CryptoNotes

A PROJECT REPORT

Submitted by

Kishan Khirasariya

Yagni Patel

In fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

Computer Engineering

LDRP Institute of Technology and Research, Gandhinagar

Kadi Sarva Vishwavidyalaya

April 2022

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **"CryptoNotes"** has been carried out by **Kishan Khirasariya (18BECE30087)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2021-22.

KSV
KADI SARVA VISHWAVIDYALAYA

Prof. Jayana Kaneriya
Internal Guide
LDRP-ITR

Dr. Shivangi Surati
I\C Head of the Department
LDRP-ITR

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **“CryptoNotes”** has been carried out by **Yagni Patel (18BECE30185)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2021-22.

Prof. Jayana Kaneriya

Internal Guide

LDRP-ITR

Dr. Shivangi Surati

I\C Head of the Department

LDRP-ITR

ACKNOWLEDGEMENT

It gives us great pleasure and joy to offer this project on " CryptoNotes ". At the start of this report, we would like to express our sincere gratitude to all those who have assisted us in this endeavor. We would not have progressed in the project without their active advice, assistance, collaboration, and encouragement. We would not have progressed in the project without their active direction, assistance, cooperation, and encouragement. We'd want to take this time to express our gratitude to all of our faculty members for their support and expertise throughout the development of this project. We would like to take this occasion to express our gratitude to our institute, LDRP, and Dr. Shivangi Surti, whose guidance and assistance have been important in our academic achievement.

We also want to express our gratitude to Prof. Jayana Kaneriya, our project guide, for taking the effort to put together all the bits and pieces we sent in as raw data; her contribution to the project's success is unrivalled. We must value the advice provided by other supervisors as well as the panels, particularly regarding our project presentation, which has enhanced our presentation skills as a result of their comments and suggestions. We want to thank everyone who has contributed to the success of our project, whether directly or indirectly. We'd also like to thank our parents, who have always been our biggest supporters and cheerleaders, as well as our biggest inspirations.

Kishan Khirasariya
[18BECE30087]

Yagni Patel
[18BECE30185]

ABSTRACT

Any android user, be it a student or professional, often finds the difficulties to store notes and passwords due to security issues. Nowadays there are so many important notes we need to store somewhere safe that we can access anytime it is required like passwords, account number, etc.

The main purpose of this project is to create an application CryptoNotes, which allows us to store important notes like password without being tense about any malware attack. CryptoNotes is general app which allows you to organize text and images. It has a simple but clean look that makes it easy to work with. CryptoNotes resolves our main concern of security.

Table of Contents

List Of Figures

1 Introduction

1.1 Introduction

The main purpose of this project is to create an application CryptoNotes, that can store your notes and passwords without being tense about any malware attack. CryptoNotes is an android application that stores your notes and passwords on your local storage. As CryptoNotes is fast, efficient, and has an encryption algorithm before storing any note will play a game-changer in the note-taking applications. CryptoNotes has built-in AES-256 encryption, only uncrackable encryption algorithm in world. Apart from this CryptoNotes has a friendly simple user interface to ease down user navigation.

1.2 Basic Workflow

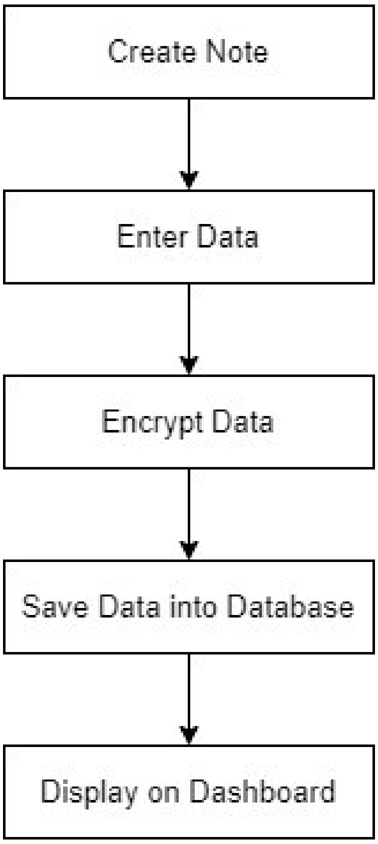


Figure 1 (Basic Workflow Diagram)

2 Technology and Literature Review

2.1 About Tools and Technology

- Java: Java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies.
- Android Studio: Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. Android Studio offers even more features that enhance your productivity when building Android apps, such as:
    - A flexible Gradle-based build system
    - A fast and feature-rich emulator
    - A unified environment where you can develop for all Android devices
- SQLite: SQLite is a lightweight and easy-to-setup relational database engine that can be easily integrated into various types of devices including portables and computers.
- Gradle: Gradle is a build automation tool for multi-language software development. It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing.

