Fisketorvet Unlimited

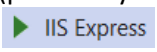
Alexandru-Nicolae Lazaroiu, alex056t@edu.easj.dk
Paula-Cătălina Curcă, paul6465@edu.easj.dk
Kristaps Mežatučs, kris52m2@edu.easj.dk
Pedro Rainho, alex056t@edu.easj.dk

Supervising Teachers:
Zuhair Haroon Khan, zukh@zealand.dk
Mohammed El Allali, moal@zealand.dk
Nilma Abbas, NIAB@zealand.dk
Ivan Rosenvinge Frederiksen, IVRO@zealand.dk

Table of Contents

Table of Figures

I. Introduction

The purpose of this project is to introduce the reader to the development process of our web application called “Fisketorvet Unlimited” bringing him in contact with the world of agile project management and development. Another goal we set was to create a report that would be explicit and succinct, allowing the reader to understand the process of developing an application even if they do not have a broad technical knowhow.
In order for the user to open the solution he must have Visual Studio(preferably VS2019Enterprise) installed, double click “fisketorvet_project_v1.sln” and then click the 

button to run it.

1.Problem definition

We were tasked with the development of an interactive online web application pertaining to the well-known Fisketorvet Shopping Mall in Copenhagen which will be able to handle and manage products separated in different stores, allowing the admin to create, delete or edit these stores and products. The registered user would in his own right be able to add products to his shopping cart and complete the purchase.

2.Contractual Obligation

Before even beginning talking about development of the project, the team made a verbal agreement with everyone present, to meet daily, either online or in person, and to catch up on any lost work, once again together, this time online, during the weekends. All of the members agreed, and we started our first meeting.

3.Inception Deck

The inception deck is a very simple common tool used in most development methodologies, at the inception of the project, that helps the team get together on the same page and ensures that all team members are working towards the same common goal, with a similar vision of the project. This points the team into the same direction from the start, therefore even though it is lightweight, it provides certain benefits to the team members. The inception deck can be created by asking 10 simple questions:

a) Why are we here?
   We are here in order to deliver a modern online solution for the shopping mall Fisketorvet, which will help them attract more customers. Our main purpose as a team is to strengthen our teamwork and apply all of the knowledge we have gathered in this first semester of our curriculum towards becoming full-fledged software engineers. Another goal of our team is to solidify all that we have learnt so far and build upon it, acquiring new skills and insight in the upcoming weeks. We are looking forward to developing as a team and we are sure it will be an educational journey.
b) Can you create an elevator pitch?
   For [new and current Customers of Fisketorvet]
   Who [want a better online experience, and easier access to offers and events].
   The [FISKETORVET UNLIMITED.PURCHASE FASTER, NOW ONLINE!]
   A new [online interaction tool]
   That [enhances the customer experience].
   Unlike [the current website]
   Our product [allows online orders, where a discount is offered when shopping with the new special card].

c) Can you make it appealing by creating a design box?
Top 3 reasons why you should buy our product:
- The possibility to make online purchases is now no longer a dream!
- The system is easy to use, intuitive and has all levels of user experience in mind.
- The system is visually appealing.

Slogan: Fisketorvet Unlimited, purchase faster, now online!

d) The NOT LIST(at this point in the inception deck we select the most important achievable goals for this project and the unattainable ones, in order to have a smoother development process by removing the unattainable parts from the product backlog)
IN SCOPE:
Documentation for possible future updates for the system
Parking reservation system
Show special events and offers for registered customers
Customer/admin log in
Adding the different shops and links to outside website
Ordering and adding to cart functionality
Product catalog
OUT OF SCOPE:
Implementation of a working, secure payment system
UNRESOLVED:
Employee log in

e) Meet your neighbours
The team members of the project are the 4 brilliant minds of the computer science students at Zealand Academy: Alexandru-Nicolae Lazaroiu, Kristaps Mezatucs, Paula-Catalina Curca, Pedro Miguel Rodrigues Marques Rainho. Together with our wonderful supervising teachers: Zuhair, Mohammed, Nilma and Ivan, who will help us when we need more insight with our problem solving.

f) Show the solution
We will use JIRA as project management tools together with a personalized excel sheet in order to simplify our development process. The development methodology is, as per requirements SCRUM.
We will create what is commonly known as a website, using C# and ASP.NET core RazorPages as the preferred programming language.

g) Stuff that keeps us up at night
Not being able to implement all the essential features is a collective threat to the development of this project.
Having no contact with real clients, in order to better specify their vision for the project.
Covid-19 pandemic related issues.

h) Size it up
The allocated time for the development of the application is 3 weeks, split in 3 sprints of 5 days each.
There is one extra week in between the upload date and the presentation, which could be used to better understand the underlying issues of the final product and try to implement a fix.

i) What's going to give?
The time for the development is fixed, therefore not flexible, 3 weeks, as previous stated.
The scope of the project is clear, therefore there is little left up to interpretation.
We believe we should deliver the best qualitative final project we can in the given timeframe, in order to receive good marks and get a good, fun learning experience.
Ease of use, we believe the final product should be intuitive and easy to use.

j) What's it going to take?
In order to deliver a sound final project, we have to properly communicate with each other and show our dedication to the team and the scope. Motivation towards reaching our common goal is also an important factor in the process of delivering a solution that satisfies both the scope of the project and the personal expectations of each team member. The project is a good incentive in order to learn new ways of interacting and creating software solutions for the current IT market.

4.Vision

With the help of the Inception Deck, we had a clear idea of what the vision of our project should be like. We first decided that the implementation of a complete, secure log in system should be in place, in order to be able to properly parse and store data. The features that we wanted to implement were primarily focused on functionality, as follows:

- A user can log in
- A user can view the products in the store
- A user can add products (even from different stores) to his cart
- A user can check out
- A user can see his order history
- An admin can add, update or delete stores
- An admin can add, update or delete products in these stores
- An admin can see a list of all previous orders

In our vision the project would also be responsive and pleasing to the eye from a design standpoint.

5. Learning goals

The primary goal of this project is to gain experience in product development and backlog creation in a professional environment. We are mainly focusing on using the experience we gained from the previous courses in order to create a realistic work environment and create a good working practice in order to bring forward a solution that would follow the expectations of our stakeholders and team members. We are also looking forward to understanding the proper way to create and manage sprints throughout the project period and how to properly cooperate as a team in order to properly structure our working process as well as honing our presentation skills and the ability to communicate problems and solutions.
Another important goal would be to better understand the basis of the used technologies and C# programming as well as the relationships between C# and Microsoft RazorPages as well as a more in depth look into CSS and HTML.
The manipulation of data came at a high priority on our learning goals before we even started working on the code we took into account how data can be stored, read and managed in the aforementioned technologies. We used .json files, which is short for JavaScript object notation, a way of storing and manipulating data for web applications.
We are also excited about the use of SCRUM as a development methodology, since this is our first time fully encasing this agile approach for project managing and we are at the same time interested in the symbiotic relationship between documentation and the delivered product at the end of each sprint.

II. Business analysis

A business analysis is necessary to better understand the requirements of the products, by better understanding some underlying values of the company, about its trade practices and connections to the outside world.

1. Business model canvas

The business model canvas is a strategic management and entrepreneurial tool that allows us to describe the business model and the practice that has held the business going. Who do they interact with? How? Why? What are the costs? What is the income of the company? All of these questions will be answered here.

<table>
<tr><td rowspan="2">Key Partners
-Fisketorvet

-The stores in the mall</td><td>Key Activities
-renting out the space for stores
-promoting the stores
-managing the mall (security, cleaning, etc.)
-creating events and offers</td><td rowspan="2">Value Propositions
- High variety of branded stores, groceries.
-Restaurants and cinemas.
-Suitable for many customer segments.
-online purchase system means online sales</td><td>Customer Relationship
- benefits for their registered customers
- lend-lease contracts to different businesses</td><td rowspan="2">Customer Segments
different contractual
artners that enter an
greement to rent a
opping area or
aintain or offer
ecurity
Copenhagen
habitants
tourists visiting the city</td></tr>
<tr><td>Key resources
-employees
-the building/space
-parking lot
-collaboration with other shops/companies in mall</td><td>Channels
-online marketing through own site
-physical ads through the city</td></tr>
<tr><td colspan="2">Cost structure
-Maintenance and general repairs costs
-Property and land tax
-Wages for mall personnel
-Facility management contracts
-Utilities</td><td colspan="3">Revenue Streams
-Leasing out space for the shops
-Marketing and maintenance charges
-Special event charges/rent</td></tr>
</table>

Figure 1. Business Model Canvas

Key Partners: The partners interested in this project, the main stakeholders, are Fisketorvet and the stores which are renting a place in the mall. These shops have a strategic alliance with the Fisketorvet company.

Key Activities: The key activities of a company are the main actions that are performed and make it successful. Fisketorvet owns a big building which is rented out to other shops/companies. In order to make sure that their business is going well they have to promote the shops renting the space. Also, the main company is responsible for managing the whole place, such as cleaning the halls or providing security. In order to promote themselves even more, the mall is organizing events and creating special offers for the customers.

Key Resources: The key resources Fisketorvet has are first of all the physical ones: the building/space they are renting to the other shops/companies with whom they have a collaboration. The parking lot is also an asset valuable to Fisketorvet. On the other hand, there are human resources: the employees, people who manage the place, agreements and contracts.

Value Propositions: When looking at the Value Propositions block, the questions that are asked for filling in this block usually are:

- What value does Fisketorvet deliver to the customer?
- Which one of our customer's problems are we helping to solve?
- Which job are we helping the customer get done?
- Which customer needs are we satisfying?
- What bundles of products and services are we offering to each Customer Segment?

Starting from the first question, we can say that Fisketorvet is a shopping centre that brings value to customers mainly since it's a big centre with several different kinds of shops that could fit the everyday needs of a regular person.
It does not necessarily only fulfil the everyday needs, there is also the possibility of fulfilling the needs that are not essential, like a night out for example something that is more commonly done during the weekend, Fisketorvets can fulfil this because it also has several restaurants ranging from fast-food to a more proper meal.
An online purchasing system means that there can be online sales, this would allow the customer to interact with the shopping mall virtually, rather than in person.
Fisketorvet can also provide some entertainment like the cinemas/IMAX rooms.
Due to the large variety of stores, Fisketorvet isn´t limited to a specific Customer Segment, in this case we are targeting customers who are children, young adults, adults and the elderly from any gender. This due to the large variety of stores, like for example toy stores for children, there are sports stores, stores that are dedicated to your health, clothing & fashion and many more which makes the shopping centre a very attractive place for almost any regular person.

Customer Relationship: Currently, Fisketorvet offers benefits for their registered customers, in the form of a benefit club, which includes extras such as: extended parking times, exclusive shopping offers, a birthday gift and invitation to exclusive VIP events. They also offer a physical information booth in the mall, as well as contact e-mail and telephone number for inquiries and reclamations which can be found on https://www.fisketorvet.dk/contact.
As for other important clients, they are the different contractual partners that enter an agreement to rent a shopping area in the shopping mall. There are also different companies that take care of the maintenance and cleanliness of the shopping mall.

Customer Segments: Fisketorvet targets mostly Copenhagen inhabitants of all social standings, as well as all tourists visiting the city looking to make a purchase in any of the present stores or. Since the mall offers a variety of services, appropriate for all ages, there is no definite age group that the mall appeases to, however it is most popular in the 18-30 age segment.
For whom is Fisketorvet creating value?
Since the array of services is so wide, this creates value for a large group of customers, almost everyone walking around the city could find themselves in need of any of the goods or services available at Fisketorvet. Firstly, people in need of clothes could visit any of the clothing stores in order to make their purchase. Secondly, customers looking to relax in a small group or with their families could join any of the entertainment centres in order to relieve some stress. Lastly, we should not forget about larger groups of friends coming together in larger groups, either to make purchases or use any of the entertainment services. People in need of medicine could also find the produce they need in the shopping mall's drug store.

Distribution Channels: Current distribution channels are split into two categories, first being the current site: https://www.fisketorvet.dk/, where we find information about the stores and where to find them, together with a small description. Lastly, we have the physical ads that can be found throughout the city.

Cost Structure: The cost structure defines all the expenses that your company will incur while operating the business model. Creating and delivering value, maintaining customer relationships and generating revenue all incur costs. Such costs can be calculated relatively easily after defining Key Resources, Key Activities, and Key Partnerships. The main costs for Fisketorvet would be paying for maintenance and general repairs, wages for the employees, land and property tax, facility management contracts which would include but not be limited to cleaning and waste management, landscaping. We should also not forget about the utility costs which would include electricity, water, sewage and so forth.

Revenue Streams: The revenue streams represent the ways your company generates money from each customer segment. Fisketorvet main revenue income comes from leasing out space for shops, as well as charging for maintenance and marketing. They would probably get some extra non-steady income from charging for some special events.

2. SWOT

After we identify how the company functions and interacts with other actors in the business sector with the help of the business model canvas, we continue to do a SWOT analysis of the company, meaning that we identify the Strengths, Weaknesses, Opportunities and Threats of Fisketorvet.

Strengths:

Since Fisketorvet is a shopping mall opened on the 10th of October 2000, it already has 20 years in which it has established itself as a reputable business and is well known throughout Copenhagen and even Denmark.

Strength often lies in variety, so the variety of services that it offers is quite large, 74 bustling stores filled with a range of products, from groceries and medicine all the way to clothes and jewelry. There is also a variety of services offered to the customer besides purchases, ranging from the 14 screens at the famous CinemaxX cinema, to the 22 restaurants where the customers can grab a coffee, a burger or even kebab or other fast foods.

The centre is located next to Dybbølsbro station next to the harbour, which is a central location that attracts many customers daily and is easily reachable.

Fisketorvet regularly hosts events for their customers such as: Fastelavn (a Danish festival), Copenhagen Cooking and Street Food Festival.

Weaknesses:

Current web page looks a bit bland, we could negate this by creating a new and more appealing website. Although located in the centre of Copenhagen, it does not directly attract inhabitants in the outskirts of the city since they have other options closer to home.

Opportunities:

When looking at our main competitor websites it is easily noticeable that only one of the malls has an order online feature and even that just re-directs to the stores' online shop. We could implement an order online feature where people look through our products on the website, order what they want and then their order is waiting for them to pick it up at a designated area. They would receive a digitalized receipt and use it to retrieve their order. This would be a free to use alternative to the personal shopper, as you wouldn't need to walk through the store on your own for a few hours until you find what you want but you could just browse through the online store, however you wouldn't receive the advice of the paid personal shopper feature as that would require paying a price for the advice and guidance of an expert in fashion.

Threats:

The COVID-19 pandemic threatens every business without an existing online purchasing system as people are in lock-down and are reluctant or unable to leave their homes. Thus, our main threats would be the online shopping websites and applications. If we are no looking at our current situation then our main threats would be all other shopping malls in the city. We aren't that worried about the malls in the near vicinity as we have the best possible location – in the centre of Copenhagen, very close to the Copenhagen central train station, next to a river as well as a few hotels which would mean that tourists would probably come to shop here. Rather we should be worried about the malls in the outskirts of the city or at least not in the centre as they would be the optimal choice for people not living directly in the centre and thus would impact the number of customers our mall would intake.

Strengths	Weaknesses	Opportunities	Threats
-Established presence of 20 years - Large variety of stores and services -Great location in the middle of Copenhagen -Regularly hosts events for their customers	-Current website looks a bit bland and doesn't have an online shopping option. -The location may not be appealing for people living in the outskirts	-Implementing and maintaining an online shopping feature -Making the website more visually appealing	-A lengthy Covid-19 pandemic -Stricter restrictions regarding Covid-19 -Competitors have years of experience in the online shopping scene -Competitors in the outskirts may gain greater popularity and syphon our customers.

Figure 2. SWOT analysis

3. PESTEL

The PESTEL (political, economic, social, technology, legal and environmental) analysis is used to identify all the external influences and factors that have direct or indirect impact on the business. The goal of the analysis is to maximize business performance by identifying how exactly these said factors interact with the business.

Political:

Denmark is known as a country with high social guarantees and thus the government requires high tax rates to keep the social infrastructure functional. This could be a problem for a newly opened shopping mall, however seeing that Fisketorvet has been open for 20 years it should not affect us that much as the owners must have gotten used to these tax rates.

Denmark is also a member state of the EU since 1973, therefore it has influence in regulations that apply to the European markets, and profits of some international trade accords. As things look right now, Denmark doesn't plan to leave the European Union.

Economic:

Denmark's economy is one of the largest in the European Union continuously ranking among the top 3 in the European Union if we sort by GDP per capita, therefore imposing itself as a strong presence on the European market. This means that Denmark's citizens have a surplus of money for spending which would also mean that an online shop would be a wise investment. This would also make Denmark an appealing destination for immigration and thus would make finding a consistent work force easier.

The strong service sector in Denmark is the most present one in the market, every year contributing more to the GDP than the extraction of raw materials or manufacturing services.
There is also a very low number of citizens that live in poverty, 5.8% registered in 2016.

Social:

Denmark's population has been growing steadily for the last few decades according to worldometers.info and Wikipedia, so we could assume that there will be no shortage of a workforce in the near future, however knowing that Denmark has a high university enrolment rate we might also assume that people would choose different career options rather than working as a store assistant.

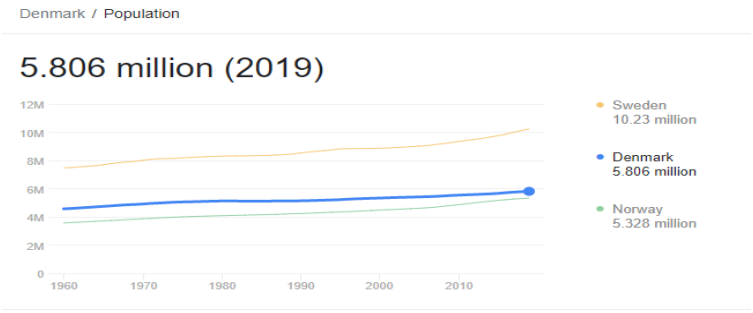


Figure 3. Nordic countries pop growth(2019)

Technological:

Denmark was the most digitalized country in the EU according to the Digital Economy and Society Index (DESI) in 2017. 94% of Danish citizens are online and actively engaged in the use of online services making their digital skills highly advanced. Keeping these statistics in mind we could assume that by creating a newer, more aesthetically pleasing website with more options such as order online, we could attract more customers to our store.
As Denmark has a very large, solid and well developed internet infrastructure, network speeds here are very fast throughout the country.
Denmark has a very present research-and-development sector, considered one of Europe's finest. This helps Denmark always be up to speed with all the rapid shift in technology in today's digitized world.

Environmental:

Seeing as the whole world is being affected by global warming it is safe to assume that Denmark will be also affected because of the low general sea-level present in the Copenhagen area. Knowing that our location is directly connected to the sea this directly affects us. Studies concerning this show that 2% of the Copenhagen area are currently living below 1m elevation levels,4 % live under an elevation of 2m and 13% live 5m above sea level. Most researchers say there is high uncertainty into the rise of the water levels in the next 80 years, this being a very long period to take into account, however. The danger zones are the island of Amager and the vicinities, as well as the southern side of Zealand island. Since Fisketorvet is right to Amager, we have decided to take a deeper look into this, and we reached to the conclusion that it is possible that Fisketorvet would be underwater until 2100 in the darkest of scenarios.

Period	Scenario	Mean sea level rise around Denmark [m]
2046-2065	RCP4.5	0.3 [0.1 – 0.4]
2081-2100	RCP2.6	0.3 [0.1 – 0.6]
	RCP4.5	0.4 [0.2 – 0.7]
	RCP8.5	0.6 [0.3 – 0.9]
	A1B in AR5	0.5 [0.2 – 0.8]
	A1B in BACC	0.6 [0.3 – 1.1]
	DMI upper estimate	1.2

Figure 4. Mean sea level rise in Denmark, brackets point to the probable period for the estimation scenario

Denmark is also a country that cares about the environment. In 2016 over 40% of the energy generated comes from renewable sources. We are all familiar with the many windmills surrounding Copenhagen and present throughout the country, it would appear they provide about 42% of the energy produced.

Legal:

For now, most of the legal problems would be the laws limiting human interaction because of the current COVID-19 crisis.

New GDPR regulations in the recent years that must be followed by everyone in the service sector.

New privacy policies

PESTEL Analysis

Politic	Economical	Social	Technological	Environment	Law
-High social guarantees -High tax rate	-Very strong economy (ranked 8th in the world and 2nd in EU in 2015) -Transfer rates for countries might be high since it is not a part of the Eurozone	-Denmark has a steadily growing population which is good for a business that needs a local work force.	-Denmark was ranked most digitalized country by a study conducted in 2017 -A very wide 4G network which allows for fast access to the internet anywhere in the country	-Climate change would have quite the impact on Denmark however the price for relocation would not be worth it.	-Laws regarding COVID-19 pose some dangers for shopping in real life.

Figure 5. PESTEL Analysis

4. TOWS matrix

After carefully creating the Business canvas model, the SWOT and PESTEL, we can begin the TOWS matrix generation.
A TOWS matrix is a management tool that allows the business analyst to intelligently take advantage of the strengths of the company, by maintaining or maximizing them and how to negate or mitigate all the weaknesses that could affect the business.

S/O:

By looking at our SWOT analysis we could assume that our locational weaknesses and strengths cancel each other out. Since Fisketorvet has been around for some 20 years we could also assume that most of our customers are coming to us because of our brand and marketing team, rather than being attracted by our online presence. We could increase the influx of customers by upgrading our website to better standards i.e., better visual appearance, the order online feature and so forth.

S/T:

The Fisketorvet mall already has a decent online website however there is no shopping online option which doesn't help the COVID-19 quarantine situation. However what Fisketorvet does have is a large variety of stores which, individually, might already have an online shop feature. By creating an order online and pick-up at the mall feature we can create a sort of online shop that doesn't take away our customers as well as minimize time spent at the mall and makes it easier to do shopping in these trying times.

W/O:

If our weakness is no online shopping option then what we have going for us is the personal shopping assistant feature, although it is currently quite expensive. We could get around this drawback by using the online order feature to negate the no online shopping weakness. We could also eventually upgrade the ordering feature to have an online shopping assistant either a bot sending some personalized ads or by having a trained professional helping you choose what suits you best. By moving this assistant functionality to the website, it would cut costs by reducing the time that one person has to spend assisting a customer.
When something is online, the wealthier persons find this more accessible, most likely because of time constrains in daily life so they might be inclined to purchase online and have it delivered home. This could lead Fisketorvet into opening their own storehouse and delivery service separate to the shopping mall district and increase revenue.
Having a premium option for the subscription might be also feasible if there is a great enough attraction for this, a way of doing it would be to host limited events, offering free gym access once a week or any other means of making the customer think the monthly fee would be worth the cost.

W/T:

Our weaknesses and threats at this point are mostly the same, no online shopping feature in times when online shopping is the go-to option. If we made a classic online shop, we would just take away our customers and cut our revenue streams, however if we implement this order online and pick up at the mall feature, we will retain the current customers and perhaps even attract new ones who would like to see the products when they pick them up and having the option to exchange them at the place they bought it in if there has been some sort of a mix-up.
Buying products online has become a very common practice nowadays, however this means there are many competitors on the scene that already have years of experience in the field (Ex. Zalando, Zara).

TOWS Analysis

		Internal Factors	
External Factors		**Strengths** (S) -Good location in the city centre (S1) -Large variety of stores (S2) -Has a decent website (S3) -Has an established presance of 20 years(S4)	**Weaknesses** (W) -No online shopping option (W1) -Expensive shopping assistant feature (W2) -Current website feels a bit bland (W3) -Location isn't appealing for people in the outskirts in the city (W4)
	Opportunities (O) -Introducing an online-shopping feature (O1) -Expanding the shopping feature(O2) -Making the website more visually appealing (O3)	**(SO Strategies)** -By implementing an online shopping feature we can take advantage of our large variety of stores (O1, S2) -By using the structure of the existing website and merely upgrading it we would save resources and time (O1,O2,O3,S3)	**(WO Strategies)** -Implementing the online shopping feature and merging it with the personal shopper feature. (W1,W2,W4,O1, O2) -Making the current website more visually appealing (W3, O3)
	Threats (T) -A leangthy Covid-19 pandemic (T1) -Stricter restrictions regarding Covid-19 (T2) -Competitors have years of experience in the online shopping field (T3)	**(ST Strategies)** -Since we don't have an online shopping feature our website can't help us with the Covid outbreak.(S3,T1,T2) -Since we have an established presance we could use that to increase the public exposure to our new website (S4,T3)	**(WT Strategies)** - Minimizing the losses caused by the covid pandemic by creating an online shopping feature(W1, T1, T2, T3) -We could also implement a premium account with possible delivery to home and maybe some other bonuses related to our current services (W4, T1, T2, T3)

Figure 6. TOWS matrix

III. User Stories

User stories are the smallest unit of work in an agile framework. It is an informal, general, simplified explanation of one software feature. Its purpose is to easily make the stakeholders understand the value of each said feature. Do not confuse user stories with software system requirements. The user stories are generated from the requirements of the stakeholders, in this case they were written in the assignment paper given to us by the teachers. Some user stories are more detailed than others, even though we wanted a uniform design of the user stories by having them all follow the same format, we decided to add some tasks for some more complex or important user stories.

US1. LANDING PAGE: As a user I want to be able to display the main page of the shopping mall containing vital information such as the mall address and opening hours in order to properly plan my shopping experience.
AC: Main page is displayed.
Non-Functional
Main page is displayed and is easy to access and has a new and refreshing design.

US2. NAVIGATION BAR: As a user I want to be able to navigate between pages and back in order to select what page I want to visit.
AC: Successfully go from one page to another. Successfully go back on pressing the back button.
Non-Functional: The user is informed of any errors and redirected to main page.

US3. REGISTER: As a user I want to be able to register an account if I don't have one, so I can benefit of all the registered user perks.
AC: The system should check the username, if not used then allow for the creation of a new account.
Non-Functional: The validation should not take more than 2 seconds.

US4. LOGIN: As a user I want to be able to login to my account in order to be recognized by the system as a registered user.
AC: Successful username and password
The user is recognized by the system
Non-Functional: A user should be able to login to the account 24/7.
After 5 unsuccessful tries to enter the password in one hour, the account will be blocked. The user will have to confirm his mail and follow the instructions mentioned there.
After the mail and password were validated and the user is successfully logged in, different levels of access will be offered (based on the account type: admin, customer, employee, etc.)
Successful log in confirmation message

US5. SHOPCART: As a user I want to be able to add my purchases to the shopping cart in order to purchase them using this functionality.
AC: The purchases that I have added to my cart should all be displayed as well as showing the total price for them and the option to order(have them prepared for me). The order option should also have a pick-up time as a required input field.
US6.OFFERS: As a user I want to be able to see all offers and events for the registered customers in order to choose an offer.
AC: The system should display all the current offers and events as well as the upcoming ones in the main page.
When clicking on one of the offers the user will land on the new
Non-Functional: Only show current and future events and offers.

US7.PARKRESERVE: As a registered user I want to be able to reserve a parking spot.
AC: The parking spot should be reserved so other users can't reserve it until it expires. It should also create a receipt for the spot with information about the reservation time, its expiration time and the user information.
Non-Functional: The reserved spot should be displayed as reserved for all other users for a selected time.

US8.MAP: As a user I want to be able to see a map of the mall, so I can know where to find the stores from where I want to buy.
AC: On the main page it should display a map of the mall (if we have time we can make it interactive)

ADMIN CRUD OPERATIONS
US9. SHOW STORES: As a system admin, I want to be able to show all of the stores in the web application, so the user can see all of the shopping options available to them.
AC: The system should display all of the stores available for the customer to see.
US10.CREATE STORE: As an admin, I want to be able to create stores to the web application, so the user can have an up-to-date version of it. Store is successfully created.
AC: The admin type user should have access to and be able to execute the Create Store operation.
US11.DELETE STORE: As an admin, I want to be able to delete stores from the web application, so the user can have an up-to-date version of it. Store is successfully deleted.
AC: The admin type user should have access to and be able to execute the Delete Store operation.
US12.ADMIN EDIT STORE: As an admin user, I want to be able to edit info about the store in order to update the latest news and offers.
AC: Admin type user should have access to edit the information displayed in all of the pages.
Non-Functional: Any other user type should not be able to see those operations or be able to execute them.
US13.VIEW ORDERS: As an admin I want to be able to view orders that have been placed in order to be able to select the one that interests me.
AC: Admin type users will be able to see a structured list of orders.
US14.DELETE ORDERS: As an admin I want to be able to delete a specific order from the list of saved orders.
AC: Admin type users are able to delete the order they choose to when pressing the "Delete" button. Order is successfully deleted.
US15. CREATE PRODUCT: As an admin, I want to be able to create a product for a specific store so the user can have an up-to-date version of the goods sold.
AC: Only the admin should be able to create a product for a specific store. Product is successfully created.
US16. DELETE PRODUCT: As an admin, I want to be able to delete a product from a specific store so the user can have an up-to-date version of the goods sold.
AC: Only the admin should be able to delete a product from a specific store. Product is successfully deleted.
US17. *VIEW CUSTOMERS: As an admin I want to be able to view a list of current registered users in order to know which users are registered.*
AC: The admin is able to see the list of current registered users.
US18. *DELETE CUSTOMERS: As an admin I want to be able to delete a customer from the customer list in order to update the list.*
AC: User is being deleted from the customer list on the pressing of the "Delete" button.
Customer list is being updated and the deleted user no longer shows up.
US19. VIEW PRODUCTS: As a user, I want to be able to view all of the products associated with the specific store.
AC: The system should display all the products associated with the specific store.

IV. Diagrams and models

1. Domain model

A domain model is a representation of conceptual classes of an organization, system, a problem or any scenario that contains objects. In our case, in software development it is a way to represent the system that we must develop. It can also be used to give a basic idea of how the system works to our stakeholders or any other interested party.

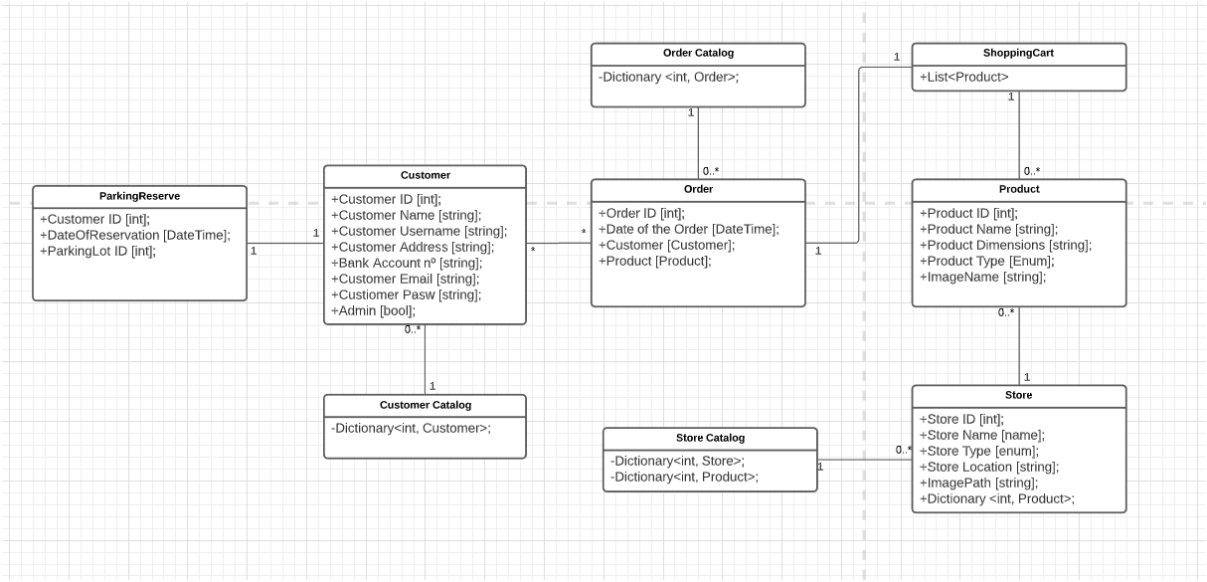


Figure 7 - Fisketorvet Domain Model

Based on the requirements for the project we first identified the main objects: Customer, Product, Store and Order. As the customer needs to be able to make an order the Customer-class is associated with the Order-class. A customer would also, in the next build, be able to reserve a parking space, thus those classes are also associated. It must be possible to add products to the shopping cart, which later will create an order, which is why they are associated. The catalog-classes serve as a repository for the methods used, so they are associated with their respective class (e.g: store with store catalog, customer with customer catalog).

2. UML Class Diagram

A class diagram is a diagram that shows the structure of a specific system. It is usually made up by classes and relationships between those classes. Each class can have a name, attributes and operations. The attributes and operations must be accompanied with a type of visibility symbol before the name. The relationships between classes can also range from inheritance, association, aggregation composition and dependency. We use multiplicities to identify how many objects from each class will take part in these relationships.
Usually, the class diagram is meant to be seen by developers because it includes all the details and implementations of the different classes.

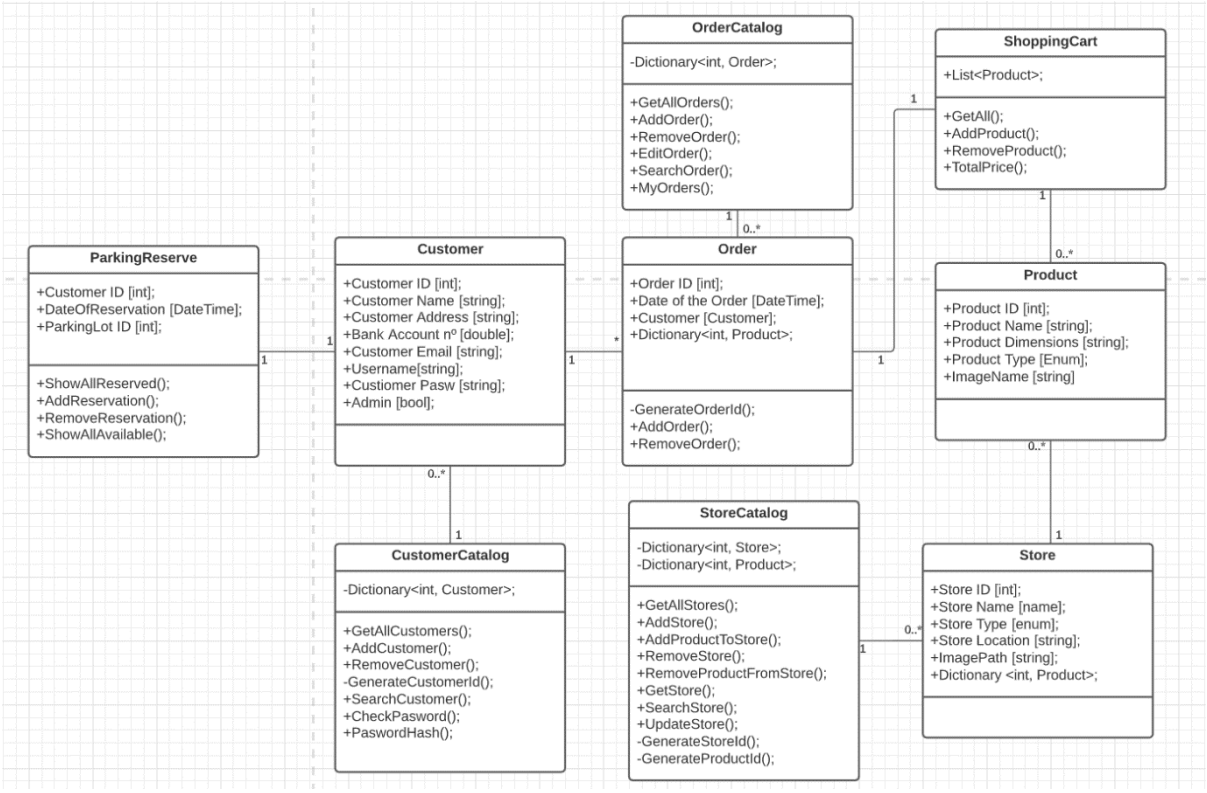


Figure 8 - Fisketorvet Class Diagram

We decided to organize all the CRUD methods in their respective catalog classes as well as adding other methods to these classes for the sake of organization. The customer class is necessary to store the customer information and for this purpose we used a dictionary. As the customer must be able to place the order, the Customer-class needs access to the functionality of the Order-class. Therefore, there is an association between these classes. The multiplicity shows that a Customer can have many orders, however an order must be assigned to a single customer.

When looking at the shopping cart class we can see the one-to-one relationship, because a shopping cart belongs to a single order. The way the shopping cart is implemented is that it gets created when a product gets added to the basket and then gets destroyed after someone proceeds to check out and all the contents get created into a new specific order. That is why we have a one-to-one relationship between shopping cart and the order.

In a single shopping cart, we can have many products that is why we have a one to many, it would not make sense having a shopping cart for every product.

The not-yet implemented ParkingReserve-class would, as the name suggests, allow customers to reserve parking in the malls' car park, doing this by using an interactive map which allows the customers see a map of reserved and free parking spots in all of the parks levels and choose the spot they wish to reserve, thus the one-to-one connection to the Customer-class. It would allow one customer to reserve only one spot at a time so spamming would not be possible.

3.Sequence diagrams

Firstly, we will touch on the basis of what a sequence diagram is and how it helps showcase the features of the project. The sequence diagram demonstrates functionality of the project and shows how the action flows in the code of the project. We will select a couple of user stories and create sequence diagrams for them.

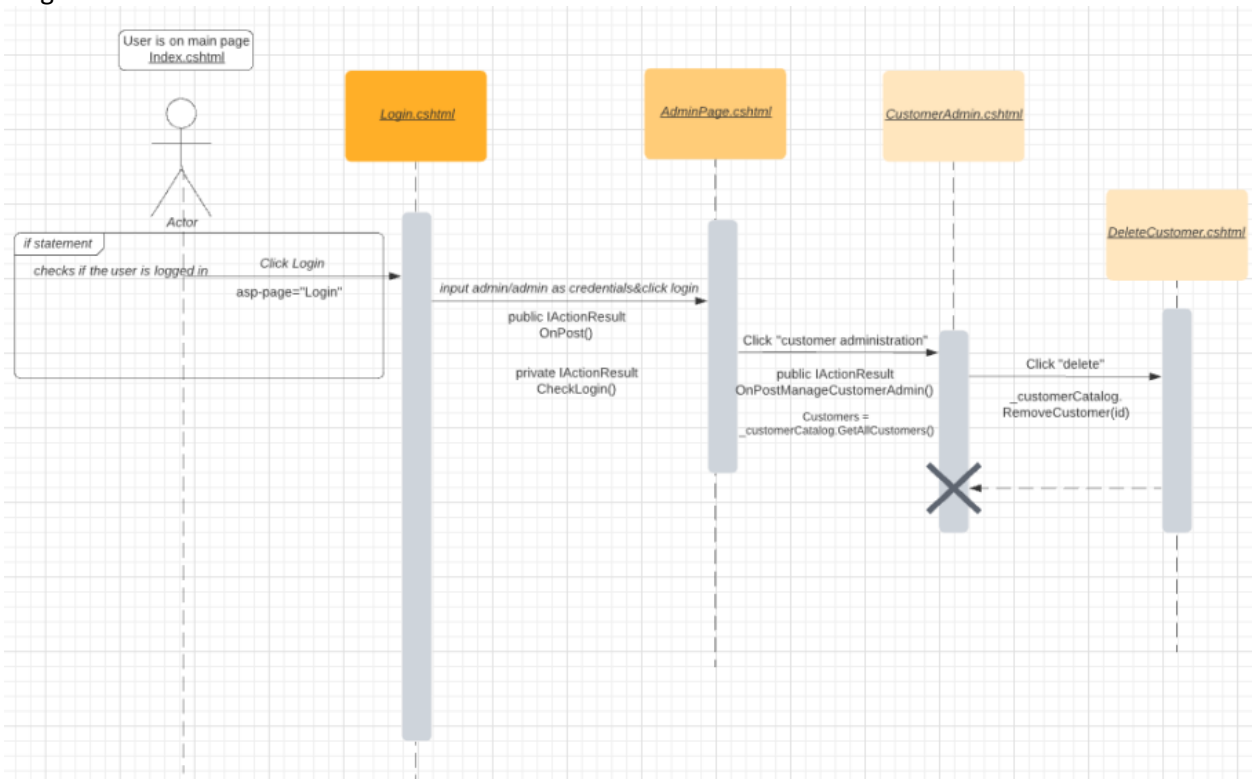


Figure 9. US11.Delete store sequence diagram

This is the sequence diagram for the Delete Customer function.
As the user opens the program he is on the main page called index.cshtml, where the page checks if the user is logged in or not by using an if() statement.

```
@{

    if (session == null)
    {
        <li class="nav-item">
            <a class="nav-link text-dark" asp-area="" asp-page="/Login">Login</a>
        </li>
    }
    else
    {
```

Figure 10. Check if user is logged in

From there he needs to log in and be recognized as an admin type user to be able to continue(do this by typing “admin” in both fields). The system then checks with another if() statement if the logged in user is admin type.

```
1 reference | pc, 1 day ago | 1 author, 1 change
private IActionResult CheckLogin()
{
    string password, username;
    username = Request.Form["Username"];
    password = Request.Form["Password"];

    bool userTry = _customerCatalogRepo.CheckPassword(username, password);
    if (userTry)
    {
        Customer = GetCustomerUsername(username);
        HttpContext.Session.SetString("cat",JsonConvert.SerializeObject(Customer));
        if (Customer.Admin)
        {
            HttpContext.Session.SetString("SessionType","adminSession");
            return RedirectToPage("/AdminSection/AdminPage");
        }
        else
        {
            return RedirectToPage("/UserSection/UserPage");
        }
    }

    return Page();
}
```

Figure 11. Login functionality check(system checks for admin type user)

The user clicks on customer administration and then clicks delete to remove the customer from the repository.

```
public IActionResult OnGet()
{
    if (HttpContext.Session.GetString(key:"SessionType") != "adminSession")
    {
        return Redirect(url:"/Unauthorized");
    }


    Customers = _customerCatalog.GetAllCustomers();
    return Page();
}
```

Figure 12. System redirect to customer management page

```
<form method="post">
    <input align="right" type="submit" class="btn btn-danger" value="Remove" style="background-color: red; color: black" asp-route-id="@customer.Id" asp-page="/Delete/DeleteCustomer" />
</form>
```

Figure 13. Deletion of customer based on his id

After the customer is deleted, he is redirected to the customer management page.

```
0 references | pc, 2 days ago | 2 authors, 3 changes
public IActionResult OnPost(int id)
{
    _customerCatalog.RemoveCustomer(id);
    return Redirect(url:"/AdminSection/CustomerAdminPage");
}
```

Figure 14. Redirecting to customer administration

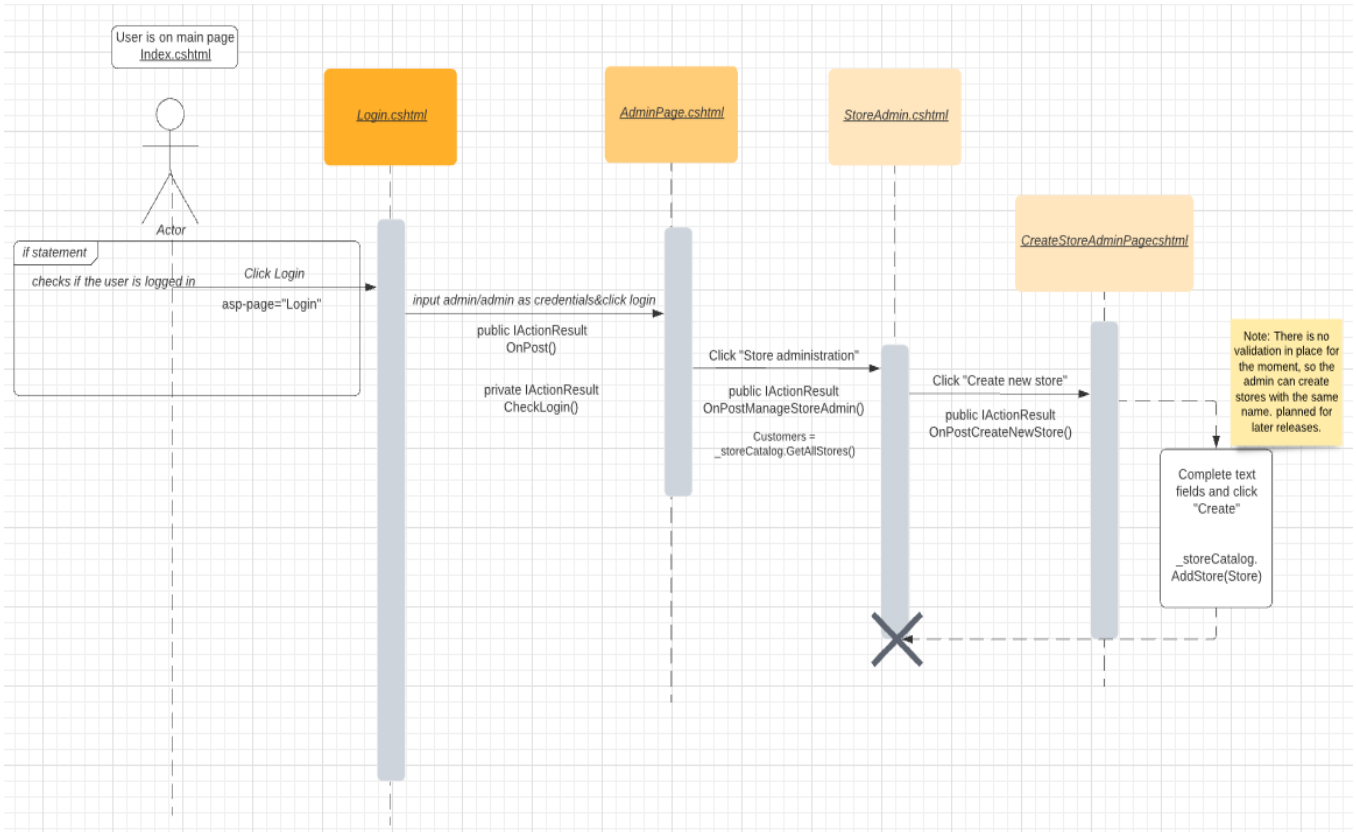


Figure 15. US10 Admin Create store sequence diagram

The user is on the main page called index.cshtml, when he opens the program, where an if statement checks whether the user is logged in. From there he needs to log in and be recognized as an admin type user to be able to continue(do this by typing “admin” in both fields). The system then checks with another if() statement if the logged in user is admin type. The user clicks on Store administration and then clicks Create new store to create a new store object. Note that there currently isn’t a a validation process in place, so that the admin can create stores with the same name, as this was planned for a later release. This is done by filling the required text fields.

```
0 references | pc, 14 days ago | 1 author, 1 change
public IActionResult OnPostManageStoreAdmin()
{
    return Redirect(url:"StoreAdminPage");
}
0 references | pc, 14 days ago | 1 author, 1 change
```

Figure 16. Redirect to store administration page

After the the store is created the user is directed back to the “Create new store” page.

```
0 references | pc, 8 days ago | 1 author, 1 change
public StoreAdminPageModel(StoreCatalog storeCatalog)
{
    _storeCatalog = storeCatalog;
     Stores = _storeCatalog.GetAllStores();
}
0 references | pc, 2 days ago | 1 author, 3 changes
public IActionResult OnGet()
{
    if (HttpContext.Session.GetString(key: "SessionType") != "adminSession")
    {
        return Redirect(url: "/Unauthorized");
    }
    return Page();
}

0 references | Paula Catalina Curca, 4 days ago | 2 authors, 3 changes
public IActionResult OnPostCreateNewStore()
{
    return Redirect(url: "/Create/CreateStoreAdminPage");
}
```

Figure 17. Redirect admin to create store page

V. GUI from implementation to design

GUI stands for graphical user interface, simply explained by what the user sees when he uses the application (buttons, formatting of text and photos).
For the Graphic User Interface we focused mainly on simplicity and accessibility so that even people no well versed in technology would be able to navigate our system without any problems. The prototype design uses colours that don't really stand out or blind you with the contrast. The color pallet changed a great deal from the initial mock to the final version in order to create a cleaner feel to it.

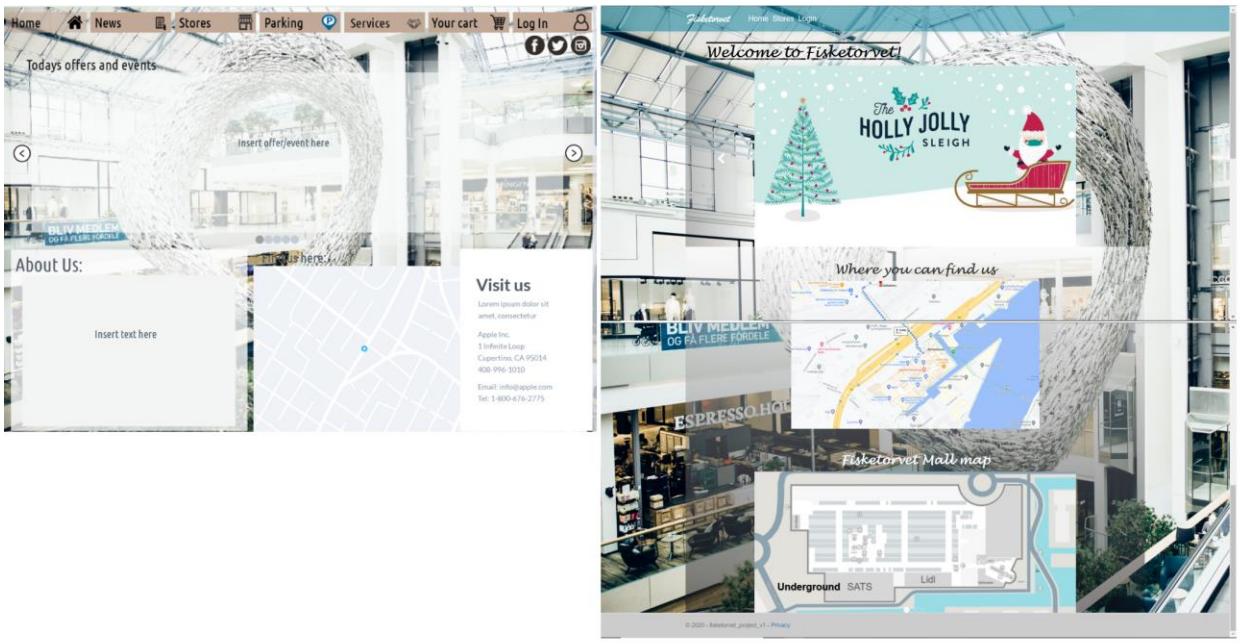


Figure 18. Initial mock of the main page vs Final result

The NavBar uses symbols as well as words to ensure that the user understands to where which button is going to take them. By using the symbols, we ensure that our customers recognize the interface rather than remember it.
We do not yet have any pictures for ease of use and accessibility next to the items present in the navbar, however we are still debating whether we should add them or not. Some buttons are also currently missing(Parking), they will come in later builds of the project together with the functionality.

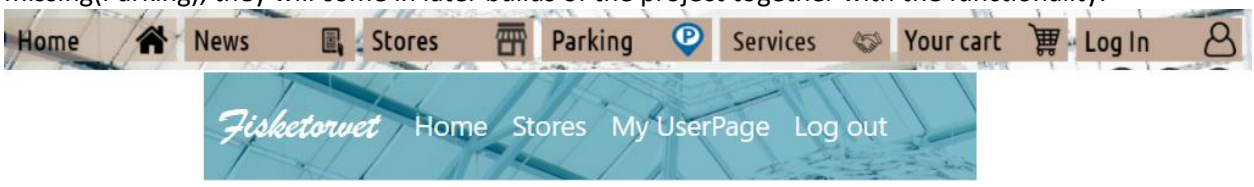


Figure 19. Initial mock of the Navigation Bar vs Final result

Knowing that users make mistakes quite frequently we decided that having cancel and exit buttons, to allow the users the ability to cancel their mistake and feel more in control.
We initially envisioned the login as a pop-up window, it redirects to a new page, however we do not know at this moment how to implement that with the given technologies. We are debating whether to change this later or not.

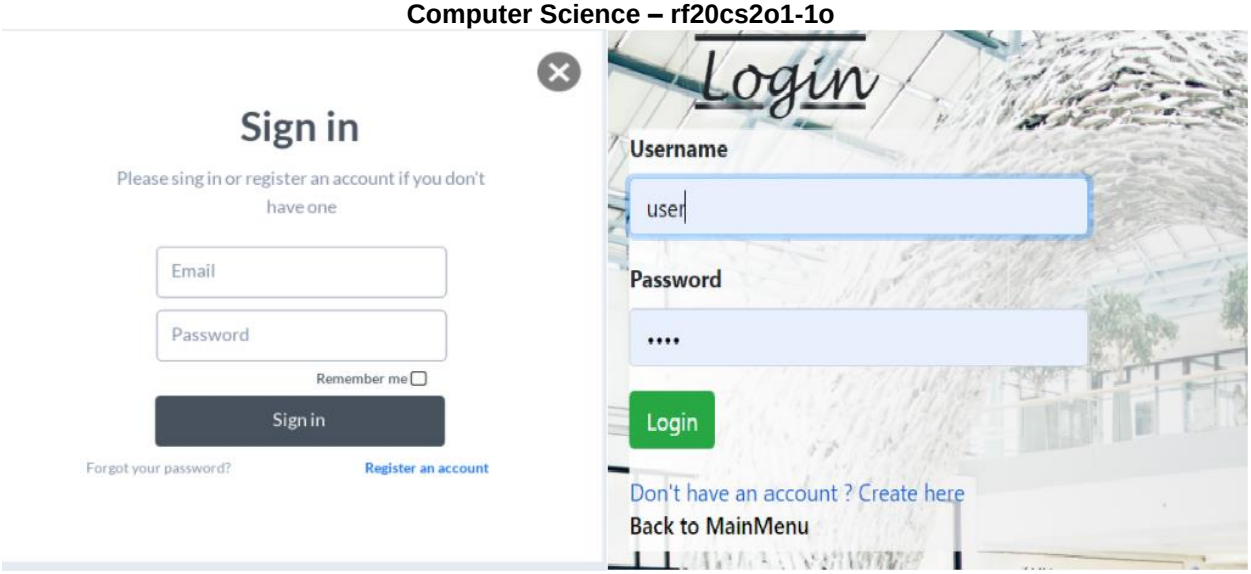


Figure 20. initial mock of the sign in functionality vs Final result

Visibility of user status is very important as well. In the prototype we changed the opacity of the selected button so that the user is aware of his location and doesn't feel lost.

Figure 21. Transparent selected "Home" button

The UI is consistent across all pages. This is to avoid confusion and misunderstanding due to deviation. A good example for this is the navbar which is the same in every page, at the same location, with the same icons and colour.
In order to facilitate user towards reaching their goals we added a functionality that whenever the user takes an action that would edit a list (shopping cart, order list etc.) the user will be redirected to the same page rather than a new one. We also added a "remember me" check-box for the log-in page, so that people can access their accounts easier and faster.
The "remember-me" checkbox is to be implemented on later builds.

VI.SCRUM and Sprints

SCRUM was the required development methodology for this project, therefore there was no flexibility in choosing a different methodology in order to manage the development of the project, however we all agree that SCRUM has a very interesting feeling to it delivering expected or in some cases even better than originally expected final product and having something to show after every finished sprint. Separated from other methodologies, it builds a very strong teamwork mentality and we believe we have made the best out of that aspect.
More so, scrum has a scrum master, which must not be confused with a boss, as his role is simply to keep the team on track and remind all of the team members about the principles of scrum.
However, next to scrum we used pair-programming into groups of 2 in order to build communication and to make our understanding of how to develop in a team.
Aside from this, we also made use of test-driven development, also known as TDD, in order to single out errors as we build layer upon layer until the point of reaching a final product.
In order to estimate the difficulty of the tasks we used the three-point-estimation method, which takes into account the best possible case, worst case scenario and the middle ground. In order to log progress and generate burndown charts we used a custom excel spreadsheet in order to keep track of the different tasks.

First sprint 23.11.2020-27.11.2020

We met on the first day of the sprint with our minds set to start discussing the Vision we have on the project and to decide how to divide the tasks and our roles. Another goal was finding the necessary tools we will require in order to create this solution.
We had daily stand-up meetings every morning, some ranging from 15 to 30 minutes, in which we always discussed with each other and found ways to help any team member for being stuck for too long in one spot.
Firstly, we decided to properly specify all of the tools we used in order to be on the same ground.
The tools used were as follow:
Microsoft Office suite, in order to have the ability of all working on the same platform, in a shared document.
Visual Studio 2019 Enterprise in order to write the code for the solution.
GitHub in order to be able to work remote and merge our contents into the same repository.
Jira, a tool that allowed the planning of the project to be more easily manageable.
LucidCharts for diagrams.
Microsoft Teams and Discord for communicating with one another and sharing ideas or links and so on.

In the first couple of days, we focused on finishing the business part of the report,
we decided splitting some tasks into 2 groups of 2 members each (ex. TOWS& PESTEL) and some tasks were done individually, such as the different blocks of the Business Canvas Model.
After generating the User Stories, we created our project and we started to draw up the Domain Model by extracting the keywords from the user stories (Ex. Store, Customer, Product). After the completion of the domain model, we created a functional Class Diagram and started building the application from the ground up.
Other tasks this sprint included:

- Creating the design part on the view page of the project.
- Add variables to the model. The logged in user will be split into 2 distinct categories (Customer and Admin).
- Creating local repositories in the form of a json in order to store the data.
- Creating authentication method called each time the user attempts to log in.
- Implementation of other functionalities.

From here on work flowed well, and we focused on implementing the user stories. As the following graph shows, we were on track in the first 2 days and we then slowly declined and fell behind the planned schedule.

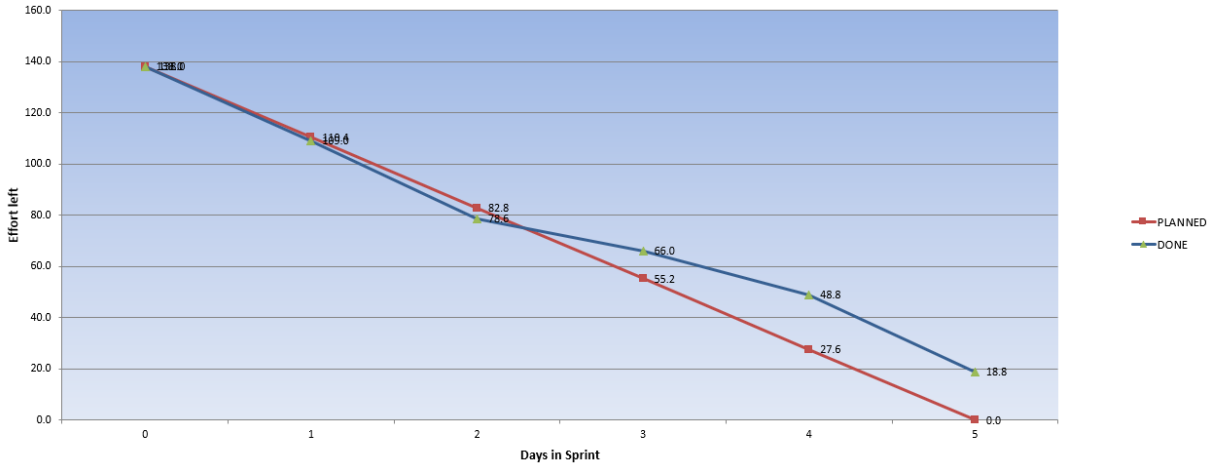


Figure 22. 1st sprint burndown chart

In retrospect to this graph, we have learned that generating a proper set of user stories and implementing them more systematically will be the best solution for the future. This was only apparent towards the end of this sprint and the beginning of the 2nd one, where the solution we had in order to deal with the lack of the original user stories, was to simply review them and work harder on them. Having the vision of the final project should have been better imprinted into our memory and follow that as the core whenever taking the next step into the development stage will be a prerequisite for the next sprints, since we have discovered that whenever we have a lack of similar vision would drag our efficiency down and we would lose focus more easily.
In order to mitigate the effect of being behind we met Saturday using Microsoft Teams, however some things remained unresolved, and would be added to the next sprint.
We managed to advance every day of the sprint which made us really excited to go forward.

What were the problems in this sprint?
In the first sprint we encountered the problem that the user stories were a bit lacking and we revised them in the 2nd sprint.
A first apparent issue was the fact that we were have used new technologies, such as Jira to track the project management and divide tasks. Learning how to use GitHub was also a welcome challenge.

Second sprint 30.11.2020-04.12.2020

For the 2nd sprint, we met mostly in person but as well as online, using Teams.
The second sprint began in force just after the 1st and we focused more on implementing functionality over design, we once more split up into groups of 2 and began our work.
First step was to pinpoint the leftover tasks from Sprint 1, this included both user stories (US2.Navigation bar, US9.Show stores, US6.Offers, US7.Parkreserve, US11.Admin delete stores) and other tasks such as revision of the product backlog and of the user stories general structure. The user stories have been separated into the ones that deal with the admin bit of the solution and all the others, this helped us to better focus and separate the admin part and the user part from one another, which in help has allowed us to continue developing the code.

After updating the product backlog, by prioritizing the user stories that we thought were more important we started working on the US10.Admin create store and we found that it was an easy implement. Other user stories such as US3.Register and US5.Shopcart were harder to implement and we have had a harder time dealing with bugs and unknown errors.

Some of the unresolved tasks from the 1st sprint have been successfully completed in the duration of the 2nd sprint (US9.Show stores, US19.View products and US13.View orders) as well as some new ones being worked on (US5.Shopcart, US3.Register), however because of the complexity the development was reaching adding some other features started becoming increasingly difficult towards the end of this sprint.

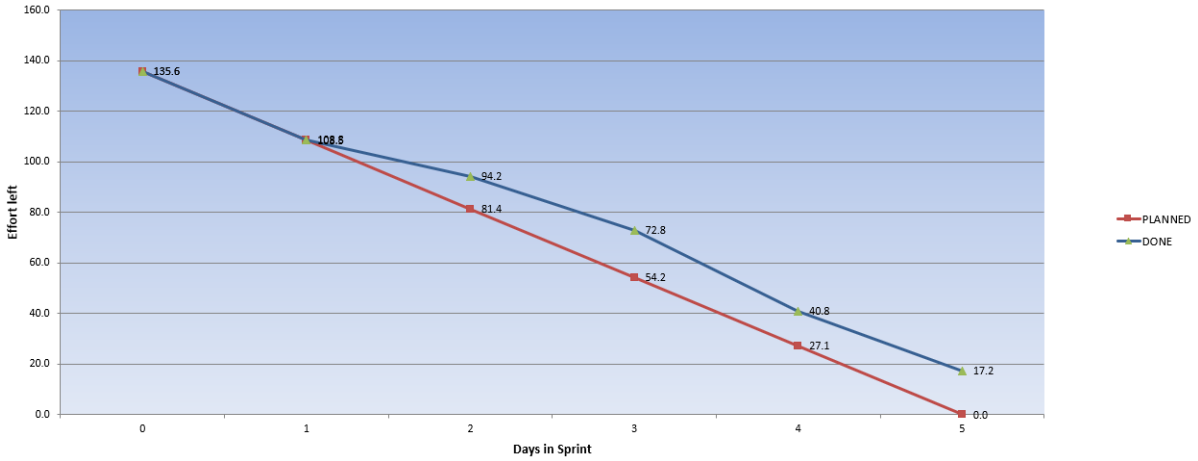


Figure 23. 2nd sprint burndown chart

As the graph shows, there was a steady start and then a tendency towards decline,simmilar to the 1st sprint. Whenever we would encounter problems they took more time than initially planned to debug. We compensated for this by doing another meeting on Saturday 5.12.2020, and by doing some extra individual work during the week. We pushed some user stories into the last sprint(Ex. US11.Delete store and US13.View orders US5.Shopcart US3.Register).

Third sprint 6.12.2020-10.12.2020

Our third sprint started in force, considering this sprint contained the last 5 days before the due date for this project we worked every day in order to properly bring forward the same product we envisioned. This sprint, once again started with the tasks from the previous sprint being identified and worked on. The product backlog, together with the user stories were once again looked over and discussed.
More functionality was implemented, mostly bug fixing and polishing from previous sprints (US11.Admin delete store, US5.Shopcart). A lot of time was taken up when we got stuck with inserting products in the store because we have used a dictionary to navigate to the stores, and another dictionary to navigate to the products within this store, and we have come to accord that this is considered bad practice.
We also added new functionality to the code in the way that the Customer can see his order history for only himself, and the admin can see all the orders in a nicely arranged list.
Another thing we added was some non-functional requirements:

- Unique username validation on register
- Password hashing for security purposes
- Automatic id generation

In this sprint we did most of the testing of the program functionalities. We then did some unit testing, however we did not document this aspect. This implied testing the buttons and that they actually take the expected action.

We also worked quite a bit on designing the updated visual part of this project version.

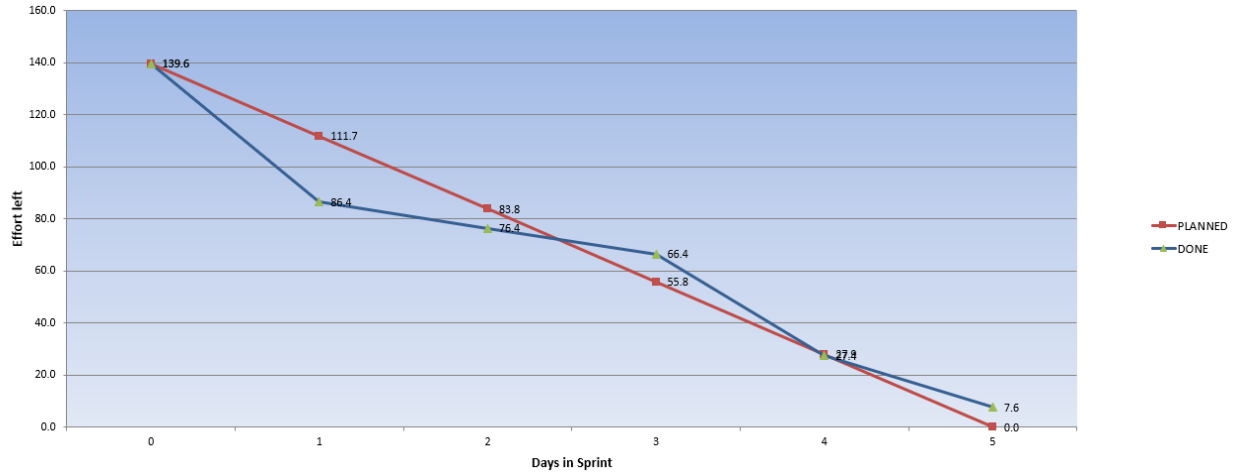


Figure 24. 3rd burndown chart

As you can see in the first two days we appeared to be on schedule, however we later got stuck and fell behind schedule. The consequences following this were that we could no longer implement the US7.Parkreserve. We also didn't add any bank account property for the customer.

VII. Retrospect

1.End result

In retrospect, we have learned a lot related to the development of an application and project management. We have gathered experience with the use of agile development methodology and we now better understand how to plan and progress through the different development phases of the project.
Regaring our learning goals, we believe we have acheived most of what we set out to learn. We got a lot of experience with report writing, backlog generation, C# and RazorPages, HTML and CSS integration into the user experience. During the generation of the solution, we have learnt a great deal about visual graphical user interfaces and how to make them more pleasing to the eye.
We believe we have followed our vision and that the end result is more functional than we set out to do and we are happy with this.

2.Future of our product

We will continue the development of the solution even after the hand in date, which is 12.11.2020. Main goals are working on the design, parking reservation system and some more validation and non-functional aspects of the project, such as load times and security. We also want to edit the footer to add contact information.
Since we enjoyed the development process in the current team format and we wish to enrich our experience as a team and individually, we will stay together as a group for the further development of the project.
We strongly believe we can improve the quality of the product, especially from a visual standpoint.

3.What to improve on

Estimation is quite hard to properly make in your first agile project, however we believe that with the experience we will gain during these years of study we will manage to better estimate in the future.

Next time we want to properly document some unit testing, and introduce it in the report.
Unfortunately, this time we did not do any unit testing documentation because of time constraints.

We believe that the customer class has become bloated with the ammount of properties inside of it, we believe we could have used the following inheritance Person : Customer, the Person would be the main class and handle some attributes of the Customer class, therefore making it more simplistic and more maintanable and easier to update.

We also believe there are several mechanisms in our code that could be improved. For example the way we implement products into a store made it very difficult to add more features regarding those products. We could have done a single file JSON file for the products and then have them contain a storeID to match their respective store.

After closer inspection, some parts of our code are being repeated several times, and we know that it is considered bad practice, however at this time we are unable to better implement the functionalities that depend on this repeated code. Because of the time constrains.

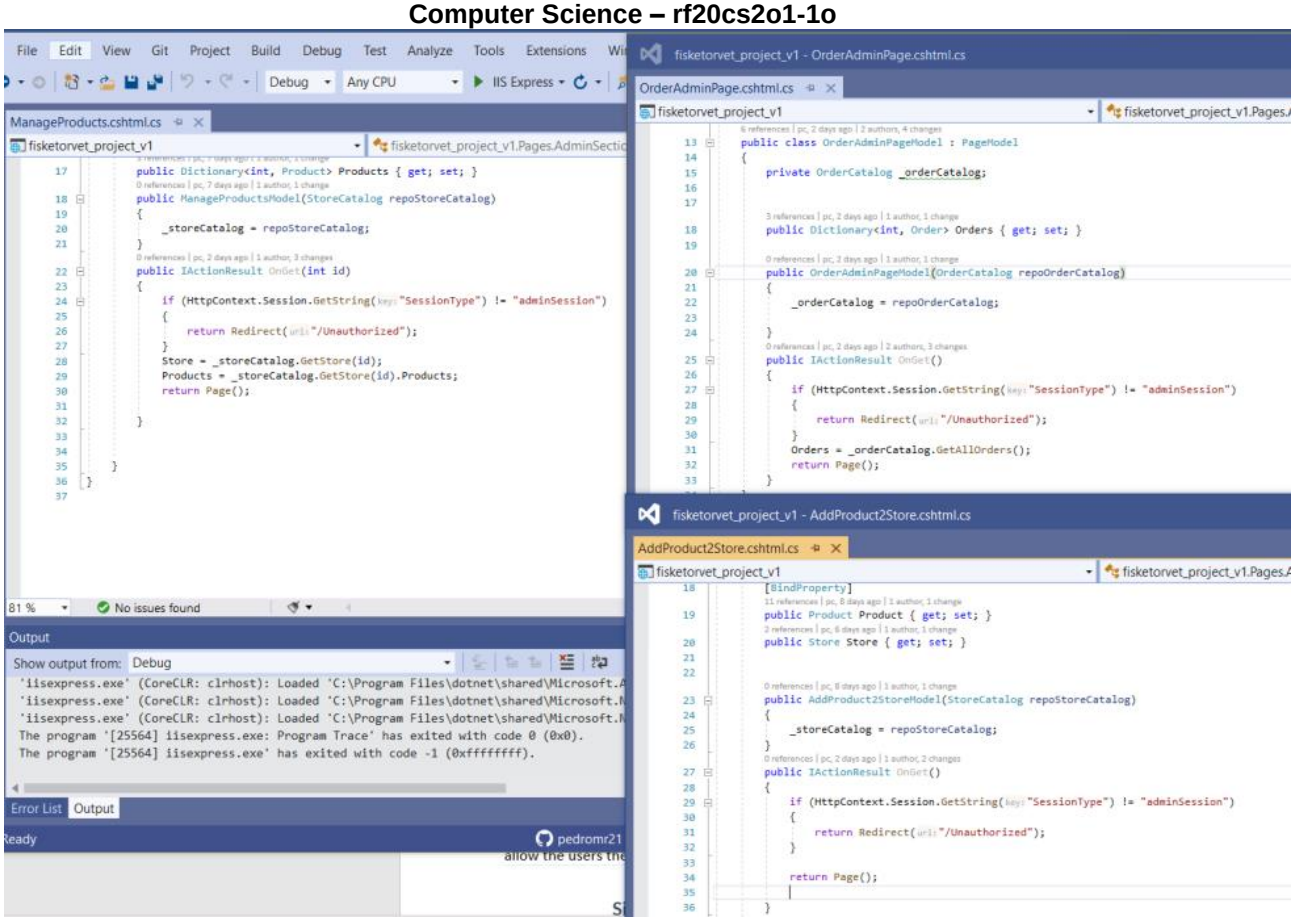


Figure 25. Instances of code repetition

In the Design part of the project there is still a lot to learn, a lot of time was spent on testing different boxsizes, transparency levels, color palletes and so on. We all agree that we have already learnt a lot during these three development weeks in this aspect, but we understand we still have a long way to go until we can make our solution look as visually appealing as possible.

4.Final thoughts

Firstly, we would like to extend our thanks to our supervising teachers for all of their assistance in these 3 weeks.
We are looking forward to further developing this product and we hope to keep you interested in the future releases!

Glossary

US-user stories
AC-functional requirements-acceptance criteria(what is the program function?)
Non-functional requirements – how should the program act?
UI – user interface
Multiplicity – shows in the class diagram and domain module how many objects are used from each class
Repository – a location where data is stored and managed
Product backlog – a report of the product
Sprint – a set time period for some tasks assigned
HTML – hypertext markup language, the standard markup language for documents designed to be displayed in a web browser
CSS – cascading style sheets, a programming language used for describing a presentation of a document
Razor Pages - a server-side, page-focused framework
NavBar – Navigation Bar

References

The dates represent the timeframe in which we accessed said references.
23.11.2020-28.11.2020
https://en.wikipedia.org/wiki/Three-point_estimation
http://www.agilemodeling.com/artifacts/useCaseDiagram.htm
https://en.wikipedia.org/wiki/Economy_of_the_European_Union
https://en.wikipedia.org/wiki/Denmark
https://www.learnrazorpages.com/
http://files.avanquest.com/file-extension-json/
https://agilewarrior.wordpress.com/2010/11/06/the-agile-inception-deck/
https://www.atlassian.com/agile/project-management/user-stories
https://agilelearninglabs.com/resources/scrum-introduction/
https://www.atlassian.com/agile/scrum
https://blog.onefire.com/why-a-business-model-canvas-is-important-to-organizational-innovation
https://www.strategyzer.com/canvas/business-model-canvas
https://www.dst.dk/en/Statistik/emner/erhvervslivets-sektorer/serviceerhverv
https://en.klimatilpasning.dk/knowledge/climate/futuresealevels/#:~:text=On%20the%20basis%20of%20existing,2100%2C%20depending%20on%20the%20scenario.
https://pestleanalysis.com/denmark-pestle-analysis/#Final_Thoughts
https://www.climatechangepost.com/denmark/coastal-floods/
https://www.world-nuclear.org/information-library/country-profiles/countries-a-f/denmark.aspx
https://investindk.com/insights/denmark-most-digital-country-in-the-eu#:~:text=Denmark%20tops%20the%20Digital%20Economy,testing%20and%20developing%20new%20technology.&text=Denmark%20is%20the%20most%20digital,Society%20Index%20(DESI)%202017
https://www.nngroup.com/articles/ten-usability-heuristics/
https://strategyzer.uservoice.com/knowledgebase/articles/1194370-how-do-i-use-the-value-propositions-building-block
https://um.dk/en/news/newsdisplaypage/?newsid=01fc577b-6bf2-4fd7-8572-5af0534cf599
29.11.2020-5.12.2020
https://olegchursin.medium.com/a-brief-introduction-to-domain-modeling-862a30b38353
https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/
06.12.2020-10.12.2020
https://stackoverflow.com/questions/3413715/how-to-create-custom-data-annotation-validators

https://www.itgovernance.eu/da-dk/eu-gdpr-compliance-dk#:~:text=In%20Denmark%2C%20the%20new%20Data,states%20to%20interpret%20and%20implement.&text=In%20Denmark%2C%20the%20fines%20are,than%20the%20Data%20Protection%20Agency