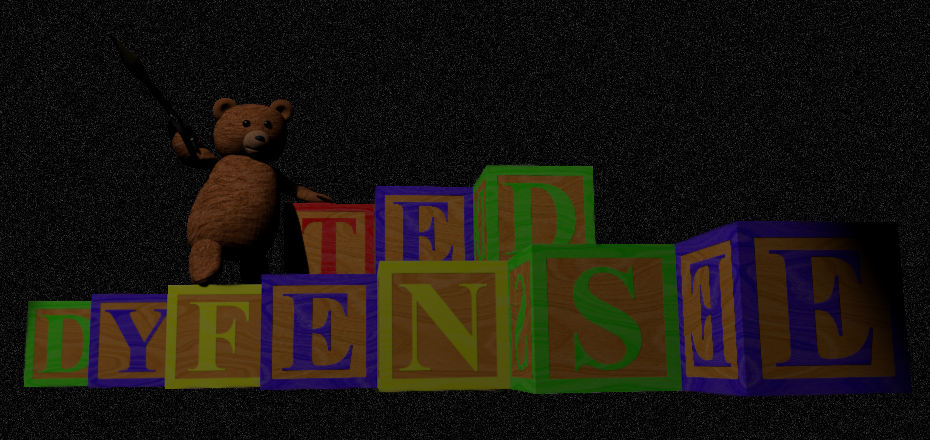
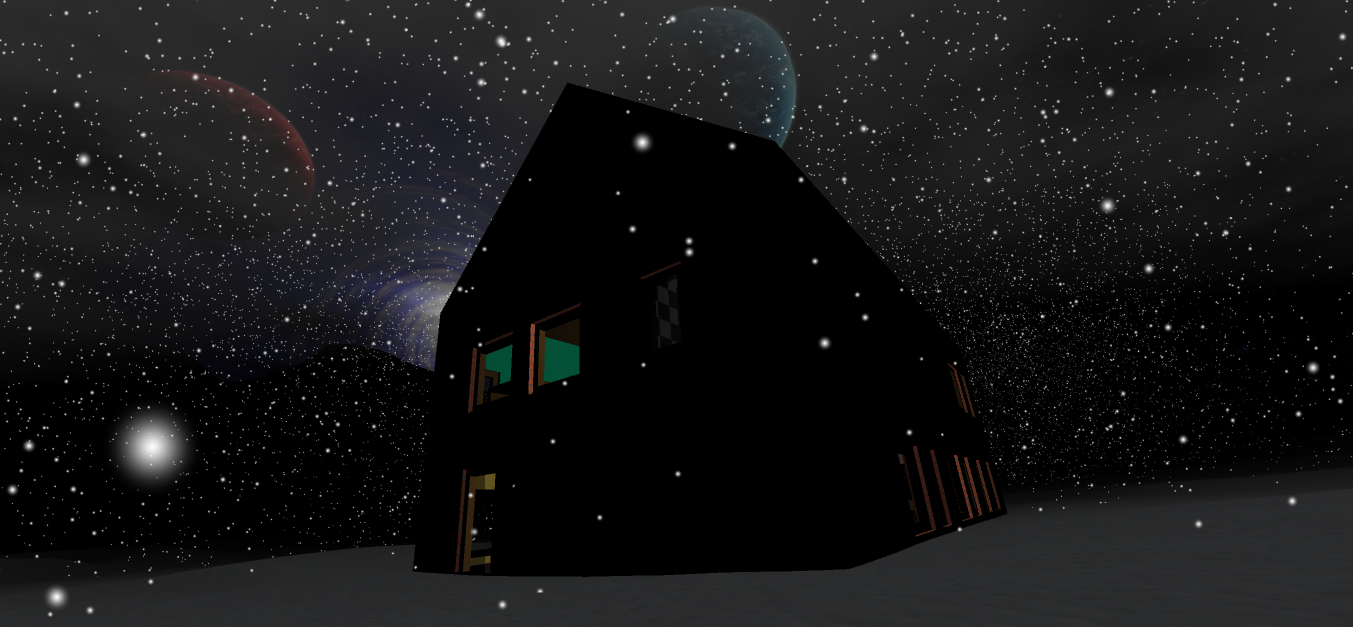
****

