

Section 1: Project Proposal

1.1. Project inception: project overview

Project Name	Freshco MVP
Date	10 th December 2020
Project Manager	Oscar Rivas
Project client	Freshco LLC
Proposed Project Start and End	January 2021 – May 2021

Identified Need

Peruvian households cannot access grocery shopping as they used to do before the quarantine that limited their ability to purchase all the products they want. According to reports from the Ministry of Health, the food markets are the main sources of contagion. In Peruvian economy, more than half of the population interacts on informal economy and the majority of the families acquire their goods through the traditional channels. Also, as the contagion cases and deaths have been increasing, the government has established strict restrictions and limits the citizens mobility with curfews. Many businesses had to stop operations and many families had their income sources reduced. This problem generates an opportunity for a disruptive business model and an innovative digital solution.

Project Purpose

Developing an ecommerce start-up is the best way to implement both the solution and the business model. Therefore, the purpose for this project is to develop the system which the business is going to use to deliver value. The core value proposition for the business is: “Your favourite food with competitive prices, direct to your door and without you having to think about it.” The proposed solution for the problems includes as main benefits: “Minimizing COVID-19 exposure”, “Reducing choice anxiety”, and “Saving time and money”.

Project Justification

According to the report of Euromonitor, the market size for the ecommerce grocery industry is 250 million Peruvian Nuevos Soles (PEN). By considering only the capital of Peru, Lima, there is an available market size of 10.91 million PEN. Furthermore, according to the report of Ipsos Peru, the aggregated value of the monthly expenses of the top 5 socio-economical family levels (A1, A2, B1, B2, C1) is approximately 2.17 billion PEN. Indeed, there is a relevant total addressable market opportunity.

The current incumbents on the market have been forced to implement some tech solutions but the evidence suggests that traditional industries are not good at implementing tech at scale. Therefore, the suitable approach for the development of the business model involves a core tech development. In the words of David Veléz, the small distinction that can generate value is “Being a tech company that happens to be on the business whereas others are just companies that use technology”.

Project Description

Undoubtedly, as technology needs to be a core part of the business model, the development of the system is crucial. The system must be able to supply the value proposition. For this development project all the required process and activities will be executed. The project will involve generic processes such as: Communication, Planning, Modelling, Construction and Deployment. The way in which they will be executed will depend on the selected development model. The budget is contemplated on the seed capital provided by the investors.

System General Description

To be able to provide the totality of the value proposition the system needs ecommerce, business intelligence and supply chain and logistics modules.

Ecommerce module is required to be able to accomplish the sale of the goods. This is conceived as a progressive web app that is delivered to the clients through the Internet either on mobile or desktop. The items are presented on the site and the customer can make purchase and keep track of them. The platform offers payment solutions to guarantee the transactions and produce financial information.

Business intelligence module is required to produce relevant information for the decision making. This works for the customers and the business. The customers will get on the ecommerce side the suggestions for purchases based on their preferences and history. The business will get the aggregated information about the transactions and can make better decisions based on the worked data.

The supply chain and logistics module are crucial in order to be able to deliver the goods to the clients. First, the items to be sold are provisioned and stored on a warehouse. While the transactions happen on the ecommerce module, the data about the orders is sent to the warehouse and the packs of the purchases can be produced. Later, with the information of the transaction the products can be delivered to the client's door in an optimized manner. With the support of the Business intelligence module the role of supply chain management can optimize operations and run smoothly.

Project Scope

As described previously, the requirements for the system are quite ambitious and complex. The development of this system would require an important investment and rigorous project operations. It could be suggested that it is impossible to complete the project without some special conditions. Indeed, the feasibility for this project is none if the scope is kept on the whole system.

Nevertheless, provided that for this case the business is an early start-up, the concept and scope need to be adjusted to be feasible. The scope must be reframed to deliver the most basic but important features of the system. The objective of the project is to develop a Minimum Viable Product (MVP) to deliver the promised value by the business and experiment to demonstrate the viability of the business model. The modules mentioned above need to be sized down to the most important requirements to find the ones that deliver the most value for the business and operations.

