



University of the Armed Forces - ESPE

CLASS NAME : OBJECT ORIENTED PROGRAMMING

INSTRUCTOR: EDISON LASCANO

NRC: 14575

HOMEWORK #: 4

TOPIC: PROJECT FUNDING AND OBJECT MODELING

Group #1 (InterByte)

Members:

- RONY STIVEN CEDEÑO MONTOYA
- DAVID GUSTAVO CEPEDA SALGUERO
- MATEO GABRIEL CRIOLLO LLUMIQUINGA
- BRAYAN SEBASTIAN CRIOLLO VEGA

Software application for the food establishment “Food and Roll”

Objects	Yes (✓)	No (X)
1.- Food	✓	
2.- Menu	✓	
3.- Restaurant		X
4.- Program		X
5.- Records		X
6.- Customers	✓	
7.- Time	✓	
8.- ID Order		X
9.- Address		X
10.-Ability		X
11.- Data	✓	
12.- Access		X
13.- Matter		X
14.- Industry		X
15.-Option		X
16.- Points		X
17.- Comments		X
18.- Reviews	✓	
19.- Friends		X
20.- Conversation		X
21.- Application		X

22.- Market		X
23.- Reservation	✓	
24.- Service		X
25.- Total	✓	
26. Chef	✓	
27. Management	✓	

Objects	Attributes	Methods
1.- Food	<ul style="list-style-type: none"> • typesOfFood : Food[] • foodCost : double • foodName : char • flavorOfFood : char 	<ul style="list-style-type: none"> • describe(Provides information about the food, such as its type, name, and flavor) • prepare() • serve()
2.- Menú	<ul style="list-style-type: none"> • burgerProducts : String • wingProducts : String • wingCategories : String • maxPrice : float • drinkProducts : String • drinkCategories : String 	<ul style="list-style-type: none"> • addBurger() • addWing() • calculateAveragePrice() • removeBurger() • removeWing() • addDrinkProduct() • removeDrinkProduct() • getProductsByCategory()
3.- Customers	<ul style="list-style-type: none"> • customerID : int • name : String • email : String • address : String • phoneNumber : long int 	<ul style="list-style-type: none"> • createAccount() • updateProfile() • login() • makeReservation() • cancelReservation()
4.- Time	<ul style="list-style-type: none"> • waitTimeOnMinutes: int • preparationTimeOnMinutes: int • restaurantOpeningTime: float • restaurantClosingTime: float • hourOrder : float • hourDelivery : float 	<ul style="list-style-type: none"> • itsOpen(check if the restaurant is open at a specific time) • calculateWaitingTime(calculate the estimated waiting time for customers) • orderDeliveryTime (records the delivery time of the order)
5.- Data	<ul style="list-style-type: none"> • customerDataBase : String • orderRegistration : float • newMenu : String 	<ul style="list-style-type: none"> • addClient() • updateMenu() • registerOrder()

	<ul style="list-style-type: none"> customer : String nameCustomer : String menu : String foodName: String 	
6.- Reviews	<ul style="list-style-type: none"> reviewID : int restaurantID: int customerID : int rating : int comments : String 	<ul style="list-style-type: none"> addReview() editReview() deleteReview()
7.- Reservati on	<ul style="list-style-type: none"> reservationID : int dateTime : Date numberOfPeople : int customerList : Customer[] reservedMenu : Reservation[] 	<ul style="list-style-type: none"> addCustomer(Customer customer) removeCustomer(Customer customer) cancelReservation() confirmReservation() modifyReservation(DateTim e newDateTime, int newNumberOfPeople) sendReservationConfirmatio nEmail()
8.- Total	<ul style="list-style-type: none"> totalAmount : float taxAmount : float discountAmount : float 	<ul style="list-style-type: none"> getTotal Amount() getTaxAmount() getDiscount()
9.- Managem ent	<ul style="list-style-type: none"> nameOfTheNewPlate: String removePlate: String customerName: String customerEmail: String phoneNumber: int address : String 	<ul style="list-style-type: none"> managingCuterOrders() managingDishes() modifyTheMenu() addDishToMenu() removeDishFromMenu() editCustomerInformation() editCustomerOrder()
10.- Chef	<ul style="list-style-type: none"> id : int name: String timetableOnHours: int 	<ul style="list-style-type: none"> getID() getName() getChefTime() addNoteToChef()

Link: [Workshop: Finding Objects, then classes, then model](#)

UML DIAGRAM

