Proceedings of National Conference on "Emerging Trends and Recent Innovations in Engineering and Technologies"
held at Nagarjuna College of Engineering and Technology, Bangalore on 22 and 23 April 2022

ANDROID NOTES USING FINGERPRINT AUTHENTICATION

[1]Anuradha R, [1]Bestha Srihari , [1]Anuka, [1]Nikith A R , [1]Parna H K, [2]Prof. Sapna R , [2]Prof. Preethi

[1]UG Students, [2]Professor,Dept of Computer science & Engineering

Presidency University
Itgalpur Rajanakunte Yelahanka, Bangalore-560064

[*]Corresponding author: sapnar@presidencyuniversity.in preethi@presidencyuniversity.in

ABSTRACT

Android Notes with Fingerprint Authentication System is a simple but effective framework for securing notes with fingerprint authentication. This system does not require registration to log in; it is open-source, and the notes can only be opened by the owner using their own phone, which scans for the owner's print while logging in. Notes on Android can be used as a personal journal, private notes, or valuable reminders. Android Notes has a variety of names, but the purpose is the same: it records notes and keeps them hidden from anyone except the phone's owner. If your phone doesn't have a fingerprint scanner than you won't be able to use Android notes. The Client has the ability to create new notes, update existing notes, and remove both old and new notes at any time. Android Studio is used to create the front end, and SQLite is used to create the back end. Fingerprint Authentication is the most advanced level of security available on any phone, making it incredibly accurate and secure.

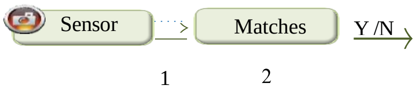
Keywords: **Android Studio, SQLite., Biometric Authentication, private notes.**

I. Introduction

Android Notes with Fingerprint is an easy but effective app for securing notes with a fingerprint. This device is also known as keyless authentication, as opposed to the conventional method of entering a password to unlock an app. This method does not require registration, but only the phone's owner has access to the notes since it looks for the owner's fingerprint. It may serve as a personal journal or a place to jot down important information. The front end of the Android notes app is built with Android Studio, while the back end is built with SQLite.

Step by step cell phone innovations empowered clients to do different assignments utilizing their cell phones. These assignments incorporate not just unassuming ones like playing versatile games and riding the web yet additionally more risky ones, specifically, the individuals who manage private data and monetary information. Along these lines, it is a dependable gadget that is needed to confirm the personality of an individual who attempts to utilize the gadget. However, conventional restricted intel dependent on arrangements like passwords, numeric PINs, and example locks has security issues, for example, secret word speculating assaults, animal power assaults, and shoulder-riding assaults. Additionally, they also have convenience issues due to the secret key which the client should remember some data and do a significant errand for login like composing a secret key and drawing an example. To order the location of these issues, unique finger impression acknowledgment is currently really utilized for a huge number, for instance, Galaxy S5, and Android Note. Finger impression acknowledgment is utilized both for opening a cell phone and for enacting other security risky functionalities in the cell phone, for instance, for affirming associations in monetary applications.

Thus, it is exceptionally basic to get the unique mark acknowledgment administration from conceivable dangers, for example, interfering with a finger impression picture between a picture sensor and a finger impression acknowledgment application and taking the finger impression information put away in a cell phone. Improperly, still, a portion of the right now sent gadgets don't appear to be properly protected against those scares. In this paper, we reveal the liabilities in the unique mark acknowledgment administration of Android Note by investigating the help application and build up potential assaults against this assistance. The note works only in Android smart phone with more than lollipop 5.1 and 1 GB RAM. It is set up with a FPC unique finger impression sensor on its back.



→Verification stages
Figure 1: biometric verification

It subsequent episode that removes a put away example from the non-unpredictable memory and reestablish unique finger impression include focuses by unraveling the example. By distinguishing and investigating a finger impression administration application on the stamped gadget, we perceived the area of the put away example.

