

# Sprint 1 Report

**Public Defender**

6\_angry\_men

April 28, 2017

## **Actions to Stop Doing:**

During this time, we do not have anything to add to this section. We believe that we have a good rhythm, but it just needs to be fine-tuned. Each of us regularly attend class, and therefore are available for each of our stand-up meetings (directly after class), we are following the Scrum practices as outlined in class, and each tries their best to contribute wherever they can. It's a good team. If we have any problems that arise in future sprints, we will add it.

## **Actions to Start Doing:**

1. Ask Others When Help is Needed
  - a. We all get busy, and sometimes we get behind on a task because something is not working. Having fresh eyes on the problem will be a helpful solution
2. Define a Standard of Being Done and Merge Procedures
  - a. Ensure that whenever we finish a task we define how to merge it and alert our PO/SM that it is potentially done.
3. Define a Standard of Documentation
  - a. This will be especially important when we start putting everything together
4. Give Ourselves Plenty of Time to Test the "Finished" Code
  - a. We should start including this metric in our Planning Poker sessions.
5. Get Better at Planning Poker
  - a. This will probably come with time, but it is a good goal to set for ourselves.

## **Actions to keep doing:**

1. Meeting at 9am, three times a week
  - a. This meeting allows the team to come together and get a game plan for the coming day or weekend.
2. Meeting on the Weekends
  - a. This gives us a time, off campus, where we can sit and plan, and work on things that have been giving us trouble throughout the week.
3. Communicating on Slack
  - a. I probably don't need to explain why communication is great.
4. Having Awesome Sprint Planning Sessions
  - a. Fully hashing out the user stories is extremely helpful.
5. Playing Planning Poker
  - a. It's a great way to estimate task complexity, and discuss why we believe that task should or should not be complex.

### **Work Completed/not Completed:**

1. Completed User Stories
  - a. As a developer I want a full development project in the bitbucket, so I can develop
    - i. Prepare/Test Android Project
    - ii. Setup Node JS Server
    - iii. DB Schema and Sample Metadata
    - iv. Wrap Docker
    - v. UI Mockup by hand
2. Not Completed User Stories
  - a. As a user I want to be able to record my interactions with police (authority) officers to hold them accountable
    - i. Tasks Unfinished
      1. Stream Audio File to Server
      2. Basic UI
      3. View Local Recordings
  - b. As a user I want to be aware of interactions with police in my community, in order to participate in the gist of the app
    - i. Tasks Unfinished
      1. Show Server Query Response in List View
      2. Query DB for a Nearby Incidence(s)
3. Partial Completion List (important tasks that were completed, but did not complete a user story)
  - a. From: As a user I want to be able to record my interactions with police (authority) officers to hold them accountable
    - i. Save sound files locally
    - ii. Server Receive Stream
    - iii. Collect Samples
  - b. From: As a user I want to be aware of interactions with police in my community, in order to participate in the gist of the app
    - i. GeoLocation API

### **Work completion Rate:**

At this time, we would gauge our completion rate at about 70% of what we sought to do, even though we only completed a single user story. We believe that we obtained this rate for a couple of reasons. First, we think that we underestimated the amount of time that would be required for a few of the tasks. Several of the tasks that we assumed would be easy turned out to be the most troublesome. Second, we never really defined what “done” would be. At the last minute, we decided that what would be considered “done” is that which is in the develop branch, and since most of the team has been working in their own branches with their own troubles, combining the code such that it all worked became a challenge. For the future, and to hopefully increase our completion rate, we plan to spend more time in planning the sprint and really

hammering out the details for each user story. We also intend to really lay out what should be considered finished, and what is not.

*Payton Schwarz, Max Glynn, Oliver Davies, Seth Larkin, Bryan McCoid, Brandon McKeand*