# 1. Brief introduction __/3

The function is the cutting of the pizza so when the player clicks/taps and drags the pointer over the pizza to the other side it looks like you cut the pizza.

# 2. Use case diagram with scenario  __14

### Use Case Diagrams



### Scenarios

**Name:** Cutting the pizza

**Summary:** The player chooses the line that cuts the pizza.

**Actors:** Player

**Preconditions:** There is a pizza to be cut, and the player is on the cutting screen.

**Basic sequence:**

> **Step 1:** Accept input form player for first point.
>
> **Step 2:** Accept input form player for second point.
>
> **Step 3:** Calculate the line of cutting.
>
> **Step 4:** Check if line is over pizza or cross another line.
>
> **Step 5:** Submit Pizza

**Exceptions:**

> **Step 4.1:** The line is not over the pizza.
>
> **Step 5.1:** No image is overlayed.
>
> **Step 4.2:** The line is over the pizza but doesn't cross a line.

**Step 5.2:** Overlay an image to summation the cut in pizza

**Step 4.3:** The line crosses another line and is over the pizza.

**Step 5.3:** Overlay an image to summation the cut in pizza.

**Step 6.3:** Make the line intersection look like it got cut by both, not just 2nd line covers up the 1st line.

**Post conditions:** The pizza cut is displayed and Submit Pizza.

**Priority:** 1
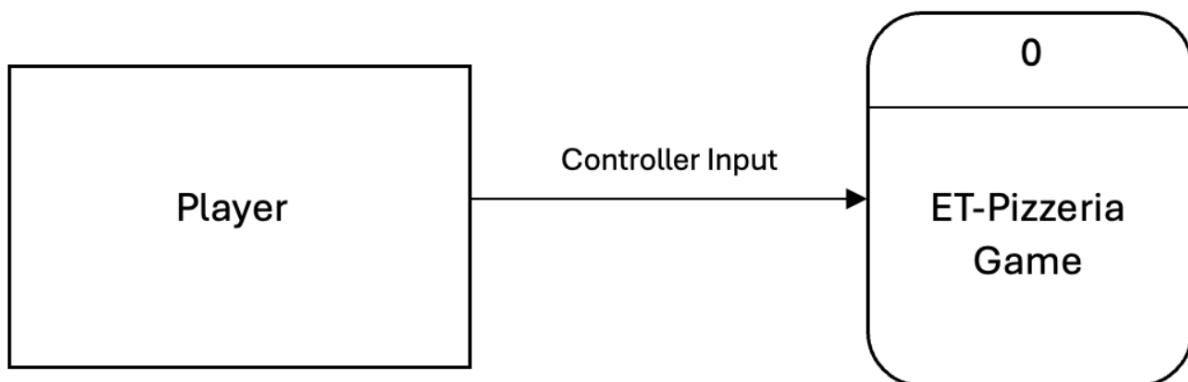
**ID:** C01

*The priorities are 1 = must have, 2 = essential, 3 = nice to have.
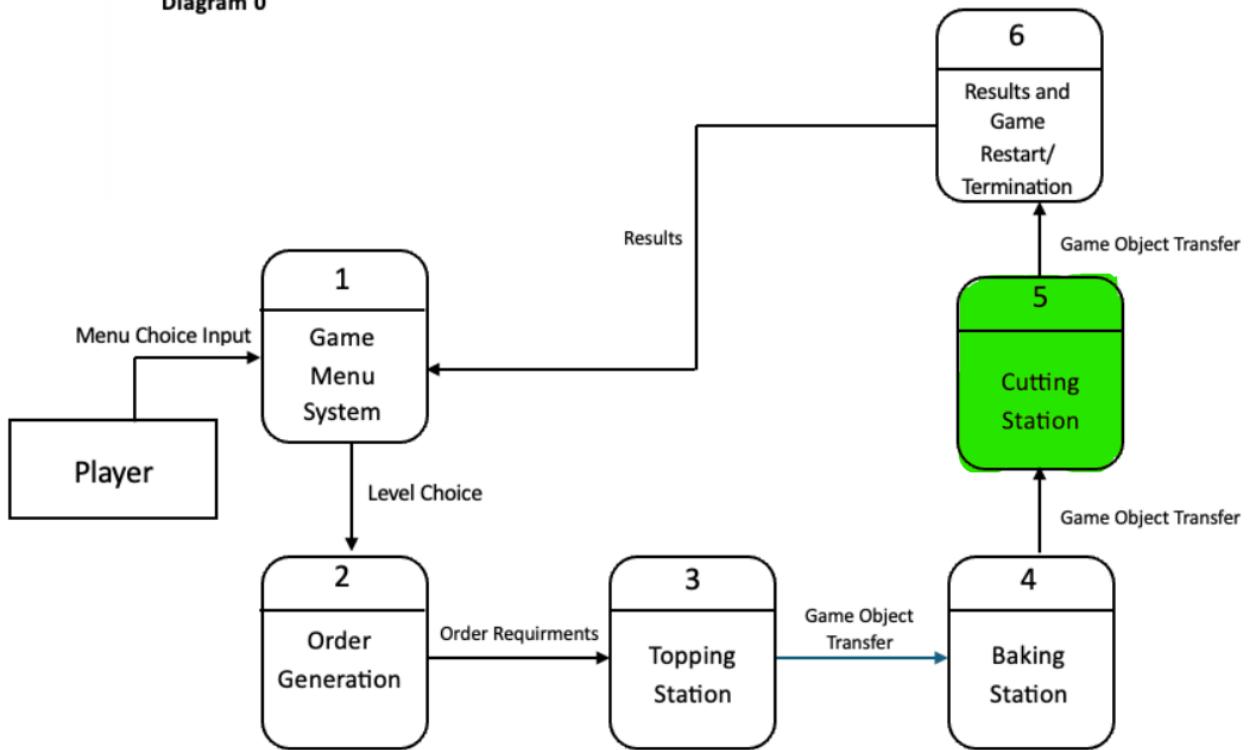
## 3. Data Flow diagram(s) from Level 0 to process description for your feature _____14
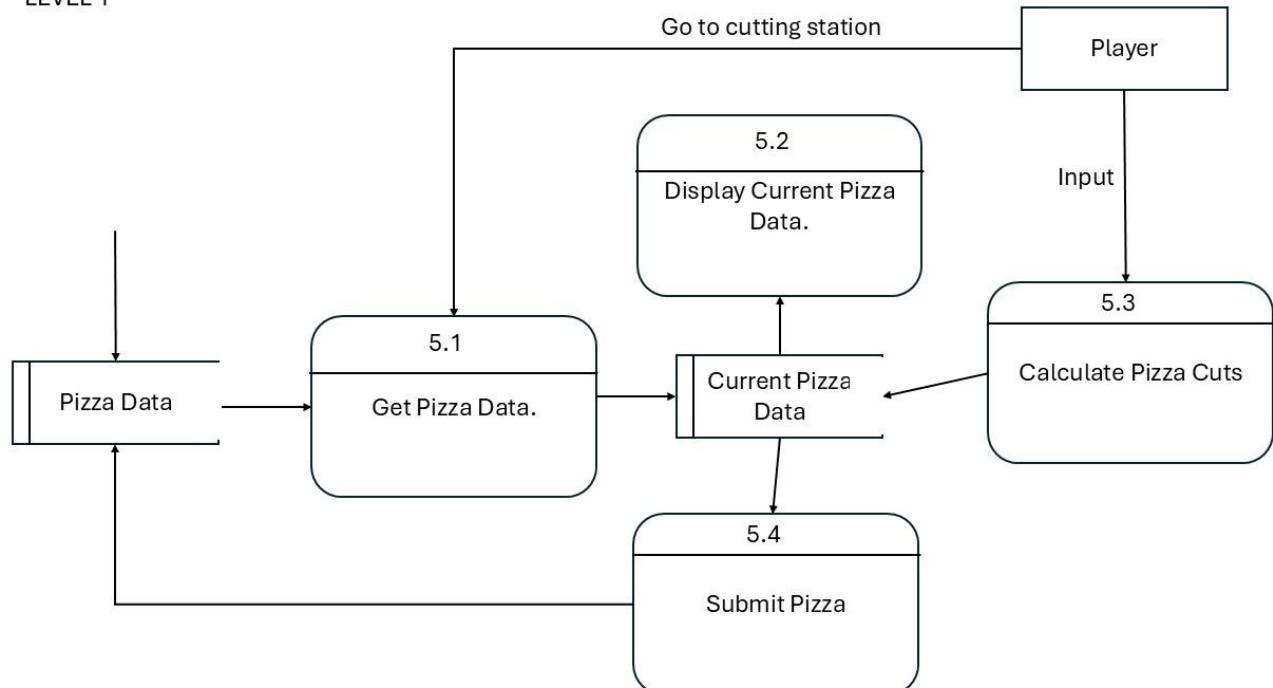
### Data Flow Diagrams

