A Project/Dissertation Finial Review Report

on

Android Notes with Smart Sticky Notes Android app

Submitted in partial fulfilment of the
requirement for the award of the degree of

B. tech/ CSE

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Mr. R Vijay
Professor

Submitted By -

Kishan Kumar (19SCSE1010690)

Prashant Gupta (19SCSE1010294)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
December, 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"Android Notes with Smart Sticky Notes Android app"** in partial fulfilment of the requirements for the award of the Bachelor of Technology submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of November, 2021 to May 2022, under the supervision of Mr R Vijay Assistant Professor, Department of Computer Science and Engineering of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any

other degree of this or any other places.

Kishan Kumar – 19SCSE1010690

Prashant Gupta – 19SCSE1010294

This is to certify that the above statement made by the candidates is correct to the best

of my knowledge.

Mr V R Vijay

Assistant Professor

Acknowledgement

Abstract

Note taking application become one of the most important on the smartphone. It has the capability of storing the data just like the assistant which is free of cost. This app is a simple but used to secure notes with Biometric Authentication, like old way where we needed passwords to be entered. Through it doesn't require user registration, only the owner of the phone can access it. This app can be used to save private notes or personal data. As biometric authentication is considered as the highest level of security of any phone, this app would be a good to store personal information. And the data of the notes is stored in the encrypted form in local as well as on the cloud so that the user can access the notes from anywhere the cloud data is only stored if the user want to store the data on the cloud. And the sticky notes of this app help to note the daily task which needs to be done.

❖ **Advantages**

- You will never forget the important meetings or tasks.
- You can also set a colour to the sticky note.
- You can add ideas, important things at any time.
- Your data is more secure because of encrypted data storing.
- Biometric Authentication and many securities system.
- Data store in local system as well as cloud.
- Accessing data from any other Android Phone.

❖ **Limitation**

- Only Android mobile users can use this application

❖ **Application**

- This system can be used by the multiple peoples to get the counselling sessions online.

Contents

List of Figures

Chapter 1

1.1 Introduction

The aim of this project is to use encrypting data in notes app for securing the data for users who want privacy and security. This app is a simple but smart idea used to secure notes with Fingerprint Authentication. It can also be referred to as Keyless Authentication, unlike old ways where it needed passwords to be entered. Though it doesn't require any user registration, only the owner of the device can access it. This App can be used to save private notes or personal data.

The data will be stored in the device using SQLite. And for the front end and functioning, you can use the platform of Android Studio. This app would need biometric authentication in the device; else it can't be used.

Here the users can add new notes, edit the old notes, and delete those that are no more required. As biometric authentication is considered as the highest level of security of any Phone, this app would be a good idea to store personal information.

In a busy schedule, we tend to forget many important things easily, and to remember these things we need to note down things in a piece of paper. In such busy schedules people need some personal assistant or a reminder to remind them about the important work that needs to be done. This android application will help them to remind, to do such important things. These sticky notes android application can help us to note the daily task which needs to be done. Important meetings, events etc. can be recorded with great ease through the use of this application. This application will allow the users to organize the data in a simpler and easy way. In this application users can view all the Notes, likewise also can manage old and new notes. User can also change the colour of the sticky notes. User can grant permission for floating notes. Only 3 notes can be floated and which can be further expanded and collapsed. These sticky notes can also be edited and deleted. Notes app makes it convenient to get work done from everywhere and anytime. You can create research notes using the familiar office-style toolbar and stored in specific box folders as simple as a push of a button. You can securely upload PDFs, Microsoft Office files, photos, and other documents or even add a link without attachments to Box. Box can preview more than 200 file types with high quality

either online or offline using hundreds of partner apps. This makes your writing and editing processes become much easier and more productive. This app has become the Winner of PC Magazine's Editors' Choice Award. With its reputation and recognition, you should feel more confident to try it out. Your box notes, a truly elevated experience.

Benefits of Notetaking

Beyond keeping track of what was said in a lecture or class activity, taking notes has various advantages. Taking good notes:

It keeps you awake. Taking notes keeps your body active and engaged, preventing feelings of tiredness or distraction.

It stimulates your thoughts. Listening intently and making decisions about what to include in notes keeps your mind engaged in what you're hearing.

Information is emphasised and organised. You'll pick on and underline the essential concepts you hear while taking notes, determining the structure of a class presentation. You'll also be able to indicate a presentation's supporting elements, making study and comprehension easier after class. It's also easy to connect classroom learning to textbook readings with such well-organized notes.

Makes a compact record that can be studied. You'll have all you need for study, learning, and review after class if you keep a set of short, well-organized notes from each class session.

Where does notetaking start

Active listening is required for taking notes. Unfortunately, only a small percentage of students are taught how to do so. When you make a conscious effort to hear and understand, especially in challenging situations or with difficult subjects, you can enhance your listening skills.

Listening Skills That Are Required

Listening well is an active process that takes commitment, energy, and focus. The following tips will assist you in listening carefully, precisely, and actively:

Distract yourself from the task at hand. You must take responsibility for concentrating on what a class gives, regardless of the distractions — noises, worries, or people. No one else can hear what you're saying. Make a commitment to listen, even if it's difficult, and limit your own distractions.

Concentrate on the main points. Imagine that you must summarise the material you record to others after class while you take notes. This will compel you to choose which pieces of knowledge are most important to provide. Make use of your additional mental speed.

To understand what you're hearing Of course, this type of thinking necessitates focus. You will be well equipped to take in and organise new knowledge if you prepare for class by undertaking prescribed reading and reviewing prior class notes. Make sure you're ready to learn.

Maintain your concentration. Refocus when your mind wanders. Bring your focus back to what you're hearing. If you're unclear or losing interest, ask a clarifying question or make a question mark in your notes. You can revisit the topic with your instructor or another student later. Always keep track of your notes so that you have a thorough record of each class.

Judge the message, not the messenger—While not all educators are great presenters, the material they deliver is nonetheless significant. Concentrate on what you're hearing rather than how it's being delivered. Determined to gain knowledge from every class presentation.

You can't take good notes unless you're actively listening. There will be no lasting learning from a class session if you don't take good notes. Listening is the first step in learning!

What are the disadvantages?

Despite the development of note-taking applications, many students still prefer the general feeling of taking notes on paper. Regular paper-based notes also have the advantage of being less prone to technical failures than tablets and phones. According to many sources [3] [4], taking a handwritten note is more helpful in memorizing important than typing. Certain limitations of the applications taking the updated notes may also prevent detection, especially the difficulty of collaborating and transferring notes taken within these applications. Some features within applications are also found to be unavailable without an internet connection. Students may not be discouraged from using these apps due to the large number of note taking apps available and their various functionality.

Apart from the restrictions of the devices themselves, there are also disagreements about whether resources should be allowed in the classroom, as they can serve as a distraction for students. Users who fall under this category should consider live pens, which allow students to write on paper while the notes are digitally displayed.

Where is it going?

Currently, the popularity of the internet has led to a decline in revenue in the paper industry [5], but pen and pencil industry revenue has grown in recent years [6]. These contradictory numbers do not overlap as

to whether students drop out of paper-based notes or not. Many students take advantage of access to all the electronic items in the classroom, but many still choose to take a paper-based note.

As note-taking technology becomes increasingly sophisticated with the use of pencils and style, we can expect to see an increase in classroom notation in a variety of ways. With the growth of online, mixed, and integrated reading lessons, we can see the power of conveying complex information, such as handwriting or diagrams, digitally. External note-taking tools also take note-taking features. For example, eBooks now use note-taking features such as annotations, highlighting, and underlining to simulate a textbook reading experience.

What are the effects of teaching and learning?

There are opportunities for ingenuity to play a significant role in developing note-taking skills for their students. The great advantage of note-taking requests is that notes can be shared between students and teachers. These notes can serve as reminders for completing assignments or as a teaching tool for explaining classroom lessons, creating discussion questions, and more. The most powerful apps are the ones that allow real-time interaction. In this way, a group of students from two to two as large as the whole class can participate in viewing, reviewing, and / or creating a document, simultaneously.

Chapter 2

2.1 Literature Survey

The survey regarding this application includes information gathering from various sources. These sources include some of the car showrooms and service centres, various related web sites and similar projects developed previously. IEEE papers are used for clearing the concepts and algorithms included in this project. E.g., Google cloud messaging paper for push message services, Dijkstra's Algorithm for finding shortest path algorithm, etc.