3 System Requirement Study

3.1 Software Requirements

- OS: Android 8 or Above
- Language: Java
- Android Studio
- Database: SQLite Database

3.2 Hardware Requirements

- 128 MB RAM
- 256 MB Storage
- Snapdragon 435 or higher than this

3.3 Functional Requirements

- Application must be designed in way to ensure that every user will get ultimate security on their data.
- System must provide the quality of service to user.
- Wide accuracy of the output.
- Software is very easy to use and save your time.

3.4 Non-Functional Requirements

- Performance: Output will be provided with appropriate accuracy to the user.
- Functionality: This software will deliver on the functional requirements.
- Flexibility: It provides the user to enter input easily.
- Learn Ability: The software is very easy to use and save your time.

4 Feasibility Study

In this phase, the project's feasibility is assessed, and a self-proposal is presented, along with a very generic project plan and some cost estimates. A feasibility study of the proposed system is to be carried out during system analysis. This is to ensure that the proposed system will not cause the organization any problems. A basic understanding of the system's primary requirements is essential for feasibility analysis.

The following are the dimensions of software feasibility:

- Technology: Is the project technologically feasible? Is it considered state-of-the-art? Is it possible to minimize defect to a level that meets the requirements of the application?
- Finance: Is it financially feasible? Is it possible to complete development at a price that both the software company and its client or market can afford?
- Time: Will the project's time to market be faster than the competitions?
- Resources: Is the organization equipped with the resources it requires to succeed?
- The feasibility analysis considers two major factors:
    1. Technical Feasibility
    2. Cost Feasibility

4.1 Technical Feasibility

This research is being carried out to determine the system's technological feasibility, or technical requirements. Any system that is created should not place a large burden on the available technical resources. As a result, there will be a lot of demand on the available technical resources. As a result, the client will be subjected to severe demands. Because very minor or no changes are necessary to implement this system, the designed system must have a low requirement.

The following methods can be used to assess technical feasibility:

1. NP-Complete
2. NP-Hard
3. Satisfiability

1. NP-Complete:

P Class: Class of all deterministic polynomial language problems.

NP Class: Class of all non-deterministic polynomial language problems.

NP Complete Problems are always solving within given time and space

2. NP-Hard:

These are problems for which there are no efficient solutions are found. Generally, complexity of these problems is more than P, NP, NP-Complete. These may include higher multiplicative constants, exponents terms or high order polynomial.

3. SAT (Satisfiability):

Boolean formula is satisfiable if there exists at least one way of assigning value to its variable so as to make it true and we denote it by using SAT. The problem of deciding whether given formula is satisfiable or not.

4.2 Cost Feasibility

This research is being carried out to determine the system's economic impact on the organization. The amount of money the corporation has to invest in the system's research and development is limited. It is necessary to justify the spending. As a result, the produced system came in under budget, which was made possible by the fact that most of the technologies used were freely available. The customized products were the only ones that needed to be acquired.

5 Design

5.1 System Design:

The next phase in the System Development life cycle is a system design. The designing part begins after the analysis of the system and is aimed at defining how to do the things. Any design must be constantly evaluated to ensure that it meets the requirements, is practical and workable in the given environment. If there are several alternatives, then all alternatives are, evaluated and the best possible solution is implemented.

Approaches to Design:

There are two main approaches to design, which are:

1) Data Centered Approach.

2) Process Centered Approach.

In both the approaches, the other factor cannot be ignored i.e., process cannot be ignored in data centered approach and vice versa. The data centered approach starts from data structures first and then the processes and the process centric approach aim at defining all the processes first and data structure at the end. Both the approaches have their advantages and disadvantages.

We use the Data Centered approach in the design of the system. The Data-Flow Diagram and the Entity-Relationship diagram form the basic input to the design phase. The Data Centered approach is the principal of Object-Oriented Design where a collection of data elements and its associated characteristics (processes) are defined as objects.

