MemberlessTeam Final Report

comp1531 19T1 assignment

Team Members:

Yanning Cao (z5135152)

Katrina Ding (z5211336)

Henry Sihavong (z5022821)

Submit Date: 28 April 2019

Table of Contents

Section 1: User stories

Team Name: MemberlessTeam

PRIORITY: low, medium, high

STORY POINTS: 1 story point = 1 day

Note: Text in red are the changes (from original user stories).

Epic story 1: As a customer, I would like to order and pay for my meal online.

ID:	US1
NAME:	Home page

User-Story Description:

As a customer, when I enter the Home page, I should be able to choose to make new order or check orders status or continue my unsubmitted order, so that I can be directed to the correct page.

Acceptance Criteria:

- Customer should be able to make a new order by clicking "Make New Order" button, which will direct the customer to Menu page and generate an order with a unique order id for the customer.
- Customer should be able to continue his/her unsubmitted order by clicking "Continue my order" button. A text box should be displayed to ask for order id. Once the customer inserts correct order id and click "Confirm" button, the order detail with this order id will be displayed to customer and let him/her confirm that this order belongs to him/her by clicking "Continue Order" button. Then the customer will be directed to menu page.
- If the order is submitted, customer is not able to continue their order.
- Customer should be able to check and refresh his/her order status by clicking "Check Status" button. A text box should be displayed to ask for order id. Once the customer inserts correct order id and click "Confirm" button, he/she will be directed to Order Detail page.
- Customer is not able to check status of order with "Not submitted" status.

- Inserted order id should be positive integer. If the order is not found in system, a propriate error message should be displayed.

Priority	Low
Size	0.5 story point (1 point = 1 day)

ID:	US2
NAME:	Create a main online

User-Story Description:

As a customer, I want to be able to create a main online, so that I can choose the ingredients I want based on my preferences.

Acceptance Criteria:

- Two types of mains are offered: a burger or a wrap.
- Once the customers click on a main, they are able to choose from "Custom Burger", "Standard Burger" or "Cancel".
- Standard Burger consists 2 buns and 1 patty by default, and Standard Wrap consists 1 patty by default.
- If customer chooses Standard Burger or Standard Wrap, they should be able to insert only quantity.
- If customer chooses Custom Burger, they can choose number of buns (e.g., 3 buns for a double burger or 2 buns for a standard single burger) and number of patties by clicking "change buns/patties" button- the number of buns cannot exceed the maximum allowable limit: 4 (only single, double and triple burgers are permitted, so buns cannot be chosen more than 4), and the number of patties cannot exceed maximum allowable limit: 10.
- Customers should be able to select ingredients and insert quantity that they want. The selected ingredients with specific quantity will be added after the customer click the "change ingredients" button.
- The quantity inserted by customer should be restricted as positive integer.
- The choices of ingredients should be provided such as tomato, lettuce, tomato sauce, cheddar cheese, swisse cheese.
- Once a customer has completed their gourmet creation, the net price of their created main will be calculated based on the chosen buns, patties and ingredients and displayed to the customer.

- Customer should be able to add the main to their order by clicking "Confirm" button.
- A message should be displayed, which indicates that the main is successfully ordered, and customer should be able to return to menu page by clicking "Return" button.

Priority	High
Size	3 story points (1 point = 1 day)

ID:	US3
NAME:	Optionally order sides and drinks

User-Story Description:

As a customer, I would like to optionally order sides and drinks so I can add more variety to my meal and won't get thirsty.

Acceptance Criteria:

- Customers should be able to choose the size of sides and drinks.
- The sides include 2 sizes of nuggets: large (6/pack) and small (3/pack) and three sizes of Fries: small (20g), medium (40g), large (60).
- Lemonade can be either bottles (600ml) or cans (375ml).
- Juice such as Orange Juice have two sizes: small (250 ml), medium (450 ml).
- Sundae have 3 sizes: small (20g), medium (30g), large (40g).
- Customers are able to insert quantity and add the ordered size by clicking "Add" button.
- The quantity inserted by customer should be restricted as positive integer.
- Customers are able to add the items to their order by clicking "Confirm" button. A message will be displayed with items that they ordered, and customer should be able to return to menu page by clicking "Return" button.

