FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

Restaurant System

Analysis

Documentation of a project for the purpose of the course BIE-SI1.

Authors:
Virna Ioana
Zaharia Tudorita
Varga Alexandra
Hutnikov Vladyslav

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

Contents

1. Specification Model

The goal of this project is to create a system for a restaurant to help customers find information about meals available in the menu. The system should also provide the following functionalities such as booking a table inside the restaurant, allow the customer to create online orders such as deliveries or direct pick-up.
After the creation of a delivery, the company handling the transport will get notified by the system.
Furthermore, the restaurant will have a recruiting system embedded on which candidates can apply for a job offer. The desktop application ought to provide a function for recruiters to create job offers.
The system should also keep record of current orders and table reservation.
Restaurant staff should have the following functions of managing reservations(edit/create/cancel/view), canceling orders, view deliveries and pick-ups to be done.
Customer should be able to view the meals, create deliveries/pick-ups and to manage the reservation for a table.

2. Business Process Model

In the domain of the restaurant there are mostly four business processes essential for the customer - reserving a table, hiring the staff, ordering food online for pickUp and deliveries handled by restaurant.

2.1 Business Domain Model

The most important domain objects in the Restaurant domain are highlighted in the following schema. The interaction happens between customers and meals having in between different objects such as reservation and order.

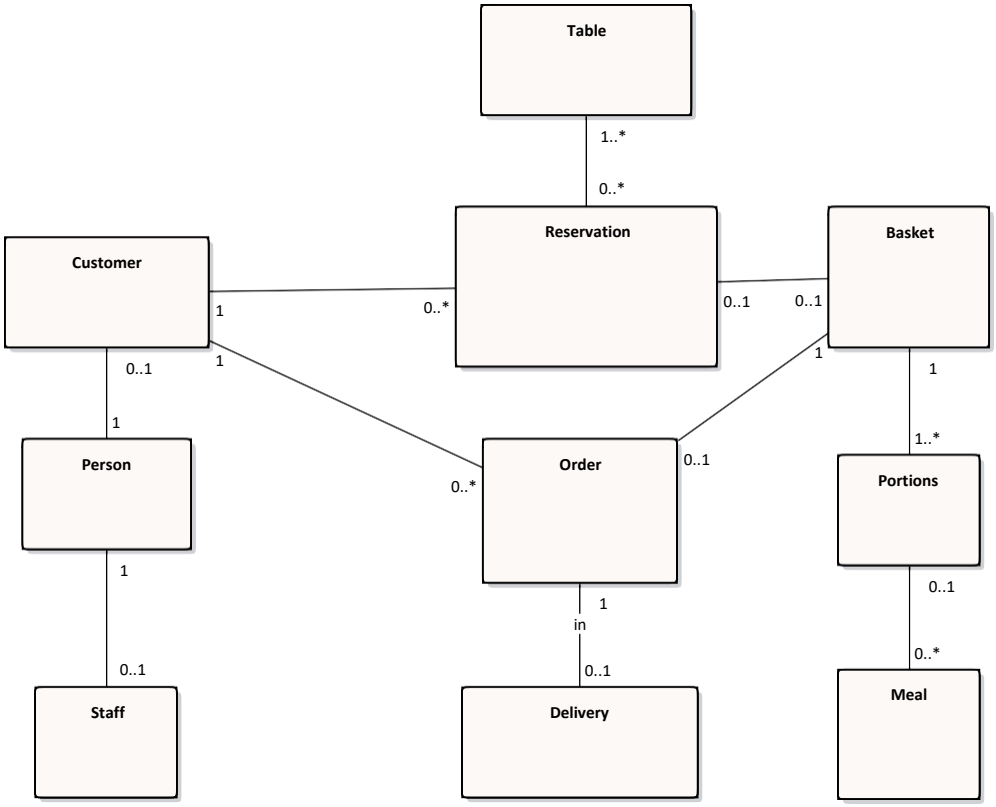

Figure 1 - Domain Model

2.2 Booking a table

Customer creates a reservation then he takes a decision if he wants to choose the meals now or at the restaurant.
In the first case of choosing the meals in the restaurant, the process of reservation is finished by a confirmation of the system.
In the other case of choosing the meals now, restaurant staff must validate the availability of meals.
If restaurant staff doesn't confirm the availability, then the client must review and change the meals. In the other case, this step is followed by another choosing payment method. If the customer chooses to pay at restaurant, the process of reservation is finished by a confirmation of the system.
Also, he can pay online, so the system will send him an invoice to pay. After the customer finish the payment, a confirmation is sent by the system which closes the reservation.