Encryption Techniques

Symmetric encryption

The Advanced Encryption Standard (AES) is used to provide symmetric encryption (AES). The AES algorithm is a symmetric block cypher that can both encrypt (encrypt) and decrypt (decrypt) data. It encrypts plain data with only one secret key and processes 128-bit data locks with 128-, 192-, and 256-bit keys. This algorithm takes data and encrypts it with the use of a password. The data is decrypted using the same password and a decryption algorithm (or the same encryption algorithm, in some cases).
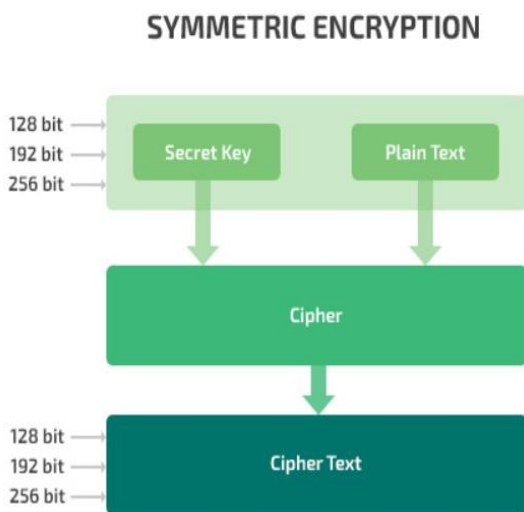
Fig. 1: Symmetric encryption principle of operation

Symmetric encryption is quick and simple to set up. However, it raises certain security concerns.

If your app employs symmetric encryption, you'll need to be careful about creating and securing keys. To prevent hackers from obtaining keys through credential stuffing or brute-forcing, keys should be complex and well-protected. Consider a secure way for distributing and storing keys to prevent them from falling into the wrong hands. Keys should not be sent over the network in plain text or written on a sticky note and taped to the display.

Despite its flaws, symmetric encryption is often used due to its speed and security. WhatsApp, Firefox, and early versions of Microsoft Outlook all use it. This mechanism is also used by ransomware that encrypts files. Attackers use symmetric encryption to encrypt files on victims' systems. Because only the key can decode data, attackers extort money from victims in return for the decryption key.

The key Generator object was created in the Aes256Class () function Object () { [native code] }. The key size is set to 256 bits. The generate Key () function is then used to generate a random key of the requested size (256 bits).

The makes function is called with the source byte array and the Cipher class object's operation mode:

Encryption mode (Cipher.ENCRYPT MODE) or decryption mode (Cipher. DECRYPT MODE).

USABILITY REQUIREMENT

The android application is designed for user friendly environment and ease of use.

IMPLEMENTATION REQUIREMENT

Implementation of the system of Android application using flutter for UI/UX design and working of application and in backend and web using css and html in front end with jsp as back end and it will be used for database connectivity. And the database part is developed by mysql.

FUNCTIONAL REQUIREMENTS

 USER

➢ **USER LOGIN**

 Description of feature This feature used by the user to login into system. A user must login with his user's name and password to the system after registration. If they are invalid, the user not allowed to enter the system.

Functional requirement

• Username and password will be provided after user registration

• Password should be hidden from others while typing it in the field.

➢ REGISTER NEW USER

Description of feature A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept a new user by unblocking him.

• Functional requirement

• System must be able to verify and validate information.

• The system must encrypt the password of the customer to provide security.

Tools and Techniques

- HTLM
- CSS
- JavaScript
- Android Studio
- Encryption
- SQLite
- Xml
- Hashing

Android Studio and Flutter

- Same UI and Business Logic in All Platforms.

- Reduced Code Development Time.

- Increased Time-to-Market Speed.

- Similar to Native App Performance.

- Custom, Animated UI of Any Complexity Available.

- Own Rendering Engine.

- Simple Platform-Specific Logic Implementation.

- The Potential Ability to Go Beyond Mobile.

Syntax

```
import 'package:flutter/material.dart';


void main() { runApp(const
  MyApp());
}


class MyApp extends StatelessWidget {

  const MyApp({Key? key}) : super(key: key);


  @override

  Widget build(BuildContext context) { return
    MaterialApp(
      title: 'Welcome to Flutter',

      home: Scaffold( appBar:
        AppBar(
          title: const Text('Welcome to Flutter'),

        ),

        body: const Center(

          child: Text('Hello World'),

        ),

      ),

    );

  }

}
```

HTML

- The Hyper Text Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser.

- HTML describes the structure of a web page semantically and originally in- clued cues for the appearance of the document.

- It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Syntax

<tag> content </tag>

Some tags which we have used are as follows –

<p> Paragraph Tag </p>

<h2> Heading Tag </h2>

 Bold Tag

<i> Italic Tag </i>

<u> Underline Tag</u>

 Tag: br stands for break line, it breaks the line of the code.

<hr> Tag: hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

CSS

- Cascading Style Sheets (CSS) is a style sheet language used for describ- ing the presentation of a document written in a markup language such as HTML.

- CSS is a cornerstone technology of the World Wide Web, alongside HTML and

JavaScript.

- CSS is designed to enable the separation of presentation and content, in- cluding layout, colors, and fonts.

- CSS has a simple syntax and uses a number of English keywords to spec- ify the names of various style properties.

- A style sheet consists of a list of rules. Each rule or rule-set consists of one or more selectors, and a declaration block.

Syntax

Some examples are –

- <h1style="background-color:DodgerBlue;">HelloWorld</h1>

- <p style="background-color:Tomato;">Lorem ipsum...</p>
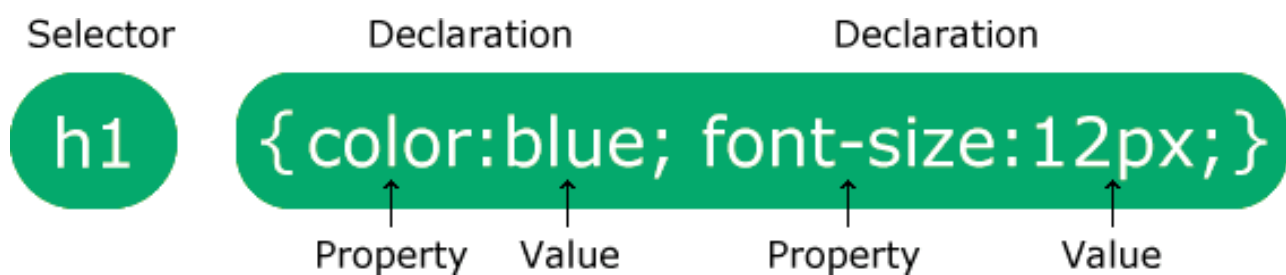
- background-color: lightblue; • height:200px;
width:50%;

JAVASCRIPT

- JavaScript often abbreviated as JS, is a programming language that conforms to the ECMAScript specification.

- JavaScript supports 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).

- JavaScript engines were originally used only in web browsers, but they are now core components of other software systems, most notably servers and a variety of applications.

  Some examples of codes of javascript language are –

- //Change heading:

  document.getElementById("myH").innerHTML = "My First Page";

  // Change paragraph:

  document.getElementById("myP").innerHTML = "My first paragraph.";

- letx=5; //Declarex,giveitthevalueof5
  lety=x+2;
  //Declarey,giveitthevalueofx+2

SQLite

SQLite is an in-house library that uses a self-contained, non-server, zero, active, SQL
information engine. SQLite code is in the public domain so it is free to use for any commercial,

commercial or private use. SQLite is the most widely distributed website in the world with more applications than we can count, including a few high-level projects.

SQLite is an embedded SQL data engine. Unlike other SQL websites, SQLite does not have a separate server process. SQLite reads and writes directly to standard disk files. A complete SQL website with multiple tables, indexes, triggers, and views, is contained in a single disk file. Cross-platform file format - you can freely copy the site between 32-bit and 64-bit programs or between large indian and small formats. These features make SQLite a popular option as a program file format. SQLite website files are the final format recommended by the US Library of Congress. Think of SQLite not as a replacement for Oracle but as a substitute for fopen ()

SQLite is an integrated library. With all features enabled, the library size can be less than 750KiB, depending on the target location and the developer development settings. (64-bit code is great. And other compiler settings such as aggressive operation and loosening of loops can cause object code too large.) There is a trade-off between memory usage and speed. SQLite usually works faster when you provide additional memory. However, performance is often good even in areas with low memory. Depending on how it is used, SQLite can be faster than a straightforward I / O file system.

SQLite is carefully tested before each release and has a reputation for high reliability. Most SQLite source code is only provided for testing and verification. The automated test suite uses millions and millions of test cases that include hundreds of millions of individual SQL statements and gains 100% branch testing. SQLite responds well to memory allocation failures and disk I / O errors. Activity is ACID even if it is interrupted by a system crash or power failure. All of this is verified by automated tests using specialized testing equipment that mimics system failures. Of course, despite all these tests, there are still bugs. But unlike other similar projects (especially commercial competitors) SQLite is open and reliable about all bugs and provides bug fixes and timely coding changes.

