Project Description:
Title — Mechanics Practice Tool
The general gist of this project is a game where a character can move along a map and has to avoid projectiles. Since the game is based on League of Legends, the two "maps" available will be areas on the Summoner's Rift. The three characters available will be Teemo, Ahri, and Malphite. These characters will vary in size, health, and speed. The projectiles that will be shooting at the player include Fizz ult and Ashe ult. There will also be a scoreboard for highest scores, which will be the max amount of time that a player has survived on either map with any character. Once the character's HP drops to 0, there will be a game over screen where players are either able to try again or are redirected to the main menu.

Similar Projects:
There is a website called LOLDodgeGame.com that features a wide variety of games relating to League of Legends mechanics. One of these games is the Dodge Game which is similar to the project I would like to create: it also features a character but the map does not have any obstacles or areas where the character cannot move and the player has to move the character around to dodge the projectiles. However, my project will differ because the projectiles I plan to implement are different and I am also incorporating character health as well as different characters and a potential stun.

Structural Plan:
Projectile Class: will hold projectile size, projectile type, projectile effect, end location
Character Class: will hold a character's speed, health, size, any debuffs
Map Class: will hold the two different maps as well as the areas on the map the character cannot pass through
Score Class: holds the scores of different players

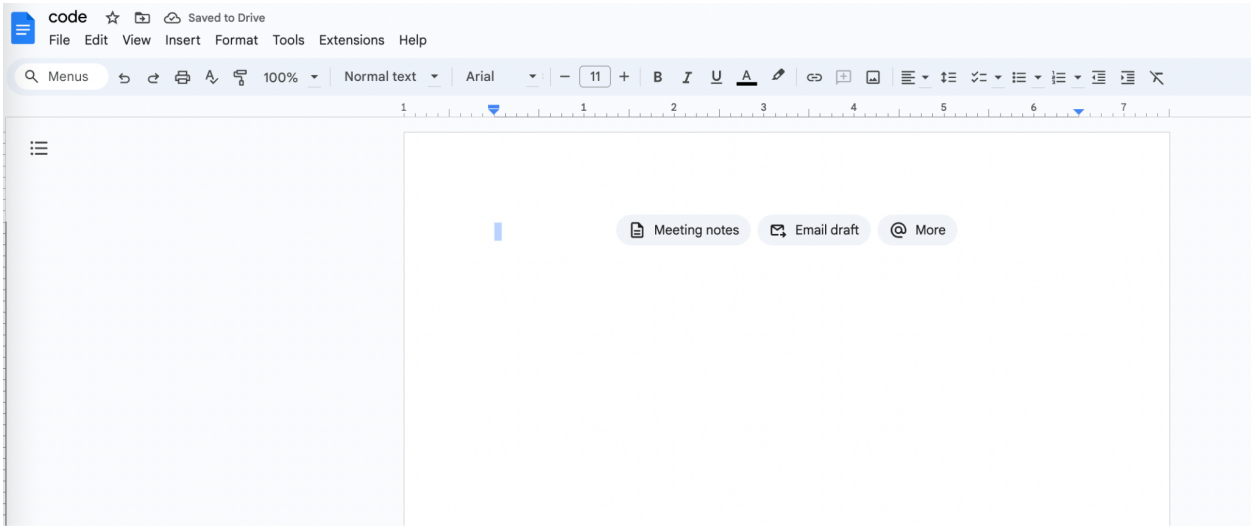
Collision Functions: will check if projectile and character collide and create the impact; will check if character and map collide; will check if projectile and map collide
Score Handling Functions: will track the time that has passed
Projectile Creations Functions: will randomly choose between the projectiles as well as locations on the map; send out the projectiles and hold the projectiles (list maybe)
Movement Functions: will move the character chosen through the mouse
Render Functions: displays the visuals of the game (what appears on screen)

Algorithmic Plan:
I think the most difficult part of the plan is going to be the projectile interactions with the map and character, since for one of the projectiles, it will not travel the whole map, stop somewhere at the map, and then do damage to the area + airborne the character if it is there. I think this will be

the most difficult portion to implement but if I think that by splitting it into smaller parts, such as the projectile + character interaction vs projectile + map, I can make things easier. Additionally, I can create variables to hold whether or not the character is stunned vs airborne or if there are any debuffs as a part of the Character Class. Additionally, by using isLegal() checkers to make sure that nothing is going out of bounds, I should be able to successfully incorporate the different kinds of projectiles into my map.

Timeline Plan:
TP 1: create and code all three classes (projectiles, maps, characters)
TP 2: create the interactions between projectiles, maps, and characters, draw functions, movement functions, collision functions
TP 3: create main menu and game over screen, character + map selection screen, and polish the interactions up

Version Control Plan: I will paste it into a Google Doc which is accessible across all my devices and saved to my school account. With every few lines of new code, I will update the doc.



Module List: n/a

Storyboard:


MECHANICS
PRACTICE TOOL
Start
Score board
Character Selection
teemo (small)
description of champion
ahri (medium)
description of champion
malphite (large)
description of champ
Map selection
mid lane
description of map
bot lane alcove
description of map
Healthbar
Score
xxx
Healthbar
Score
xxx
GAME OVER
You survived xxx time.
Try Again
Main Menu