High level Requirements

The most important features for the system would be the ones that account for the complete macro process of the business:

- The clients should be able to register on the site and provide personal data and their preferences.
- The clients should be able to select a predefined bundle of items and customise it.
- The clients should be able to register or select an address for delivery and they should be able to select a delivery window on predefined dates.
- The clients should be able to execute payments by card.
- The clients should be able to receive an invoice and track the purchase.
- The clients should be able to review the purchase, provide feedback and rate it.
- The system should be able to make recommendations on the beginning of every new purchase based on the previous data.
- The system should be able to output data that can be processed to generate insights for the business.

High level Risks and Feasibility

Since the project is a venture capital backed initiative, the development is expected to produce results as an MVP. Therefore, the feasibility is strictly related to the business case and the technical aspects. Some of the most important risks are:

- Project Management risks
- Technical risks
- Organisational risks
- External risks

Project Objectives	Success Criteria	Person Approving
Scope		

Freshco system MVP	Released on time, at/under budget	Oscar Rivas
--------------------	-----------------------------------	-------------

Time

4.5 months to completion	Completion: May, 2021	Oscar Rivas
--------------------------	-----------------------	-------------

Cost

\$12.2k USD Aprox.	Project cost </= \$20k USD	Oscar Rivas
--------------------	----------------------------	-------------

Summary Milestones	Due Date
Sprint 1	Week 1
Sprint 2	Week 4
Sprint 3	Week 7
Sprint 4	Week 10
Sprint 5	Week 13
Sprint 6	Week 17

Estimated Budget

\$20k USD

Stakeholders	Role
Oscar Rivas	Freshco LLC CEO and Project Manager
Elmo Calatoyud	Freshco LLC COO and Agile Coach
Carles Almendez	Freshco LLC CTO and cloud engineer
Cami Xiom Fernan	UX/UI Designer and Front-end developer
Jose Elchaman	Full Stack developer
Andrea Palomino	Data scientist
Karlek Freidory	Warehouse manager
Jorge Castillos	Angel Investor
Emprende UP	Incubator

Others:

- Clients
- Delivery team
- Suppliers

Technical and physical environment

The system is going to be deployed on the cloud; therefore, we won't consider physical instances. Instead, we will describe the deployment based on a cloud structure.

The client through its device accesses the ecommerce through the Web browser. There, it accesses an instance of static files that compose the landing page that are located on a storage instance. There, the specific web app dedicated processing instances will do the work. Through queries they will connect to the database instances and make the exchanges. The other components will be connected through REST APIs. Some on them could be either the payment entities such as banks or others.

Performance goals

By considering the cloud deployment, the main goals of the system should be scalability and adaptability. The system needs to be scalable to attend a continuously growing demand. It must be adaptable to implement new features over a compatible underlaying architecture

1.2. The development process: choice and justification

Development model choice

The chosen developing model is Agile development. It is argued that traditional system development models are better suited for larger structured system development. Indeed, Agile offers an alternative to the traditional models because it is based on the premise that nowadays uncertainty is the rule. This means that to face the current issues, Agile offers an alternative to the perfectionist complete implications of the traditional models. This kind of approaches have emerged to offer a way to face the changing customer needs and the continuous advancements in technology. On its core they are characterized by flexibility and adaptability.

Development framework explanation

The chosen framework to develop under the Agile concept is Scrum. Scrum is one of the most used models of Agile development. Scrum is not only focused on the system development, but it is also focused on the human and project management aspects. Under the core principles of the Agile Manifesto, it is relevant to point how Scrum fits in the model. First, the main priority is the continuous delivery of value threw working software. On fixed timeboxes the specified requirements are developed. This is achieved due to the division of big problems into smaller ones. Then, these small problems are prioritized and on each cycle are selected to be worked on. Furthermore, to be able to deliver the developments for the small problems on the fixed timeboxes the team must only focus on what generates value. In contrast to traditional development models, the focus remains on productive activities instead of giving time to bureaucratic or non-essential activities. The constant delivery also implies that there should be a constant testing to deliver the appropriate working products. Feedback is an essential part of the process. At the end of every fixed period there is a delivery presentation and the customers will test and judge the product. Based on what they experienced, they will be able to communicate properly their concerns and opinions about the delivery. Also, as there are many changes or unexpected situations that develop, there is a need to realign the course. This means that contrary to perfect planning expected on traditional development, on Agile models there is an important focus on the ability to pivot and shift the focus of the actions. Finally, accountability is an essential characteristic of Scrum because of the constant customer revisions. Since there would be constant deliveries, only the team would be responsible of guaranteeing the success of the project. This means that on every cycle they must own the quality, own the schedule, fulfil the expectations, and keep a tight budget.