The SQLite code base is supported by an international team of full-time SQLite engineers. Developers continue to increase SQLite capabilities and improve its reliability and performance while maintaining back-to-back interface with published interface, SQL syntax, and website file format. Source code is free for anyone who wants it, but professional support is also available.

The SQLite project was started in 2000-05-09. The future is always hard to predict, but the goal of the developers is to support SQLite by 2050. Design decisions are made with that in mind.

We the engineers hope you will find SQLite useful and we ask you to use it well: to make good and beautiful products fast, reliable, and easy to use. Seek forgiveness as much as you forgive others. And just as you got SQLite for free, so also give it away, pay off the debt up front.

Extensible Markup Language (XML)

Extensible Markup Language (XML) is a simple, flexible text format based on SGML (ISO 8879). Originally designed to meet the major electronic publishing challenges, XML also plays an important role in broad data exchange on the web and elsewhere.

This page describes the work done on W3C during XML Tasks, and how it was built. W3C operations occur in Working Groups. Working Groups within the XML function are listed below, along with links to their individual web pages.

You can find and download the official official specifications here, because we publish them. This is not a place to find tutorials, products, courses, books or other information related to XML. There are links below that can help you find such resources.

You will find links to W3C Recommendations, Recommended Recommendations, Effective Drafts, compliance check sites and other documents on the pages of each Working Group. Each

document contains email addresses that you can use to send ideas or queries, for example, when you write software to use and detect problems or errors.

Please do not email us asking us to help you learn a language or specific details; there are many resources on the internet, and the people who plan and develop the information are very busy. We like technical ideas and error.

If your organization would like to join W3C, or would like to formally participate in a working group (and have the necessary resources to attend meetings), you can learn more about the Consortium.

Hashing

To check the integrity of data, hashing employs a mathematical method. A hash function takes some data and generates what appears to be a random string of symbols. This string is always the same length, but the character sequence changes. Each input is given a unique value by the ideal hash function. And the hash produced by the same input will always be the same. As a result, hashing can be used to verify data integrity.

Hashing is used to encrypt user passwords in Android apps. A hashing algorithm makes recovering the original password from the hash virtually hard.

Syntax

```
String targetString = "Hello";

MessageDigest messageDigest = null;
byte[] digest = new byte[0];

try {
    messageDigest = MessageDigest.getInstance("MD5");
    messageDigest.reset();
    messageDigest.update(st.getBytes());
    digest = messageDigest.digest();
} catch (NoSuchAlgorithmException e) {
    e.printStackTrace();
}

BigInteger bigInt = new BigInteger(1, digest);
String md5Hex = bigInt.toString(16);

while( md5Hex.length() < 32 ){
    md5Hex = "0" + md5Hex;
}

Log.d("Encoded string: ", md5Hex); It was originally published on
https://www.apriorit.com/
```

Output:

```
Encoded string: 8b1a9953c4611296a827abf8c47804d7
```

Table 1: Tasks in Note Taking

Abstract Task	Description
Note Creation	The task of writing notes while in class or while taking notes from a source. It also includes getting copies of notes from a peer.
Note Management	Editing and organizing the notes.
Note Use	Using the notes to study for an exam or as reference while working on an assignment

Encryption and Decryption of Data on Images Android Project

Android is a mobile device operating system and software platform.

This is a project that encrypts text on photos in order to send secret info over the internet.

The sender encrypts the data with a key, which is then handed to the receiver to decrypt and retrieve the data.

INTRODUCTION

Data encryption on photos ensures that data is transmitted safely and securely between the sending and receiving parties.

The sender's desired data/text is chosen first, followed by an image from the sender's current mobile device.

The chosen text is then encrypted in the image, making the data unreadable to anybody else.

The image is delivered to the receiving party after encryption, and the receiver decrypts the data using the key provided by the sender.

This application's decryption operation can only be executed by an authenticated receiver.

OBJECTIVE

The main goal of this app is to provide a safe and private transmission of text by encrypting it on a picture with a key that can only be decoded by an authenticated recipient using the same key on the same app.

KEY FUNCTIONALITY

• Start
• Chosen text/retrieve data from current device
• Browse an image from the device
• Encrypt the chosen text onto the image
• This image is sent to the receiver
• Receiver now decrypts the data

SOFTWARE AND HARDWARE REQUIREMENTS:

Java, JDK, Android SDK, My Eclipse

Android Mobile Device

Data Flow Diagram:

sender → Encryption decryption → Receiver

DB

Level 0

Sender → register → Login → Home

Change password

encryption

decryption

Level 1

```
Sender ───▶ Register ───▶ Login ───▶ Home
                │               │          │
               ▢▢             ▢▢▢          ├──────────────┐
                                            ▼              ▼
                                       Encryption     Decryption
                                        │      │       │       │
                                        ▼      ▼       ▼       ▼
                                      Text   Image    Text   Image
                                               │              │
                                               ▼              ▼
                                            Browse         Browse
                                               │              │
                                               ▼              ▼
                                             Show           Show
                                               │              │
                                              ▢▢            ▢▢▢
```
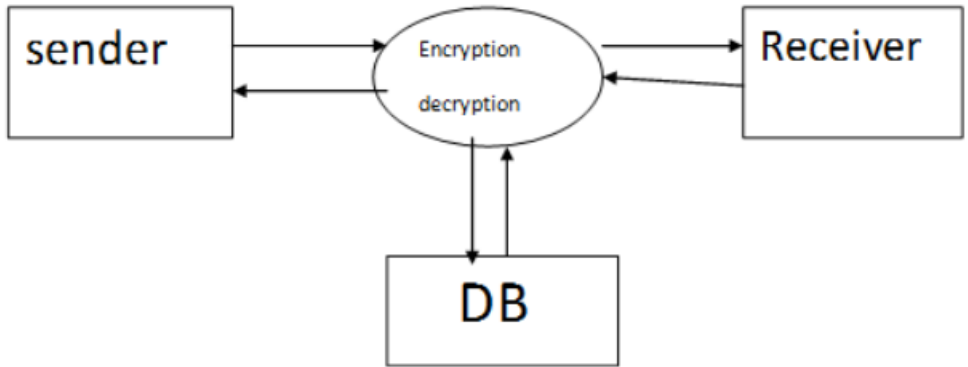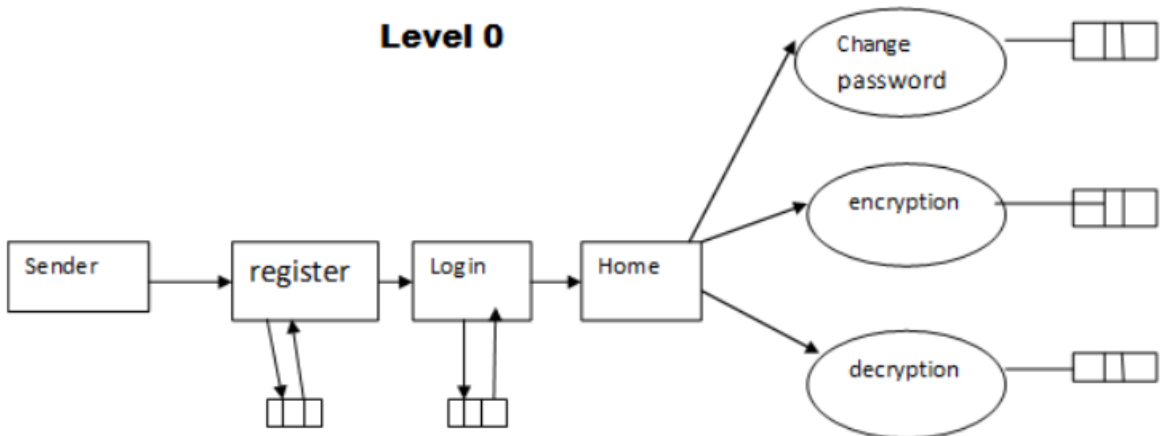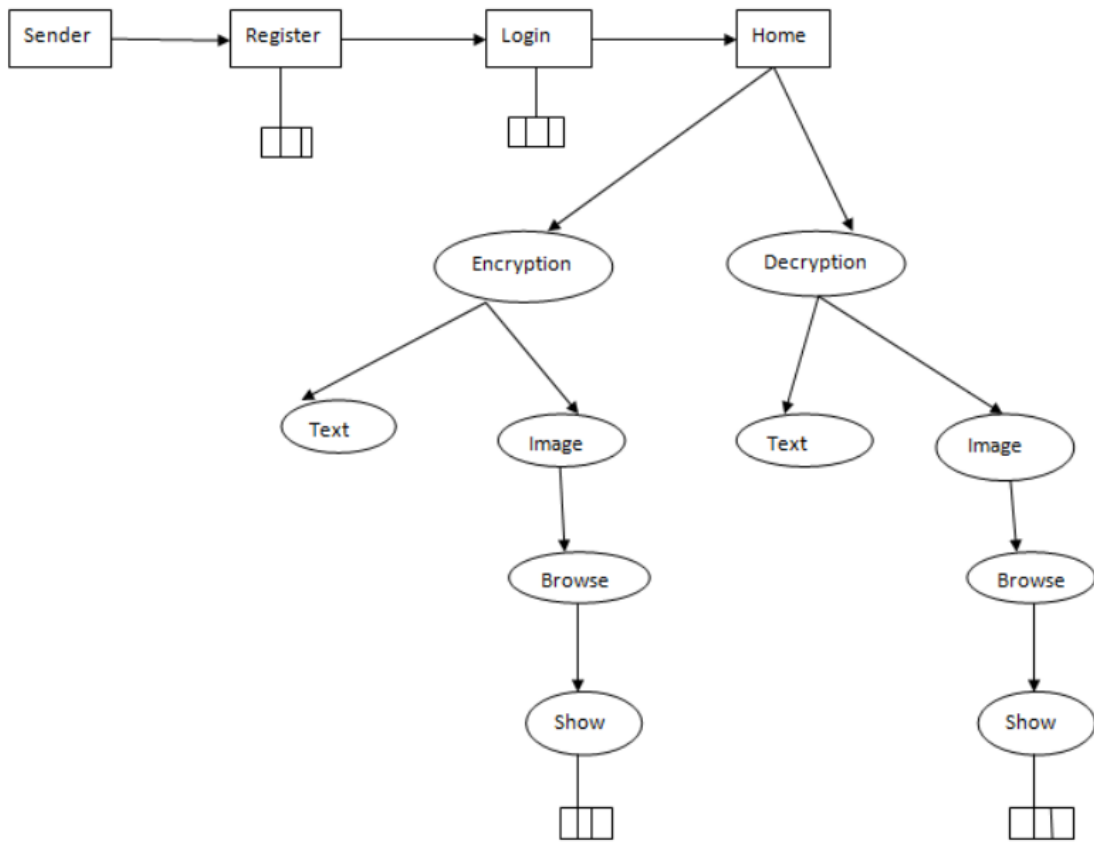
Level 2

