

BOOREAKOUT *A SPOOKY ATARI BREAKOUT*



INSPIRATION

Recently I have been playing a game called ‘Peglin’ which is an alternative take on Atari’s classic ‘Breakout’ game. It introduces roguelike elements and combines it with deckbuilding to make the original gameplay more varied. This inspired me to create my own take on the game. Taking into account that this is the first time I delve into game development, a game as complex as Peglin would be beyond my current abilities so I decided to create my own version of the classic Breakout first.

After I had the first prototype running I noticed that every level felt very similar as the objective is to destroy every block. Unhappy with that, I knew I wanted the game to have a different core objective. Still unsure what that objective would be, I focused on the theme of the game first. Brainstorming several aesthetic ideas, for example a neon arcade style or a space battle where your ball is a laser hitting spaceships. In particular one of the ideas I had was to light the ball on fire and make it a literal fireball. Sticking with this idea I expanded the game with other typical wizard props like runes, gems and an ancient tome containing the spell to summon a ghost.

Taking inspiration from the theming I also came up with a twist on the game’s objective. Instead of destroying each block in a level, the player now had to light skulls on fire to complete the summoning of a ghost. Instead of having to destroy each block which can make the original game feel tedious at times, players now only had to target the unlit candles. Once the player achieves their goals, they are rewarded with a little victory screen and an image of the summoned ghost. That is also where the title of the game, BOOreakout, came from.

ART STYLE

Now that I knew where I wanted my game to go, I had to figure out how I wanted to present my game. Personally I like to challenge myself trying new styles I've never done before and after taking inspiration from 'Metal Slug', 'Owlboy', 'Cave Story' and the aforementioned 'Peglin', I decided to go with a pixel art style.

Frankly I wasn't happy with my first attempts, which you can see in the design iterations with the first stone, paddle and steelblock. I decided to take a step back and looked at examples from the games I mentioned before and what makes their sprites so appealing. I learned that in that low res environment there is a big emphasis on creating contrast to show form. I also noticed a technique called dithering similar to cross hatching in pencil drawings to create different tones without using a different color. Trying to implement what I learned I made a set of 4 brushes



SOLID PIXEL BRUSH



TEXTURED SOFT EDGE BRUSH



DITHERING BRUSH

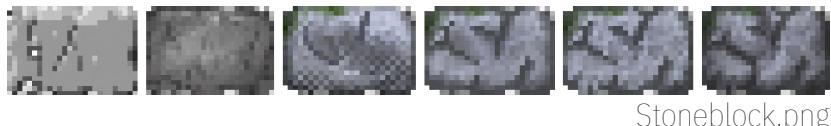


NOISE BRUSH

First I use the solid pixel brush to create the shape. Then I use the soft edge and dithering brush to shade and establish form to the object. Lastly I use a combination of the noise brush to add texture and use the other 3 brushes with different opacities to refine that texture. While that was my general approach I often had to go back and forth between shading and texturing to achieve what I was looking for

One approach I want to emphasize, because I extensively used it when creating the golden highlights on the book, was to draw the center of the highlight with the solid pixel brush first and then adding a shine to it going over the center with a bigger soft edge brush in the same color with reduced opacity.

DESIGN ITERATIONS



Stoneblock.png



Ball.png



Skullblock.png



Crystal.png



Steelblock.png



Paddle.png



Particle.png



Sandblock.png



Mossblock.png

REASONS BEHIND DESIGN CHANGES



HARD TO TELL INDIVIDUAL BLOCKS APART



While implementing a new texture ingame I discovered new problems that I didn't notice looking at the individual sprite when drawing. One thing that happened to me often was that, when the same block is placed next to each other they start to blend into each other. The way I fixed it is to go over them with the soft edge brush and darken the shadow around the edges



ARTIFACTS APPEARING

A unique problem occurred when implementing the sandblock. Like the problem before this one only became obvious when the texture was ingame. Here the shadow on the right becomes a very obvious line when you zoomed out, so I needed to adjust the shape



TOME BREAKDOWN



First I blockout the tome



I tried to emphasise the binding



And decorate it



Unhappy with the look I changed it to gold



Shaded the pages



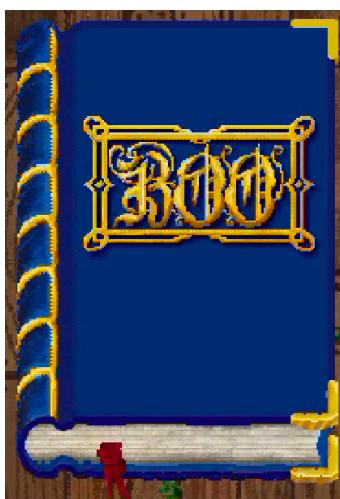
Added texture to the pages



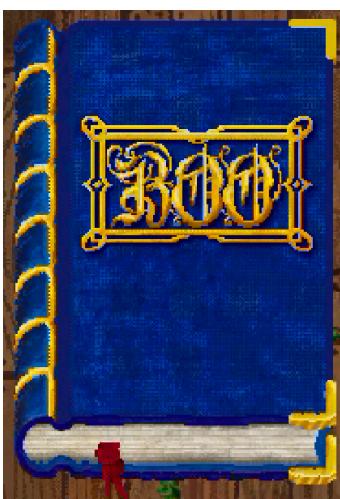
Added more shading so you can see the pages



Adding form to the bookmark



Flipped the bookmark



Texturing the cover to make it less flat



Decorating it to make the book feel more special



Adding runes to tie the tome into the theme

GAME CODE

The game is written without the use of a game framework. I used C++ and OpenGL for game logic and rendering respectively. The shaders are written in GLSL and it uses the GLFW a library for application window handling. To load and decode image files I used stb_image.

The game makes use of a typical game loop using a delta time to make the experience independent from the speed of the PC and potential framerate hick-ups. To keep the application window responsive this loop also contains a check of the current operation system's message pump. The game is split into three states, GAME_MENU, GAME_ACTIVE and GAME_WIN. The game starts out in the menu and upon pressing the spacebar, the game state is switched to 'active' and the actual game starts.

The game makes use of simple 2D box and sphere collisions; the ball travels at a constant velocity and depending on the object it collides with either destroys a block, lights a candle, bounces off the wall/the paddle or ends the game if it collides with the lower bound of the screen.

Levels are stored in separate text files which each contain a 2D matrix of numbers. Each number maps to a certain block in-game like 6 being a skull and 2 being a regular destroyable block. That way I did not need to hard-code levels in the game's code but instead could quickly change the layout on the fly.

If I were to extend the game, I would add a multiplayer mode making the game more akin to an air hockey game or perhaps add a level editor.