

Avatar: The Last Airbender Game



Created by:

Dareen Atallah
Weiyu Huang
Peter Liu
Vince Robbins



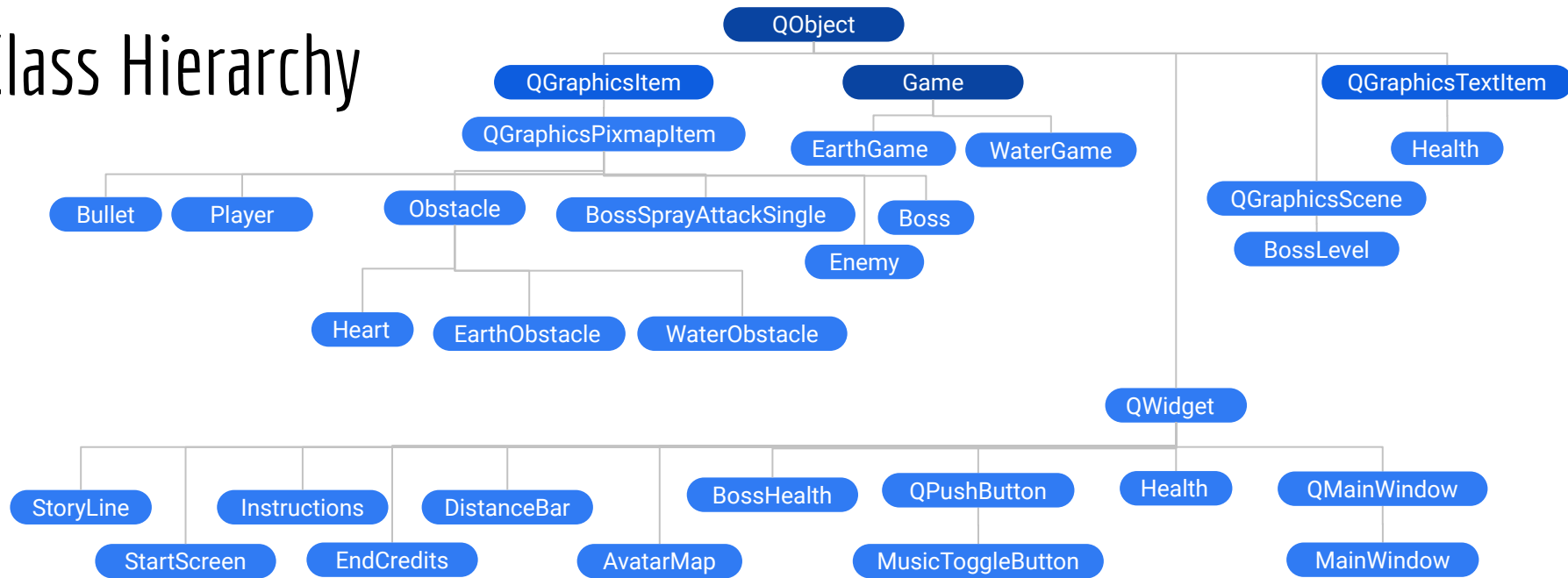
Prelude

The game is based on the TV show **Avatar: The Last Airbender** where the Avatar, Aang, is the spirit of light and peace manifested in human form. He is supposed to control the four elements, water, earth, fire, and air and is tasked with keeping the Four Nations at peace. He is sent on his journey to visit the three nations, Water, Earth, and Fire in order to acquire their elements. To keep all the nations at peace, he must ultimately defeat the fire lord and return peace to all.

How Our Game Works

- 1) Begin at the **Water Nation** and move the Avatar up, down, left, or right to avoid all obstacles that come your way (i.e. water icicles and birds), then acquire your water bending power, a **water bomb**.
- 2) Once you've survived the Water Nation level you move onto the **Earth Nation** level and avoid all obstacles which move faster (i.e. vines and birds). You acquire your earth power, **invincibility**.
- 3) After the Earth Nation level you are moved to the final boss in the **Fire Nation**. Fireballs are emitted from every direction, and you must avoid them and use your elements to kill the final boss.

Class Hierarchy



Uniqueness of Our Game

- Each new scene is created by instantiating an object of the next scene, setting it as the central widget for mainwindow and then deleting itself.
- This game is based on a TV show that was created many years ago and sums up all the chapters of the show.
- This game was created by 4 people where 75% of them never watched the show.

StartScreen

This is the first screen of the game!

