Rotterdam School of Management
Erasmus University

Assignment 3:

UML Diagrams

CONCLUDISHES

Group 11

Khoi Gia Pham 523755

Yingqi Jia 592804

Xu Cao 602109

Zi-Xin Chen 592198

Alexander Skakavac 635584

Course: Designing Business Application

MSc Business Information Management

Rotterdam School of Management, Erasmus University

1. Introduction

ConcluDishes is an application that connects customers to restaurants that have surplus food. The application's objective is to reduce food waste in the restaurant industry, help individuals buy cheaper surplus food, and fight climate change. This report presents two Unified Modelling Language (UML) Diagrams based on the most critical functional requirements (FR) in the Software Requirements Specification (ConcluDishes, 2022). The first diagram is the Class diagram which entails the FR1, FR2, and FR3. And the second one is the Activity Diagram which illustrates FR3, FR4, and FR5.

1.1 Functional Requirements

ID	Customer Interface	ID	Administrative Interface
FRX.1	Register account	FRX.11	Administrator account set up
FRX.2	Login account	FRX.12	Login account
FRX.3	Display featured restaurants	FRX.13	Browse the admin dashboard
FR1	**Search for restaurant**	FRX.14	Manage restaurants
FRX.4	Search for items from the menu	FRX.15	Manage customers
FR2	**Select an item from the menu**	FRX.16	Manage orders
FRX.5	Customize options for a selected item	FRX.17	Logout account
FRX.6	Add an item to the current order		
FR3	**Review current order**		
FR4	**Provide delivery and payment details**		
FR5	**Place an order**		
FRX.7	Receive order confirmation		
FRX.8	Request assistance with a problem		
FRX.9	Cancel order		
FRX.10	Rate order		
FRX.11	Logout account		

Table 1. ConcluDishes Functional Requirements

1.2 Non-Functional Requirements

ID	Category	Requirement
NFR1	**Customer satisfaction**	**Usability**
NFR2	**Data privacy**	**Security**
NFR3	**System performance**	**Capacity**
NFRX.1	System performance	Availability
NFR4	**System performance**	**Scalability**
NFR5	**System performance**	**Reliability**
NFRX.2	System performance	Portability
NFRX.3	System performance	Compatibility

Table 2. ConcluDishes Non-Functional Requirements

1.3 Definitions

ID	Definition	Description
D1	System	ConcluDishes platform.
D2	Customer profile	Profile of a customer [U1A] that includes information about customer name, phone number, email, delivery address.
D3	Restaurant profile	Profile of a restaurant [U1B] that contains information about restaurant ID, name, location, contact phone number, menu, food category, price range.
D4	Customer input	The information that the customer [U1A] provides to the system [D1].
D5	Restaurant input	The information that the restaurant [U1B] provides to the system [D1].
D6	Transaction	Occurs when one customer [U1A] purchases the meal on the system [D1].
D7	Food item	A food item that is offered by restaurants [U1B] on the system [D1].
D8	Search query	A search request provided by the customer [U1A] to the search widget of food item [D7].
D9	Search field	A search field available on the top-bar of system [D1] that enables the customer [U1A] to search for desirable food items [D7].
D10	Search result	The output of food category [D12] matching the search query [D8] provided by the customer [U1A].
D11	Purchase Dates	The available dates for purchase the food item [D7].
D12	Food category	The type of food item [D7]: Italian, Pizza, Burger, American, Seafood, BBQ, Bread & pastries, Sushi, Turkish, Comfort Food, Halal, Dessert, Others.
D13	Food Dietary	The tag of food item [D7]: Vegetarian, Vegan, Gluten-free, Halal, Allergy Friendly
D14	Price	The total amount a customer [U1A] pays for ordering a food item [D7].
D15	Price range	Euro symbols (€) to denote restaurant [U1B] menu price range €-inexpensive, ~~€€~~-mid range price, ~~€€€~~-expensive meal, ~~€€€€~~-high class meal.
D16	Delivery option	The way a customer [U1A] prefers to take the food item [D7]: only pick-up, only delivery, support both pick-up and delivery
D17	Delivery fee	The amount charged for delivery service.
D18	Payment method	The preferred way of paying selected by the customer [U1A]: Credit card, Debit card, iDEAL, PayPal, Apple Pay, Google Pay, Cash on Delivery
D19	Delivery address	The form that the customer [U1A] must fill in contains the following delivery information: Country, County/Province, City/Town, ZIP/Postal Code, Address Line 1, Address Line 2.

