# Product Backlog - Mobile Device Authentication

Contents

[Product Backlog - Mobile Device Authentication 1](#_Toc433284751)

[General Overview 2](#_Toc433284752)

[Estimated Story Sizes 2](#_Toc433284753)

[Story # 1 –13 points 2](#_Toc433284754)

[Story # 2 – 8 points 2](#_Toc433284755)

[Story # 3 – 5 points 2](#_Toc433284756)

[Story # 4 – 8 points 3](#_Toc433284757)

[Story # 5 – 3 points 3](#_Toc433284758)

[Story # 6 – 8 points 3](#_Toc433284759)

[Story # 7 – 5 points 3](#_Toc433284760)

[Story # 8 – 8 points 3](#_Toc433284761)

[Cumulative Product Backlog List 4](#_Toc433284762)

[Story pre- and post-conditions 4](#_Toc433284763)

[Story # 1 4](#_Toc433284764)

[Story # 2 4](#_Toc433284765)

[Story # 3 5](#_Toc433284766)

[Story # 4 5](#_Toc433284767)

[Story # 5 5](#_Toc433284768)

[Story # 6 5](#_Toc433284769)

[Story # 7 6](#_Toc433284770)

[Story # 8 6](#_Toc433284771)

[Iterations (Sprints) 6](#_Toc433284772)

[Sprint 1 6](#_Toc433284773)

[Sub Tasks 6](#_Toc433284774)

[Available End-of-Sprint Functionality 7](#_Toc433284775)

[Sprint 2 7](#_Toc433284776)

[Sub Tasks 7](#_Toc433284777)

[Available End-of-Sprint Functionality 7](#_Toc433284778)

[Sprint 3 7](#_Toc433284779)

[Sub Tasks 7](#_Toc433284780)

[Available End-of-Sprint Functionality 7](#_Toc433284781)

[Design 7](#_Toc433284782)

[Tap Sequence 7](#_Toc433284783)

# General Overview

This document is a product backlog that will refine our user stories through estimating and sizing, as well as identify the sub-tasks (subsets of user stories) that will be implemented. Also, this document will break down the required stories into three separate sprints.

# Estimated Story Sizes

## 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.”*

## 

## Story # 3 – 5 points

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

## 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.”*

## 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.”*

## 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.”*

## 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.”*

## 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.”*

# Cumulative Product Backlog List

|  |  |  |
| --- | --- | --- |
| Story # | Story Title | Point Size |
| 1 | Initial tap sequence set-up. | 13 |
| 2 | Record acceptable tab sequence. | 8 |
| 3 | Provide tap sequence request for authentication. | 5 |
| 4 | Tap sequence comparison comparison\algorithm. | 8 |
| 5 | Display accepted or rejected message to user. | 3 |
| 6 | Record tap sequences history. | 8 |
| 7 | Check for tap sequence history threshold. | 5 |
| 8 | Offload Data to text file to be stored elsewhere | 8 |
|  | Total | 58 |

# Story pre- and post-conditions

## Story # 1

**Pre-Conditions:**

The phone user installs the application to require the configuration of Tap Sequence to access use the desired content of the application.

**Post-Conditions:**

Authentication pattern will stored locally on device for future user authentication.

## Story # 2

**Pre-Conditions:**

User has entered the tapping sequence that they want.

**Post-Conditions:**

Stored the collected pattern in a memory.

## Story # 3

**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

**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

**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

**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

**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

**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.

# Iterations (Sprints)

## Sprint 1

Size in Points: 21

### 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 will be stored in memory.

## Sprint 2

Size in Points: 16

### 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.

## Sprint 3

Size in Points: 21

### Sub Tasks

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

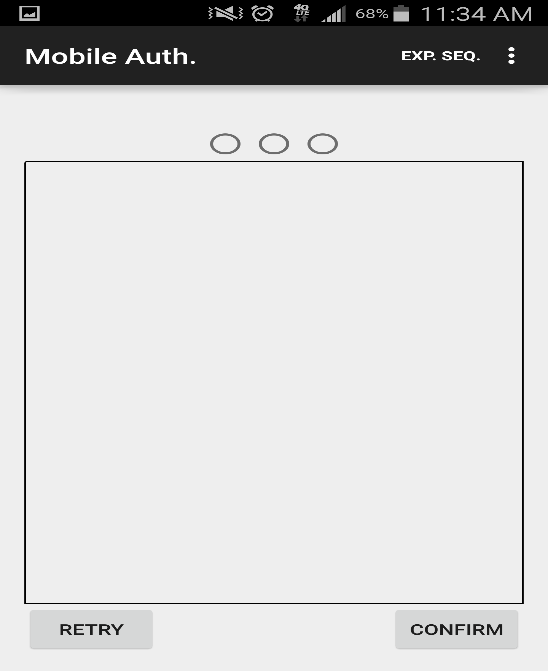
### Available End-of-Sprint Functionality

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.

# Design

## Authentication

* UI in which the user can user a select or entire of area of the screen to enter tap sequence for authentication



## Tap Sequence

Provides the user with a simple to use means of configuring a tap sequence

* The three adjacent radio button indicate the current sequence currently being configured
* Rectangular area provides represent the area in which the use can enter a tap sequence
* The retry button will reset the enter tap sequence entry and the radio button
* The confirm button will proceed to the next tap sequence iteration to be configured if previous sequence is correct. Once all sequences match and validated user information will be stored.