5.2 UML Diagrams:

5.2.1 Data Flow Diagram

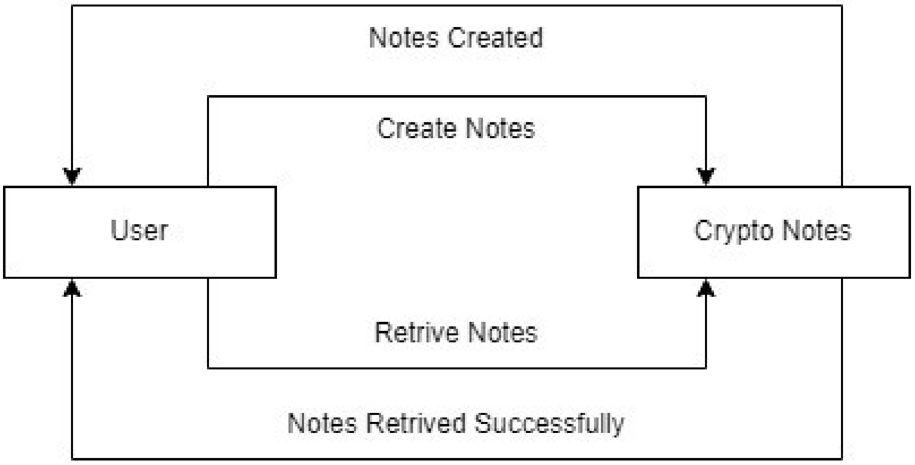


Figure 2 (0 Level DFD Diagram)

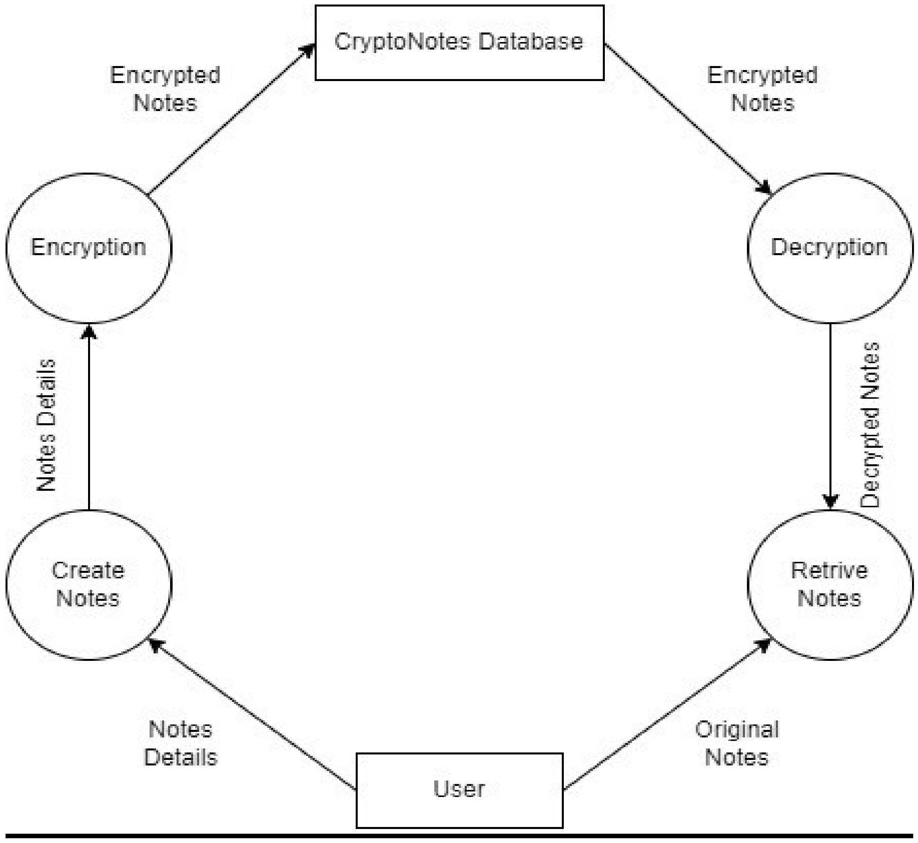


Figure 3 (1 Level DFD Diagram)

5.2.2 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases) and any dependencies between those use cases. Use case diagrams are formally included in two modeling languages defined by the Unified Modeling Language (UML).