The Use of Colour in Notes

Apart from the use of colour, we did not find much evidence to suggest that other features were used. There were no note flags or the use of handwriting recognition (i.e., to literally change their writing into text, not OneNote performance recognition enable search) and there is only one instance of text directly written on the page. Students may not do it find these features useful or may not have known they existed. Although they may be very helpful in some cases or for other students, it did not seem necessary for this particular group. As mentioned earlier, this is a function where the speed it is important and extra effort, no matter how small, will likely be avoided.

Research conclusions

From the case study, we found that students tend to use the device more quickly or often stop using it. As mentioned, the number of notes one student had remained unchanged during the semester but declined to two other students. It is reasonable to assume that if the user does not see the real benefit from the device, then their use it will drop when the initial excitement is over. While there are many other factors that may be involved, this it may be the cause of the decrease in notes. Although we did not try to measure students' perceptions of how fast they can put their notes in their machines, it seems logical in some of our results, that input speed can be part of it

problem. And, as we have seen in previous studies, the response to the device seems to be very personal affects the decision to use the device.

We have noted the low utilization of features provided by the electronic medium. This is not just a stage creation note, but and others as well. There seems to have been little or no effort on rearranging, editing, or tagging categories during storage and use of sections. This raises doubts as to how well these sections are supported.
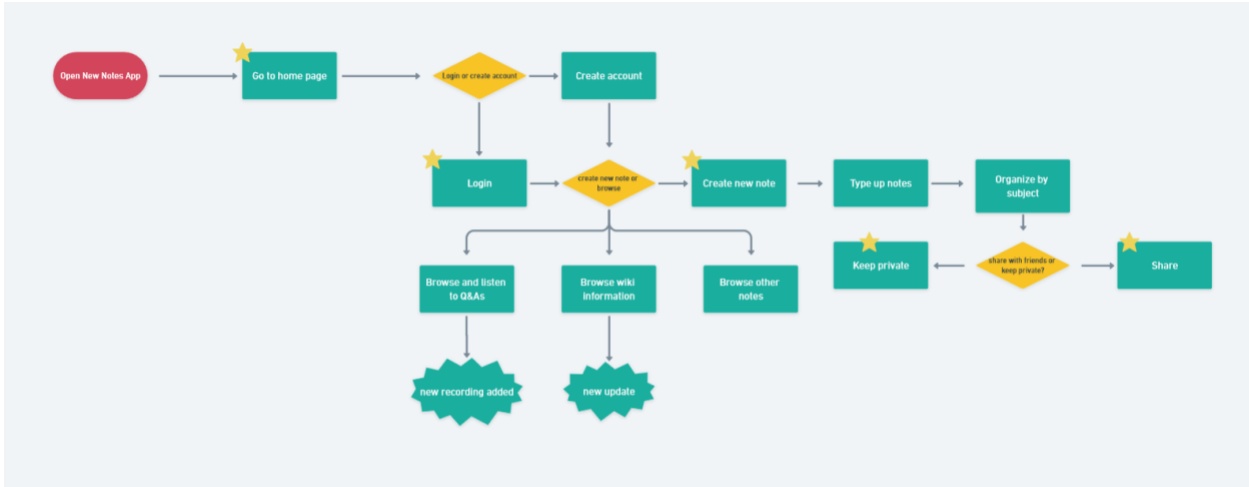
INTERVIEW