D20	Delivery time	The estimate time for an order to arrive at the customer [U1A].
D21	Delivery confirmation	Once the food item [D7] has been delivered, the system [D1] will notify the customer [U1A].
D22	Payment confirmation	Once the customer [U1A] has successfully transferred the price [D14], the system [D1] will confirm the transaction.
D23	Invoice	Once payment has been finalized, the system [D1] will generate an invoice containing the following information: Order reference, Price [D17], Payment Method [D18], Transaction Date.
D24	Transaction ID	Each transaction has a unique ID within the system [D1].
D25	Database	The system database stores information over customer profiles [D2], restaurant profiles [D3], and transactions [D6].
D26	GDPR	General Data Protection Regulation: law that aims for consistent protection of consumer and personal data across EU nations (Wolford, n.d.).
D27	Pick-up time	The preferred time for customer [U1A] to pick up the food item [D7].
D28	Deposit	The amount of money that the customer [U1A] pays to the restaurant [U1B] for the order.
D29	Holding account	The system's [D1] deposit account, where all the deposits [D29] will be held until the order is completed.
D30	Delivery courier	The list of delivery partners a customer [U1A] can select to deliver their order.
D31	Food basket	The list of food items [D7] that the customer [UA1] selected for purchase.
D32	Menu page	The list of food items [D7] that are offered by a restaurant [U1B].

Table 3. Definitions

1.4 User classes

ID	User Class	Description
U1	User	Individual that has downloaded the system [D1] and uses the system
U1A	User - Customer	A customer [U1A] who is searching for or ordering food items from restaurants [U1B] through the system
U1B	User - Restaurant	A restaurant [U1B] that is offering food items to customers [U1A] through the system.
U2	Administrator	Employee of the system [D1] that is in charge of the maintenance of the system [D1] and its functionalities

Table 4. User classes

1.5 Assumptions

ID	Assumption
A1	Customers [U1A] own a smartphone or tablet to be able to download the application.
A2	The surplus food item [D7] complies with health regulations.

A3	Customers [U1A] agree to the GDPR [D26]; personal data will be securely stored.
A4	The system [D1] has enough drivers to deliver the items [D7] from the restaurant [U1B] to the customer [U1A].
A5	Customers [U1A] want to buy food items [D7] that would otherwise be disposed of.
A6	The customers [U1A] will be willing to buy the food item [D8] at a lower price from the restaurants [U1B].
A7	The users [U1] will rate the system [D1] truthfully.
A8	The system [D1] can handle 57,000 concurrent users [U1] at the same time.
A9	The customers [U1A] have payment methods accepted by the system [D1].
A10	The customers [U1A] will confirm the delivery confirmation [D22] once they receive the food items [D8].
A11	Our market share is 0.1% of Too Good To Go ApS– the top incumbent player in the same industry.
A12	The number of customers in Year 1 is 57,700 users, which is 0.1% of that of Too Good To Go ApS.
A13	The number of establishments with restaurants in Year 1 is 1540 restaurants, which is 0.1% of that of Too Good To Go ApS.

Table 5. Assumptions

1.6 Prioritization

ID	Priority	Description
P1	High	Requirements that are critical for the functionality of the system.
P2	Medium	Requirements that are not critical for the functionality of the system but are beneficial to the system and customer experience.
P3	Low	Requirements that do not affect the functionality of the system but might be beneficial to the system and customer experience.

Table 6. Requirements Prioritization

4. Class Diagram

4.1 Used Functional Requirements

FR1. Search for restaurant

FR1.1 The system [D1] shall give the customer [U1A] the option to perform search query [D8] for available restaurant profiles [D3], food categories [D12], or food items [D7] by entering keywords or parts of keywords into the search field [D9].

FR1.2 The system [D1] shall refresh the search results [D10] on the input of each letter from the customer.

FR1.2.1 If the search query [D8] matches with restaurant profiles [D3] or food category [D12] on the system, the system [D1] shall display the search results [D10] by the following orders:

FR1.2.1.1 Restaurant profiles [D3] and that restaurant's food category [D12] and price range [D15]

FR1.2.1.2 Food category [D12]

FR1.2.1.3 Alphabetical orders of the search results [D10] from [FR1.2.1.1] and [FR1.2.1.2]

FR1.2.2 If the search query [D8] does not match with any restaurant profiles [D3] or food category [D12] on the system, the system [D1] shall display the message "We didn't find a match. Try searching for something else instead."