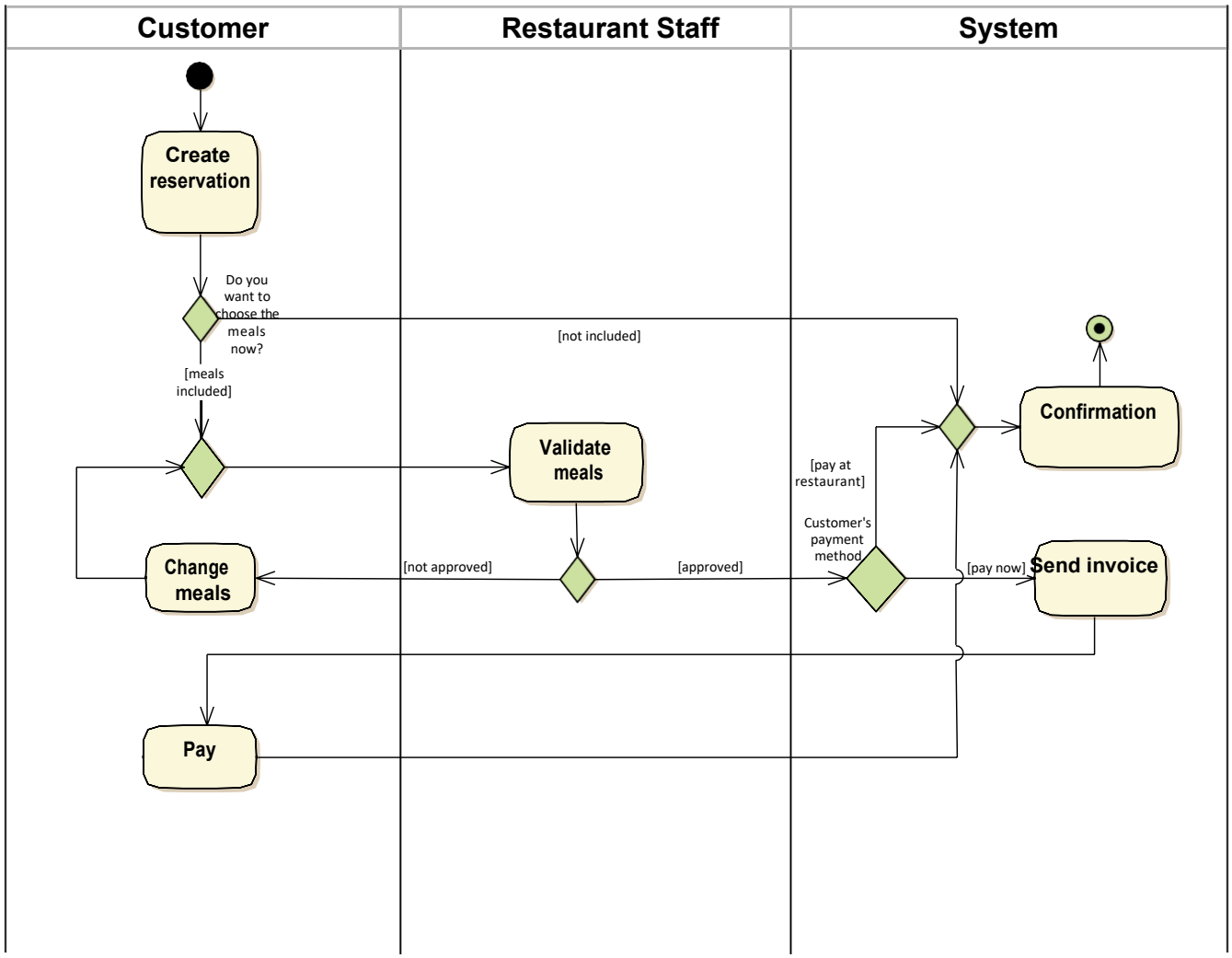

Figure 2 - Book a table

2.3 Order & Pick-up

Customer creates an order then the restaurant staff takes a decision to either validate it or not. In the case of rejecting the order, system sends a notification to the customer.
In the other case of validating the order, system sends a confirmation and then the restaurant starts preparing the food in parallel with choosing if the order is a delivery or a pick-up. If it's a delivery, system sends a notification and delivery staff will send a driver to the restaurant. In the other case, for a pick-up, client goes to the restaurant to pick up the food directly from there. Once the parallel actions are finished, the restaurant decides whether it's a pickup or delivery once again and in case it's the driver, it will give the food else it will give to the customer and the activity ends by paying the order by customer in both cases.

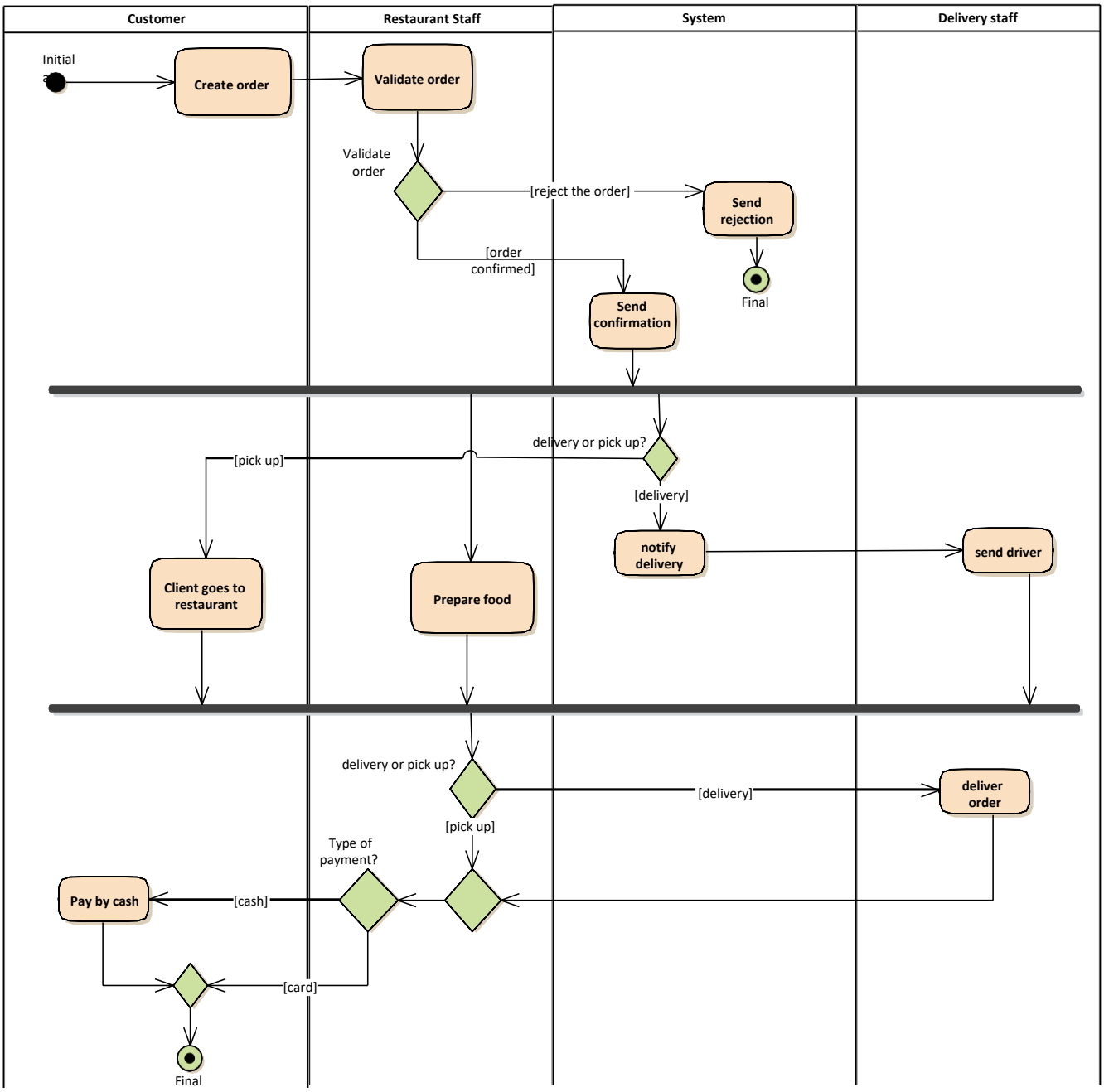

Figure 3 - Order and pick-up

3. Requirements Model

3.1 Functional Requirements

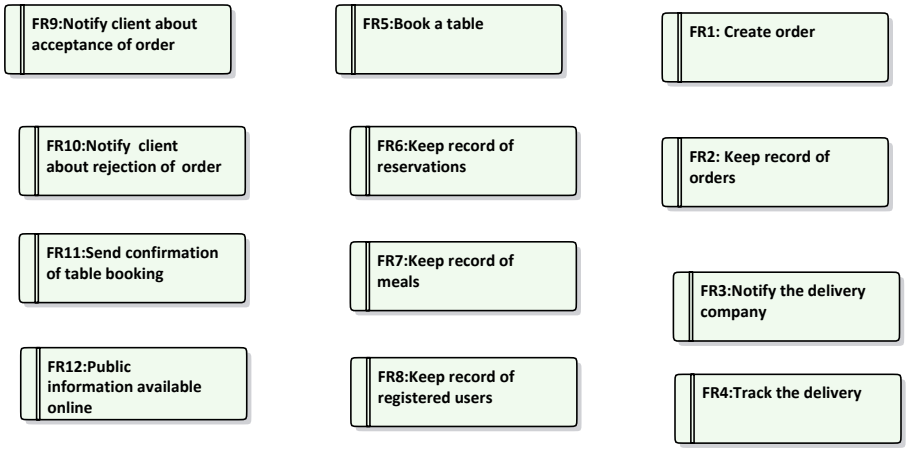

Figure 4 - Functional requirements

3.1.1 FR1: Create order

Priority: High

The system provides a function for creating an order. Client will choose based on his preferences if he wants a delivery or pick - up. For a delivery, customer must specify the address and a phone number.
The order will be handled through the web application.