Justification

Start-ups operate under circumstances that difficult having any certainty at all. They operate based on a hypothesis for a business model that must be verified with the customer's feedback. Many times, it happens that entrepreneurs come up with a solution and try to find a problem for it. And many times, they

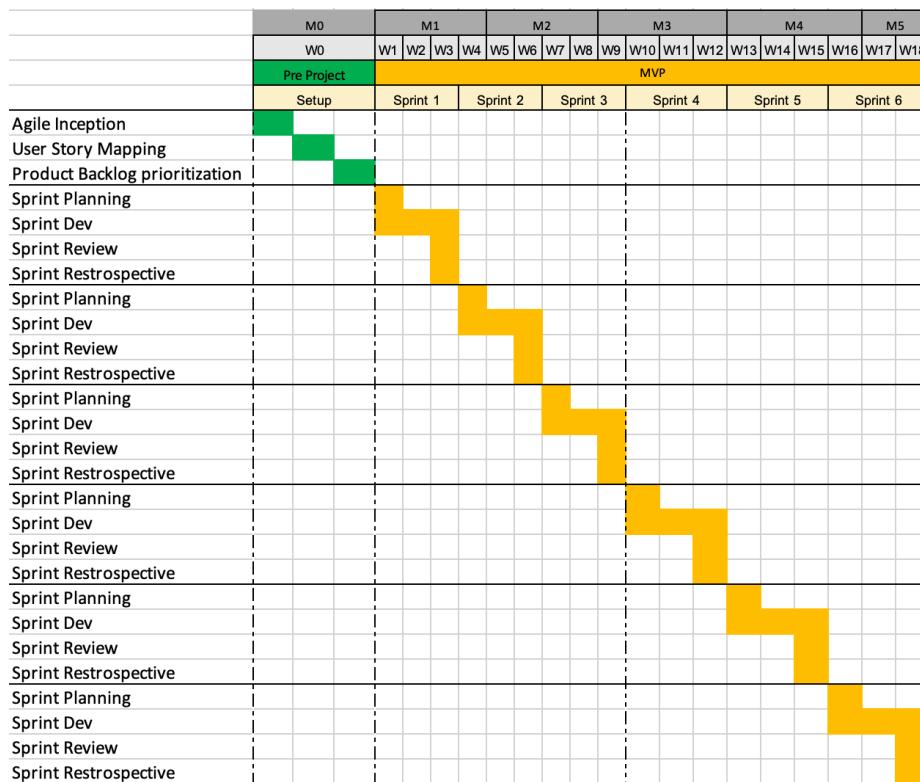
end up failing. Under the spirit of the lean, start-up is how it could be possible to follow a path to correctly identify a problem and build a customer-centric solution. Once it has been demonstrated that the solution is appropriate and valid, it must demonstrate that the business is feasible. This means that while it is true that a solution might be perfect for a problem, it might not be sustainable in the long term. This might be due to several factors such as existing competition, entry barriers, costs, no existing demand, etc.

To become sustainable companies, start-ups must go beyond the solution phase and achieve the product/market fit. This means being able to find a sustainable solution that has an addressable market to contest. Certainly, for this context Agile development is the best choice. As stated before, Agile development works best on uncertain contexts such as those for start-ups. Start-ups are actively looking for what works best for them. Changes and crisis are inevitable, as a matter of fact they are part of their DNA. Therefore, the Agile model is the one that best fits both their context and needs. Also, as the project is focused on a tangible product/service, it is preferable to adapt a framework that also contemplates operations management. Scrum is the most suitable framework because it is compatible with the kind of project developed in the case. Furthermore, as the nature of the development of the product demands constant human intervention, Scrum is optimally suited to tend those needs.

