



---

# COMP1531 GROUP PROJECT

## “GOURMET”

---

Title Page



**Group Name:** XJ\_Aven\_Dheeraj

**Team Members/zID's:**

Aven Au (z5208734)

Dheeraj Viswanadham (z5204820)

Xingjian Li (z5190719)

APRIL 28, 2019

TERM 1

## **COMP1531 19T1 Project Milestone 1: Product Backlog**

### **Client Requirements:**

The client is interested in *developing an online self-service ordering application*. The client feels that such a system would *provide faster delivery of meals to customers, enable customers to build their own gourmet creations which would result in improving the revenue* at the different franchises of the food chain.

### **Problem Statement:**

Develop an online self-service ordering application (system), which *enables customers to build their own gourmet creations and enables staff to service online orders and maintain inventory*.

### **Epic Story 1:**

- **As a customer, I would like to be able to place an order of any available mains along with sides and drinks if I choose so**

ID	CUS1
Name	Create a burger
User Story Description	As a customer, I should be able to create a burger on the system so that I can have a customised meal according to my tastes.

Acceptance Criteria	<ul style="list-style-type: none"> <li>Customer is able to choose a preselection of burgers with: <ul style="list-style-type: none"> <li>different number and type of buns (e.g., 3 sesame buns for a double burger or 2 muffin buns for a standard single burger).</li> <li>Different number and type of patties (e.g., 2 chicken patties, vegetarian, beef).</li> <li>Other optional extras such as ingredients of their choice: tomato, lettuce, tomato sauce, cheddar cheese, swiss cheese etc.</li> <li>Choose Quantity of buns they want</li> </ul> </li> <li>Only single, double and triple burgers are permitted. The number of buns and patties must be within allowable limits: <ol style="list-style-type: none"> <li>The customer can only choose buns in the range of [2, 4]</li> <li>The customer can only choose patties in the range of [1, 3]</li> <li>The amount of buns and patties must be in a one-to-one relationship (ie. 2 bun-1 patty, 3 bun-2 patties, 4 bun- 3 patties)</li> </ol> </li> <li>If a particular ingredient is not available, the customer will not be able to select the particular item to their order.</li> </ul>
Priority (1 = Lowest and 10 = Highest)	10
Size (1SP = 1 Day)	2 SP

ID	CUS2
Name	Create wrap
User Story Description	As a customer, I can create a wrap on the system so that I can have a customised meal according to my tastes.
Acceptance Criteria	<ul style="list-style-type: none"> <li>Customer is able to choose a preselection of wraps with: <ul style="list-style-type: none"> <li>Different types of wraps sheets (e.g. wholemeal wrap, etc).</li> <li>Different type and number of patties (e.g., 2 chicken patties, vegetarian, beef).</li> <li>Other optional extras such as ingredients of their choice: tomato, lettuce, tomato sauce, cheddar cheese, swiss cheese etc.</li> <li>Choose the number of wraps they want</li> </ul> </li> <li>The number of wraps and patties cannot exceed the maximum allowable limit.</li> <li>Customer should be able to cancel the order.</li> </ul>
Priority (1 = Lowest and 10 = Highest)	10

Size (1 SP = 1 Day)	2 SP
---------------------	------

ID	CUS3
Name	Select any sides
User Story Description	As a customer, I should be able to order sides of my choice on the system, so that I can have a more fulfilling meal.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Customer is able to choose whether they want any sides, and if so: <ul style="list-style-type: none"> <li>◦ Select the type and quantity of sides (ie., a 6 pack nuggets, 3 pack nuggets, fries [a small fries = 75g, a medium fries = 125 g, a large fries = 250g], sundaes (Chocolate/Strawberry) [small, medium, large]).</li> </ul> </li> <li>• The number of sides cannot exceed the maximum allowable limit.</li> <li>• Customer should be able to cancel the order.</li> </ul>
Priority (1 = Lowest and 10 = Highest)	6
Size (1 SP = 1 Day)	2 SP

