Project Plan - Sonic

Game type: 2d Platformer
For reference
https://www.youtube.com/watch?v=9kOAdhUlkt0

We intend to create a replica of an original sonic game. This will be a side scrolling platformer. In this game you will play either of two characters either sonic or knuckles. We will be focusing on creating one level for the minimal deliverable and aim to add more. The characters will have the ability to defeat multiple enemies in their path and have the ability to make it throughout the entire level.

Entities

Most of the sprites for this game will be downloaded from the internet.

- Sonic - the main character of the game and his states will be the properties as our other character (Knuckles). They will both have the same state. And they will have multiple actions (idle, running, spinning, jumping, and death).
- Crab - This will be one of the primary enemies. It has two states alive or dead and 1 action of walking.
- Bee - This will be another primary enemy. It will have two states alive and dead and 1 action of flying.
- Bricks, Invincible boxes - to map out the area in which sonic will run on, also if sonic jumps on an invincible box he will have a 50/50 probability to either allow him to not be affected by any enemy for 15 seconds or to gain more rings. (Temporary Ability to be Invincible - An Extra)

- Rings - They are a pickup item that sonic can get as he goes throughout the level. It will one have one animation.
- Hills, Clouds, Bushes - We will use these entities in order to decorate our level.
- Scores - To keep track of points
- Time - To countdown the amount of time the player has to finish the game.

Controls:

We will be using a keyboard to keep track of the controls of this game. We will be using two letter keys, them being the x and y keyboard keys.

- Directional Pad - We will use the Arrow Keys to simulate a directional control pad: left and right. Left and right will accelerate Sonic in those directions up to a maximum speed.
- “X” Button - The X Button will be used to jump. If it is held while Sonic is moving upwards it will slow Sonic’s deceleration.
- “Z” Button - The Z Button will be used to cause Sonic to Spin. If it is held while sonic is moving upwards it will slow Sonic’s deceleration.

Game Interactions

Most of the interactions of this Sonic game will be collision based, however we will also have state based interactions as well relating to the rings that sonic can pick up.

- Rectangle based collisions - collisions in Sonic will use axis aligned bounding boxes. Bounding boxes will be applied to all interactable entities. In most cases these will be identical with the entity's dimensions.
- Sonic’s Interactions - Sonic will own any interaction that he is involved in regardless of which entity is affected by the interaction. That is, Sonic will handle interactions with enemies regardless of who ends up dying.
- Sonic will land on platforms if he is falling upon them from above.
- Sonic will have his velocity reduced to 0 when an object.
- Sonic will die if he interacts with a crab unless if he lands on that crab, which would cause Sonic to not die and for that specific crab to die and for Sonic to jump.
- Sonic will die if he interacts with a bee unless if he lands on that bee, which would cause Sonic to not die and for that specific bee to die and for Sonic to jump.
- Sonic will change state throughout the game as long as he continues to interact with rings in which a ring will grant Sonic a single ring.
- Enemies - These Enemies will be interacting with the environment.
  - If they hit a “wall” the enemies will reverse direction.
  - If they reach the end of a platform they fall or reverse direction.
  - Enemies will bounce off one another.

Timeline

Week 3

- GitHub Repository established.

- Start drawing sprites
- Start animating all sprites
- Running
- Jumping
- Spinning
- Left and Right control movement

Week 4 (Prototype)

- Implement a scene manager and activate the background scrolling.
- Start constructing the first level's layout.
- Complete the drawing process for each entity, then add them to the prototype.

Week 5

- Continue updating entity information and drawing stubs.
- All entities should have bounding boxes.
- Add platform landing, gravity, and jump mechanics to the main character.

Week 6

- Display HUD addition and debugging
- The first level's entire level design is complete
- Add enemy and block collisions

Week 7

- Include ending and death for the main character.

- Balance the movement of the main character and the first level for amusement and difficulty.

Week 8

- Using the overall concept as a guide, we should create any missing components, such as opponents.
- Add different palettes to assets
- Implement tests, to make sure our program is behaving in the manner in which we want it to do so.

Week 9

- Finishing touches on all entities and newly added ones.
- Complete Dr Eggman Boss Battle.

Week 10

- Add simple title screen
- Finishing touches for anything that as not been implemented yet
- Clean up bugs
- Add death screen.