Priority	Medium
Size	2 story points (1 point = 1 day)

ID:	US4
NAME:	Modify order

User-Story Description:

As a customer, I should be able to modify my order so that I don't have to re-order the items if I changed my mind.

Acceptance Criteria:

- Each time when the customer return to Menu page, the current order with ordered id, ordered items, order status and total price will be displayed under the menu.
- Customer should be able to delete a particular item by clicking the "Delete" button which is displayed beside each ordered item.

Priority	low
Size	0.5 story point (1 point = 1 day)

ID:	US5
NAME:	Checkout order

User-Story Description:

As a customer, I should be able to checkout so that my order is completed.

Acceptance Criteria:

- The order will be completed when customers click the "Confirm" button under their order details on Menu page.
- Once the customer checks out, the order details will be displayed on screen and the order status will be changed to "Pending".
- Customer should be able to refresh the order status by clicking "Refresh" button.
- The customer should have an option to enter in their email to have this receipt sent to them.

Priority	low
Size	1 story point (1 point = 1 day)

ID:	US6
NAME:	Check order status

User-Story Description:

As a customer, I should be able to check the status of my order at any point, so I know when to collect it.

Acceptance Criteria:

- Customer can check status at any time by clicking "Check Order Details" button on home page and enter their order id even if they quit or close the browser and re-enter it.
- Customer can see their ordered items, order status and total price on the page.

- Customer should be able to refresh the page by clicking "Refresh" button.
- Once the order is picked up, the order status will be changed to "Picked Up", but customer should still be able to send receipt to their email as long as the order has not been deleted.

Priority	Medium
Size	2 story point (1 point = 1 day)

Epic story 2: As a staff member, I would like to view and update the orders, so I know what orders to make and indicate when it's done

ID:	US7
NAME:	Staff current order

User-Story Description:

As a staff member, I should be able to view the current orders so that I can have an idea what to prepare next.

Acceptance Criteria:

- The staff can view the order at any point in time
- If there are orders, a list of the orders is shown along with their details.
- Staff can filter the order status to display only certain status of the orders to show by selecting a specific status and clicking "Refresh Page" button.

Priority	Medium
Size	2 story points (1 point = 1 day)

ID:	US8
NAME:	Staff order update

User-Story Description:

As a staff member, I should be able to update the status of an order of when it's ready for pickup so that I can acknowledge the completion of an order and move onto another.

Acceptance Criteria:

- Each order has its own "Prepare" and "Ready" buttons displayed beside the order.
- A staff member is able to click on "Prepare" button to indicate that staff is preparing the order. The order status will be changed to "Preparing".
- A staff member is able to click on "Ready" button to indicate that it's ready of pick up. The order status will be changed to "Ready".
- After an order is being picked up by customer, "Prepare" and "Ready" button is no longer displayed. A "Delete" button is displayed beside the order instead.
- A staff member is able to delete the order by clicking "Delete" button.
- Staff is able to refresh the page by clicking "Refresh" button at the top.

Priority	low
Size	1 story point (1 point = 1 day)

Epic story 3: As a staff member, I want to keep track of the various ingredients and refill stock depending on how low the inventory levels are.

