**Project Documentation:**

**Description of Elements**:

Front-End:

|  |  |
| --- | --- |
| Element | Description |
| Desktop Executable (INSERT\_NAME\_HERE): | When clicked will open the game to the Title screen. |
| Title screen: | This screen has the following interactive buttons:   * Play: Opens the Slot Machine Game Screen * Credits: Opens the Credits Screen   Selecting any of these buttons will cause the screen to fade to black before loading the appropriate screen. |
| Play Screen: | The screen of the game containing the interactive slot machine. On this machine, there is a wallet (displayed as winnings!) displaying the amount of money the user has. To the side of the machine, there is a panel which shows the payouts for each picture. The mechanism of the wheel is as follows:   * A coin can be collected from the tray and inserted into the coin hole. This will increment the number of spins by 1 and decrement the User’s winnings count by 1000. * If there is a non-zero number of spins, the lever can be pulled. * Once the lever has been pulled, each wheel begins spinning. After a couple second delay, the first wheel stops. This repeats for the 2nd and 3rd wheels. * Depending on the result, the amount in the wallet is increased based on the payout displayed in the side panel.   The top left corner includes an interactive button allowing the User to navigate back to the main menu. The bottom right corner contains an interactive button allowing the User to navigate to the credits screen. The top left corner contains the two options: A toggle button for sound, and a dropdown menu for selecting difficulty. |
| Credits Screen: | This screen contains the message “Thanks For Playing!”, along with a few high-level documentation points, including the game engine used, along with a link to the site it can be found. There is also mention of the Github page where all files and documentation will be uploaded. At the bottom of the screen there is an interactive button allowing the User to navigate back to the main menu. |

Back-End:

|  |  |
| --- | --- |
| Script | Description |
| Slot\_Title\_Menu.gd | Title Screen script, which contains the following functions:   * \_ready(): initializes all of the buttons when the title screen is pulled up. * \_on\_button\_pressed(scene\_to\_load): Loads the proper screen to load next. Then, the FadeIn functions show() and \_transition() are called. * \_on\_FadeIn\_transitioned(): calls the function get\_tree().change\_scene(scene\_path\_to\_load), which takes the value of scene\_path\_to\_load and loads the appropriate screen   Contains the following variable:   * scene\_path\_to\_load: stores the screen to load, based on which button is pressed |
| MenuButton.gd | Script used by the title screen buttons Play and Credits to export the proper screen name as a string to the variable “scene\_to\_load”. This variable is then used in by the Slot\_Title\_Menu.gd script. |
| slot-button-credits.gd | Script used by the credits button in the Slot Machine screen. Contains the function \_on\_slotbuttoncredits\_pressed(), which transitions the game to the Credits screen when the button is pressed. |
| Slot-mainmenu.gd | Script used by the Back to Main Menu button in the Slot Machine screen. Contains the function \_on\_slotmainmenu\_pressed(), which transitions the game to the Title screen when the button is pressed. |
| Return\_to\_Main.gd | Script used by the Back to Main Menu buttons in the Credits screens. Contains the function \_on\_Button\_pressed(), which transitions the game to the Title screen when the button is pressed. |
| FadeOptions.gd | Script used by the Title screen to create a fade to black transition when one of the buttons on that screen is pressed. Contains the following functions:   * \_transition(): Plays the fade\_to\_black animation * \_on\_AnimationPlayer\_animation\_finished(anim\_name): if the animation that just finished was “fade\_to\_black”, sends a signal “transitioned” to Slot\_Title\_Menu.gd to begin \_on\_FadeIn\_transitioned(). |
| Lever.gd | Script used by the lever, which contains the following functions:   * \_on\_slider\_drag\_started(): checks if the lever is being pulled * \_on\_slider\_drag\_ended(\_value\_changed): checks whether or not lever was pulled all the way down or not * \_on\_Lever\_animation\_finished(): the lever animation will only be played at a specific time and not anytime * \_on\_slider\_value\_changed(value): changes the value of the slider and will have the lever go back up after the lever was pulled   Contains the following variable:   * rng: random number generator * dragging: a Boolean if an item is being dragged or not * rand\_base: the lowest random integer |
| Wallet.gd | Contains the following functions:   * \_ready(): When loaded, sets the wallet default amount to be 1000 * \_process(delta): the winnings value will change when the coins are selected and the coins shown in the holder will match that value * \_on\_Wheel1\_left\_icon(left\_icon): refers to the variable “icon1” * \_on\_Wheel2\_middle\_icon(middle\_icon): refers to the variable “icon2” * \_on\_Wheel3\_right\_icon(right\_icon): refers to the variable “icon3” * \_identify\_result(icon1, icon2, icon3): determines the total winnings and adds onto the total * \_on\_Spins\_returned(): adds 1000 when the spins are returned back to coins   Contains the following variables:   * icon1: can have an integer value from 0 to 9 and represents the first wheel * icon2: can have an integer value from 0 to 9 and represents the second wheel * icon3: can have an integer value from 0 to 9 and represents the third wheel * result: the winnings for that spin * free\_spin: a Boolean if 5 free spins would be given or not * winnings: total amount that has been won and shown * total: actual total amount * coin: image of the coin   Contains the following dictionary:  Text  Description automatically generated with medium confidence |
| WalletAmount.gd | Gets the integer value from Wallet.gd and keeps updating and displays onto the screen |
| Wheel1.gd  Wheel2.gd  Wheel3.gd | Contains the following functions:   * \_ready(): sets everything in place and makes sure everything is set to default * \_on\_Lever\_pulled(rand\_base): determines the difficulty chosen and randomizes the time when the wheels stop when the lever is pulled * \_on\_Wheel1\_animation\_finished(): adds the number of rotations by 1 for the left wheel * \_on\_Wheel1\_frame\_changed(): determines the number of rotations and when to stop rotating using the timer for the left wheel * \_on\_Wheel2\_animation\_finished(): adds the number of rotations by 1 for the middle wheel * \_on\_Wheel2\_frame\_changed(): determines the number of rotations and when to stop rotating using the timer for the middle wheel * \_on\_Wheel3\_animation\_finished(): adds the number of rotations by 1 for the right wheel * \_on\_Wheel3\_frame\_changed(): determines the number of rotations and when to stop rotating using the timer for the right wheel   Contains the following variables:   * difficulty: has a string value to determine the difficulty level * rng: random number generator * rotations: integer value for the number of rotations * timer: integer value for the time that keeps the wheel spinning * ready: a boolean value where it determines if it is in default setting or not * right\_icon: the value that indicates which icon will be shown for the right wheel * middle\_icon: the value that indicates which icon will be shown for the middle wheel * left\_icon: the value that indicates which icon will be shown for the left wheel |
| Spins.gd | Contains the following functions:   * \_ready(): makes sure everything is set to default * \_on\_Lever\_pulled(\_rand\_base): * \_on\_Wallet\_freespins(number): adds 5 spins to total spins amount * \_on\_CoinReturn\_animation\_finished(): subtracts the total amount of spins and displays the updated text for amount of spins remaining * \_on\_drop\_coin\_insert\_coin\_inserted(): when coin is inserted adds the amount to the spin and edits the text for the amount of spins remaining   Contains the following variables:   * spins: integer value for the total amount of spins * string: display text format * actual: actual text displayed |
| drop\_zone.gd  drop\_zone\_2.gd | Both scripts define where the coin will go when it is selected and dragged over either the coin slot or anywhere else on the screen. |
| Node2D.gd | Contains the following functions:   * \_ready(): makes sure everything is set to default * \_on\_Area2D\_input\_event(\_viewport, \_event, \_shape\_idx): describes what happens when the coin is being dragged and when the coin is released * \_physics\_process(delta): makes the movement of the coin look smoother. (based on information gotten from Youtuber Bramwell: https://www.youtube.com/watch?v=iSpWZzL2i1o) * \_input(event): describes what something happens when the mouse is being clicked |