Green University of Bangladesh
Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: (Fall, Year: 2021), B.Sc. in CSE (Eve)

Course Title: Mobile Application Development Lab

Course Code: CSE 402 Section: 191-EB

Lab Project Name: Secure Text

Student Details

Name		ID
1.	Md Atickur Rahman	191015121
2.	Pallab Kumar Paul	191015018

Submission Date	**: December 30, 2021**
Course Teacher's Name	**: Md. Fahim Arefin**

[For Teachers use only: Don't Write Anything inside this box]

Lab Project Status

Marks: ………………………………… **Signature: ....**

Comments: .... **Date: ....**

Table of Contents

Chapter 1 Introduction..2

1. Introduction..3

2. Design Goals/Objective..3

Chapter 2..4

Design/Development/Implementation of the Project..4

1. Features..4

2.1.Subsection..4

2. Challenges Face..10

2.1.Subsection..10

Chapter 3 Performance Evaluation..11

1. Output..11
2. Discussions..14
3. Bugs..14

Chapter 4 Conclusion..15

1. Introduction..15
1. Practical Implications..15
2. Scope of Future Work..15

References..16

Chapter-1

Introduction

1. Introduction

The use of mobile devices is increasingly rise in our daily life and many information is exchanged and stored on these devices. Thus, there are some risks that must be prevented as information leakage. So, we need some protection of our communication between sender & receiver.
Here, we try to develop an application to perform this security function, using some algorithm, implemented on the Android platform, enabling the user data encryption & decryption of our devices.

2. Design Goals/Objective

A. To understand how to represent an app in Android Studio.
B. Implement Encryption XML & JAVA code via real life and gaining an idea how it would work.
C. Implement Decryption XML & JAVA code via real life and gaining an idea how it would work.
D. Implement Button, Text-view, Toast, on Click Listener, Animation, android Intents and many more features in Android Studio via real life and gaining an idea how it would work.

Chapter 2

Design/Development/Implementation of the Project

1. Features

1.1.Simplicity: Here, we try to maintain to design the all XML as simple & gorgeous way. We, avoid any unnecessary text or text fields or buttons that decrease user satisfaction level. We try to implement all the features as user can access & get their information quickly and easily. We, maintain standard gap between buttons, text fields, text views, images etc. We maintain clear uncluttered screens with obvious leads to the next step in the app process.

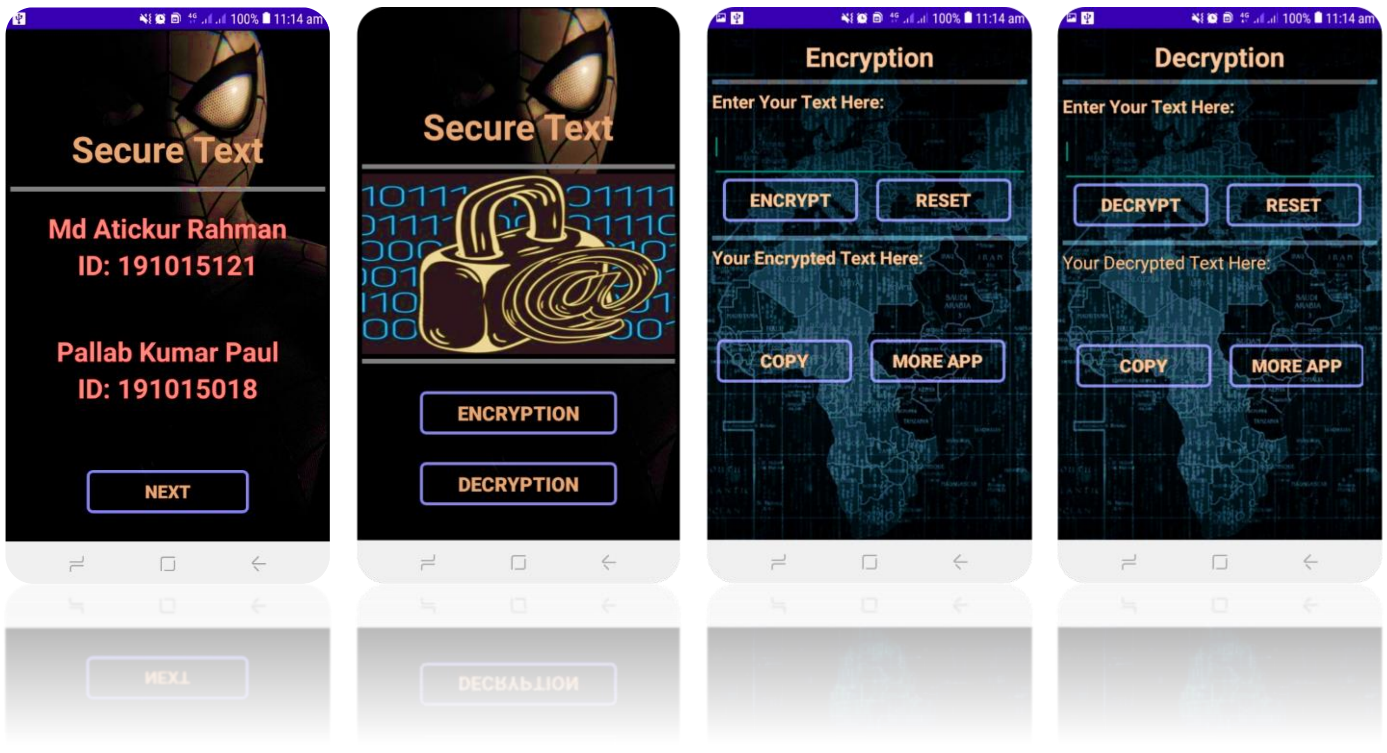


Fig: 1.1. Maintain Simplicity

a. **Text View:** In all text view we set required layout height, width, gravity, text color, text size and style. In all text field we set required text. Such as- in 1st window first we set in text field our application name- “Secure Text”, then we give our Name & ID in next text fields.

b. **View:** We implement many view to separate our text view in many window and also separate our view flipper. Here, we implement our required layout width, layout height and background color. We set individual ID for every view to identify this view uniquely.