**Diagram 0**

**6** — Results and Game Restart/ Termination

**1** — Game Menu System

**5** — Cutting Station

**2** — Order Generation

**3** — Topping Station

**4** — Baking Station

Player

Menu Choice Input

Results

Game Object Transfer

Level Choice

Game Object Transfer

Order Requirments

Game Object Transfer

**LEVEL 1**

Player

Go to cutting station

Input

**5.2** — Display Current Pizza Data.

**5.3** — Calculate Pizza Cuts

**5.1** — Get Pizza Data.

Current Pizza Data

Pizza Data

**5.4** — Submit Pizza

**Process Descriptions**

Manage Cutting Station*:

IF pizza AND player are on the cutting station

Get input from player

IF cut line is on pizza

Add cut line to pizza

IF line cross another cut line.

Add intersection to pizza

ELSE

Don't add intersection to pizza

END IF

ELSE

Don't add cut line to pizza

END IF

ELSE

Don't get user input

END IF

## 4. Acceptance Tests _____9

**Cutting Pizza feature**

Try to cut the Pizza 100 times

We are testing if:

The game doesn't crash.

Add cut line to pizza when it should.

Don't add cut line to pizza when it should.

Add intersection when it should.

**Example for divided feature**
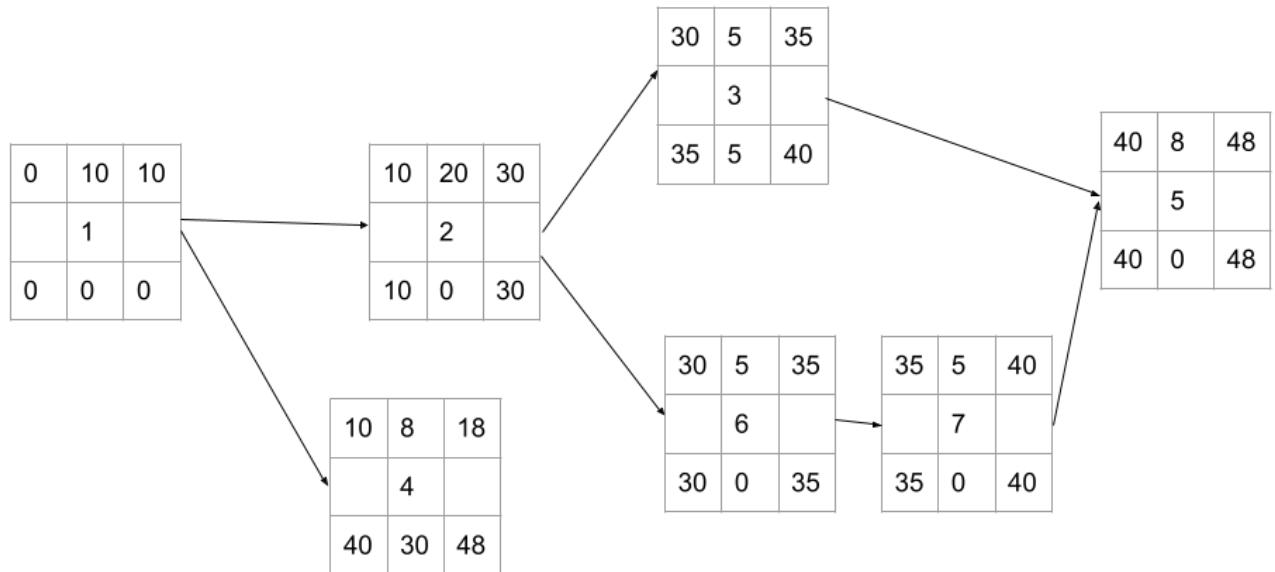
| Input | Pass Tests? | Notes |
|---|---|---|
| The line is over the pizza. | T | Add a cut line to the pizza |
| The line crosses over another cut line. | T | Add a cut line and intersection to the pizza |
| The line is not over the pizza. | T | No change in animation |
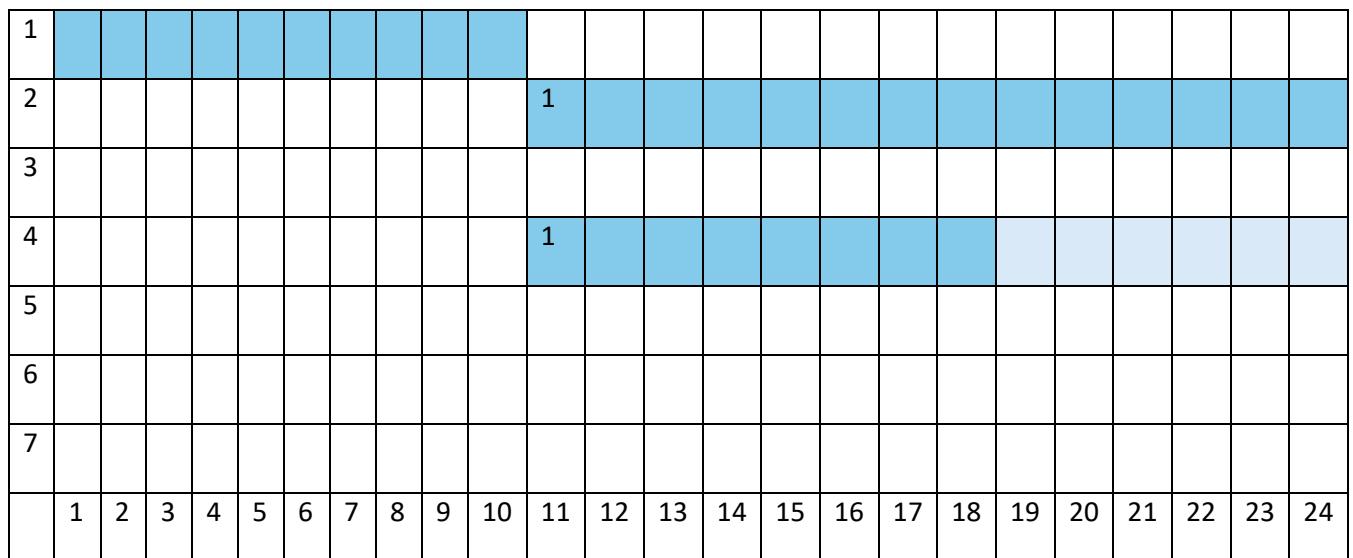
# 5. Timeline _____/10
## Work items