FR1.3 The system [D1] shall give the customer [U1A] the option to apply additional sorts and filters to the search results [D10] under [FR1.2.1]

FR1.3.1 The system [D1] shall provide the customer [U1A] with the below sort options:

FR1.3.1.1 Recommended for you (default option)

FR1.3.1.2 Most popular

FR1.3.1.3 Highest rated

FR1.3.2 The system [D1] shall provide the customer [U1A] with the below filter options:

FR1.3.1.1 Price range [D15]

FR1.3.1.2 Payment method [D18]

FR1.3.1.3 Dietary [D13]

FR1.3.1.4 Delivery method [D34]

Rationale: The user interface of most prevailing food delivery apps provides a search field [D9], which allows the customers [U1A] to quickly search for desirable and available restaurant profiles [D3], food categories [D12], or food items [D7]. This requirement is not critical for the system's functionality but is beneficial to the system and customer experience.

Source: ConcluDishes Business Case (2022), Antala (2019)

Author: KGP

Priority: Medium (P2)

Dependency: [FRX.4] [FR2]

History: 25/09/2022 (First version), 29/09/2022 (Revision), 02/10/2022 (Final Version)

FR2 Select food item on menu

FR2 The system [D1] shall require the customer [U1A] to select the desirable food item [D7] when the customer [U1A] lands on a restaurant menu page [D32].

FR2.1 When the food item [D7] is selected, the system [D1] shall retrieve the following specifications about the food item [D7]:

FR2.1.1 Name

FR2.1.2 Price

FR2.1.3 Ingredient information

FR2.1.4 Customizing options

FR2.1.5 Amount

FR2.1.6 Delivery time

FR 2.2 The system [D1] shall give the customer [U1A] the option to add the selected food items [D7] into the food basket [D31].

FR 2.3 When the customer [U1A] decides not to add the selected food item [D7] into the food basket [D31], the system [D1] shall allow the users to return to the restaurant menu page [D32].

FR2.4 When food item [D7] is not offered by the restaurant [U1B], the system [D1] shall display the message “This item is not available. Please select something else instead”.

Rationale: For the customer [U1A] to receive their desired order, they must be able to select a food item [D7] effectively and efficiently. Effective selection entails selecting the right food items

[D7], which the system [D1] can achieve by providing customer [U1A] with specifications about the food items named above. Popular food apps such as Too Good To Go and Uber Eats currently use these information items. Efficient selection entails selecting food items [D7] with as few steps as necessary to effectively convey the food items [D7] information. The previously mentioned food providers have these steps in their applications.

Source: ConcluDishes Business Case (2022)

Author: ZC

Priority: High (P1)

Dependency: [FR1], [FR3]

History: 27/09/2022 (First version), 29/09/2022 (Revision), 03/10/2022 (Final Version)

FR3 Review current order

FR3.1 The system [D1] shall allow the customer [U1A] to review the food basket [D31] at any given time.

FR3.1.1 If the customer [U1A] adds items [D7] to the food basket [D31], the system [D1] shall allow the customer [U1A] to view:

FR3.1.1 Items [D7]

FR3.1.2 Price [D14]

FR3.1.2 Delivery fee [D17].

FR3.1.2 If the customer [U1A] adds no item [D7] to the food basket [D31], the system [D1] shall display the message "Your food basket is empty"

FR3.2 The system [D1] shall enable the customer [U1A] to modify the order in the food basket [D31] with the following functions:

FR3.2.1 Delete items [D7].

FR3.2.2 Change the number of items [D7].

FR3.2.3 Schedule the delivery time [D20].

FR3.2.4 Select delivery options [D16].

FR3.2.5 Add promotional code.

Rationale: The system [D1] should allow the customer [U1A] to check the food basket [D31] whenever the customer [U1A] wishes to see it. The system [D1] will let the customer [U1A] see

the items [D7] in the food basket [D31], the price [D14], and the delivery fee [D17]. The system [D1] will also let the customer [U1A] modify the order if they want to delete items [D7], change the number of items [D7], schedule the delivery time [D20], select delivery options [D16], and add a promotional code.

Source: ConcluDishes Business Case (2022)

Author: AS

Priority: High (P1)

Dependency: [FR1], [FR2], [NFR1].

History: 28/09/2022 (First version), 29/09/2022 (Revision), 2/10/2022 (Final version)

4.2 Class Diagram Description

This class diagram contains three main functional requirements (FR1, FR2, FR3). And there are six classes in the diagram, that are Customer, Restaurant, Food basket, Food item, Food category, and Delivery address.

As shown in the diagram, the system keeps track of the customer information, including customer ID, name, phone number, and email address. Each customer can search for restaurants, food categories, and items. A search operation takes keywords as input and returns restaurants, food categories, and food items as outputs. One customer has only one food basket. And a customer can review the food basket, which includes the food item names and the item quantity. A customer can also modify the food basket (delete food items, change the number of food items, schedule the delivery time, select delivery options, and add a promotional code). One food basket may include zero or many food items, and one food item can be added to zero or many food baskets.

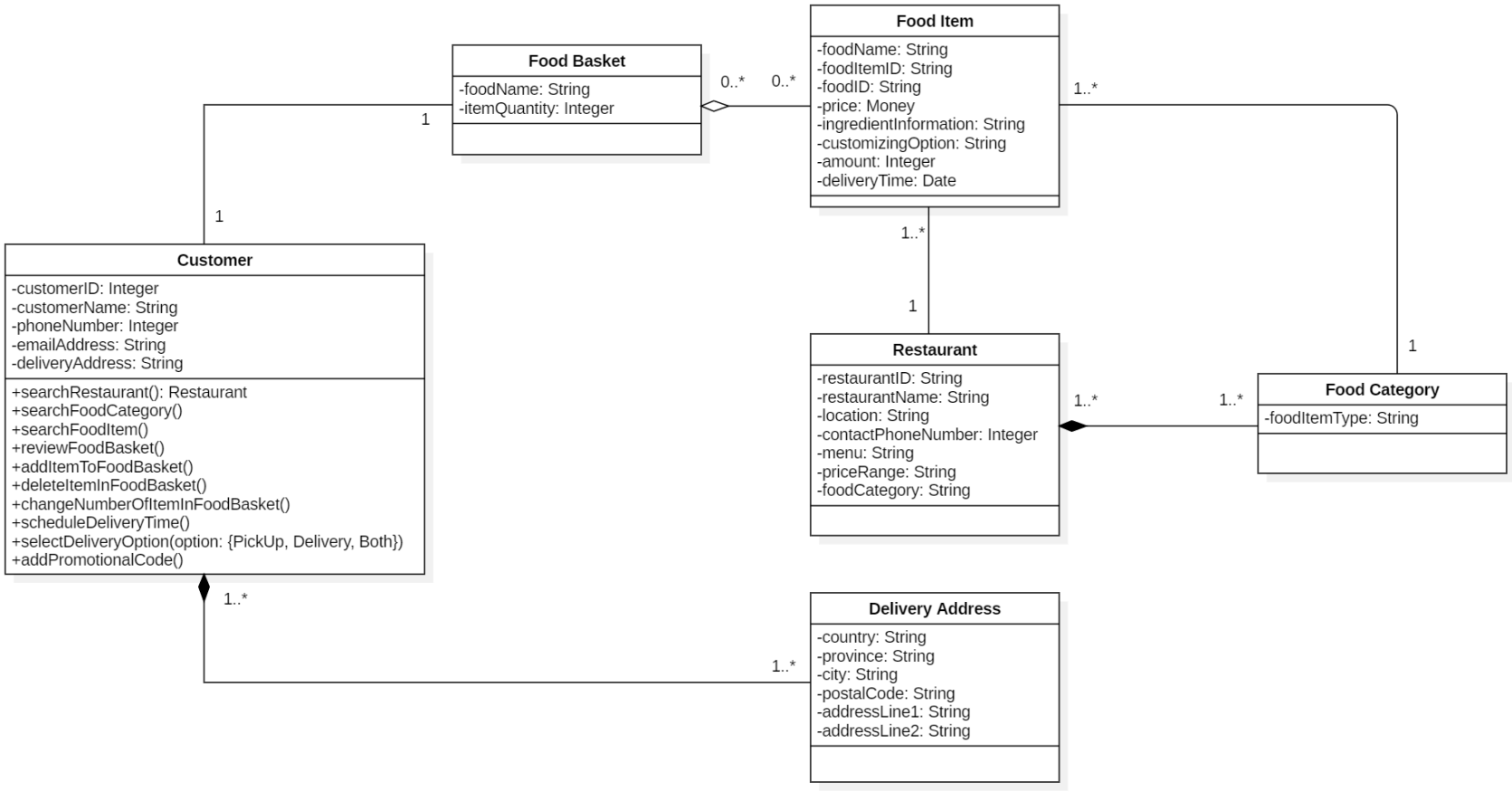
The delivery address is also filled in this stage with address line, postal codes, city, province, and country. One customer can have multiple delivery addresses, and one delivery address is associated with one or many customers.