3.1.2 FR2: Keep record of orders

Priority: High

The system will keep record of customer's current deliveries and pick-up orders. An order should have information about client's address, order id, list of meals, date.
The deliveries are managed only in the desktop application of the restaurant.

3.1.3 FR3: Notify the delivery company

Priority: High

The system will automatically notify the delivery company about a new order that was placed.

3.1.3 FR4:Track the delivery

Priority: Medium

The system can monitor the route of the order and the stages until it'll be delivered. The restaurant staff will the check the status of order in the desktop application.

3.1.4 FR5:Book a table

Priority: High

This is the main functionality, when the clients can book a table to the restaurant by the web application.
The system will provide a function for the customer to book a table. At the moment of creation, customer needs to set the exact time and date of the booking and also the number of people who will attend on that particular day. Moreover, the system must check if there is a table available for that particular time otherwise the reservation will be rejected.

3.1.5 FR6:Keep record of reservations

Priority: Medium

The system will keep records of reservations for each customer.
For each reservation, the following information will be needed to be stored such as reservation time & date, number of persons who will come, reservation id, customer's personal information.
The reservations are managed by restaurant staff and also by the client using the web application. Customer will be able to create, delete or modify reservation as for the restaurant staff, he can find the reservation or cancel it.

3.1.6 FR7:Keep record of meals

Priority: Medium

System keeps record of meals ordered using the desktop app. For each meal, the system should know name of meal, price, and quantity.
Restaurant staff can add new meals, modify information about meals or delete a certain meal.

3.1.7 FR8:Keep record of registered users

Priority: High

The system saves the details about the users registered to the restaurant application. System must keep the following information about a user such as e-mail, postal address, name, and person id.
Restaurant staff will be able to find a specific customer, see his profile and update it. Furthermore, customer can register in the system and update some information about his account.
The management of users is done in the web application.

3.1.8 FR9:Notify client about acceptance of order

Priority: Medium

After making a reservation clients will wait until it'll be approved by the system, they'll receive a notification of acceptance if the order could be taken and prepared by the restaurant staff.

3.1.9 FR10:Notify client about rejection of order

Priority: Medium

After making a reservation clients will wait until it'll be approved by the system, they'll receive a notification of rejection if the order couldn't be taken or prepared by the restaurant staff.

3.1.10 FR11:Send confirmation of table booking

Priority: Medium

After making a table booking clients will wait until it'll be approved by the system, they'll receive a notification of acceptance if the booking can take place or not.

3.1.11 FR12:Public information available online

Priority: High

Users can find information about each meal, viewing a list of meals or the meal's details. Each meal has information about ingredients, allergens, price, and the quantity in grams.
The information will be available online through the web application.

3.2 Non-Functional Requirements

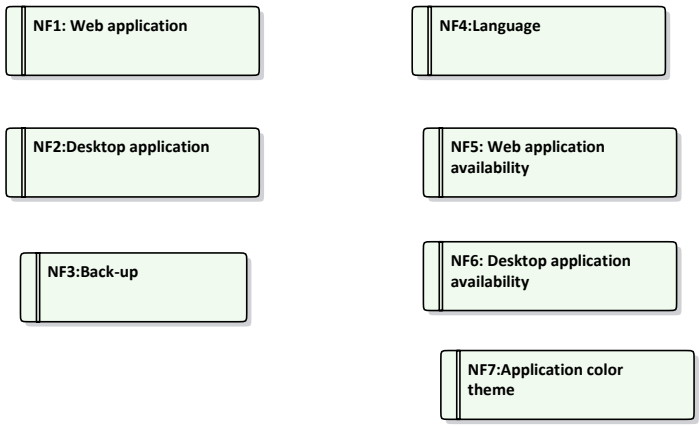

Figure 5 - Non-Functional Requirements Model

3.2.1 NF1: Web application

The information about restaurant meals as well as their availability should be present in our web application.
The web server will be running on the same server as the central database, in real life it should be located in the restaurant building.

The application ought to be running in Chrome 94+.

3.2.2 NF2:Desktop application

The system handles the management of order, deliveries, cancellations,
records. The app should be available online directly connected to the central
database.
The system doesn't use authentication, consequently it can be used by any type of user.
Any user of the system will be able to take a view at the orders, confirm/reject them etc.

3.2.3 NF3:Back-up

The data about client's orders and reservation must be backed up daily in a separate database.

3.2.4 NF4:Language

The system will be localized in Czech and English. The Czech and English translations will be delivered with the application.

3.2.5 NF5: Web application availability

The system should be available at least 90% of time for bookings and orders.

3.2.6 NF6: Desktop application availability

The desktop app must be available at least 99% of time when the restaurant is open.

3.2.7 NF7:Application color theme

The application ought to be realized in the theme of red and white.

4.Use Cases Model

In this chapter it is described the utilities of each actor through a series of use cases.
The most important ones are the following:

- Create order
- Register customer
- Add a reservation
- Cancel reservation
- Validate order
- Cancel order

3.3 Use Case Model

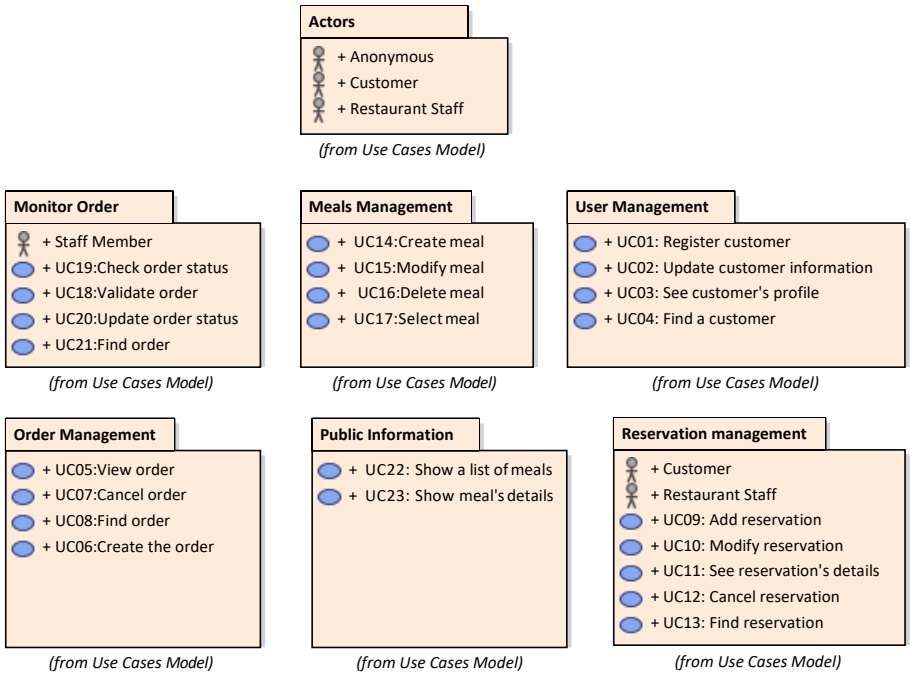

Figure 6 - Use Case Model

3.4 Actors

This package describes the actors of the system.

