Final Project Report

Table Of Contents:

Project: <https://github.com/Succinctzic/CamsGame>

Main “puppeteer”:

* CamsGame class within the project (Creates and changes the screens and loads the music)

Screens:

* GameOver (Opened when the user dies, user can exit the game or to the menu)
* Menu (Called when the game is first opened, and if the user wants to restart the game)
* Instructions (Called from the main menu for new users that need some help)
* Maps (Called from the main menu when the user wants to change the maps and music)
* PlayScreen (Called from the menu when the user wants to start the game)

Scenes:

* Hud (At the top of the playscreen, displays info like time, rounds, money, and HP)
* SelectTurret (Sets up the position and cost of the turret buttons)
* TurretInfo (At the side of the screen, is called when a specific turret is selected, displays info about the specific turret)

Entities:

* Ants (Sets the initial position, movement, and rotation of the ants)
* Bullet (Sets the velocity of the bullets and gives them the specific ant they are trying to hit along with how much damage it deals)
* Turret (Holds the fire rate, damage, range, and turret type)

Scratches:

Scratch 1: Raid

* <https://github.com/Stefan335/CCFBRaid>

Description:

* Created the initial side panel that allowed the user to select press on different images

Scratch 2: Menu

* <https://github.com/Stefan335/CCFBMenu>

Description:

* Creating a simple main menu by the use of tables, music, and text buttons

Scratch 3: Turrets

* <https://github.com/Stefan335/CCFBTurrets>

Description:

* Placing simple turrets on the screen at the click of a user

Scratch 4:

* <https://github.com/Succinctzic/CCFBSprites.git>

Description:

* First scratch ever, simple textures being drawn

Scratch 5:

* <https://github.com/Succinctzic/CCFBSprite>

Description:

* Has multiple sprites with movement

Program overview/introduction:

You and your family have decided to have a nice picnic on a Saturday evening when all of a sudden, a brigade of ants attempt to steal your food! Armed with bottles of Raid you must defend your family, and more importantly, your food! Place bottles of raid near the path the ants take to try to kill off these pesky ants and bugs hungry for your food.

Disclaimer/List of known bugs:

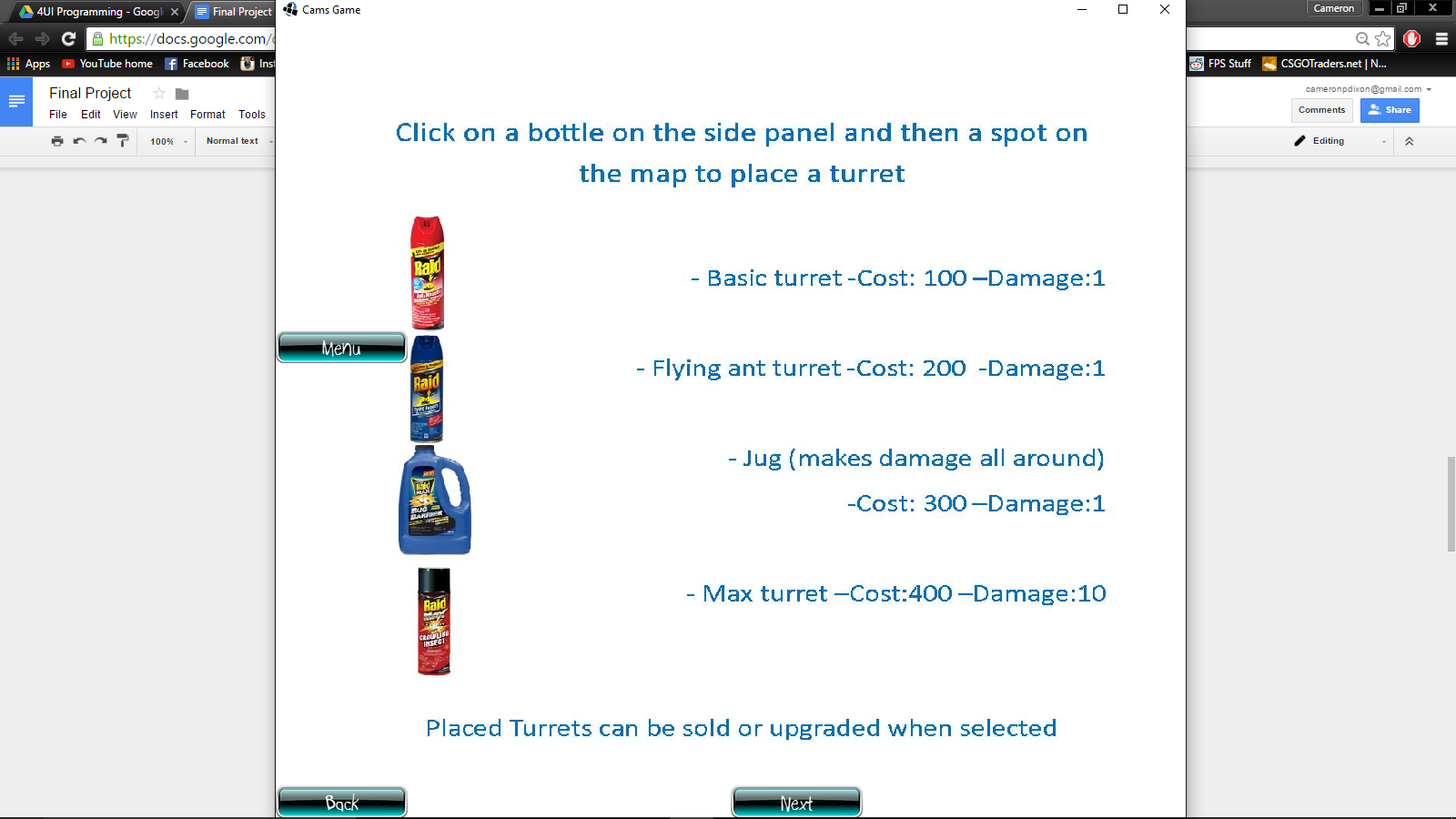
* Bullets for a specific ant stays on the screen when the ant reaches the end of the path and is removed
* Sometimes the selection of maps and music gets bugged when the game has been running for a long time
* First turret placed can be placed on the path
* When a specific turret on the map is selected, if you continue to click it the turret type selected will continue to switch between its previous position and the new position it was given
* Turrets will sometimes shoot at ants despite them being slightly out of the displayed range

