



UC Leuven
Limburg

MOVING MINDS

COMPLETE ANALYSIS

Date: 24/12/2019

Jordy Van Esbroeck, Jonas De Winter and Baljit Singh

1.Contract

1. Name of the team: Bob's burgers

2. The team members

Jordy Van Esbroeck, Jonas De Winter and Baljit Singh (Team 5)

3. Roles

Who's going to perform what role?

Team leader: Jonas

Spoc(single point of contact): Jordy

Writer + taskmaster:

Baljit

Give a description of

the role

Team leader: maintains the peace in the team, distributes the tasks. Motivates the team and they are working properly. Quality control of the content. Makes sure all tasks are in good progress.

Spoc: communication with the team, lecturers and

the customer. Writer: write a report of the

meetings.

Taskmaster: organizing the meetings.

4. Collaboration:

What is good collaboration?

- Organising a meeting before working, so that we know the progress.
- We work together and are on the same page.
- Communicate with everyone.
- Work must be evenly divided: everyone knows what to do and against what deadline.
- Be punctual, if late let us know via message
- Maintain peace and fun.

How to collaborate?

- Communicate with team before taking a big decision: organising a meeting or via messages in group.
- Make a communication channel for communication (Discord + Messenger).
- We will listen to everyone with respect.
- Share problems on time, so we can find an alternative solution on time.
- No stressing, but planning!

5. The topic

Topic of the project: Mc Donald's

Short story:

The idea is to make a system which helps the Fast-food giant to easily maintain their stock as well as organising shifts, so the employees know when they work. We want to make sure that all the data is in one place, so it's easily accessible for the client (Mc Donald's) and developers. We also want to make sure we can check the popularity of the product of the day/week etc. This way we can make your stock more efficient without leftovers in the future.

Jordy Van Esbroeck



Jonas De Winter



Baljit Singh



2. Business analysis

1. The business

The CEO of McDonald's, Steve Easterbrook has decided to make the company more efficient.

McDonald's is a fast food restaurant where people can buy food such as burgers, French fries, salads etc. Also there is a possibility to buy soft drinks and milkshakes. This can either be purchased separately or in menu or in small packages (Happy meal).

Menu can only be purchased per person: 1 or more burger(s), French fries (to choose from 3 different sizes: small, medium or large) or 1 type of salad and added with a drink (to choose from small, medium or large) which can be refilled once.

Happy meal (for kids until 12 years old) : 1 burger, French fries (to choose from small, medium or large) with 1 kind of drink (to choose from 3 different sizes: small, medium or large) with 1 dessert (ice cream/yummy big cookie) and a toy (to choose between a boy toy or a girl toy).

Customers have the choice to either order traditionally by coming inside, placing an order, paying the bill and taking their order or through driveway: the customer comes in a car and places an order via a speaker (which is inside a wall) while he/she is seated in the car. Then the customer pays the bill (paid with cash or in various credit cards) and receives the order while seated in the his/her lovely car.

The customer comes in, goes to the counter, places an order, the worker adds the order to the cash desk, makes the bill. When the bill is paid, the cook receives the order and prepares it. When the meal is ready: the counter worker delivers it to the customer and smiles back. (not possible in any other way)

The employees just have to give in the order and when the customer pays, the ingredients needed to complete the order get subtracted from the total stock of that ingredient. When the stock gets under 25% the manager gets a message if he wants to order more stock or leave it like this and do it another time. He can also generate a stock overview to see how much waste he had last month so he can lower the stock of an ingredient

A big problem is that they now have no way to check the popularity of items so they don't have a clue what to advertise. That's why he requested that we find a solution for McDonald's, so they can keep track of all their information.

We will also have to be able to keep track of the employee's shift so it's easier for them to keep track of when they'll need to come in for work. They first have to log on so they can only see their own shift(s). Also by collecting all McDonald's information, they can get rid of all the paper copies and become even more organized and it'll be more environment friendly.

We just make a system for McDonald's itself so no website where the client can check all the products.

2. The stakeholders

give an overview of all stakeholders for your business

- Customers: Orders food and pays the bill
- Cashiers: Takes the order from the customer
- Cooks: Prepares the order
- Managers: Keeps everything in balance by taking strategic decisions and assigns shifts.
- Investors: Invests in the Mc Donald's and votes for future idea's
- Suppliers: Supplies ingredients and drinks to Mc Donald's

3. The PIECES framework

use the PIECES framework to define problems/opportunities and define the business requirements that are a result of this

	Problem/opportunity	solution
Performance	Manually ordering ingredient: because it can have human error, it takes too long.	Automatic orders of ingredients after confirmation from the manager: lower chances for human errors, really quick service
	Checking each ingredients individually for the expiration date: It takes a long time (check every ingredient)	The ingredients get automatically checked, the system keeps track of all expiration dates and sends alert(s) when an ingredient is about to expire.
Information and data	Unstructured data on paper. (Shifts listed on paper: example: Difficult to read/understand for other people)	Structured data on computer (example: shifts)
	Miscommunication about the shifts	They can see their schedule on the app
Economics, costs, profits	Running out of ingredients for popular products.	Keeping track of popularity by day/week. (The system will remember daily sold items, so when an item is sold a lot, extra stock can be arranged for the future.

Control and security	Losing important documents, because it can have bad consequences. (example: information leak: the competition won't have same information as McDonald's.)	Back up data on Cloud, so stock can be recovered in case of accidentally deleted documents.
	Checking own schedule for the workers (opportunity)	Setting up an account for the workers, so they can only check their individual shifts.
Efficiency, people, processes	Waste of food: past expiration date.	Database of the expiration date. (food costs money)
		Keep an object for every stock which contains expiration date and the amount of the ingredient object.
	Saving paper (opportunity) (paper with shifts every week)	Digitalised database: this will include information about shifts, overview of most sold products, stock etc.
Service	Stock of ingredients runs out, No ingredients, no items to sell --> no profit!	Never run out of stock by keeping track of the ingredients inside the storage.

4. The business events

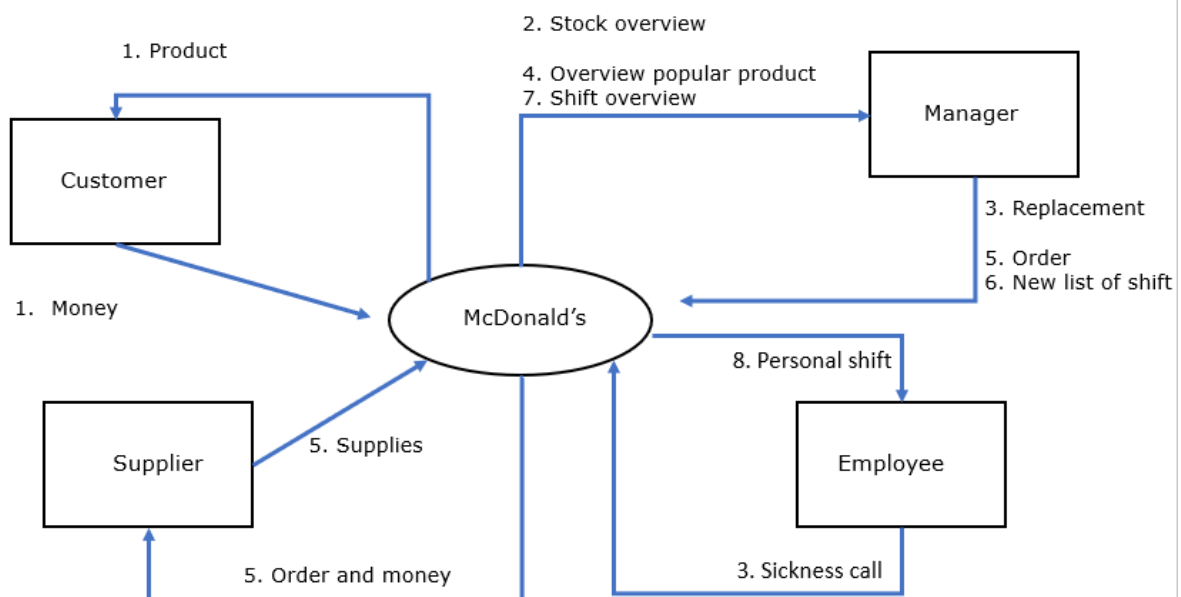
describe the business events

Recap: a business event is something that happens in real world at some moment in time and your business has to respond to or something that is triggered by time

1. Customer orders a product
2. Manager wants stock overview
3. Worker calls in sick
4. Manager wants overview popular product
5. Manager places order(Manager receives a message to order ingredients for popular product, increasing the limit of the default stock limit)
6. Manager makes shift list for the next week
7. Manager wants shift list
8. Worker wants his own shift list

5. Context diagram

describe the communication between your business and the external entities in a context diagram. Make sure the communication caused by the business events is reflected in this diagram.



6. Business use-cases

For each business event, define and describe the business use-case (reaction of the business on the business event)

1. Create an order
2. Generate stock overview
3. Find replacement for sick employee
4. Generate overview popular product(s)
5. Create order with supplier

6. 7. TO BE situation

Define the ideal situation for your business, so that all future models can be derived from it (TO-BE situation)

Also define the difference between must-have and nice-to-have business requirements

Don't mention the system yet, business analysis is about the business!

- Manager has to be able to :
 - Change shifts based on availability of employees
 - Get overview of popular products, stocks and employees
 - Get overview of bill from supplier
 - Pay the supplier
 - Order extra stock (for the popular product)
- Customer has to be able to:
 - Place the order
 - Receive the order
 - Pay the bill
- Supplier has to be able to:
 - Receive the order
 - Deliver supplies
 - Receive payment
- Employee has to be able to:
 - Handle the order
 - Check availability of the order
 - Put order in database
 - Receive payment
 - Check shift(s)

3.Requirements model

1. The scope of this project

In this paragraph, you define the scope for this project

The CEO of Mc Donald's, Steve Easterbrook has decided to make the company more efficient.