The purpose of this activity was to obtain information on how students react to taking notes to assess the impact of workplace expertise and the development of electronic note-taking systems. Additional formal studies are required in order to understand the impact of note-taking systems on learning. In this section we present what we have seen about the work of
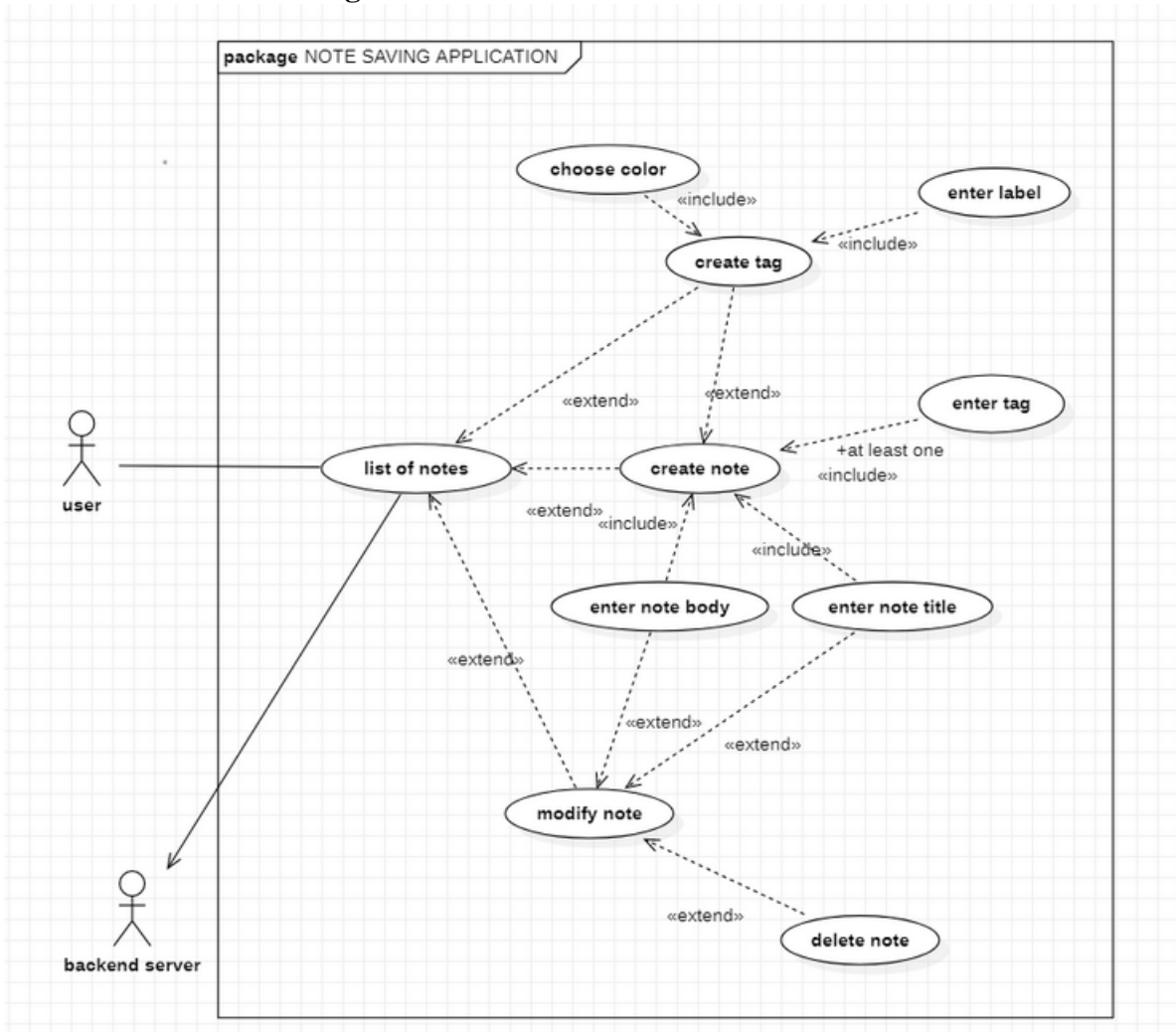
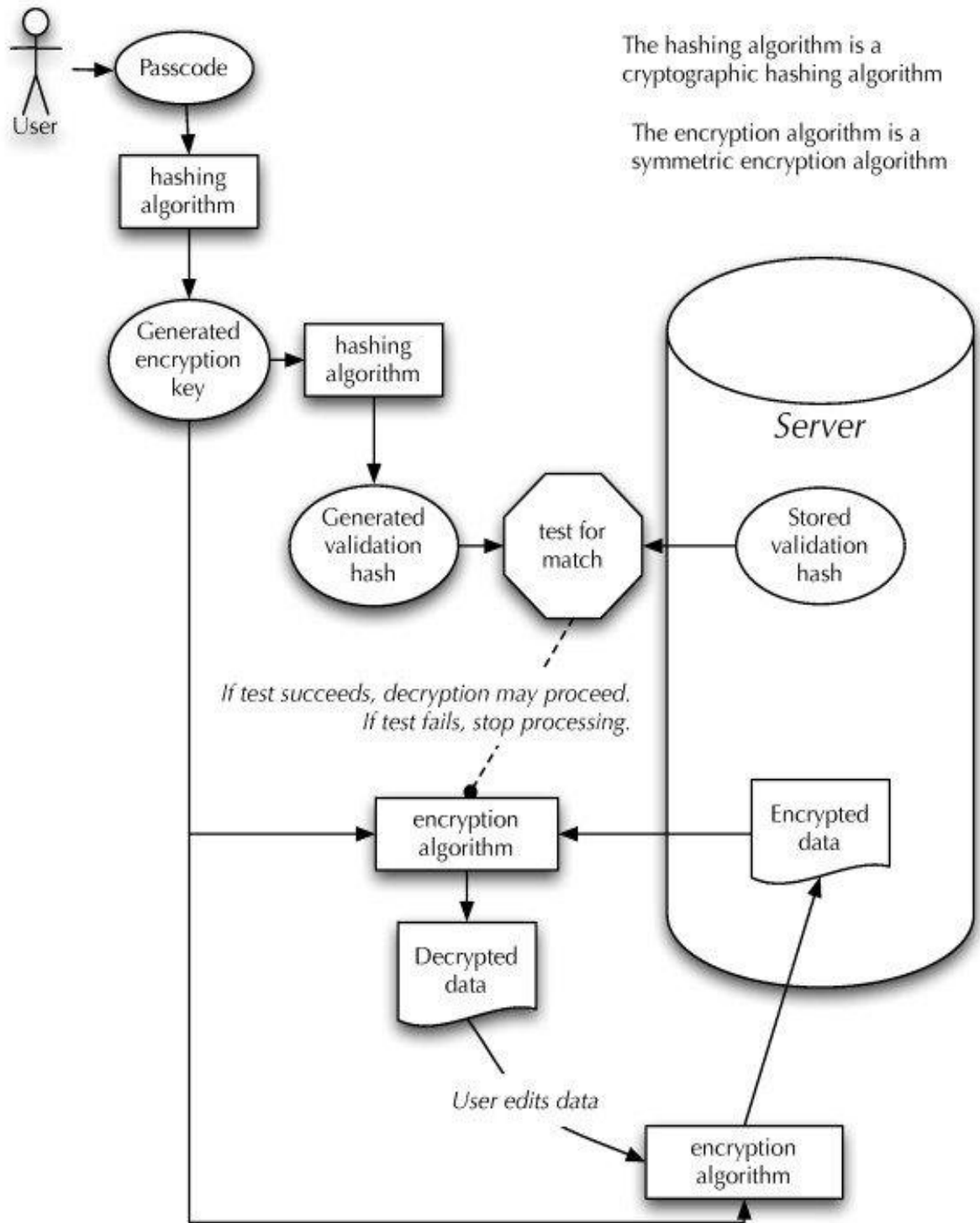taking notes in class based on our lessons.
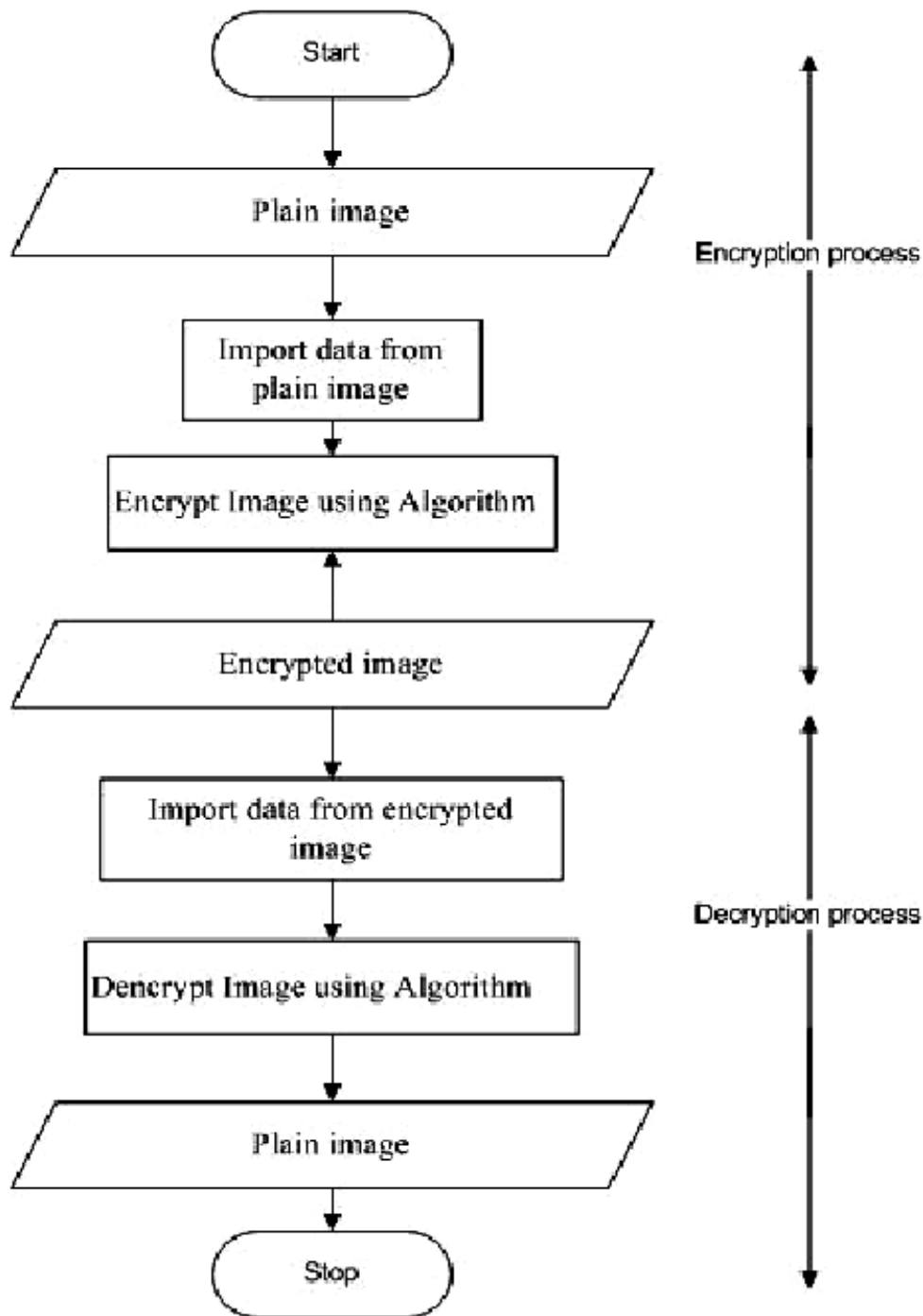
Diagrams