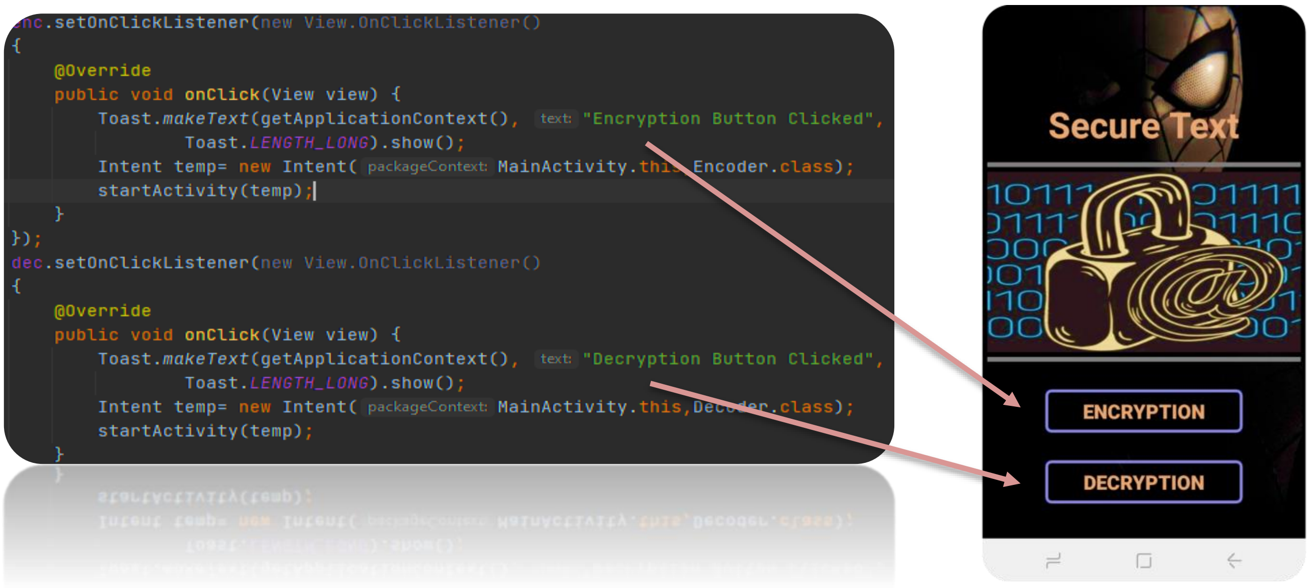
c. **Button:** Here, we implement many buttons as our needs. In XML part we set its required width, height, gravity, margin, background, text size and color. We set individual ID for every button to identify this button uniquely. We set button name as our needs. We, set on click listener for every button and make their work perfect.

d. **Edit Text:** Here, we implement 2 edit text in encryption page & decryption page. We set required layout height, width, gravity, text color, text size and style for this edit text. We set individual ID for every edit text to identify this uniquely.

1.2. Speed: We try to maintain fast loading screens. We avoid big size of data or text or any elements that’s harmed our application speeds. No user likes waiting, particularly when they have to look at screen. We try to keep it simple and keep it quick. We ensure that user don’t get any types of speed problem when they use our application.

1.3. Good Image Resolution: We try to maintain good resolution picture use in our app. We, use many pictures as background & view flipper part and we use high resolution image in this. In this part we try to maintain user satisfaction. We test our app in different screen and improve our image resolution.

1.4. On Click Listener: To define the click event handler for every button, we use android: on Click Listener to the button element in our XML & java layout. We set the value for this attribute must be the name of the method we want to call in response to a click event. The Activity hosting the layout must then implement the corresponding methods.



```
nc.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View view) {
        Toast.makeText(getApplicationContext(),  text: "Encryption Button Clicked",
                Toast.LENGTH_LONG).show();
        Intent temp= new Intent( packageContext: MainActivity.this,Encoder.class);
        startActivity(temp);
    }
});
dec.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View view) {
        Toast.makeText(getApplicationContext(),  text: "Decryption Button Clicked",
                Toast.LENGTH_LONG).show();
        Intent temp= new Intent( packageContext: MainActivity.this,Decoder.class);
        startActivity(temp);
    }
```

Fig: 1.4 on Click Listener

1.5. View Flipper: It is a simple view animator that will animate between two or more views that have been added to it. Only one child is shown at a time. If requested, can automatically flip between each child at a regular interval. We set view flipper in main activity XML & Java layout. We, create an image array and store 3 image. We set time interval 3 second of every image & this image are coming from left part in screen & out from right part in screen.