| Task | Duration (Hours) | Predecessor Task(s) |
|---|---|---|
| 1. Cutting Station Design | 10 | - |
| 2. Programming | 20 | 1 |
| 3. Testing | 5 | 2 |
| 4. Artwork | 8 | 1 |
| 5. Polish | 8 | 3, 7 |
| 6. Documentation | 5 | 2 |
| 7. Installation | 5 | 6 |

## Pert diagram



| 0 | 10 | 10 |
|---|----|----|
|   | 1  |    |
| 0 | 0  | 0  |

| 10 | 20 | 30 |
|----|----|----|
|    | 2  |    |
| 10 | 0  | 30 |

| 30 | 5 | 35 |
|----|---|----|
|    | 3 |    |
| 35 | 5 | 40 |

| 40 | 8 | 48 |
|----|---|----|
|    | 5 |    |
| 40 | 0 | 48 |

| 10 | 8  | 18 |
|----|----|----|
|    | 4  |    |
| 40 | 30 | 48 |

| 30 | 5 | 35 |
|----|---|----|
|    | 6 |    |
| 30 | 0 | 35 |

| 35 | 5 | 40 |
|----|---|----|
|    | 7 |    |
| 35 | 0 | 40 |

## Gantt timeline

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 2 |   |   |   |   |   |   |   |   |   |   | 1 | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| 3 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 4 |   |   |   |   |   |   |   |   |   |   | 1 | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ░ | ░ | ░ | ░ | ░ | ░ |
| 5 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 6 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 7 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

| | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | | | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | | | | | | | | | | |
| 3 | | | | | | | 2 | | | | | | | | | | | | | | |
| 4 | | | | | | | | | | | | | | | | | | | | | |
| 5 | | | | | | | | | | | | | | | | | 7 | | | | |
| 6 | | | | | | | 2 | | | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | 6 | | | | | | | | | |

| | 46 | 47 | 48 |
|---|---|---|---|
| 1 | | | |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |
| 6 | | | |
| 7 | | | |