The system also keeps track of the restaurant information, which includes Restaurant ID, name, location, contact phone number, menu, price range, and food category. The food category is identified by a given name, while the food item is identified by food name, ID, price, ingredient information, customizing option, amount, and delivery time. Each restaurant must have at least one food category, and each food category can belong to one or many restaurants. Each restaurant can sell one or many food items, but one food item belongs to only one restaurant. One food item belongs to only one food category, and one food category can include one or many food items.

4.3 Class Diagram Assumptions

ID	Assumption
A1	The restaurants [U1B] filled in the information required and agreed to their data being stored and displayed on the system [D1]
A2	The customer [U1A] registered an account to use the system [D1]
A3	The customer [U1A] consent to their data being stored in the system [D1]
A4	All the food items [D7] contain the information required by the system [D1]

4.4 Class Diagram



5. Activity Diagram

5.1 Used Functional Requirements

FR3 Review current order

FR3.1 The system [D1] shall allow the customer [U1A] to review the food basket [D31] at any given time.

FR3.1.1 If the customer [U1A] adds items [D7] to the food basket [D31], the system [D1] shall allow the customer [U1A] to view:

FR3.1.1 Items [D7]

FR3.1.2 Price [D14]

FR3.1.2 Delivery fee [D17].

FR3.1.2 If the customer [U1A] adds no item [D7] to the food basket [D31], the system [D1] shall display the message "Your food basket is empty"

FR3.2 The system [D1] shall enable the customer [U1A] to modify the order in the food basket [D31] with the following functions:

FR3.2.1 Delete items [D7].

FR3.2.2 Change the number of items [D7].

FR3.2.3 Schedule the delivery time [D20].

FR3.2.4 Select delivery options [D16].

FR3.2.5 Add promotional code.

Rationale: The system [D1] should allow the customer [U1A] to check the food basket [D31] whenever the customer [U1A] wishes to see it. The system [D1] will let the customer [U1A] see the items [D7] in the food basket [D31], the price [D14], and the delivery fee [D17]. The system [D1] will also let the customer [U1A] modify the order if they want to delete items [D7], change the number of items [D7], schedule the delivery time [D20], select delivery options [D16], and add a promotional code.

Source: ConcluDishes Business Case (2022)

Author: AS

Priority: High (P1)

Dependency: [FR1], [FR2], [NFR1].

History: 28/09/2022 (First version), 29/09/2022 (Revision), 2/10/2022 (Final version)

FR4 Provide delivery and payment details

FR4.1 The system [D1] should require customer [U1A] to select the preferred delivery option [D16].

FR4.1.1 If the customer [U1A] chooses the pick-up method, the system [D1] should request the customer [U1A] to select a pick-up time [D27].

FR4.1.2 If the customer [U1A] chooses the delivery method, the system [D1] should request the customer [U1A] to fill in their delivery address [D19].

FR4.2 The system [D1] shall require the customer [U1A] to select one out of the preferred delivery options [D16] below:

FR4.2.1 Credit card.

FR4.2.2 Debit card.

FR4.2.3 iDEAL

FR4.2.4 PayPal.

FR4.2.5 Apple Pay.

FR4.2.6 Google Pay.

FR4.2.7 Cash on Delivery.

FR4.3 The system [D1] shall redirect the customer [U1A] to the portal of their selected payment method [D18] from [FR4.2].

Rationale: For the system [D1] to function correctly, it is essential that customers [U1A] can conveniently place orders and pay for them, making this functional requirement a high priority [P1]. The customer [U1A] must specify their preference [D16] for receiving food item [D7] and payment method [D18].

