Mauricio Rodriguez

1) Feature Description:

I am in charge of making the food and point system for the game Monolith. Our game is a restaurant focused experience that will place the player through challenging levels. For this game I am incharge of making the cooking feature for the player. Along with this that point system because they seemingly tied in together well. The player will have to focus on making sure the food is what the customer asked for and that it does not burn. Along with this based on that the points will be rewarded to the player.

2) Use Case Diagram:
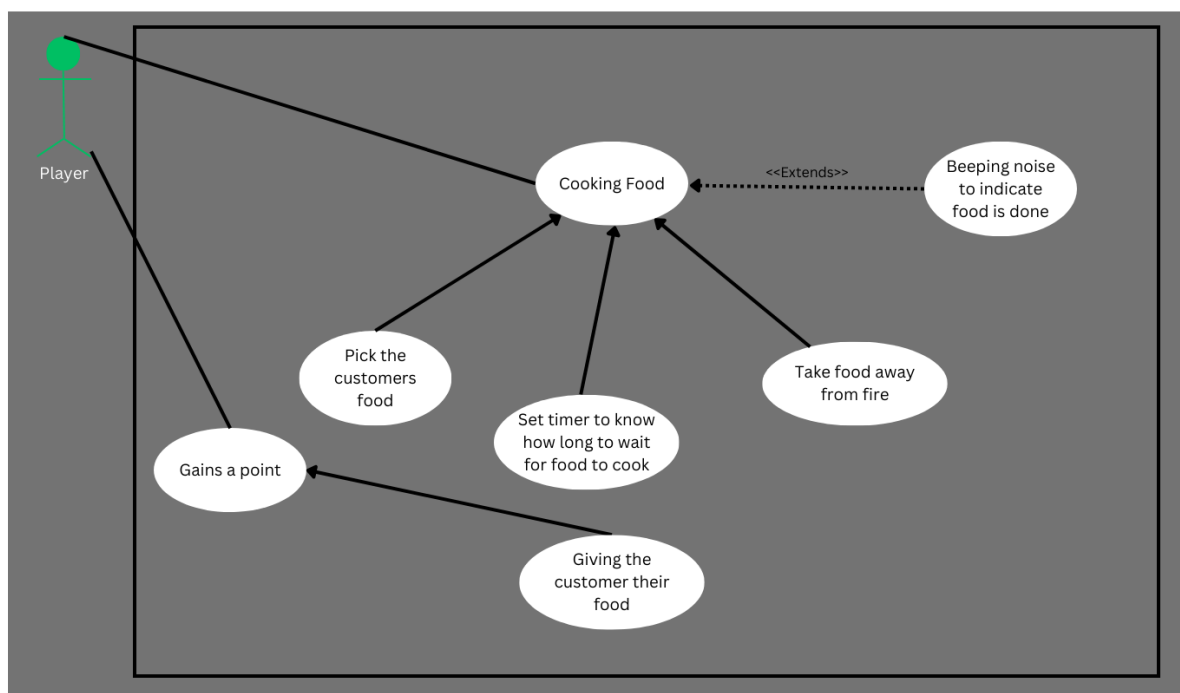
Scenarios:

Name: Cooking customers' food.

Summary: The player will cook the food desired by a customer in the game and will receive points for doing so.

Actors: The player

Preconditions: The player must have received an order from the customer

Basic Sequence:

1. Player seats the customer.

2. The customer then gets the customers order.
3. The player walks into the kitchen and selects the type of food the customer asked for.
4. They throw it into the fire and wait until the timer clocks down or until they hear a ding.
5. Then they take the food out and take it to the customer.
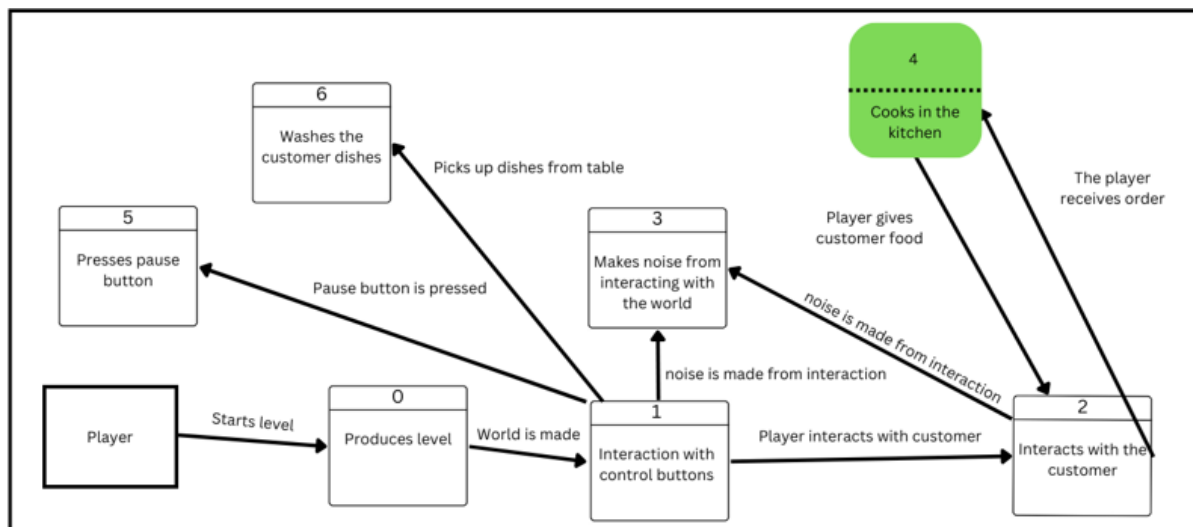6. The player then receives points for their service.
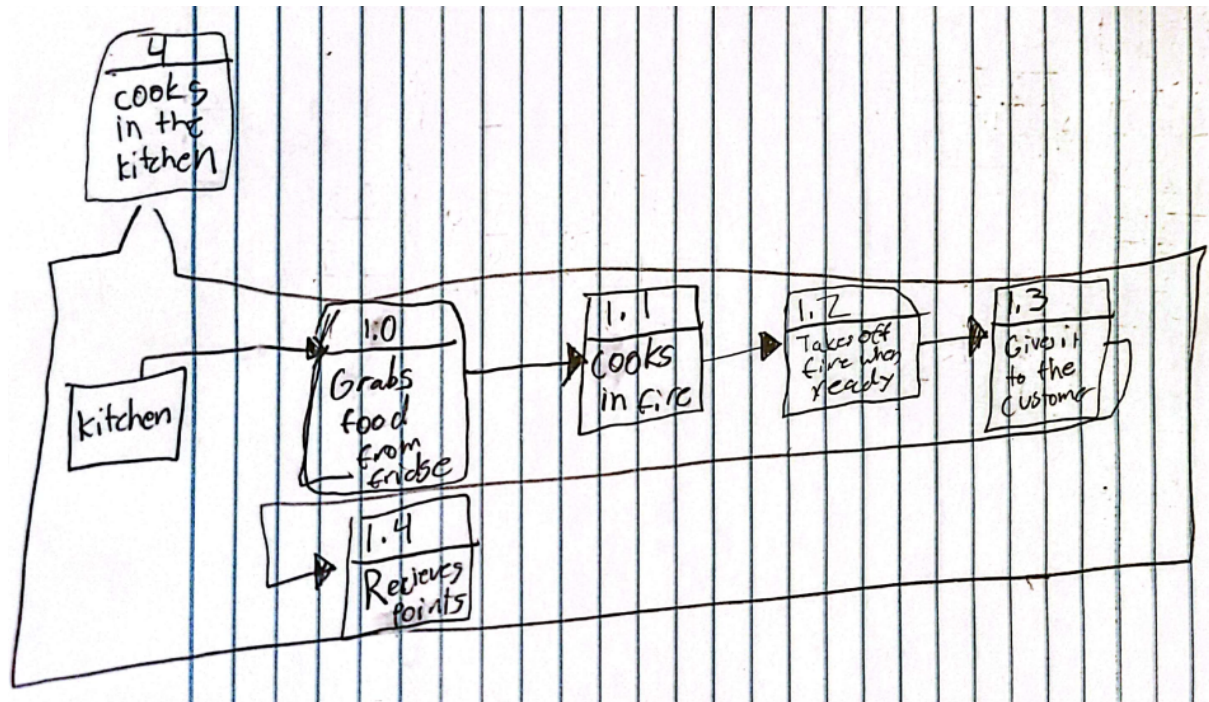
Exceptions:

Post Conditions: The player is satisfied and leaves a good review.

Priority: 1

ID: M01

3) Dataflow:

Process Description: When the player chooses to cook they enter the kitchen. From the kitchen they must find food in the fridge that is the food order they received from the customer. Then the player cooks it in the fire. Then the customer takes the food to the customer. Then the player receives points for giving the customer their food.

4) Acceptance test:

This test the input from the player when giving the customer their food.

Input	Output	Notes
'E'	Thank you from the customer.	The player is holding a plate In this case the player also gets points from giving the customer food.
'E'	You do not have this customer's order.	The player is holding nothing
'E'	You have uncooked food. You cannot give this to a customer	Have uncooked food
'E'	The customer does not want burned food.	Have burned food

5) time:

Tasks	Duration(hours):	Predecessors Tasks
1. Kitchen creation	5	- -
2. Fire creation	3	1
3. Fridge creation	3	1
4. Food creation	2	3
5. Food spawnable food	3	4
6. Create a trash can	2	1
7. The creation of the cooking along with the visual timer for food	4	5,2
8. Score different amounts of points with each food item.	3	4
9. Allow the food to be intractable with the player	2	8
10. Allow the fire to accept the food to be in the fire and know when it is done so it does not accept it again.	3	2,7,9
11. Allow the trash can to delete burned or unwanted food.	2	6
12. Point tracker	3	10
13. Let the player obtain the points associated with the food.	2	12,11
14. Allow the food to be eaten by the customers so it disappears.	3	13

Gantt chart (spreadsheet)

	Work	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Job #																																									
1		▓	▓	▓	▓	▓																																			
2							▓	▓	▓																																	
3										▓	▓																															
4												▓	▓																													
5														▓	▓																											
6																▓	▓																									
7																		▓	▓																							
8																				▓	▓																					
9																						▓	▓	▓																		
10																									▓	▓																
11																												▓	▓	▓											
12																															▓	▓	▓								
13																																			▓	▓	▓				
14																																							▓	▓	▓

Network diagram (hand-drawn, activity-on-node)

Node (activity 1): 0 | 5 | 5 — 0 | 0 | 5

Node (activity 6): 5 | 2 | 7 — 5 | 2 | 7

Node (activity 3): 5 | 3 | 8 — 5 | 0 | 8

Node (activity 11): 10 | 2 | 12 — 10 | 0 | 12

Node (activity 2): 9 | 3 | 8 — 5 | 0 | 8

Node (activity 4): 8 | 2 | 10 — 8 | 10 | 110

Node (activity 8): 10 | 3 | 13 — 10 | 0 | 13

Node (activity 5): 10 | 3 | 13 — 10 | 0 | 13

Node (activity 9): 13 | 3 | 16 — 13 | 0 | 16

Node (activity 7): 13 | 4 | 17 — 13 | 0 | 17

Node (activity 10): 17 | 3 | 20 — 17 | 0 | 20

Node (activity 12): 20 | 3 | 23 — 20 | 0 | 23

Node (activity 13): 23 | 2 | 25 — 23 | 0 | 25

Node (activity 14): 25 | ? | ? — 25 | 0 | ?