Sprint Review Report

SLYYDE

Sprint Review Report

Sprint 3

Version 1

4/2/2019

Andrew Dodel, Yuxin Zhang, Sundar Sampath, Brendan Campbell, Andrew Jia

Software Engineer

Prepared for

CS 1530

Spring 2019

University of Pittsburgh

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 4/2/2019 | Version 1 | Andrew Dodel |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  |  |  |  |
| *Brendan C.* | Brendan Campbell | Software Engineer | 4/2/2019 |
| *Andrew Jia* | Andrew Jia | Software Engineer | 4/2/2019 |
| *Sundar Sampath* | Sundar Sampath | Software Engineer | 4/2/2019 |
| *Yuxin Zhang* | Yuxin Zhang | Software Engineer | 4/2/2019 |
| *Andrew Dodel* | AndrewDodel | Software Engineer | 4/2/2019 |

**Table of Contents**

**REVISION HISTORY II**

**DOCUMENT APPROVAL II**

**1. INTRODUCTION 1**

**2. SPECIFIC GOALS 1**

**3. ANALYTICS 2**

**4. CONCLUSION 3**

# Introduction

The goal for this sprint was to improve upon the backend system’s functionality and the general aesthetics of the user-view, since the game functionality was finished in sprint 1. To this end, a few features for the backend were necessary to implement so that all the components can be combined in the upcoming sprint 4. These features were allowing a user to upload pictures to the server, allowing the server to store high scores, and allowing the server to manipulate those high scores.

Due to the unpredictable nature of working with Flask and the time needed to work through the errors associated with implementing new server functionality, most of the story points were allocated towards these features than any other development.

# 2. Specific Goals

***2.1 Story Name & Number***

**Store High Scores**

***2.1.1 Story Description:***

Create a way for a list of high scores to be stored in the database

***2.1.2 Story Acceptance Criterion***

An organized list of times and associated users must be saved in the app.db file

***2.1.3 Story Dependencies***

This story is dependent on the “Basic Database Construction” story from sprint one, and builds off it.

***2.1.4 Story Challenges***

Lists or Maps can’t be stored in a database table, and these are the most intuitive way of doing high scores, so a workaround method will be required.

***2.1.5 Story Assigned to***

Andrew Dodel

***2.1.6 Story Points***

*2*

***2.1.7 Status: Completed or not***

Completed.

***2.2 Story Name & Number***

**Add High Scores**

***2.2.1 Story Description:***

Create a way to evaluate a score received from a user and add it to a high score list if it qualifies to be in the top 10.

***2.2.2 Story Acceptance Criterion***

Must be able to add score to a high score list if empty or if the score is worthy of being on the list. Pre-existing scores should be adjusted accordingly.

***2.2.3 Story Dependencies***

This story is dependent on the “Basic Database Construction” story from sprint one, and on the “Store High Scores” story.

***2.2.4 Story Challenges***

Since high scores will be stored as long strings, it will be a complicated process to parse them, identify scores and associated users, map the two, evaluate an input score, and then adjust the map accordingly before re-forming and storing the high score string.

***2.2.5 Story Assigned to***

Andrew Dodel

Andrew Jia

***2.2.6 Story Points***

*4*

***2.2.7 Status: Completed or not***

Not Completed. Working functionality in Java, just needs to be converted to Python.

***2.3 Story Name & Number***

**Upload Picture**

***2.3.1 Story Description:***

Create a way for a user to upload a picture and have it stored on the server.

***2.3.2 Story Acceptance Criterion***

An image file (and only an image file) should be able to be uploaded and stored in the server upload folder

***2.3.3 Story Dependencies***

This story is dependent on the “Basic Database Construction” story from sprint one, and builds off it.

***2.3.4 Story Challenges***

I have never implemented something like this before, and expect to have lots of challenges working through errors.

***2.3.5 Story Assigned to***

Andrew Dodel

***2.3.6 Story Points***

*6*

***2.3.7 Status: Completed or not***

Completed.

***2.3 Story Name & Number***

**Client-side CSS Improvement**

***2.3.1 Story Description:***

Improve client-side CSS to be more visually appealing, and give pages a uniform feel.

***2.3.2 Story Acceptance Criterion***

CSS modifiers should be made and applied to existing webpages.

***2.3.3 Story Dependencies***

This story is dependent on all stories that entailed the creation of .html files.

***2.3.4 Story Challenges***

Finding a visually appealing color scheme and creating a way for it to be uniformly applied to all HTML elements will be the most difficult part of this story.

***2.3.5 Story Assigned to***

Yuxin Zhang

***2.3.6 Story Points***

*3*

***2.3.7 Status: Completed or not***

Not Completed.

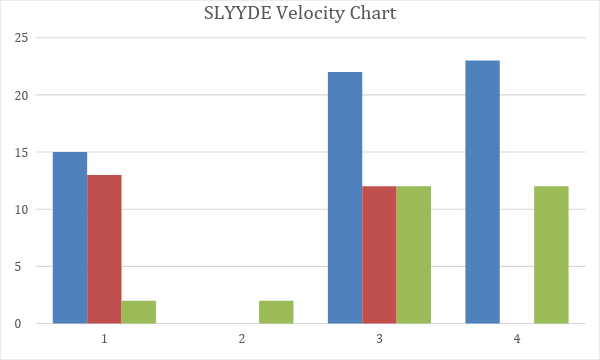
# 3. Analytics

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

### 

### 

## 3.2 Sprint Velocity



# 4. Conclusion

Sprint 3 was mostly about finishing up most of the functionality behind everything before combining everything in Sprint 4. Sprint 4 will also include the requesting and sending of pictures from the server to the game client, the finishing of the game client, and the implementation of authentication/encryption on user login. As a whole, our group is in a very good position going into the last sprint of the project. The only two missed stories are the CSS, which is a work in progress, and the high score adding, which will probably be finished just after the report is due.