ID	CUS4
Name	Select any drinks
User Story Description	As a customer, I should be able to order drinks of my choice on the system, so that I can have a more fulfilling meal.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Customer is able to choose whether they want any drinks, and if so: <ul style="list-style-type: none"> <li>◦ They should be able to select the type and quantity of drinks as following: <ol style="list-style-type: none"> <li>1) Bottled drinks are stocked in either cans (375 ml) or bottles (600 ml)</li> <li>2) Drinks such as orange juice will be served in varying sizes (e.g., a small = 250 ml, a medium = 450 ml etc)</li> </ol> </li> </ul> </li> <li>• The number of drinks cannot exceed the maximum allowable limit.</li> <li>• Customer should be able to cancel the order.</li> </ul>

Priority (1 = Lowest and 10 = Highest)	6
Size (1 SP = 1 Day)	2 SP

ID	CUS5
Name	Self-ordering
User Story Description	As a customer, I should be able to self-order my gourmet creation on the system, so that I can order faster as I don't need to wait for the waiter.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Customer is able to: <ul style="list-style-type: none"> <li>◦ View details of their order (e.g. current ingredients selected and their respective quantities, number of mains or drinks they have ordered, etc).</li> <li>◦ View the base price of burgers and wraps, and see the additional price associated with each ingredient, allowing the customer to see the overall net-price of their order.</li> <li>◦ Select additional options (such as gluten-free, no onion/garlic)</li> </ul> </li> <li>• Add any in-store discounts available to their checkout cart.</li> <li>• Select if they want take-away.</li> </ul>
Priority (1 = Lowest and 10 = Highest)	7
Size (1 SP = 1 Day)	3 SP

ID	CUS6
Name	Select a Wrap
User Story Description	As a customer, I can choose a wrap on the menu so that I can quickly select my favourite meal.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Customer is able to choose a preselection of premade wraps with: <ul style="list-style-type: none"> <li>◦ Beef Wraps</li> <li>◦ Chicken Wraps</li> <li>◦ Veggie Wraps</li> </ul> </li> <li>• Customer should be able to cancel the order.</li> </ul>

Priority (1 = Lowest and 10 = Highest)	10
Size (1 SP = 1 Day)	2 SP

ID	CUS7
Name	Select a Burger
User Story Description	As a customer, I can choose a burger on the menu so that I can quickly select my favourite meal.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Customer is able to choose a preselection of premade wraps with: <ul style="list-style-type: none"> <li>○ Cheeseburgers</li> <li>○ Double Cheeseburgers</li> <li>○ Chicken Burgers</li> <li>○ Double Chicken Burgers</li> <li>○ Veggie Burgers</li> <li>○ Double Veggie Burgers</li> </ul> </li> <li>• Customer should be able to cancel the order.</li> </ul>
Priority (1 = Lowest and 10 = Highest)	10
Size (1 SP = 1 Day)	2 SP

### Epic Story 2:

- **As a customer, I would like to be able to checkout my order and check the order status at any time after the order has been completed**

ID	CUS6
Name	Order confirmation
User Story Description	As a customer, I should be able to check and confirm my order so that I can make sure I ordered the correct items I want.

Acceptance Criteria	<ul style="list-style-type: none"> <li>• The customer should be able to see all the contents they have ordered</li> <li>• If the customer disagrees with the contents, they can make changes appropriate to them</li> <li>• If customer orders more items charge them more</li> <li>• The customer should be able to view the price for all items and total price</li> <li>• A “confirm order” button at the bottom of order details, once the customer clicks the button, proceed to payment</li> </ul>
Priority (1 = Lowest and 10 = Highest)	5
Size (1 SP = 1 Day)	2 SP