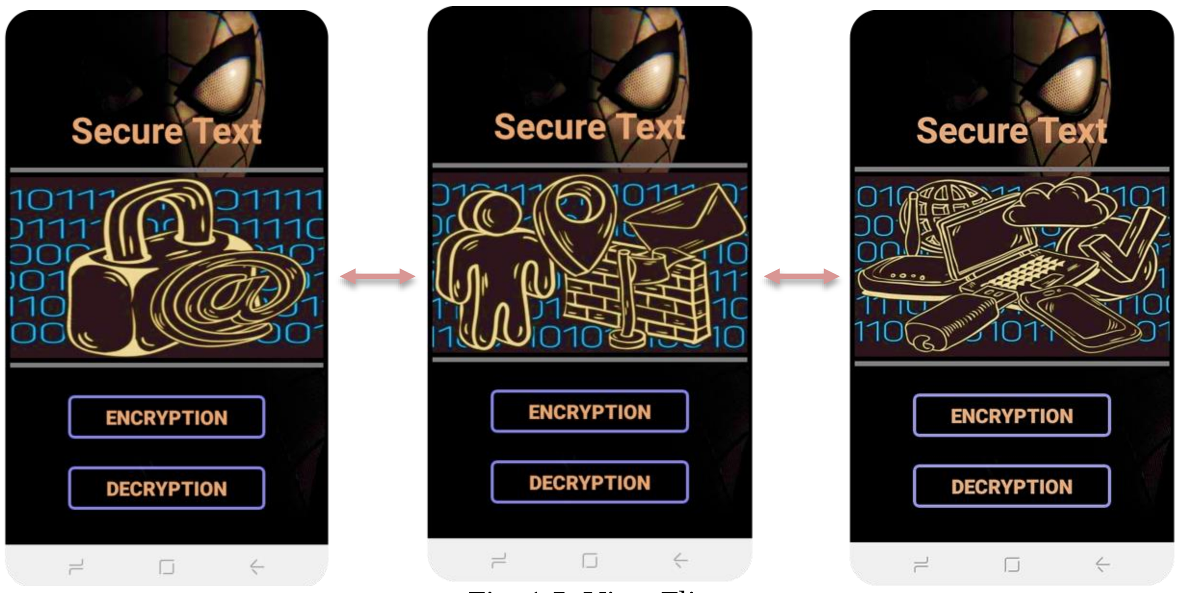
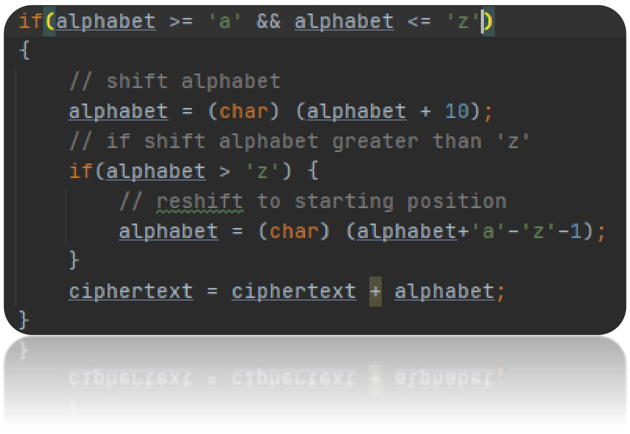


Fig: 1.5. View Flipper

1.6. Security: Unfortunately our apps main working principle is maintain security in our text. So, we works lot of place to maintain our security & achieving our goals. If user use this app they can secure their text & hacker don't understand the data if they get it. So, user data remain in safe zone.

1.7.Encryption: In cryptography, encryption is the process of encoding information. This process converts the original representation of the information, known as plaintext, into an alternative form known as ciphertext. Ideally, only authorized parties can decipher a ciphertext back to plaintext and access the original information. Here, we try to implement this technique. We use simple Ceasar Cypher algorithm for encrypt our text & here our key is 10. For encrypt any word we add 10 and modulus by 26.



```
if(alphabet >= 'a' && alphabet <= 'z')
{
    // shift alphabet
    alphabet = (char) (alphabet + 10);
    // if shift alphabet greater than 'z'
    if(alphabet > 'z') {
        // reshift to starting position
        alphabet = (char) (alphabet+'a'-'z'-1);
    }
    ciphertext = ciphertext + alphabet;
}
```

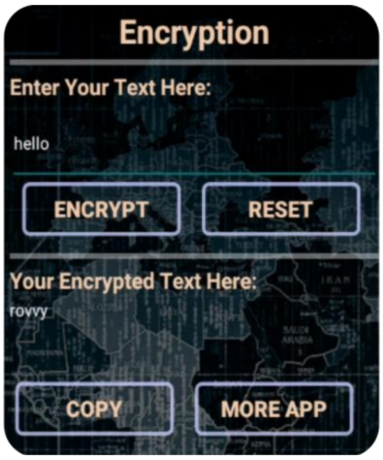
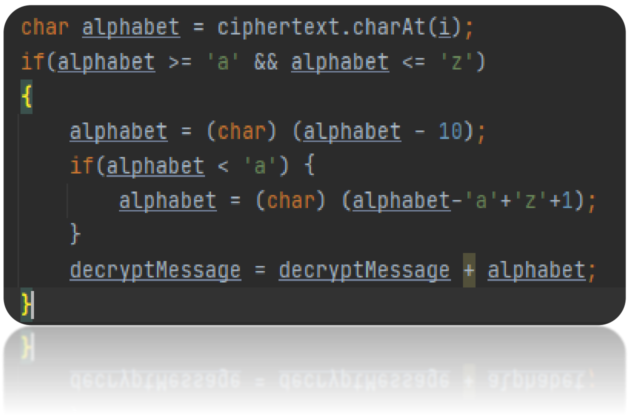


