbers	It should show validation message
12	TC-012	Password Validation	Check the password when passing only numbers	1. Enter a value in numbers which is in between 8-32 2.Click on Register button	1. 12345678	It should show validation message
14	TC-014	Required Fields	Verify if blank spaces are passed in required fields.	1. Go to the Site. 2. Passed blank spaces in required fields. 3. Click on the Register button	N/a	Those Blank spaces should trim and Validation error message for required fields should visible.
15	TC-015	Required Fields	Verify user can verify its Email ID	1. Go to the Email.	test22@gmail.com	User should get a verification link

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
				2. Click on the verification link.		and able to verify his/her Email ID.
16	TC-016	Phone Number Validation	Verify if the length of the phone number is incorrect i.e. less than 10.	1. Enter phone number less than 10 digits. 2.Enter all required fields. 3.Click on Register Button	91901122	It should show the validation error message for phone number length.
17	TC-017	Phone Number Validation	Verify if the length of the phone number is incorrect i.e. more than 10	1. Enter phone number less than 10 digits. 2.Enter all required fields. 3.Click on Register Button	9190112244556 6	It should show the validation error message for phone number length.

Test Cases for Login Page

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
1	TC-01	User Interface	Check all the text boxes and buttons	Check Page	• UI should be perfect • Text boxes and button should be aligned
2	TC-02	Required Fields	Check the required fields by not filling any data.	1. Enter invalid username 2. Enter correct password 3. Click on Login Button	User should not log in and should show proper error message
3	TC-03	User Login	Check When passing a correct username and invalid password	1. Enter valid username 2. Enter incorrect password 3. Click on Login Button	User should not log in and should show proper error message
4	TC-04	User Interface	Check Keeping Password	1. Enter valid username 2. Do not enter password 3. Click on Login Button	User should not log in and should show proper error message
5	TC-05	User Login	Check when pass correct email and password	1. Enter valid username 2. Enter valid password	User should log in

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
				3. Click on Login Button	
6	TC-06	User Login	Check if the image password is entered in encrypted form.	1. Enter valid username 2. Enter password 3. Click on Login Button	Password is entered in encrypted form
7	TC-07	Signup Option for new users	Check whether the signup link for the new user is working	Click Signup link	Clicking signup link takes the user to signup page successfully
8	TC-08	Forgot Password	Verify user should get an error message when he/she enters not registered email id.	1. Click on the Forgot password link. 2. Enter unregistered email id and click on the send button.	User should get an error message.
9	TC-09	Reset Password	Verify user should get an error message when he/she enters the previous password.	1. Go to the reset password link. 2. Enter the previous password. 3. Click on the Reset Password button.	User should get an error message.
10	TC-10	Reset Password	Verify user able to reset his/her password	1. Go to the reset password link. 2. Enter a new password and a confirm password. 3. Click on the Reset Password button.	Users should get the success message and the password should get reset.
11	TC-11	Reset Password	Verify user should get an error message when password and confirm password not matches	1. Go to the reset password link. 2. Enter a different new password and a confirm password.	Users should get an error message.

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
				3. Click on the Reset Password button.	
12	TC-12	Reset Password	Verify user should able to login with a new password.	1. Go to the reset password link. 2. Enter a new password and a confirm password. 3. Click on the Reset Password button. 4. Log in by using the new password.	User should able to login
13	TC-13	Reset Password	Verify if the user enters a new password that does not cover the basic requirements of password then the user should be displayed error message	1. Go to the reset password link. 2. Enter a new password that does not cover the basic requirements. 3. Click on the Reset Password.	Users should get an error message.
14	TC-14	Required Fields	Verify if blank spaces are passed in required fields.	1. Go to the Site. 2. Passed blank spaces in required fields. 3. Click on the Login button	Those Blank spaces should trim and Validation error message for required fields should visible.
15	TC-15	Welcome Email	Verify new users should get the welcome email once after the login.	1. Go to the Email. 2. Enter Login Email.	Users should get a welcome email on his/her email id.
16	TC-16	User Login	Verify when passing incorrect Email and correct password	1. Enter incorrect Email. 2. Enter the correct password. 3. Click on the Login Button.	User should not be able to log in and the error message should be displayed.
17	TC-17	User Login	Verify when passing both incorrect Email and password	1. Enter incorrect Email.	User should not be able to log in and the

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
				2. Enter the correct password. 3. Click on the Login Button	error message should be displayed.
18	TC-18	User Forgot Password.	Verify Forgot Password sends a forgot password link.	1. Click on the Forgot Password link. 2. Enter Email and click on the send button. 3. Now go to mail7.io and enter the email id.	User should get the forgot password link on his/her email id.

Chapter 5

Conclusion

5.1 Conclusion

The aim of the project was to create a website which would be able to create a graphical password from its user,store them into a database and create a graphical login page.

The authentication in our system will be highly secure and it can reduce all the extra efforts of maintaining the records.

5.2 Limitations of the Work

Our system accepts only objective questions.

Errors can be costly.

It is time-consuming as the time it takes to maintain the color scheme of a page and to make lists, tables and forms.

It is browser dependent.

Bibliography

[1]https://www.ijraset.com/research-paper/graphical-password-authentication-system

[2]https://www.researchgate.net/publication/224229789_A_graphical_password_authentication_system/link/0deec52c95e5bdae49000000/download

[3]https://www.ijcsmc.com/docs/papers/June2017/V6I6201784.pdf

[4]https://core.ac.uk/download/pdf/162009043.pdf

[5]https://www.ijert.org/research/graphical-password-authentication-system-IJERTV2IS90953.pdf

[6]http://www.ijtrd.com/papers/IJTRD70.pdf

[7]https://www.sciencedirect.com/science/article/abs/pii/S1071581905000625

[8]http://www.ijesit.com/Volume%202/Issue%202/IJESIT201302_16.pdf

[9]https://www.seminarsonly.com/Labels/Graphical-Password-Authentication-Project.php