Jiya Patel
Period 2

Jiya's Pizzeria

Description

My project is a replication of the game "Papa's Pizzeria" except with some modifications and additional features. The objective will be to get the most points you can by playing the game. You can get points by following the recipe on the receipt accurately and completing your pizza before the time runs out. Following the receipt instructions will include putting your pizza in the oven at the right temperature, for the right amount of time, putting the right toppings on, the right amount of toppings on, and more.

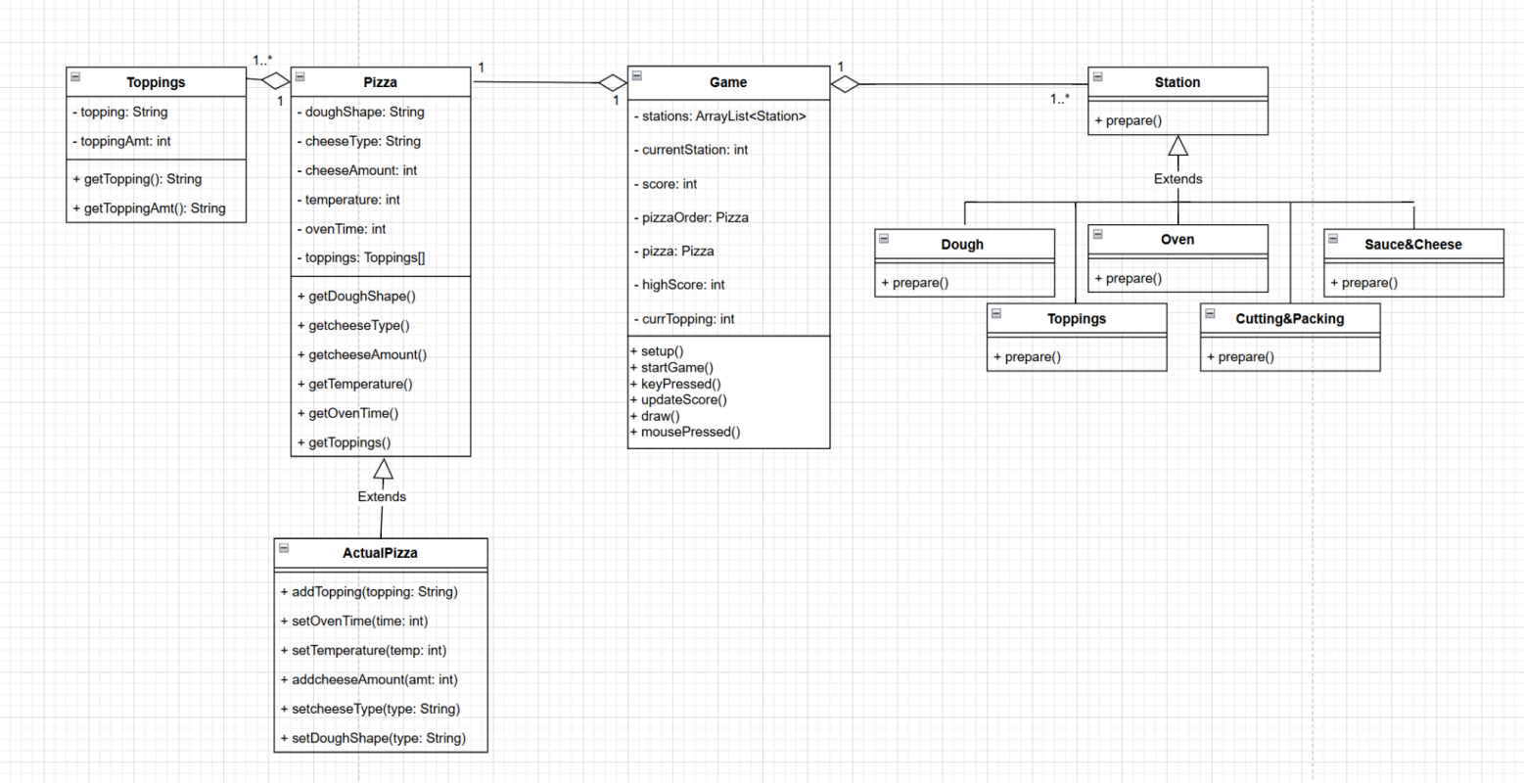
The libraries I will need for now will be Dashed Lines by Jose Luis Garcia del Castillo, which I will be using to allow the user to cut the pizza by holding down and dragging the mouse. Other than that, I will be using buttons, shapes, a display timer, and a high score display for multiple runs at the same time. I will also use the arrow keys to switch between the different stations in the game such as the dough-making station, the oven, the cheese/sauce station, toppings station, and cutting/packing station.