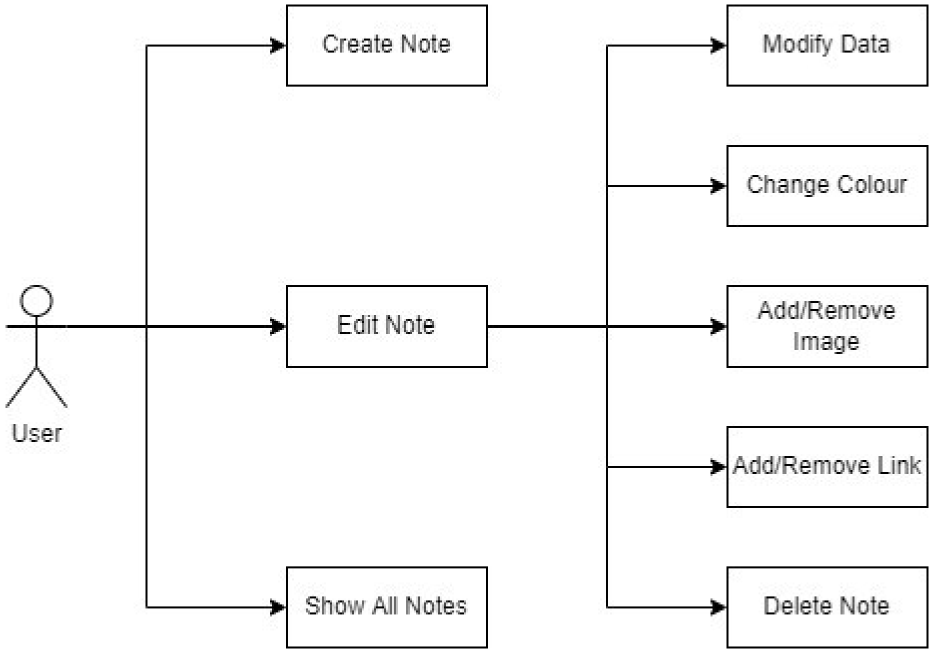


Figure 4 (Use Case Diagram)

5.2.3 Sequence Diagram

A sequence diagram is a Unified Modeling Language (UML) diagram that illustrates the sequence of messages between objects in an interaction. The sequence diagram represents the flow of messages in the system and is also termed as an event diagram.

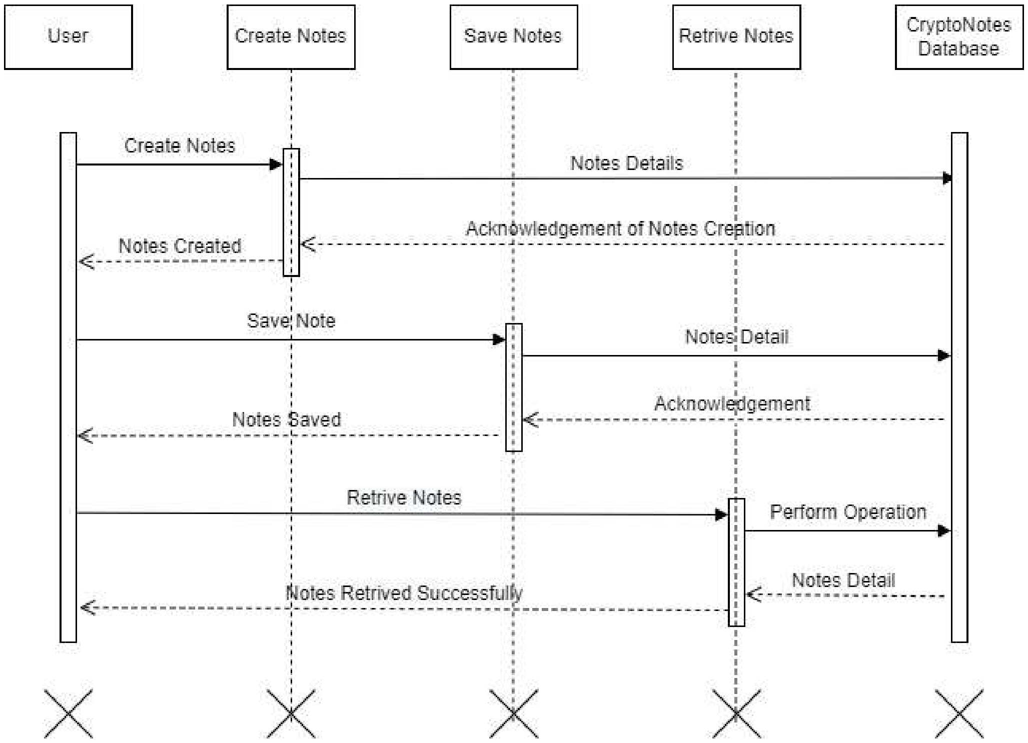


Figure 5 (Sequence Diagram)

5.2.4 Activity Diagram

An activity diagram provides a view of the behavior of a system by describing the sequence of actions in a process. Activity diagrams are similar to flowcharts because they show the flow between the actions in an activity; however, activity diagrams can also show parallel or concurrent flows and alternate flows.

