# Iteration 2 Report - Mobile Device Authentication

Contents

[Iteration 2 Report - Mobile Device Authentication 1](#_Toc435739735)

[General Overview 2](#_Toc435739736)

[End of iteration Functionality 2](#_Toc435739737)

[Stories 2](#_Toc435739738)

[Story # 3 –5 points 2](#_Toc435739739)

[Story # 4 – 8 points 2](#_Toc435739740)

[Story # 5 – 3 points 2](#_Toc435739741)

[Sub Tasks 2](#_Toc435739742)

[Available End-of-Sprint Functionality 2](#_Toc435739743)

[Changes made to the Stories 3](#_Toc435739744)

[Lessons Learned 3](#_Toc435739745)

[User Stories Still to be implemented 3](#_Toc435739746)

[Story # 4 – 8 points (Carry-Over from Iteration 2) 3](#_Toc435739747)

[Story # 6– 8 points 3](#_Toc435739748)

[Story # 7 – 5 points 4](#_Toc435739749)

[Story # 8– 8 points 4](#_Toc435739750)

[Subset of Stories for next Iteration 4](#_Toc435739751)

[Sprint 3 4](#_Toc435739752)

[Sub Tasks 4](#_Toc435739753)

[Available End-of-Sprint Functionality 4](#_Toc435739754)

# General Overview

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

# End of iteration Functionality

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

## Stories

### Story # 3 –5 points

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

### Story # 4 – 8 points

1. *~~“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.”~~* Wasn’t completed in time, will be available next iteration.

### 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.*

## 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.~~ – Wasn’t completed in time, will be available next iteration.
3. The phone will be able to display the results of the login attempt to the user.

# Changes made to the Stories

During this sprint, we really didn’t need to remove functionality to these stories that we implemented. However, we didn’t successfully complete story 4 on time. When we were implementing the algorithm for comparing tap sequences, we kept receiving a result of ‘infinity’ instead of an actual numbered response. This is going to be fixed and available in the next iteration.

# Lessons Learned

We successfully implemented the user interface to ask the user to input their pattern for recognition, and we are also now successfully capturing the testing sequences for authentication. This part of the iteration went very well.

What didn’t go well so far is the implementation of the comparison algorithm. Things that hurt the team progression during this iteration were procrastination, a lack of communication, and a poor usage of when team members had available time and resources. We probably could have implemented the comparison algorithm on time if we didn’t face such issues. In the final iteration, we will be sure to make up for story work that wasn’t completed on time.

# User Stories Still to be implemented

### Story # 4 – 8 points (Carry-Over from Iteration 2)

*“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.”*

## 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 3

Size in Points: 29

### Sub Tasks

1. Successfully compare any login attempt against the acceptable authentication tap sequence.
2. Add functionally to record all tapped sequences locally in a text file or database.
3. Add code and logic to check the count of history records up against a threshold.
4. Add code and functionality to offload data to a text or Excel file that can be used later on a PC.

### Available End-of-Sprint Functionality

1. A history of data tap sequences will be available on the device for review.
2. A threshold will be in place to ensure that data for tap sequences isn’t recorded indefinitely.
3. Offloaded data in an acceptable file format will be available for review from outside devices such as a PC.