1.3. The development process: planning

As the Agile development process does not consider a formal timetable, the appropriate way to portray the development cycle would be one that has the appropriate Scrum instruments.

Figure 1: Timetable



For a fixed 6-month period the development of the system will take place. This will be enough time to develop an MVP for public release. The release is contemplated at the end of the period. Every Sprint has a fixed length of 3 weeks. We only account workdays, which accounts for 15 days. On every Sprint there are the same recurrent stages, which are the Sprint planning, Sprint development, Sprint Review and Sprint Retrospective. The content on every Sprint should be determined on the Sprint planning and it would be

extracted from the Product Backlog. There is the possibility that certain issues transcend the sprints and those are grouped on epics. Also, there is a special period named as Pre-project that accounts for the project setup.

Even though for Scrum the documentation is expected to be minimal, there are still some instruments or documents that need to be considered. The expected documentation to get on this project of the Scrum artifacts is:

Agile deck

A deck of options aimed at avoiding misunderstandings and fostering dialogue in the initial phase of a project. It is a collection of ten tough questions and exercises that help by promoting the dialogue and producing high level distill of the project.

Product backlog

Document that contains the set of functionalities, improvements, technology, and bug fixes that must be incorporated into the product through successive development iterations. It represents everything that customers, users, and those interested in the product expect. Everything that involves work to be done by the team has to be reflected in the Backlog. It is the responsibility of the Product Owner to make this definition and establish the priorities of each requirement. It is a high-level document, which contains generic (not detailed) descriptions, and which is subject to modifications throughout development, it is in continuous growth and evolution. Higher level of detail at higher priority and lower level of detail at lower priority.

Sprint backlog

It is the list that breaks down the functionalities of the Product Backlog into the tasks necessary to build an increment: a complete and operational part of the product. Breaks down the project into appropriately sized tasks to determine daily progress and identify risks and issues without the need for complex management processes. It is a support tool for direct team communication. Only the team can modify their tasks during the same sprint, and it should be visible to everyone, ideally on a whiteboard or wall in the same physical space where the team works.

Sprint review

Feedback session. Each product build iteration includes a detailed review with customers. Once the sprint cycle is finished, a meeting is held in which it is defined what part of the planned work has been completed and what part remains pending. Regarding the completed work, a review (demo) is made to the Product Owner and other users who may be involved.

Sprint retrospective

A final review meeting of each iteration, which allows the process participants to provide feedback on the process (not the product) identifying what was done well and what can be improved, trying to introduce a component of continuous improvement in the process. All team members carry out an assessment of the work carried out in the last Sprint, identifying points for improvement with a view to the next ones to be carried out. The feedback obtained in this meeting can be included among the functionalities to be built in future Sprints.

Taking in consideration some important documentation used on traditional models, it is relevant to mention that some diagrams based on UML should also be considered.

Section 2: Development process documentation

2.1. Pre-project

2.1.1. Agile Inception

On Agile projects, the project chartering is often not explicitly discussed. On traditional models, this would involve a lot of time spent on analysis and requirement gathering. Therefore, a way to engage the team and getting them to discuss is necessary. The Agile Inception is a project setup event that on a lightweight manner distils the project to its core and generates a mutual understanding on the team. This is done through the Agile deck, a set of 10 questions that help aligning the expectations.

Purpose of the meeting

Here the members of the Scrum team presented themselves, talked about what they do and described what they like. Here we have 6 key members of the Scrum team. The members are the following: Oscar Rivas (CEO and Product Owner), Carles Almendez (CTO and Cloud engineer), Cami Xiom Fernan (UX/UI Designer and Front-End developer), Elmo Calatoyud (COO and Scrum Master), Yorkline St Lauden (Data scientist), and Jose Elchaman (Full Stack Developer). Also, the members declared what their understanding of the problem was. In general, the perspective of the was directed towards supply chain management, digitalization, biosecurity measures and data handling.

Figure 2: Purpose of the meeting



Elevator-Pitch

Here the team members made a high-level description of the project and the solution. The pitch is the following: For the population of Metropolitan Lima, who are exposed to a high contagion risk because of the high concentration of people and the difficulty of the implementation of biosecurity measures, FRESHCO is an ecommerce of food/beverages, which means to position as the principal ecommerce of food and beverages by delivering value based in tech. Compared to Supermarkets or other e-commerce, our product guarantees a complete personalization on the purchase of groceries and home supplies.

Figure 3: Elevator-Pitch



Design a box

The members designed a concept box to portray how the product hypothetically would look if it was boxed and sold in a supermarket. The slogan of the product is: “Tailored food to your door without the need of thinking”. An attractive feature that the product offers: 99% precise recommendations based on machine learning.

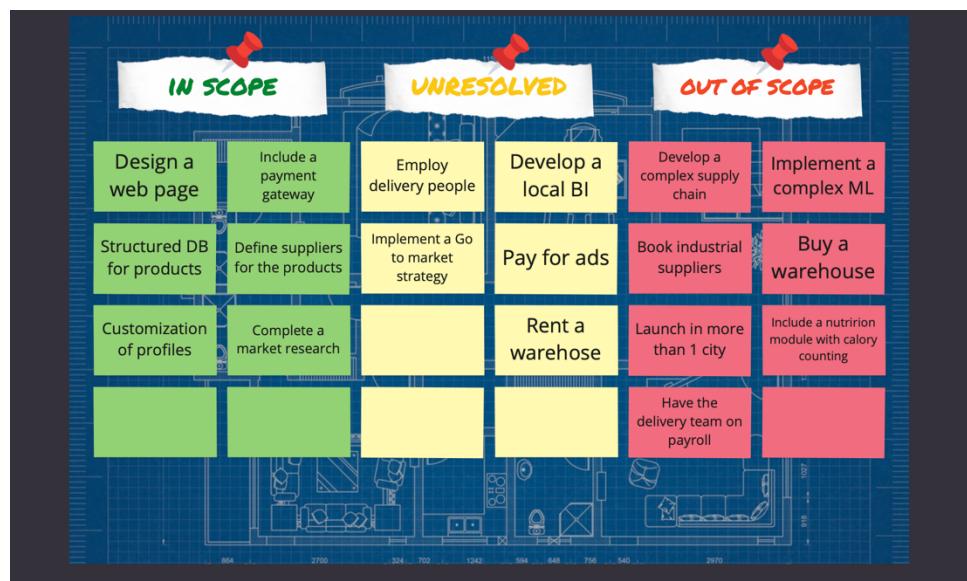
Figure 4: Design a box



Limiting the scope

The members defined on a high level what is going to be considered on the scope of the project. In general, what is considered on scope is related to the basic features of the ecommerce module of the system. Elements regarding BI and SCM modules are considered either unresolved or out of scope.

Figure 5: Limiting the scope



Know your neighbours

The members defined other agents or parties that are involved and directly or indirectly influence the project. The members positioned themselves at the center as the core team. They also identified as allies the incubator and a logistics solution supplier. Some 3rd parties that were considered relevant are either big services suppliers or state entities

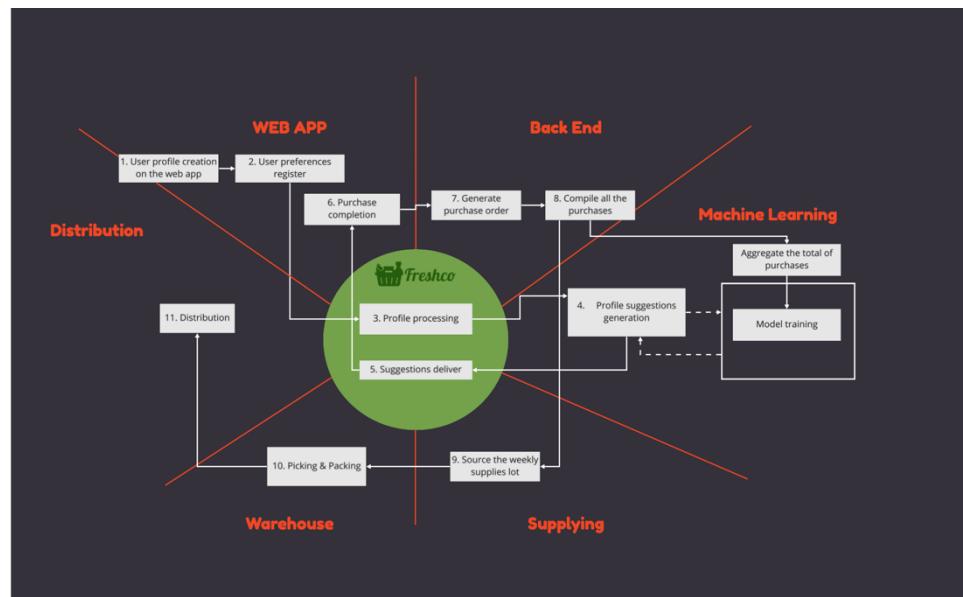
Figure 6: Know your neighbours



Frame the solution

The team defined their collective initial high-level idea of what the architecture of the solution could be. Here there is a main flow of which general activities the system executes and different areas in which it does so. All the steps concern the fulfilment of an order.

Figure 7: Frame the solution



What keeps us awake at night?

The team elaborated a Risk Matrix to identify which possible risks are associated with the development of the project. They later classified on different ranks according to probability and severity. In general, all the risks are associated with operational concerns. The most critical are related to the influence on SC and less critical ones have to do with the systems internal functioning.

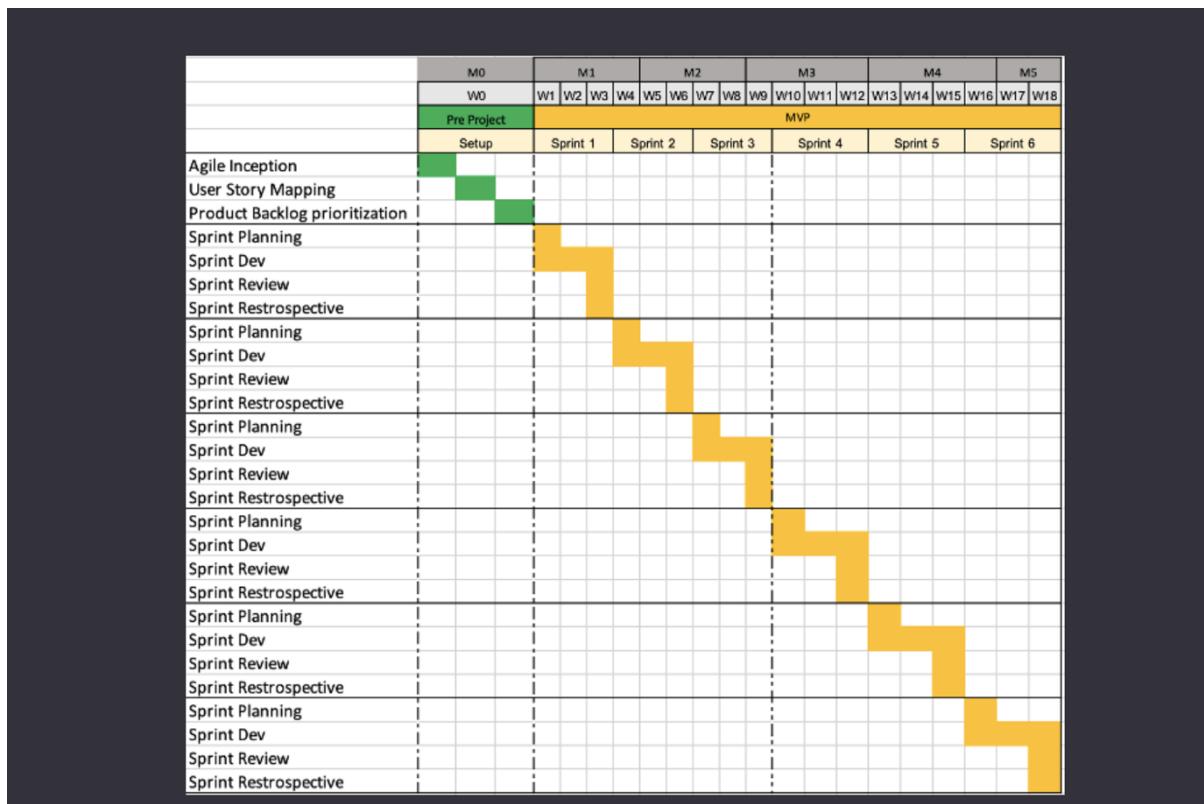
Figure 8: What keeps us awake at night?



Global estimation

The team defined a high-level estimate of schedule for the operations. Here they defined that the Sprints are going to be 3 weeks long. Also, they defined that each release or phase is composed of 3 Sprints. In addition, there are 3 releases that are considered, with the 3rd one being the public release of the MVP.

Figure 9: Global estimation



Trade-offs

The team identified the most important aspects to consider while the project is executed. They discussed and prioritized them. In general, the most important aspect is the delivery on schedule. Quality and assurance are important aspects but less rigid than delivery on schedule. Also, delivery on budget and the scope are slightly more flexible and simplicity is the most flexible aspect.

Figure 10: Trade-offs



How much?

The members identified and estimated of how much the project is going to cost. There is already an established budget, however, they projected the costs of salaries for the complete duration of the project. By considering the quoted average salary rates of the different roles, the estimate goes as high as 13 thousand USD. It is important to note that the co-founders manifested that they will not receive payments during the first 3 years, and this is established on their vesting deal.

Figure 11: How much?

USD			
Data Scientist	Full stack dev	Designer	Total
1,068.54	842.28	801.40	12,204.99
Co founders dont cash on the first 3 years			

2.1.2. User Story Mapping

On Agile development processes documentation is not a priority. It is often argued that documentation itself does not provide business value and is commonly discussed that it is not an effective way of capturing requirements. Therefore, for the Agile development model we applied the user story mapping to represent the requirements.

Framing the problem

The team members previously identified some problems that they considered to be the basis of the project. The co-founders working from the business perspective also provided an additional perspective as a Business Case. On this part of the project, we also considered the input of the parallel lean start-up process.