App Flow Chart

Authentication UML Diagram

Data Encryption Flow Diagram

The hashing algorithm is a cryptographic hashing algorithm

The encryption algorithm is a symmetric encryption algorithm

User

Passcode

hashing algorithm

Generated encryption key

hashing algorithm

Generated validation hash

test for match

Stored validation hash

Server

If test succeeds, decryption may proceed. If test fails, stop processing.

encryption algorithm

Encrypted data

Decrypted data

User edits data

encryption algorithm

Photo Encryption Diagram

```
                    ┌──────────────┐
                    │    Start     │
                    └──────────────┘
                           │
                           ▼
        ╱─────────────────────────────────────╲
       ╱            Plain image                 ╱          ▲
      ╱───────────────────────────────────────╱           │
                           │                               │
                           ▼                               │
              ┌─────────────────────────┐                  │  Encryption process
              │   Import data from      │                  │
              │     plain image         │                  │
              └─────────────────────────┘                  │
                           │                               │
                           ▼                               │
         ┌─────────────────────────────────┐               │
         │ Encrypt Image using Algorithm   │               ▼
         └─────────────────────────────────┘
                           ▲
        ╱─────────────────────────────────────╲
       ╱          Encrypted image              ╱
      ╱───────────────────────────────────────╱
                           │                               ▲
                           ▼                               │
              ┌─────────────────────────┐                  │
              │ Import data from encrypted│                │
              │        image             │                 │
              └─────────────────────────┘                  │  Decryption process
                           │                               │
                           ▼                               │
         ┌─────────────────────────────────┐               │
         │ Dencrypt Image using Algorithm  │               │
         └─────────────────────────────────┘               │
                           │                               │
                           ▼                               │
        ╱─────────────────────────────────────╲            │
       ╱            Plain image                 ╱           ▼
      ╱───────────────────────────────────────╱
                           │
                           ▼
                    ┌──────────────┐
                    │    Stop      │
                    └──────────────┘
```
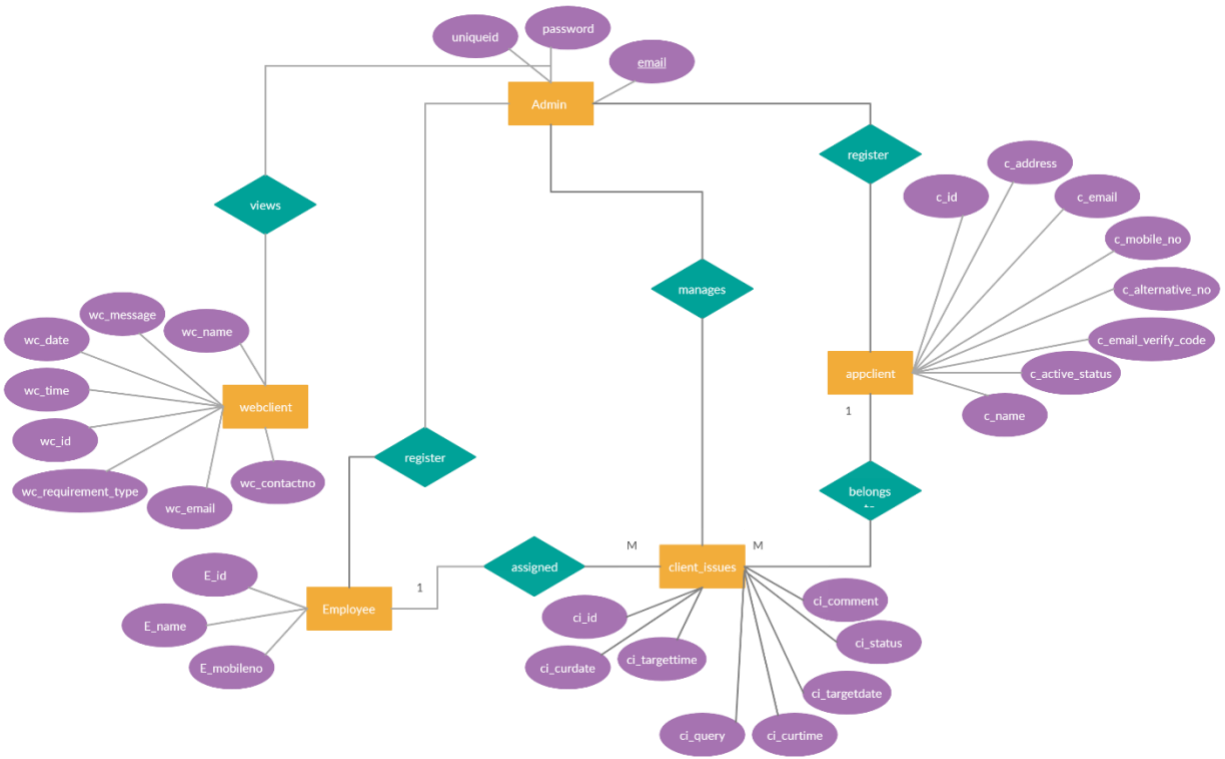
30

User Access diagram

Relationship Diagram

c

Login

String
uname,password

Login()

System

Login()

Registration()

Text   Encryption

String key,text

Encryption()

Text decryption

String   key

Decryption()

Registration

String name

Registration()

