CODE NINJAS®

Platinum Belt Project Checklists

Name: _____

Start Date: _____

Activity 01: Gravity Trails

To submit your game for grading, the following requirements must be met. Look for other ways that you can make your project unique!

Planning & Organization

- ❑ The Gravity Trails Planning Document is complete.
- ❑ The hierarchy is organized and game objects are properly named.

Design & Creativity

- ❑ The game contains two complete 2D levels.
  - Level 1 uses the Explore the Forest background.
  - Level 2 uses a background from the Unity Asset Store.
- ❑ Level 2 has a theme, for example, desert, space, ocean, etc.
  - Use the Asset Store to decorate the scene.
  - Use a variety of textures and colors in the scene decorations.
  - Use assets that fit the chosen theme.
- ❑ The scenes include unique assets for enemies, throwables, and collectables.

Coding & Logic

- ❑ The Avatar
  - Can go through the scene using the Gravity Flip script.
  - Can collect and use the throwables to remove the enemies from the scene.
- ❑ Enemies
  - Can collide with the Avatar and throwables.
  - Move within their constraints.

Game Experience & User Interface

- ❑ The Teleport Script
  - The Avatar can continue to Level 2 after all enemies are defeated.
- ❑ Enemy Collision Script
  - The scene restarts when the Avatar collides with an enemy.
- ❑ Create a Start screen that tells the player how to play your game.
- ❑ Create win and lose scenes for Codey.
- ❑ The game should be playtested by one to three playtesters and improved after feedback.

Activity 02: Codey Raceway

To submit your game for grading, the following requirements must be met. Look for other ways that you can make your project unique!

Planning & Organization

- ❑ The Codey Raceway Planning Document is complete.
- ❑ The hierarchy is organized and game objects are properly named.

Design & Creativity

- ❑ The game contains a complete track with a finish line.
  - The track should use at least 10 pieces.
- ❑ The game has a theme, for example, desert, space, ocean, etc.
  - Use the Asset Store to decorate the scene.
  - Use a minimum of 5 setting assets in the scene.
  - Use a variety of textures and colors in the scene decorations.
  - Use assets that fit the chosen theme.
- ❑ Have a minimum of 4 different obstacles on the track.

Coding & Logic

- ❑ The Codey model is functional.
  - Codey can move around.
  - Codey stays on the track.
  - Codey collides with other objects.
  - Codey plays its animations when moving.
- ❑ The obstacles on the track block Codey's movement.
- ❑ The game contains a complete lap system.
  - Checkpoints
    - Codey can't "cheat" by going backwards.
    - The number of triggered checkpoints in the inspector is updated every time Codey collides with one.
  - Timers
    - There are working countdown and game timers.
    - The timers update every second, counting down to 0.
    - The timer texts have unique colors.

- ❑ The game contains item boxes that spawn at four different locations.
    - Item boxes should float and rotate.
    - When Codey collides with an item box, it disappears and reappears
    - The Powerup list is updated.
- ❑ There are at least 3 different power-ups.
    - When the space bar is pressed current power-up is used.
    - The Shell and NavMeshshell can be used to destroy obstacles.
    - A third custom power-up can be picked up and used.

Game Experience & User Interface

- ❑ Codey respawns when it moves off the track.
- ❑ Create a Start screen that tells the player how to play your game.
- ❑ Create a win and lose scenes for Codey.
    - If the timer runs out before Codey completes the lap, the lose scene loads.
    - If Codey passes all checkpoints and makes it back to the start before the timer runs out, the win scene loads.
- ❑ The game should be playtested by one to three playtesters and improved after feedback.

Activity 03: Sulky Slimes

To submit your game for grading, the following requirements must be met. Look for other ways that you can make your project unique!

Planning & Organization

- ❑ The Sulky Slimes Ninja Planning Document is complete.
- ❑ The Hierarchy is organized and game objects are properly named.

Design & Creativity

- ❑ The game contains 3 Levels of complete 3D Scenes.
  - Each scene has a basic ground, platform, and catapult structure.
- ❑ The game has a theme, for example, desert, space, ocean, etc.
  - Use the Asset Store to decorate the scene.
  - Use a minimum of 5 setting assets in the scene.
  - Use a variety of textures and colors in the scene decorations.
  - Use assets that fit the chosen theme.
- ❑ The scene includes unique assets for the slime character, targets, and collectable item.

Coding & Logic

- ❑ The Mouse Manager Script
  - Clicking, holding, then releasing the left mouse click launches the slime character toward the target platform.
  - The values of the click Start Location variable update depending on where the mouse is clicked.
- ❑ Obstacles
  - The slime character collides with and knocks down target items.
- ❑ The Collectable Script
  - The collectable object disappears if the slime character collides with it.

Game Experience & User Interface

- ❑ The Lives Manager Script
  - The total number of lives decrease with each launch.
  - When the player runs out of lives, the scene reloads.
  - If the slime character collides with the collectable, the next level loads.
- ❑ Create a Start screen that tells the player how to play your game.
- ❑ Create win and lose scenes for Codey.
- ❑ The game should be playtested by one to three playtesters and improved after feedback.

Activity 04: Chef Codey

To submit your game for grading, the following requirements must be met. Look for other ways that you can make your project unique!

Planning & Organization

- ❑ The Chef Codey Planning Document is complete.
- ❑ The hierarchy is organized and game objects are properly named.

Design & Creativity

- ❑ The game contains a 3D room game object.
- ❑ The game has a theme, for example, desert, space, ocean, etc.
  - Use the Asset Store to decorate the scene.
  - Use a minimum of 5 setting assets in the scene.
  - Use a variety of textures and colors in the scene decorations.
  - Use assets that fit the chosen theme.
- ❑ At least 5 total interactable objects and 2 final products
  - For example, if the objects added are bacon and fruit, the final products can be egg with bacon or toast with fruit.

Coding & Logic

- ❑ The Interact Script:
  - The code checks for picking up items.
  - Pressing the space bar updates the item held variable.
  - The triggerName variable updates depending on where Codey is.
  - Codey "picks up" items when the space bar is pressed.
- ❑ The Stove Object/ Interact Script:
  - The stove updates the triggerName variable update if Codey is inside the box collider.
  - The stove completes different tasks depending on whether Codey is holding an item when inside the box collider.
- ❑ The Stove Script
  - If Codey is standing in the collider and holding an item when the space bar is pressed, the item on the stove appears and Codey's item disappears.
  - Codey can pick up the item after it's been on the stove.
  - There are particle effects when an item is placed on it.

- ❑ The Receivers Object
  - Codey's triggerName variable updates when inside the box collider.
  - Pressing the spacebar while Codey is holding the item at the Receivers puts the item on the plate.
- ❑ The Second Item (Egg in the tutorial)
  - There is a second item that Codey must take first to the stove, then to the receivers.
- ❑ 3 Additional items
  - There are 3 additional items that Codey must take first to the stove, then to the receivers.
- ❑ Complete Cooking
  - The stove prevents Codey from interacting with it while an item is cooking.
  - There is an obviously different particle effect when it is cooking and when cooking is completed.

Game Experience & User Interface

- ❑ Create a timer for your game (look back at Codey Raceway).
- ❑ Create a Start screen that tells the player how to play your game.
- ❑ Create win and lose scenes for Codey.
- ❑ The game should be playtested by one to three playtesters and improved after feedback.