Fig: 1.7. Encryption Input

1.8.Decryption: Decryption is the process of transforming data that has been rendered unreadable through encryption back to its unencrypted form. In decryption, the system extracts and converts the garbled data and transforms it to texts and images that are easily understandable not only by the reader but also by the system. Here, we try to implement this technique. We use same Ceasar Cypher algorithm for decrypt our text & here our key is 10. For decrypt any word we subtract 10 and modulus by 26.



```
char alphabet = ciphertext.charAt(i);
if(alphabet >= 'a' && alphabet <= 'z')
{
    alphabet = (char) (alphabet - 10);
    if(alphabet < 'a') {
        alphabet = (char) (alphabet-'a'+'z'+1);
    }
    decryptMessage = decryptMessage + alphabet;
}
```

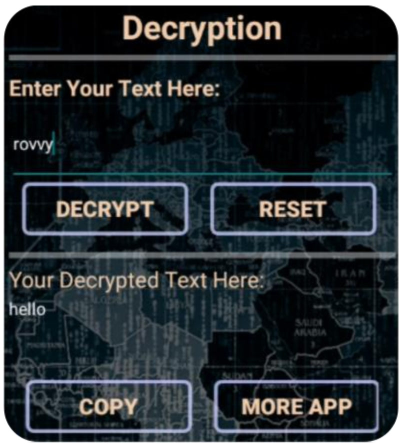
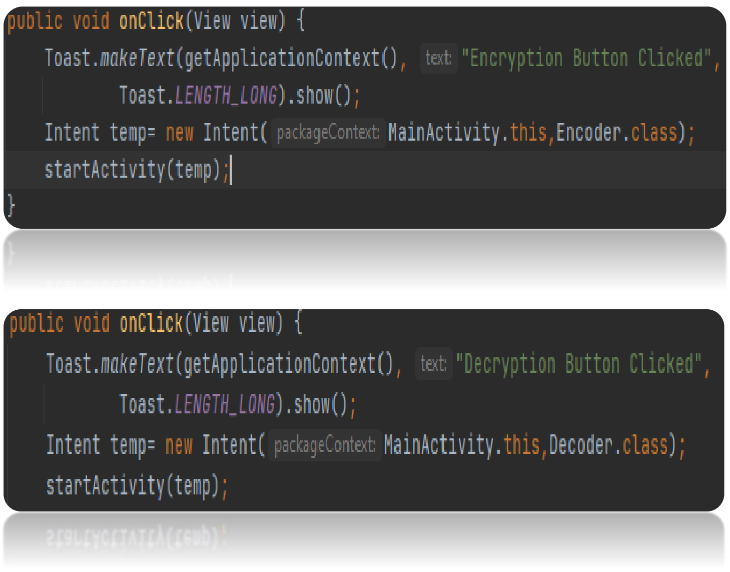


Fig: 1.8. Decryption Input

1.9. Push Notifications: In our applications we try to push some notifications using Toast. When we click any button we see notifications in bellow. Such as in below if we clicked Encryption button system show notification "Encrypted Button Clicked" and if we clicked Decryption button system show notification "Decryption Button Clicked".



```
public void onClick(View view) {
    Toast.makeText(getApplicationContext(), text: "Encryption Button Clicked",
            Toast.LENGTH_LONG).show();
    Intent temp= new Intent( packageContext: MainActivity.this,Encoder.class);
    startActivity(temp);
}
```

```
public void onClick(View view) {
    Toast.makeText(getApplicationContext(), text: "Decryption Button Clicked",
            Toast.LENGTH_LONG).show();
    Intent temp= new Intent( packageContext: MainActivity.this,Decoder.class);
    startActivity(temp);
```

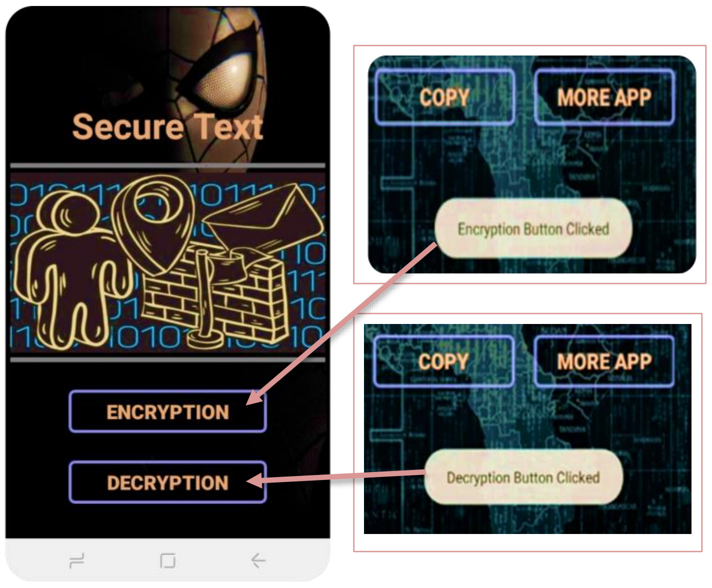
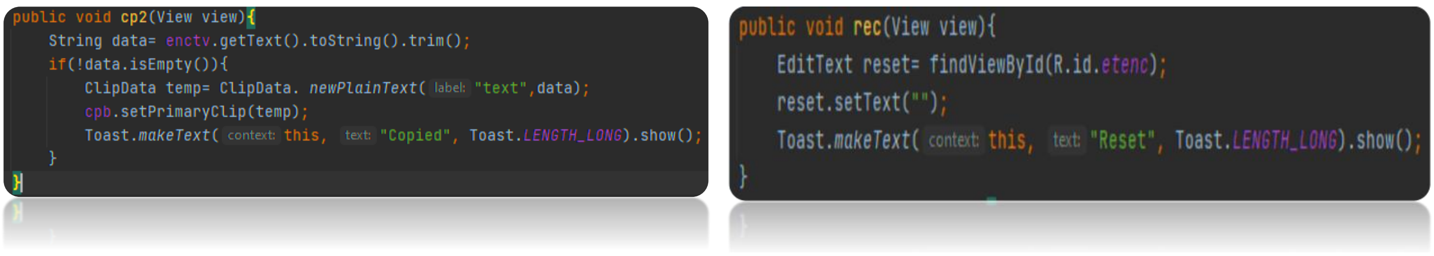


Fig: 1.9. Push Notifications

1.10. Copy & Reset: We use copy button to copy text view & edit text field data. We use Reset button to reset text view & edit text field data. To copy any text field data we use get text method & to reset any text field data we use set text method.



```
public void cp2(View view){
    String data= enctv.getText().toString().trim();
    if(!data.isEmpty()){
        ClipData temp= ClipData. newPlainText( label: "text",data);
        cpb.setPrimaryClip(temp);
        Toast.makeText( context: this,  text: "Copied", Toast.LENGTH_LONG).show();
    }
}
```

```
public void rec(View view){
    EditText reset= findViewById(R.id.etenc);
    reset.setText("");
    Toast.makeText( context: this,  text: "Reset", Toast.LENGTH_LONG).show();
}
```

Fig: 1.10. Copy & Reset

1.11. Intents: An intent is to perform an action on the screen. It is mostly used to start activity, send broadcast receiver, start services and send message between two activities. In Encoder java part we create relations between all applications & in Decoder java part we create relations between only email applications.

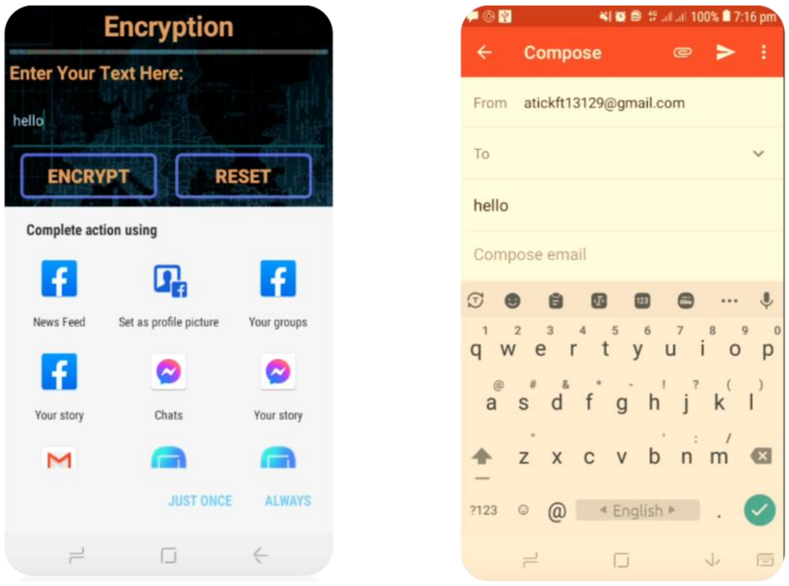


Fig: 1.11. Android Intents

2. Challenges Face

To implement our application we faces many challenges.

2.1. Our first challenges is lack of resources & platforms. We don’t get enough resource to improve our work or if we face any problem we don’t get enough resource to solve the problem.

2.2. Our next challenges is wrong assessing and development directions. Sometimes we move on wrong direction, we faces many errors. Most we overcome & sometimes we avoid it.

2.3. Our next challenges is to maintain syntax. Sometimes we faces logical & syntax error. We use debug from overcome it.

2.4. To implement encryption & decryption java code we faces lots of challenges. We use simple Ceasar cypher algorithm & we apply this only character in a to z and A to Z. Here we avoid integers 0 to 9 and special characters.

2.5. To implement main activity java code we faces many challenges. Lots of functions, buttons, text fields, edit text in this layout. Maintain all this features are totally challenging.

Chapter 3

Performance Evaluation

1. Output

Output measure that our Application is perfectly run or not. Let's check it one by one.

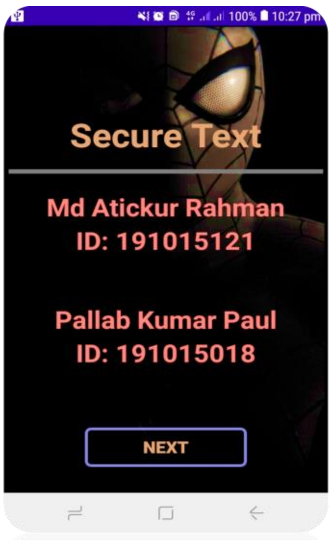


Fig: 1. 1st window

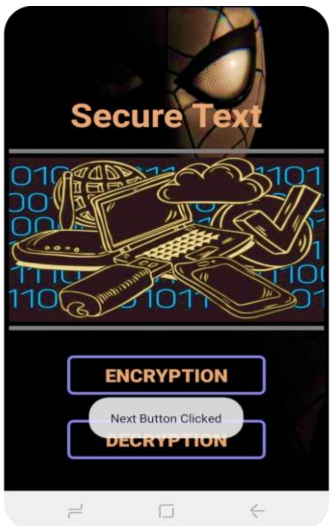


Fig: 1. 2nd window

In 1st window we see that, our text view, view & button are perfectly placed.
In Next button I use toast notification & link this layout into 2nd window. We see that if we tap on Next button system show notification- "Next Button Clicked". So, my toast option work perfectly. Again when I click next button it move on new window that is my 2nd window. So, my on click listener work perfectly.

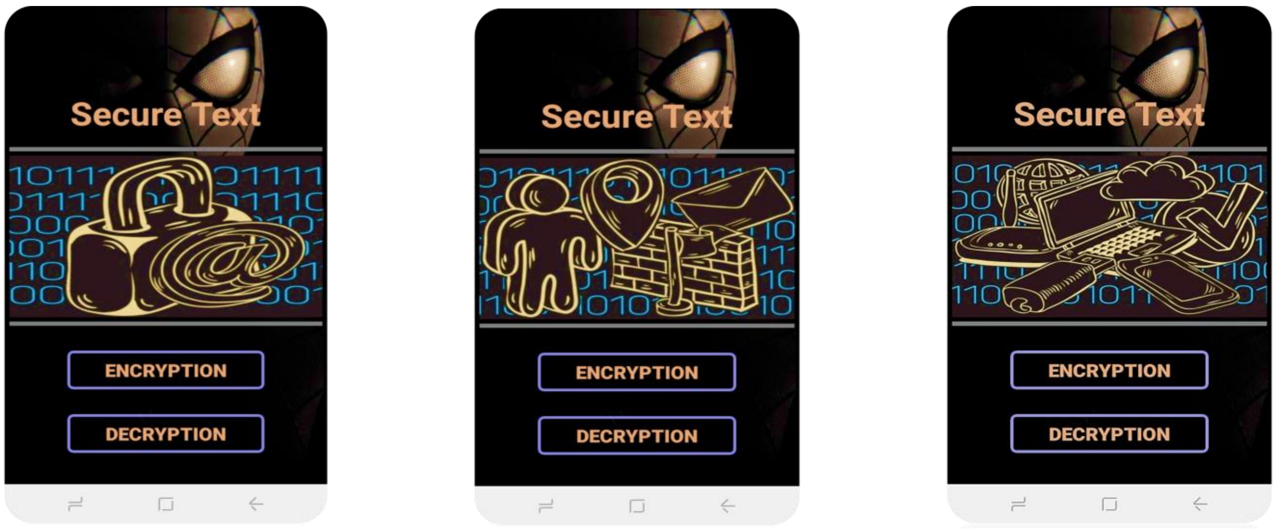


Fig: Flipper Check

In this 2nd window we see that, our text view, flipper & buttons are perfectly placed.
Here in flipper part I implement 3 picture & I see they animated perfectly.
In Encryption & Decryption button I set toast notifications & I see that those notifications are working perfectly. When I click Encryption button it move into Encrypt XML as I need and when I click Decryption button it move into Decrypt XML as I need. So, here all my functions are working perfectly.

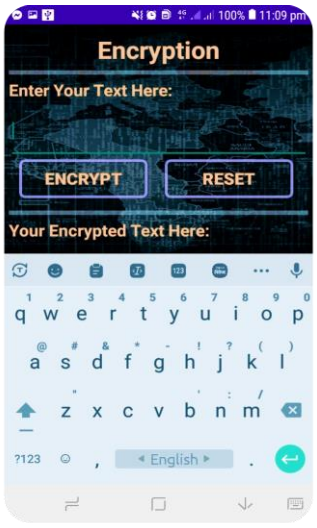


Fig: Edit Text Check

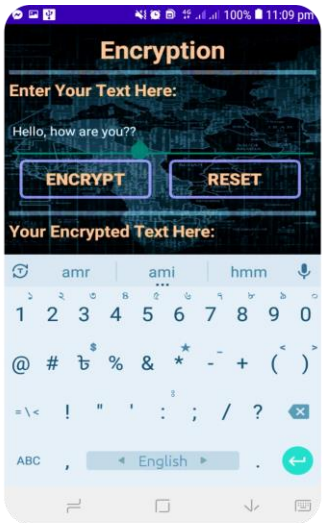


Fig: Edit Text Check

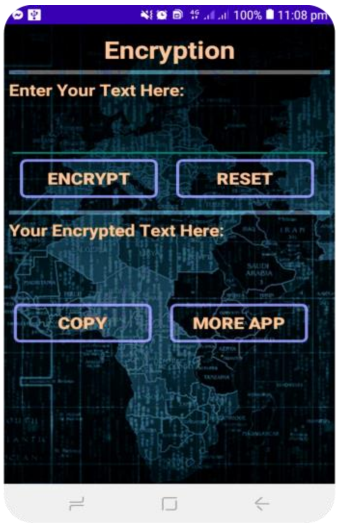


Fig: Reset Button Check

In this window we see that, our text view, view & button are perfectly placed.
1st we see that we have edit text field. In this field we type our required text that we want to encrypt. Here, we type- "Hello, how are you??"
When we click Reset button our text field text are automatically erase. So, here all this elements works perfectly.