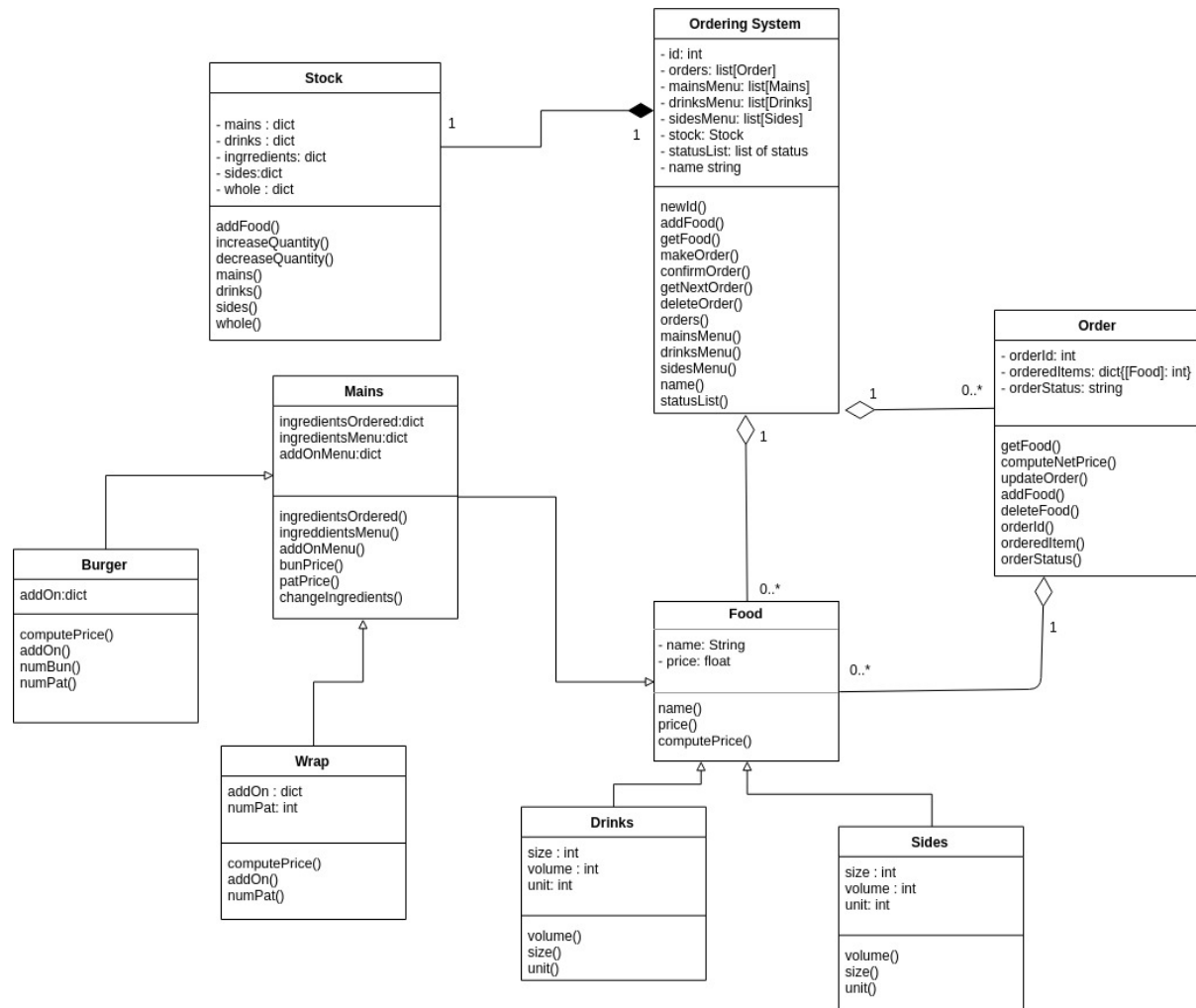| ID:	US9
NAME:	Maintaining inventory

User-Story Description:

As a staff member, I should be able to update the inventory levels of the stock according to the amount that customers have ordered so that the we can know the amount of ingredients we still have.
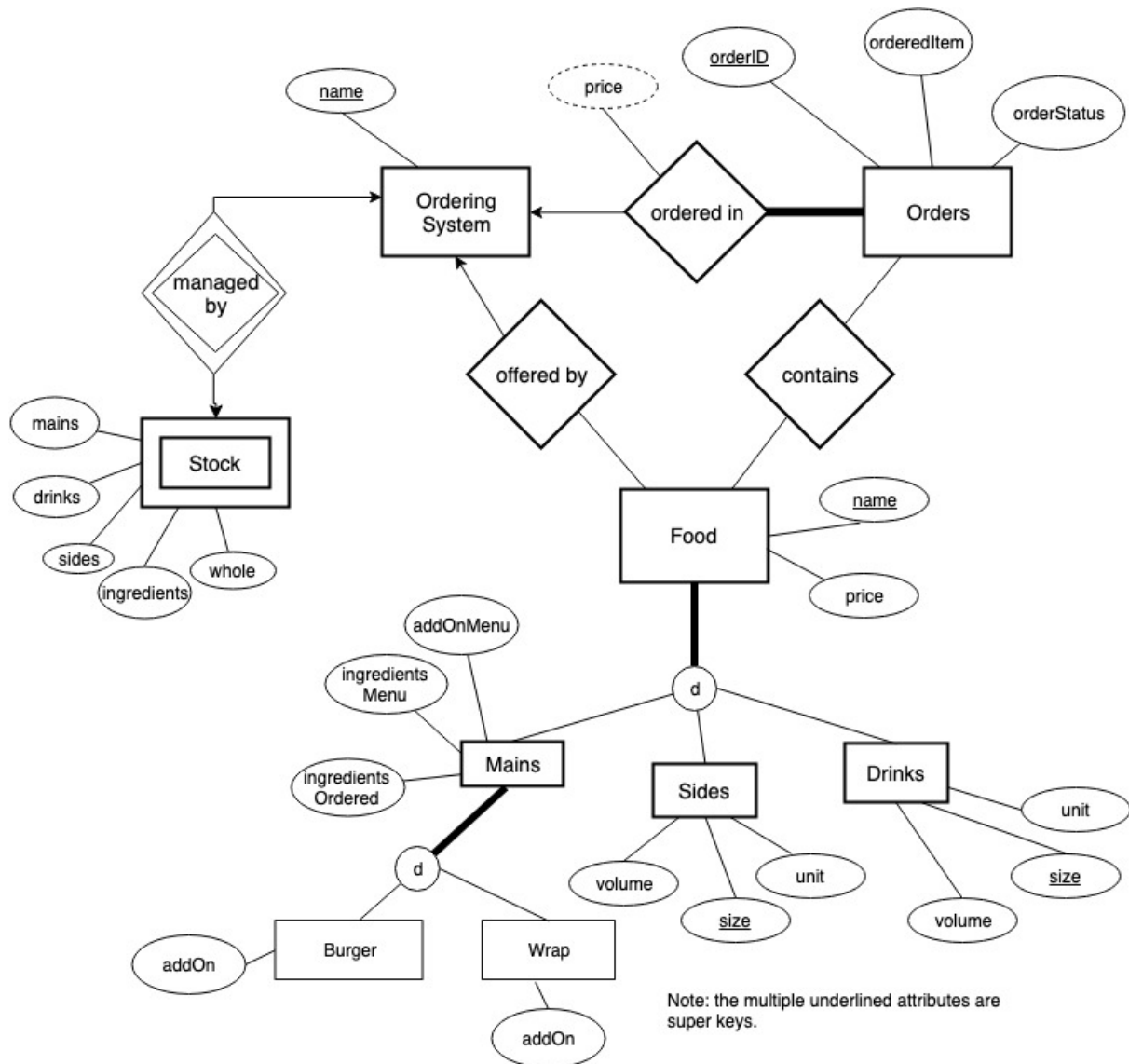
Acceptance Criteria:

- On Staff page, a staff is able to see the inventory level by clicking the "Stock" link at any time.
- Burgers, wraps, nuggets should all stocked in whole quantities.
- Bottled drinks (e.g. Lemonade) should be stocked in either cans (375 ml) or bottles (600 ml).
- Juice such as Orange Juice should be input as varying sizes: small (250 ml) and medium (450 ml).
- Sides such as fries and Sundae will need to be stocked by weight (in grams).
- When customers place an order, Inventory should be reduced the according amount.
- Staff is able to refill the stock by selecting the item to refill and quantity. The quantity of specific item in stock will be increased by the number inserted by staff after the staff click "Refill" button.
- The inserted quantity should be restricted as positive integer.
- On the Stock page, a staff is able to return to Staff page (order list page) by clicking the "Go back to Order list" link at any time.