How Does it Work?

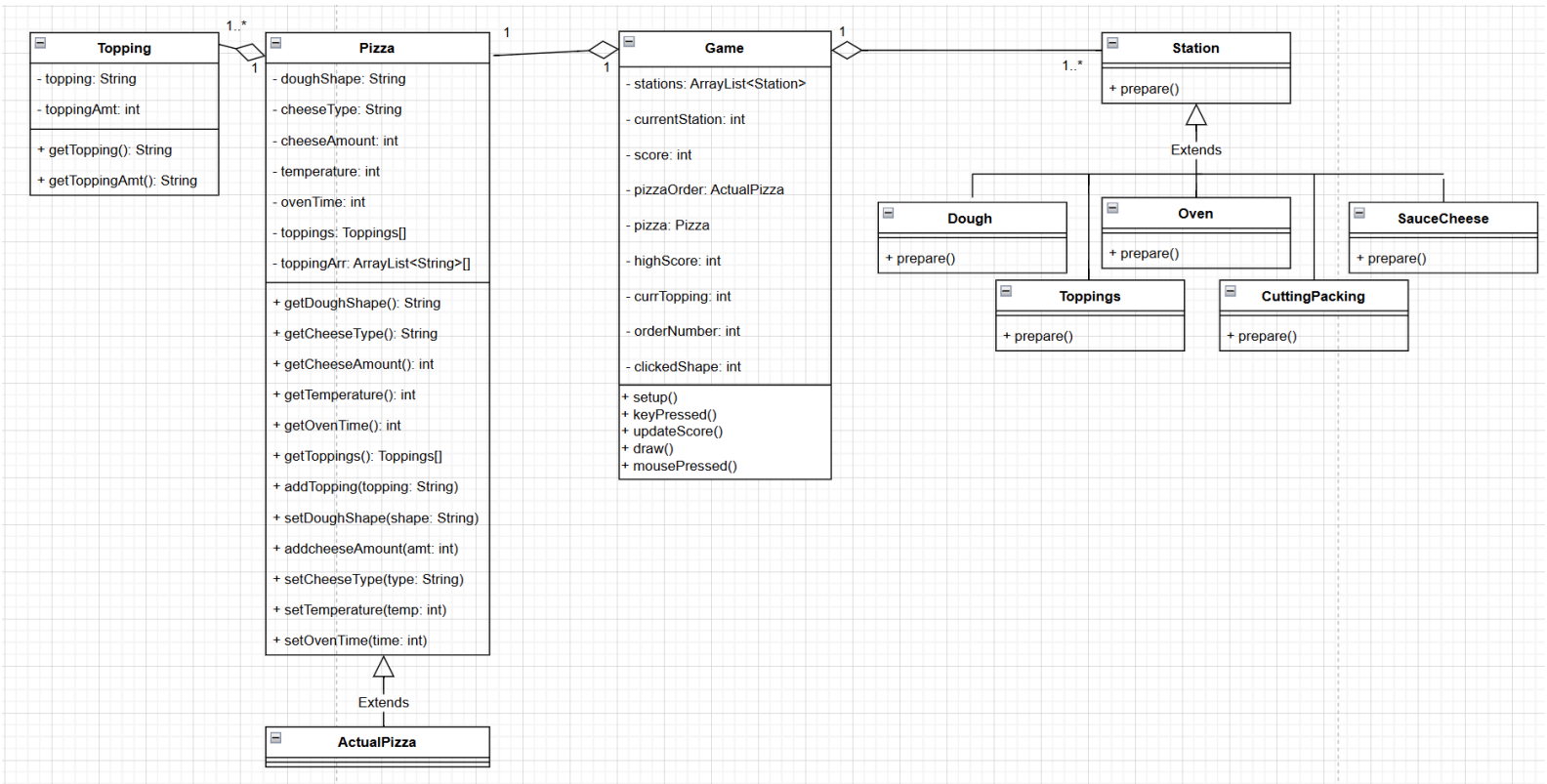
Objective: Finish making the pizza accurately in the given time + optimize the amount of points

