Universidad de las Fuerzas Armadas

CLASS NAME :OBJECT ORIENTED

PROGRAMMING
TEACHER: EDISON LASCANO
NRC:14575 WORKSHOP #:19
TOPIC:Project Inspection

Rubric

GitHub /100

1. Presentation /10
2. Interview /5
3. Project Definition /5
4. IEEE 830 /10
5. UML
   a. Use Case Diagram /10 : system, actors, use cases 9:40 - 9:50
      missing use cases, missing functionality, wrong name of actor
   b. Class Diagram /10 : classes, attributes, methods, relationships, good names
      10:05-10:17
6. Test Cases /10 10:30-10:45
7. Code Quality /10 code smells, pitfalls, clean code (-0.1)
8. Validation /10 10:30-10:45
9. Verification /20 test at least 10 test cases from the developers and create 10 different test cases on your own. Every test that fails is worth -1.

Evidence

GitHub

1. Presentation
2. Interview
3. Project Definition
4. IEEE 830
5. UML
   a. Use Case Diagram
   b. Class Diagram
6. Test Cases
7. Code Quality
8. Validation
9. Verification

Team 1: Inter Byte
Project: Food and Roll Orders
Customer: Jessica Sango, Chifa Michelena, Worker
GitHub URL: https://github.com/davidcepeda1/OOP14575_INTERBYTE.git
Inspector Team 6, Leader: Frederick Tipán

1. Rony Stiven Cedeño Montoya
2. David Gustavo Cepeda Salguero
3. Mateo Gabriel Criollo Llumiquinga
4. Brayan Sebastian Criollo Vega

Rubric

GitHub /100
Defense /100

1. Presentation ***8.18***/10
2. Interview 4/5
3. Project Definition 4.6/5
4. IEEE 830 9.7/10
5. UML
   a. Use Case Diagram 8/10
   1.2 One requirement is missing, is about table reservation (-1)
   1.1 More specific with relations (can a chef create its own review?) (-1)
   b. Class Diagram 6/10
   Class names are wrong (-1)
   1.1 Restaurant Information
   1.2 Rating & Reviews
   1.3 Shopping cart
   1.4 Restaurant chef
   Dependencies: Restaurant chef class needs a create dependency for the Menu class (-1)
   Associations: Manager and Customer class need an association for Menu class
   Customer doesn't have Rating & Reviews as an attribute, the composition relation doesn't make sense. The same with the Shopping cart class.
   Cart item class doesn't have association with the Plate class

   Attributes begin with low camelcase (-1)
   Restaurant chefChefName
   Method begin with low camelcase (-1)
   Manager -> RegisterNewPlate()
   Observations: use generalizations, compositions and aggregations don't need a verb

6. Test Cases 9/10
   The real output from each test case is missing(-1)
   They have 10 test cases
7. Code Quality 7/10

Simplify the names of the attributes form the RestaurantInformation class (-1)

Same for the userId attribute in the user class

A lot of methods don’t have a body, therefore doesn’t have a real functionality (-1)

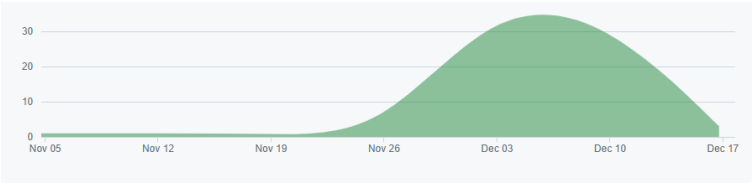
The main class has too many methods on it. There is the display of the Menus, some validations, csv files management and more. (-1)

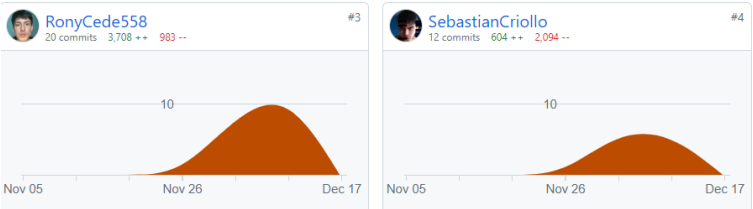
8. Validation 8/10
   The chef can't do any action (-1)
   The Customer can’t do rating (-1)
9. Verification 5/20

As a chef user you can’t do anything
There are serious validation problems with the inputs in many parts of the code, example: inserting a new ID, entering the quantity or the cost. They are detailed in our Test Cases.
The data from the Menu json is always deleted, therefore you can’t really do anything with the program.

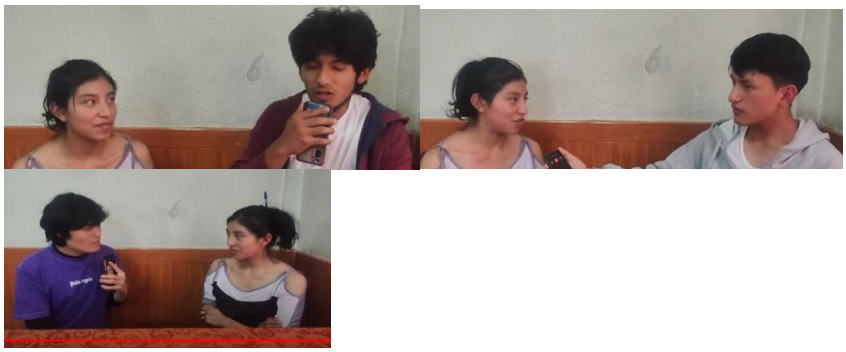
Evidence

GitHub





1. Presentation
2. Interview



3. Project Definition

Background

"Food and Roll", a company born from the dream of a young entrepreneur passionate about gastronomy and fast food, experienced great success in its time. However, due to the COVID-19 pandemic, the restaurant world changed drastically, resulting in increased competition in the food and roll market. Given the pandemic, many restaurants faced the difficult task of digitizing themselves to continue operating and meeting the needs of their customers.

Unnecessary context

As "Food and Roll" wasn't a competitive digital restaurant, it was necessary to improve its own services and adapt to the new demands of the market. Thus, in the field of technology, the range of products and the quality of service offered by Food and Roll have become more innovative and attractive.

The name of the project is not the same as the name of the Restaurant according to the interview

The problem is rooted in the daily experience of customers interacting with the establishment. The growing trend of customers opting for online ordering options, driven by the desire to avoid unnecessary waits, and the pressing need to optimize table management, especially during peak demand, stand out as essential factors. Recognizing the urgency to streamline the ordering process, facilitate menu visualization, and provide efficient table reservation options, the vision of an application was formulated that would not only address these operational challenges but also enhance the overall customer experience. This historical approach reinforces the need for "Food and Roll" to adapt to a constantly changing digital environment while emphasizing the strategic relevance of the initiative in the current landscape.

Doesn't mention how the restaurant is currently operating with the funtionlaities that software is trying to improve.

4. IEEE 830

1.2 System Scope

Functionality not detailed later

The name of the system is "Food and Roll Orders." It is a food delivery system that includes functions such as "Order for dining in the restaurant," "Home delivery," "Table reservations," "Interactive menu," "Rate Chef and Restaurant," "Total price," "Delivery cost," and "Estimated delivery time display." What it won't do is "Currency exchange from USD to any

User Characteristics:
Details the general characteristics of intended users, including their level of education and technical expertise.
Constraints:
Lists limitations that developers must consider, such as business policies and hardware restrictions.
Assumptions and Dependencies:
Identifies external factors that may affect requirements, establishing contextual conditions.
Future Requirements:
Outlines possible improvements that could be implemented in future versions of the system.

3. Specific Requirements:

Just go straight to the point, don't detail what each point is about

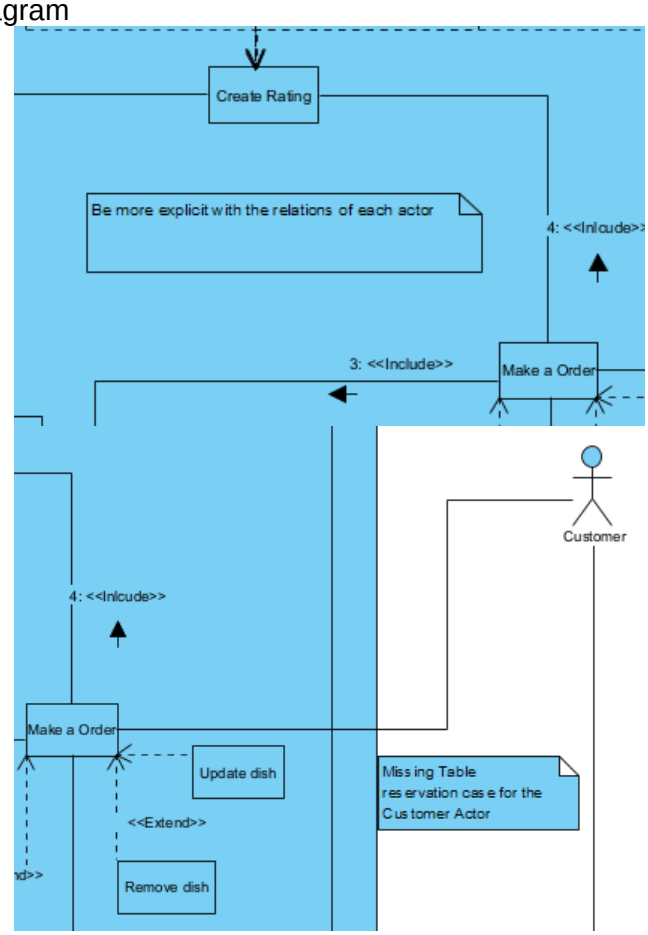
External Interfaces:
Describes requirements affecting the user interface and interactions with other systems.
Functions:
Details specific actions the software must perform, organized by user types, objects, goals, or functional hierarchy.

Product Functions:
Summarizes the main functions performed by the system, such as order management, menu display, and delivery tracking.

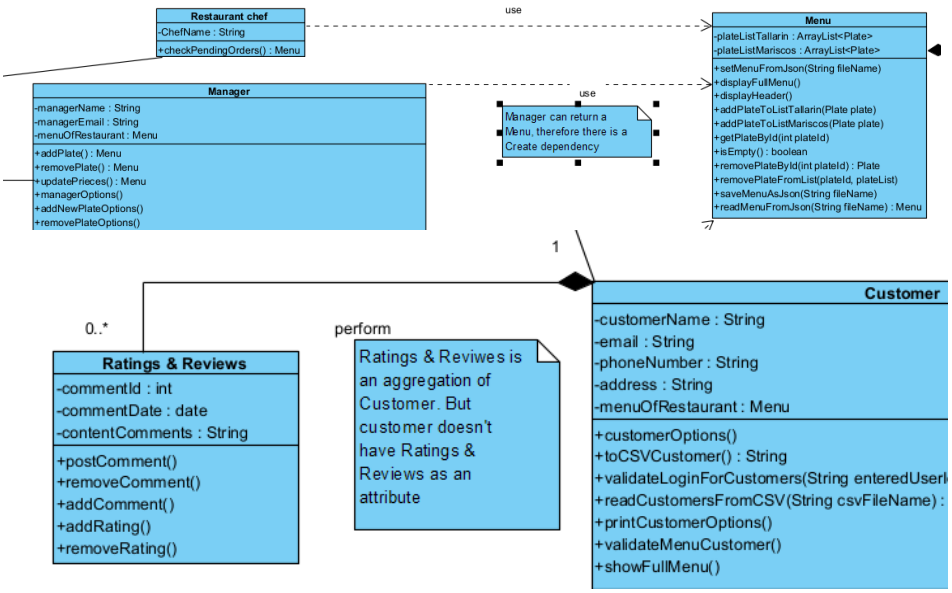
It is not mentioned later

5. UML

a. Use Case Diagram



b. Class Diagram



6. Test Cases

1	Test cases							
2	Food and Roll Orders							
3	ID	Summary	Priority	Preconditions	Input data	Steps	Expected result	
4	1	Check the layout of the initial screen	Low	Have the program or the .jar file	N/A	1. Open the command console. 2. Enter the directory where the program is located java -jar "Food and Roll Orders"	The program runs smoothly and the main restaurant data such as the menu of options can be displayed	missing real output

7. Code Quality

```
public class RestaurantInformation {
    private String addressOfRestaurant;
    private String phoneNumberOfRestaurant;
    private String emailOfRestaurant;
    private String locationReference;
```

Redundant attribute names. We already know that this is a class referencing the Restaurant

```
public static void verifyLogin() {

}
```

Even though it is in the UML, this method doesn't have a body

```
public Menu checkPendingOrders () {
    Menu menu = new Menu();
    return menu;
}
```