ID	CUS7
Name	Checking order status
User Story Description	As a customer, I can check the order status so that I can see when my order is ready to collect.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Once the valid payment is completed, the customer should receive a unique order number at the specific store they ordered from.</li> <li>• Customer is able to enter the order number to see all the contents that were ordered and the order status:</li> <li>• If the order is not ready, then the order status will be “Your Order Is Currently Being Made”.</li> <li>• If the order is ready, then the order status will change to “Your Order Is Now Ready For Collection”</li> </ul>
Priority (1 = Lowest and 10 = Highest)	5
Size (1 SP = 1 Day)	2 SP

### **Epic Story 3:**

- **As staff, I should be able to view all customers’ orders live and update their order status**

ID	SUS1
Name	View Order Status
User Story Description	As staff, I should be able to view current orders status at any time so I can provide faster service to customers
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Staff is able to view all data on orders that are placed <ul style="list-style-type: none"> <li>○ Ingredients needed</li> <li>○ Meals that are being ordered</li> <li>○ Ingredients Left</li> </ul> </li> <li>• If the order is complete the order should disappear from the staff menu</li> </ul>
Priority (1 = Lowest and 10 = Highest)	7
Size (1 SP = 1 Day)	1 SP

#### Epic Story 4:

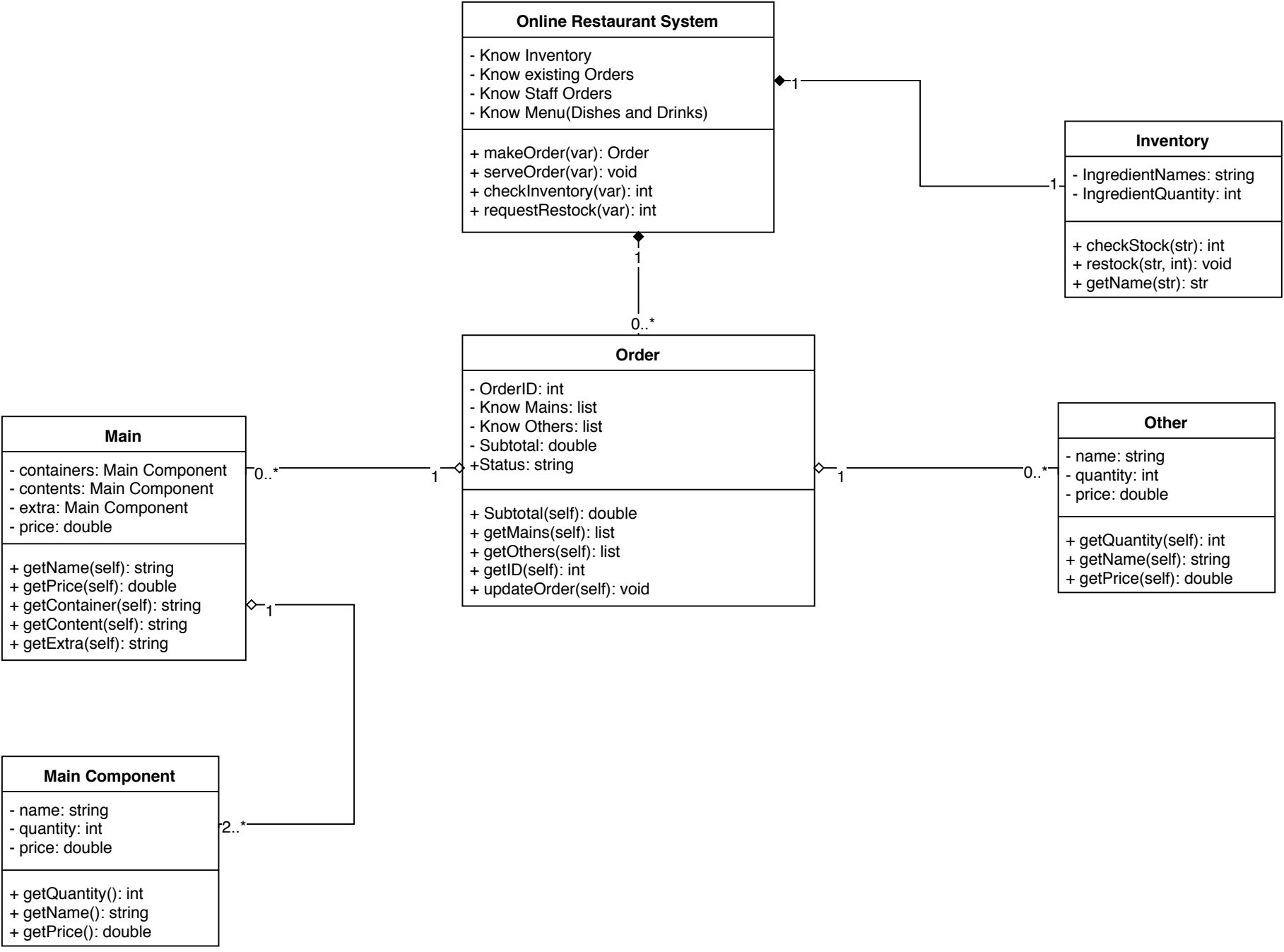
- **As staff, I should be able to view inventory situation at any time, and request for a restock when necessary**