1. User will press the "play" button and receive a receipt which lists their instructions/ingredients. Once they press "play," their timer will start and show up on the top of the screen
2. By clicking the right and left arrow keys, the user can switch between different parts of the pizza-making process, such as the dough-making station, the oven, the cheese station, toppings station, and cutting/packing station.
3. At the dough station, the user will draw a circle with their mouse and the pizza will form into that circular shape.
4. At the oven station, there will be a temperature button where the user has to adjust the temperature to be the right temperature. Then the pizza will go on the oven and the user has to cook it for the right amount of time. If the user cooks it for a longer time or at a higher temperature, points will be deducted from their final score. Points will also be deducted if the pizza is in the oven for less time than it needs or a lower temperature.
5. At the cheese station, the user will sprinkle the amount of cheese that the receipt says to put, and the type of cheese, whether it's normally grated or chunks of mozzarella (such as for a Margarita Pizza).
6. At the toppings station, the user will put toppings based on what the receipt says (pepperoni, mushrooms, basil leaves, pineapples, olives, onions, green peppers, etc). The user will have to click the right bowl of toppings and put them evenly throughout the pizza. Points will be deducted if the wrong toppings are put and also if the wrong amount is put (the amount is specified on the receipt).
7. At the final station I will make the user cut the pizza using a dotted line animation feature that will draw the dotted line where and when the user's mouse is held down, and when the mouse is released it will cut it in that line. There will be an animation to pack the final pizza and the user can click a button that says "done." They will be scored and the high score will be updated if necessary.

Previous UML Diagram



Updated UML Diagram



Current Functionalities

- Play button to start game
- Dough station: selecting a pizza shape (either circular or sicilian)
- 20 minute timer counting down once game started
- Using arrow keys to switch between stations
- Randomly generating an order/instructions for each time a game is started

Next Meeting Functionalities

- SauceCheese Station: putting sauce on pizza, putting the cheese as well
- Oven Station: adjusting the oven temperature, implementing a timer for the oven
- Toppings: dragging and placing different toppings onto the pizza
- CuttingPacking: dragging a dotted line and releasing to cut the pizza
- Maybe try to implement the score calculation at the end of the game

Issues

1. Before, when the user clicked the circle first (which would draw the circle on the screen), but then clicked the square, both the circle and square would be there. This resulted in the circle and square layered on top of each other which would happen if the user changed their mind and clicked the opposite shape after clicking one of them. To fix this issue, I added the variable "clickedShape" and initialized it to 0. 1 would represent the circle and 2 for the square. I added a condition to check if the shape on the screen was already a circle or square, because then I wouldn't change it to be the same, but if it was something else, then I would change the clickedShape variable and redraw it in the draw function, thereby replacing the old one and making for a cleaner, more functional look.
2. My buttons wouldn't work when clicked but I realized it's because draw() will overwrite it every time it runs since it runs every single second and so whatever I'm doing that I want to put on my screen as a constant would have to be done outside of my mousePressed if statement.
3. Another major issue was the countdown. At first I used seconds(), minutes(), and hours() but realized that this was for the actual time and not for your own custom countdown. So I did more research and used millis() and also discovered nf() which adds leading zeros to make sure the time on the countdown made sense and looked like a traditional clock down. But then, the numbers would layer on top of each other, so I added a rectangle that would go on top of the old one each time in the draw() function in the Game class.
4. Also, when switching the current station the user is on through use of the arrow keys, I had originally copy pasted a snippet of code from my kernels project, which I then realized was different from mine since I didn't want the user to loop backwards or forwards around but instead only go in a linear direction either back or front. Also, I added a condition to deal with the out of bounds error I was getting as a result of modifying my code.
5. Another issue was sizing the background for the Dough station (the wood counter background) to fit the screen since I kept getting the error that the background image Im importing must be the same size as screen. I found an image that would proportionally enlarge to my screen size, adjusted the screen size a little to match, and put it in a picture-enlarging generator to fix the issue.