Priority	high
Size	3 story points (1 point = 1 day)

Section 2: Class Diagram

Stock

- mains : dict
- drinks : dict
- ingredients: dict
- sides:dict
- whole : dict

addFood()
increaseQuantity()
decreaseQuantity()
mains()
drinks()
sides()
whole()

Ordering System

- id: int
- orders: list[Order]
- mainsMenu: list[Mains]
- drinksMenu: list[Drinks]
- sidesMenu: list[Sides]
- stock: Stock
- statusList: list of status
- name string

newId()
addFood()
getFood()
makeOrder()
confirmOrder()
getNextOrder()
deleteOrder()
orders()
mainsMenu()
drinksMenu()
sidesMenu()
name()
statusList()

Order

- orderId: int
- orderedItems: dict{[Food]: int}
- orderStatus: string

getFood()
computeNetPrice()
updateOrder()
addFood()
deleteFood()
orderId()
orderedItem()
orderStatus()

Mains

ingredientsOrdered:dict
ingredientsMenu:dict
addOnMenu:dict

ingredientsOrdered()
ingreddientsMenu()
addOnMenu()
bunPrice()
patPrice()
changeIngredients()

Burger

addOn:dict

computePrice()
addOn()
numBun()
numPat()

Wrap

addOn : dict
numPat: int

computePrice()
addOn()
numPat()

Food

- name: String
- price: float

name()
price()
computePrice()

Drinks

size : int
volume : int
unit: int

volume()
size()
unit()

Sides

size : int
volume : int
unit: int

volume()
size()
unit()

Section 3: ER design

Note: the multiple underlined attributes are super keys.

Section 4: Log Book

Week 2: Meeting date: 25/02/2019

Have meeting during the lab time. Distribute the workflow of the user Story.

Decisions:

- Name of group made to be "Memberless Team".

- For lab: combined user story for login of manager and admin.

- For lab: removed user story for additional staff.

- Making separate repos for labs and assignment.

Responsibilities:

Milestones achieved/Reflection:

Obstacles:

Week 3: Meeting date: 04/03/2019

Decisions:

Responsibilities:

- For lab: Started making use case diagram together.

- Katrina does detailed specification for main usage scenario.

- For Assignment: Started making user stories together.

- Katrina and Yanning finish it off.

- Henry edits it.

Milestones achieved/Reflection:

Obstacles:

Week 4: Meeting date: 11/03/2019

Decisions:

Responsibilities:

- For lab: Yanning creates CRC cards.

- For lab: Henry makes UML class diagram from CRC cards.

Milestones achieved/Reflection:

Obstacles: get confused by the uml diagram and don't know if we drew it in the right way.

Week 5: Meeting date: 18/03/2019

Decisions:

Responsibilities:

Milestones achieved/Reflection:

Obstacles:

Week 6: Meeting date: 25/03/2019

Decisions:

Responsibilities:

- For Assignment: Henry makes a UML class diagram.

- Yanning and Katrina edit it.

Milestones achieved/Reflection:

Obstacles: Not sure if we need to create so many subclasses.

Week 7: Meeting date: 01/04/2019

Decisions:

- For Assignment: txt file created to explain functions.

Responsibilities:

- Katrina writes classes for Order and Ordering System.

- Yanning writes classes for Mains(Burger, Wrap), Ingredients, and Food.

- Henry writes classes for Sides, Drinks, and Stock.

Milestones achieved/Reflection:

- Henry: Finished making classes assigned but it didn't work with the system created. Yanning had to fix it and make changes to System.

Obstacles:

- Henry: Still not use to using github and not that strong in coding.

Week 8: Meeting date: 08/04/2019

Decisions:

Responsibilities:

- Henry writes flask for Staff.

- Yanning writes flask for ordering process.

- Katrina does the rest.

Milestones achieved/Reflection:

- Henry didn't do his part.

Obstacles:

- Henry doesn't understand flask.

Solution:

- Yanning did Henry's part.

- Katrina improved it and applied OCP.

Week 9: Meeting date: 15/04/2019

Decisions:

Responsibilities:

- Yanning writes up remaining flask codes and all the tests.

- Katrina improved the code with OCP principles.

- Henry creates the ER diagram, updates log book and velocity chart.

Milestones achieved/Reflection:

Obstacles:

Week 10: 22/04/2019

Decisions:

Responsibilities:

-Yanning updates UML and writes report

- Katrina adds details

Milestones achieved/Reflection:

Obstacles:

Section 5: Velocity Chart