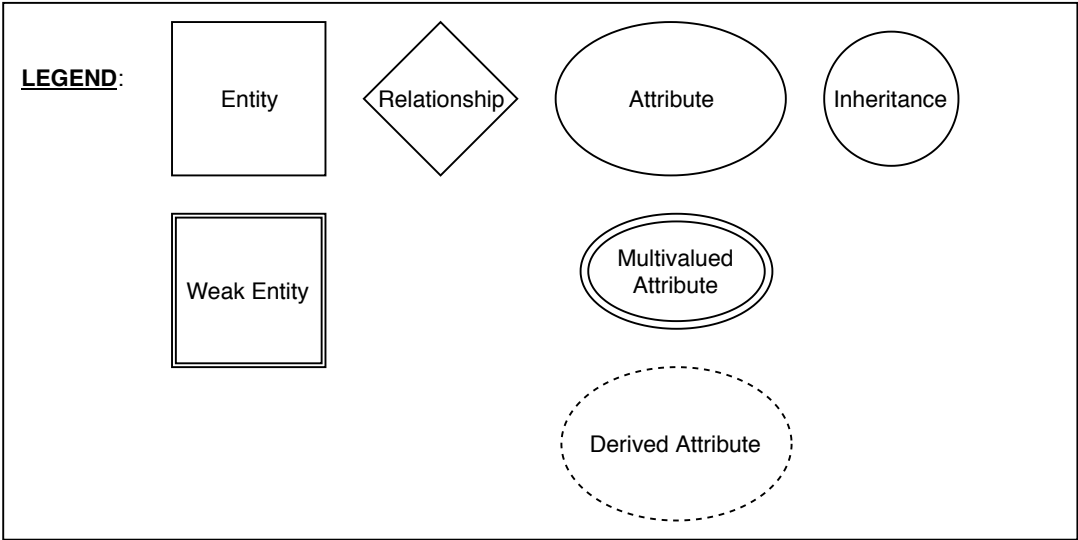
ID	SUS2
Name	Check Inventory
User Story Description	As staff I should be able to see how many items on the menu are left, allowing me to show customers whether a dish/drink is available.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Staff is able to view how much ingredients are left <ul style="list-style-type: none"> <li>○ <b>Ingredients</b> <ul style="list-style-type: none"> <li>■ Burger Buns</li> <li>■ Wraps</li> <li>■ Patties</li> </ul> </li> <li>○ <b>Drinks</b> <ul style="list-style-type: none"> <li>■ Bottled <ul style="list-style-type: none"> <li>• Cans (375ml)</li> <li>• Bottles (450ml)</li> </ul> </li> <li>■ Orange Juice</li> </ul> </li> </ul> </li> </ul>



