Hogwarts Hobo Game
Product Backlog

Original Requirement	Story	Initial Priority	Engineering Task	Effort Estimate/Risk Estimate	Sprint Details
Score Counter/ Timer	Keep count of the number of seconds that have passed in the game.	2	Have a counter running throughout the runtime of a game session	Expected: 2 hours +/- 1 hour Actual: 3 hours	Created an object GameTimer to keep track of time the player is alive in the game. The GameTimer object gets the time it was created. Another method from this class is called continuously for each frame of the game, getting the time it was called. This latter time is subtracted from the former, the difference being the time player has been alive. This is then displayed continuously.
Train	Object for displaying trains	3	Implement an object to store the train's attributes and methods	Expected: 2 hours +/- 1 hour Actual: 3 hours	Constructor made to set elements of the train Function made to create Trains offscreen Coordinate system then used in another

					function to move trains across the screen Impact detected by continuously comparing train and hobos coordinates for overlap
Track	Interleave oncoming trains on each track	3	Have two train objects spawn within an interleaving fashion	Expected:3 hours +/- 1 hour Actual: 4	Created Track Class and object. Track calls all the drawing functions of the objects that will be on it: Trains, Planes and the hobo Tracks move the train across the game screen and remove them from the game after they pass the player-visible area Random generator for planes created with the probability of 11/10000, if there are no planes existing.
Magical Hobo paper plane hints	Give players hint of oncoming trains, usually to no avail	2	Have an object send delayed message about arriving trains	Expected: 4 hours +/- 1 hour Actual: 5 Hours	Created Plane Class and object. Function has an array that holds 1 plane and its current location and another function constantly draws it until it is picked up. Created a detection function to detect when the hobo picks

					up the paper plane object. Created the drawing function to display the plane onto to game screen When the plane is picked up a green flash is shown to indicate which tracks the next few trains will come from.
Health Bar	Keeps track of the player's health.	2	Implement an object to keep track of the number of times the player has been hit.	Expected: 2 hours +/- 1 hour Actual: 3 hours	Displays a bar that decreases each time the hobo is hit by a train and informs the player of the health they have left. Implemented a method for the game over screen that displays when the size of the health bar reaches zero.
Hobo Character	Displays the player character	3	Implement an object with the hobo's attributes and methods for display and movement	Expected: 1 hour +/- 30 minutes Actual: 2 hours	Implemented an object with attributes for position, speed and lives left. Implemented two instance methods: one to display the Hobo (draw) and one to move it (move).

March 30, 2020

Test Plan

Test	Result1	Result2	Result3
Trains			
Do trains move in the same direction?	yes	yes	yes
Do trains move at an appropriate rate?	no	yes	yes
Do trains spawn at a normal rate?	no	yes	yes
Are length of trains reasonable?	yes	no	yes
Do trains visually take up the entire track?	yes	yes	yes
Does the train hitbox cover the entire train?	no	yes	yes
Is there enough space around train clusters?	yes	no	yes
Health Bar			
Does the bar deplete with each impact?	yes	yes	yes
Are the features of the bar differentiable from the trains?	no	yes	yes
Does the game end when health = 0?	no	no	yes

Does the length of the bar stay within its boundaries?	yes	yes	yes
Is there a screen displayed to notify the player that the game has ended?	no	yes	yes
Timer			
Does the Timer display as intended?	yes	yes	yes
Does the Timer count up as intended?	yes	yes	yes
Is the final time displayed after the game ends?	yes	yes	yes
Hobo			
Does the hobo display?	yes	yes	yes
Does the hobo move in the correct direction when the appropriate key is pressed?	yes	yes	yes
Does the player lose control of the hobo once the game finishes?	yes	yes	yes
Planes			
Do plane indicators show up?	yes	yes	yes
Does plane randomly generate correctly	no	yes	yes
Is the indicator interactable?	no	no	yes
Does the information	no	yes	yes

March 30, 2020

appear correctly? (green flash)			
Does the plane spawn in a reachable area?	yes	no	yes
Correct spawn frequency as desired?	yes	yes	yes
Is the information expected?	yes	yes	yes

Results Report

Tests	Durations of test	Pass Rate
Train		
Trains moved left to right	20 seconds	100%
Trains have consistent speed that can be dodged	40 seconds	100%
Game has sufficient amount of trains for proper difficulty	2 minutes	90%
Random train lengths do not become too long	1 minute	95%
No gaps in between trains on adjacent tracks	30 seconds	100%
Player takes damage from all parts of the trains	40 seconds	100%
Able to move in between the spaces between the trains?	2 minutes	95%
Health Bar		
Depletes with each impact	30 seconds	100%
Distinctive, identifiable features	5 seconds	100%

March 30, 2020

Displays game over screen when Health = 0	10 seconds	100%
Bar stays within expected boundaries	20 seconds	100%
Timer		
Timer is visible throughout gameplay	20 seconds	100%
Timer counts up for duration of game	30 seconds	100%
Final time displayed after end of game	40 seconds	100%
Hobo		
Displays	5 seconds	100%
Movement: left, right, up, down	~15 seconds	100%
Planes		
Visible at all times?	1 min of playtime	100%
Randomly generates	5 mins of playtime	100%
Is the plane always interactable	5 mins of playtime	95%
Information displays correctly	5 mins of playtime	100%
Plan always spawns in a reachable area	5 mins of playtime	100%
Plane spawn rate is as desired?	5 mins of playtime	100%
Information as expected	5 mins of playtime	95%