MC Donald's is a fast food restaurant where people can buy food such as burgers, French fries, salads etc. Also there is a possibility to buy soft drinks and milkshakes. This can either be purchased separately or in menu or in small packages (Happy meal).

Menu can only be purchased per person: 1 or more burger(s), French fries (to choose from 3 different sizes: small, medium or large) or 1 type of salad and added with a drink (to choose from small, medium or large) which can be refilled once.

Happy meal (for kids until 12 years old) : 1 burger, French fries (to choose from small, medium or large) with 1 kind of drink (to choose from 3 different sizes: small, medium or large) with 1 dessert (ice cream/yummy big cookie) and a toy (to choose between a boy toy or a girl toy).

Customers have the choice to either order traditionally by coming inside, placing an order, paying the bill and taking their order or through driveway: the customer comes in a car and places an order via a speaker (which is inside a wall) while he/she is seated in the car. Then the customer pays the bill (paid with cash or in various credit cards) and receives the order while seated in the his/her lovely car.

The customer comes in, goes to the counter, places an order, the worker adds the order to the cash desk, makes the bill. When the bill is paid, the cook receives the order and prepares it. When the meal is ready: the counter worker delivers it to the customer and smiles back. (not possible in any other way)

The employees just have to give in the order and when the customer pays, the ingredients needed to complete the order get subtracted from the total stock of that ingredient. When the stock gets under 25% the manager gets a message if he wants to order more stock or leave it like this and do it another time. He can also generate a stock overview to see how much waste he had last month, so he can lower the stock of an ingredient.

A big problem is that they now have no way to check the popularity of items, so they don't have a clue what to advertise. That's why he requested that we find a solution for Mc Donald's, so they can keep track of all their information.

We will also have to be able to keep track of the employee's shift so it's easier for them to keep track of when they'll need to come in for work. They first have to log on so they can only see their own shift(s). Also by collecting all Mc Donald's information, they can get rid of all the paper copies and become even more organized and it'll be more environment friendly.

We just make a system for McDonald's itself so no website where the client can check all the products.

- Manager has to be able to :
 - Change shifts based on availability of employees
 - Get overview of popular products, stocks and employees
 - Get overview of bill from supplier
 - Pay the supplier
 - Order extra stock (for the popular product)
- Customer has to be able to:
 - Place the order
 - Receive the order
 - Pay the bill
- Supplier has to be able to:
 - Receive the order
 - Deliver supplies
 - Receive payment
- Employee has to be able to:
 - Handle the order
 - Check availability of the order
 - Put the order in
 - Receive payment
 - Check shift(s)

2. The functionalities

give an overview of all functionalities that will be implemented in this project

1. Create an order
2. Generate stock overview
3. Find replacement for sick employee
4. Generate overview popular product(s)
5. Generate overview of shift(s)
6. Assign shift(s)
7. Create order with supplier

3. The actors

Give an overview of all actors

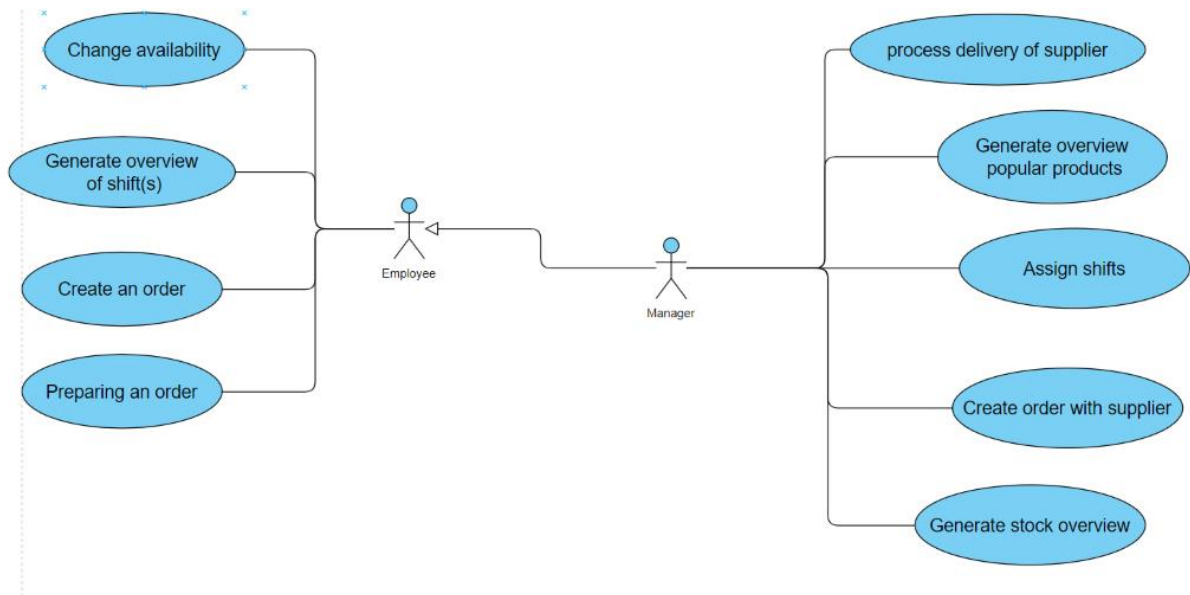
(roles) that are going to interact with your system/app

Person: person is not an actor, because a person is unable to intervene.

- Manager
- Employee

4. The use-case diagram

Make the use-case diagram for your project



5. Use-case descriptions

Make the description for each use-case in your diagram: for each use-case:

Name	Create an order
Summary	A customer wants to make an order for food in our McDonald's
Actors	Employee
Precondition	-
Scenario	<p>The employee puts the order in database. He looks at the ingredients and checks if there are still enough ingredients in stock to finish the order.</p> <p>If a product is out of stock => exception1</p>
Exceptions	<p><u>Exception1: out of stock:</u></p> <p>the employee gives the message that the product isn't available and the customer can order something else. In case the customer doesn't want to place an order anymore: employee aborts order</p>
Postcondition	Customer receives order.

Name	Generate overview of shift(s)
Summary	A employee can see his shift(s)
Actors	Employee
Precondition	-
Scenario	Employee generates his/her overview to see when he has to work. He/she is able to see his shift(s) for a period of time. If the manager hasn't already assigned the employee's shift(s) => exception1
Exceptions	<u>Exception1: unassigned shift(s)</u> : The employee sees a message "You have no shifts assigned yet".
Postcondition	Employee sees his/her shift(s).

Name	Generate overview for popular product(s)
Summary	The manager wants to know about all the sold products
Actors	Manager
Precondition	-
Scenario	The manager generates an overview of all products. By using the popularity data in the system the popular product will appear on top (sorted by descending popularity). In case no product is sold => exception1
Exceptions	<u>Exception1: no product sold</u> : Managers sees the following message "Today no products were sold, don't give up!"
Postcondition	Managers gets overview of all products

Name	Assign shift(s)
Summary	The manager assigns shift(s) for the upcoming month.
Actors	Initiating actor: Manager Receiving actor: employees
Precondition	-
Scenario	<p>He gives in the name of the employee date and time of the shift. First he checks if he is available. If the employee is available he clicks on "assign". Otherwise the system searches for an available employee for that time.</p> <p>If an employee who was assigned to the shift calls in sick. The manager has to search somebody else for that shift.</p> <p>In case of overwriting a shift => Exception1 While closing the app and not every shift is filled in =>Exception2</p>
Exceptions	<p><u>Exception1: overwriting a shift:</u> Manager gets a warning before overwriting. He then gets the option to overwrite it or cancel.</p> <p><u>Exception2: Open shifts:</u> The manager gets a warning that not all shifts are filled in. He then can decide either to save and close or go back to the app and fill the missing shift(s).</p>
Postcondition	Shifts are assigned.

Name	Create an order with the supplier
Summary	Manager wants to order ingredients for stock.
Actors	Initiating actor: manager Receiving actor: supplier
Precondition	-
Scenario	The manager generates a stock overview. He then gets a message for every ingredient that is low on stock. He then calls the supplier and orders the stock he wants to order. In case the supplier is out of stock => exception1
Exceptions	<u>Exception1: out of stock:</u> In case the supplier is out of stock. The manager has the option to Either wait until the product is back in stock or try to find another supplier.
Postcondition	The ingredients are on their way.

Name	Generate stock overview
Summary	Manager wants to inspect the stock.
Actors	Manager
Precondition	-
Scenario	By pressing the button "generate stock overview", the manager generates a stock overview.
Exceptions	-
Postcondition	Manager sees the stock overview

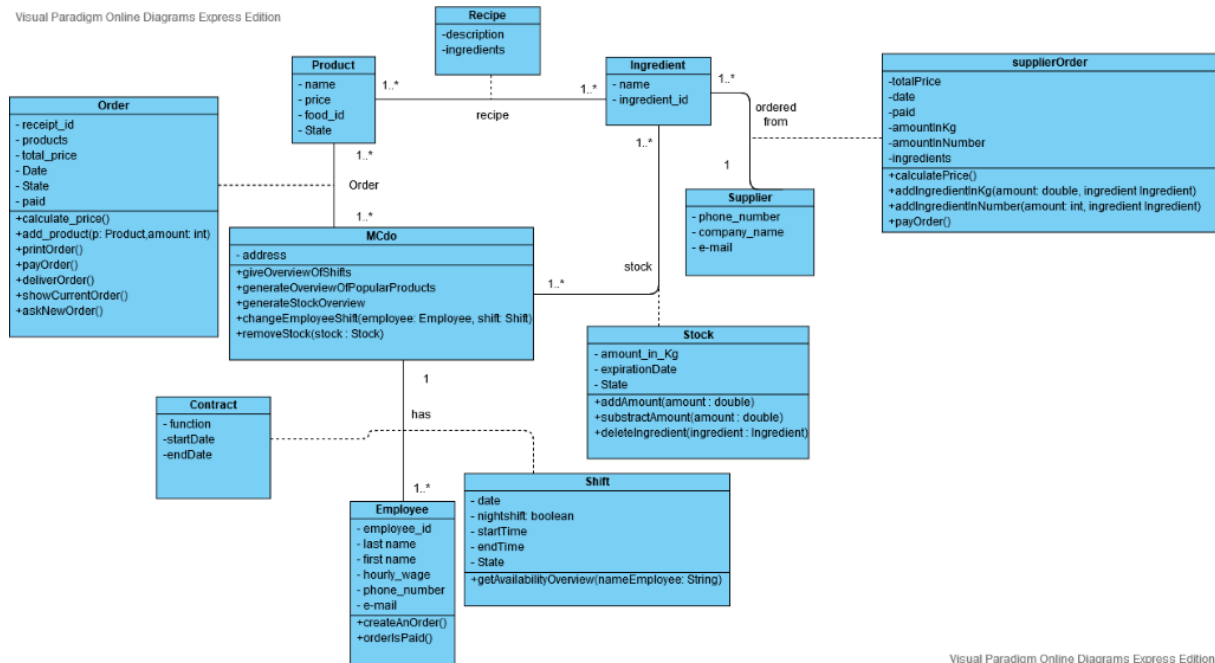
Name	Preparing an order
Summary	The order of the customer has to be prepared
Actors	Employee
Precondition	Order has to already be created
Scenario	First the cook selects the next order that he will prepare. Then he looks at what ingredients he needs and grabs them. Then he prepares the food product by product. Afterwards he/she gives it to the cashier, which then will be given to the customer.
Exceptions	-
Postcondition	Customer gets his order

Name	Changes availability
Summary	The employee want to change the days he is available for work.
Actors	Employee
Precondition	Employee calls the manager to brief him that he'll be unavailable for certain days.
Scenario	The employee then changes the days he is available in the system.
Exceptions	-
Postcondition	The employee's availability has changed

Name	Process delivery of supplier
Summary	The employee wants to process the delivery of the supplier
Actors	Employee
Precondition	The supplier has delivered the products
Scenario	The employee checks what was on the order and checks if every ingredient is there. Then he adds the ingredients that are delivered to the stock . He also changes the values of the stock overview. Exception 1: not all ingredients that are ordered are there
Exceptions	Exception 1: the employee goes to the manager to tell him not every ingredient is delivered. The manager then places a new order for the missing ingredients.
Postcondition	The stock is refilled with the all the ingredients that where delivered.

4. Business model

1. The class diagram



2. The model dictionary

Product	This is where the different products are going to be saved in the system. You'll be able to check at what price a product is being sold for. A product can be a single burger or a menu.
Order	A summary of what the customer ordered. You'll be able to see what items are often bought together as well as see which product gets ordered a lot in general and by this way gets popular. This information is kept for a year.
Ingredient	The different kinds of ingredients that McDonald's has access to. This informations gets updated once a month.
Shift	The date, time and place of the shifts, so the employee knows where and when to work. The information is saved 4 months after the date.
McDonald's	The location of where the McDonald's is located. This will almost never change unless a McDonald's gets added or deleted or the address of a McDonald's is changed.
Stock	The amount that we have of a specific ingredient at a specific McDonald's. This one get updated everyday.

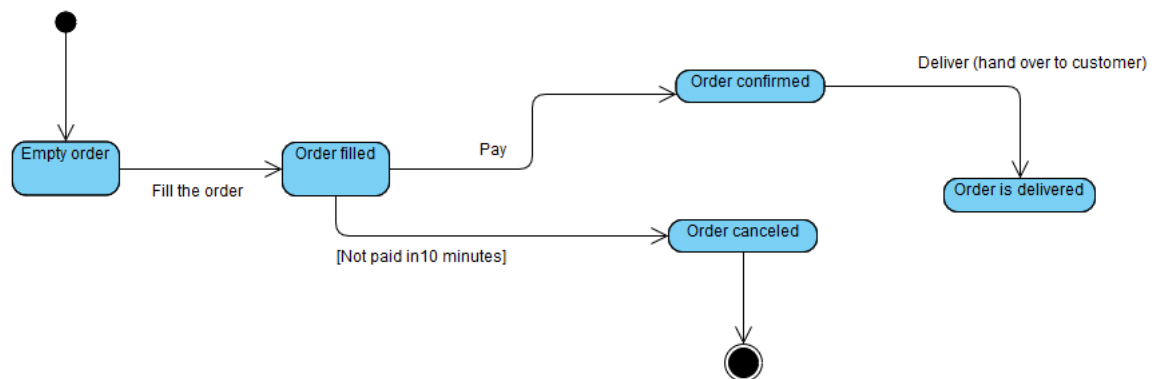
Supplier	The contact information of the supplier of a specific ingredient. There is only one supplier a product. This information will be kept forever.
Contract	The function of an employee this can be, for example 'manager' or 'cook'. The data of an employee will be deleted 4 months the end_date of the contract.
Employee	Information of the people working at the McDonald's. An employee gets deleted out of the system, if he decides he does not want to work there anymore.
Recipe	The amount of ingredients and the kind of ingredients that is needed for a specific product.
SupplierOrder	The information from the order that is placed with the supplier. This information is deleted after a year.

5.Dynamic Layer

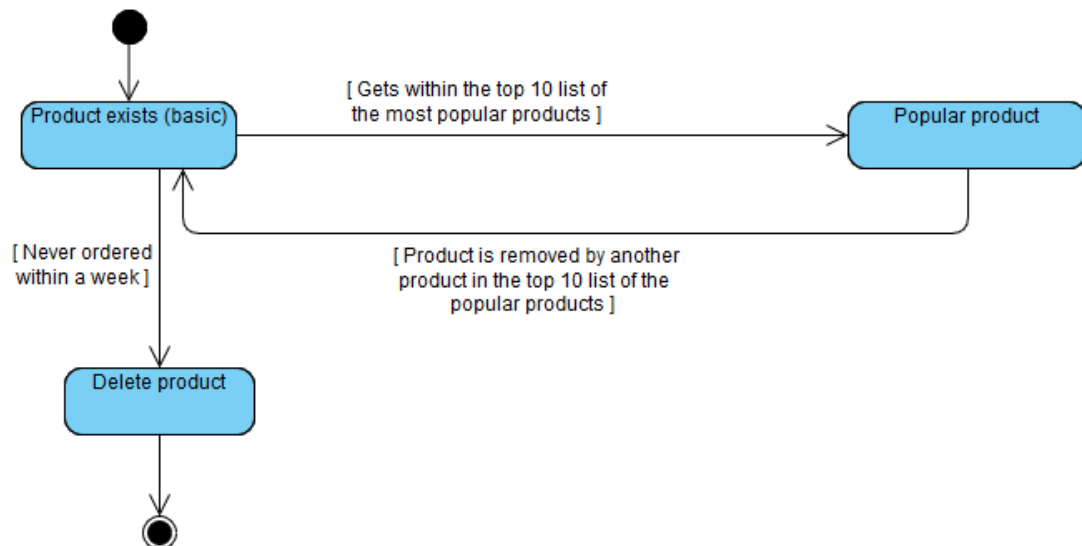
1. State diagram: name class

In this paragraph, you add a state diagram for each dynamic class in the business model

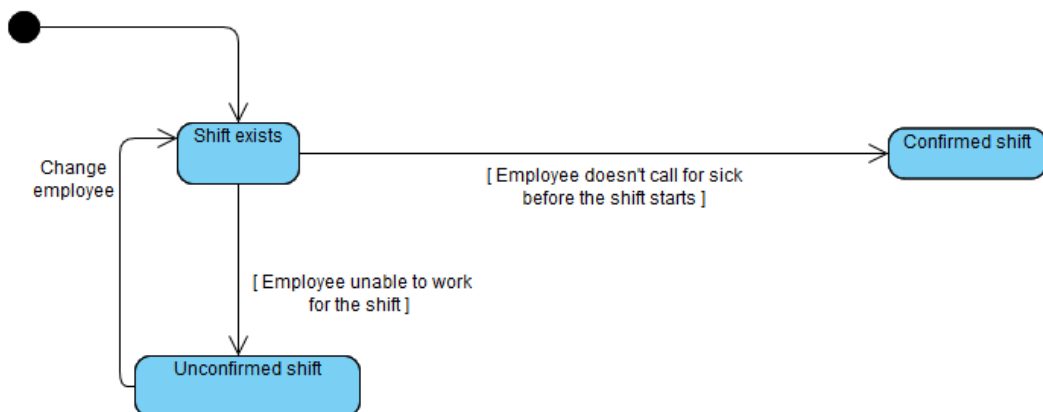
Order:



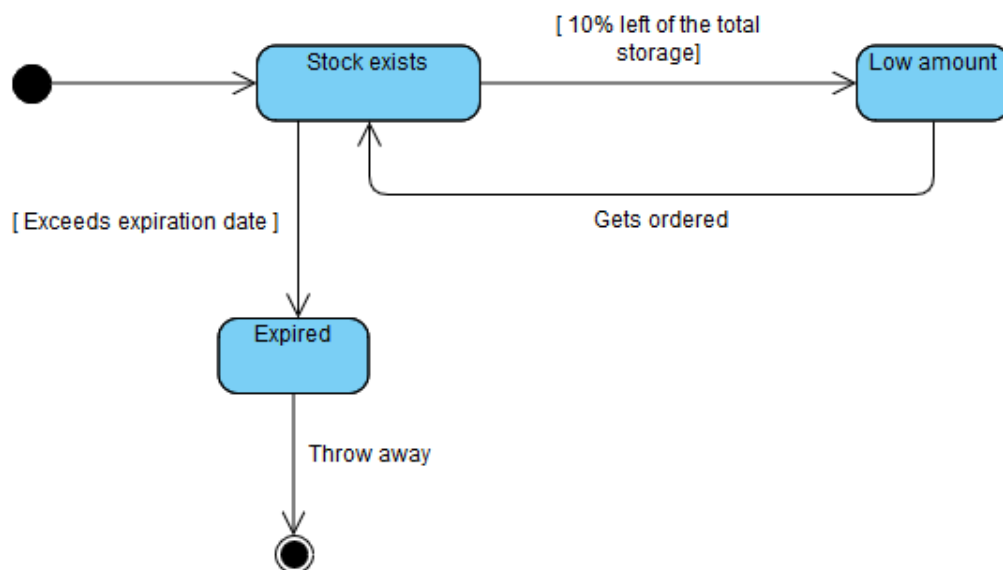
Product:



Shift:



Stock:



Extra information

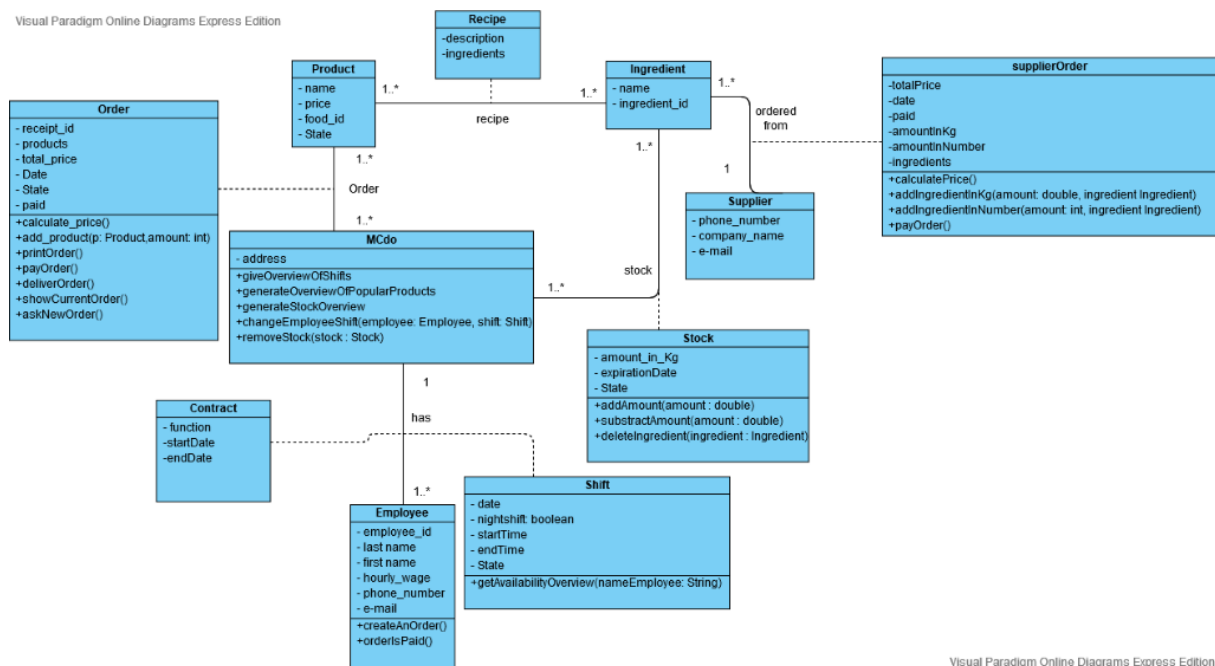
Order	Order is always possible so we don't have a state where we have to check if an order is possible or not.
-------	--

2. The class diagram V2

In this paragraph, you add the class diagram after creating the state diagrams
Don't forget: add state variable in every dynamic class and add operations for all events in state diagram which weren't operations in class yet

Dictionary:

Stock: Example: a carrot with an expiration date on Wednesday will be a different item than a carrot that will expire on Friday, it will have a separate space in the list of stocks. Once it's expired it gets removed from that stock list.



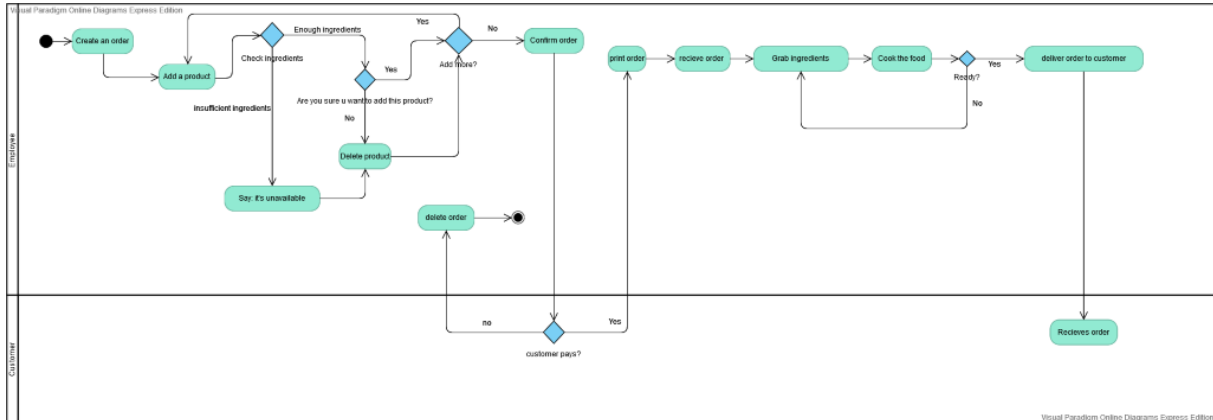
3. Activity diagram: name workflow

In this paragraph, you add an activity diagram for each workflow in your project. Make sure every member of the team creates at least 2 activity diagrams.

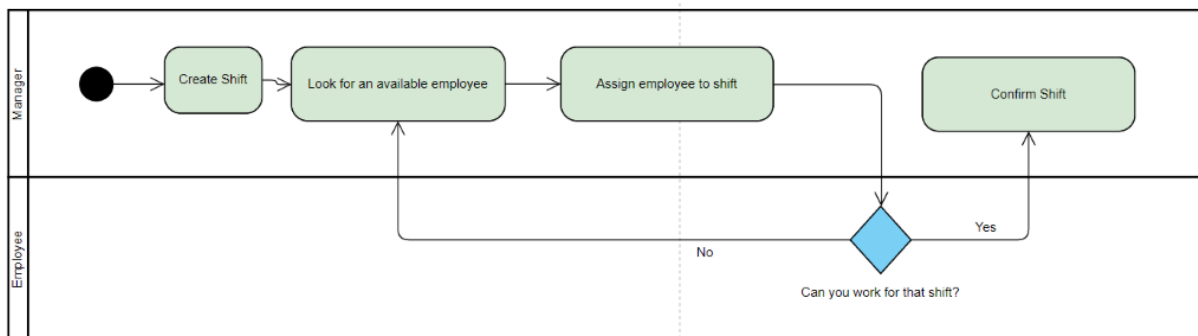
If you prefer bpmn notation, that's OK as well

//order, stock order, shift assign

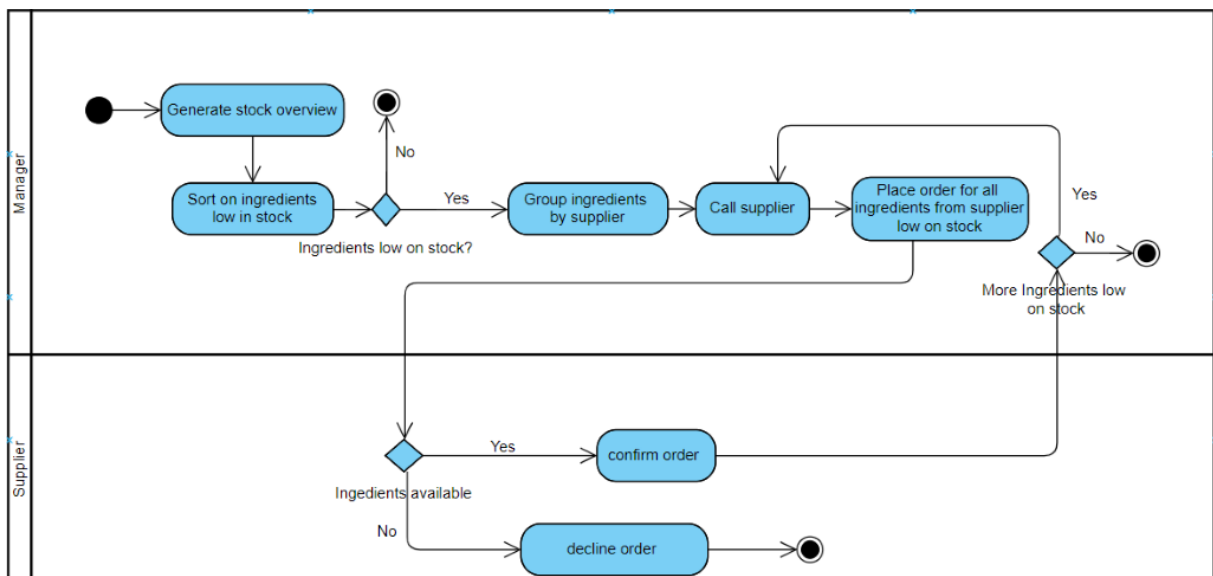
Order:



Shift:

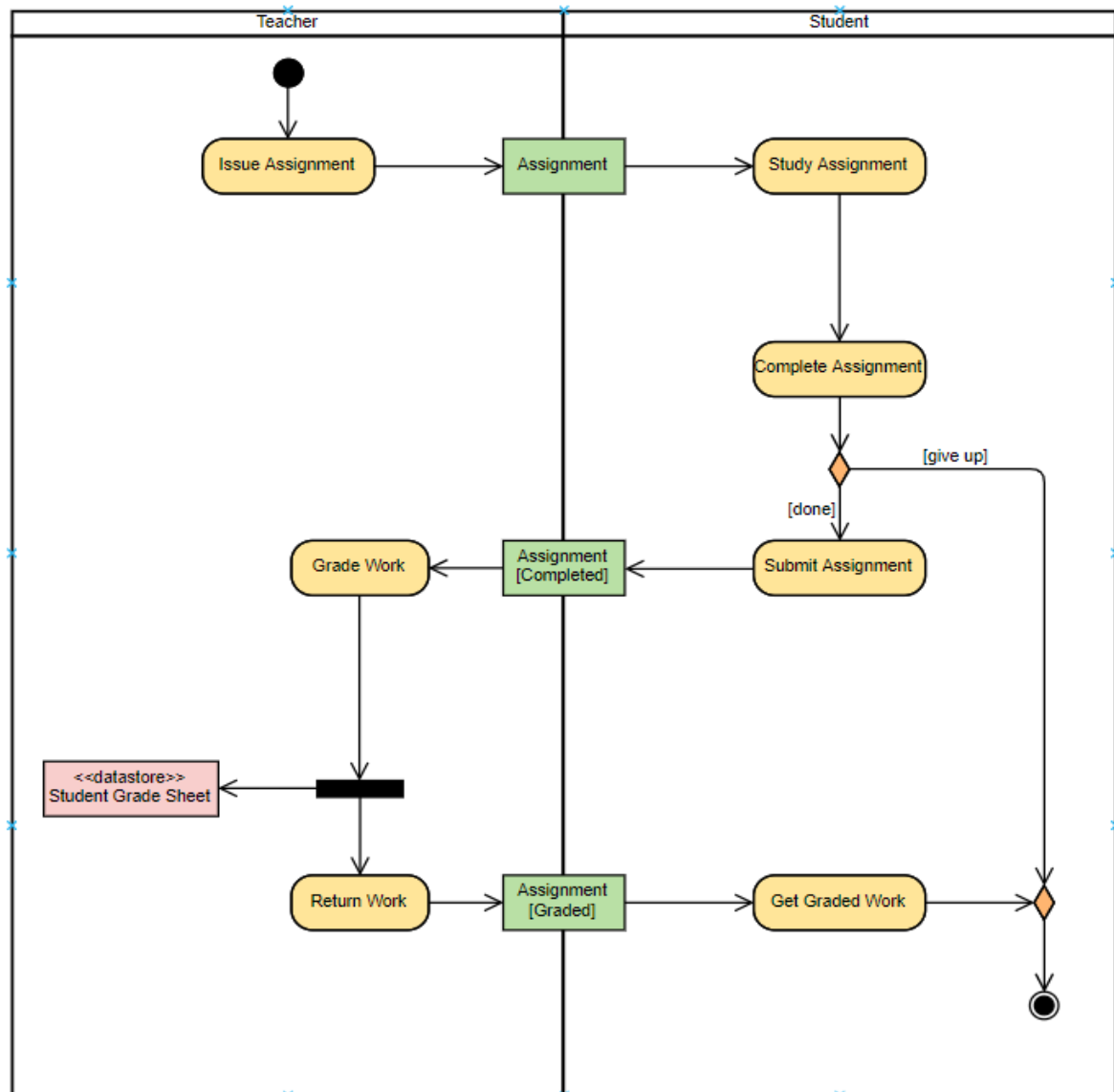


Supplier:

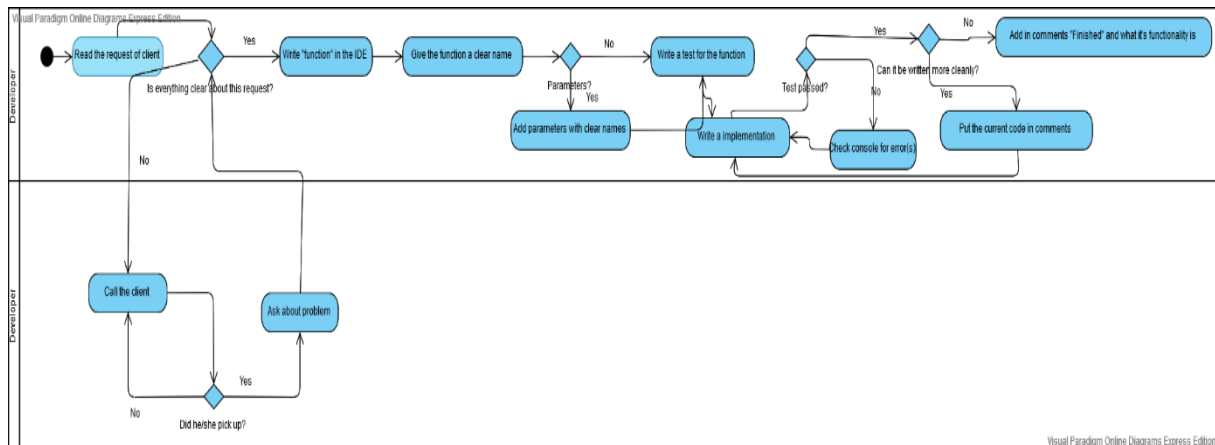


The following diagrams are from self thought idea's:

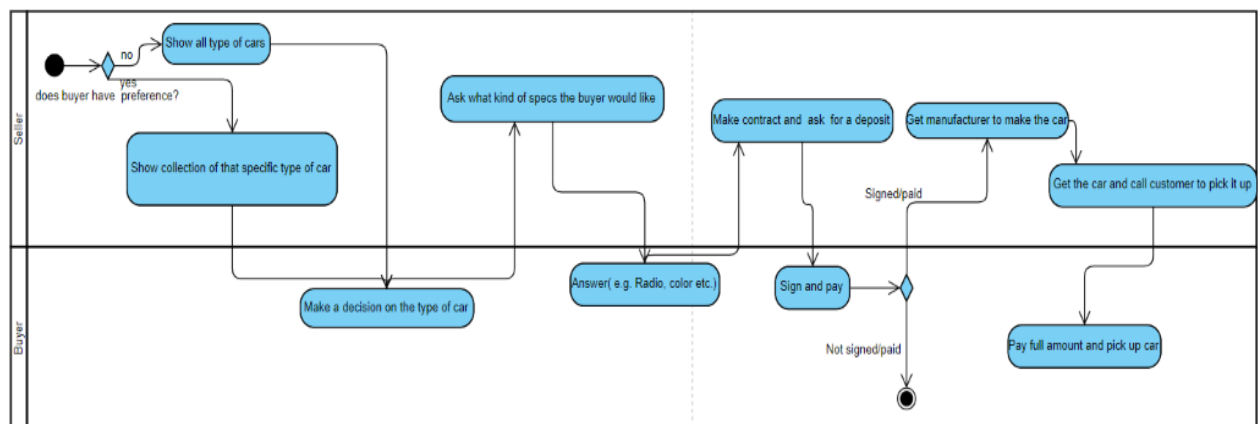
Completing an assignment:



Writing a function:



Buying a car:

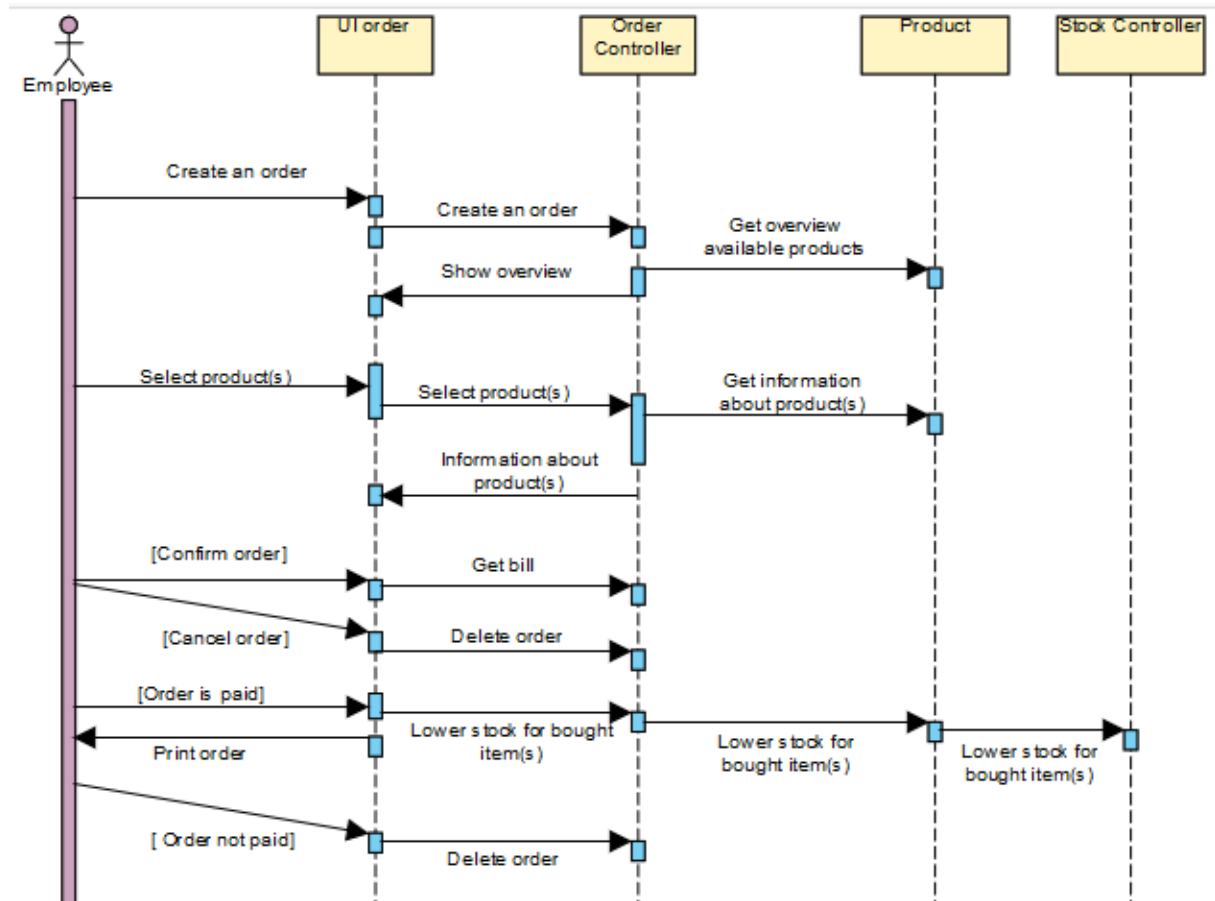


6.Application layer

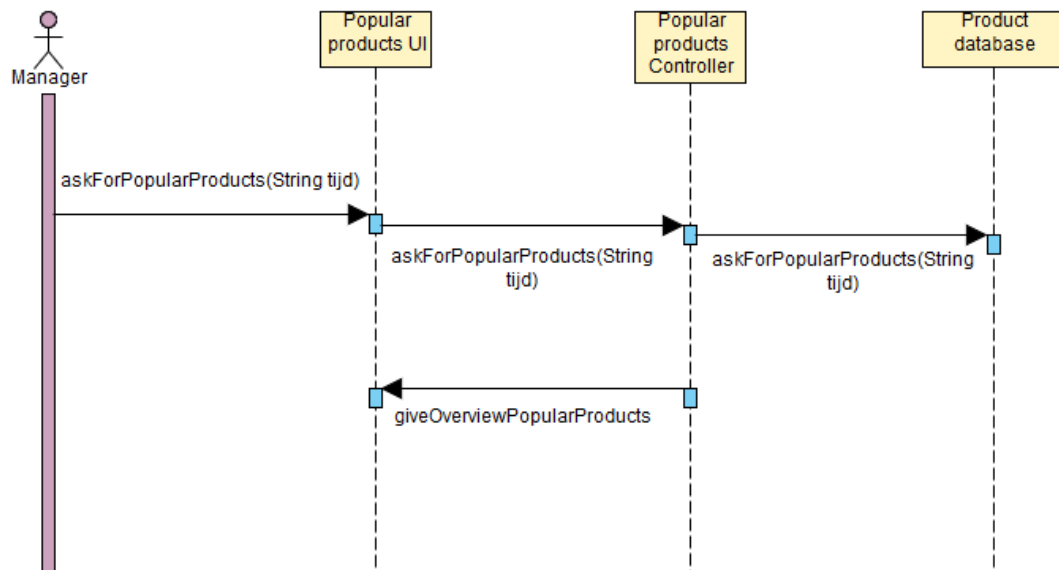
1. Sequence diagram: name use-case

In this paragraph, you add a sequence diagram for each use-case in the requirements model

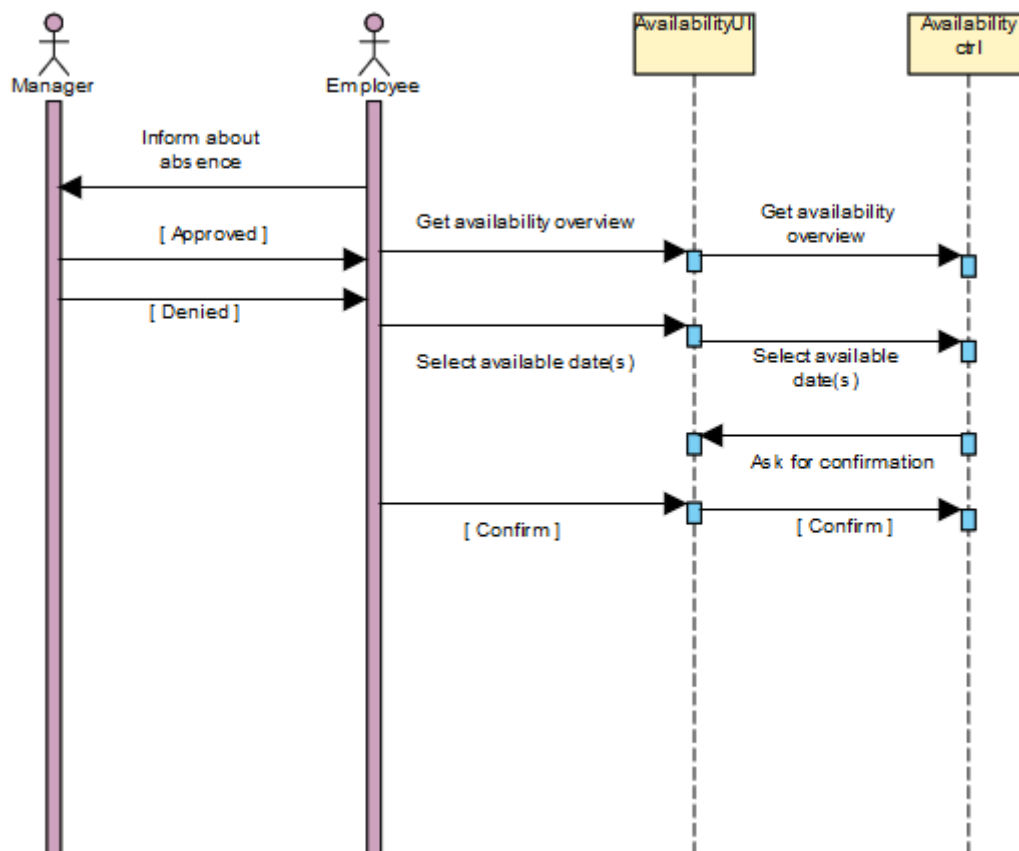
Create an order:



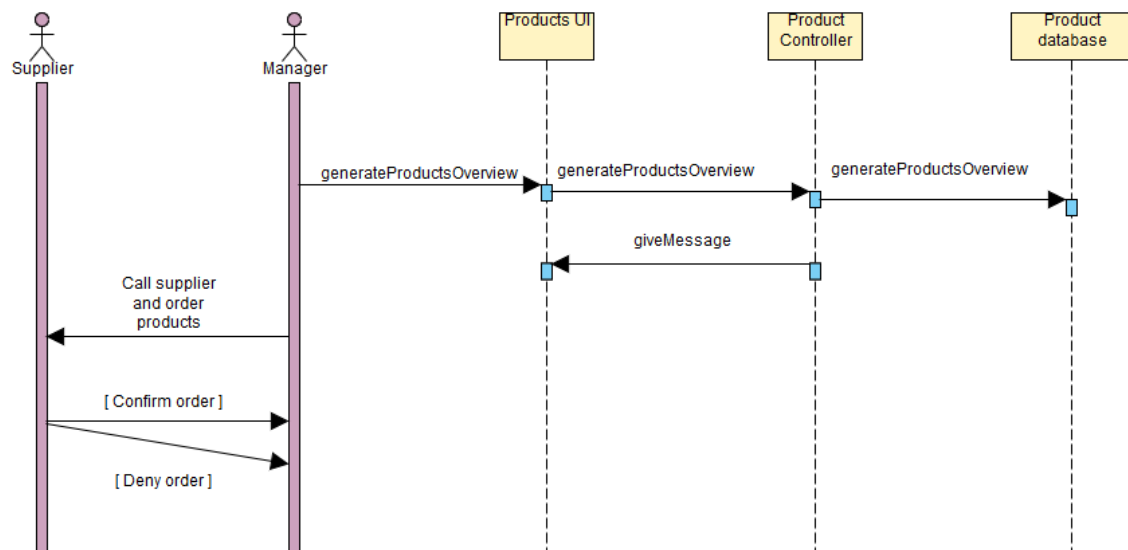
Generate overview for popular product(s):



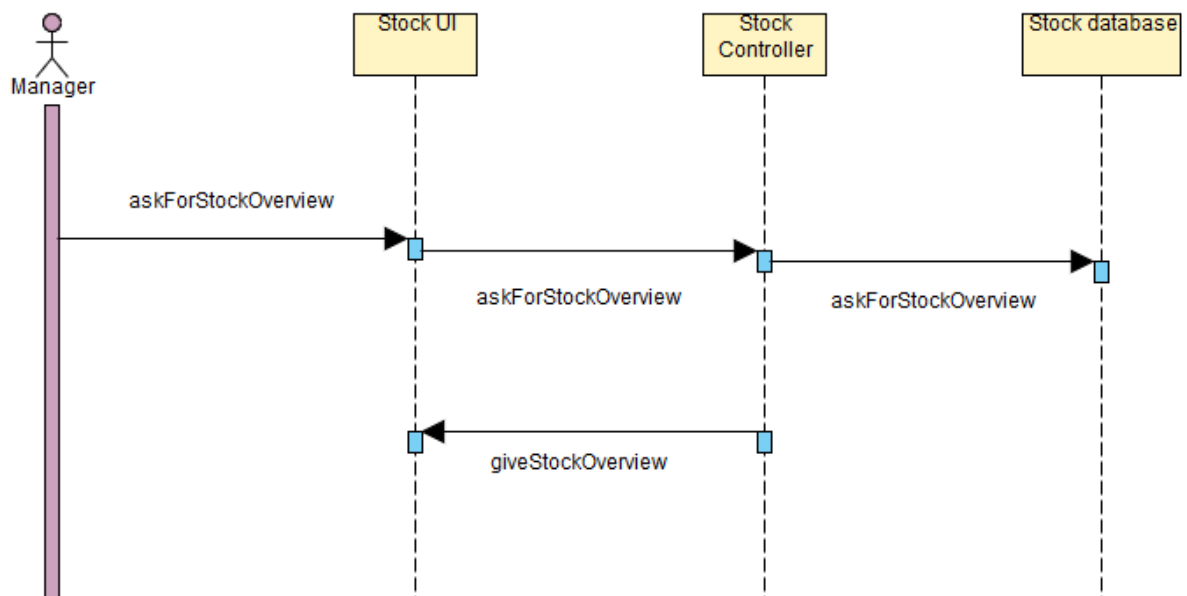
Assign shift(s):



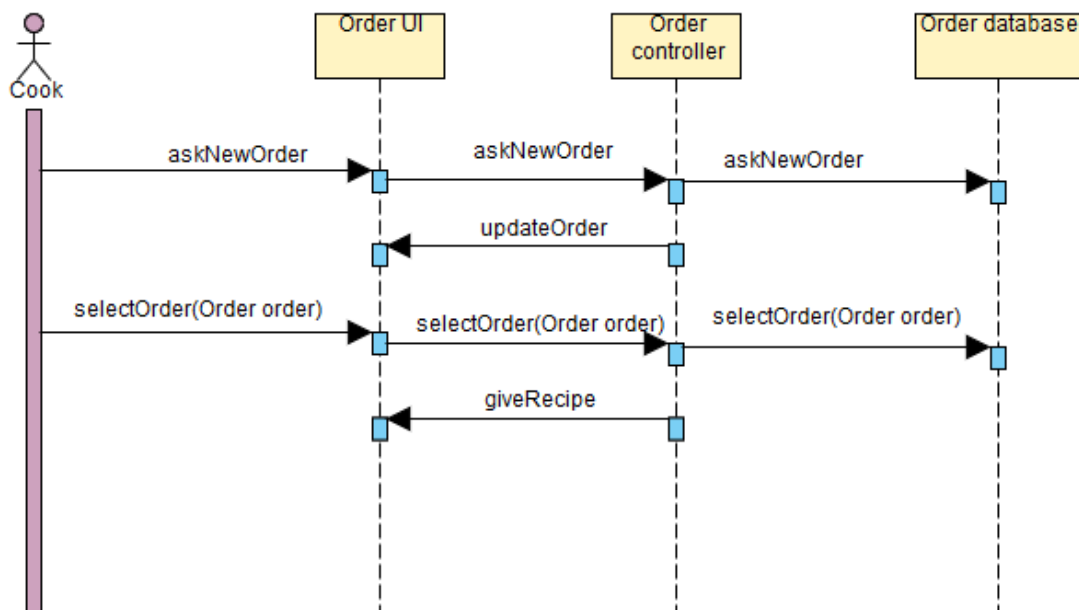
Create an order with the supplier:



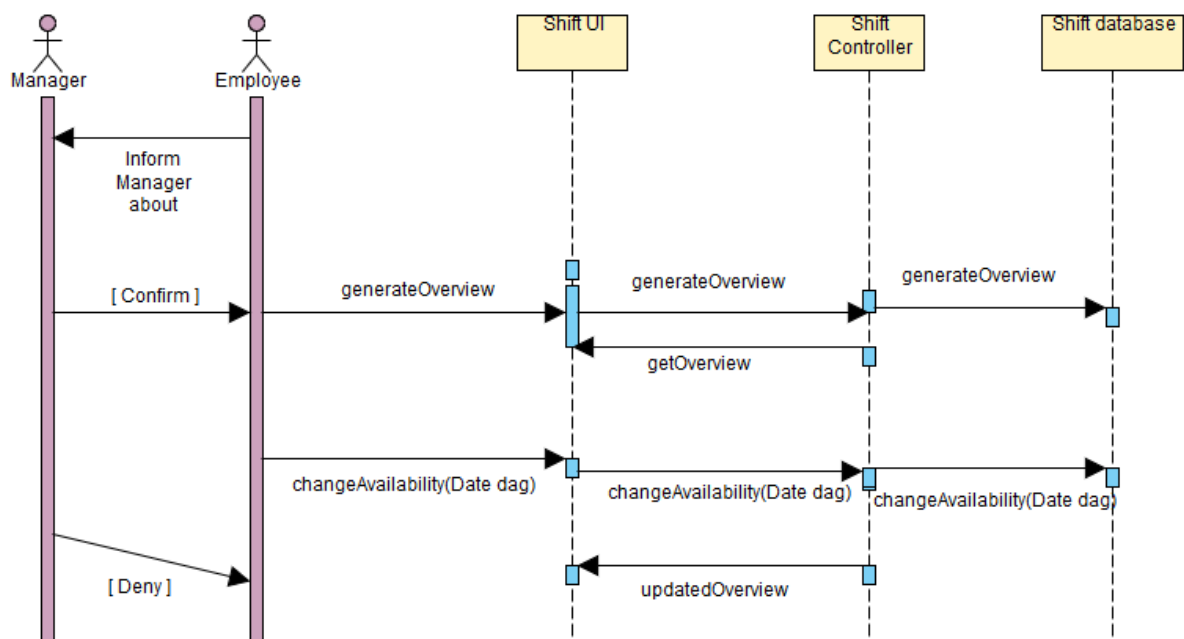
Generate stock overview:



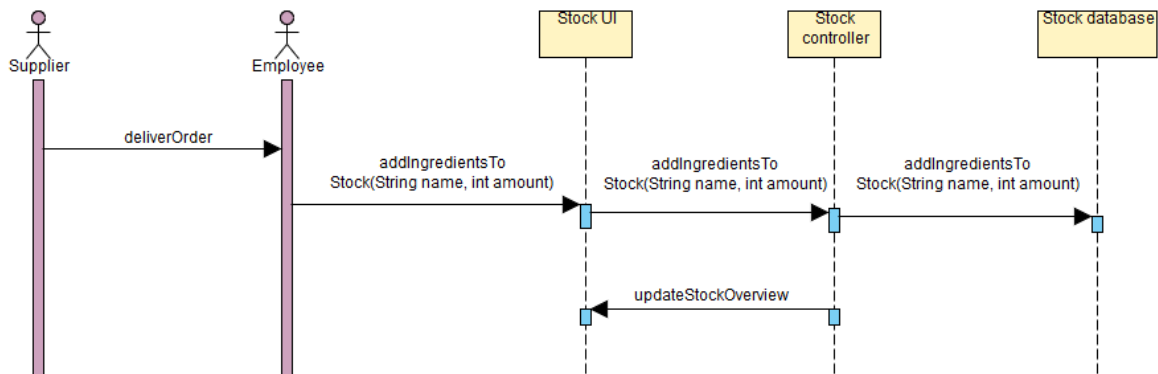
Preparing an order:



Changes availability:



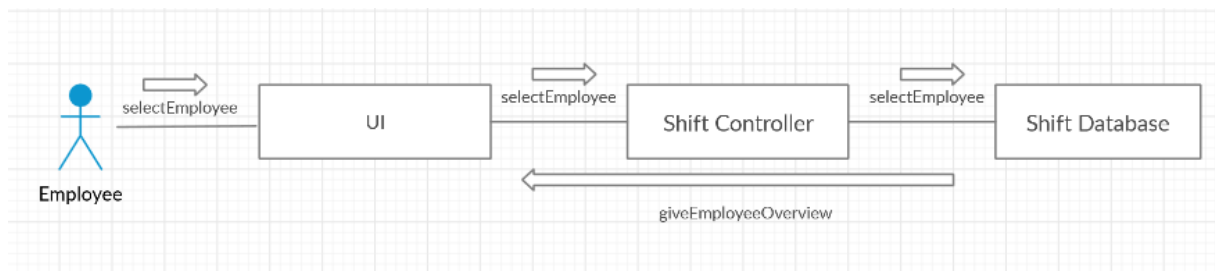
Process delivery of supplier:



2. Collaboration diagram: name use-case

In this paragraph, you add a collaboration diagram for each use-case in the requirements model

Generate overview of shift(s):

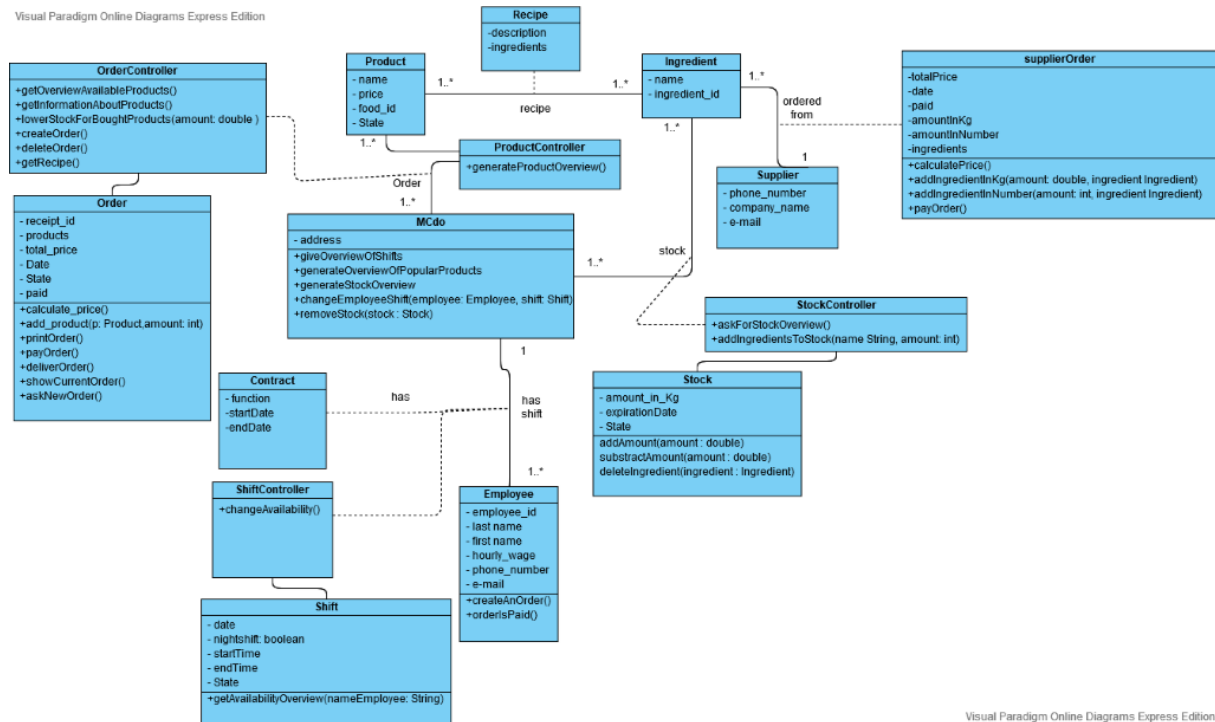


3. The class diagram for application layer

In this paragraph, you add the class

diagram after creating the sequence and collaboration diagrams

Don't forget: add operations as needed and add the controller classes



4. The model dictionary for application layer

In this paragraph, you add the model dictionary as made in the business

model, but elaborated on extra information

Product	<p>This is where the different products are going to be saved in the system. You'll be able to check at what price a product is being sold for. A product can be a single burger or a menu.</p> <p>Extra info about Generate overview for popular product(s) : String tijd: The manager gives the name of the time period: Example: askForPopularProducts(1/12) --> day of the month. For popularity of the whole month --> askForPopularProducts(December)</p>
Order	<p>A summary of what the customer ordered. You'll be able to see what items are often bought together as well as see which product gets ordered a lot in general and by this way gets popular. This information is kept for a year.</p>
Ingredient	<p>The different kinds of ingredients that McDonald's has access to. This informations gets updated once a month.</p>

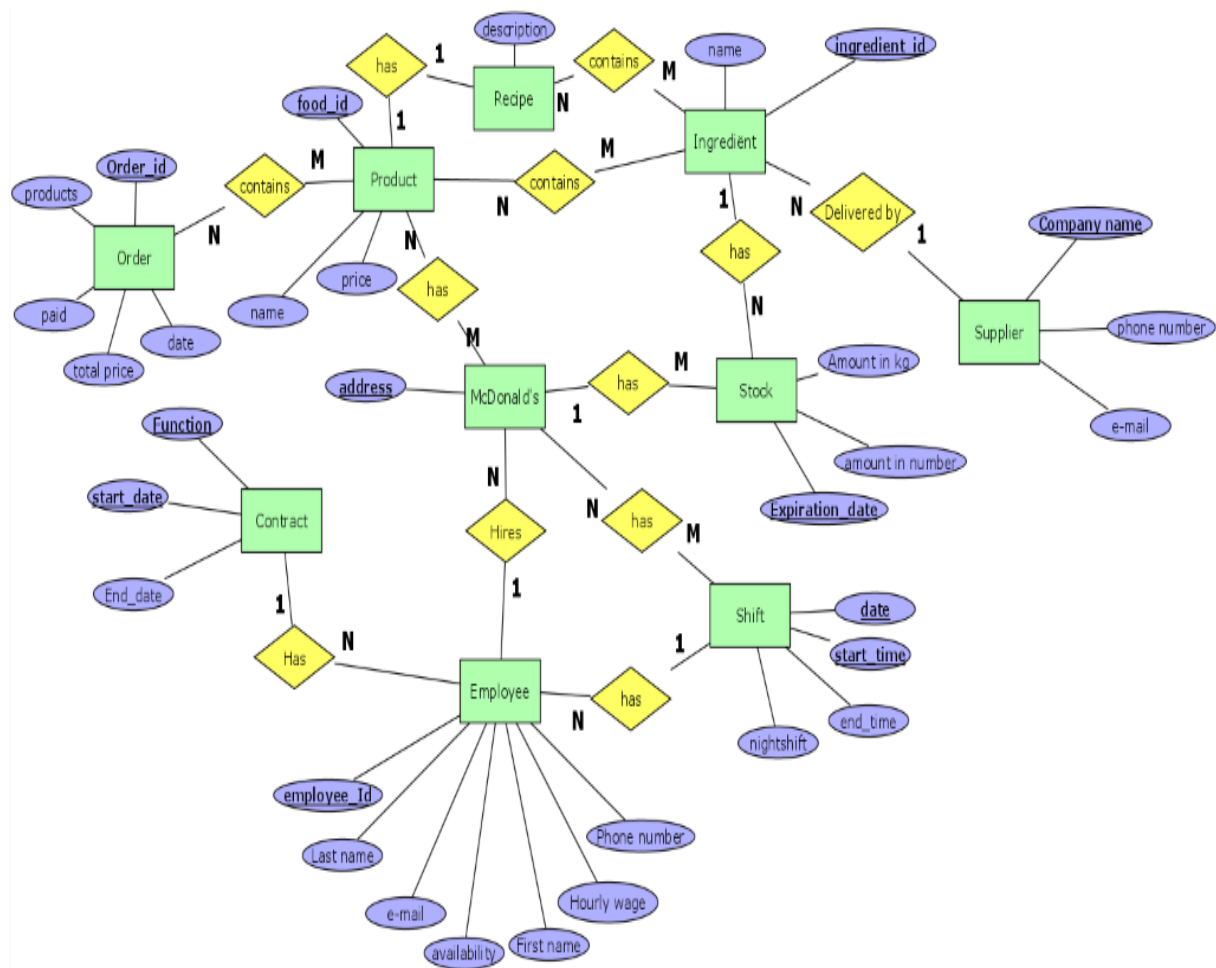
Shift	<p>The date, time and place of the shifts, so the employee knows where and when to work. The information is saved 4 months after the date.</p> <p>Extra info about Changes availability: Default value of availability is true</p>
McDonald's	The location of where the McDonald's is located. This will almost never change unless a McDonald's gets added or deleted or the address of a McDonald's is changed.
Stock	The amount that we have of a specific ingredient at a specific McDonald's. This one gets updated everyday.
Supplier	<p>The contact information of the supplier of a specific ingredient. There is only one supplier a product. This information will be kept forever.</p> <p>Extra info about Create an order with the supplier: Deny order: with this we mean that the supplier tells the manager what products are unavailable for the order he wants to place. So basically those ingredients won't be added to the order.</p>
Contract	The function of an employee this can be, for example 'manager' or 'cook'. The data of an employee will be deleted 4 months after the end_date of the contract.
Employee	Information of the people working at the McDonald's. An employee gets deleted out of the system, if he decides he does not want to work there anymore.
Recipe	The amount of ingredients and the kind of ingredients that is needed for a specific product.
SupplierOrder	The information from the order that is placed with the supplier. This information is deleted after a year.

7. Technical design

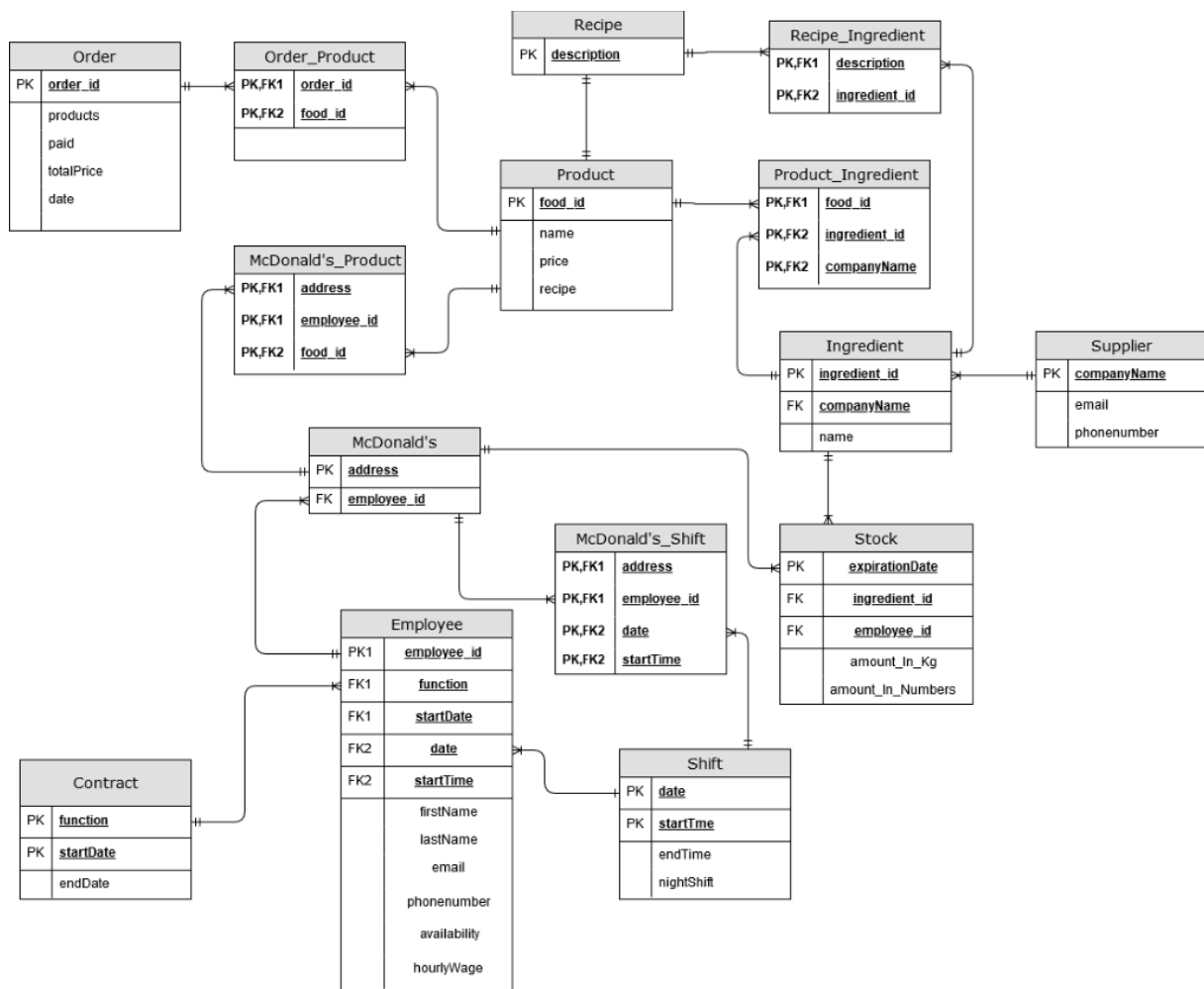
1. Data model

In this paragraph, you add the data models

- Conceptual data model



- Relational data model



2. Design patterns

In this paragraph, you add a description of design patterns you can use in the implementation of your project and describe why and with what purpose you'll use them

None of us three have the OPO – OOO and therefore we have no knowledge of patterns.

That is why are leaving this part blank.

Thank you

Team 5 - Bob's burgers