TedDyfense



<Team Picture>

Development team:

Abdullah Tezcan Bouwkunde

Esmeralda Tomasöa Technische Natuurkunde

Pieter Kools Werktuigbouw

Damien Crielaard Technische Natuurkunde

Arjan van Ramshorst Werktuigbouw

Jeroen Methorst Nanobiologie

**Introduction**

This document will introduce the reader with TedDyfense, a game which we have been developing the past 10 to 12 weeks in order to complete our minor project. The document will focus on the technically challenging components, the code quality and its management, the art and design process, and finally the development process we used to achieve our final result.

**Target Audience**

**Platform and Controls**

The game has mainly been tested on windows platforms, and any debugging was focused on achieving a working Windows build, though we have also tried a few Mac builds which worked without problems.

The controls are simple and resemble the average shooter, with a few additions.

|  |  |  |  |
| --- | --- | --- | --- |
| **Action** | **Key** | **Action** | **Key** |
| Movement | W,A,S,D | Place block | F + Left Mouse Button |
| Jumping | Space | Remove block | F + Right Mouse Button |
| Running | Left Shift | Pause game | P |
| Fire weapon | Left Mouse Button | Reload weapon | R |
| Aim through sights/scope | Right Mouse Button | Switch between 1st/3rd person | F5 |
| Change weapon/blocks | Middle Mouse Button (Scroll) | Enable Torch | T |

**Story and setting**

The story is about a young child, "Little Johnny", having a nightmare and the only thing standing between him and his nightmares is the player, "Little Johhny's Teddybear".

The setting of the game is a nightmare version of Little Johnny's house, placed into a surreal environment. The house is filled with darkness and shadow and enemies like clowns and bats are coming from every corner of the house.

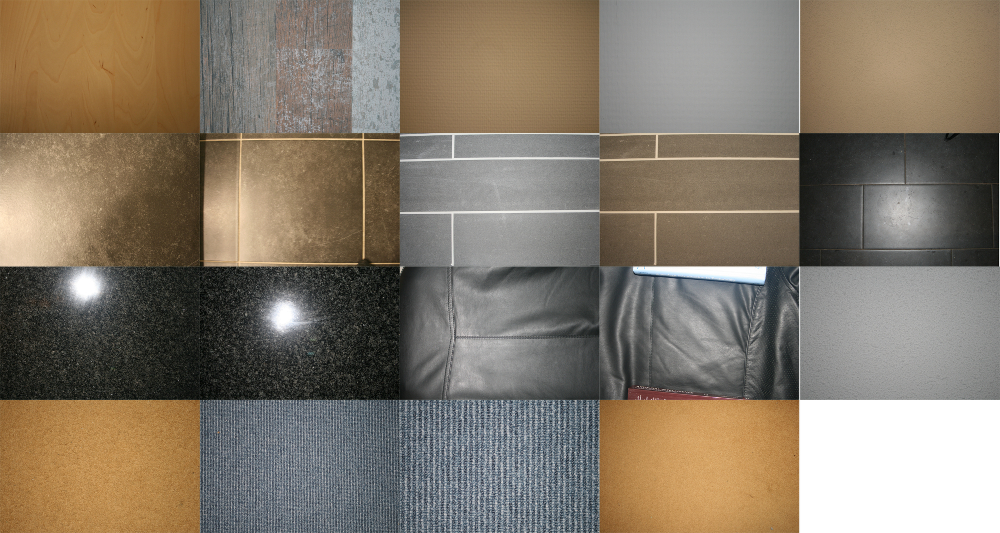
As Little Johnny's Teddybear, you are tasked with defending Little Johnny from his own nightmare. To accomplish this, various heavy weaponry (Colts, AKs, Rocket Launchers, Shotguns) can be found throughout the house for you to use, but don't stray too far from Little Johnny.

**Technical Components**

**Code Quality**

**Art, Design, and Style**

Textures - Jeroen

Most of the textures in the game have been created from photographs made by myself. These photos were digitally edited to reduce differences in brightness and allowing them to be tiled seamlessly.  
   