	<ul style="list-style-type: none"> <li>• Small (250ml)</li> <li>• Medium(450ml)</li> </ul> <ul style="list-style-type: none"> <li>○ <b>Sides</b> <ul style="list-style-type: none"> <li>■ Fries (in grams)</li> <li>■ Chick Nuggets(in pieces)</li> <li>■ Chocolate Sundae</li> <li>■ Strawberry Sundae</li> </ul> </li> <li>• Staff is able to view what items are needed and how much of it is needed. (Same Items as the point above)</li> <li>• If ingredients left is less than the ingredients needed for a meal, inform the customer that the meal is unavailable</li> <li>• If (ingredients left / ingredients needed ) is less than or equal to 5, inform staff that the specific item is low on stock</li> </ul>
Priority (1 = Lowest and 10 = Highest)	3
Size (1 SP = 1 Day)	2 SP

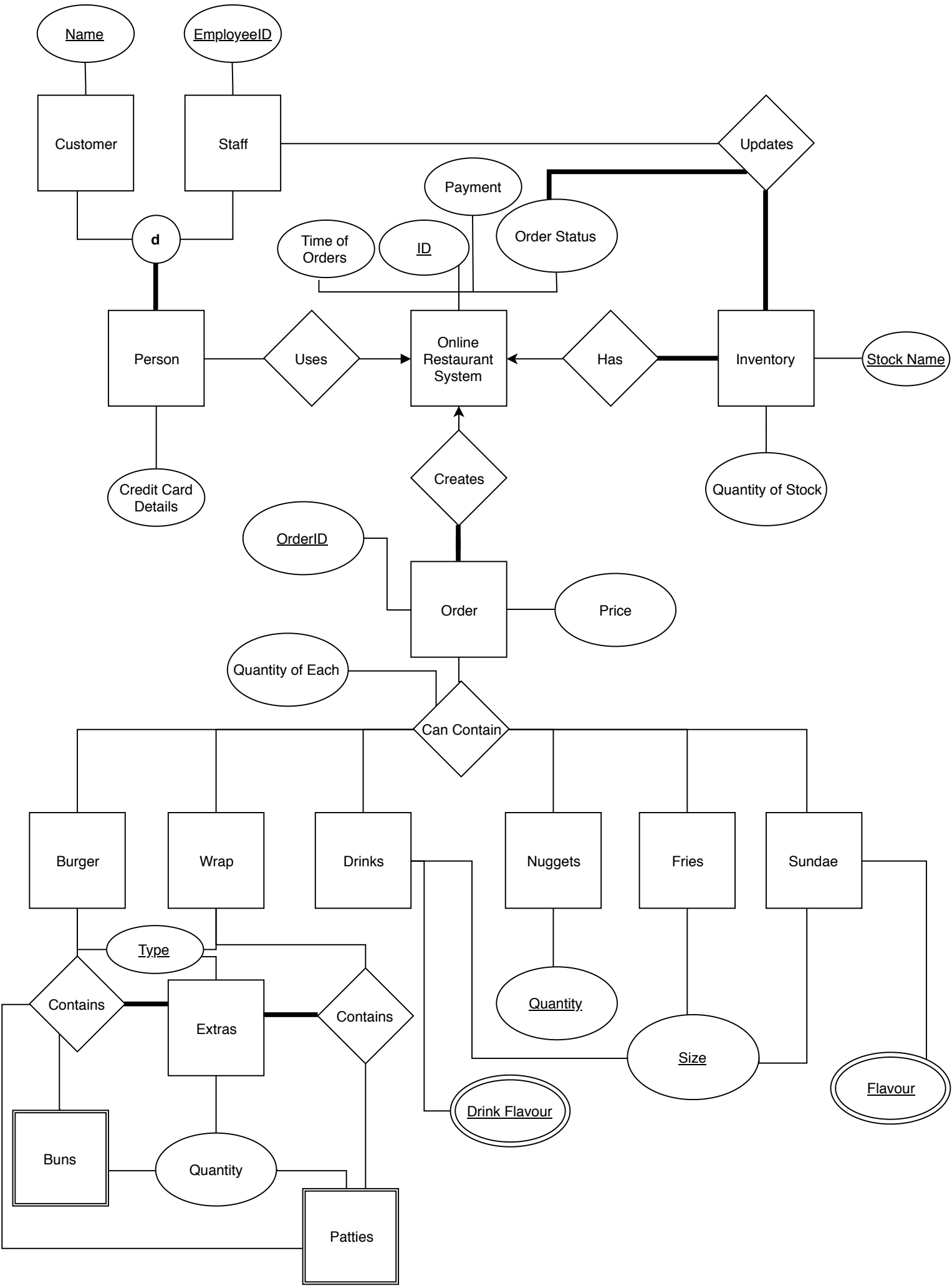
ID	SUS3
Name	Maintain Inventory
User Story Description	As staff, I should be able to restock instantly when requested, so I can make meals available all the time.
Acceptance Criteria	<ul style="list-style-type: none"> <li>• Staff is able to request for a restock at any time</li> </ul>
Priority (1 = Lowest and 10 = Highest)	6
Size (1 SP = 1 Day)	2 SP