Image encryption

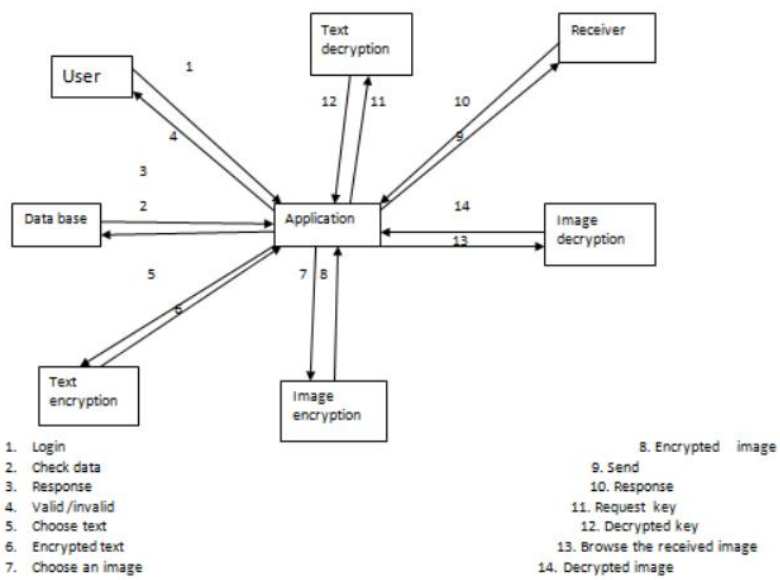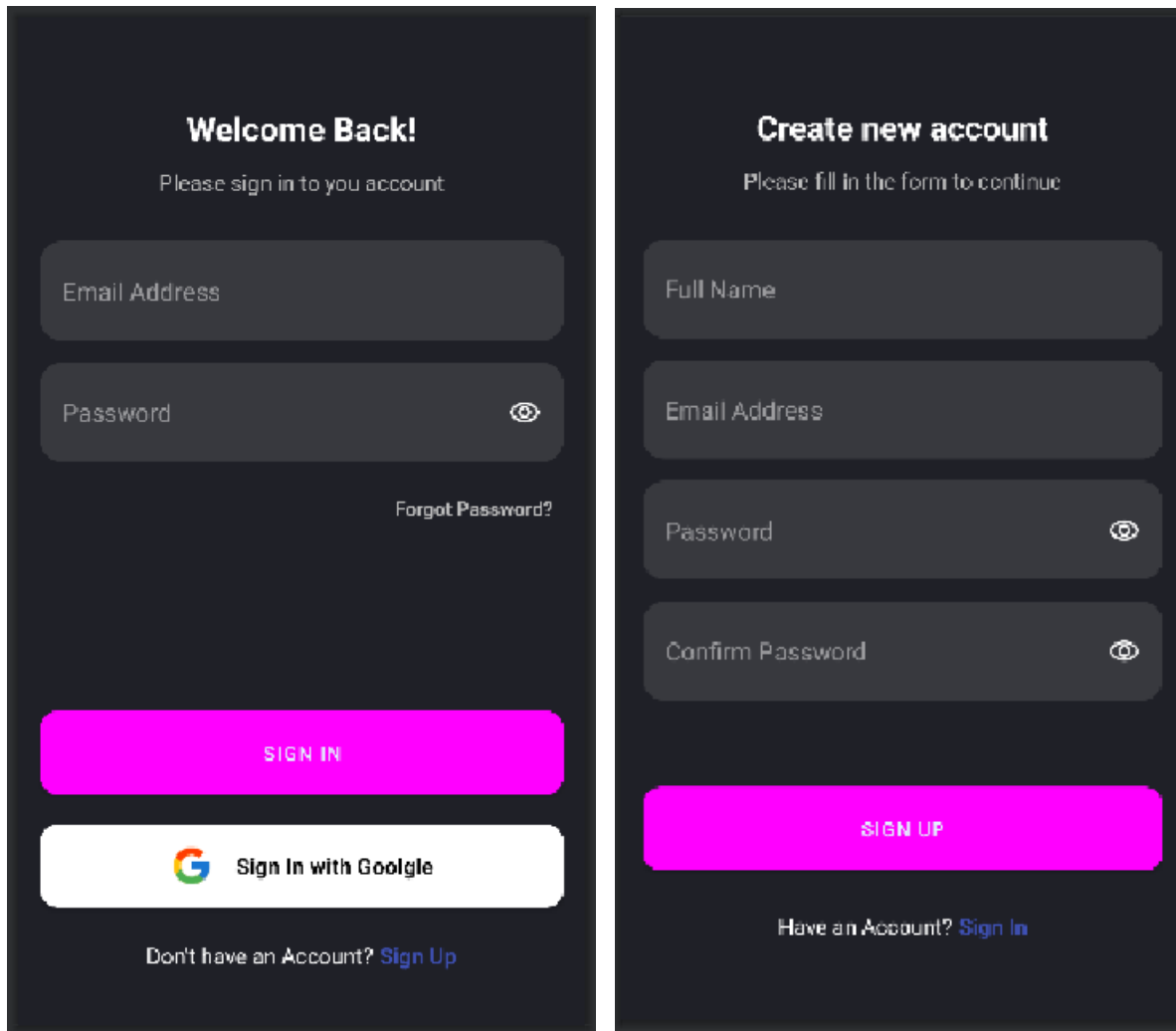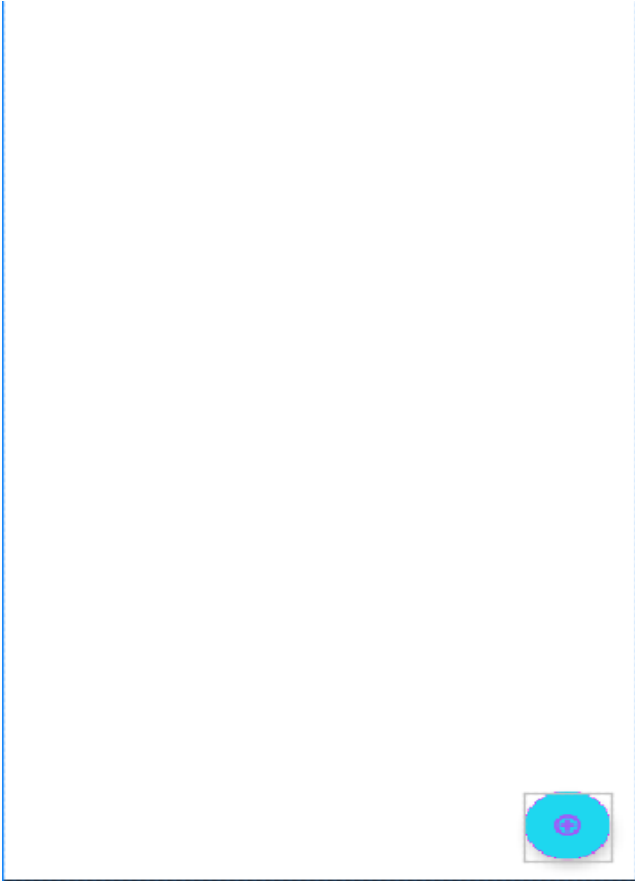Photo,code,text

Image encryption()

Send()

Browse()

Image decryption

Getphoto,encrypted   key

Show()

Browse()

Collaboration Diagram:

1. Login
2. Check data
3. Response
4. Valid /invalid
5. Choose text
6. Encrypted text
7. Choose an image

8. Encrypted   image
9. Send
10. Response
11. Request key
12. Decrypted key
13. Browse the received image
14. Decrypted image