*(Small example of the different photos used in texture creation)*

A small amount of base textures were taken from CGTextures. These textures were used in the creation of larger textures. (for example the planets in the skybox, these were made with small parts of 5 different textures). The sources of these 3rd party textures have all been documented and can be found in the GitHub directory (\Graphics\Textures\\_Base\_Textures\3rd Party\Sources.txt).

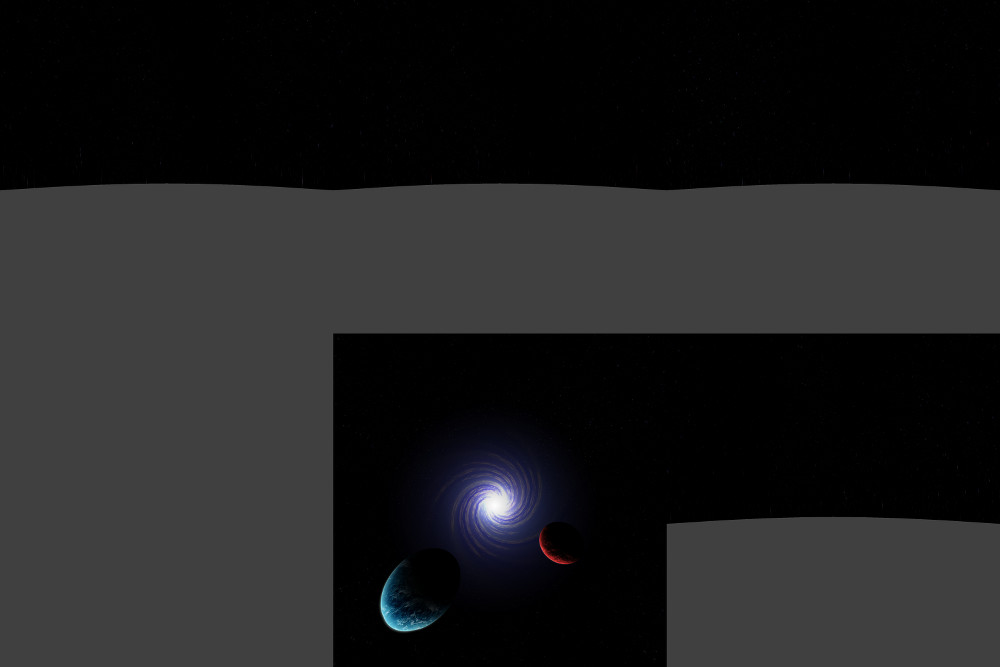
These textures have been used everywhere in the game. Walls, floors, furniture, weapons, the player character, and also the enemies.

Skybox - Jeroen

The game's skybox was created from a large texture built up from several smaller textures. The planets were created from base textures which have been projected on top of a sphere, while the galaxy was created using a rotation script in GIMP. The stars were generated from noise.

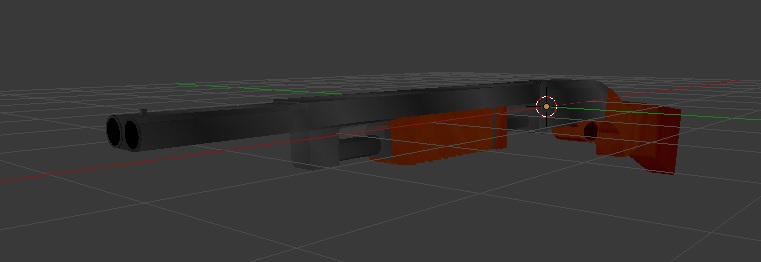


Then, to create the 6 skybox textures to be used in Unity, I imported the skybox texture into blender and projected in on top of a large dome covering most of a small cube. Then I used environment mapping to map the dome onto the cube, resulting in a file which could be used as a skybox in Unity.



Weapon Models and Textures - Jeroen

The weapon models were created using Blender by tracing over photographs of the actual weapons from different angles, this process took some time but resulted in pretty accurate models of the weapons, accurate enough for our game.



The textures of the weapon models were created by UV-mapping the individual parts of the weapons in Blender, exporting the UV-mapping to a .png file, and then adding textures to the exported UV-map.

*(Example of the Shotgun Texture)*

**Development Process**

**Conclusion**