**Use Case Specification**

**Use Case 1.  Rotate through Item Category**

**Description**: Rotate through menu items horizontally to select food category

**Primary Actors**: Player, Food Item Categories(Pizza, Cold drinks, Burger and Fries)

**Preconditions**: All food items added to the Menu Grid

**Basic Flow of events**:

1. User will use keyboard left, right button to rotate item categories
2. On each click the food item view will be updated
3. User can again click on the left, right button to perform the same case again

**Use Case 2.  Rotate through Item Type in a Category**

**Description**: Rotate through menu items vertically to select specific item from a category

**Primary Actors**: Player, Food Items(Pizza, Cold drinks, Burger and Fries)

**Preconditions**: All food items added to the Menu Grid

**Basic Flow of events**:

1. User will use keyboard up and down button to rotate items
2. On each click the food item view will be updated
3. User can again click on the up, down, left, right button to perform the same case again

**Use Case 3.  Track remaining time for the level**

**Description**: Remaining time for each level

**Primary Actors**: Player, Timer Clock

**Preconditions**: Level should be started

**Basic Flow of events**:

1. Timer will be started when each level starts
2. User should track the timer while playing the game
3. User will lose the game if timer is 0

**Use Case 4.  Select a food item to add it to plate**

**Description**: Select specific item from a category to add it to plate

**Primary Actors**: Player, Food Items(Pizza, Cold drinks, Burger and Fries)

**Preconditions**: All food items added to the Menu Grid

Desired food item at main cell

**Basic Flow of events**:

1. User will hit enter button to place the food item to plate
2. On each click the food item will be updated
3. User can again click on enter button to perform the same case again

**Use Case 5.  Complete the level**

**Description**: Compete the each level

**Primary Actors**: Player

**Preconditions**: Level should be started

**Basic Flow of events**:

1. Reshuffle the menu items in menu grid
2. Select the appropriate items to prepare the order
3. User will complete the level when all the items are added to the plate

**Use Case 6.  Compare player’s plate with the model plate**

**Description**: Comparing each food item from player’s food plate with each item of pre-build plate

**Primary Actors**: Developer, Food plates

**Preconditions**: All food items added to the plate

**Basic Flow of events**:

1. User will add all the food items one after other
2. If the time is not up and all the items has been added to plate required then each food items from both the plates are compared using iterator pattern
3. Developer will iterate through items from both plates if the number of items on both the plates are same.

**Use Case 7.  Identify selected item**

**Description**: Know which item to move to plate

**Primary Actors**: Developer

**Preconditions**: User should select the item from menu item grid

**Basic Flow of events**:

1. User will select the item from the menu item grid
2. Algorithm will check the selected item will the given pre-build order
3. If the selected item is there in the pre-build order plate then add item to given plate

**Alternate Flow of events**:

1. User will select the item from the menu item grid
2. If selected item is not in the pre-build output plate then don’t add item to the plate

**Use Case 8.  Move selected item to plate**

**Description**: Move selected food items to desired position on the plate

**Primary Actors**: Developer, Food item, plate

**Preconditions**: All food items added to the Menu Grid

Desired food item at main cell

**Basic Flow of events**:

1. User will hit enter button to place the food item to plate
2. Developer has to place the selected food item to desired position on the plate
3. User can again click enter to perform the same case again

**Use Case 9.  Decrement the timer**

**Description**: End game when timer expires

**Primary Actors**: Developer

**Preconditions**: Player should be playing the game and game should be in started state

**Basic Flow of events**:

1. Create timer which will track current level time
2. Reduce the time on every second
3. If the value of timer becomes 0 then “Game Over” screen should appear.

**Use Case 10.  Randomly generate items on model plate**

**Description**: Randomly selects  food items and food categories to create pre-build plate

**Primary Actors**: Developer, Food plates, Food items

**Preconditions**: All food items added to the Menu Grid

**Basic Flow of events**:

1. Developer has to randomly select food category
2. Randomly select food items from each selected category
3. Regenerated food plate for each level.