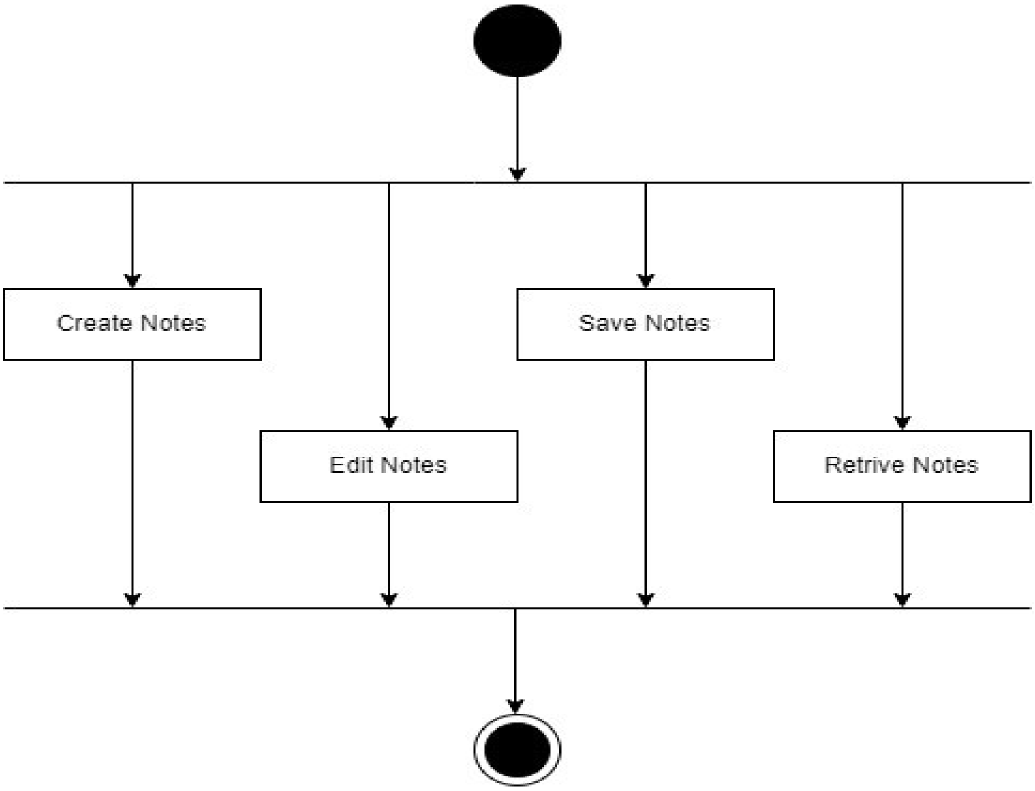


Figure 6 (Activity Diagram)

6 Implementation

Users create a note and add the data like title, subtitle and the text data into their respective fields. Application system capture the data, give it to the AES-256 encryption algorithm. It takes the data and encrypts the data. Further it forward the data the database driver and it store it to the SQLite database with the simple interface of room wrapper.

At a time of retrieval of data, user select the note which is user trying to fetch the data from database. Selected note's unique ID sent to the function, and it takes the data from database with the SQLite database driver. It returns the data from database and give the data to the algorithm; it decrypts the data and return the decrypted data to the application, and it displays the data.

7 Testing

7.1 Testing Plan

Planning Steps

1. Functionality Testing
2. Usability testing
3. Interface testing
4. Security testing

1. Functionality Testing:

Test for – database connection, forms used in the activities for submitting or getting information from users, Shared Preferences Testing.

Check all the links:

- Test the outgoing URLs from all the activities from specific domains under test.
- Test all internal links.
- Intents jumping from one activity to another.
- Test to check if there are any orphan activities.
- Lastly in link checking, check for broken links in all above-mentioned URLs.

Test forms in all pages:

Forms are an integral part of any app. Forms are used to get information from users and to keep interaction with them. So what should be checked on these forms

- First check all the validations on each field.
- Check for the default values of fields.
- Wrong inputs to the fields in the forms

Shared Preferences testing:

Shared Preferences are small files stored on a user's mobile. These are basically used to maintain the session mainly login sessions. Test if the Shared Preferences are encrypted before writing to the user mobile.