4.2.1 Anonymous

A person willing to register for a restaurant and also he can view the menu list. He can find public information about the menu, his reservation and his account using the web application.

4.2.2 Customer

A person who can see the restaurant menu, check the meals, view the products, book a table, create an order.
Customer must also be registered before via the web application.

4.2.3 Restaurant Staff

The person who can manage the orders, check the order's status, update information about availability of a table. Furthermore, staff can view information about orders, reservations. Staff can also manage the list of meals through the desktop application.

3.5 User Management

This package defines use cases regarding the management of registered users/customers.

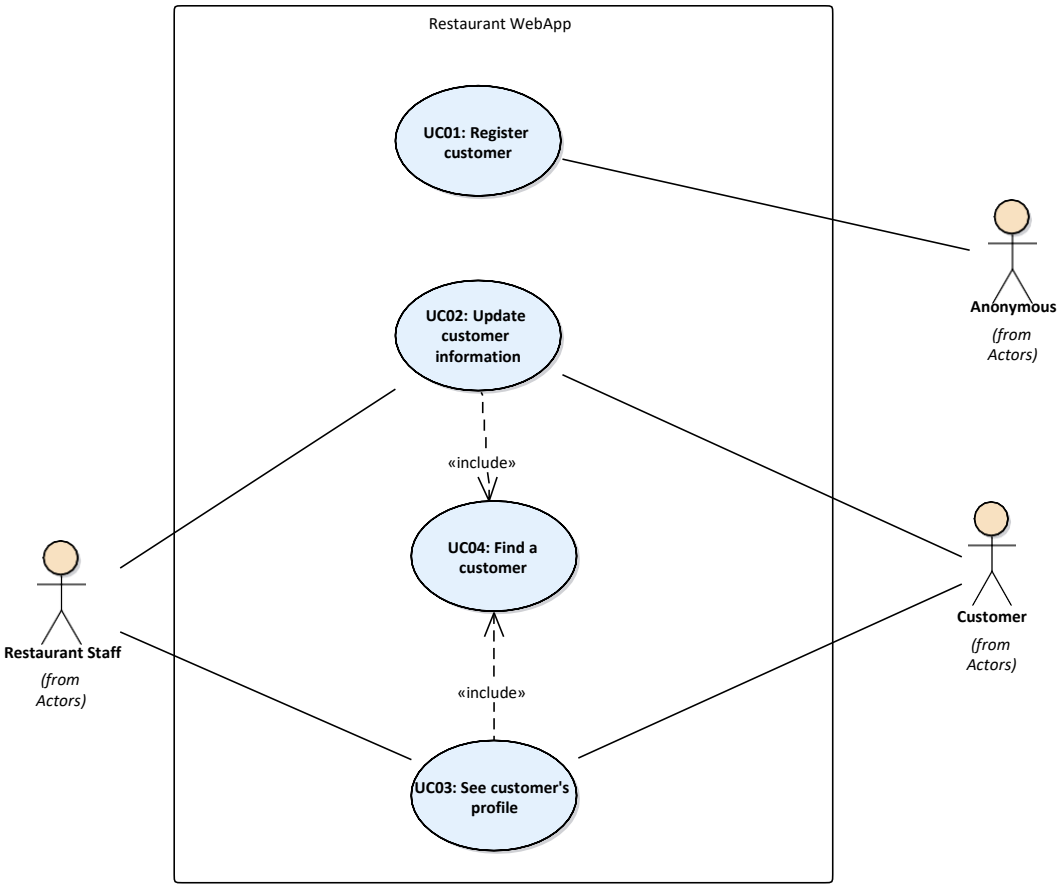

Figure 7 - Customer Management

4.3.1 UC01: Register customer

The customer can register specifying his name, e-mail address and postal address. The system provides unique ID.

Basic Path: Customer registration

The customer can register to restaurant application specifying his name, e-mail address and postal address. The system provides unique ID for the customer.

1. The use case starts when the customer enter the application and wants to register.
2. The system shows the registration from with the following fields: first name, last name, e-mail address, postal address. All the fields are mandatory.
3. The new customer fills in all the fields
4. The customer finish the registration using "Register" button.
5. The system validates the provided information.

6. The system generate a new unique ID for the newcustomer.
7. Thesystemsavesthenewcustomer'sinformationandpromptshimaboutsuccessfulreaderregistration.

Exception: Duplicate e-mail address
This alternate scenario defines the processing of the situation, when the system detects that the e-mail address specified in the registration form already exists in the database.
1. The system detects duplicate e-mail address specified in the registration form, which already exists in the database.
2. The system shows an error message about the existing e-mail address, including the name of the reader registered with the address.
3. The new customer checks the name of the registered reader and confirms the identity of the requesting customer. In
the case of already being registered, the system cancels the registration. In the case of a different customer, the scenario returns to the step 3 of the main scenario and the customer inserts a different e-mail.

Exception: Mandatory data missing
This alternate scenario defines the processing of missing mandatory data in the registration form detected in the step 4 of the main scenario.
1. The system detects missing mandatory data in the registration form.
2. The system shows message about the missing mandatory data and highlights the appropriate fields in the form.
3. The scenario returns into the step 3 of the main scenario.

4.3.2 UC02: Update customer information

The customer or the restaurant staff can update information about a registered user after finding him. The user can update only the customer's name, e-mail
address and postal address. It is not possible to update the customer's ID.

4.3.3 UC03: See customer's profile

The user can see the profile of a customer after searching him/her.
In the profile, the following information are listed:
• Information about the customer (name, ID, addresses)
• List of all customer's reservations, including all their details

4.3.4 UC04: Find a customer

The user can find a record of the customer by his/her name or ID

Basic Path: Finding a customer
This scenario describes the process of finding a customer based on his/her information.
1. The use case starts when the user needs to find a customer in the system. The user uses the button to find customer .

2. The system shows a search dialog with the following fields: first name, last name, ID.
3. The user fills the appropriate fields and confirms.
4. The system searches the database for the customers according to the specified data. The system matches the

registered customers with the appropriate values, including partial matches.

