Name: Gayathri Poluri
Class Name: CPSC 5270 Graphics/Game Project (25 SQ)
Git Hub Link: https://github.com/gayathripoluri/prince-s-pursuit

Prince's Pursuit

Summary:

Prince's Pursuit is a 2D Game where he doesn't run aimlessly but to prove his true love to his enchantress (Princess). To reach his queen and prove his love, he needs to fight the odds that the universe throws at them, so the princess runs in a forest, and what? Forest doesn't welcome him, but gives him harsh treatment, where the cursed animals try to attack him(sprites like wolves, bears), he can counterattack them using magic bolts. Initially, the magic bolts are 5, and during his run, he will get a few more, and every time he kills an animal, he can be rewarded with a gem or a life, or possibly nothing. The terrible weather (fog, rain) in the middle of the run interrupts him. (CPUParticles2D)

When life throws its tantrums, the universe has a way of rewarding. He will get rewarded with a magical tree in one out of n runs. The magical tree can randomly give 3 gems at a time or slow down time. The Interesting Part of the game is that the prince needs to reach the end of the run(i.e, reach the princess) in a limited time with a certain amount of gems that he collects during his run to win the love of the Princess ("No Gems -No Love Policy")

My approach emphasizes modular growth and individual efficiency. I will start by creating the project skeleton in Godot and putting in place the essential features, which include adversary spawning, item tossing, mobility, a countdown timer, stamina tracking, and simple weather effects. I will then use OpenGameArt.org to import the necessary. Clean pattern-based architecture will be used to layer in branching endings, power-ups, and score tracking (Factory Method, Observer, Strategy, etc.). The finished build will be submitted to GitHub along with source files, tested on other devices, and exported as a stand-alone .exe.

Primary Aesthetics:

1. Sensation:

A rich visual and emotional experience is what Prince's Pursuit is intended to be. The forest is alive; it does not simply exist. Every leaf sways, and every raindrop sings. Real-time effects like fog, wind, and falling leaves are created by the game using Godot's CPUParticles2D, and they look stunning when combined with carefully made pixel graphics. Every action, whether it's a jump, a throw, or a gem collected, is accompanied by a pleasing visual pop or aural cue, and the prince moves fluidly. A magical and reactive world is created for the user by the parchment-style HUD, shimmering gem trails, and ambient sound (soft rain, owl screeches, footfall on moss).

2. Fantasy:

Prince's Pursuit is fundamentally a video game that tells a fairy tale. The adversaries are defenders of a cursed road, not just impediments. The forest is magical and alive; it is

more than just a piece of ground. Although they are uncommon, magical trees serve a purpose by providing gem clusters or gifts that can change the passage of time. With each gem, the player gathers information about the princess's enchantment and the reasons behind her inability to simply say yes to anybody who approaches her. She awaits evidence of devotion rather than quickness.

3. Narrative:

The prince's pursuit is the latter part of a longer plot, not just a random run. The princess's spell is gradually unlocked by gem collecting; this curse can only be lifted if she witnesses the prince deliver both courage (speed) and caring (gems). She vanishes into the mist if he gets there too late. She turns away if he doesn't gather enough.

This is about being deserving, not simply about winning.

Primary Mechanics:

1)Movement:

The player simply has to steer the prince through danger as he runs forward on his own. The woodland transforms into a flowing maze of roots, holes, and dangers as you use the spacebar to jump and the down arrow to slide. The movement is responsive, physics-based, and precisely calibrated**.**

2) Item-Throwing:

Equipped with rare magic bolts and enchanted knives, the prince can defeat or stun fugitives. Every throw is a risk; if you strike a wolf, you might win a jewel. You lose your valuable weapon, Miss. The aim-based throwing system necessitates fast reactions and precise timing.

3)Time Management:

It takes 180 seconds to get to the princess through the forest, but additional seconds are added by gems found along the way (+5 each). Players must decide whether to hurry or not because of this trade-off between speed and collection. Or do I investigate? Your judge and motivator is the luminous timer at the top of the screen.

4)Enemy Avoidance:

Every enemy is a moving puzzle, whether they are charging boars, diving owls, or lunging wolves. Avoid them by sliding and jumping, or risk losing your stamina. On top of that, the weather (fog conceals adversaries, rain slows mobility) makes sure that no run is ever precisely the same.

Structural Tree of Project in Godot:

```
Prince's Pursuit Scene Structure
- Main (Node2D)
  - Player (KinematicBody2D): Movement, stamina, items
    - Sprite (AnimatedSprite2D): Run, jump, slide
    - CollisionShape2D: Physics
  - LevelGenerator (Node): Terrain, enemies
    - TileMap: Forest paths, roots
    - AnimalPool (Node): Wolves, owls, boars
    - GemSpawner (Node): Gems, magic bolt, shield
    - TreeSpawner (Node): Enchanted trees
  - WeatherManager (Node): Rain, fog, wind
    - CPUParticles2D: Weather particles
  - TimerManager (Node): 60-second timer
  - HUD (CanvasLayer): Timer, stamina, knives, gems
    - TimerBar (ProgressBar): Observer updates
    - GemCounter (Label): Gem count
  - ParallaxBackground: Trees, fireflies
    - ParallaxLayer1 (motion_scale=0.2): Distant trees
    - ParallaxLayer2 (motion_scale=0.5): Fireflies
```