# Iteration 1 Report - Mobile Device Authentication

Contents

[Iteration 1 Report - Mobile Device Authentication 1](#_Toc434521456)

[General Overview 2](#_Toc434521457)

[End of iteration Functionality 2](#_Toc434521458)

[Stories 2](#_Toc434521459)

[Story # 1 –13 points 2](#_Toc434521460)

[Story # 2 – 8 points 2](#_Toc434521461)

[Sub Tasks 2](#_Toc434521462)

[Available End-of-Sprint Functionality 2](#_Toc434521463)

[Changes made to the Stories 3](#_Toc434521464)

[Lessons Learned 3](#_Toc434521465)

[User Stories Still to be implemented 3](#_Toc434521466)

[Story # 3 – 5 points 3](#_Toc434521467)

[Story # 4– 8 points 4](#_Toc434521468)

[Story # 5– 3 points 4](#_Toc434521469)

[Story # 6– 8 points 4](#_Toc434521470)

[Story # 7 – 5 points 5](#_Toc434521471)

[Story # 8– 8 points 5](#_Toc434521472)

[Subset of Stories for next Iteration 5](#_Toc434521473)

[Sprint 2 5](#_Toc434521474)

[Sub Tasks 5](#_Toc434521475)

[Available End-of-Sprint Functionality 5](#_Toc434521476)

# General Overview

This document is a summary review of our first of three iterations.

# End of iteration Functionality

The following stories and sub-tasks were successfully implemented during the first iteration.

## Stories

### Story # 1 –13 points

*“As a phone user, I want the ability to for my device to recognize my tap sequences, so that my phone in a later story will be able to successfully recognize me by my acceptable pattern for logging in.”*

### Story # 2 – 8 points

*“As a developer, I need the ability for the phone application to record the acceptable tapped sequence, so that I can ensure there is a base pattern for comparison at future login attempts.”*

## Sub Tasks

1. Create initial program in Java
2. Create UI display.
3. Create event handlers for tap pressing.
4. Capture the tap sequences and store them in memory.

## Available End-of-Sprint Functionality

1. A new application in Java.
2. A display that can read and process tap sequences.
3. Tap sequences are stored in memory.

# Changes made to the Stories

During this sprint, we really didn’t need to change, add, or remove functionality to these stories that we implemented. Everything went as planned.

# Lessons Learned

A lesson that we learned at the end of this iteration is that we probably shouldn’t have started the project early without a backlog of stories. The reason we started early was for our team members to become familiar with programming in Java and understanding how to use Android Studio. However, with no backlog implemented or any gathered requirements our project direction out of sync.

Each story needed to be broken down into smaller pieces. This allows for a better understanding of step-by-step operations that will be used to complete the overall task.

If we could do it differently next time, we would have considered to improve our communication about programming, the UI options, and the ideas before actually starting to code without an approved idea.

We could have used better coordination between team members in regards to who will be doing a specific part or feature. We fell into the traps of everyone trying to write the code for the same components at the same time.

We need to commit code and do it often, in smaller segments and not large chunks.

Finally, we should test code and test often. We need to do a better job of testing code before commits and after.

# User Stories Still to be implemented

## Story # 3 – 5 points

*“As a phone user, I want the ability to enter a tap sequence for authentication.”*

**Pre-Conditions:**

Display at area within a screen that a user can enter their pattern.

**Post-Conditions:**

Comparable authentication pattern stored locally with user current tap pattern.

## Story # 4– 8 points

*“As a phone user, I expect the phone to successfully compare my taping pattern to my acceptable pattern, so that I can ensure that my device can know that it is me.”*

**Pre-Conditions:**

User has entered a tapping sequence to be used for comparison.

**Post-Conditions:**

Grant or deny access to application for the user.

## Story # 5– 3 points

*“As a developer, I need to be able to show the accepted or rejection message back to the user for their login attempt, so as to ensure a response was properly acknowledged.”*

**Pre-Conditions:**

User has attempted to enter a tap sequence pattern, and that pattern has been successfully compared to the successful one.

**Post-Conditions:**

Displays notification of invalid authentication or send user the main screen of the application.

## Story # 6– 8 points

*“As a developer, I need the phone to record tapped in data that the user inputs, so that I can ensure that data is valid and working.”*

**Pre-Conditions:**

User entered tap sequence patterns as attempts for logging in.

**Post-Conditions:**

Data for the tap sequence attempts will have been recorded locally in a history table.

## Story # 7 – 5 points

*“As a developer, I need to ensure that the phone doesn’t record tapped data indefinitely, so that I can ensure memory is not wasted. I need to check against a threshold.”*

**Pre-Conditions:**

A certain number or limit of tap sequence patterns as attempts for logging in must exist in local storage.

**Post-Conditions:**

A certain number for a threshold will be successfully compared against.

## Story # 8– 8 points

*“As a developer, I need the phone application to offload data to be stored onto a PC, so that I can ensure that data is retained for any future purposes.”*

**Pre-Conditions:**

A limit of tap sequence patterns as attempts for logging in must exist in local storage or table.

Connect the mobile device via USB to computer to copy stored tap pattern data.

**Post-Conditions:**

Data will be offloaded and cleared from local storage, ready to record more data.

Disconnect the mobile device from computer.

# Subset of Stories for next Iteration

## Sprint 2

Size in Points: 16

Stories

### Sub Tasks

1. Create UI buttons for the recording of an official authentication process.
2. Add a component or module to request an authentication pattern.
3. Add logic to compare future logins with recorded authentication tap sequence in memory.
4. Add code to display results to end-user.

### Available End-of-Sprint Functionality

1. UI layout and functionality to actually record an authentication tap sequence from the user.
2. The phone should be able to successfully compare any login attempt against the acceptable authentication tap sequence.
3. The phone will be able to display the results of the login attempt to the user.