“The White Guy in the White World”

Jonathan Plasman

348652, CMV1D

Haxe

Index

Introduction (3)

Classes (4)

Introduction

“The White Guy in the White World” is a short storybook game about The White Guy who lost all his friends because all his friends are white and he could not see them. He goes on an adventure to search his friends and must make some decisions which ultimately leads to either a failing or a victory end state. When someone reaches, the victory state the person can enter his name in the leaderboard. The story itself is vague and abstract as there is little information about the main character and the world he lives in.

Note that for unknown reasons the .exe file of the game does not work, however the game works when compiled straight from Haxe.

Classes

Main

The game has only 4 classes, and the first and biggest one is the Main class. This one starts off with a lot of variables, about half of them are TextFields. These are used to display text on the screen, but I used way too many although I did remove some. There are some variables for buttons (which uses a template from the Button class) which are used in the main menu. There is also a variable called stopTimer which is later used to make sure a part of the function TooLate is not used.

The first function new calls the function gamestart and togglescreen and plays the music which uses a template from the class Menu, which should be named SoundManager or something like that. The function gamestart simply creates some buttons which leads to the credits, leaderboard, start of the game, play/stop music and quitting the game. The play/stop music is switched in the functions OnPlayMusicButton and OnMusicButton where it simply removes the previous child and replaces it with the current one.

When you start the game (function startgame (not to be confused with gamestart)) removes all children, adds the background again, sets 2 variables for a textformat (which I should have done at the top of the code instead of adding it for every function with textfieldfunction) and calls a function from the Database class. An event listener is added to the answer buttons and both take you further in the story. This is repeated a few times in OnNextQuest, OnWalk, OnBluffing, OnFight, OnStay, OnEnd, OnBlackButton, OnBlack2Button, OnBlack3Button, TooLate and Victory. Whenever there is an ending in the game I added functions to take the player back to the main menu. OnBlack4Button and OnBackButton do the same and I should have removed one of them, they simply take you back to the menu. The function OnWalk also has a timer based on the actuate library. When the timer runs out the function TooLate is called and the game is over. When the player chooses on time the Bool of stopTimer goes to “true” and the screen does not jump to the TooLate ending.

Every Button and TextField has a Roll and Out function which makes the text bigger when hovering over it with a mouse, it also plays a SFX sound.

OnCreditGame and OnLeaderBoardButton both call their data from the Database class which is then presented on the screen with textfields and a button to back to the menu.

Button

This function automatically converts the bitmapdata to a new bitmap which saves a lot of space in the Main class.

Menu

Like I said before, this class should be called SoundManager since it is a soundmanager. It creates a soundchannel where any music can be chosen in the Main. This also saves some space and makes the code clearer, this is also done for the SFX sounds.

Database

A connection is made with the highscores.db, credits.db and the Q1.db. To request certain information from them I used a request function, sometimes limited by 10 and sometimes limited to only 1 line of text. These are requested from the Main class and put in TexFields with either a MouseEvent or just plain text.