Source: ConcluDishes Business Case (2022)

Author: YJ

Priority: High (P1)

Dependency: [FRX.5], [FR1], [FR5], [NFR4], [NFR5]

History: 27/09/2022 (First version), 29/09/2022 (Revision), 01/10/2022 (Final Version)

FR5 Place an Order

FR5.1 The system [D1] should require the customer [U1A] to request an order from the restaurant[U1B].

FR5.1.1 The restaurant [U1B] may confirm the requested order.

FR5.1.2 The restaurant [U1B] may deny the requested order.

FR5.1.3 The system [D1] should forward this message to the requesting customer [U1A].

FR5.2 The system [D1] should require customer [U1A] to select one of the filled delivery address(es) [D19]

FR5.3 The system should require the customer [U1A]to transfer the expense [D28] and the delivery fee [D17].

FR5.3.1 The system [D1] should provide the customer [U1A] with the option to choose their preferred payment method [D18].

FR5.3.2 The system [D1] should guide the customer [U1A] to the following steps to fulfil their selected preferred payment method [D18].

FR5.3.3 The system [D1] should transfer the deposit [D29] to the holding account[D30] after the customer [U1A] confirms the delivery confirmation [D21]

FR5.3.4 The system [D1] shall receive the customer payment confirmation [D22]

FR5.3.5 The invoice [D23] will be sent to the customer [U1A] once the payment is completed.

FR5.4 The system [D1] should provide the restaurant [U1B] with the delivery information [D28] of the customer [U1A].

FR5.5 The system[D1] should inform the customer [U1A] once the restaurant [U1B] ships the food items [D7].

Rationale: To make the system [D1] function properly, it is necessary that customers [U1A] can easily place orders, making this functional requirement a high priority [P1]. The customer [U1A] must request orders from the restaurant [U1B] to buy food items [D7]. After the restaurant [U1B] confirms the requested order, the customer [U1A] can fulfill the delivery information [D21], after which the customer can transfer the deposits [D29].

Source: ConcluDishes Business Case (2022)

Author: XC

Priority: High (P1)

Dependency: [FRX.6], [FR4], [NFR1], [NFR5]

History: 27/09/2022 (First version), 29/09/2022 (Revision), 02/10/2022 (Final Version)

5.2 Rationale for the Activity Diagram

Activity diagram is suitable for designing procedural logic, business process and workflow (Sommerville, 2011), which matches our needs in illustrating FR3, FR4, and FR5. This type of diagram allows us to document the behavior of customers, restaurants and the system across multiple use cases. Another advantage of the activity diagram lies in its ability to describe parallel activities using forks and joins. And we have parallel activities in the system's displaying added food items and the user's modifying food baskets.

We did not choose the sequence diagram because our case includes several scenarios across different use cases and paths with conditions. The state-machine diagram is not suitable for us either because it is designed to document the behavior of a single object across scenarios. In our case, we have multiple objects, and the emphasis is not on the state of the objects.

5.3 Activity Diagram Description

The diagram begins when the customer starts reviewing the food basket. The system reacts based on the food items added. If no items are added, the system displays the message: "Your food basket is empty". The customer will be taken back to review their food basket. If at least one item is added, the system should display the price, items, and delivery fee.

The system then allows the customer to delete and/or change the number of items, as well as manage delivery options and time. The customer can also add promotion codes if applicable. If no items are left at the end of this step, the system will display the message: "Your food basket is empty". If at least one item remains, the customer should provide payment and delivery details. In this step, we assume providing delivery and payment details is one activity. If the payment and delivery detail provided is invalid, the system will direct the customer back to review their food basket. If the payment detail is valid, the restaurant will receive the requested order, and they will decide whether to confirm or deny the order.

