MineBike Game Documentation

Andrew Weller & [your name here]

July 2019

Contents

0   This Manual

Hello anonymous developer! Welcome to the MineBike project. If this is not your desired destination you may leave this document now.

This manual is intended to inform the user about the mineBike GAME portion of the code. This manual does not relate to the middleware or the database portion. If you are looking for documentation on that, you should search elsewhere.

If you are not already aware of what the project is, mineBike is a minecraft mod designed as an alternative to traditional physical therapy for quarantined hospital patients who are unable to go outside to excercise.

Please refer to the table of contents if you wish to find what you are looking for quickly. Otherwise reading this manual in chronological order should get you up to speed with the game with zero prior knowledge.

1 Setup

Step 1: Clone Repository

```
$ git clone https://github.com/andrewwellercs/mineBikeCopy
```

Step 2: Unzip Forge Source

Unzip "MinecraftMod/Forge Source/forge-1.7.10-10.13.4.1492-1.7.10-src.zip" into "MinecraftMod/".

Be sure to overwrite any files.

Step 3: Run gradle scripts

Mac/Linux:

```
$ ./gradlew setupDecompWorkspace --refresh-dependencies
```

```
$ ./gradlew eclipse
```

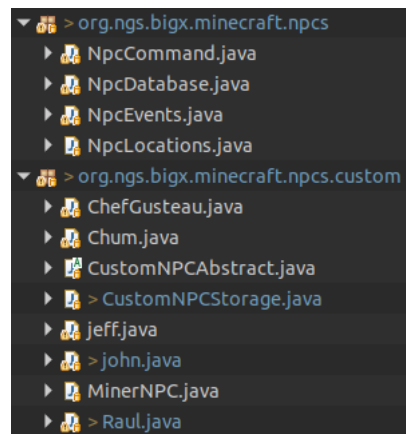This will create an eclipse workspace in the MinecraftMod/ directory.

Step 4:

2 Codebase Overview

The Codebase looks very big, but there are only a few very important parts to it. Most of the code is either not necessary for manipulating the game or is deprecated. Let's take a look at some important packages.
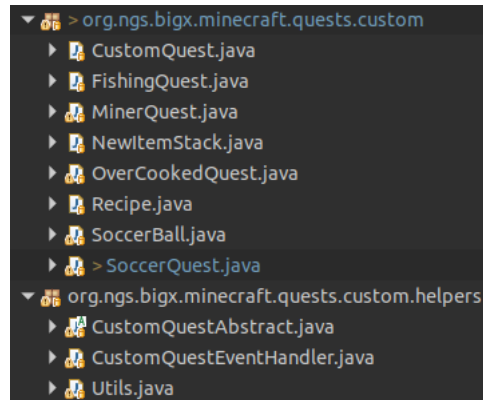
2.1 org.ngs.bigx.minecraft.npcs

This package is largely responsible for the creation of NPCs in the overworld map. Creating npcs in quests is another story and you can learn about it in the npcs section.

Noteable classes in here include **CustomNpcAbstract** which combined with a simple addition to **CustomNpcStorage** is a one-stop shop to adding simple npcs into the game. More on that in the npcs section.

2.2 org.ngs.bigx.minecraft.quests(.custom)

This package is responsible for the "quests" or "minigames" that the game has, excluding the original ChasingQuest which was in the game. The questing system is a modular way to build minigames.

2.3 org.ngs.bigx.minecraft.client.gui.hud
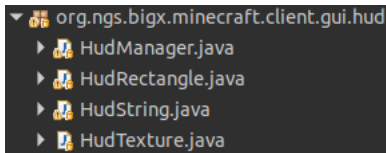
This package is very short and specialized. The tools in here can be used to add HUD elements to the game with no knowledge of OpenGL.

More on the usage of this package can be found in the Hud section.

3 Npcs

You might wonder why npcs are first in this guide and not quests. The answer is simple. Npcs are easier to add to the game. So to get your feet wet it is recommended to do this first. Also every quest usually has an npc attached to it. That is the pattern for this game. You start quests by talking to npcs. So once you have created your own npc you can then attach a quest to them and start testing that in the next section.

3.1 Simple Npcs (overworld)

Adding Npcs to the overworld is easy.

Figure 1: A custom NPC made using the CustomNpcAbstract class

You can add your own npc by creating a new class the extends the class **CustomNpcAbstract**. Located in the package **org.ngs.bigx.minecraft.npcs.custom**.

```
1  package org.ngs.bigx.minecraft.npcs.custom;
2
3  import org.ngs.bigx.minecraft.npcs.NpcDatabase;
8
9  public abstract class CustomNPCAbstract
10 {
11     //Three properties of an NPC
12     protected String name;
13     protected Vec3 location;
14     protected String texture;
15
16     //important that this method is called after the instantiation of the object
17     public void register()
18     {
19         NpcDatabase.registerNPC(name, location);
20     }
21
22     public abstract void onInteraction(EntityPlayer player, EntityInteractEvent event);
23
24     public Vec3 getLocation() {return location;}
25     public String getName() {return name;}
26     public String getTexture() {return texture;}
27 }
```

Figure 2: The CustomNpcAbstract class

There are three protected variables inside this class that should be set in the constructor in order for this class to work as intended.

```
protected String name;
```

This variable will change the display name of the npc in game. It is also used as an identifier in other pieces of the code so it should be **unique** and not overlap with any other npcs that exist either in the **CustomNPCStorage** class or the **NpcDatabase** class, located in org.ngs.bigx.minecraft.npcs.custom and org.ngs.bigx.minecraft.npcs packages respectively.

```
protected Vec3 location;
```

This variable will change the location that the npc spawns at in the overworld. It should be created with the Vec3.createVectorHelper method.

```
protected String texture;
```

This variable will change the texture that the npc has in the game. It should be a file location of an image. It technically shouldn't matter where

8

the image is located but sometimes it doesn't work if the images is in a weird folder. In order to guarantee that it works it should be located somewhere in the src/main/resources/assets/customnpcs/ folders, by convention all the skins for customnpcs have been placed inside of the .../customnpcs/textures/entity/ folders

The variable must be set in a certain way. In order to set the texture to a texture in customnpcs/textures/entity/humanmale/ the texture variable should be set as follows:

```
customnpcs:textures/entity/humanmale/Example_Texture.png
```

You can find new textures online by searching for "minecraft skins" online. Note that you will have to find one that is in the right format for the version of the game this mod is using - 1.7.10.

Here is an example of an Npc made using this formula. You can also view this in the project as Raul.java in org.ngs.bigx.minecraft.npcs.custom

Figure 3: An example of an npc made with the CustomNpcAbstract

3.2   Advanced Npcs (quests/moving)

If you wish to add an Npc that *does something* into the game, continue reading.

4 Quests

5 Hud