## ER Diagram of Restaurant System

- Assumptions:**
- 1) Staff can access the staff options with the staff password "Wacdonald's".
  - 2) Orders can contain anything i.e. no one specific food or drink, but they must contain something for the order to be valid.
  - 3) Time of Orders attribute is visible for staff to see which order comes first when they cook the order.
  - 4) ID in System correlates to OrderID.
  - 5) Payment attribute for Online Restaurant System is only included for "realism" but we haven't actually included it in either the User Stories or code.



## Logbook

```
#####
      ###LOG BOOK DETAILS SUMMARY###
```

#Maintain a log book through the entire project that records:

#date of regular, stand-up meetings

#summary of decisions made in stand-up meetings, requirements elicited and key design decisions (hand-written user-stories, CRC cards etc.)

#responsibilities allocated to each team member and tasks to be accomplished for the next meeting

#progress of tasks using a velocity chart (a hand-drawing will suffice, no sophisticated tool needed), summary of decisions made in stand-up meetings

#milestones achieved

#reflection if assigned tasks (decided from last meeting) have been achieved

#any obstacles

###MARKS will be awarded for the log book (in weeks 7 and 10)

```
      ###END OF LOG BOOK DETAILS SUMMARY###
```

```
#####
```

Meeting\_1 (IN-PERSON):

date&time: Tuesday, 05/03/2019

main content: Milestone 1: product backlog (10%, due Sunday wk3)

description:

1. High Level Epic Stories from the problem statement
2. Each epic story broken into user stories - Each user-story must

define:

a unique story identifier (e.g., UC1)

a short description of the feature based on the Role-Goal-Benefit template (Refer to the RGB model described in the lectures)

an estimate for the implementation of the user story in user story points (e.g., UC1 = 2 User story points, where each point = 2.5 hours)

priority of implementation

3. acceptance criteria for each user story (Refer to the 3 C's model described in the lectures)

```
#####
```

#LINK TO MILESTONE1:

[https://docs.google.com/document/d/1aBwQOpnV7o\\_H\\_ftrXtBuz9svHMVAZJGa\\_9rk\\_2St8N4/edit?usp=sharing#](https://docs.google.com/document/d/1aBwQOpnV7o_H_ftrXtBuz9svHMVAZJGa_9rk_2St8N4/edit?usp=sharing#)

```
#####
```

Meeting\_2 (WEB):

date&time: Thursday, 07/03/2019

main content: Milestone 1: product backlog (10%, due Sunday wk3)

description:

1. Improved High-Level Epic Stories from the problem statement
2. Split the Epic stories
3. Correct errors in user-stories
4. Improved acceptance criteria for each user story (Refer to the 3 C's model described in the lectures)

#####

Start coding on 21/03/2019 9:33pm  
Created some classes on master branch  
2 branches are required for milestone2:  
1) Customer - Self Ordering  
2) Staff - Service Order  
TODO:  
1) Figure out the contents of each classes  
2) Distribute workload  
---XJ

#####

Milestone1 Achieved at the end of week4  
The reflection from tutors is basically simplify the redundant complicated functions in the user stories to reduce the complexity of implementing them

#####

Meeting\_3 (WEB)

date&time: Thursday, 28/03/2019

main content: Milestone 2 part 1: class diagram

description:

- 1) construct the classes and the basic structure
- 2) modified the code to match class diagram

#####

Milestone2 achieved at the end of week6  
1) finished the class-diagram  
2) coding on going

Reflection:

Had to change some aspects of the class diagram but for the most part, tutor said it was quite good as it had all the necessary classes implemented correctly and all we had to do was remove the customer and staff class as well as remove the dish.py and drinks.py so that we can directly access the: burger, wraps, fries, nuggets, bottles, etc, classes directly via "food" class which we had to merge drinks and dishes into.

#####

Meeting\_4 (WEB)

date&time: Thursday, 04/04/2019

1) Discussed how the overall implementation of the program would work and developed the conceptual framework of the program.

2) Delved into the minutiae for each code file and align with User Stories and overall concepts.

3) Split up the work for each member and worked together on issues that each individual was facing.

#####

Meeting\_5 (IN-PERSON)

date&time: Wednesday, 10/04/2019

1) Discussed steps going forward from Milestone 2 with Sam and Aarthi: Dheeraj and Aven writing most of the code required for Milestone 3 frontend as XJ did majority of Milestone 2 backend.

2) Discussed how the system would work and what minimum, required pages there would be to implement a working system. So far, we have home page (adding your order), order screen (editing your order) and confirmation page (showing the confirmed, final order information). Each page (except home obviously) will have a "Go back to home screen" button similar to the Week\_07/08 Lab.

3) Started Milestone 3 frontend, specifically concentrating on the "templates".

#####

Meeting\_6 (WEB)

date&time: Thursday, 11/04/2019

1) Further discussed how the system would work following the changes in the specifications such as the addition of sundaes (small, medium and large).

2) Split up the roles where XJ would handle the changes in backend whilst Aven and Dheeraj would split up the front-end between themselves.

3) Planned next tentative meeting on Saturday, 13/04/2019 to catch-up on the progress of each member and if anyone was stuck and needed help.

#####

Meeting\_7 (IN-PERSON)

date&time: Tuesday, 16/04/2019

1) Discussed with tutor on "pickle" implementation as well as other parts regarding back-end and front-end.

2) Had a short-update on roles and what each person was up to, and some of front-end, such as mains, was already done by then.

3) Planned next tentative meeting on Friday, 19/04/2019

#####

Meeting\_8 (WEB)

date&time: Thursday, 18/04/2019

1) Had some problems with make\_order function in the backend (routes.py) and discussed what we could do to fix it.

2) Entire customer front-end is finished whilst staff front-end was partly done.

3) Most of pickle is done and fixed some bugs in code.

- 4) Had updates of where each team-member was up to.
- 5) Planned next tentative meeting on Saturday, 20/04/2019

#####

#### Meeting\_9 (WEB)

date&time: Saturday, 20/04/2019

- 1) Nearly finished entire front-end for staff.
- 2) Editing some parts of customer front-end due to finding some bugs in the code.
- 3) Nearly done with changes to backend.
- 4) Planned next tentative meeting on Monday, 22/04/2019

#####

#### Meeting\_10 (WEB)

date&time: Monday, 22/04/2019

- 1) Entirety of code was finished on Sunday, 21/04/2019.
- 2) Did some simple styling to make front-end look more "appealing".
- 3) Started on diagrams and implementing the changes to the docs file [including User Stories, Logbook, ER diagram, etc]
- 4) Everyone was in unanimous agreement and happy with the code and progress so far, and had successfully resolved any conflicts they had previously, in a fair and equal manner.