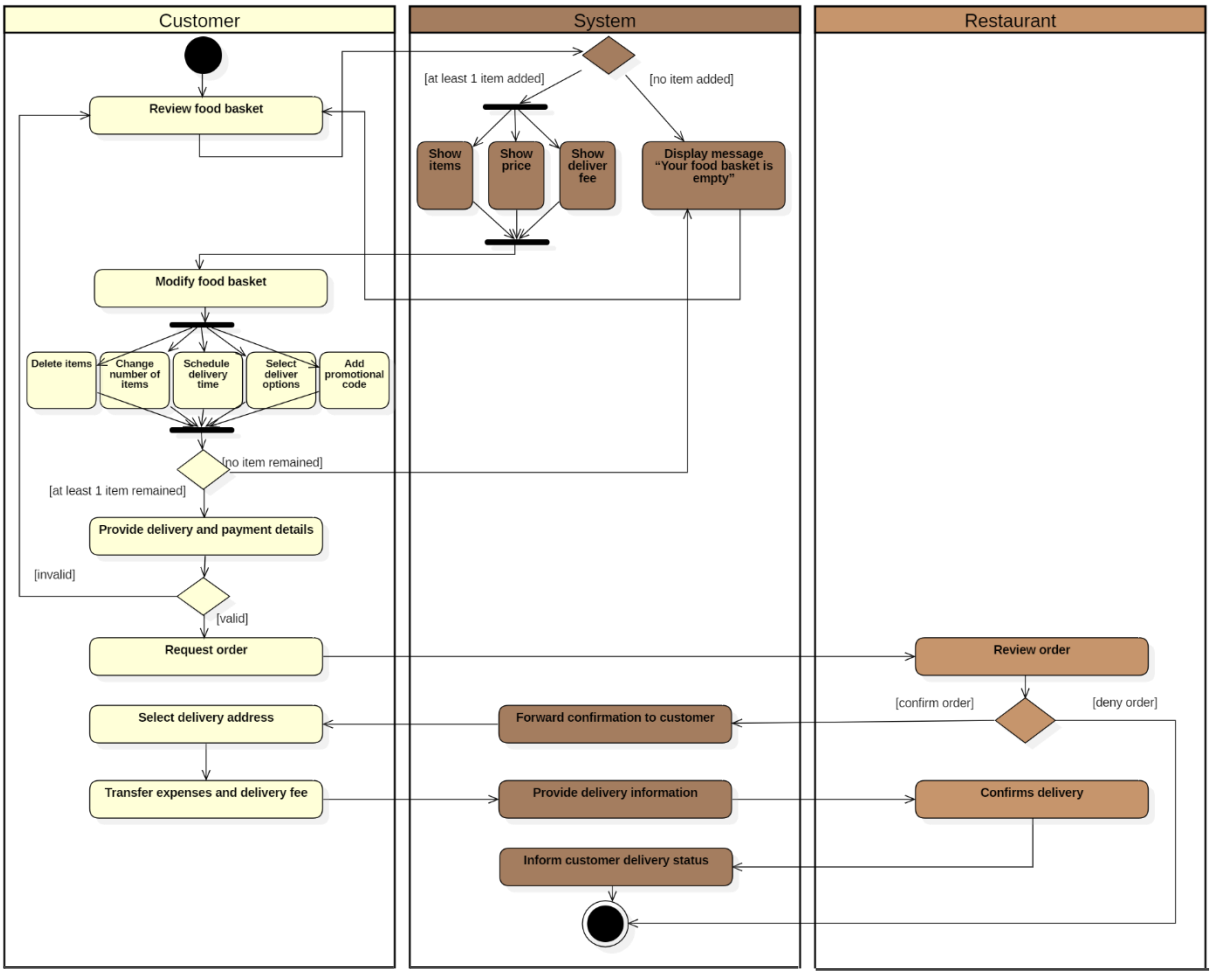
If the restaurant denies the order, the process will end. If the restaurant confirms the order, the system should forward a confirmation to the customer and request them to select a delivery address (assuming they have already provided one or several addresses beforehand). The customer will then confirm and transfer the expenses to the system. Then the system shall provide the

restaurant with the delivery information, and when the restaurant has confirmed the delivery, the system shall notify the customer of the delivery status. The process then ends.

5.4 Activity Diagram Assumptions

ID	Assumption
A1	Activity concerning FR4 only includes providing delivery and payment details.
A2	Customers [U1A] always review the food basket before providing delivery and payment details
A3	Customers [U1A] providing delivery and payment details step happens when customers want to fill in new delivery and payment information.

5.4 Activity Diagram



Reference

Antala, K. (2019, February 28). 6 Customer Experience Aspects That Any Food Delivery App Should Ensure. *CMARIX Blog*. https://www.cmarix.com/blog/6-ux-necessities-food-delivery-apps/

Sommerville, I. (2011). *Software Engineering, 9/e*. Dorling Kindersley.