

Retro VR Slot Machine v1.1

CARMEN dimichele

22-September 2024

.....
IMPORTANT UPDATE

The reels will not render properly without this update. Go to Assets | Materials | slot machine | reels | RotatedOpaque.shader and modify it as per

<https://docs.unity3d.com/Manual/SinglePassInstancing.html>:

```
...
struct appdata_t {
    float4 vertex : POSITION;
    fixed4 color : COLOR;
    float2 texcoord : TEXCOORD0;

    UNITY_VERTEX_INPUT_INSTANCE_ID // insert this line
};

struct v2f {
    float4 vertex : POSITION;
    fixed4 color : COLOR;
    float2 texcoord : TEXCOORD0;
#ifdef SOFTPARTICLES_ON
    float4 projPos : TEXCOORD1;
#endif

    UNITY_VERTEX_OUTPUT_STEREO // insert this line
};

v2f vert (appdata_t v) {
    v2f o;

    UNITY_SETUP_INSTANCE_ID(v); // insert this line
    UNITY_INITIALIZE_OUTPUT(v2f, o); // insert this line
    UNITY_INITIALIZE_VERTEX_OUTPUT_STEREO(o); // insert this line

    o.vertex = UnityObjectToClipPos ( v.vertex );
    o.color = v.color;
}
...
```

.....

Retro VR Slot Machine is a fully functional retro slot machine including fully materialized slot machine and pedestal models, images, video clips, sound effects, and scripts, in Unity for use in VR projects.

The slot machine can be configured for a predetermined sequence of outcomes or use the random, 91.5% return-to-player, pay table with a 1 to 3 credit multiplier. The big win produces a 'genie fortune card' reward along with a fanfare sound effect.

The slot machine also has a circuit breaker with push-button reset, which shuts the slot machine down when it's 'blown', triggering sparks and smoke particle effects

This package depends on the SteamVR plugin and contains:

- Two demo scenes
- Scripts for operating the slot machine game play, the machine's ratcheting handle action, and the demo room control panel
- Eight 44.1kHz .wav sound effects
- Thirteen PBR materials with over thirty-seven 2048x2048 textures
- Ten 260x180@30fps .avi reel mechanism video clips
- Ten prefabs:
 - Slot machine prop with three scripts and internal wiring to operate the machine, handle mechanism, and blown fuse system. This prefab has a 21 mesh, 20373 polygon model and uses 21 materials.
 - Blow fuse effect. This prefab uses two particle systems to create smoke and sparks when the slot machine blows a fuse.
 - Spark light game object. This prefab is a light object simulating a spark used in the blow fuse particle system.
 - Ornate copper and marble pedestal static prop. This prefab has a 6 mesh, 127465 polygon model and uses 2 materials.
 - Slot machine on a pedestal as a single prefab
 - RNG controller. A prefab with only a script to centralize the Unity system random number generator to reduce the chance of multiple instances initializing in-sync.
 - Demo room and Empty demo room. A small six-sided room using 7 materials with studio lighting, 3 slot machines on pedestals, and a control desk used in the demo scene.

- Desk and Control desk. These prefabs make-up a control desk with buttons, text, and a script for letting you access some of the internal functions of the slot machines in the demo scene.

.....

SETUP

- Create a new 3D project, import the SteamVR Plugin, and make sure the OpenVR package is installed

The steps in getting setup may be different depending on the version of Unity you are using. I am writing this while using Unity version 2019.1.5f1 Personal. Start the Unity editor with your new 3D project, go to the Unity Asset Store, search for the "SteamVR Plugin", and download and import it into your project. "Accept All" SteamVR recommended project settings. Click "Package Manager" under the Window menu item. Look for the "OpenVR" package and install it if it is not already.

- Import the Retro VR Slot Machine package

Click the "Asset Store" tab on the Unity main scene pane, click the "My Assets" icon next to the "Search for assets" bar on the Asset Store home page and click the "Download" button next to the "Retro VR Slot Machine" package in the "My Assets" list. After downloading is complete, click the button again, now labeled "Import", and hit the "Import" button on the bottom-right of the pop-up dialog to import all of the Retro VR Slot Machine assets. In Edit | Project Settings | XR Plug-In Management, make sure "OpenVR Loader" is checked.

- Open the Demo Scene

Double-click the "Demo Scene" asset under Assets | Retro VR Slot Machine | Demo | Scenes in the project navigator.

- Hit play!