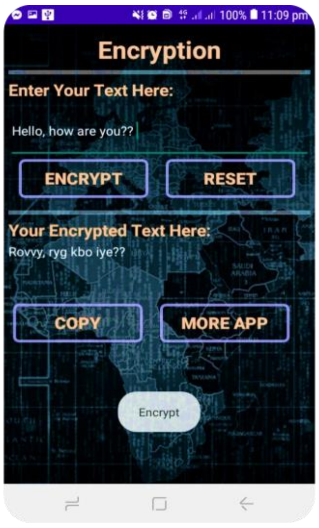


Fig: Encrypt Check

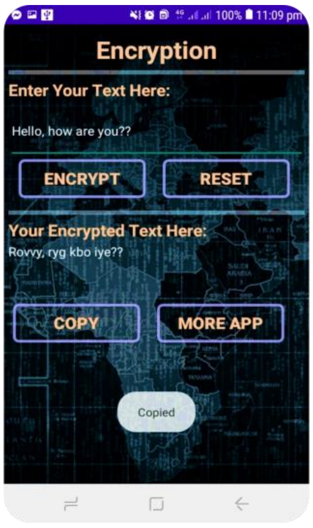


Fig: Copy Check

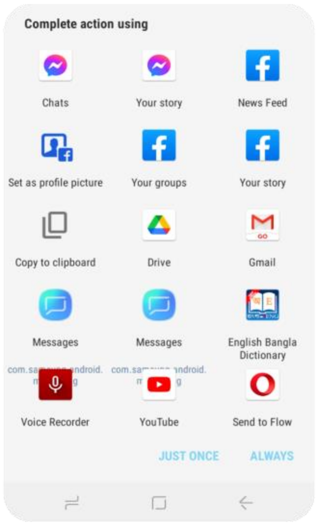


Fig: Intents Check

When we click Encryption button system automatically encrypt our given text for rules of Chaser cypher. Here, our encrypted text is: “Rovvy, ryg kbo iye”.
Again when I click on copy button system automatically copied this text and show the notifications.
Again when I click more app button system show all apps in my mobile, so our intents works perfectly. So, here all this elements works perfectly.

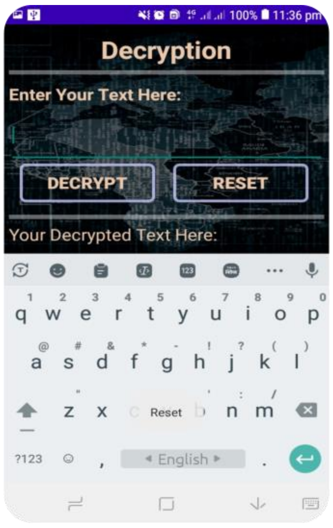


Fig: Edit Text Check

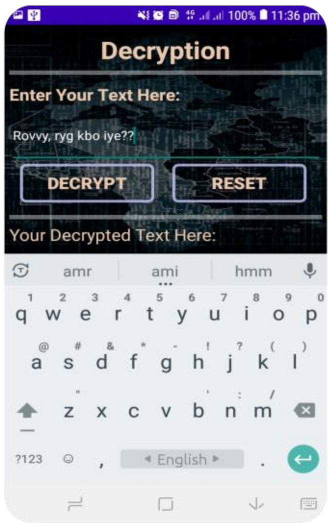


Fig: Edit Text Check

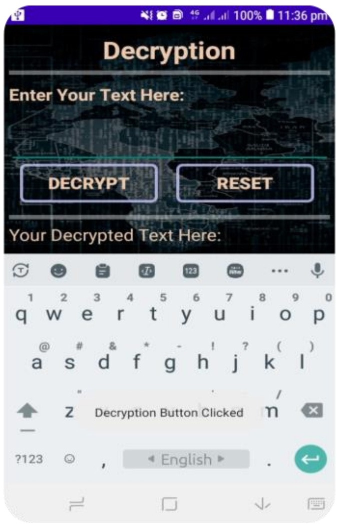


Fig: Reset Button Check

In this window we see that, our text view, view & button are perfectly placed.
1st we see that we have edit text field. In this field we type our required text that we want to decrypt. Here, we type- “Rovvy, ryg kbo iye??”
When we click Reset button our text field text are automatically erase. So, here all this elements works perfectly.

