

# Oakland Inconvenience Tracker

## Sprint Review Report

Sprint 2

1.0

11/7/18

Dylan Sporrer

David Rowan

Brandon Donahue

Adam Farabaugh

Software Engineers

Prepared for

CS 1530

Fall Term 2018

University of Pittsburgh

## Revision History

Date	Description	Author	Comments
11/6/18	Initial Creation	Dylan Sporrer	
11/7/18	Update SID 06	Brandon Donahue	
11/7/18	Final Version	Dylan Sporrer	

## Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Dylan Sporrer	Software Eng.	
	David Rowan	Software Eng.	
	Brandon Donahue	Software Eng.	
	Adam Farabaugh	Software Eng.	

## Table of Contents

Revision History	2
Document Approval	2
1. Introduction	4
2. Specific Goals	4
3. Analytics	5
3.1 Sprint/Product Burndown Chart	5
3.2 Sprint Velocity	5
4. Conclusion	6

## 1. Introduction

The primary goal of this sprint was to begin to create the individual actors of the system but not the links between these actors. Each story covers the development of one of the primary actions of the system, namely: the class responsible for allowing users to enter listing information, the live map which displays listings, and the database communicator class which translates information to and from the database which supports listings and profile information.

Of the planned stories, only the creation of the listing class was completed. An unexpected degree of complexity arose during work on the other two stores, halting progress until the actor's exact specifications can be addressed in the next sprint.

In meeting less than a majority of its goals, this sprint has failed to transition the project into a stage in which early forms of all user-side actions can be carried out. Going forward, the individual actors must be finalized before the team can begin to assemble the system as a whole.

## 2. Specific Goals

### 2.1 Create Listing Submission Class (SID: 05)

#### 2.1.1 Story Description:

This story covers the creation of a user interface to allow for the input of listing information and the class which collates and communicates these inputs to the database communicator

#### 2.1.2 Story Acceptance Criterion:

This story is considered complete when a user can input a complete set of listing information and see this information reflected in a JSON file capable of being read by the database communicator

#### 2.1.3 Story Dependencies:

This story has no dependencies

#### 2.1.4 Story Challenges:

Uncertainty in where to handle the collation of inputs (whether in the HTML file or in a separate driver) gave initial pause in the process of this story

#### 2.1.5 Story Assigned to:

Dylan Sporrer

#### 2.1.6 Story Points:

2

#### 2.1.7 Status:

Complete

### 2.2 Develop Live Map Display of Listings (SID: 06)

#### 2.2.1 Story Description:

This story covers the ability to update the live map to reflect changes in the database, giving us tools to communicate with the map to add markers, descriptions, and other visual representations of our data.

#### 2.2.2 Story Acceptance Criterion:

This story is considered complete when other classes of the system can communicate with the map and send it data which it will then update visually for the user.

#### 2.2.3 Story Dependencies:

Database class.

#### 2.2.4 Story Challenges:

Difficulty in learning new technologies for the Google Maps API and deliberation on the format of data being transferred caused disruptions in the story.

#### 2.2.5 Story Assigned to:

Brandon Donahue

#### 2.2.6 Story Points:

2

#### 2.2.7 Status:

Incomplete

## 2.3 Database Communicator Class (SID: 07)

### 2.3.1 Story Description:

This story covers creating a class to create a database connection string that connects to the amazon RDS database, allowing for other classes to be built upon it for easy database manipulation. It also entails the creation of the basic set of these classes.

### 2.3.2 Story Acceptance Criterion:

This story is considering complete when one is able to connect to the database with a OK status and it is possible to accept data from the user interfaces as well as the database.

**2.3.3 Story Dependencies:** Dependent on availability of AWS RDS SQL Server Services

**2.3.4 Story Challenges:** Difficulties using the selected toolset to achieve the desired goals as well as a greater depth of complexity than was initially expected lead to disruptions in the story.

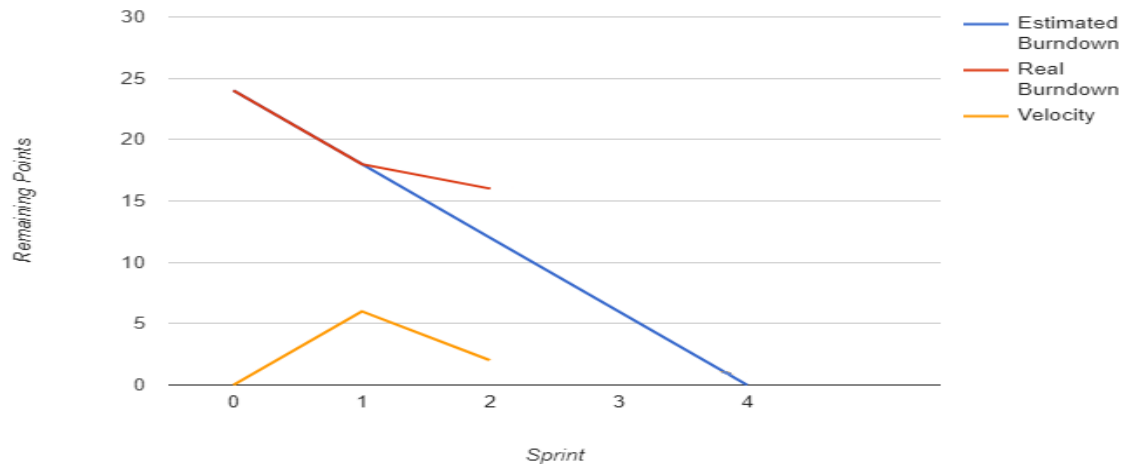
**2.3.5 Story Assigned to:** David Rowan, Adam Farabaugh

**2.3.6 Story Points:** 2

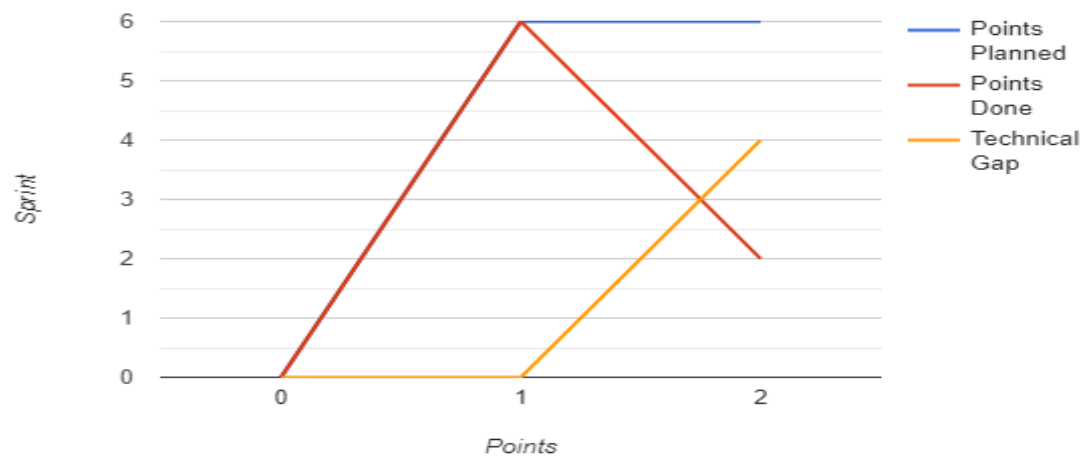
**2.3.7 Status:** Incomplete

## 3. Analytics

### 3.1 Sprint/Product Burndown Chart (sample chart shown below)



### 3.2 Sprint Velocity (sample chart shown below)



## **4. Conclusion**

This second sprint was able to be partially completed thanks to the foundation established in the previous sprint, but it was not without issues. The development team consensus is that a lack of thorough communication between members working on somewhat disparate systems has resulted in some stories having over-lapping scopes. To avoid this problem, the next sprint will focus heavily on the integration of existing systems, ideally resulting in a product which carries out the primary function of the design from start to finish.