Even though it is in the UML, this method always return an empty Menu class

```
stomer customer = new Customer(customerName, email: customerEmail, phoneNumber: customerPhoneNumber, address: null,
veToCSVCustomer(customer);
```

Every time you enter an Adress, it is never used, and it is instead replaced with null

8. Validation

```
*=======================================*
Chef Option
1. Register new chef
2. Login
3. Exit
Select an option:
2
*=======================================*
Chef Option
1. Register new chef
2. Login
3. Exit
Select an option:
 *================Customer Options=======================*
 1. Show Full Menu
 2. Add Dish To Menu
 3. View Cart Details
 4. Return
 Select an option:
```

9. Verification

					from the menu.	
4	Allow the chef to view the restaurant's orders.	Medium.	Confirmed password, registered and previously logged-in chef	N/A	1. Access the chef section. 2. Enter the access password. 3. Select the option to review orders.	The chef can review the list of orders that have been placed.

This test case states that the chef can view the orders, but we cannot access the chef's menu.

7	Modify an order.	Medium	Registered customer, logged in, and having placed an order	Selection of menu data, order, registered user.	1. Log in to the restaurant system with valid credentials. 2. Access the option to modify the order, requesting the order ID. 3. Modify the quantity of dishes, add or remove items from the order. 4. Confirm the changes made.	Send a confirmation message about the changes made.

This test case was not tested in the system as that option does not exist

```
Please enter only numbers for the dish ID.
Enter the ID of the dish to add the cart: :
3
*===========camaron===============*
Price: $1.0
Availavility: s
Description: camaron
================================================
Enter the quantity:
as
Exception in thread "main" java.util.InputMismatchException
        at java.base/java.util.Scanner.throwFor(Scanner.java:947)
        at java.base/java.util.Scanner.next(Scanner.java:1602)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2267)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2221)
        at ec.edu.espe.foodandrollorder.model.ShoppingCart.quantityOfDishes(ShoppingCart.java:68)
        at ec.edu.espe.foodandrollorder.model.ShoppingCart.addDishToCart(ShoppingCart.java:52)
        at ec.edu.espe.foodandrollorder.model.Customer.customerOptions(Customer.java:54)
        at ec.edu.espe.foodandrollorder.view.FoodAndRollSystem.customer(FoodAndRollSystem.java:208)
        at ec.edu.espe.foodandrollorder.view.FoodAndRollSystem.main(FoodAndRollSystem.java:52)
C:\Users\Admin\POO Gabriel Vivanco\review\OOP14575_INTERBYTE\06-Code\02-Code\Food and Roll Order V2.0.0\FoodAndRollOrder\nbproject\build-impl.xml:1330: The fol
C:\Users\Admin\POO Gabriel Vivanco\review\OOP14575_INTERBYTE\06-Code\02-Code\Food and Roll Order V2.0.0\FoodAndRollOrder\nbproject\build-impl.xml:936: Java ret
BUILD FAILED (total time: 21 minutes 14 seconds)
```

Validation problem when entering a letter instead of a Number

Input	Description	Expected Result	Actual Result
1	Register a new chef in the Chef Options	Display another menu or ask the user other information	Insert again an option value
2	Login as an existing a chef	Display the chef user interface	Insert again an option value
Comité del Pueblo	Enter the Adress of a new customer and save it in a .csv file	Save the data in the correct field inside the Customer csv	Every time it is saved as null
-500	Insert the quantity of a dish as a Customer	Show an error, because you could not accept negative values as a quantity	Passed without problems
16	Insert the ID of a manager when there is already another manager with the same ID	Show an error because each ID must be unique	The manager is created without problems
Choclo	Introduce a new dish	The new dish must be added to the menu	All the dishes on the menu are deleted
5	Register a new user(id) to enter	The system should not allow repeat users to enter	The system allows anything to be entered, even repeated
-74	Add the price of a dish	Would not allow negative values as a currency	Passed without problems
L	Enter if the dish is available or not	Ask for YES/NO as the only input	Doesn't matter the input, it will pass
twenty five	Add a new dish	Error, enter a number	The system read characters
one	Add new plate options	Error, enter a number, and ask for option	Enters in a bucle
two	Enter quantity of the dish	Error, enter a number	The system crash