**Game Design Document**

***Doodlebeat***

| Name | E-mail |
| --- | --- |
| Erica Nguyen | **ericangu@usc.edu** |
| David The | **dthe@usc.edu** |
| Jonathan Wang | **jwang414@usc.edu** |
| Ramya Gopinath | **rgopinat@usc.edu** |
| Yanhao Jin | **yanhaoji@usc.edu** |
| Jun Shao | **jshao290@usc.edu** |
| Jiayi(Carrie) Chang | **Jiayicha@usc.edu** |
| Chinmay Sankhe | **csankhe@usc.edu** |
| Rachita Kagalvadi Shivappa | **Kagalvad@usc.edu** |
| Saurav Sudhakar Borse | **borse@usc.edu** |

# Role assignments:

Audio - Chinmay

User Interface - Jun

UX / Game Feel - Chinmay & Yanhao

Analytics - Jonathan & David

Programming

* Physics - Erica
* Timing of player hitting notes - David

Design

* Level Design - Ramya & Rachitha

Version Control - Saurav

Layout design - Carrie

Producer - Chinmay

Team Lead - Erica

# GitHub and WebGL Gameplay Links

Github: <https://github.com/csankhe/Rhythm-game>

WebGL: <https://ericapikapika.itch.io/csci-526-prototype>

# Game Introduction

## 1.1 Idea

A puzzle-rhythm game where the player draws lines to hit the beat. The lines must be drawn in a way so that they do not intersect.

## 1.2 Game Element

1. Beats to hit
2. Shapes to drag to hit the beat
3. Lines to draw

## 1.3 General Description

Our game is a rhythm-puzzle game. The gameplay mechanic is a hybrid between Two Dots (a puzzle game where players connect dots of the same color by drawing a path between them, and paths cannot overlap) and Osu (a rhythm game where the player hits music beats by tapping and sliding them across the screen). In our game, the player connects dots of the same color in time to the music by drawing a path between the dots. The paths cannot overlap.

## 1.4 Goal

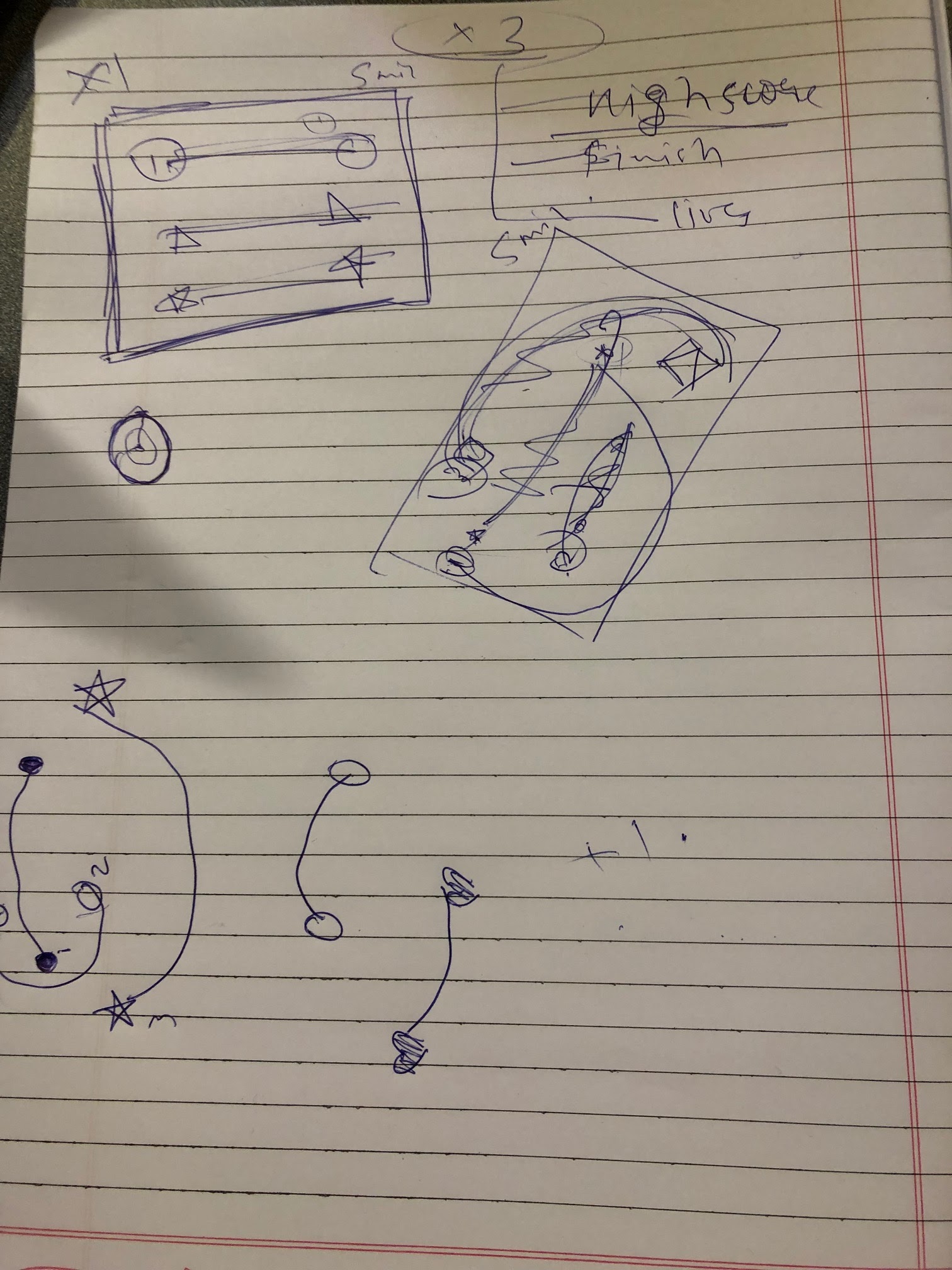
Get the highest score.

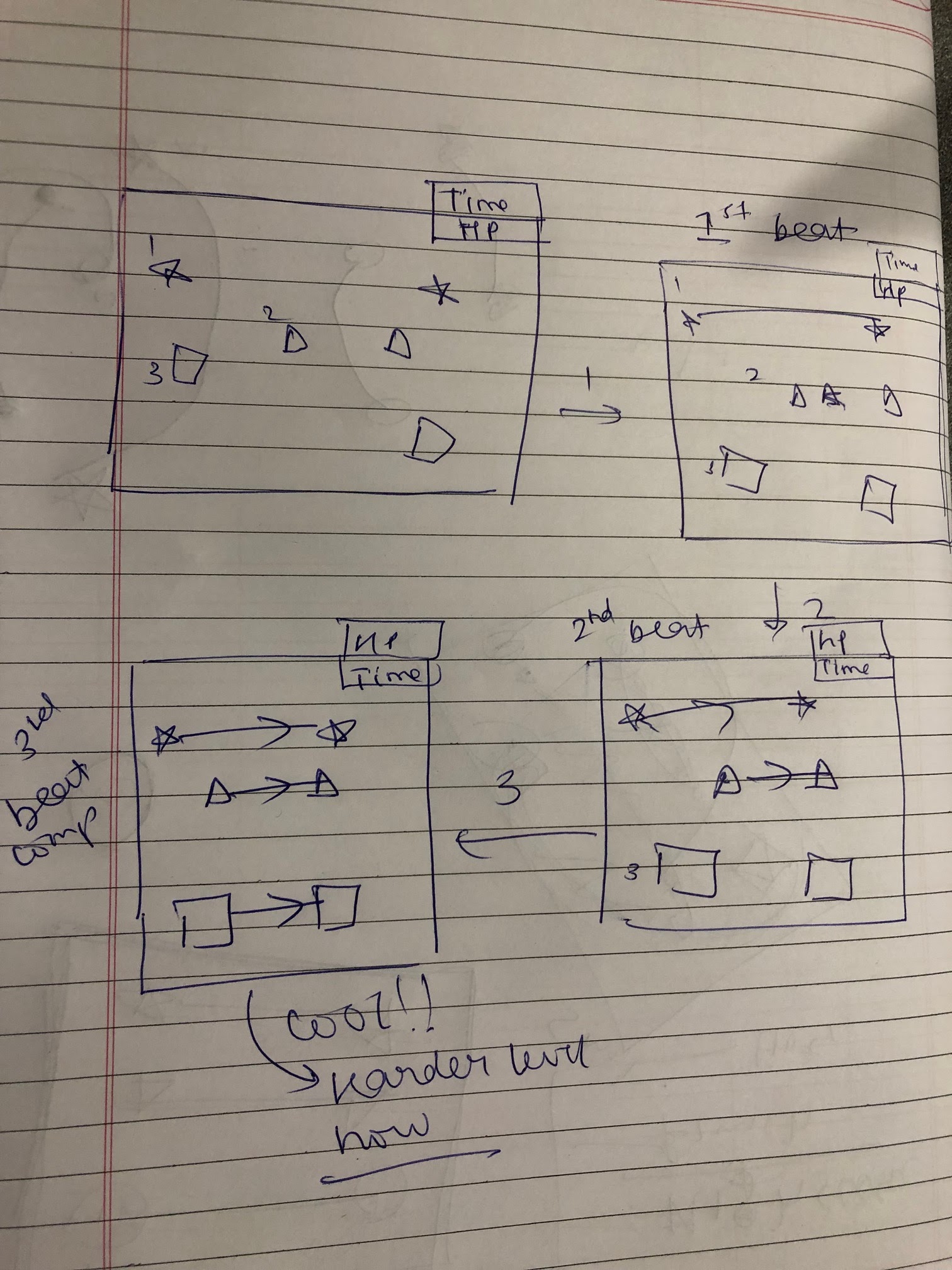
## 1.5 Mechanics (How to Play)