3.6 Order Management

This package describes the creation, cancellation, or modification of an order.

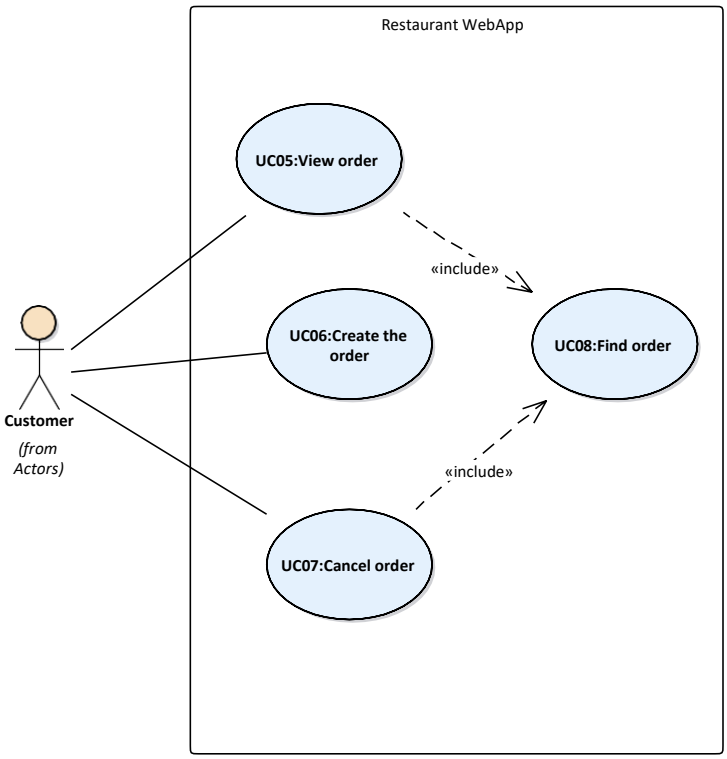

Figure 8 - Order

4.4.1 UC06:Create the order

Use case depicting how to create an online order such as a pick directly from the restaurant or a delivery.

Basic Path: Basic Path

Basic path for creating an online order via the web site page where customer must put his personal data for a delivery.

1. Actor clicks on button "Create order"
2. System displays form with fields about address, phone number, name and choosing it it's a pick up or delivery
3. Actor fills in data fields and clicks on button "Create order"
4. System sends a notification about order beingcreated

Exception: Missing data path
Missing path data refers to the case when the customer has chosen for example a pick up and didn't fill in the correct information for phone number ,email or name. It can also refer to when customer chose delivery and has missing data for address field.

1. System sends a notification about some fields being incorrect
2. Scenario goes back to step 2

4.4.2 UC08:Find order

Use case depicting how to find an order in the web site of the restaurant.

Basic Path: Basic Path
Basic path for finding an order which is already registered in the system.

1. Actor clicks on button "Find order"
2. System display a new window with a field for id.
3. Actor inserts the id number of order and clicks on button "Search order"
4. System displays window with order's details

Exception: Wrong id path
Exception path in case the order id inserted was not valid.

1. System displays notification about order being notfound
2. Scenario goes back to step 2

4.4.3 UC05:View order

Use case depicting how to check the status of an order who is processing now.

4.4.4 UC07:Cancel order

Use case depicting the steps of searching for an online order and then trying to cancel it.

3.7 Reservation management

This package defines use cases regarding the process of booking a table.

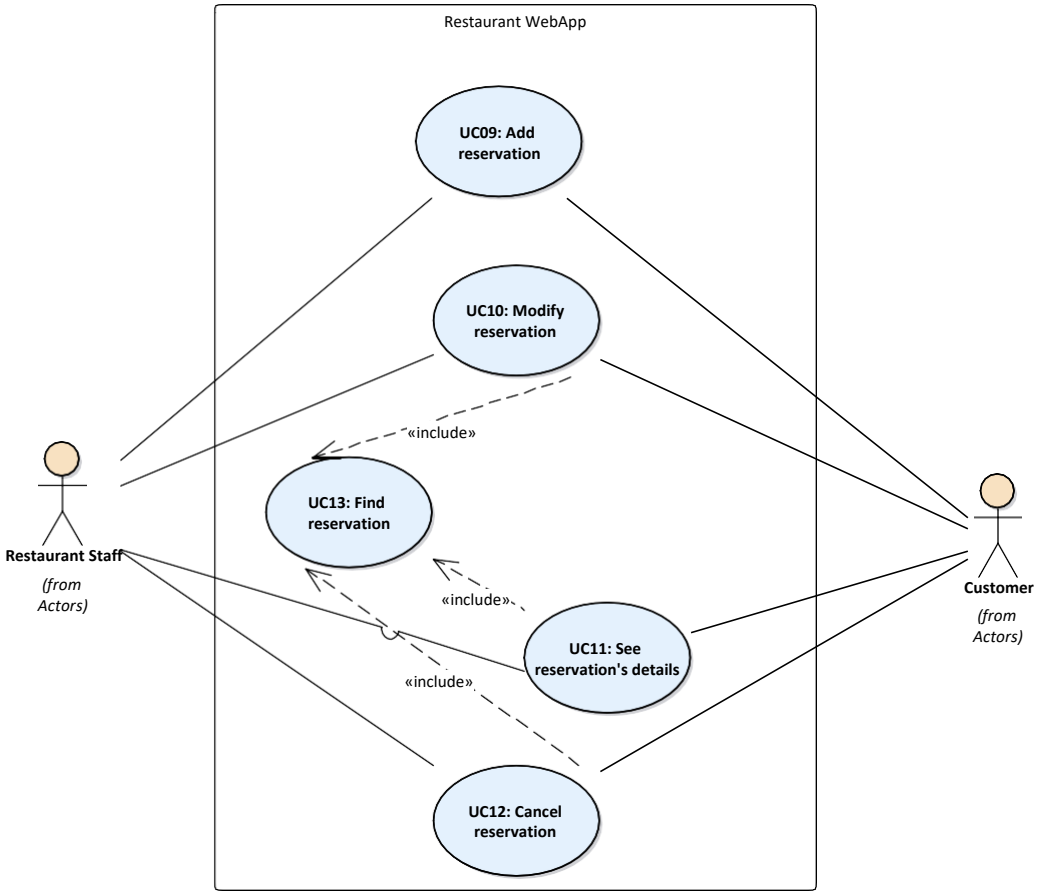

Figure 9 - Reservation management

4.5.1 UC09: Add reservation

The user can add a new reservation.

Basic Path: Adding a reservation

1. The use case starts when a user wants to add a new reservation into the system. The user uses the action to add a reservation.
2. The system shows a form to add information about the new reservation with the following fields: date, time, and number of persons.
3. The user fills in the information about the reservation.
4. The system confirm if the reservation it will be possible.
5. If the user wants to choose meals, he/she will press the button "Choose meals". This action isn't mandatory.
6. The system shows a form from where the customer can choose his meals.
7. The system notifies restaurant staff about the meals.
8. Restaurant staff confirms the meals.
9. If the user wants to pay now, the system will send an invoice to customer for payment.
10. The customer finish the payment
11. The system generate a new unique ID for the actual reservation.
12. The system sends a confirmation to Restaurant staff and to customer with reservation's details.