Contains objects of kind:

- MainWindow //used to switch to the next screen
- QPushButton //(x2) one is used for beginning the game and one to mute //music
- QLabel //used to format the title of the game and add shadows and emphasis
- QGridLayout //used to organize the widgets added to screen
- QMediaPlayer //creates an object to play the music we choose
- MusicToggleButton //used to mute the music

Important Functions:

void create_storyline() const; //this is a public slot that creates the storyline

AvatarMap

This is where you gain access to all the levels!

Contains objects of kind:

- MainWindow //used to connect the next level to the current nation's //button
- QPushButton //(x5) the buttons used for each nation i.e. Water, Fire, Earth
- int //keeps score of how many levels have been beaten

Important Functions:

create_WaterGame(); //slot that begins the water game

create_EarthGame(); //slot that begins the earth game

create_BossGame(); //slot that begins the boss game

Game

This is where all the classes like obstacle, player, and hearts come together!

Contains objects of kind:

- QPixmap //used to create the images of the player and obstacles
- Player //used to create the player and control movements
- QGraphicsScene //a surface for managing a large number of 2D objects
- distance_bar //the distance the player has traveled throughout the game

Important Functions:

void generate_heart(); //creates heart objects and adds them to the scene and
//destructs after a certain amount of time

void generate_enemy(); //creates enemy objects and adds them to scene and
//destruct after a certain amount of time

Player

This is where the controls for the player are found!

Contains objects of kind:

- bool //(x2) used to keep track of if a bomb has been launched and if it can be launched in order to keep track of timing
- QTimer //used to keep track of the the time until the next bullet or bomb can be shot

Important Functions:

void keyPressEvent(QKeyEvent* event) override; //one is used to create the //water bomb and air bullet, it also creates the pixmap and launch of each item //it also handles mass shooting by using a timer

void timerEvent(QTimerEvent *) override; //checks if the ^, v, <, or > keys have //been clicked and if so it moves the player 10 pixels in that position

Bomb/invincibility features built-in to Player

Obstacle

This is where the basic functions of an obstacle are created!

Contains objects of kind:

- QMediaPlayer //assigns the sound effect for the obstacle
- int //sets the volume of the music

Important Functions:

virtual void move(); //used to force each derived class to contain a move
//function

void play_sound_effect(); //used to play and end the sound effect

void set_move_timer(QTimer* timer); //slot that connects the timer when it
//times out to the move() function

Enemy

This is where the enemies are created and their actions defined!

Contains objects of kind:

- QTimer //begins the enemy spawning
- QPixmap //used to create the image of the enemy
- Game //the object that takes in the specific game being played so specific //images and behavior will be used

Important Functions:

void update_graphics(); //a slot that updates image when timer goes off to //create flapping motion

void dead_graphics(); //once hit the image is changed to a dead image

void move(); //checks if the object has been collided with and if so deletes the //object

Health

This is where the player's health is kept track of!

Contains objects of kind:

- int //keeps track of the players health

Important Functions:

void decrease(); // used to decrease the health once hit

void increase(); //used to increase the health once a heart has been hit

int getHealth(); //returns the current health so that it can be displayed to the
//screen

Boss

This is where the final boss is created! !

Contains objects of kind:

- BossHealth //keeps track of the boss's health and how much damage he has received

Important Functions:

unsigned short int decrease_health(unsigned short int amount); //when the //boss is hit this updates the the health bar and shrinks it

unsigned short int get_health() const; //will return the health if the boss to //declare whether he is dead yet or not

void sprayAttack(); //slot that is called after a certain amount of time goes and //releases fireballs in every direction

MusicToggleButton

This is where music is played and muted!

Contains objects of kind:

- QMediaPlayer //used to add the selected mp3 to the class
- bool //checks if sound is on

Important Functions:

void toggle(); //creates a toggle button by adding a pixmap and sets the icon
//and resizes it and checks if the button has been clicked and if so, mutes the
//sound and changes th pixmap

Difficulties with the Game

- Checking for memory leaks and making sure everything has been destructed, while avoiding multiple destruction
- Ambiguity of functions in Qt, not really knowing how and what they do
Ex: a paint event, collision handling
- Understanding each other's code
- Learning how to read documentation
- Deciding how to break up tasks at the beginning because we did not know what we are capable
- Continuously trying to keep each other updated on each other's code and merging files accurately

Avatar:

The Last Airbender

Qt Demonstration

That's it!

Thank you for listening!