Validate your JS files:

It is very important to check if there are no visual or syntactical errors in your JavaScript code, which can lead to malfunction of your app and can also look weird.

Database testing:

Data consistency is very important in an application. Check for data integrity and errors while you edit, delete, modify the data or do any DB related functionality. Check if all the database queries are executing correctly, data is retrieved correctly and updated correctly.

2. Usability Testing:

Test for navigation:

Navigation means how the user surfs the activities, different controls like buttons, boxes or how users use the links on the activities to surf different activities.

Content:

Content should be logical and easy to understand. Check for spelling errors. Use of dark colors annoys users and should not be used in app themes. These are common accepted standards like as I mentioned above about annoying colors, fonts, frames etc.

Content should be meaningful. All the anchor text links should be working properly. Images should be placed properly with proper sizes. These are some basic standards that should be followed in app development. Your task is to validate all for UI testing.

3. Interface Testing:

The main interfaces are:

Web server and application server interface.

Application server and Database server interface.

Check if all the interactions between these servers are executed properly. Errors are handled properly. If database or web server returns any error message for any query by application server then application server should catch and display these error messages appropriately to users. Check what happens if user interrupts any transaction in-between? Check what happens if connection to web server is reset in between?

4. Security Testing

Following are the test cases for security testing:

Try to login with wrong credentials and check the message that is displayed for wrong credentials.

Try to enter wrong data into the forms, or try to leave some fields empty while filling any form and check the error message.

7.2 Testing Strategies

❖ White Box Testing

White box testing (WBT) is also called **Structural or Glass box testing**. White box testing involves looking at the structure of the code. When you know the internal structure of a product, tests can be conducted to ensure that the internal operations performed according to the specification. And all internal components have been adequately exercised.

Why do we do White Box Testing?

To ensure:

- That all independent paths within a module have been exercised at least once.
- All logical decisions verified on their true and false values.
- All loops executed at their boundaries and within their operational bounds internal data structures validity.

Need of White Box Testing?

To discover the following types of bugs:

- Logical error tends to creep into our work when we design and implement functions, conditions or controls that are out of the program
- The design errors due to difference between logical flow of the program and the actual implementation
- Typographical errors and syntax checking

Limitation Of WBT:

Not possible for testing each and every path of the loops in program. This means exhaustive testing is impossible for large systems. This does not mean that WBT is not effective. By selecting important logical paths and data structure for testing is practically possible and effective.

❖ Black Box Testing

- Black box testing treats the system as a **"black-box"**, so it doesn't explicitly use Knowledge of the internal structure or code. Or in other words the Test engineer need not know the internal working of the "Black box" or application.
- Focus in black box testing is on functionality of the system as a whole. The term 'behavioral testing' is also used for black box testing and white box testing is also sometimes called 'structural testing'. Behavioral test design is slightly different from black-box test design because the use of internal knowledge isn't strictly forbidden, but it's still discouraged.

- Black box testing occurs throughout the software development and Testing life cycle i.e., in Unit, Integration, System, Acceptance and regression testing stages.
- Advantages of Black Box Testing
    - Tester can be non-technical.
    - Used to verify contradictions in actual system and the specifications.
    - Test cases can be designed as soon as the functional specifications are complete.
- Disadvantages of Black Box Testing
    - The test inputs need to be from large sample space.
    - It is difficult to identify all possible inputs in limited testing time. So writing test cases is slow and difficult. Chances of having unidentified paths during this testing.

7.3 Testing Methods

Different types of testing method are used,

Unit Testing:

- In it analyst tests the program making up a system. The software units in a system are the modules and routines that are assembled and integrated to perform a specific function.
- It focuses on modules, independently of one another, to locate errors. This enables the tester to detect errors in coding and logic that are contained within the module alone.

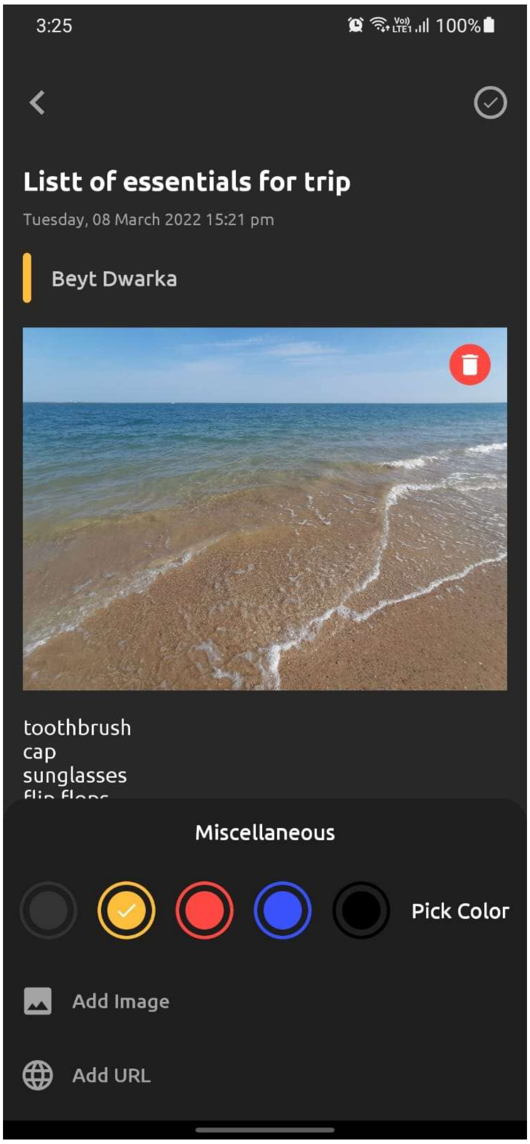
Bottom-Up Unit Testing:

- It can be performed from the bottom up, starting with the smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing, a short program executes the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

Top-Down Unit Testing:

- As the name implies, begins with the upper-level modules. However, since the detailed activities usually performed in lower-level routines are not provided, stubs are written. A sub is a module can be called by the upper-level module and that, when reached properly, will return a message to the calling module, indicating a proper interaction occurred.

8 Screenshots



3:25 100%

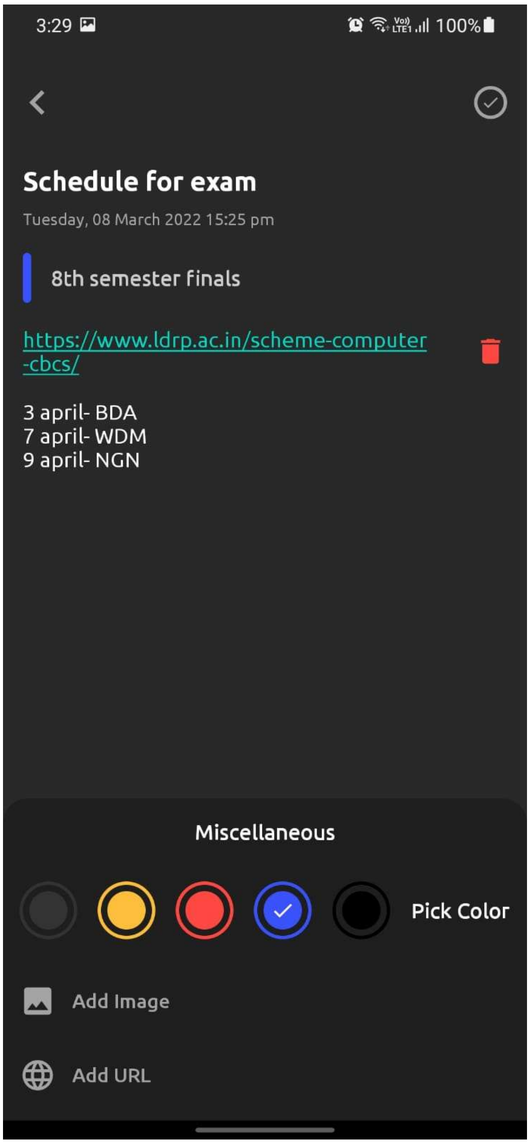
Listt of essentials for trip

Tuesday, 08 March 2022 15:21 pm

Beyt Dwarka

toothbrush
cap
sunglasses
flip flops

Figure 7 (Image Function)



3:29 100%

Schedule for exam

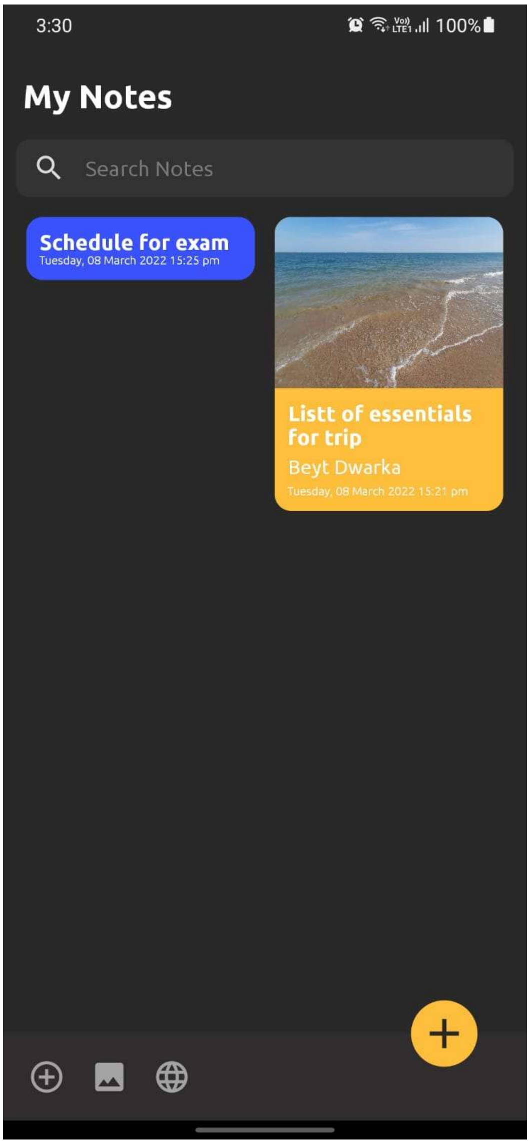
Tuesday, 08 March 2022 15:25 pm

8th semester finals

https://www.ldrp.ac.in/scheme-computer-cbcs/

3 april- BDA
7 april- WDM
9 april- NGN

Figure 8 (Link Function)



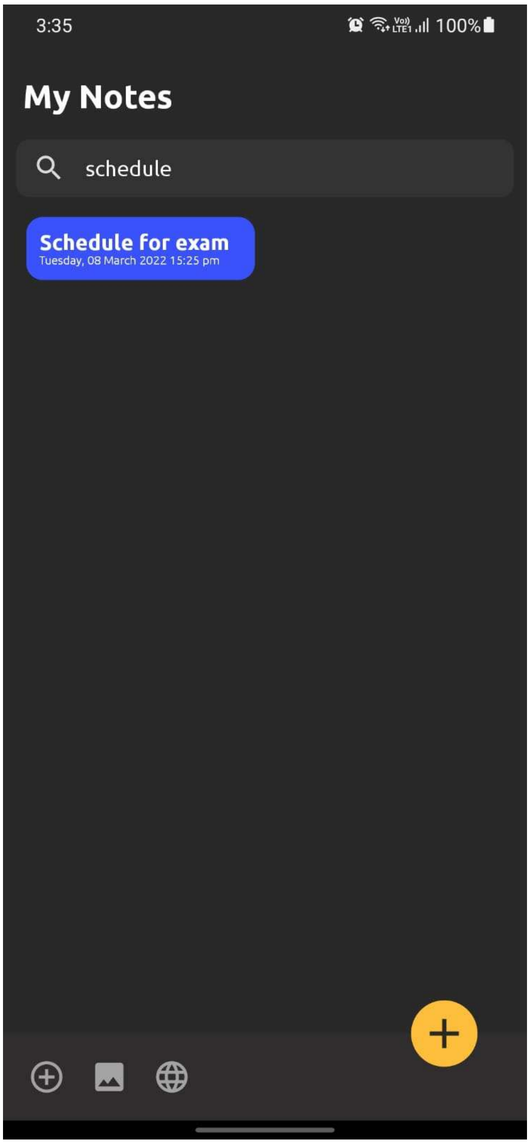
My Notes

Search Notes

Schedule for exam
Tuesday, 08 March 2022 15:25 pm

Listt of essentials for trip
Beyt Dwarka
Tuesday, 08 March 2022 15:21 pm

Figure 9 (Home Page)



My Notes

schedule

Schedule for exam
Tuesday, 08 March 2022 15:25 pm

Figure 10 (Search Function)

9 Conclusion

9.1 Future Scope

- We can add authentication and move the database to cloud storage so that every user can access their data whenever he wants from wherever.

9.2 Conclusion

- CryptoNotes will help people to store crucial information without being tense about any malware attack.
- It aims to store the notes in the local storage.
- CryptoNotes resolves main concern of security.
- Free Forever.

10 References

- https://developer.android.com/
- https://android-developers.googleblog.com/
- https://www.youtube.com/
- https://www.visual-paradigm.com/
- https://stackoverflow.com/
- https://developer.android.com/studio
- https://www.sqlite.org/index.html
- https://medium.com/