Exception: Any possibility

This alternate scenario describes the processing of a reservation addition, when there are no possibilities with date, time and number of persons introduced.

1. The scenario starts when the restaurant doesn't have any possibility regarding information introduced.
2. The system shows an error message that the user must introduce different information.
3. The scenario return to the step 2

Alternate: No meals

This alternate scenario described the processing of a reservation addition when the customer doesn't want to choose meals online.

1. The scenario starts when the user doesn't want to choose the meals online.
2. The scenario jumps to the step 11 of the main scenario.

Alternate: No pay

This alternate scenario described the processing of a reservation addition when the customer doesn't want to pay online.

1. The scenario starts when the user doesn't want to pay online.
2. The scenario jumps to the step 11 of the main scenario.

4.5.2 UC10: Modify reservation

The user can edit information about an existing reservation in the database.

4.5.3 UC11: See reservation's details

The user can see detail of a reservation.
In the book detail, the following information about the reservation are shown:

- Information about the reservation (date, time, number of persons, meals, payment)
- The main customer who made the reservation

4.5.4 UC12: Cancel reservation

The user can cancel a reservation.

4.5.5 UC13: Find reservation

The user can find a reservation by its ID or the main customer who made the reservation.

Basic Path: Finding reservation

1. The use case starts when a user wants to find a reservation. The user uses the button to search for a reservation.
2. The system shows a list of reservations with the following fields: ID and the main customer who made the reservation.

The system also shows filters for filtering the reservation with any of the displayed values.

3. The user selects one of the reservations in the list.

Alternate: Filtering

This alternate scenario describes the way of filtering the list of reservations.

1. The scenario starts when the user fills in values in the filters.
2. The user confirm the filtering.
3. The system filters out the displayed reservations according to the provided criteria.

3.8 Meals Management

This package describes how to create, modify, or delete a meal.

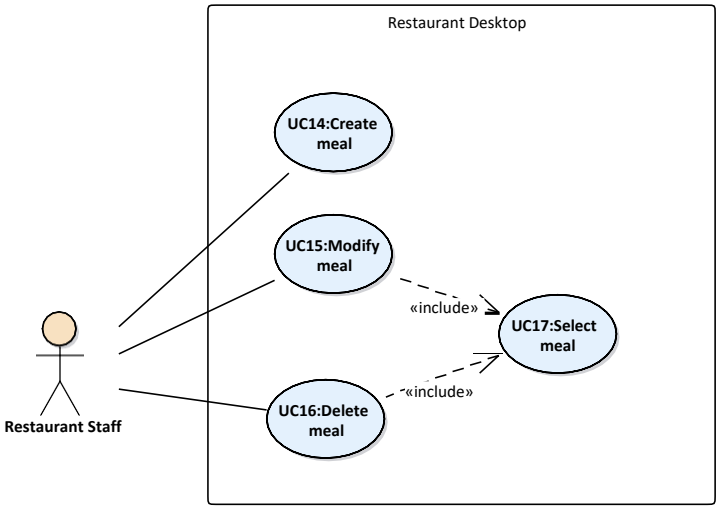

Figure 10 - Meals Management

4.6.1 UC14:Create meal

This use case highlights the creation of a meal.

Basic Path: Basic Path

The main path shows how to successfully create a meal and add it to the database.

1. Actor clicks on button "add meal"
2. System displays new window
3. Actor fills in data fields about meal
4. Actor clicks on button "create meal"
5. System displays a notification about the meal beingcreated

Exception: Wrong data path

1. System display warning about some fields beingincorrect
2. Scenario returns to step 2

4.6.2 UC15:Modify meal

This use case highlights how to modify an existing meal.

4.6.3 UC16:Delete meal

This use case highlights how to delete an existing meal.

4.6.4 UC17:Select meal

This use case shows how to select a meal.

Basic Path: Basic Path

1. Actor clicks on button "find meal"
2. System displays a new window for searching
3. Actor fills in name of the meal
4. Actor clicks on button "search meal"
5. System displays information about meal

Exception: Missing meal path

1. System displays notification that meal doesn't exist
2. Scenario returns to step 2

3.9 Monitor Order

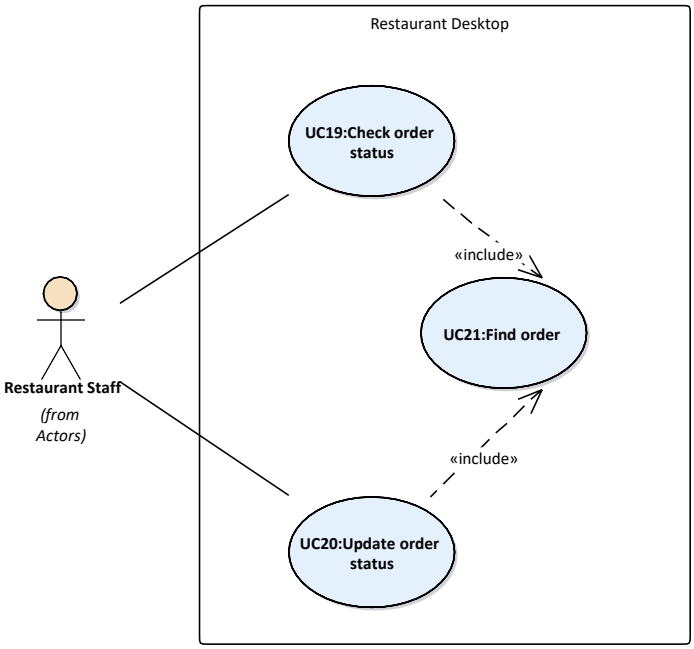

Figure 11 - Monitor Order

4.7.1 UC18:Validate order

Use case depicting how to

Basic Path: Basic Path
The staff reviews the order
The employee decides whether they will be able to do it, if so then the scenario goes on, if not, then the order is canceled
The system sends an order confirmation
Staff start preparing an order

1. The staff reviews the order
2. The employee decides whether they will be able to do it, if so then the scenario goes on, if not, then the order is canceled
3. The system sends an order confirmation
4. Staff start preparing an order

Exception: Reject order

4.7.2 UC19:Check order status

Use case depicting the status in case of the order such as created, in preparation, waiting for a driver, delivered or has just been canceled.

4.7.3 UC20:Update order status

Use case depicting how to update an order found in the search list and update the status for example from being prepared to delivered.

Basic Path: Basic Path
The employee enters the order code
The system returns information about the order

1. The employee enters the order code
2. The system returns information about the order

4.7.4 UC21:Find order

Use case depicting how to find an order for a specialized user in the desktop application of the restaurant.