The first time you hit play in this new project, SteamVR will ask you to generate SteamVR Input actions. Click "Yes" to open the SteamVR Input window, "Yes" to use the example actions.json files, "Save and generate" toward the bottom of the window to compile using the defaults. After the compilation completes, hit the "X" in the top-right of the SteamVR Input window to close it. This compilation will sometimes list errors in the Unity Console which will clear after the scene is played. Hit the Unity editor play button again to actually play the demo this time.

Click a controller trackpad to teleport around the space. Touch the "Bet" and "Spin" buttons to play the slot machine. After betting, you can also spin the slot machine by grabbing its handle and pulling it toward you. The control desk in the center of the demo room lets you perform other functions internal to the slot machine, like blowing its fuse.

.....
THEORY OF OPERATION

The "Slot Machine" prefab includes all necessary objects for a fully functional slot machine. This prefab uses the SlotMachine.cs script to perform the overall functions of the slot machine. SlotMachine.cs exposes the following public fields:

- Sounds

The collection of sound effect assets used for the various functions of the slot machine.

Bet Sound:	played when the bet button is pressed
Win Sound:	played for each credit that is won
Short Fanfare Sound:	played after a jackpot is won
Flip Card Sound:	played when a new belly window card is presented
Bootup Click Sound:	played during bootup sequence
Blown Fuse Sound:	played when the slot machine fuse is blown
Circuit Breaker Sound:	played when the circuit breaker reset button is pressed

- Video Clips

The collection of video clip assets used by the slot machine for simulating its spinning mechanical reels. The "Spin Video Clip" is played when a game play is started. One of the "Winning Video Clips" is selected based on the predetermined game outcome and played directly after the "Spin Video Clip" to complete a game play. A "Loser Video Clip" is selected at random to play after the "Spin Video Clip" in the case of a non-winning outcome.

- Images | Belly Images

The collection of image assets used by the slot machine as belly window cards. The first image in this list (Element 0) is shown in the slot machine belly window on reset. The next card in the sequence is presented in the window on a jackpot outcome. The sequence will recycle to Element 0 when the list has been exhausted.

- Play | Starting Credits

The number of credits the slot machine will start with on reset.

- Play | Auto Play

Sets the slot machine to play by itself if enabled. The "Until No Credits" mode will stop auto playing when no credits remain on the slot machine credit meter. The "Add Credits" mode will add 10 credits to the credit meter when auto play has exhausted the slot machine credits, then continue to play.

- Play | Setup Sequence

A sequence of outcomes the game will follow as it is played. The sequence will recycle when the end has been reached. Outcomes are "No Win", "Win 2", "Win 5", "Win 10", "Win 14", "Win 18", and "Win Jackpot". Each win pays its stated amount times the number of credits bet.

Winning the jackpot pays 25 times the bet and awards the player with a new belly image card. The slot machine standard game play operates on a random, 91.5% return-to-player payable when this list has no elements. See the SlotMachine.cs code for details.

- Play | Handle Locked

Exposes the condition of the slot machine handle lock. The handle is restricted from being pulled to spin a game when it is locked. The handle is locked when no credits have been bet. The handle becomes unlocked when a credit is bet, allowing the handle to be pulled to spin a game.

The "Slot Machine" prefab also uses the HandleMotion.cs script in performing the slot machine handle action. HandleMotion.cs exposes the following public fields:

- Sound | Handle Thud

The sound effect asset played when the handle is released, springing back to its resting position

- Sound | Ratchet Click

The sound effect asset played for each ratchet position as the handle is pulled

- Ratchet Degrees

The angular degrees of rotation for each click of the ratchet as the handle is pulled

The slot machine "reels" game object uses the "reels" material. The "reels" material uses the custom RotatedOpaque.shader. This shader allows rotating the spinning reels video clips to match the orientation of the cylindrical reels mesh without having to unnaturally rotate the video content. This shader exposes the following fields:

- Albedo

The texture that will be mapped onto the mesh

- Tiling/Offset | X, Y

The horizontal and vertical tiling factors for mapping the albedo texture to the mesh

- Tiling/Offset | Z, W

The horizontal and vertical offset values for adjusting the mapping of the albedo texture to the mesh

- Rotation (deg)

A frame rotation applied to the albedo texture when mapping it onto the mesh, in degrees

- Color Filter

A color that is applied by multiplying the RGBA components of the Color Filter setting with the albedo texture

- Render Queue

Which order of rendering this material will follow

- Enable GPU Instancing

Selected to render multiple instances of the same mesh at once, using a small number of draw calls

- Double Sided Global Illumination

Selected to have the Unity lightmapper account for both sides of the mesh when calculating global illumination

The "Blow Fuse" prefab contains the "sparks" and "smoke" particle systems and is used to simulate an electrical fuse being blown in the slot machine. Call the BlowFuse() public method in SlotMachine.cs to trigger the blow fuse simulation. This will play the sparks and smoke particle systems and blown fuse audio clips, then turn off all slot machine electrical lighting and render it unplayable. A circuit breaker reset button on the back of the slot machine pops out when the fuse is blown. The player can push the button back in to turn the slot machine on and begin the bootup sequence. The slot machine credit meter will be reset to the Starting Credits value after the bootup sequence is complete.

The "Pedestal" prefab is a prop with no moving parts or scripts. The "Slot on Stand" prefab hides the pedestal top to use the slot machine base atop the pedestal instead.

It is necessary for one and only one "RNG Controller" prefab to exist in the scene. The only purpose for this prefab is to act as a centralized random number generator. If we use multiple slot machines in the same scene and allow each to have their own RNG instance, they can sometimes get in sync even with initializing the RNG on system ticks.

.....
LIGHTING TIPS

- All meshes have handmade uv maps and do not need Unity lightmap generation.
- The "Pedestal" prefab has no moving parts and can be completely set to static for high-quality lightmap baking.
- The "Slot Machine" prefab is mostly static. The bet button, spin button, handle, and circuit breaker are movable and not set to static in the prefab.

- The credit meter, bet meter, and paid meter are text meshes with "Mesh Renderer | Lightmap Static" turned off in inspector to avoid warnings about Global Illumination being ignored for some meshes with invalid normals.
- Speed-up baking the "Demo Scene" by using the "Enlighten" Lightmapper with "Ambient Occlusion" off. It won't look as good as the default demo scene setup, however.

.....
CREDITS

Fonts

- Uses and redistributes "erbos_draco_1st_open_nbp.ttf" by Nate Halley (<http://fontstruct.com/fontstructions/show/703185>). Erbos-Draco-1st-Open-NBP was built with FontStruct (<http://fontstruct.com>). FontStruct is a trademark of FSI FontShop International GmbH. Licensed under Creative Commons Attribution Share Alike (<http://creativecommons.org/licenses/by-sa/3.0/>). A copy of the license in a redistributable archive can be found in the Assets | Fonts | Redistributable folder. No changes were made to this font.
- Uses and redistributes "Merriweather-Regular.ttf" (<https://github.com/EbenSorkin/Merriweather>), licensed under the SIL Open Font License, Version 1.1 (<http://scripts.sil.org/OFL>). A copy of the license in a redistributable archive can be found in the Assets | Fonts | Redistributable folder. No changes were made to this font.
- Additional fonts used without redistribution were Aladin (<https://fonts.google.com/specimen/Aladin>) and Lobster (<https://fonts.google.com/specimen/Lobster>), licensed under the SIL Open Font License (https://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL_web), and Open Sans (<https://fonts.google.com/specimen/Open+Sans>), licensed under the Apache License, Version 2.0 (<http://www.apache.org/licenses/LICENSE-2.0>).

Materials and Textures

- All models were made in Blender v2.79b and exported to Unity using FBX directly.
- Some textures were from <https://freepbr.com> and <https://cc0textures.com>.
- Uses "[free] Copper very old.mat" and "[free] Marble polished white.mat", free from LEX4ART (<https://assetstore.unity.com/packages/2d/textures-materials/real-materials-vol-0-free-115597>).
- Uses "3DText.shader" by Eric Haines (Eric5h5) (<http://wiki.unity3d.com/index.php?title=3DText>). Licensed under Creative Commons Attribution Share Alike (<http://creativecommons.org/licenses/by-sa/3.0/>). No changes were made to this shader.
- Uses "Lobby-Center_2k.hdr" (http://www.hdrlabs.com/gallery/flashpanos_hollywood/pano.html?Popcorn_Lobby&). Free from HDRLabs.com.

Sounds

• Uses "240281__kgatto__thuds.wav" (<https://freesound.org/people/kgatto/sounds/240281/>) by kgatto (<https://freesound.org/people/kgatto/>). Published by freesound.org (<https://freesound.org>). Licensed under Creative Commons Attribution 3.0 Unported (<https://creativecommons.org/licenses/by/3.0/>). Sounds were clipped, resampled, equalized, and normalized.

.....

Reach-out if you have any questions:

- email: dimichelec@gmail.com
- website: <https://dimichelec.wixsite.com/carmendimichele>
- unity connect: <https://connect.unity.com/u/5c9060b9edbc2a25a143419a>