User Manual:

* Once the game is loaded up you will be taken to the Main Menu screen
* There will be a total of five screens in the menu, “Play”: starts up the main game, “Instructions”: gives basic instructions on objectives, how the game is played and the stats of each ant/turret, “Maps/Music”: lets the user choose the music and map for when they play, “Levels”: allows the user to skip to different levels from the start, “Exit”: exits the game



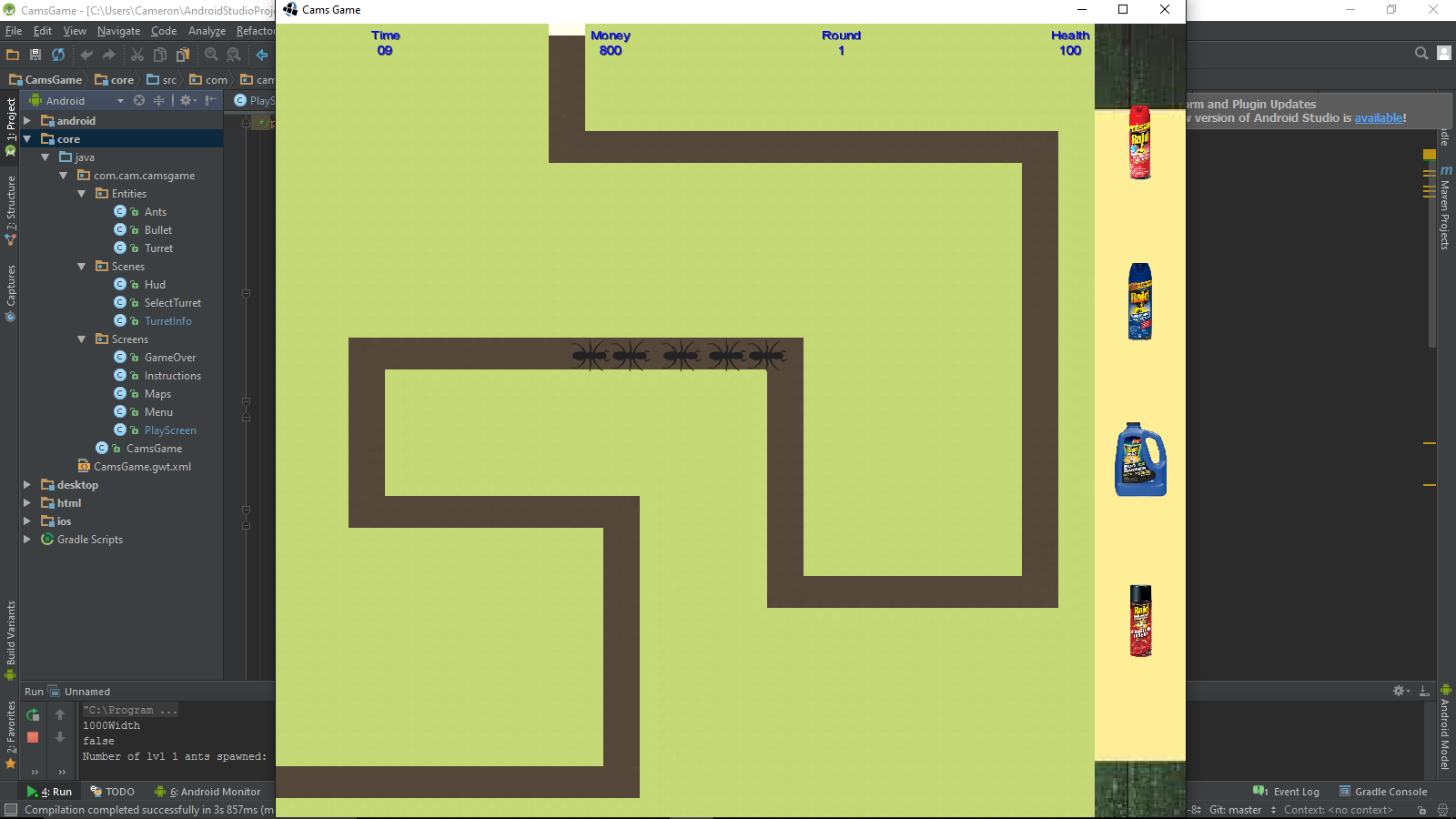
* Instructions: Pressing the “Next” or “Back” buttons cycles through the three pages of instructions, pressing the menu button will return the user to the menu



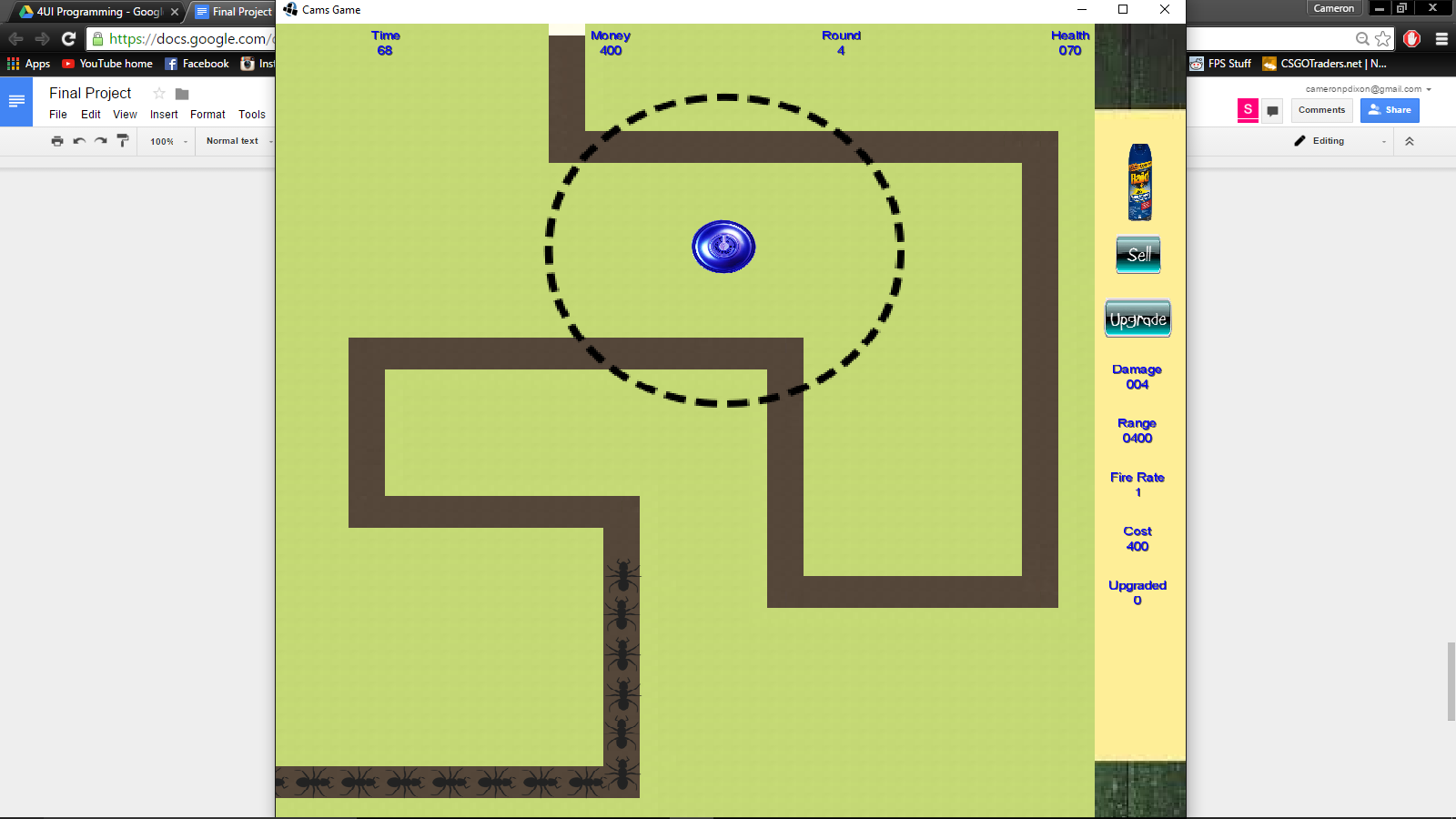
* Maps/Music: The bottom left side of the screen lets the user change the map for the game by pressing the “Change Map” button which cycles through a total of four maps while the bottom right side of the screen lets the user change the background music for the game by pressing the “Change Music” button which cycles through a total of four songs. The map/music chosen before leaving the screen will be the map/music loaded up after pressing play



* Levels: By pressing either the “Increase Starting Level” or “Decrease Starting Level” buttons the starting level will increase or decrease by a total of ten up to an infinite amount. The starting level will be displayed on the screen and the level displayed will be the starting level when the presses the “Menu” button and then the “Play” button to start the game
* Play: When the user is ready they will need to press the play button to start the game. Once the button is pressed the game will start immediately and enemies will begin to spawn. In the first round a total of five ants will spawn and as the round number increases so does the number of ants spawning at the start of each round



* Ants will follow along the path on the selected map until they reach the end. Once an ant has reached the end of the path the health (displayed in the top right corner) will decrease and the ant will disappear
* The health of the user starts at 100 but will go down for every ant that reaches the end of the path. Once the health reaches zero, the game is over
* The user must stop the ants from reaching the end of the path depleting the health. To do this they will need to place turrets on the map that will automatically shoot and kill ants
* To place a turret the user must press on one of the bottle of Raid on the right side panel (the selected bottle will be highlighted in red) and then press on a spot on the map for it to be placed. Turrets cannot be placed on the path. Each turret types have their own specific damage, range, firing speed, and cost



* Placing a turret will decrease the total amount of money that the user has which is displayed at the top of the screen. Each turret has a cost and the user will not be able to place turrets that they cannot afford. In the first round the total starting money will be $800. For every ant that is killed the money will increase
* Pressing on a turret that has been placed on the map will display that specific turrets stats, including its damage, cost, and fire rate which are also displayed in the instructions screen. There will also be a circle around the turret displaying its range and a sell and upgrade button. Pressing sell will remove the turret from the map and give back some of the money spent on it. Pressing upgrade will increase the stats of the turret selected for $5000. To deselect a turret press anywhere else on the screen
* Much like the turrets, different ants can have different stats such as damage to your health and speed. These stats are displayed in the instructions screen. As levels progress, more difficult ants will spawn. Killing more powerful ants will reward the user with more money



* A timer will be shown in the top left corner to see how long the user can survive the ant onslaught (in seconds)
* If the health of the user completely depletes to zero the game will end. This will load up the game over screen where the user has one of two options, exit the game in shame or return to the menu to see if they can try to beat their time and maybe try a different map



List of sources:

* <https://www.youtube.com/watch?v=D7u5B2Oh9r0>
* <https://www.youtube.com/watch?v=P8jgD-V5jG8>
* [http://gamedev.stackexchange.com/questions/60787/libgdx-drawing-sprites-when-movin](http://gamedev.stackexchange.com/questions/60787/libgdx-drawing-sprites-when-moving)-orthographic-camera
* <https://www.youtube.com/watch?v=7idwNW5a8Qs>
* <https://www.youtube.com/watch?v=NsxNE9uk1ew>
* <https://www.youtube.com/watch?v=DOpqkaX9844&index=4&list=PLXY8okVWvwZ0qmqSBhOtqYRjzWtUCWylb>
* <https://www.youtube.com/watch?v=q2qoiTqGsh8>
* <http://www.norakomi.com/tutorial_mambow2_music.php>
* <http://www.angelcode.com/products/bmfon>

Notes to future programmers:

* Use a different targeting system instead of just looking for the first one in the array that’s in the range
* Balance the game better. A balanced game is more fun than just destroying every ant that enters the screen
* Use better management of layouts and sprites
* Make it look nice
* Try doing less hard coding

Favourite programming technique:

Our favourite programming technique was switching screens. In order to do this first we had to create a new java class for our screen. That new class would then need to be declared as a screen within the CamsGame class which would act as our main puppeteer calling each of the classes. Within each of the screens when a button is clicked it calls a method within CamsGame which changes the current screen. Within each separate screen the buttons need to be disabled when that is not the currently displayed screen, this is because if the buttons are not disabled the user can still click them, which will cause fatal errors. Other features within the switching of screens included the movement of information between the seperate private screens through the median of which is the pupeteer, this was difficult at first, but through the use of methods within the CamsGame class, information is easily passed to methods in PlayScreen which can manipulate and use the information.