Basic Path: Basic Path
Basic path include of how to find an order through the list of orders in the restaurant database.

1. Actor clicks on button "Find order"
2. Window displays a list with last orders created and data fields for searching a specific order.

3. Actor fills in data about an order such as id, data and clicks on button "Search order"
4. System displays a list of filtered orders who complete the requirements.
5. Actor clicks on the order.
6. System displays information about the order such as preparation time, client's information etc.

3.10 Public Information

This package defines use cases regarding the displaying of public information in the web application.

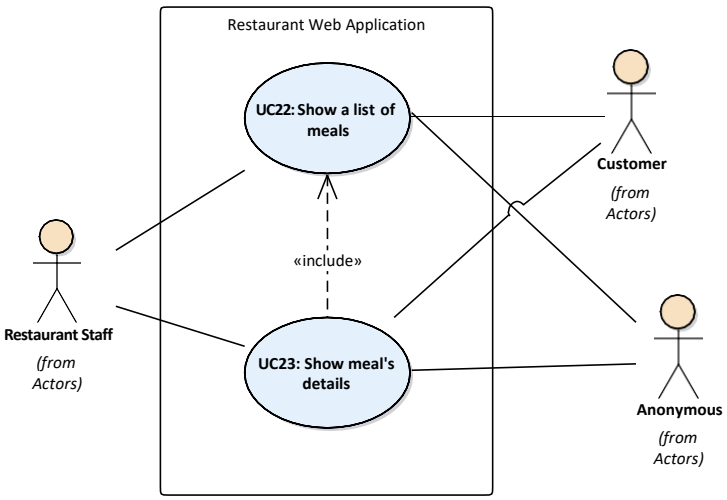

Figure 12 - Public information

4.8.1 UC22: Show a list of meals

The user can display the list of meals from the restaurant. The list of meals can be filtered using search function.

Basic Path: Show all

1. The user selects option to show a list of meals.
2. The system displays the search box where the search criteria (ingredients, name) can be filled in and confirmed.
3. The system displays the list of all meals.

Alternate: Search

The search path offers the option to fill in search criteria to restrict the displayed meals.

1. The user fills filtering information (name, ingredients) in the corresponding search boxes and confirms the search.
2. The system searches for reservation corresponding to the search criteria and displays them in the list.

4.8.2 UC23: Show meal's details

The user can display the detail of a meal including information about the ingredients.


FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

5. Domain Model

5.1 Restaurant

This section describes the domain of the restaurant, defining the properties and relations of the important objects of the domain.

The main entity is meal defining a dish consumable by a person. A basket can contain a list of meals having a total price. Each meal must have a name, price, and quantity.

There entity Customer as well as the Staff are special type of Person object. A person ought to have first name, last name and an unique ID.
Staff mainly occupies with accepting or declining Reservation and Orders.
The customer can create a reservation. In the reservation data structure there should be information about number of persons involved, the proposed date of reservation, status of the reservation etc.
Each booking is linked to a table in restaurant. Tables should contain data about the number of people and availability of the table in real-life.
Furthermore, customers can create orders represented by Order entity, having a phone number for contact and preparation time until finished.
Delivery is a special type of Order entity. Customers can also create deliveries but they should also put a delivery address .

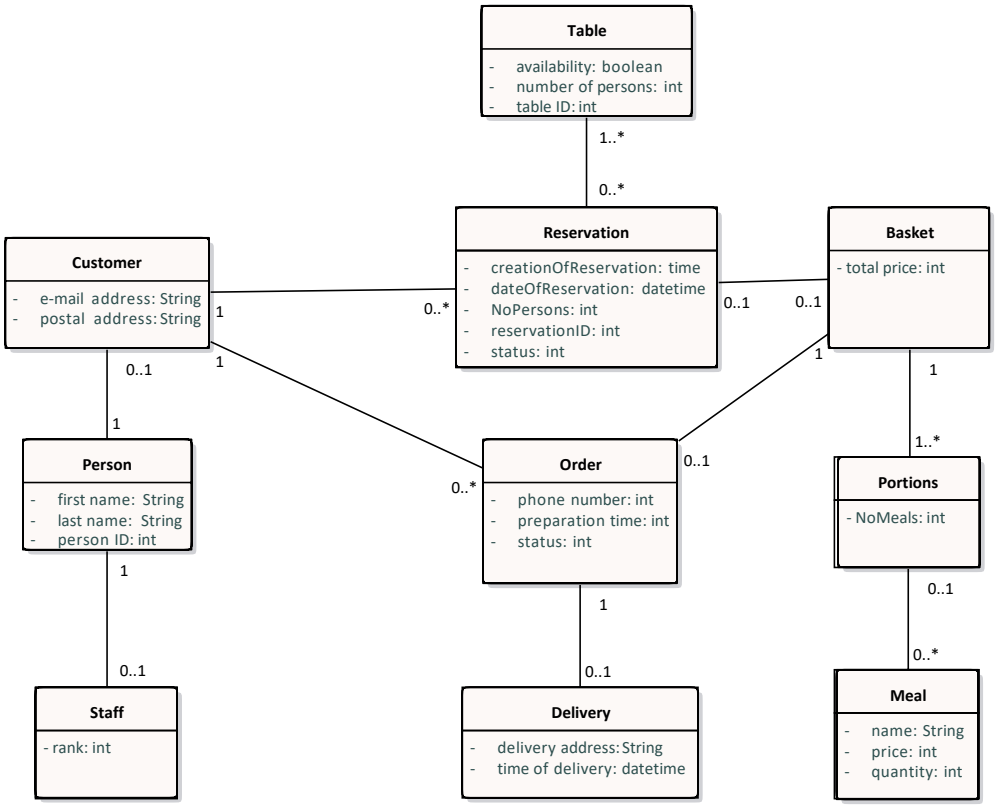

Figure 13 - Domain Model

5.1.1 Basket

Entity depicting a list of meals with the total price.

Attribute title	Description
total price	The total price of each meal added.

5.1.2 Customer

Customer is a special type of person who can create reservations and orders.

Attribute title	Description
e-mail address	The e-mail address of the customer needed for receiving information.
postal address	Postal address used for restaurant deliveries.

5.1.3 Delivery

Delivery is an extension of Order , having more fields needed for successfully delivering the list of meals.

Attribute title	Description
delivery address	The address for delivering the meals.
time of delivery	The supposed time of arrival for the meals to arrive at the destination.

5.1.4 Meal

Meal represents the dish available at the restaurant.

Attribute title	Description
name	Name of the dish.
price	Price of the dish.
quantity	Total quantity in grams of the dish.

5.1.5 Order

Order entity holds the data for making an online pick-up order.

Attribute title	Description
phone number	Phone number used for calling in case the order status is changed.
preparation time	The amount of time needed to prepare meals or the waiting interval for the customer until he can pick it up.
status	The state of the order such as opened, processing or closed.

5.1.5.1 State Machine

This diagram represents the possible states of an order. Furthermore, the order can be of type pick-up or delivery.

