Prototype 12 Design Brief

A perspective-based 3d endless runner where you need to use your wands’ powers to traverse the landscape.

Inheritance

* Object: wands! Each should have a different particle effect. (Let’s build off prototype 8.)
* Object: obstacles! Different types of obstacles should react differently to the wand effects.
* Object: course pieces! Different course pieces should have different tracks.

Polymorphism

* Object: Wands: all wands should have a Cast() option and a different Recharge() coroutine, but different effects for each Cast().
* Object: Obstacles should have different OnTriggerEnter() options for the player (but that’s already accounted for), but also different Enchanted(), Destroyed(), or Diverted() (or whatever) options for each obstacle.
* Object: Course Piece. Each one should have different Track() functionality, whether it’s vertical or horizontal.

Encapsulation

* Wand
  + Name, can get but not set
  + Position on track, can get or set
  + Projectile effect (particle effect with collider?)
* Obstacle
  + Name, can get but not set
  + Position, can get or set
  + Speed coming at player, can get or set
* Course Piece
  + Begin/End Left/Right Horiz Pos, can get.
  + Begin/End Left/Right Vert Pos, can get
  + Some sort of is-grounded for an X position

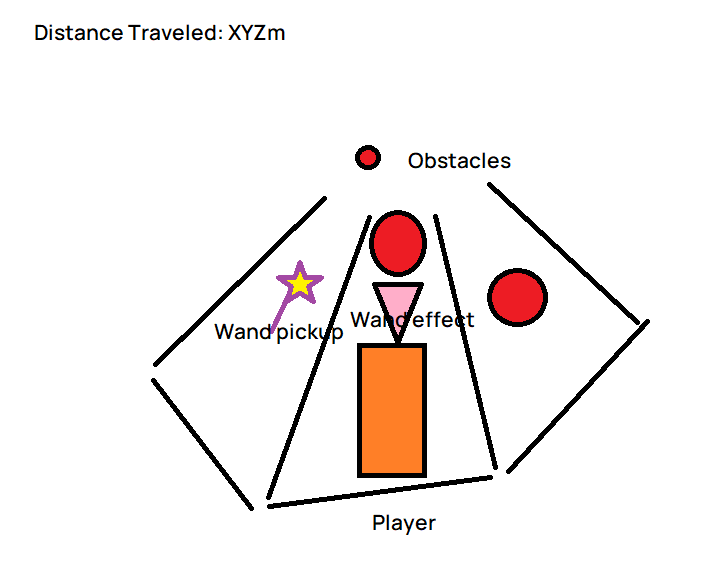
Abstraction

* SpawnObject (for wands, obstacles, and pieces)
* MoveObject (for wands, obstacles, and pieces)

Shows that inheritance should be something like

CourseObject -> Wand/Obstacle/CoursePiece -> Each individual piece

Course pieces should spawn to some X limit, and continue to spawn. They can be held in an array. Similarly, wands and obstacles should also spawn in this way.

While moving the player with actual physics is a cool idea, we should actually continue to move the player forward so that the skybox effect looks right.

**Tasks to complete**

* Set up Main
* Set up the player and PlayerController in Main
* Set up player animations in Main since that created a slew of problems last time
* Set up track generation
* Set up obstacle generation and interaction
* Set up powerup generation and interaction (including wand animations)
* Set up Sfx, Bgm, Pfx
* Polish Game Logic and make it interact with Title
* Playtest and polish
* Optimize
* Publish and ensure it works correctly in WebGL