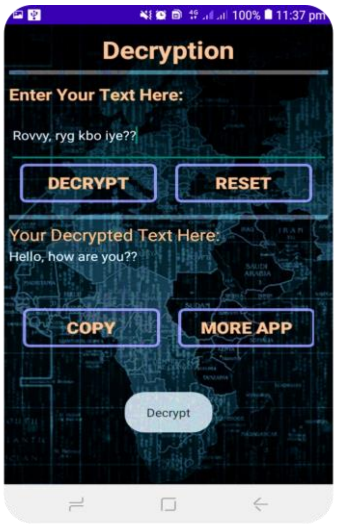


Fig: Decrypt Check

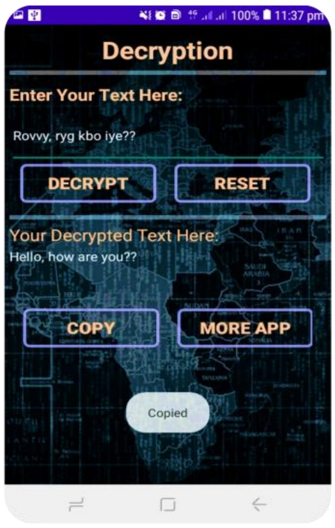


Fig: Copy Check

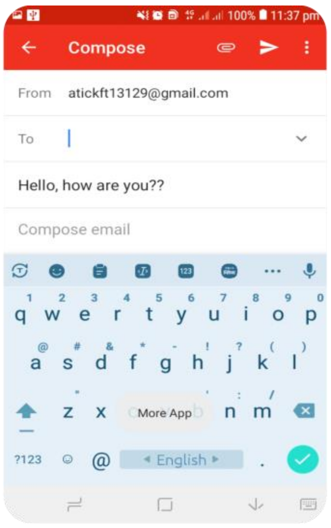


Fig: Intents Check

When we click Decryption button system automatically decrypt our given text for rules of Chaser cypher. Here, our Decrypted text is: “Hello, how are you??” So our decryption algorithm works perfectly.
Again when I click on copy button system automatically copied this text and show the notifications.
Again when I click more app button system show email apps in my mobile, because here I link up only mail app using intents. So our intents works perfectly.
So, here all this elements works perfectly.
To, check all output we say that our every elements for all XML and Java layouts works perfectly.

2. Discussions

2.1.Achievements

To make a plan we started our project & it is our big achievement that we able to fully implement our plan in this project. That is totally amazing.

2.2.Bugs

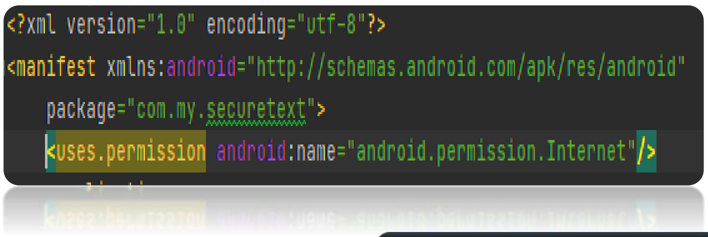
To implement this project we faces many problems. Something I remember & something I forgot.

2.6. We faces problem to implement android intents, my class teacher solve my problem.

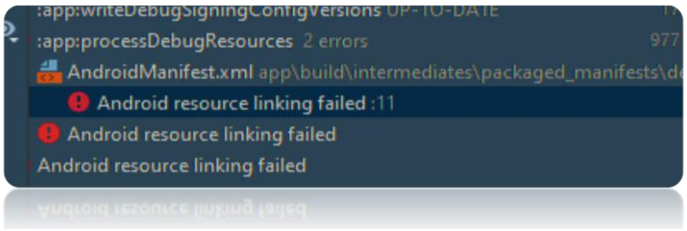
2.7. We faces some problem to implement on Click listener & buttons, using some reference we solve it.

2.8. We faces some problem to implement drawable a layout, which name is outframe1, we can’t solve it. Sometimes we avoid this layout.

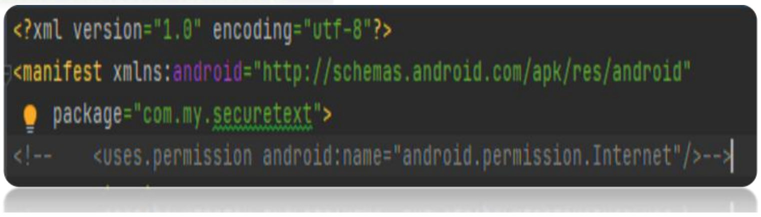
2.9. We faces some problem to implement Android Manifest XML part & we can’t solve it. So, we put it on comment & avoid it.



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.my.securetext">
    <uses.permission android:name="android.permission.Internet"/>
```



```
:app:writeDebugSigningConfigVersions UP-TO-DATE
:app:processDebugResources  2 errors                                   977
  AndroidManifest.xml app\build\intermediates\packaged_manifests\de
    Android resource linking failed :11
  Android resource linking failed
Android resource linking failed
```



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.my.securetext">
<!--    <uses.permission android:name="android.permission.Internet"/>-->
```

Fig: 2.4. Internet Permission Errors

Chapter 4
Conclusion

1. Introduction

Here, we try to develop an application to perform security function, using Some algorithm, implemented on the Android platform, enabling the user Data encryption & decryption of our devices.

2. Practical Implications

We gain lots of idea about Android studio. To clear our basic knowledge This project helps us lot. We, implement practically many the features, that Increase our satisfaction level.

3. Scope of Future Work

Here, this project is created for only educational purpose. We considered the most Important requirements only, many more features and details can added to Our project in order to obtain more user friendly applications. In future I can Upgraded it and may become part of amazing android life.
Implement coding I use some different conditions, variables and functions To complete my work perfectly and I always avoid syntax and logical error.

References

1. https://www.investopedia.com/terms/e/encryption.asp#:~:text=Encryption%20is%20a%20means%20of,makes%20the%20original%20information%20unreadable
2. https://www.envertis.com.au/android-app-development-features/
3. https://developer.android.com/guide/components/intents-common#ViewMap
4. https://stackoverflow.com/questions/16927103/onclicklistener-in-android-studio
5. https://www.techopedia.com/definition/1773/decryption
6. https://javahungry.blogspot.com/2019/02/caesar-cipher-program-in-java.html