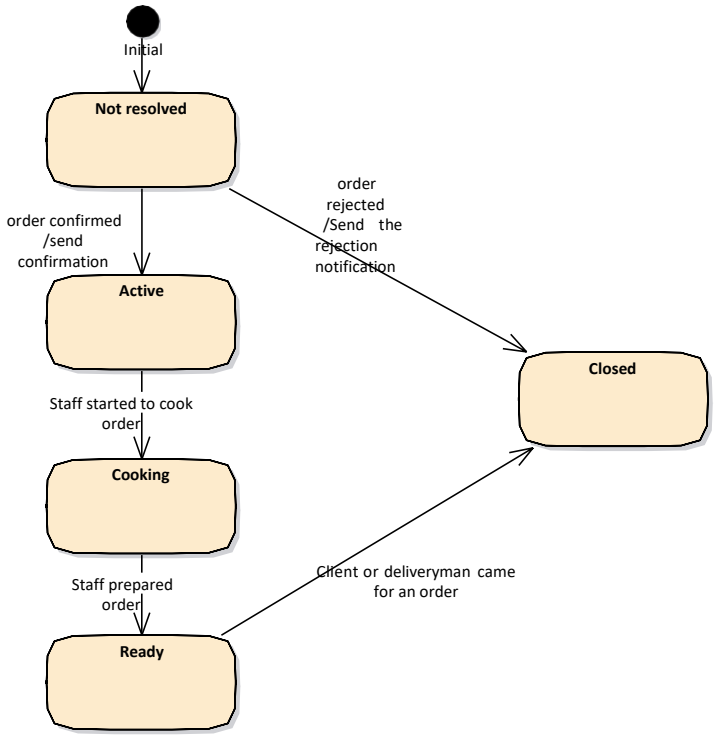

Figure 14 - State Machine

5.1.6 Person

A generic entity for a person interacting with the restaurant.

Attribute title	Description
first name	First name of the person.
last name	Last name of the person.
person ID	Unique ID of the person.

5.1.7 Reservation

Reservation represents the entity of booking a table for a specific time via the web site.

Attribute title	Description
creationOfReservation	Represents a timestamp for the date on which the reservation was created.
dateOfReservation	Date of reservation represents the supposed date of arrival at the restaurant.
NoPersons	Represents a quantifying attribute for the number of people present in the reservation.
reservationID	
status	The status highlights the state in which the reservation is such as open, processing or closed.

5.1.7.1 State Machine

This diagram describes the states of a reservation.
When the reservation is created, it is considered active. When the time for it is less then 2 days, the reservation is marked as Coming event and a notification e-mail is sent to the customer. If the reservation is kept, it will be consider in pending. After this, it must be paid by the customer.
In other case, if the client wants to canceled it, he can when he receive the notification.
After all these steps, the reservation will be consider finished (closed).

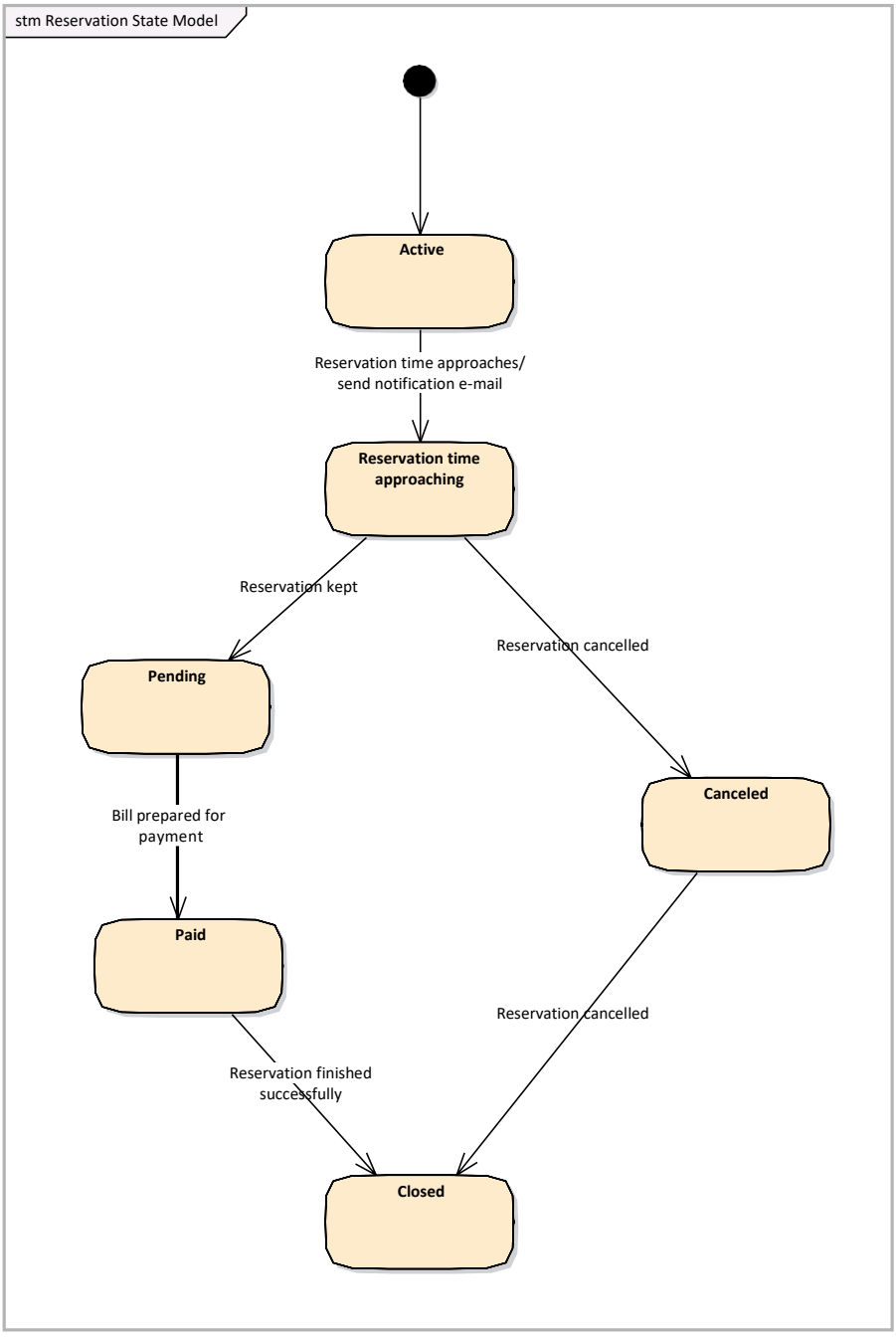

Figure 15 - Reservation State Model

5.1.8 Staff

Special type of Person having the capability to accept or decline orders.

Attribute title	Description
rank	Rank in the company.

5.1.9 Table

Table represents a physical table in the restaurant available for reservations.

Attribute title	Description
availability	
number of persons	
table ID	


FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE