UNIVERSITY OF TARTU

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Institute of Computer Science

Computer Science

Enno Eller

Simplifying Mobile Social Media Authentication On Android

Bachelor Thesis (6 ECTS)

Supervisor:	Huber Flores, MSc
Supervisor:	Satish Srirama, PhD

Tartu, 2015

Simplifying Mobile Social Media Authentication On Android

Abstract:
Nowadays, smartphones are very common and are being used in everyday life. Even though mobile phones were originally invented as calling devices, smartphones allow the user to communicate in different ways including social media, instant messaging, recording and watching videos, etc. Recent statistics presented by Pew Research Center as of January 2014 claim, that 74% of online adults use social media. In case they own a smartphone, they probably use social media on it as well, but with restrictions that come with the size of the device, affecting how we view content and also type. Typing on smartphones can be frustrating, but more so when the keyboard size prevents us from succeeding with authentication and we have to type the same text numerous times, which can lead to shorter passwords decreasing the security of the accounts. This paper proposes a solution to such occurrences by using pattern recognition rather than typing. Patterns allow the screen to be used more efficiently, giving the user more room for accuracy errors. Survey results indicate that approaching authentication in this way is feasible.

Keywords: Mobile, Android, Authentication, Pattern

Mobiilse Sotsiaalmeedia Autentimise Lihtsustamine Androidil

Lühikokkuvõte:
Tänapäeval on nutitelefonid väga levinud ja neid kasutatakse igapäevaselt. Kuigi mobiiltelefonid loodi algselt helistamiseks, nutitelefonid võimaldavad kasutajal suhelda erinevatel viisidel, nende seas sotsiaalmeedia, kiirsõnumid, lindistada ja vaadata videoid jne. Pew Research Center poolt thetud iljutised uuringud jaanuarist 2014 väidavad, et 74% internetti kasutavatest täisealistest tarvitavad ka sotsiaalset meediat. Juhul, kui need isikud omavad nutitelefoni, on tõenäosus, et nad kasutavad sotsiaalmeediaid ka oma nutiseadmel, kuid piirangutega, mis tulenevad seadme suurusest. Suurus mõjutab, kuidas me infot vaatame ja teksti sisestame. Trükkimine nutitelefonil võib osutuda masendavaks, seda enam, kui klaviatuuri suurus takistab meil autentimise edu ja me peame sama teksti sisestama mitmeid kordi. Sellised olukorrad võivad viia lühemate paroolide kasutamiseni, mis omakorda vähendab meie kontode turvalisust. Antud töö pakub välja lahenduse sellistele olukordadele kasutades trükkimise asemel mustreid. Mustrid võimaldavad efektiivsemat ekraani kasutust, mis annavad kasutajale rohkem ruumi täpsuse vigade vältimiseks. Uuringu tulemused näitavad, et selline lähenemine autentimisele on võimalik.

Võtmesõnad: Mobiilne, Android, Autentimine, Muster

Contents

1

Introduction

Social media has become a part or our life, whether we communicate by it, express ourselves or use it for entertainment. Originally these media sites were developed for use in PC, but over the years as smartphones are becoming more common and capable, most of these sites have moved closer to us, into our pockets. With the innovation of smartphone social media applications, one would guess that the way we authenticate ourselves would change, to accommodate these devices, but so far they are mistaken.

1.0.1 Motivation

Applications on Android are mostly configured to keep the user signed in or memorize the credentials, though there are still applications that have not gone or are still implementing these methods, like Skype[1][2]. Implementing automatic authentication may increase the usability of an applications, it also raises a security concern, where the device holder has access to the data, which normally would be hidden behind authentication requirements.

Realising this security issue, one might keep his applications signed out, whenever not using the device or log out purposely when lending the device to somebody. After doing so, the user would have to authenticate all the applications the hard way - typing. Given the inconvenience of the conventional typing authentication process on our devices, one might use the application less.

1.0.2 Contributions

The goal of this thesis is to describe current methods being used for social media authentication on mobile devices and introduce a new approach that would increase the user experience of the process.

[1] http://www.skype.com/en/

[2] http://www.androidauthority.com/skype-update-v5-5-makes-it-easy-to-sign-in-625877/

A new, pattern-based authentication method for mobile devices will be designed and implemented. The method will use the screen of the device in a way that is easier for the user and as a result leads to less mistakes and higher user experience.

The objectives of the new method are the following:

- To make the authentication process more enjoyable for the user, by providing better use of the screen of the mobile device.
- To ease the integration of automating authentication for mobile social media applications, giving the developer opportunity to focus more on the application functionalities.
- To provide applications with multi-user support, for devices being shared amongst users.

To integrate pattern-based authentication into an application, a Java library for Android[3] was created. Application developers can use this library to provide users the option of saving their credentials and authenticate with pattern instead. This method achieves better screen usage, which makes it easier for the user.

The new method was evaluated and compared to the traditional authentication process by conducting a study in a set of 20 people. According to the results, the proposed method is easier and more user-friendly. Participants were less repulsed by the proposed method than traditional authentication process.

1.0.3 Outline

Chapter 2: describes the most popular authentication automation tools used on Android. It also takes a closer look at numerous previous studies that use other mechanisms on the mobile devices to simplify authentication.
Chapter 3: describes the research question that this thesis focuses on and its necessity.
Chapter 4: describes in detail the development process and the final proposed mechanism. Also describes, how one would implement this library in their app.
Chapter 5: explains the evaluation of the proposed solution as well as takes a closer look at user feedback regarding the proposed method compared to traditional approach.
Chapter 6: concludes the thesis with a summary and future research directions.

[3] https://www.android.com/

2

A Review in Social Media Authentication

2.1 Authentication Process Usability

Authentication process usability in this context refers to automating or simplifying the process of authenticating the user accessing the phone or features on the phone. This section discusses some of the tools used on mobile phones or tablets to simplify authentication.

2.1.1 Android AccountManager

Most of the applications are using Android AccountManager[4], see Figure 2.1, as a tool to automate authentication. It is a built in centralized registry that can hold user credentials or even authentication tokens which are generated via application server. Though it requires implementing various components, increasing the complexity, it is still a good method for single user device. For example Google[5], Facebook[6], and Microsoft Outlook[7] each use this method.

2.1.2 Service Providers

The next most used method of authentication is provided by social media websites. These providers are giving developers the option to let users authenticate by using accounts on the social media websites. Users do not have to create new accounts to these sites, but will refer to their already existing accounts on social media as a way of registration. On android the user needs to have that social media application installed and logged in to use

[4] http://developer.android.com/intl/zh-CN/reference/android/accounts/AccountManager.html

[5] https://play.google.com/store/apps/ - Google

[6] https://play.google.com/store/apps/ - Facebook

[7] https://play.google.com/store/apps/ - Microsoft Office Outlook

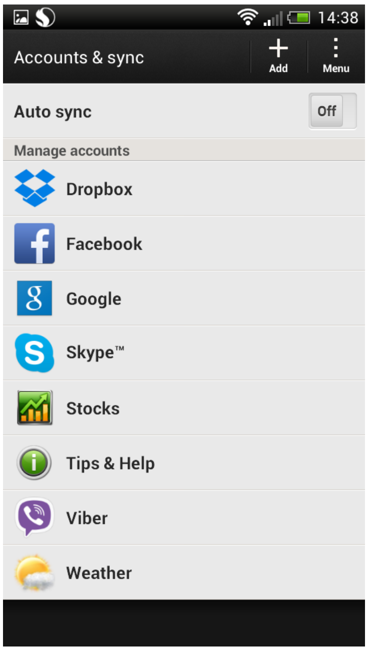


Figure 2.1: Android AccountManager

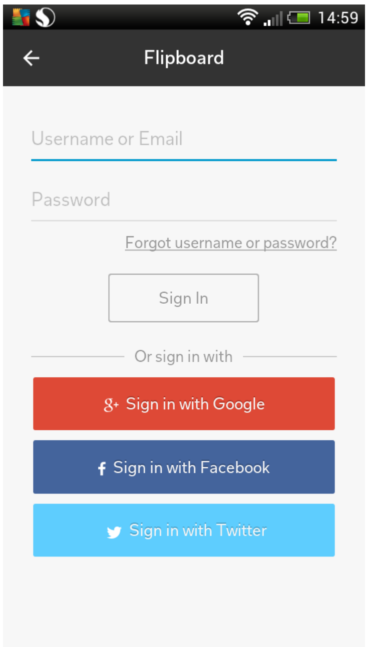


Figure 2.2: Service providers

this method. The most known two providers are Google and Facebook. As an example of how service provider tools appear on an application, *Flipboard: Your New Magazine*[8] app is used, seen on Figure 2.2.

2.1.2.1 Google Identity Platform

Google is providing android developers with an application programming interface (API) which allows users to register and authenticate using Google account, but also allows developers to integrate other Google services into their applications: payments via Google Wallet, sharing with Google+, saving files to Drive, etc.[9]

2.1.2.2 Facebook SDK

Facebook has a software development kit (SDK) for android developers. Just as with Google API, the SDK allows authentication via Facebook account and also provides more services - sharing on Facebook, sending application invites via Facebook, etc.[10]

[8] https://play.google.com/store/apps/ - Flipboard: Your News Magazine

[9] https://developers.google.com/identity/

[10] https://developers.facebook.com/docs/android

2.2 Previous Research

2.2.1 Visual Login

Visual login refers to a class of mechanisms that rely on the selection of icons or photo images to produce a password value. Visual login is a knowledge-based approach like passwords. Instead of alphanumeric characters, users must remember image sequences. Visual images are presented to the user for selection by tiling a portion of the user's graphical interface window with identically sized squares, grouped into a 5 x 6 matrix. The surface of each square displays a bit-mapped image or thumbnail of some picture supplied in a predefined digital format. Selecting the correct sequence of thumbnail images authenticates the user to the device. (5)

2.2.2 Tap Pattern

Gesture interaction with mobile devices has become common-place. One class of gestures that is widely used is tapping. While key strokes are usually perceived as single events, taps have an implied duration in time. Single taps, long presses and double taps are common examples. These simple patterns can be detected efficiently with very crude algorithms, relying solely on timers. Tap patterns, however, can be more generally defined as a sequence of intervals of ”on” and ”off” times that is, the ordered time distances between and within taps. (8)

2.2.3 Fingerphoto Recognition

The intention of the fingerphoto recognition is that for authentication the user simply positions his finger close in front of the camera in order to capture a biometric sample. The algorithms for finger detection and quality assurance check continuously the preview images of the camera after the capture process has been initiated by the user. The results of the algorithms are calculated in real-time and are displayed on the graphical user interface. A photo is automatically taken when all criteria for the fingerphoto recognition are fulfilled. (13)

2.2.4 Token-based Authentication

Authentication on smartphones does not have to include doing something to or with the device, instead it can involve a physical token. The token can be so small it could be carried on a key chain and it automatically unlocks the smartphone whenever its owner wants to use it. The token is based on magnetic fields detected by the smartphone's compass or on an acoustic transmitter that generates a signal picked up by the handset's microphone. All the user has to do is carry the token with him. (1)

2.2.5 Arm's Flex

One of the human behaviours considered being unique is arm's flex. It is a gestural pattern i.e. the way people bend their arm for picking a phone when responding to incoming calls. That arm's flexing is considered as a subset of gesture pattern in lower limb gesture. Every person who bends their arm will have different strength measured by accelerometer using smartphone even if they own same arm's flex pattern visually.(9, 11)

2.2.6 Multitouch Image-based Authentication

Multitouch image-based authentication password can consist of multiple rounds, where in each round the user can mark multiple points on an image. Click points have the advantage that they can be entered quickly, even with multiple fingers simultaneously, while drawing complex patterns requires more time. Multitouch authentication uses background images as cues and determines the image for the next round based on the user's input in the current round. Thus, the user can instantly recognize if the points selected in the previous round were correct or wrong (expected vs. unexpected image in next round). A back button allows for correction of errors. Each image should also only appear once in a password sequence to prevent memory interference between two instances of the same image. (10)

2.3 Continous Authentication

It is not always good to have a phone authenticate the user once and let him keep using the phone till it gets locked automatically by a timer or physically by the user. Hence there are methods that monitor the phone even when the phone is unlocked and only when there is sufficient evidence that the current user is not the smartphone owner, traditional user authentication is activated. The next sections describe continuous authentication methods.

2.3.1 Gait Recognition

The term gait recognition describes a biometric method which allows an automatic verification of the identity of a person by the way he walks. Gait recognition is based on wearing motion recording sensors on the body in different places: on the waist, in pockets. The wearable sensors can be accelerometers (measuring acceleration), gyro sensors (measuring rotation and number of degrees per second of rotation), force sensors (measuring the force when walking) etc. (3, 12)

2.3.2 Keystroke Analysis

This method of authentication analysis the detailed timing information that describes exactly when each key was pressed and when it was released by the person typing. (2)

2.3.3 Location Information

Phones and tablets nowadays come with built-in GPS systems. GPS individually or in cooperation with cell towers allows a phone to acquire its current location which is analysed against previous data. (14)

2.3.4 Orientation Sensor

A user has a unique way to hold and operate his/her smartphone while working on some applications and such behavioral biometrics can be captured from the readings of the orientation sensor. User's behavioral biometrics of up-down flicks and left-right flicks from the orientation sensor are monitored to authenticate. (7)

2.3.5 TouchScreen Gestures

As long as the smartphone is used, gestures are monitored to authenticate the user continuously. Continuous authentication in done on the background using intercepted touch data from normal user-smartphone interactions. The detection approach is invoked on-demand whenever touch inputs are received and is transparent to the smartphone user. Selected touch gesture information are collected including gesture type, X and Y coordinates, directions of the finger motion, finger motion speed, pressure at each sampled touch point and the distance between multi-touch points. In total, there are 53 features for each touch gesture. The six most frequent and useful gestures: down to up swipe, up to down swipe, left to right swipe, right to left swipe, zoom-in, and zoom-out. Since a smartphone user may apply different levels of touch pressure at different stages of a touch gesture, they are divided into three segments, (i) the beginning of a touch motion, (ii) the main touch motion, which is the longest segment and (iii) the end of a touch motion. (4, 6)

2.4 Summary

The world of android and mobile in general is filled with means to authenticate the user. All of the methods discussed have their advantages and disadvantages, but they all serve the same purpose of keeping our device and data secure. The most commonly used methods, Android AccountManager and service providers, accomplish the task of automating authentication well, but either have the user tie their account to some other social media or do not have the support for multiple users. There has been a lot research done on the authentication for mobile, with interesting methods being developed, but so far none of them is considering support for multiple users. The next chapter describes the problem in more detail.

3

Problem Statement

In the few years previous to 2010, tablets started to circle the market and nobody saw exactly how it would affect the devices we own. In 2010 Steve Jobs, the co-founder of Apple Incorporation, predicted that tablets would overtake PCs (personal computer). Slowly tablets have reduced the sales gap and in 2015 that prediction will probably come true[11].

Though smartphones are becoming more common as means to use social media, the authentication process remains the same as in personal computers. That in mind, when we look at these devices, we can immediately notice the size and lack of peripherals compared to PCs, raising the concern of input difficulty. In particular, inputting complicated passwords, that require precision to the last letter.

Applications in Android are mostly configured to keep the user signed in, but there are those who like to keep their devices logged out of apps for security reasons or for the fact that they share a device. In such case the authentication process is inevitable.

3.1 Research Question

The proposed solution to the problem of conventional username-password authentication method is to modify traditional input mechanism to better fit the characteristics of a smartphone. These devices have other mechanisms to capture data from the user, to simplify the input data process for recurrent authentication process. Mechanism, the screen of the device, can be used to introduce gestures from the user as patterns.

This thesis aims to evaluate whether a pattern based method of authentication is more user-friendly than the traditional approach. To do so, such a mechanism is developed and implemented for a mobile application as proof of concept. A survey is then conducted to compare this method to the traditional approach in terms of simplicity and repulsiveness.

[11] http://www.extremetech.com/computing/185937-in-2015-tablet-sales-will-finally-surpass-pcs-fulfilling-steve-jobs-post-pc-prophecy

3.2 Summary

As mobile devices are slowly taking over from PCs, applications need to consider the characteristics of a smartphone and evolve accordingly. This thesis attempts to find a solution to the level of difficulty that traditional authentication methods have on mobile devices. A pattern based authentication method is developed to raise the simplicity of the process and is implemented in an app. A survey is conducted to determine whether the proposed method is more suitable for mobile devices. The next chapter describes the architecture of the created solution.

4

Pattern-Based Authentication method

In Chapter 2 multiple alternatives to automating authentication were discussed. This section will focus on the solution developed as part of this theses.

4.1 Short Description of the Solution

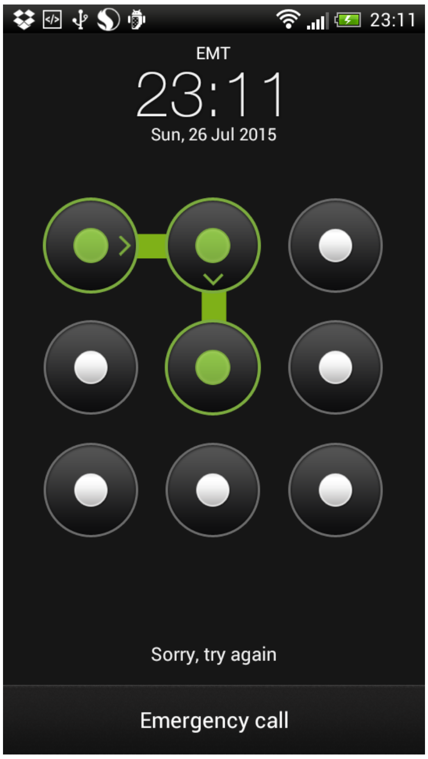
This solution is a library that developers can apply in their applications with very little coding. All needed to do is import the library into a project and refer to it.

The library provides easy registration for new credentials, storing them and retrieving for authentication. The user only inserts the credentials once and protects them with a pattern which is used for retrieving them. The pattern and the credentials are stored locally in a database, which is application specific and can only be accessed by it.

An example project is used in this work for demonstration purposes.

4.2 Android Lockpattern

As mentioned in the previous section a pattern is used to protect the credentials. Android users might already be familiar with the lockpattern even when they have not heard the term itself. It is used for locking the device from unwanted access to it, shown in the Figure 4.1. It is a 3x3 matrix consisting of circles/dots. To draw a pattern a user must press on a circle and drag through others to make a pattern and release to verify it. Even though the user only sees a picture of the pattern it is actually a string of numbers representing the dots where the user changed direction. For example a string "1-7-8" would be a pattern "L". Android lockpattern is known to users and is very easy to understand. Typing is taken out of the authentication process, which reduces the amount of errors made, and therefore used in this solution.



Sorry, try again

Figure 4.1: Android lockpattern

4.3 Supporting Multiple Users

The key factor what makes it different from previous solutions is the support for multiple users and multiple accounts per user. Mobile devices are commonly personal, but tablets in the other hand could be used in a household by the whole family. Needless to say that having to authenticate numerous times during a day might be a pain.

4.4 Data Storage

Every android comes with built in database system - SQLite[12]. It is easy to use, yet powerful to handle large amounts of data. Though no device will ever have the amount of users to even slightly test the database system, the benefit of using it is the simplicity and scalability allowing future changes in the database without losing any data already stored.

[12] https://www.sqlite.org/

Since the library itself does not directly save data or query for it from the database, an Android component Content Provider[13] is used. It manages access to the data, encapsulates it and provides mechanisms for defining data security. Content Providers can be used in two ways: provide access to it for all applications or just the one with permissions. In this case only the application intended to use it is given permissions declining any intruders from sniffing around.

4.5 Structure of the Library

This section will go more specific into the structure, classes and flow of the library to give insight of how it works.

As mentioned numerous times previously, this solution is a library for Android applications. To be effective in the world of coding, reusing code made by self or others is essential. The same rule is implied here by using a library that withdraws the lockpattern function from Android source code. That is to say amongst other libraries used to create the final product of an application, these two libraries would be used in an application component tree as shown in the Figure 4.2.

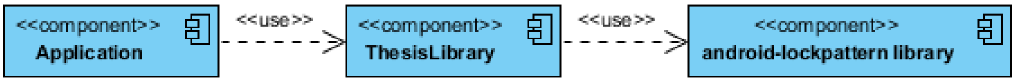


Figure 4.2: Example of application component diagram

The code within the library is somewhat sectioned: interface to the library, activities/visual presentation, content provider and database management. The most substantiate part of the code is database management with the highest complexity giving the library flexibility. The database currently is located internally in the device, but the code is designed to allow moving it to another path on the device or entirely off to a cloud. All the data columns are defined in a "database contract" for easy access and modification, the database will automatically update itself if the underlying data model is changed and using singleton pattern only one instance of the database can run at any given time to prevent data corruption. The *database management class* including the other classes can be seen in the class diagram in the Figure 4.3.

The code within the library can also be divided into sections: interface to the library, activities/visual presentation, content provider and database management. It may seem much for so trivial task, every little component serves its purpose to

[13] http://developer.android.com/intl/zh-CN/reference/android/content/ContentProvider.html

Though applications and libraries are developed for present requirements, the future can not be foreseen and should always be somewhat considered. To make this library flexible to unseen changes design patterns are used.

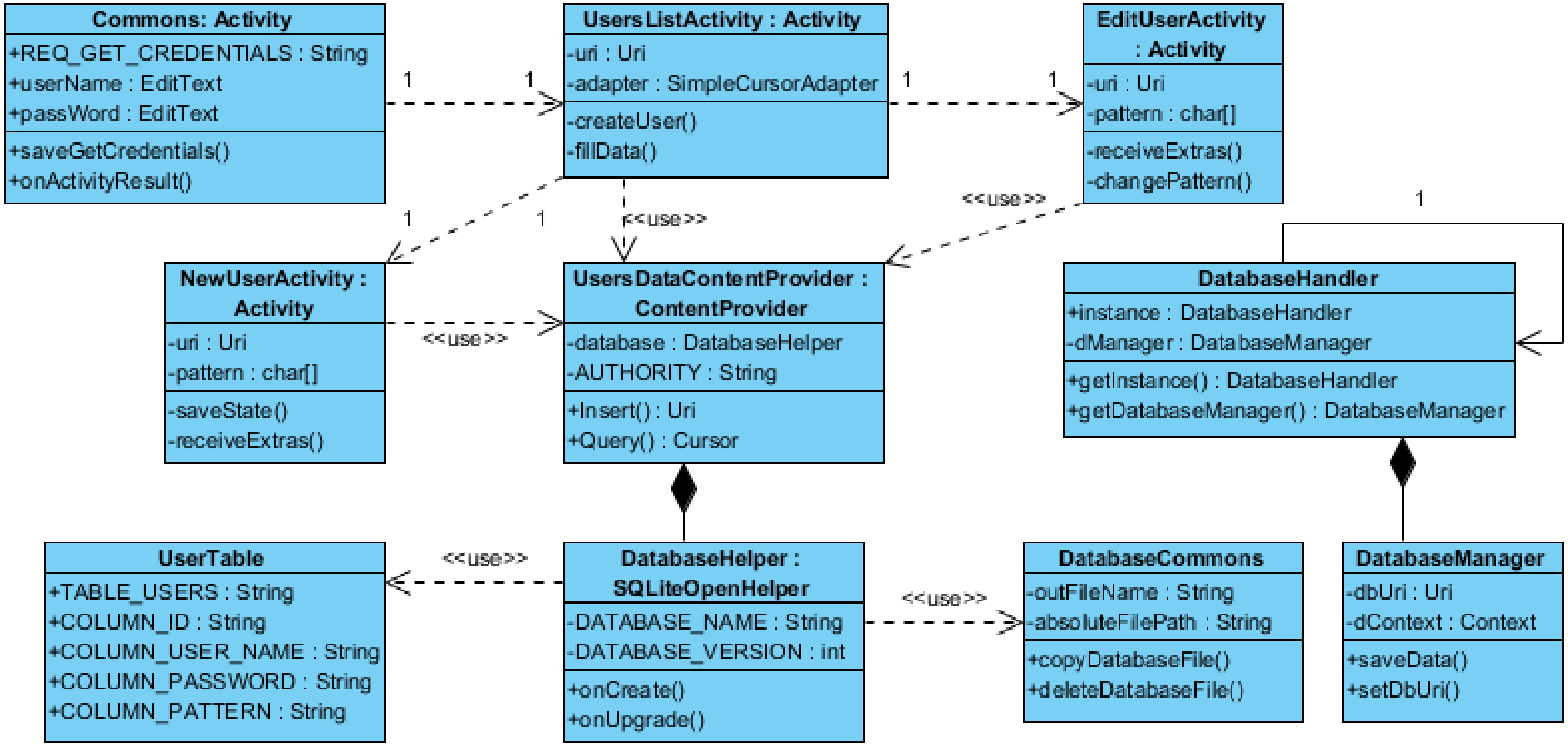


Figure 4.3: Class diagram of the library

4.6 Flow of the Library

To better understand what the library does, it is good to know the flow of it. For better visualisation a few diagrams are used.

Often less is more, this is the design ideal behind the library. A user has two main actions that he will take to authenticate: register an a account and start authenticating with it. In other words, one action is to get the credentials of the account into the phones memory and the other to access them.

When a user is logging into an application, using this library, he is asked whether he would like to save the credentials. If so he is taken to an activity to verify the password once more. When the password is verified, a new activity is presented to create a pattern for future authentication, and finally the credentials with the newly created pattern are stored to the database. This procedure is also illustrated on Figure 4.4 as a sequence diagram.

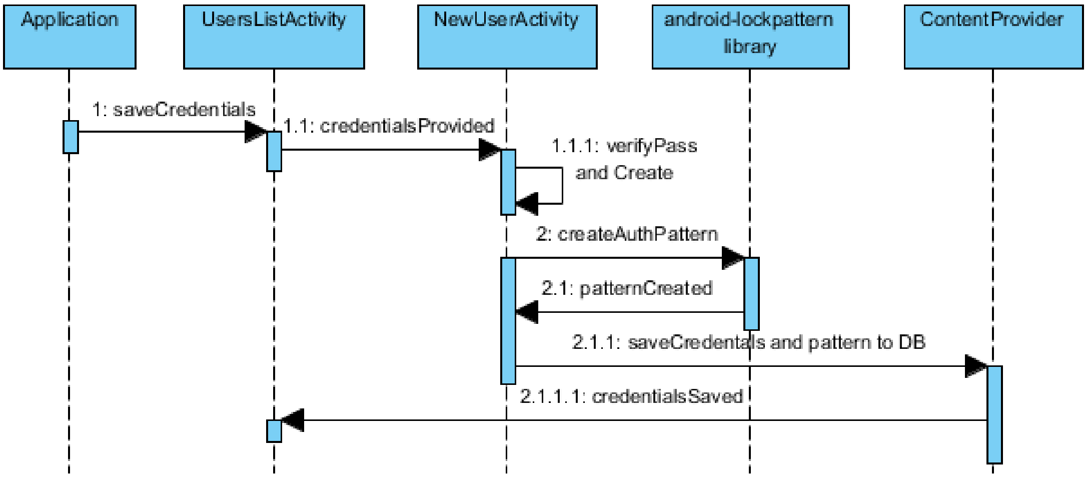


Figure 4.4: Sequence diagram of registrating new account

With one or more credentials stored in the database the user can authenticate using them. Either finishing the registration sequence or accessing the *UsersListAcivity* from the the application, the user has a list of stored credentials presented to him. Clicking on a chosen account the library will get the information from the database and ask the user to verify the account by inputting the pattern. On a successful verification the credentials are passed to the application. Illustrative sequence diagram is shown on the Figure 4.5.

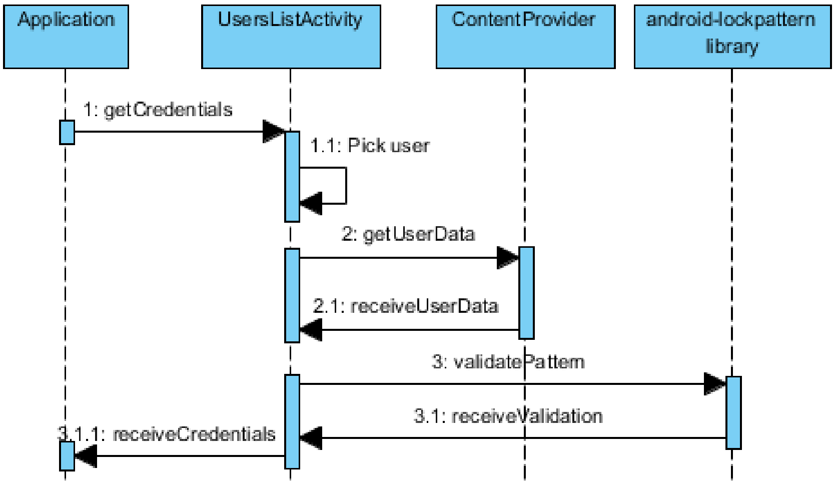


Figure 4.5: Sequence diagram of authentication

The previous descriptions and sequence diagrams of the registration and authentication process give the idea of a successful interpretation of the library. Though the user is given choices to back out of the process or the process could be failed. A flow chart covering these possibilities is seen in the Figure 4.6.

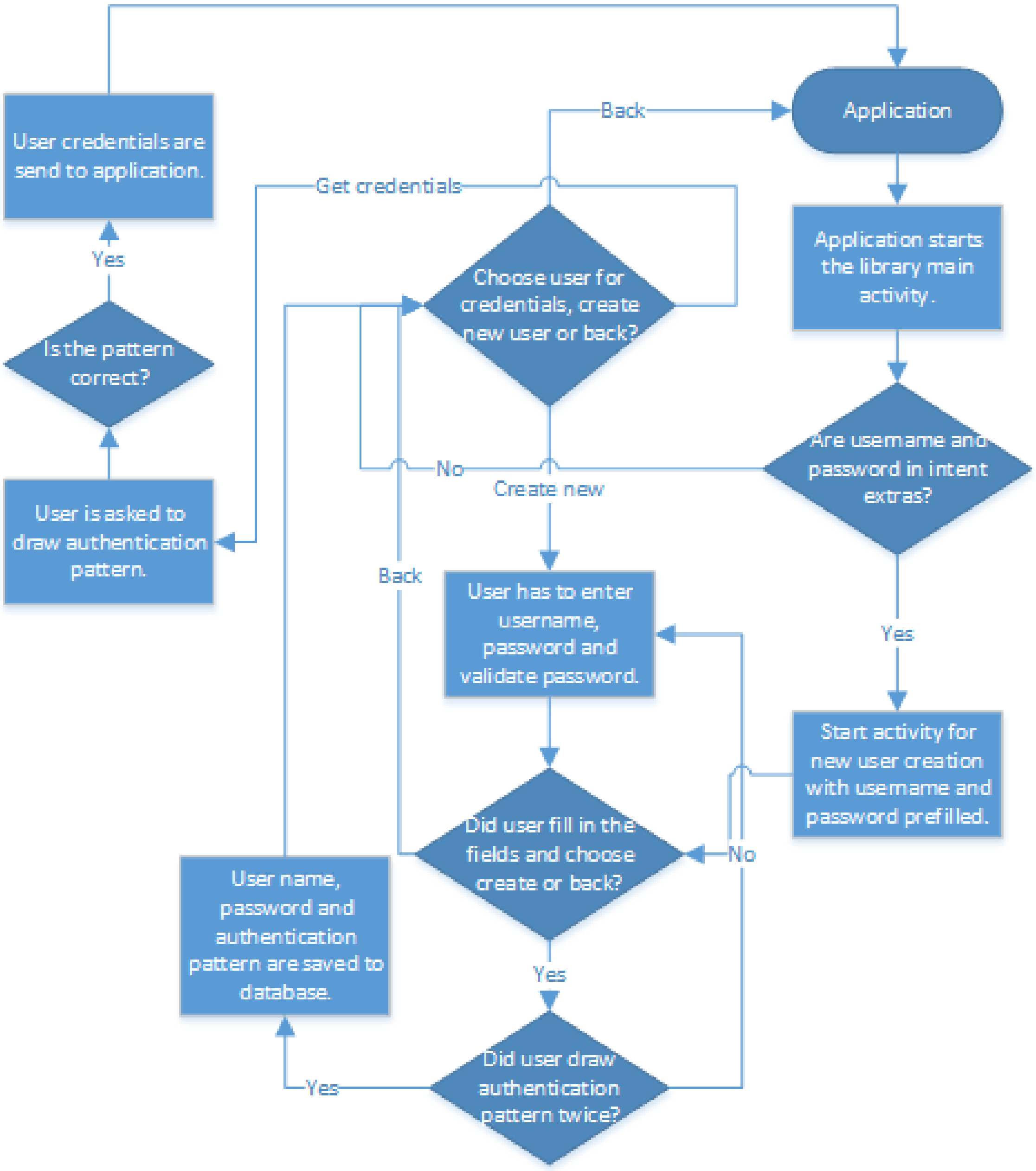

Application
Back
Get credentials
User credentials are send to application.
Application starts the library main activity.
Choose user for credentials, create new user or back?
Yes
Is the pattern correct?
Are username and password in intent extras?
No
Create new
User is asked to draw authentication pattern.
Back
User has to enter username, password and validate password.
Yes
Start activity for new user creation with username and password prefilled.
Did user fill in the fields and choose create or back?
No
User name, password and authentication pattern are saved to database.
Yes
Yes
Did user draw authentication pattern twice?

Figure 4.6: Flow chart of registration and authentication process

4.7 Importing the Library into a Project

This section is describing how to include this library in a project in terms of getting the library in a workspace and the code necessary to run it.

In order to use this library, both this and the Android-Lockpattern library are needed. This library is accessible in the projects repository[14], and also Android-Lockpatter library is accessible on github[15]. For development, eclipse is needed, as these libraries only work

[14] https://github.com/ennoeller/ThesisApi

[15] https://github.com/haibison/android-lockpattern

there. Within eclipse, both libraries have to be imported as projects, alongside with application project, which will include them.

In order for the library to work, a few lines of code have to be added to the application project. One row needs to be added to the project properties file, also make sure that a reference to the library is there:

```
manifestmerger.enabled=true
android.library.reference.1=../ThesisApi
```

Example usage of the library in an activity would look something like this:

```
import thesis.thesis.Commons;

public class Main extends Commons {

    @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      userName = (EditText) findViewById(R.id.username);
      passWord = (EditText) findViewById(R.id.password);

      Button bCallLibrary = (Button) findViewById(R.id.bCallLibrary);
      bCallLibrary.setOnClickListener(new View.OnClickListener() {
         public void onClick(View v) {

            saveGetCredentials(userName.getText().toString(),
               passWord.getText().toString());
         }
      });
   }
}
```

4.8 Summary

A pattern-based mechanism for authentication was developed as a library for Android. It uses patterns and therefore is very easy to use and also supports multiple users. For the end users the flow is kept minimalistic to increase simplicity and transparency. Use in a project is made very simple with an interface, as can be seen from the sample code.

In the next chapter, we compare the usability of the proposed mechanism with the traditional method of authentication to determine whether the developed method improves user experience.

5

Case Studies

Typing on a smartphone is made easier to the user by including auto-correct, but the same tool can not be used for authentication process. Therefore users are left with the task to hit every key accurately or repeat the task until they get it right.

In this chapter we present the results from a survey made to determine, whether the proposed authentication method increases the user experience.

5.1 Validation

In order to validate the hypothesis, the proposed solution is compared to the conventional ”input credentials” method in terms of simplicity and user experience. A use case based on social media sign-in was developed. The application is primitive to only test the authentication process.

In the validation, a tablet LG G Pad 8.3[16] was used. It has a 8.3 inch screen with the resolution of 1200x1920 pixels and is running Android 4.4.2, otherwise called as KitKat[17].

5.1.1 Experimental Setup

In the first application, only the conventional method is used as shown in Figure 5.1. The second application uses the proposed solution, which is divided into two parts: registering an account and authenticating with the stored account. Screenshots of the credentials saving process are seen in Figure 5.2. Once the library is integrated within the app, a button *Save/Get Credentials* appears (1). With the username and password filled in, pressing the mentioned button, the user is requested to validate his password by typing it for the second time (2). Pressing the button *Create*, the user can introduce a pattern (3), after drawing it twice, the process of credentials registration is complete and the user is taken to a list of accounts already stored (4). Screenshots of the authentication sequence

[16] http://www.gsmarena.com/lg_g_pad_8_3-5673.php

[17] https://www.android.com/versions/kit-kat-4-4/

are seen in Figure 5.3. The second step of authentication will either start from where registration ended or from a newly opened applications empty main page (1). Pressing the *Save/Get Credentials* button, a list of stored accounts is presented to the user (2). Selecting an account in the list, the user is prompted to draw a pattern (3) and on a successful authentication the user is taken back to the main page with the username and password filled in (4).

To compare these two methods, a questionnaire was composed |Appendix A|. It consists of 10 questions, first 2 are general questions about participants satisfactory to authentication process today. The next two sections, questions 3-6 and 7-10, are more specific to the authentication methods implied to applications used in this case study.

5.1.2 Methodology

There were 20 participants between the ages of 20 and 30, all day-to-day smartphone users. None of them have expertise in computer science, hence they are only end users. They were asked to first answer the first two questions, then they would perform authentication on the application with the conventional method and respond to question 3-6. Then they would register an account and authenticate themselves with the second application, also change the stored password or pattern and answer the last four questions.

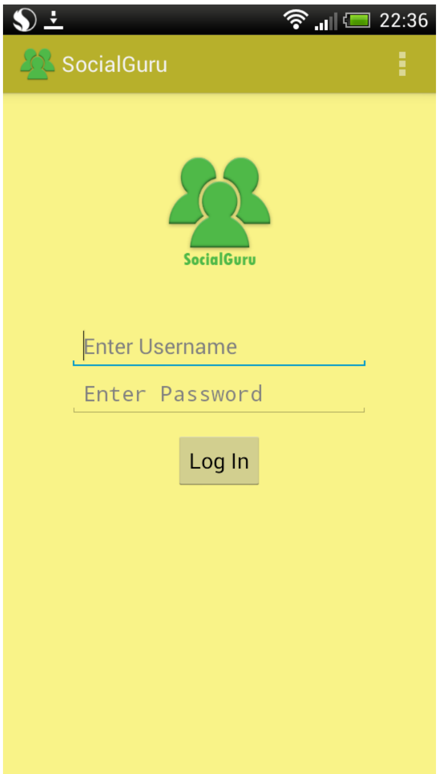


Figure 5.1: Application with only conventional method.

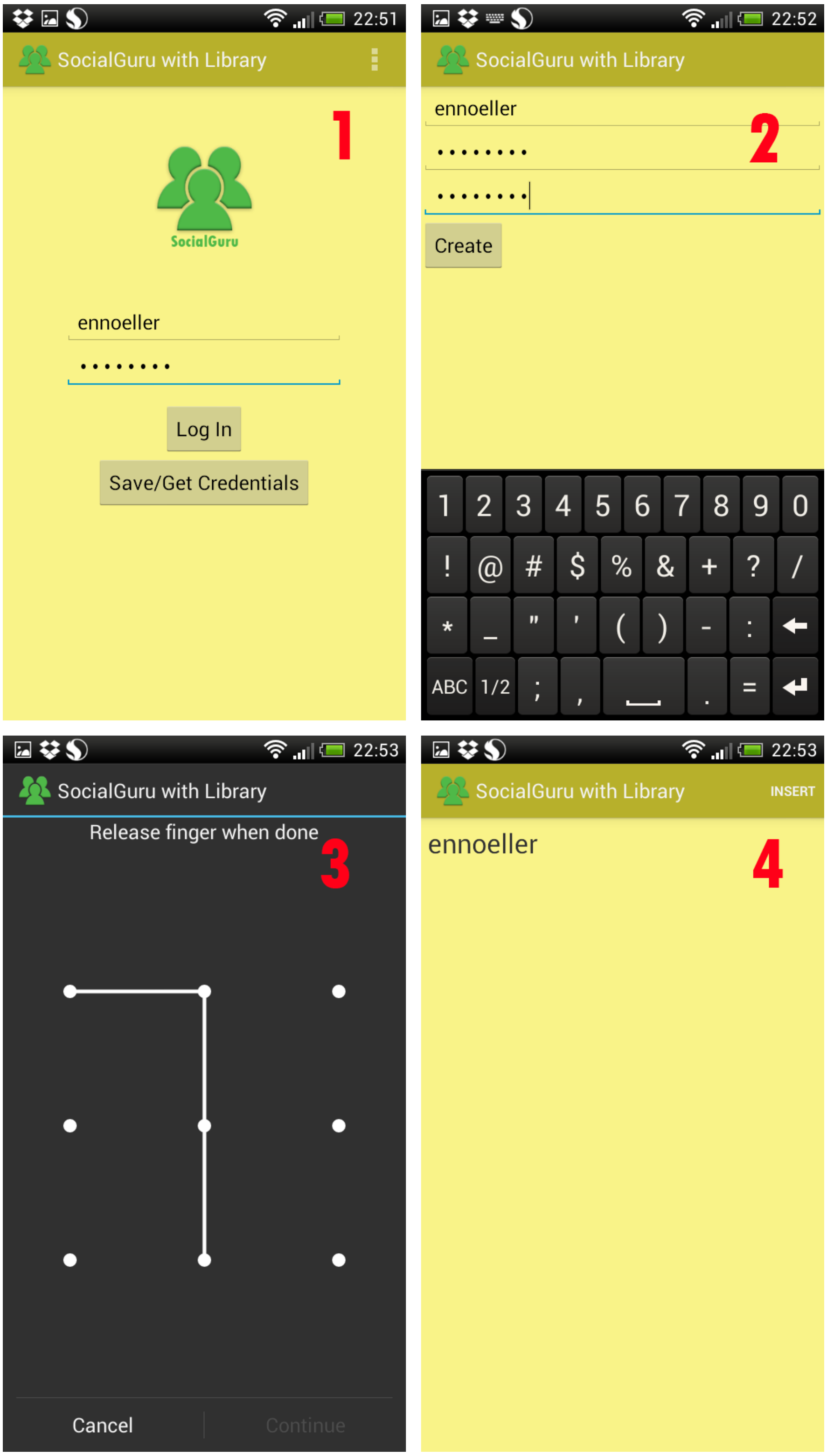


Figure 5.2: Credentials saving process with the library.

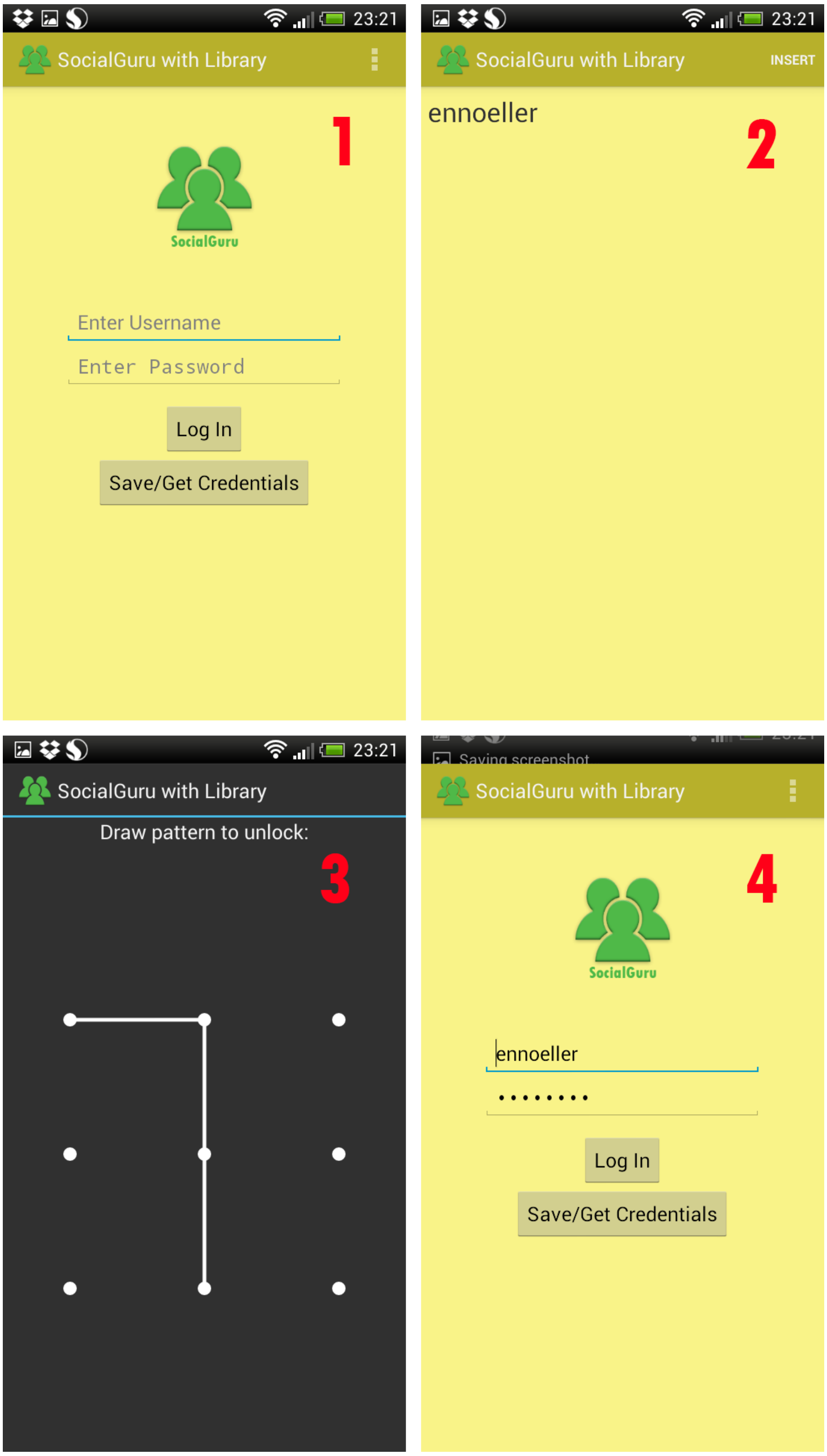


Figure 5.3: Athentication process with the library.

5.2 Results

First two questions were about participants opinions about authentication methods used in applications. 65% of them, shown in Figure 5.4, find that applications do use suitable login methods. 70% of participants, who find applications not to use suitable methods, brought out that the process should include less typing.

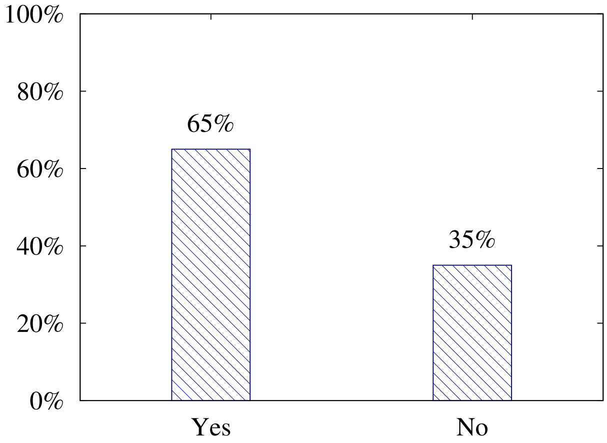


Figure 5.4: Participants satisfaction with the login methods used in apps.

5.2.1 Participants Opinion on Conventional Login

Most of the participants found the authentication method to be normal (45%) or even hard (34%) for them as seen on the Figure 5.5, implying to the size of the keyboard on the device. Similar results applied to the question of, whether they were annoyed by the process, shown at Figure 5.6. Given the answers values of 1 to 3, 1 being ”Easy/Low”, the average to both of these questing would be 2.15.

As seen from the Figures 5.7 and 5.8, the conventional login method would push users away from the application or make them change their passwords to something easier to type. 35% of the participants would definitely use the application less and 30% are considering it, making the application less appealing to 65% of users because of the authentication process. This method is also pushing 70% of the users to change their passwords, making them more vulnerable to intruders.

This shows how damaging the conventional method could be for the user base and reputation of the application. Also users are put to danger by allowing themselves to use weaker passwords.

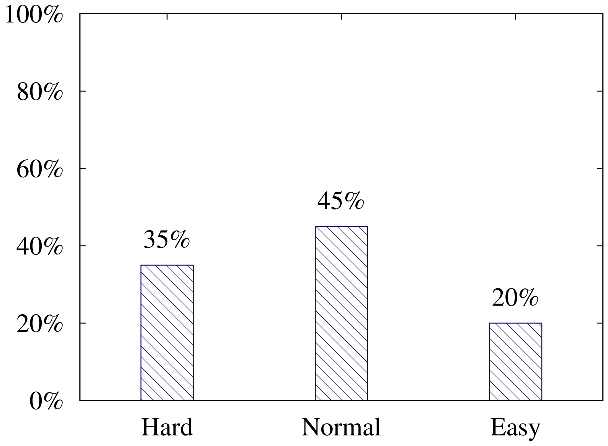


Figure 5.5: Complexity in the authentication process with conventional method.

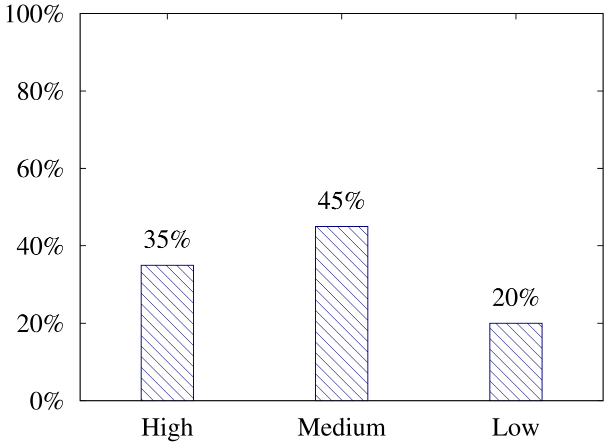


Figure 5.6: Level of annoyance caused by conventional method.

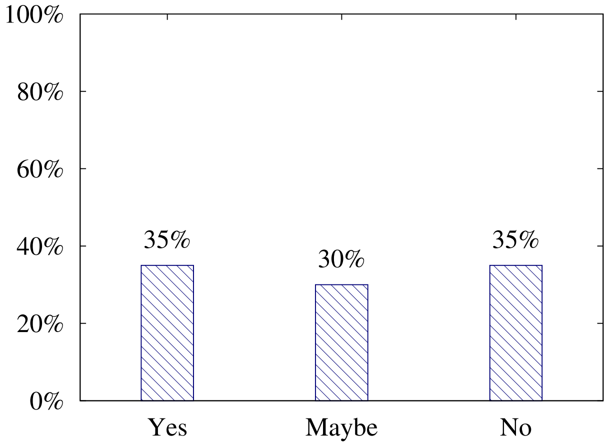


Figure 5.7: Participants being pushed away by the inconvenience.

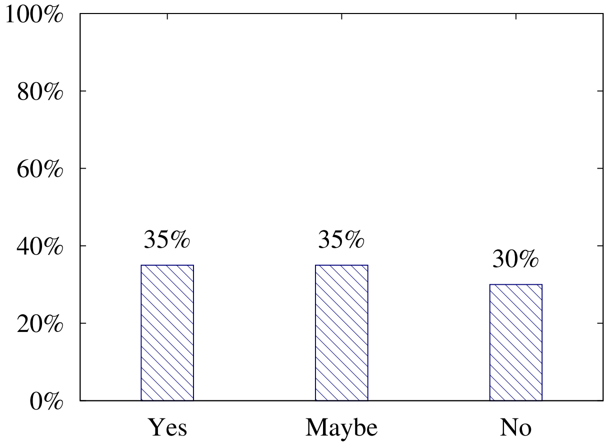


Figure 5.8: Discomfort making users change their passwords.

5.2.2 Participants Opinion on the Proposed Solution

The participants were using this method for the first time and were asked questions about the simplicity of it, to see whether this solution could be justified.

Compared to the conventional method, the proposed solution seems to be easier to use, see the Figure 5.9. Nobody thought it was hard. 75% of the participants found the method easy and 25% normal. Given the answers values of 1 to 3, 1 being ”Easy”, the average would be 1.25. This is also reflected on usage of the application, where nobody answered they would definitely use the application less, shown on Figure 5.10, because of the provided method. Only 15% of participants would consider using it less.

The given method was composed of two steps. Before authenticating themselves, they had to register their credentials with the application. 75% of the participants found the process easy and nobody felt confused, as seen on Figure 5.11. When asked to change the password or pattern saved in the application, some participants felt confused in the process - 25%, but 55% found it intuitive, as seen in Figure 5.12.

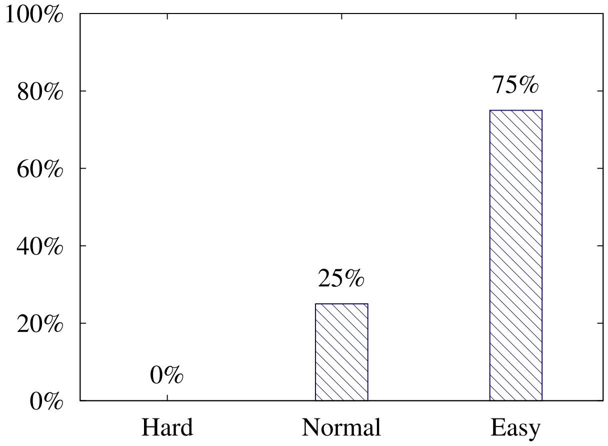


Figure 5.9: Complexity in the authentication process with pattern based method.

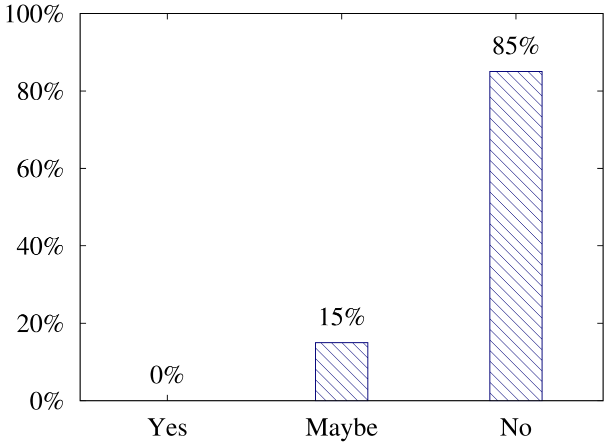


Figure 5.10: Participants being pushed away by the discomfort.

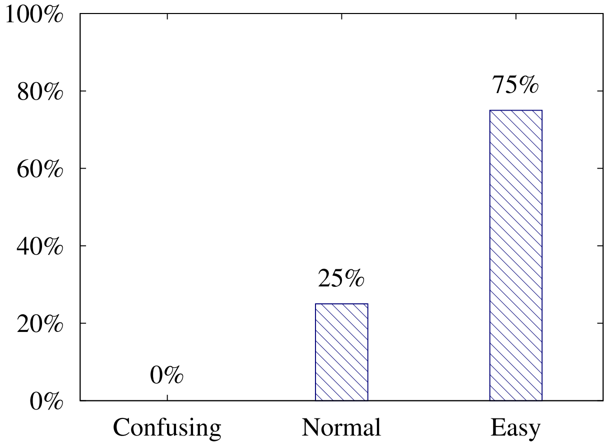


Figure 5.11: Simplicity of the credentials saving process.

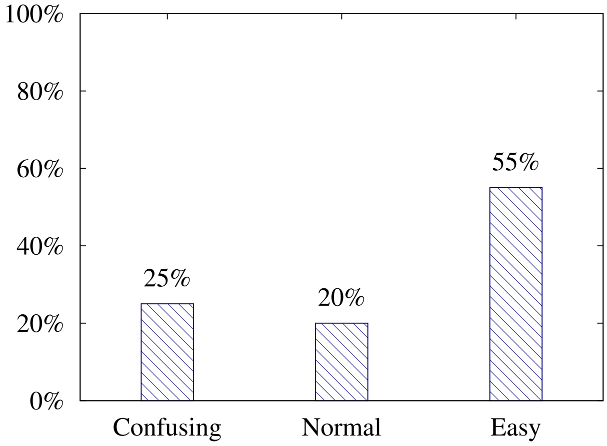


Figure 5.12: Simplicity of the password or pattern changing process.

5.3 Summary

In this chapter the usability of the proposed mechanism is validated. With the conventional model, 35% of the participants found it hard and annoying. 65% of participants felt the process to be repulsive and 70% sensed the need to change their passwords to something easier to type, therefore less secure. Opposed to the conventional method, the proposed solution was easier to the user, as 75% of the participants found it easy to use and therefore they were not losing interest in the application, as only 15% found it slightly repulsive. To store credentials, users have register their credentials and sometimes change their passwords. Although these are additional steps in the process, most participants find them easy to complete.

6

Conclusions and Future Directions

Mobile devices are becoming more common every day, but the authentication methods used are still those designed for PCs. The traditional username-password authentication method does not meet the characteristics of a smartphone and is therefore inconvenience to the user. It degrades the user experience, hence puts the mobile applications in a bad light.

To solve this problem, a library was developed to make the mobile authentication process more user-friendly. Instead of typing, a pattern-based authentication is used. It utilizes a small screen of a mobile device in better manner, giving the user more room for mistakes. This result in a better user experience.

Although, the usability of the developed solution was validated, it needs more work until it can be used in a real application. The credentials stored in a database are simple text and need to be encrypted. The interface could be made more intuitive and attractive for the user. Also range of tests should be conducted for any security and stability issues.

All in all, the feedback received from the survey group was strongly positive, implying there is room for improvements in the mobile authentication methods. It gives hope that, in the future the mobile device users could be introduced to new methods in authentication that appreciate the characteristics of a smartphone and are user friendly.

Bibliography

[1] H. Bojinov and D. Boneh. Mobile token-based authentication on a budget. In *Proceedings of the 12th Workshop on Mobile Computing Systems and Applications*, pages 14–19. ACM, 2011. 5

[2] A. Buchoux and N. L. Clarke. Deployment of keystroke analysis on a smartphone. In *Australian Information Security Management Conference*, page 48, 2008. 7

[3] M. O. Derawi, C. Nickel, P. Bours, and C. Busch. Unobtrusive user-authentication on mobile phones using biometric gait recognition. In *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, pages 306–311. IEEE, 2010. 6

[4] T. Feng, Z. Liu, K.-A. Kwon, W. Shi, B. Carbunar, Y. Jiang, and N. Nguyen. Continuous mobile authentication using touchscreen gestures. In *Homeland Security (HST), 2012 IEEE Conference on Technologies for*, pages 451–456. IEEE, 2012. 7

[5] W. Jansen. Authenticating users on handheld devices. In *Proceedings of the Canadian Information Technology Security Symposium*, pages 4–6, 2003. 5

[6] L. Li, X. Zhao, and G. Xue. Unobservable re-authentication for smartphones. In *NDSS*, 2013. 7

[7] C.-C. Lin, D. Liang, C.-C. Chang, and C.-H. Yang. A new non-intrusive authentication method based on the orientation sensor for smartphone users. In *Software Security and Reliability (SERE), 2012 IEEE Sixth International Conference on*, pages 245–252. IEEE, 2012. 7

[8] D. Marques, T. Guerreiro, L. Duarte, and L. Carriço. Under the table: Tap authentication for smartphones. In *Proceedings of the 27th International BCS Human Computer Interaction Conference*, page 33. British Computer Society, 2013. 5

[9] A. F. P. Negara, E. Kodirov, M. F. A. Abdullah, D.-J. Choi, G.-S. Lee, and S. Sayeed. ArmŠs flex when responding call for implicit user authentication in smartphone. *Int. J. Secur. Its Appl*, 6:879–83, 2012. 6

[10] D. Ritter, F. Schaub, M. Walch, and M. Weber. Miba: Multitouch image-based authentication on smartphones. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 787–792. ACM, 2013. 6

[11] S. N. Srirama, H. Flores, and C. Paniagua. Zompopo: Mobile calendar prediction based on human activities recognition using the accelerometer and cloud services. In *Next Generation Mobile Applications, Services and Technologies (NGMAST), 2011 5th International Conference on*, pages 63–69. IEEE, 2011. 6

[12] S. N. Srirama, C. Paniagua, and H. Flores. Social group formation with mobile cloud services. *Service Oriented Computing and Applications*, 6(4):351–362, 2012. 6

[13] C. Stein, C. Nickel, and C. Busch. Fingerphoto recognition with smartphone cameras. In *Biometrics Special Interest Group (BIOSIG), 2012 BIOSIG-Proceedings of the International Conference of the*, pages 1–12. IEEE, 2012. 5

[14] H. Takamizawa and N. Tanaka. Authentication system using location information on ipad or smartphone. *International Journal of Computer Theory and Engineering*, 4(2):153–157, 2012. 7

7

Appendices

Appendix A

1. Do you think that applications implement suitable methods to login?

   Yes No

2. If you answered no, how would you describe a suitable login method?

   ____

   ____

   ____

3. How difficult is to authenticate using apps that require login credentials?

   Hard Normal Easy

4. How annoying would you rate the login process in a mobile app?

   High Medium Low

5. You are using this application on daily basis. For some reason the application will not leave you signed in and you have to authenticate yourself often (once a week). Would you use the application less because of the inconvenience of authentication process?

   Yes Maybe No

6. You are using this application on daily basis. For some reason the application will not leave you signed in and you have to authenticate yourself often (once a week). Would you change your password to one that is easier to type (weaker password)?

   Yes Maybe No

7. After using this new method based on lockpattern, how difficult is to authenticate?

   Hard Normal Easy

8. You are using this application on daily basis. For some reason the application will not leave you signed in and you have to authenticate yourself often (once a week). Would you use the application less because of the inconvenience of authentication process?

   Yes Maybe No

9. How intuitive (easy to complete) was the process of saving your authentication credentials?

   Confusing Normal Easy

10. How intuitive (easy to complete) was the process of editing the password or pattern?

    Confusing Normal Easy

Appendix B

Non-exclusive licence to reproduce thesis and make thesis public

I, **Enno Eller** (date of birth: 20th of April 1991),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

1.1 reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

1.2 make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

Simplifying Mobile Social Media Authentication On Android

supervised by Huber Flores, MSc and Satish Srirama, PhD

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 04.08.2015