#####

#### Feedback of milestone3:

Since we deleted the functional branches after we merged them to master, some commits disappeared, and it caused some confusion of work distribution and contribution.

Aven's commits are affected most, because most of his works were on a deleted front-end branch. Aven built the home page, order custom main page, add extra page, order premade main page and the related function. He also helped XJ and Dheeraj to build order confirmation page and order side/drink/dessert page.

Dheeraj built order side/drink/dessert pages, staff main page, staff serve order page, staff check inventory page, staff refill inventory page and the related functions.

XJ built order confirmation pages, check order status page, staff view orders page, staff login page and the related functions. He also helped Dheeraj to build staff serve order page. He also built the backend and added CSS style to the frontend.

All of us helped each other when we encounter any difficulties, the working atmosphere was great during the iteration3(milestone3).

#####

#### Meeting\_11 (WEB)

date&time: Saturday, 27/04/2019

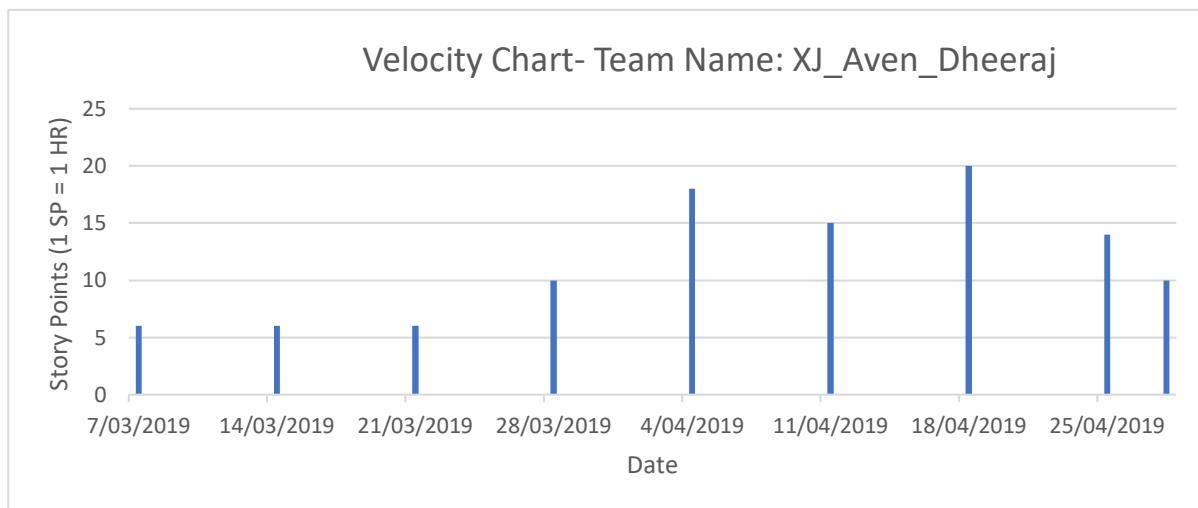
- 1) Latest document files successfully uploaded onto the master branch and combined into a Final report file.

2) Aven updated the user stories to reflect the changes for Milestone 3, XJ worked on the final Class Diagrams whilst Dheeraj worked on the ER diagram.

3) Everyone was happy with the final product and the report.

#####





**Data:**

Date	Story Points (1 SP = 1 HR)
7/03/2019	6
14/03/2019	6
21/03/2019	6
28/03/2019	10
4/04/2019	18
11/04/2019	15
18/04/2019	20
25/04/2019	14
28/04/2019	10