**Work in Progress Report #5**

**Major developments/breakthroughs(reference specific code please):**

Guests overlapping: Before, when the guest was dragged, the hitbox would overlap with another guest’s; therefore causing the guests to combine together and follow the cursor. To fix this, we moved the input onto the screen level and have the drag function only work on the most current guest that was last clicked on. We did this under the “touchDragged” function.

public boolean touchDragged(int screenX, int screenY, int pointer) {

if (arliGuests.get(nTarget).getBoundingRectangle().contains(vTouch)) {

arliGuests.get(nTarget).drag(vTouch, viewport);

}

Server, being able to serve drinks**:** The server, after walking to the first table can receive an order. After receiving an order, the server can walk to the bar, drop off the order and receive a drink 2 seconds later. They can pick up the drink and bring it to a table. It can’t drop it off yet. However, progress is visible.

public void carryDrink(SpriteBatch batch, boolean bHasOrder, int nClickedBar) {  
 this.bHasOrder = bHasOrder;  
  
 if (bHasOrder && nClickedBar == 1) {  
 if (arrived()) {  
 nTimer++;  
 }  
 }  
 if (nTimer >= 120) {  
 batch.draw(txtDrink, 300 + fW, 350, 50, 50);  
 if (nClickedBar >= 2) {  
 bHasDrink = true;  
 }  
 }  
 if (bHasDrink) {  
 System.out.println("HAS DRINK");  
 nTimer = 0;  
 batch.draw(txtDrink, fX, fY + 10, 50, 50);  
 }  
 }

**Major Challenges/setbacks(reference specific code please):**

Integrating guests :

Since we’re working on the same file (sprCustomer), some members of the team have to stay on standby before adding their component to the customers. For example, Sarah works on the guests’ ability to line up in a queue and then needs to pass the file to Maddie to add in “nTarget” code (code that’ll help isolate the guests when being dragged). Another member would be working on other components of the game that needs to be fixed/readjusted. Everyone has something to do but to have visual progress we need to finish up the new SprCustomer (previously was SprGuest) code. When we combined Sarah’s and Maddie’s code, we came into many new errors that we weren’t expecting.

**Any modifications to your specifications/release schedule:**

N/A

**Description of your scratch/test program:**

SctMultiGuest: The guests should walk down, and then they line up if not dragged. When dragged, you can place the guest on the table and it’ll animate the guest sitting, once you sit the guest the next guest in line should move down.

**Describe the generic concept you needed to test out:**

We needed to test the array list of guests and tables and how they will interact with each other. We had to combine the stacking code and dragging code together and see if both can function without any errors, and we’re now testing for the customer to update their status’ throughout the game (e.x. Ordering, then drinking, and then paying).

**Source any website/book that helped you with that concept:**

<https://stackoverflow.com/>

<https://libgdx.badlogicgames.com/ci/nightlies/docs/api/>

<https://github.com/DaphneLai/POLYGONE-Final/blob/master/core/src/gdx/objects/Shape.java>

<https://github.com/Ameer-Mushani/Sort>

<http://3ui.sgrondin.ca>

<https://libgdx.badlogicgames.com/ci/nightlies/docs/api/com/badlogic/gdx/graphics/glutils/ShapeRenderer.html>

**Describe the code and the lesson that you learned from it:**

N/A

**Describe any challenges that you enjoyed in integrating this scratch code into your major project:**

There are a lot of conditions that are nested and entwined together. It’s challenging to isolate the errors. But we’re learning how to find and troubleshoot such problems. When we combined all everyone’s cde we came into many errors that have not been easy to fix.

**Peer Assessment:**

Sarah: 100

Maddie: 100

Daphne: 100