We also discovered that the example was scrambled, but that a similar key and starting vector are hard-coded and are very similar across all devices. This plan brings about helplessness that an angry for the client and might be effectively validated in the event that she/he overwrites an example by another example duplicated from his/her own gadget. Moreover, by inspecting the design of the decoded design record, we had the option to reestablish all the component focuses and making the unique finger impression design. This infers that a cautiously phony example as per the document structure likewise may breeze through the confirmation assessment.

Despite the fact that we zeroed in on a particular gadget in directing our tests, the specialized mistakes we have found in this gadget are a typical misdirection that designers may fall into. Hence, we prescribe a couple of conceivable counter measures to direct those liabilities. We envision that the discoveries we acquired through our examination might be utilized as an overall rule to plan a safe unique mark check administration on cell phones. It gives the underlying data about the association of a conventional unique mark framework, a standard organization for a finger impression design, and the message correspondence system of Android measures.

II. Literature Survey

We have defined the execution method in this paper. To begin, download and install the app on your smartphone. When the app is installed, a user is given a sensor task to verify, such as touching their finger for a fingerprint scan. When a user accesses an application, the server unit compares the fingerprint stored with the encrypted and database-stored fingerprint. The device and the user are then verified.

The user can now access all of his or her info. When a user selects an application from the computer, the application requests verification in order to complete the user's request.

This two-step authorization approach addresses the need for the user to have better authorization. The models have now been matched, and the user's requirement has been met. Both of these measures help to deter fraud and increase protection.

In this paper, we presented related works and fingerprint bio-metric performance analysis, demonstrating that performance assessment is based on surveyed works using various criteria and established methods. When compared to password and token-based authentication, bio-metrics offers significant advantages.

Fingerprint verification is ideal because it not only detects features but also authenticates them. It is also preferable because a person's fingerprint remains the same throughout their lives and their ridge pattern remains the same

III. Biometric Verification Using Fingerprint

Numerous applications which manage fingerprints, including our objective gadget, utilize the unique finger impression minutia designs dependent on ISO. As per that the norms, there are four primary qualities. It numerous unmistakable edge types, two significant among them is an edge finishing and an edge bifurcation, which are as often as possible utilized in many settings [7–10], where an edge finishing sees for a point where an edge abruptly closes and an edge A variation is a point where a single edge splits into two. The firm instances of proof can be seen in Figure 2 and 3.

Biometric confirmation, which analyzes with the put-away layout, which is made up of various minutiae focuses, the features extracted from the current sensor image. By contrasting each unique mark in a put-away the test is completed when the format matches that from the sensor. The print's synchronization is expanded as each point matches. The client is allowed to get to the objective gadget just if the score is bigger than a predefined edge, the client is allowed to get to the objective gadget.

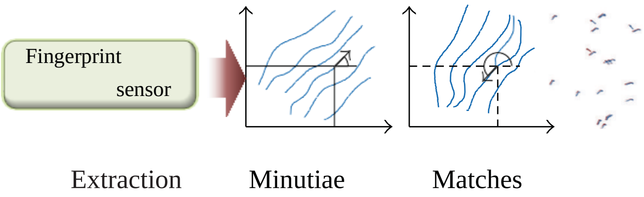


Figure 2: Matching of fingerprints.

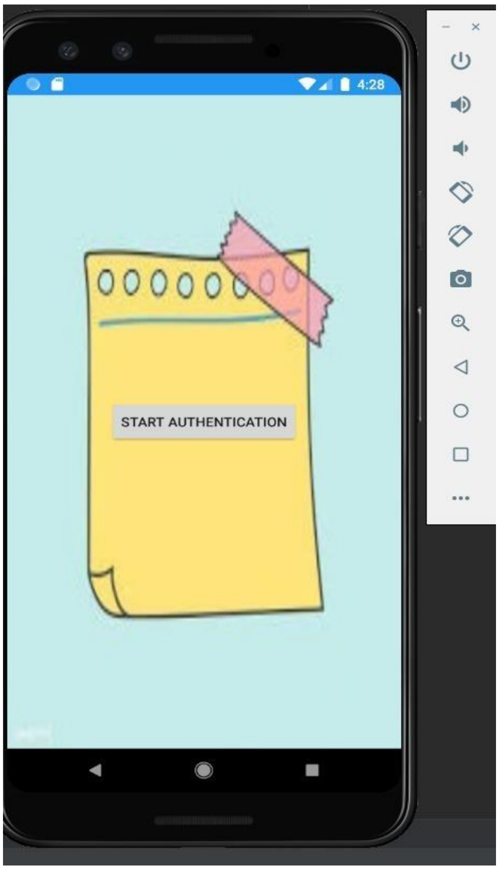


Figure 3: Authentication

IV. Existing System

Fingerprint authentication is also being used by a growing number of smartphones to authenticate their customers. Not only are fingerprints used to open these smartphones, but they are also used in Android notes. This is the highest degree of protection that any phone can have, making it extremely precise and secure. This framework is simple and easy to use from the user's perspective. It's easier to open an account by placing your finger on a scanner than it is to type a long password. Furthermore, while most people make the mistake of losing their passwords, you are unlikely to lose your biometrics. This system does not require registration, but only the phone's owner has access to the notes. This can be seen as a personal journal. This can go by a variety of names, but it serves the same purpose. This software would not work if your phone does not have a biometric feature. The user has the ability to create new notes, update existing ones, and erase them. Android Studio was used for the front end, and SQLite for the back end.

For fingerprint authentication, the minutiae-based algorithm is commonly used. The classification of fingerprints is an important aspect of this algorithm because it allows for a substantial reduction in the number of fingerprints used for each recognition procedure.

V. Components

Requirements for software:

1. Windows XP, Windows 7(ultimate, enterprise)
2. Android Studio
3. Java, XML and SQLite

Requirements for Hardware:

1. Android Phone with Built in Biometric
2. Processor –i3
3. Memory – 1GB RAM

VI. Proposed System

6.1 Objective:

In the proposed application, user would be able to access an Android Notes across the globe. The user would be able to add, edit and delete notes whenever he wants. There are no Registration formalities for this app and the owner can directly access the notes using their phone. This application is used in only fingerprint phones. Android Notes can be used as they want like personal notes, important notes and personal diary. The working of this application can be seen by executing various modules of the project

6.2 Tools:

The primary tools, which is expect to use for this project is Android Studio and SQLite. We will explore the possibility of using java and xml for developing our application. By using these tools, and programming it to make a collaborative effort in developing an Android Notes application. We except that it will be the primary resource for developing our project.

6.3 Needs:

1. Text Fields: The user will type text into a text field in our application. It may be a single line or a series of lines. When you touch a text area, the mouse appears and the keyboard appears. Text fields consider a variety of various exercises, such as subject selection, in addition to writing (cut, copy, paste). The <EditText> object can be used to add a text field to your layout. In most cases, you can do so with an element in your XML style. Text fields can accept a variety of input types, including numbers, dates, passwords, and email addresses.

2. Layouts: A format characterizes the visual construction for a UI, like the UI for an action or application gadget. Each format has a bunch of qualities that characterize the visual properties of that design. There are not many regular credits among every one of the designs and there are different qualities that are explicit to that format. Following are normal attributes and will be applied to every one of the formats.

3. Buttons: The Button is a UI control which is utilized to play out an activity at whatever point the client snaps or tap on it. Buttons in android will contain a book or a symbol or both and play out an activity when the client contacts it. Various kinds of catches accessible are Image Button, Toggle Button, Radio Button. The buttons used for our project is Add Button, Delete Button and Update Button.

4. Menu: In android, Options Menu is an essential assortment of menu things for an action and it is valuable to executions that universally affect the application, like Settings, Search, and so forth in the event that, on the off chance that we characterize things for the alternative's menu in both action or part, at that point those things will be joined presentation in UI.

6.4 Flow Chart:

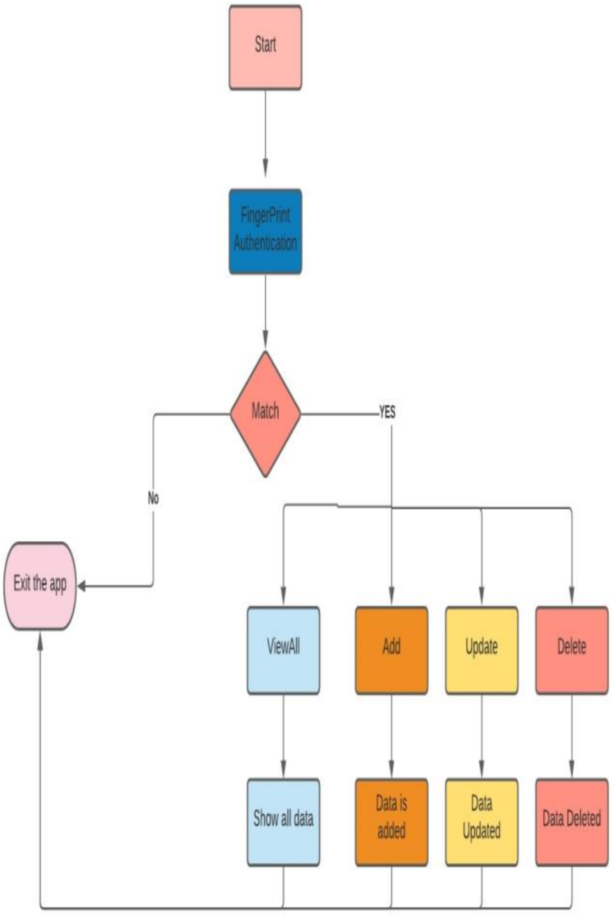


Figure 4: Flow Chart

6.5 Executive Summary:

When it comes to getting started with Android fingerprint authentication, there are a few things to keep in mind. We must first ensure that the lock screen is safe, that is, that it is secured by a pin, password, or pattern, and then confirm that at least one fingerprint is registered on the handset. If validated, gain access to the Android KeyStore to store the key used to encrypt/decrypt an entity, create an encryption key, and the cypher, and begin the authentication process by implementing a callback class that handles authentication events. These moves will be implemented later. Before you begin, make sure you have permission to use the touch sensor and fingerprint verification to check the safe lock screen in Android.

The first move is to check the protected lock screen, which is achieved by using keyguard manager and fingerprint manager, as well as gaining access to the Android keystore and generating a key. The second stage is to go to the Android keystore and produce the key that will be used to encrypt the data. If the key is ready, the next move is to create an android cypher that uses the key we created earlier. Develop the Android Fingerprint App Now it's time to put all of these methods together to create the Android Fingerprint Authentication App.

The third step is to create an Android fingerprint authentication callback method, and the final step is to create the callback class such that we may accept event notifications and know whether or not the authentication was successful. A real-world interface with a touch sensor can be used. In either case, the software can be tested in the simulator as well. We must customize the fingerprint accessing the security menu before we can use the software.

You must use the command to mimic the finger touch when the machine asks for a fingerprint. After fingerprint authentication is complete, the next move is to add a text field to the application so that the user can access it and type whatever he likes. The next move is to add a button to the application so that the user can edit, erase, and change their notes with the click of a button. This would make it easier for them to access the application. When any button is pressed and modifications are made, the callback class is activated, allowing us to receive event notifications and know if the notes have been effectively changed, added, or removed.

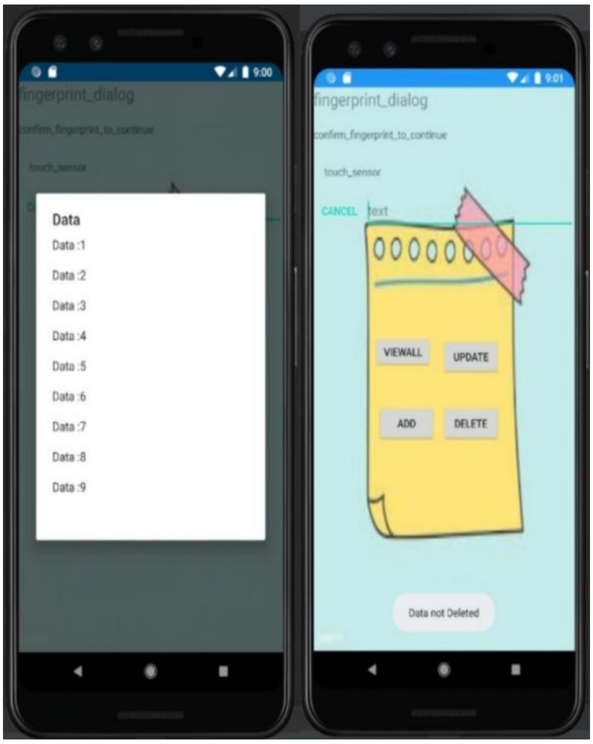


Figure 5: Data set

After this the next is that we have create a menu button where all the notes will be saved and it will be easy for the user to find the notes which he typed previously.

VII. Advantages

1. High security and assurance.
2. User Experience.
3. Non-transferrable
4. Biometrics are difficult to imitate or steal.
5. Convenient and fast.
6. All has a collection of biometrics that is unique to them.

VIII. Disadvantages

1. Runs only on Build in Biometric phones.

IX. Conclusion

In this report, we have reviewed how the Android notes using fingerprint are proposed and their features. Where is a unique finger impression acknowledgment application to recognize a couple of openings in the finger impression acknowledgment of Android Notes. The specialized defects we have found in this software is that this application cannot be used by the user who doesn't have a fingerprint smartphone: however, just the user of the smartphone can get to these notes using fingerprint as it looks for the user's print. To reduce these openings, we propose to utilize a physiological characteristic where it is based upon a person's physical characteristic which is assumed to be relatively unchanging. We envision that the outcomes we got through our application might be utilized as Android notes with a protected biometric check application on smartphones. Nonetheless, this application is utilized as significant notes, private notes, and personal diary which can give numerous names.

However, however, it assumes a comparative part of recording notes and guarding it against each other than the smartphone user. Accordingly, if there is no biometric highlight on the telephone, this application can't be utilized. In these android notes, the user can create new notes and edit existing ones, as well as delete them if necessary. Additionally, Android Studio is used for front-end programming, and SQLite is used for back-end programming. Fingerprint Authentication is the highest degree of security that any phone can have, and it is uncovered and depicted in this article. It is incredibly accurate and uses special mark application.

X. Future Scope

1. The notes can be organized in an easier way through this application
2. This application will be user friendly since the user interface is simple.
3. There is no limit set in order to write the notes.
4. This application can be accessed at anytime and anywhere.

XI. References

[1] Akanksha Bali, Shivangi Goswami, and Shagun Sharma, "Biometrics Security in Mobile Application Development and its Applications," International Journal of Scientific and Technical Advancements, Volume 5, Issue 1, pp. 51-60, 2019

[2] PayPal, "Pay faster with your fingerprint," 2014

[3] Pantech, "Pantech unveils VEGA LTE-A, world's first LTE-A with fingerprint recognition and rear touch," 2013,

[4] M. Bishop, *Computer Security: Art and Science*, Addison-Wesley, Boston, Mass, USA, 2003.

[5] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM Systems Journal*, vol. 40, no. 3, pp. 614–634, 2001.

[6] A. K. Jain, K. Nandakumar, and A. Nagar, "Biometric template security," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 579416, 17 pages, 2008.

[7] A.K. Jain, "Technology: biometric recognition," *Nature*, vol. 449, no. 7158, pp. 38–40, 2007.

[8] ISO/IEC, "Information technology—biometric data interchange formats—part 2: finger minutiae data," *ISO/IEC International Standard* 19794-2, 2011.

[9] ANSI and INCITS, "American National Standard for information technology—finger minutiae format for data interchange," *ANSI INCITS* 378-2009, 2009.

[10] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Fingerprint image reconstruction from standard templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1489–1503, 2007.