Chapter 3

Application Images

Screenshot 1 — SignupActivity.java:
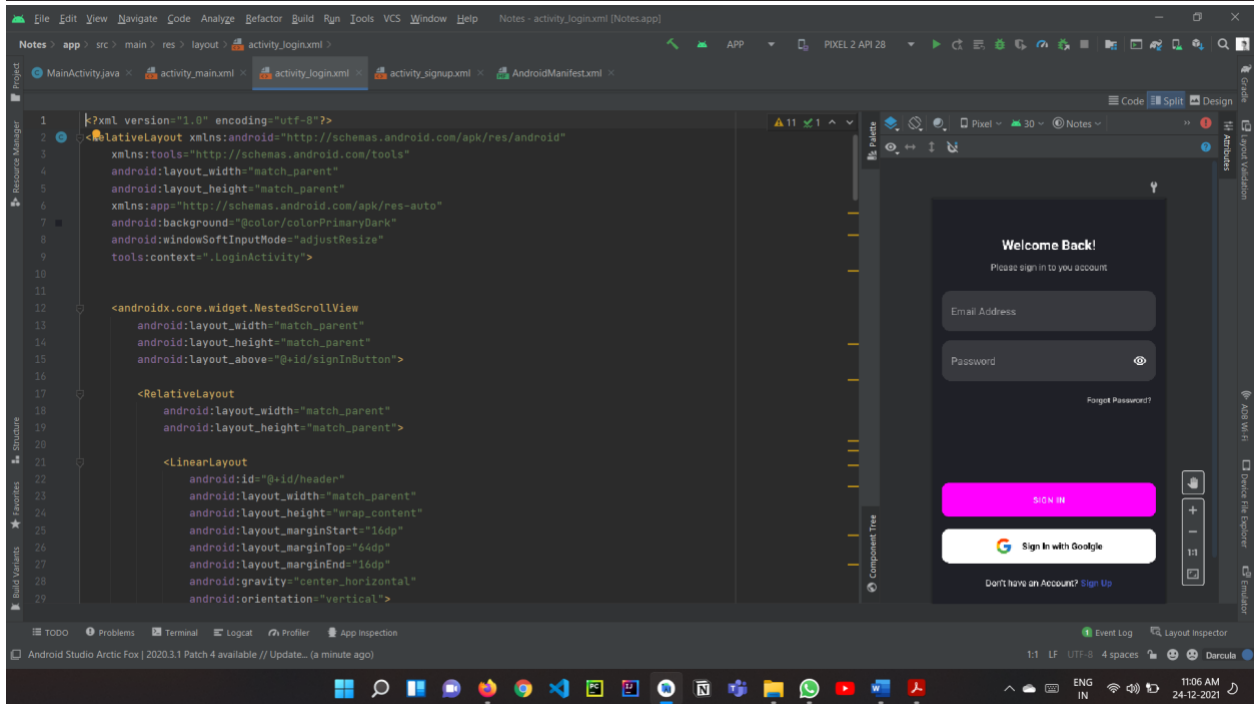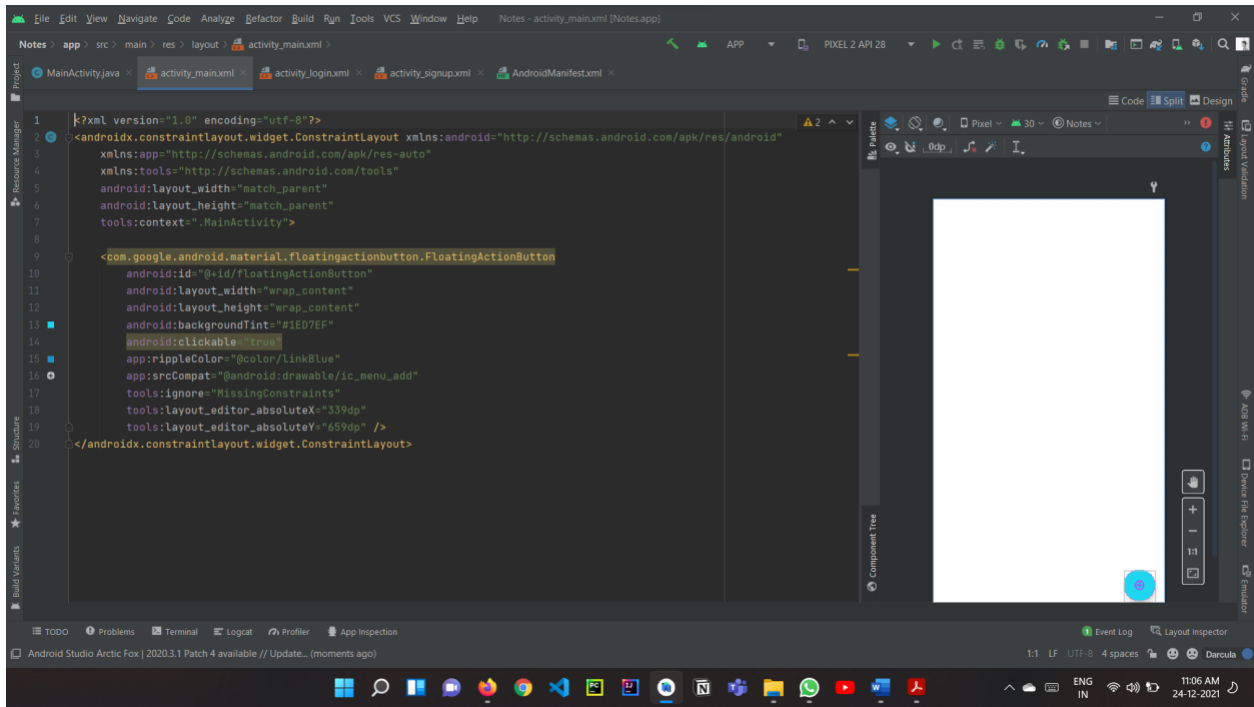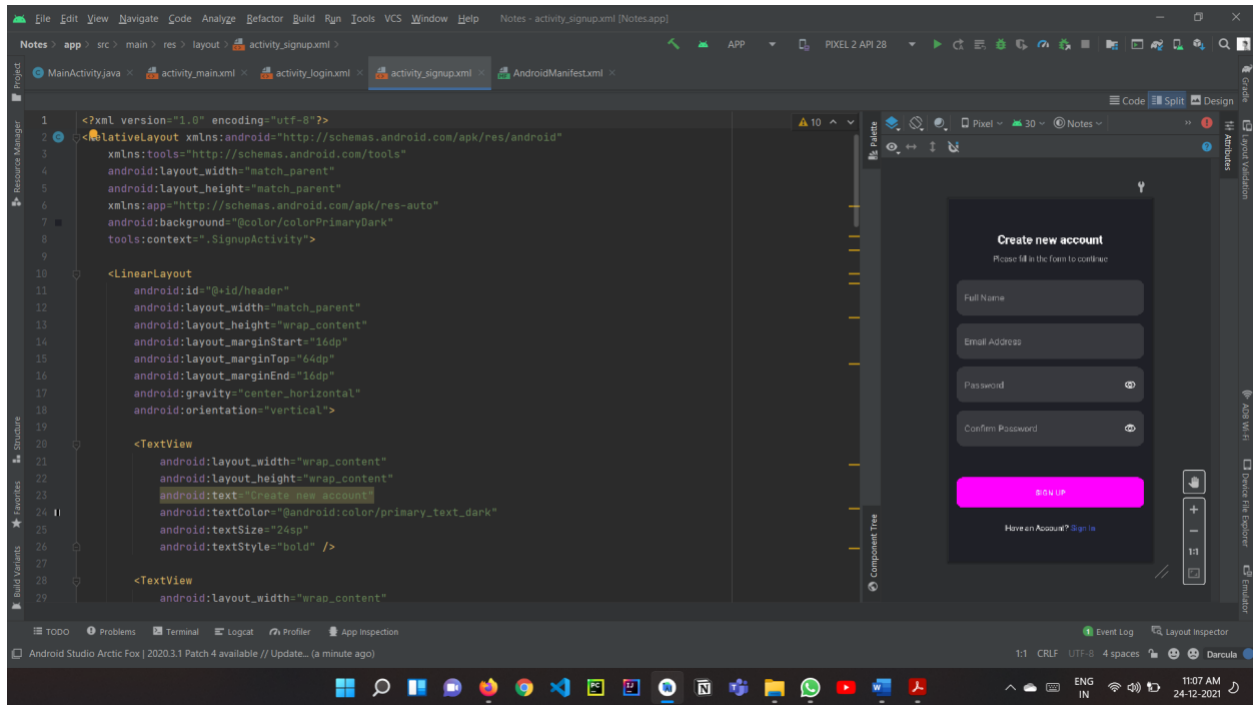
```java
package com.example.notes;

import ...

public class SignupActivity extends AppCompatActivity {

    private MaterialButton signUpButton;
    private EditText nameEditText, emailEditText, passwordEditText, confPassEditText;
    private TextView signInTextView;
    private boolean isPasswordVisible = false;
    private boolean isConfPassVisible = false;

    @SuppressLint("ClickableViewAccessibility")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_signup);
        widgets();

//        signUpButton.setOnClickListener(new View.OnClickListener() {
//            @Override
//            public void onClick(View v) {
//                if (emailEditText.getText().toString().isEmpty() || nameEditText.getText().toString().isEmpty()
//                        || passwordEditText.getText().toString().isEmpty() || confPassEditText.getText().toString().isEmpty()) {
//                    Toast.makeText(SignupActivity.this, "Empty Fields are not allowed", Toast.LENGTH_LONG).show();
//                } else if (!Utils.isValidEmail(emailEditText.getText().toString())){
//                    Toast.makeText(SignupActivity.this, "Email is not valid", Toast.LENGTH_SHORT).show();
//                }else if (Objects.equals(passwordEditText.getText().toString(), confPassEditText.getText().toString())) {
//                    mAuth.createUserWithEmailAndPassword(emailEditText.getText().toString(), passwordEditText.getText().toString())
//                            .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
```

Screenshot 2 — LoginActivity.java:

```java
package com.example.notes;

import ...

public class LoginActivity extends AppCompatActivity {

    private MaterialButton buttonLogin;
    private CardView buttonGoogleLogin;
    private EditText editTextEmail, editTxtPassword;
    private TextView forgotPasswordTV;
    private TextView signUpTV;
    private boolean isPasswordVisible = false;

    @SuppressLint("ClickableViewAccessibility")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        initViews();

//        buttonLogin.setOnClickListener(new View.OnClickListener() {
//            @Override
//            public void onClick(View v) {
//                if (editTextEmail.getText().toString().isEmpty() || editTxtPassword.getText().toString().isEmpty()) {
//                    Toast.makeText(LoginActivity.this, "Empty Fields are not allowed", Toast.LENGTH_LONG).show();
//                    return;
//                } else if (!Utils.isValidEmail(editTextEmail.getText().toString())) {
//                    Toast.makeText(LoginActivity.this, "Email is not valid", Toast.LENGTH_SHORT).show();
```

The project was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- The System has scope for modification in future if it is necessary.

- Updation of information of customer becomes so easier.

- It provides a friendly graphical user interface which proves to be better when compared to the existing system.

CONCLUSION

Make sure you capture both the key concepts and the relevant details, whether your notes are long or short. Spend a few minutes after class summarizing your notes, or revising and filling up gaps in your notes—it will help you remember what you learned in class. Reviewing your notes right before your next lesson can assist you in making connections across classes.

Notes that you may use to study for tests, compose papers, and start projects are good notes.

REFERENCES

Evernote & Penultimate

OneNote

Notability

Google Keep

[1] Academic Success Center

[2] Bauer, A., Koedinger, K.R. (2007). Selection-based note-taking applications. *Human-Computer Interaction Institute*.

[3] Mueller, P.A., Oppenheimer, D.M. (2004). The pen is mightier than the keyboard: Advantages of longhand over laptop note-taking. *Psychological Sciences, 25(6),* 1159–1168.

[4] Herbert, W. (2014). Ink on paper: Some notes on note taking*. *Association for psychological science.*

[5] Andres, L., Zentern, A., Zentner, J. (2014). Measuring the effect of internet adoption on paper consumption.*Policy Research Working Paper.*

[6] Molla, R. (2014). Remember pens and pencils? They're doing just fine. *The Wall Street Journal.*

Acronyms

B.Tech.	Bachelor of Technology
Encryption	For Encrypting Data for privacy and security
Hashing Algorithms	For Storing the data