Slot Machine Exam

Documentation

Completed by John Francis Balmeo
06/22/2023

Unity Version:

The exam project used Unity 2021.3.26f upon development.

System Setup:

Main Classes and Controllers

- SpinButton.cs: This script is responsible for handling the spin button click event. It references the SlotMachineController script to initiate the reel-spinning process.
- SlotMachineController.cs: This script controls the functionality of the slot machine. It handles spinning the reels, calculating scores, updating UI elements, and managing game variables.

Objects:

- Image[] reelImages: An array of Image components representing each reel's symbol placeholder.
- Sprite[] symbols: An array of sprite symbols used as reel symbols.
- int score2Symbols: The score value for matching 2 symbols on a line.
- int score3Symbols: The score value for matching 3 symbols on a line.
- int score4Symbols: The score value for matching 4 symbols on a line.
- int score5Symbols: The score value for matching 5 symbols on a line.
- int totalScore: A variable to store the cumulative score.
- int moneyCount: A variable to store the available money count.
- int minBid: The minimum bid amount.
- TMP_Text winningsText: A TextMeshProUGUI component to display the current winnings.
- TMP_Text moneyText: A TextMeshProUGUI component to display the available money count.
- TMP_Text bidText: A TextMeshProUGUI component to display the current bid amount.

Data Sources and Editing:

The data sources for the slot machine symbols and scoring combinations are defined within the code:

- The array symbols[] contains the sprite symbols used as reel symbols.
- The array lines[][] defines the line combinations for scoring.

To edit the symbols or add new symbols:

- Modify the symbols[] array by adding or replacing sprite symbols.

To edit or add new scoring combinations:

- Modify the lines[][] array by adding or changing the line combinations.

Additional Notes:

Scalability

- The current implementation randomizes numbers per box.
- The current implementation supports 20 predefined line combinations.
- By adding new sprite symbols to the symbols[] array, the system can handle more symbols.
- The UI elements can be enlarged or adjusted to accommodate extra features or information.

Flexibility

- The minimum bid amount (minBid) can be adjusted according to the game requirements.
- The scoring values (score2Symbols, score3Symbols, score4Symbols, score5Symbols) can be modified to balance the game.
- The UI elements (winningsText, moneyText, bidText) can be restyled or customized to fit the game's theme.

MVC Implementation

- The SlotMachineController script acts as the controller, handling game logic and managing data.
- The reelImages array represents the view, displaying the reel symbols to the player.
- The winningsText, moneyText, and bidText components serve as the view, presenting the game's state to the player.

Future Improvements/Recommendation

- Fix inefficient combination lines and bugged scoring.
- Addition of scroll animations per reel.
- Addition of music to increase entertainment.
- The UI can be further improved by adding more informative elements, such as a paytable or player statistics. Creation of a Better UI Design.