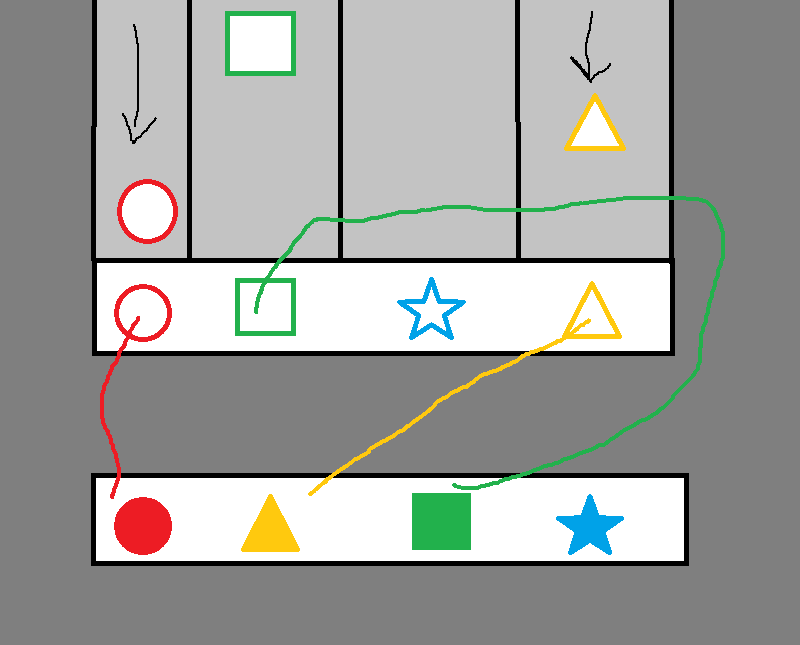
Mouse click + drag, to draw paths.

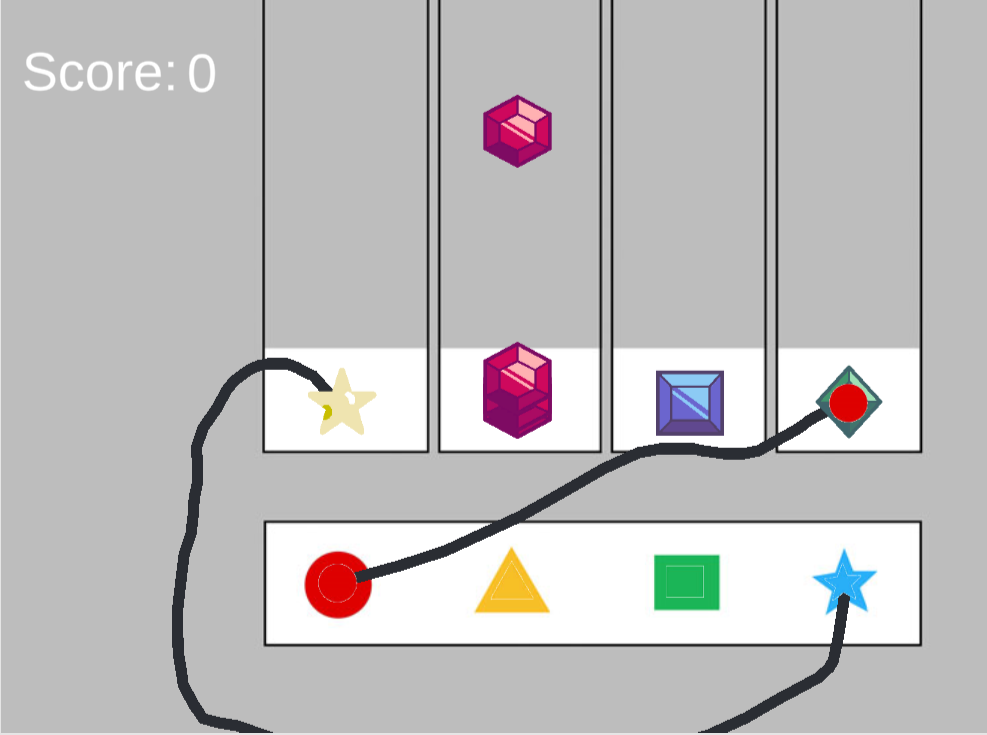
## 1.6 Idea Drawing

Ideation:





Playable Base Mechanic Prototype:



## 1.7 Rewards and Limits Ideation

Currency

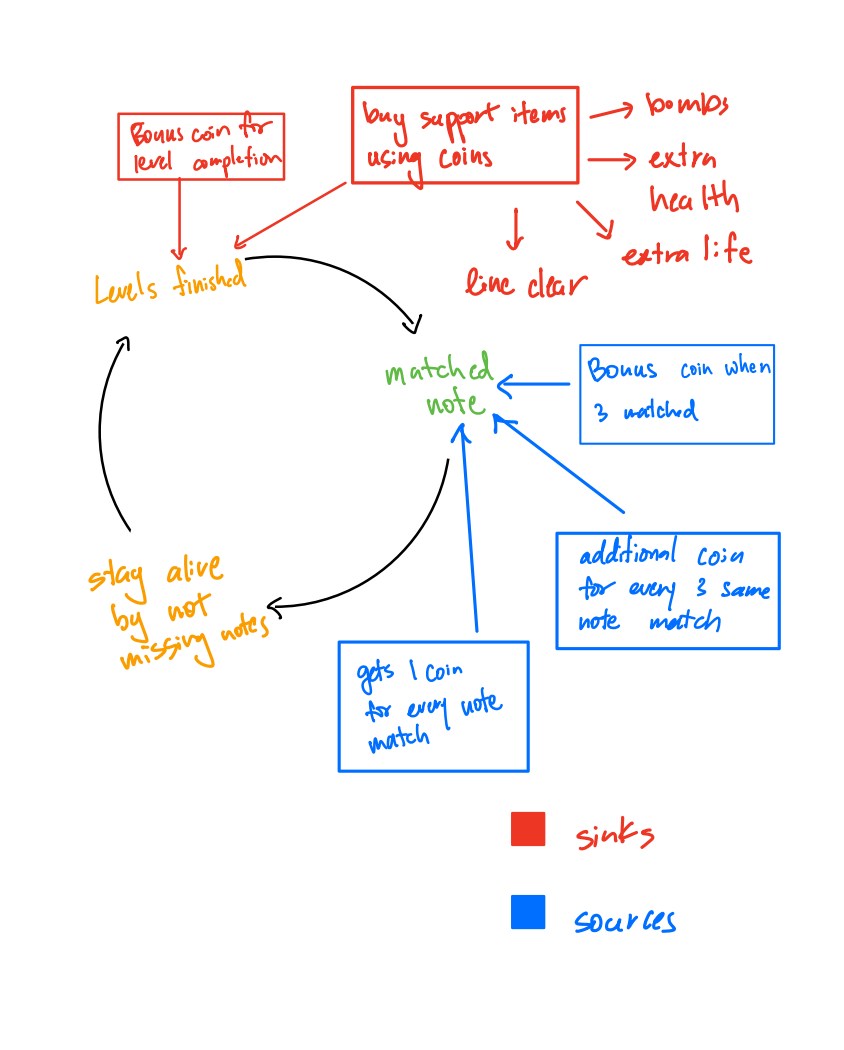
Currency = Points/100 (this is how player gets coins)

Delete incoming gems can be bought with currency

Delete lines can be bought with currency

Restore set amount of health

If you lose you can buy another life for current game (can be implemented later)



Professors notes/ suggestions:

* Improve the understandability of the game by including tutorials (implement single lane/ double lanes.)
* Limit the number of lines by some game play / rewards system such that player gets to clean his/her lines
* Solve the confusion about where to connect the lines with slots (white box area) by including some rewards system such that the user gains more points if he waits for the beat to fall at that specific area.

Sinks:

* Restoring health costs points



Game Economy and Unlocking of levels:

The Game has multiple levels each containing different songs but these levels are locked and can only be unlocked using the game currency of “stars/g\_bucks”.

1. How to get this stars/ g\_bucks?

By solving a level to perfection the user can get a maximum of 5 stars. These stars are based upon the level of completion by the user/ the number of points that user collects in that level.

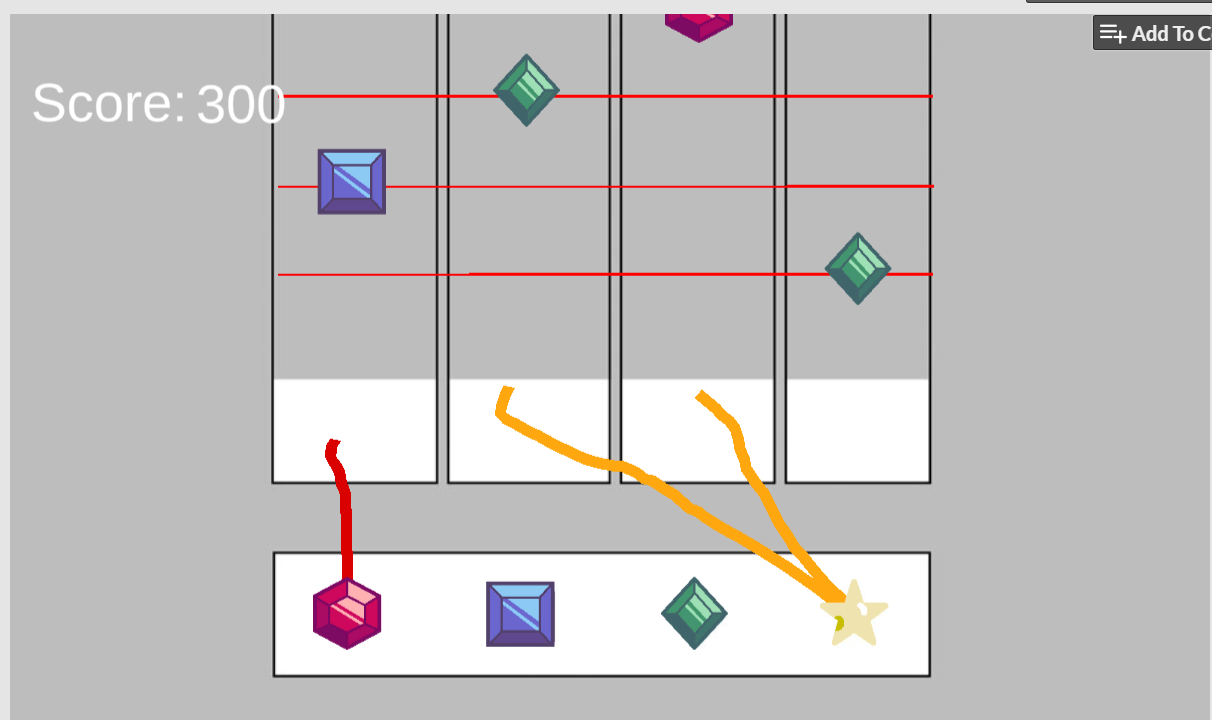
1. Where to use this stars/g\_bucks?

These stars/ g\_bucks are used to unlock new levels. The user gets choice on which level/song they want to unlock as each cost 3 or 4 stars depending on the difficulty of the song/ level. The user can consume g\_bucks to solve all the levels.

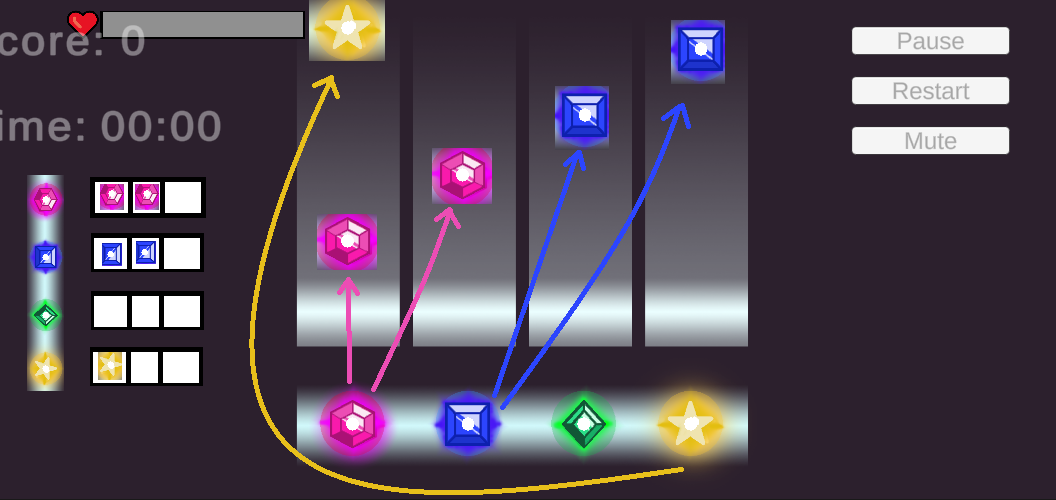
# Development

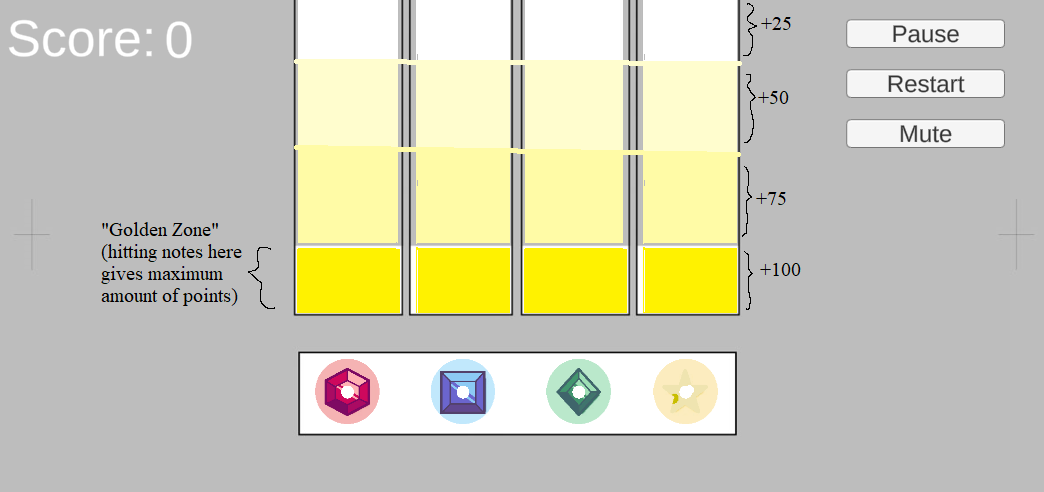
## 2.1 Screenshots and diagrams

Greybox Prototype screenshot:

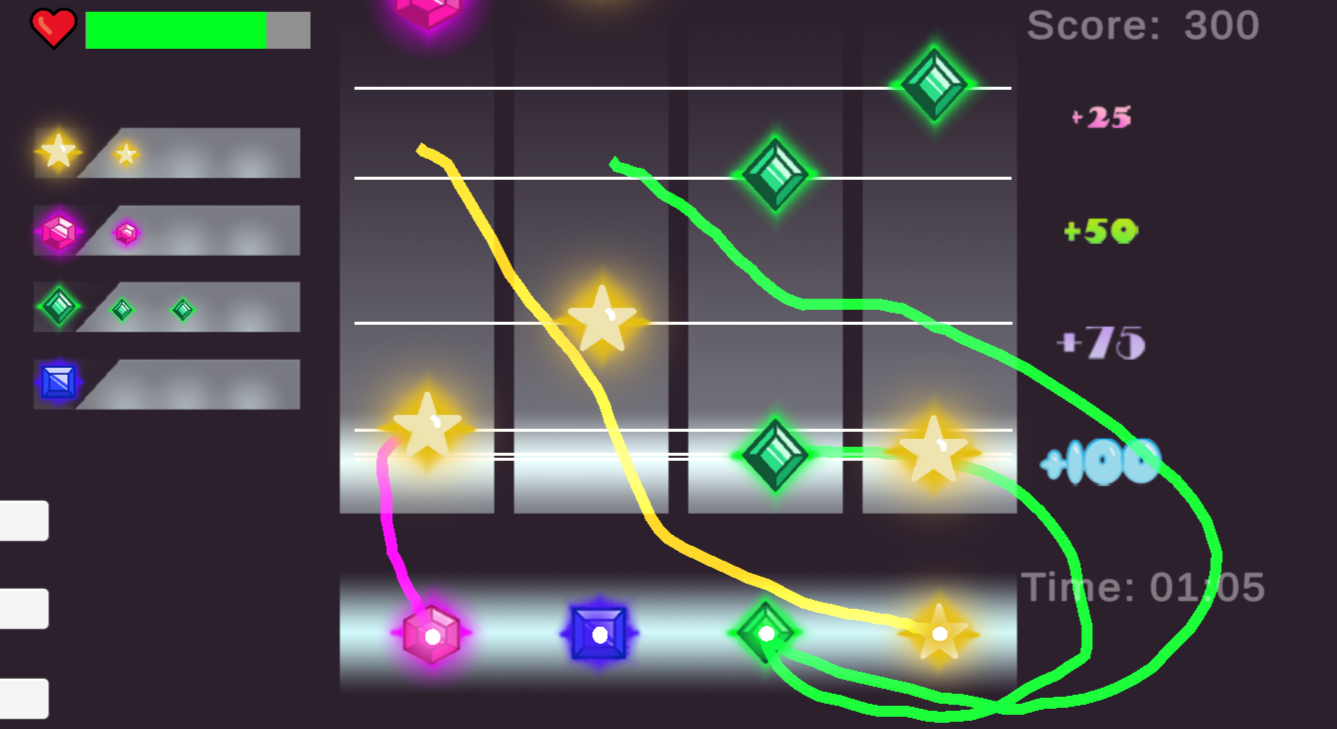


Mockup of match-combo mechanic to clear lines (connecting 3 of the same shapes or 4 different shapes clears all lines):

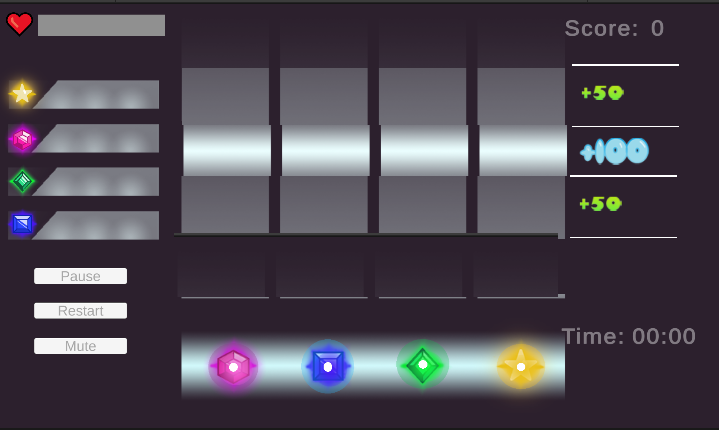


Mockup of scoring system:

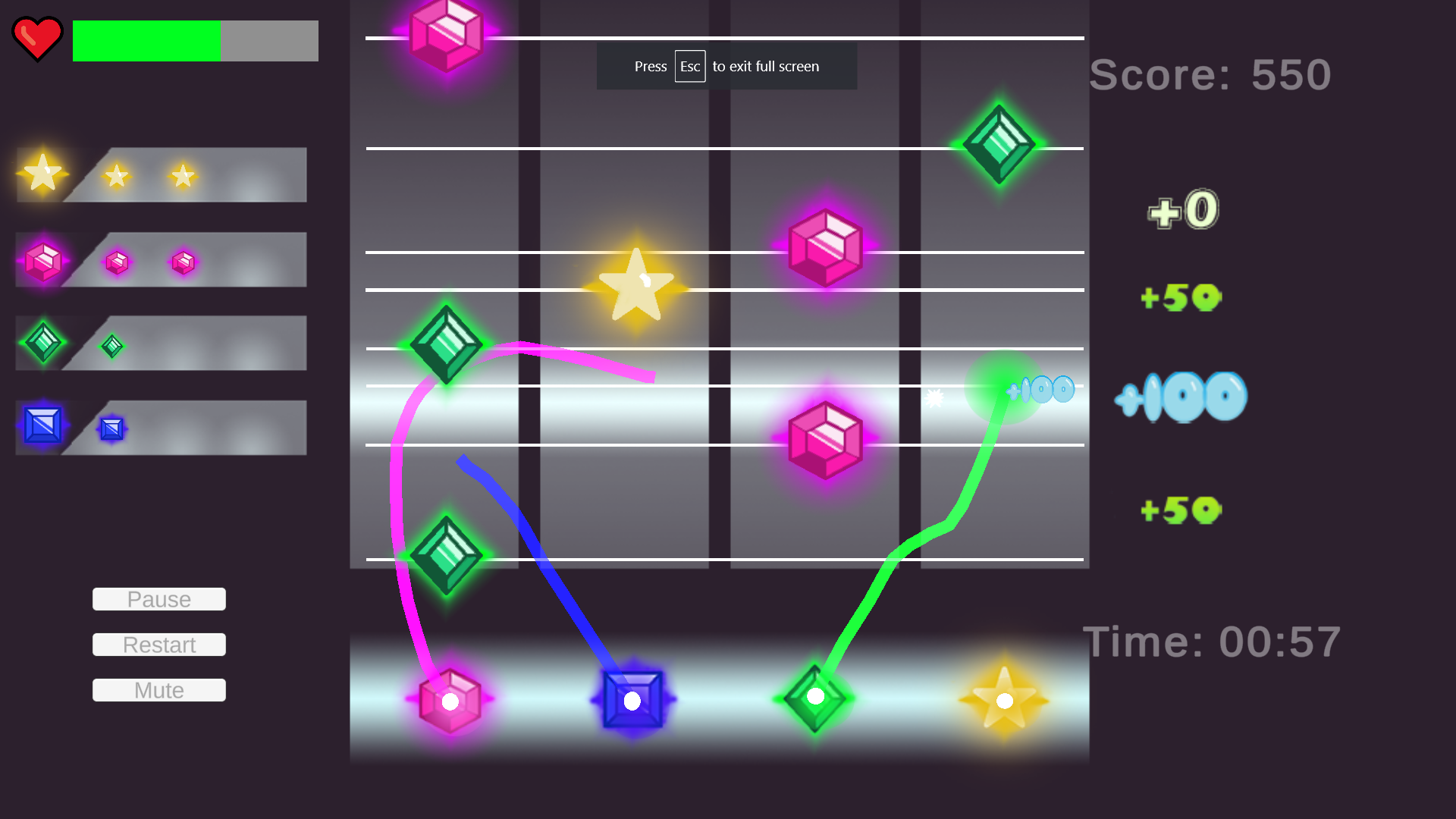
Midterm screenshot:



Mockup of new scoring layout, in response to players hitting notes too early during playtests:



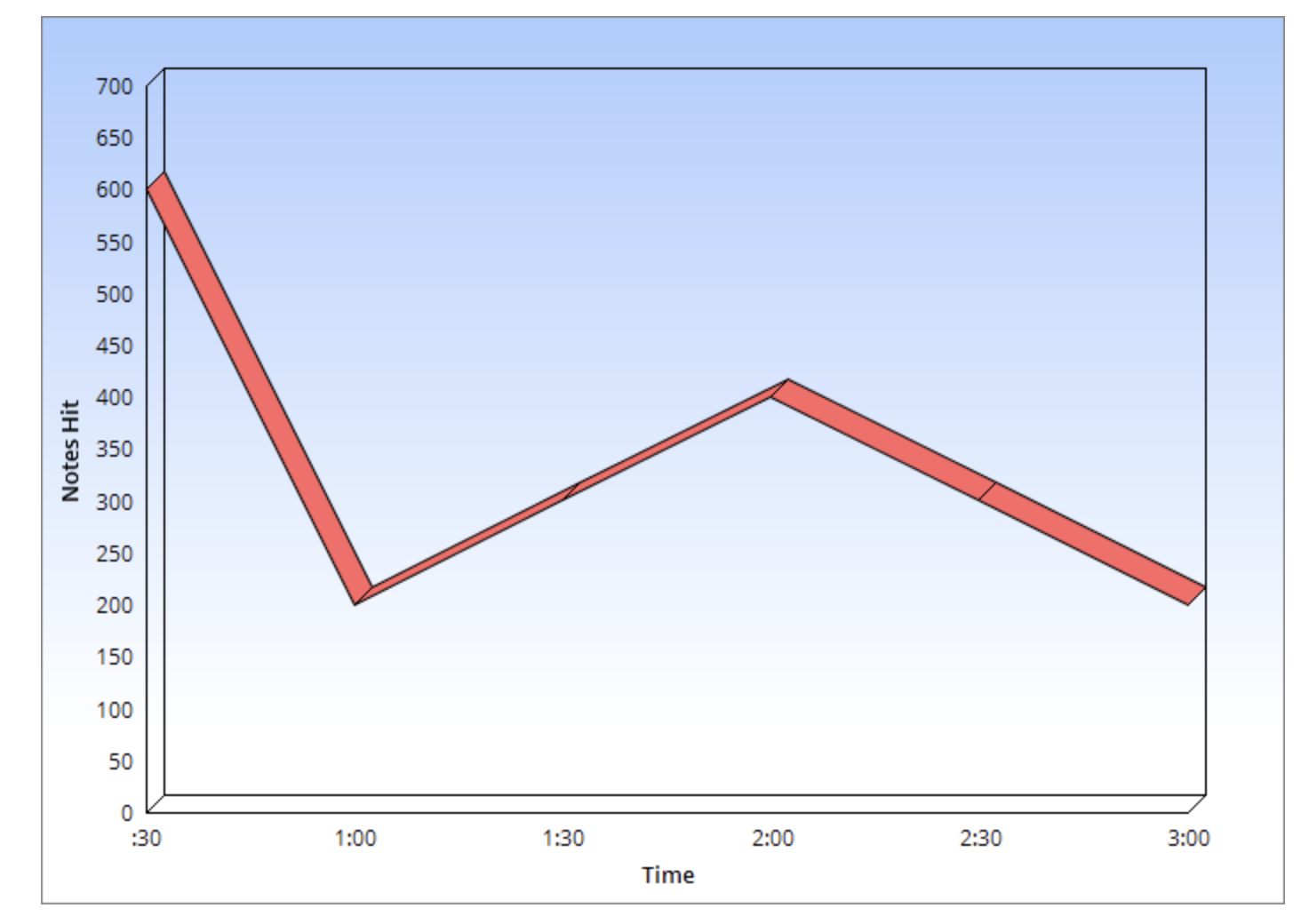
Final gameplay screenshot:

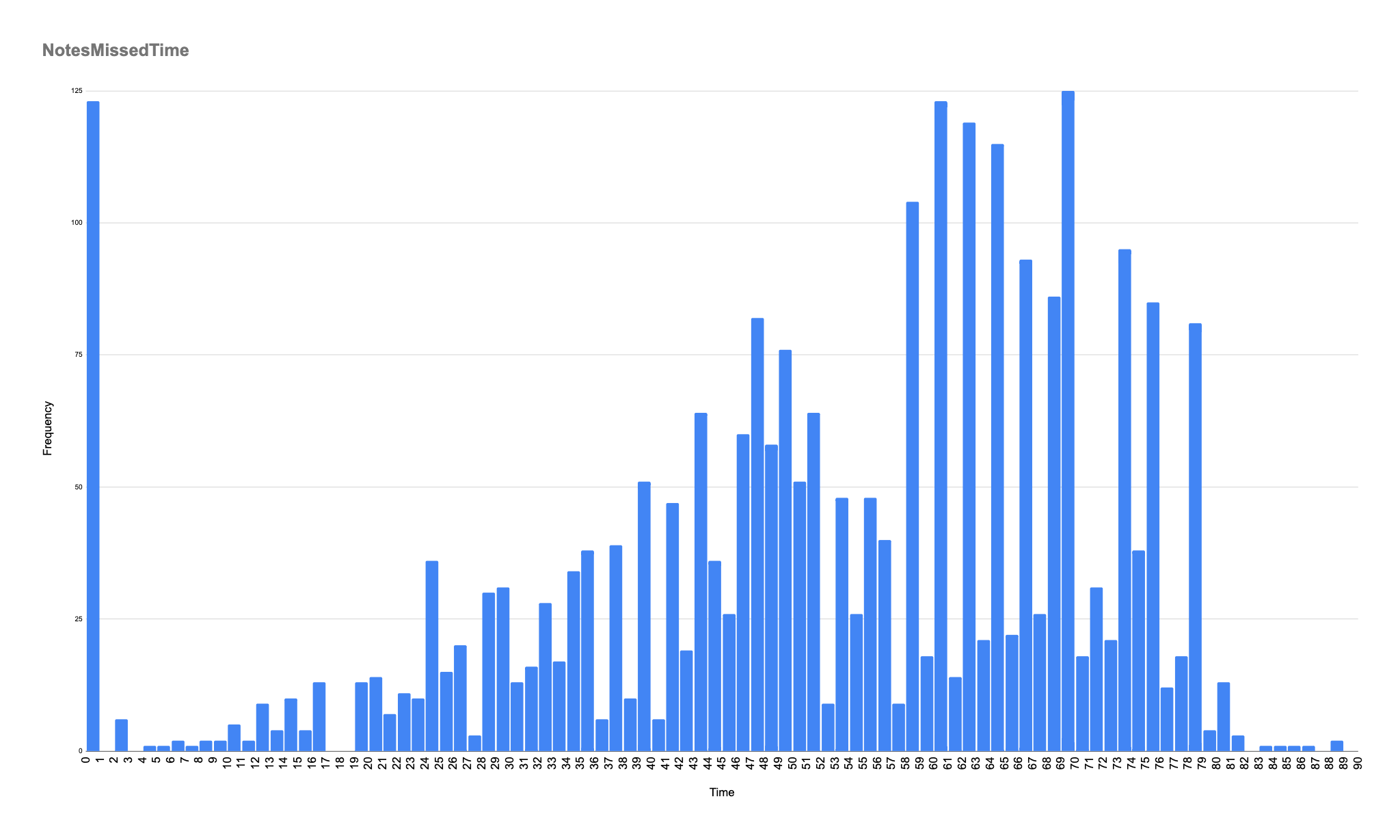


## 2.2 Analytics to Consider:

* record average number of misses over time span

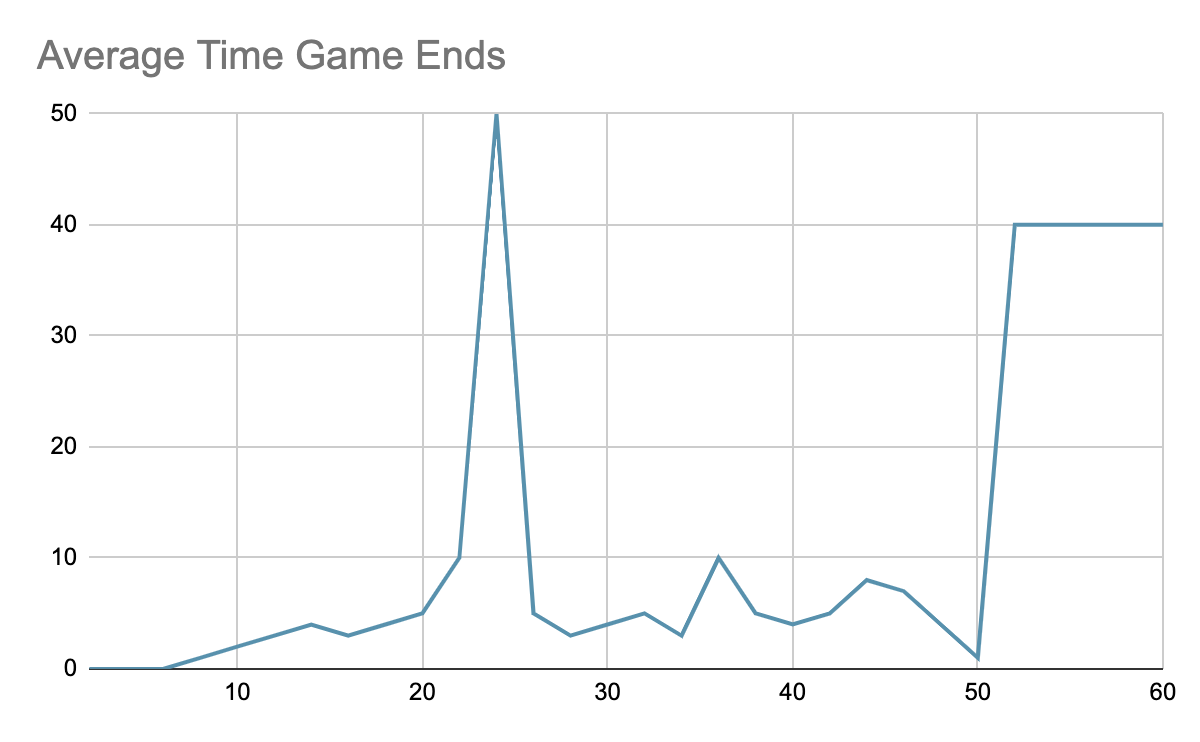
Implemented Notes\_Missed(Time, sum missed), as gradient increase that determines how hard that particular spot in the level is

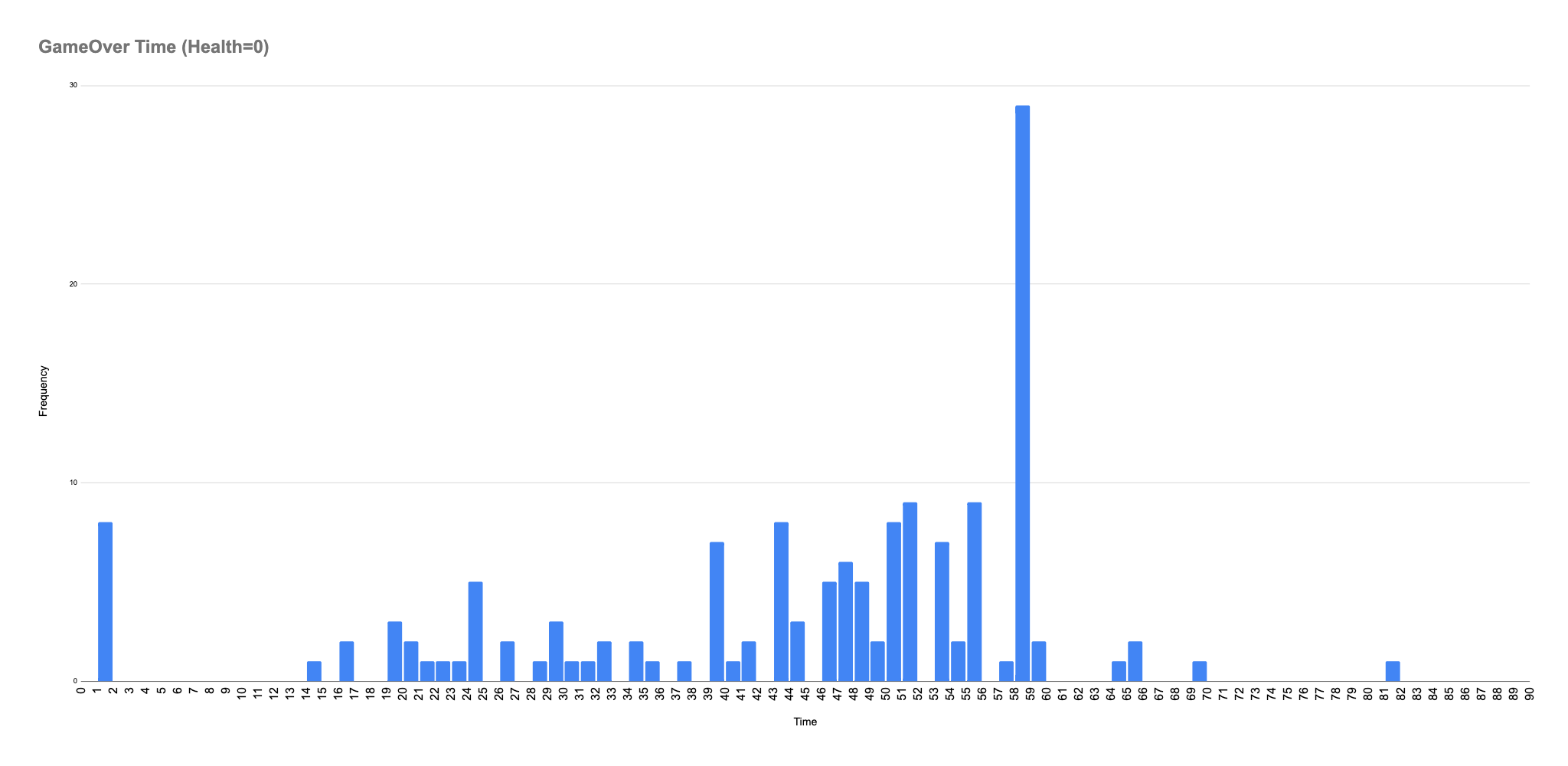
* 



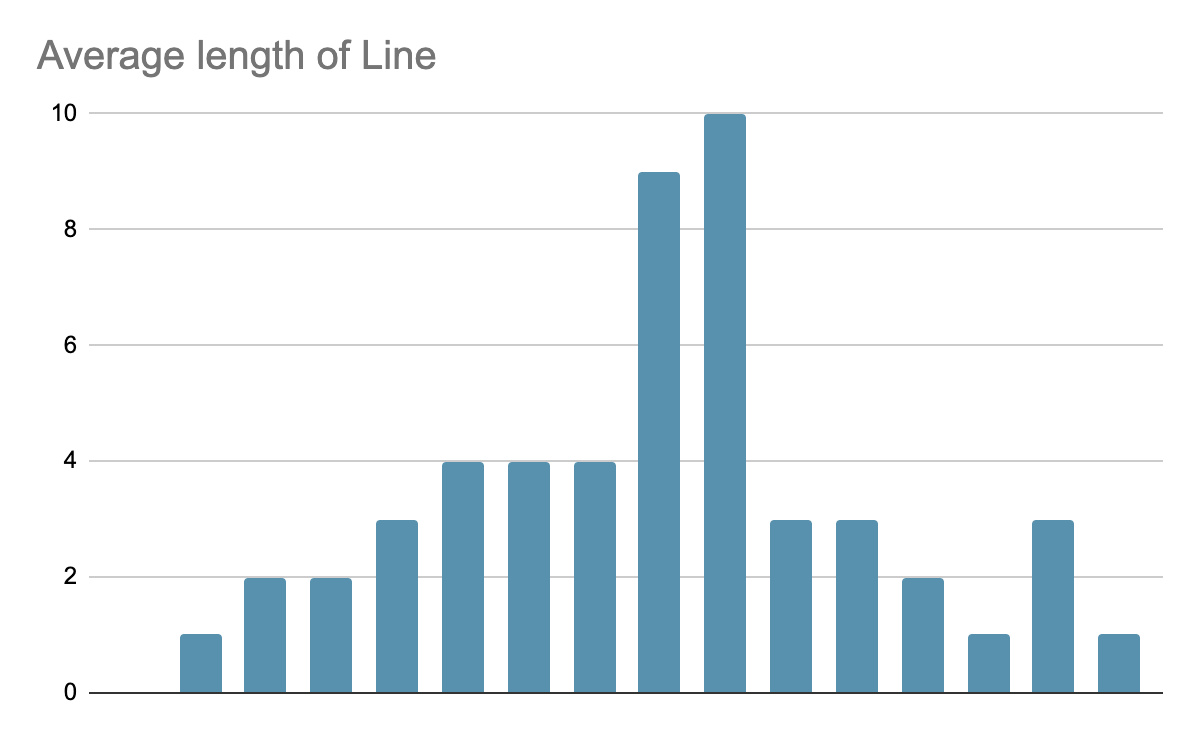
* average time game ends per unit time. (after the song finish the game will end, peak points indicate moments where people all fail/ quit)

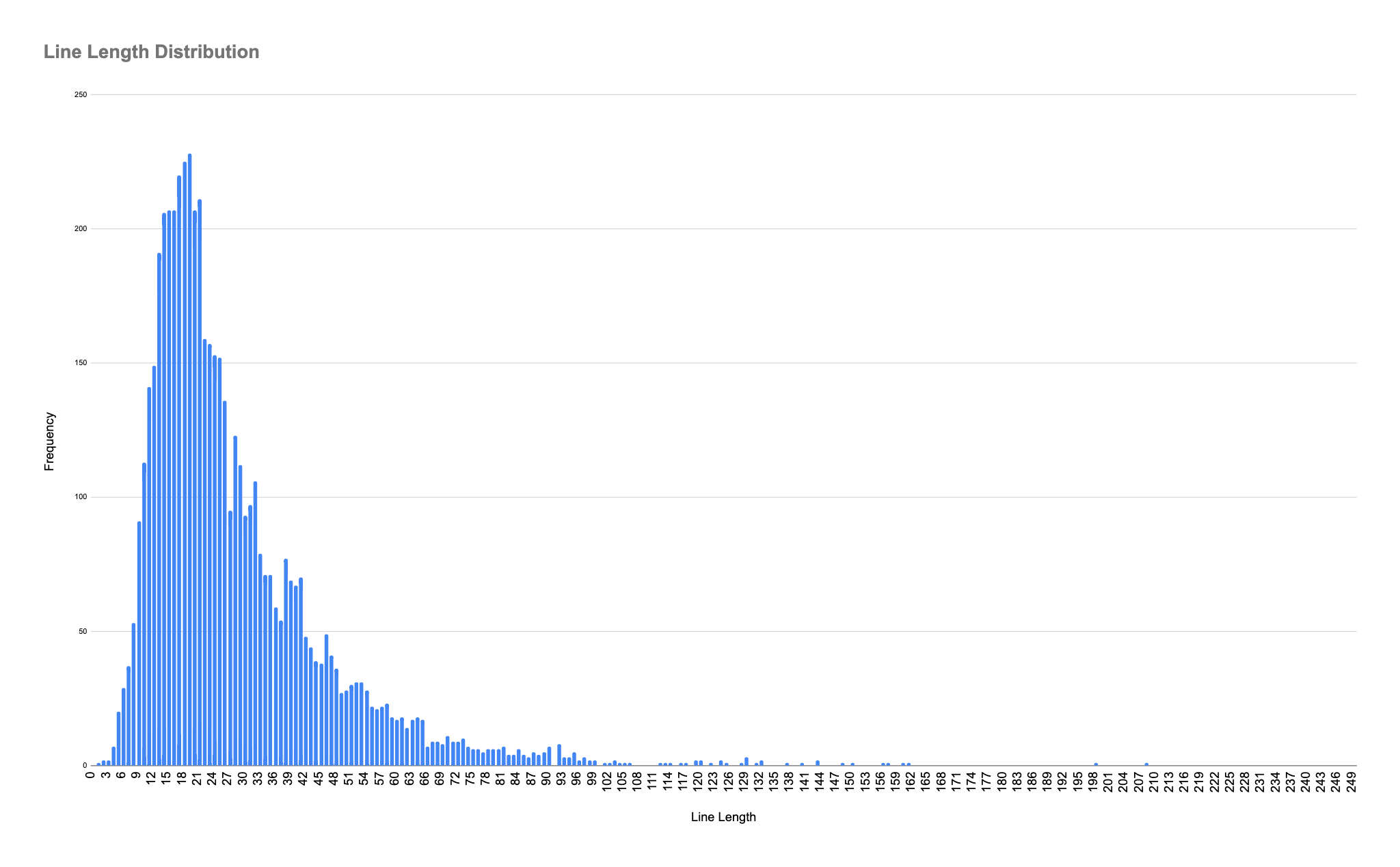
Implemented TimeNoHealth (game ends when health = 0)



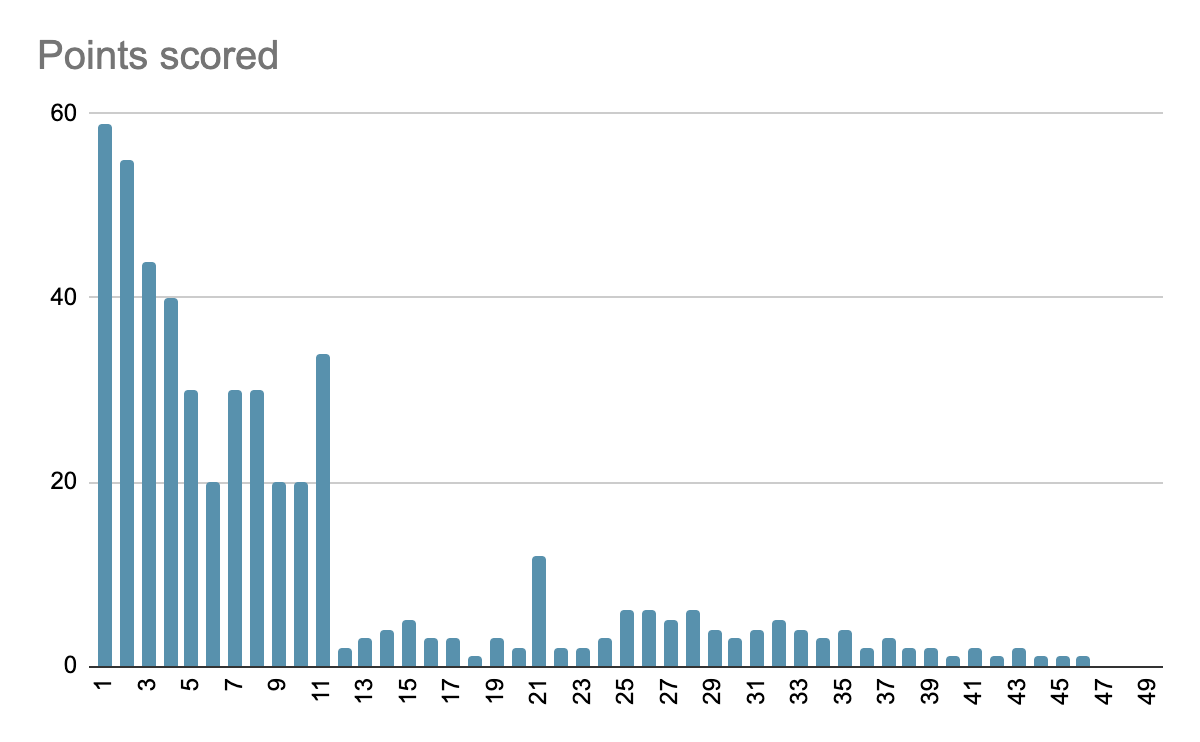
* 
* average length of line per unit length (specified by the distance from shape to another)

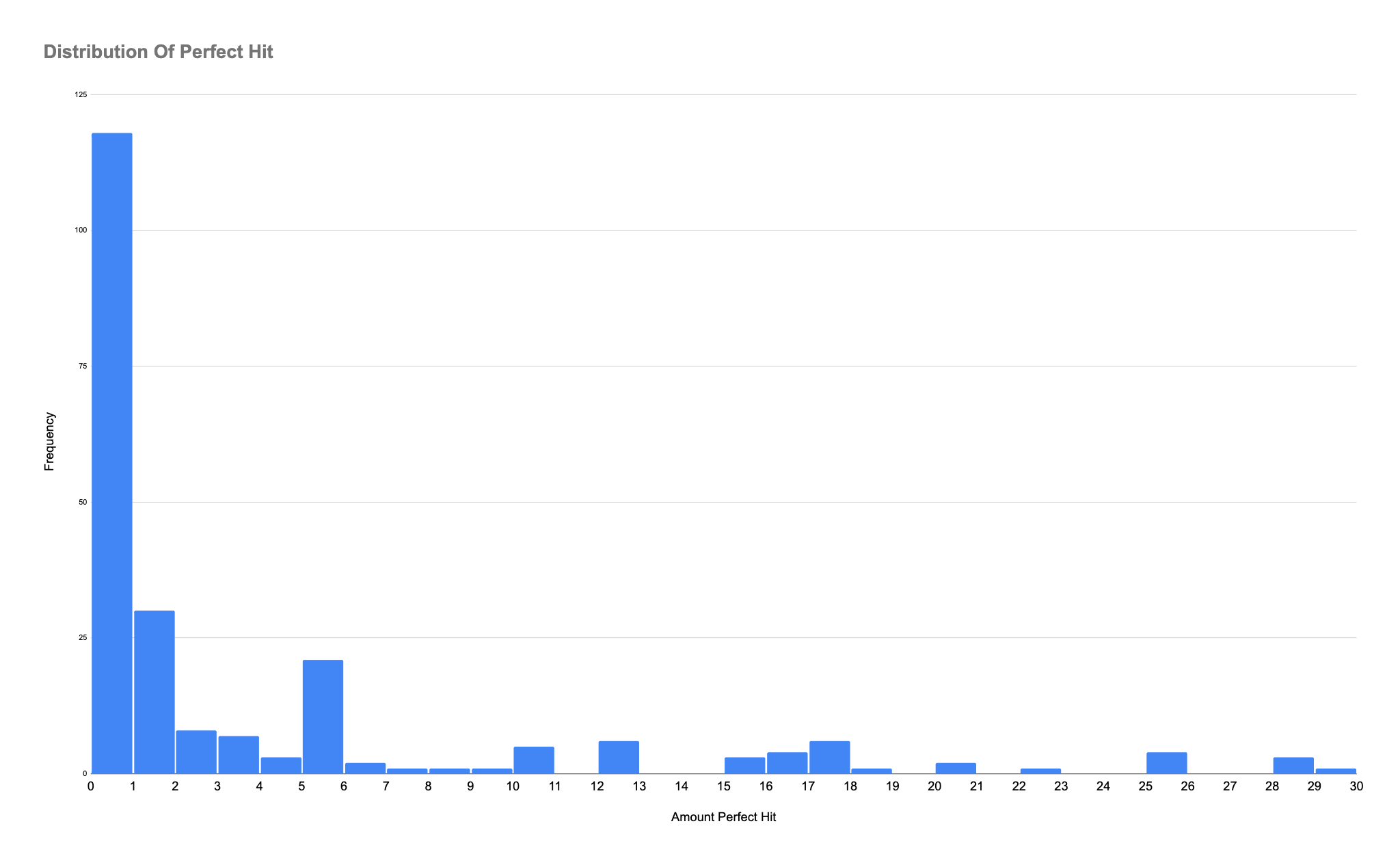
Implemented Line\_Length (unit by length of people drawing or holding down mouse)





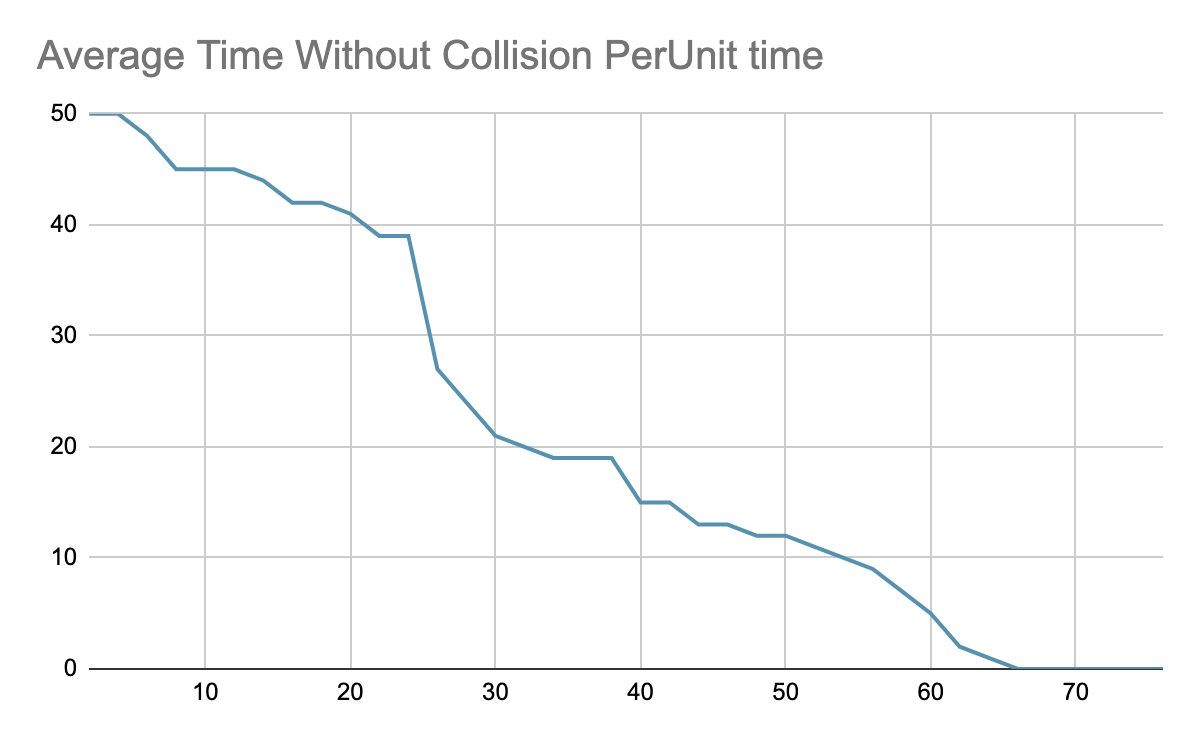
* Amount of Perfect Hit (implemneted in AmountPerfect)

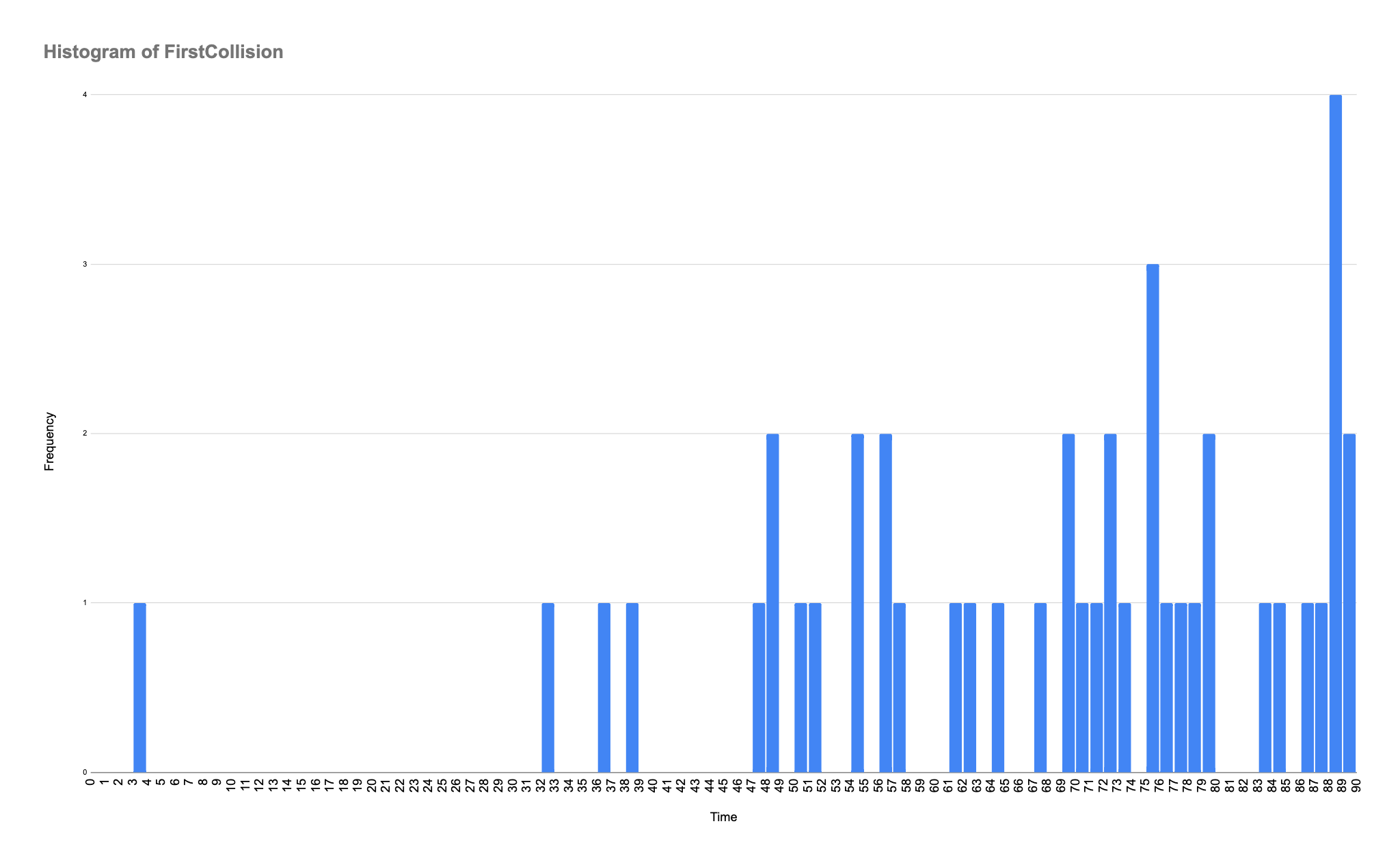




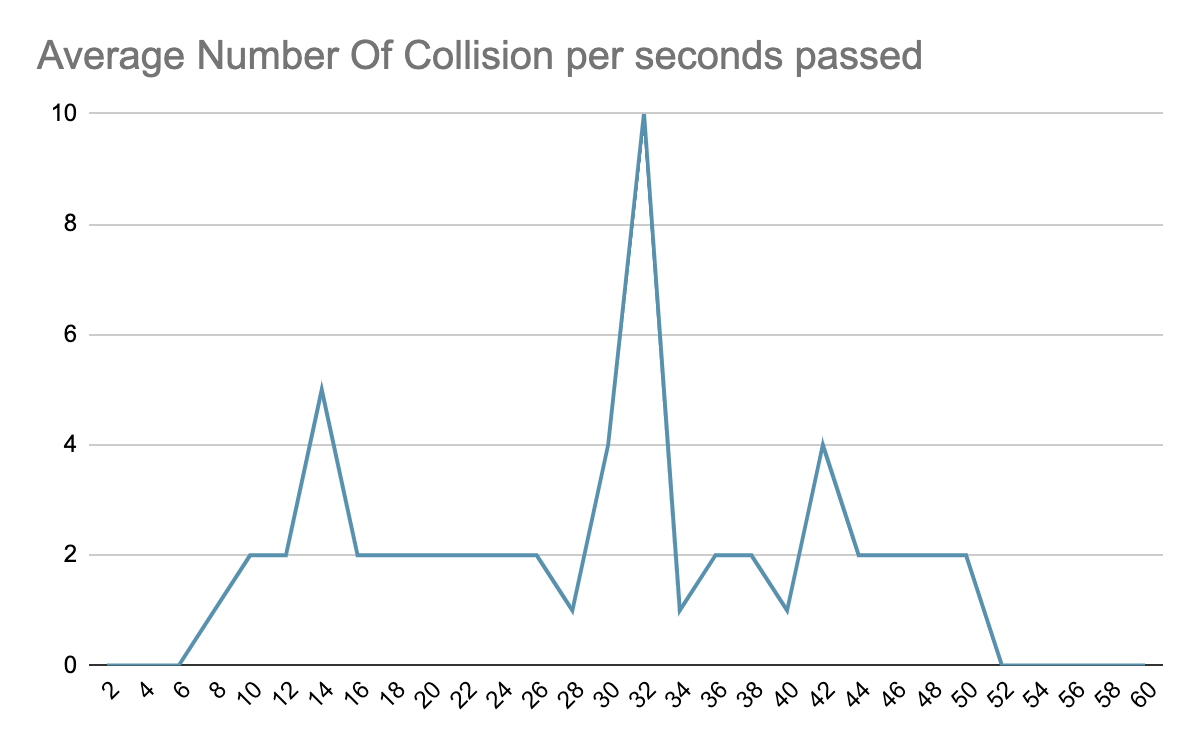
* average time user draws a line without colliding, (As level gets harder, supposedly there should more collision since it should be harder)

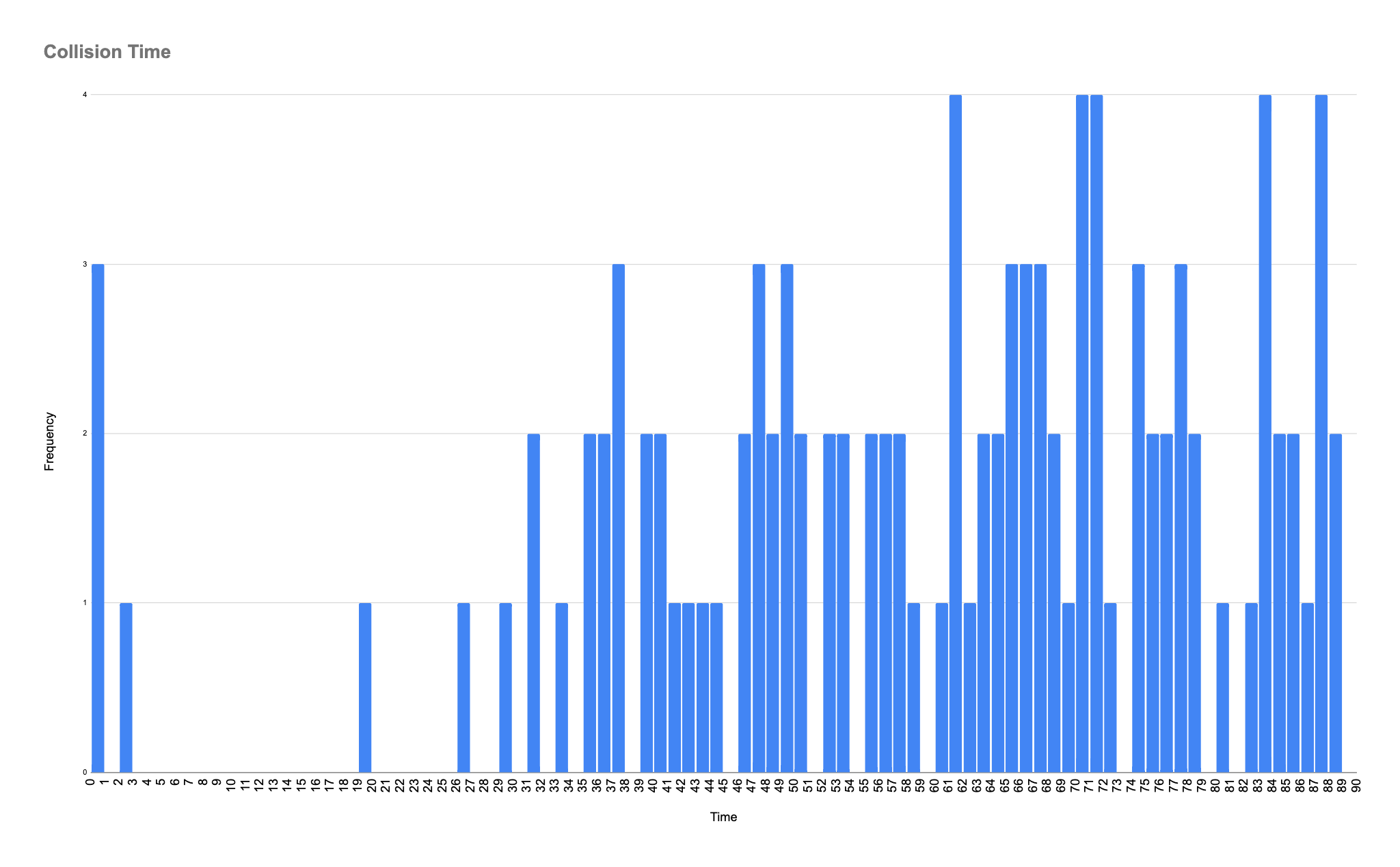
Implemented First Collision





* average amount of collision per time span, (high peaks will indicate at that point the level is too hard, we should drop more bombs)





## 2.3 Analysis of analytics

Feedback we gathered from playtests + how we addressed that feedback: <https://docs.google.com/spreadsheets/d/18HbvWaAj-l95LL1lpjGcKw-VpD5v8gzqjLrtZZ6btqE/edit?usp=sharing>

Further detail on analytics data: <https://drive.google.com/file/d/1toUo0GUcw4XPqN5OJWuy90VN5lF5OHB-/view?usp=sharing>

## 2.4 Final roadmap

Plan to do:

* Place the perfect zone in the middle of the lane instead of the bottom
* Fix line-crossing bug
* Make the endgame screen opaque so that the text is readable and the player is not still able to draw lines after game over
* fix restart button in tutorial
* update visuals for where to hit notes in tutorial2
* add clap-count sound to beginning of level
* from the main menu, when the player presses Start Game, they go to the song selection menu
* Fix timer to match song length
* Fix buttons being offscreen in tutorial3 and MainScene
* Add post-level screen showing the number of notes perfect, good, etc.

What will not make it into the final:

* Unlocking new song levels with points earned from playing
* Powerups that cost points