Figure 12: Framing the problem

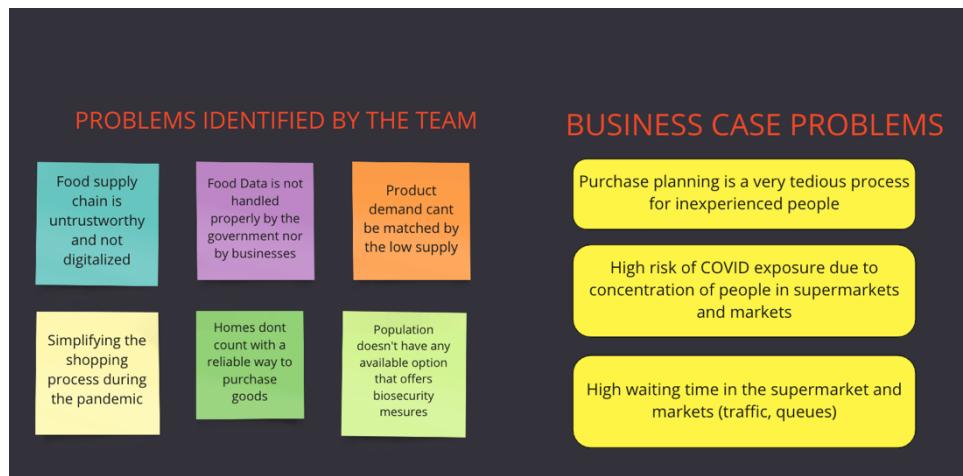
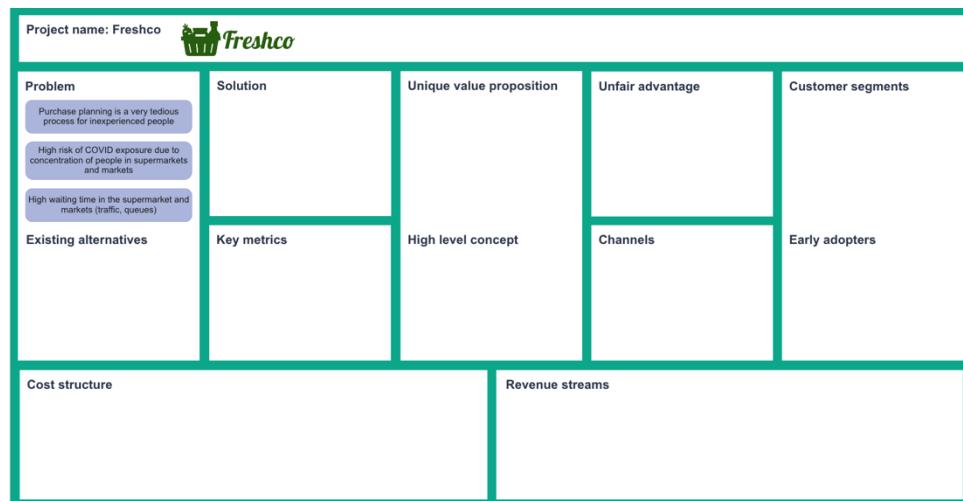


Figure 13: Lean canvas - Problem



User Roles

The team members conducted a brainstorming exercise to identify the possible users. Each of the users wrote as many as they could imagine and after that they discussed and identified similarities that could be useful to identify general users. This includes both clients, possible clients, and employees. In addition, the co-founders provided a general idea of what the customer segment likely is.

Figure 14: General User roles



The general descriptions for the general users are:

Shopper: Any person that can be attracted to visit the page or try the service. Does not necessarily complete a purchase but has the interest to do so.

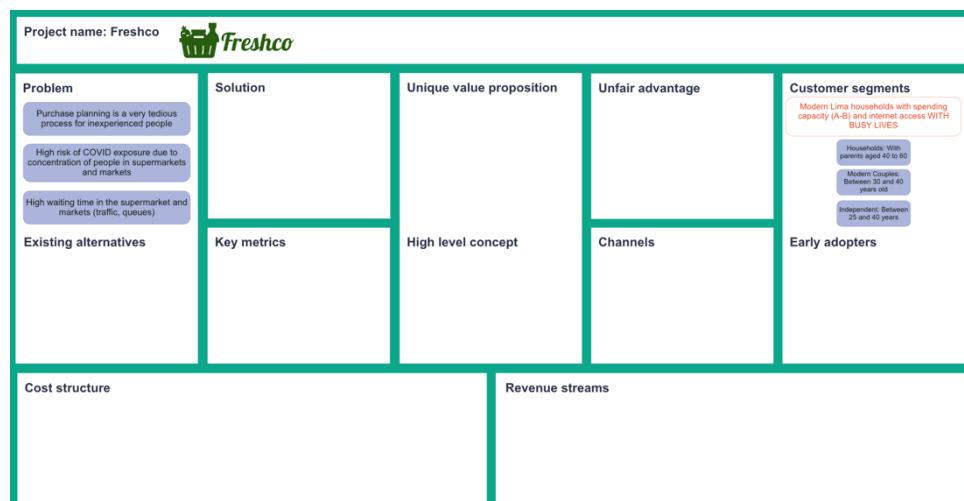
Homeowners: People that oversee managing the operations and finances of the households. They have decision power and execute the purchasing decisions.

Clients: People that complete at least an order and have the intention of making more purchases.

Warehouse operators: People that oversee picking and packing the products for its delivery.

Business operators: People that oversee the business side operations and require valuable information to make decisions.

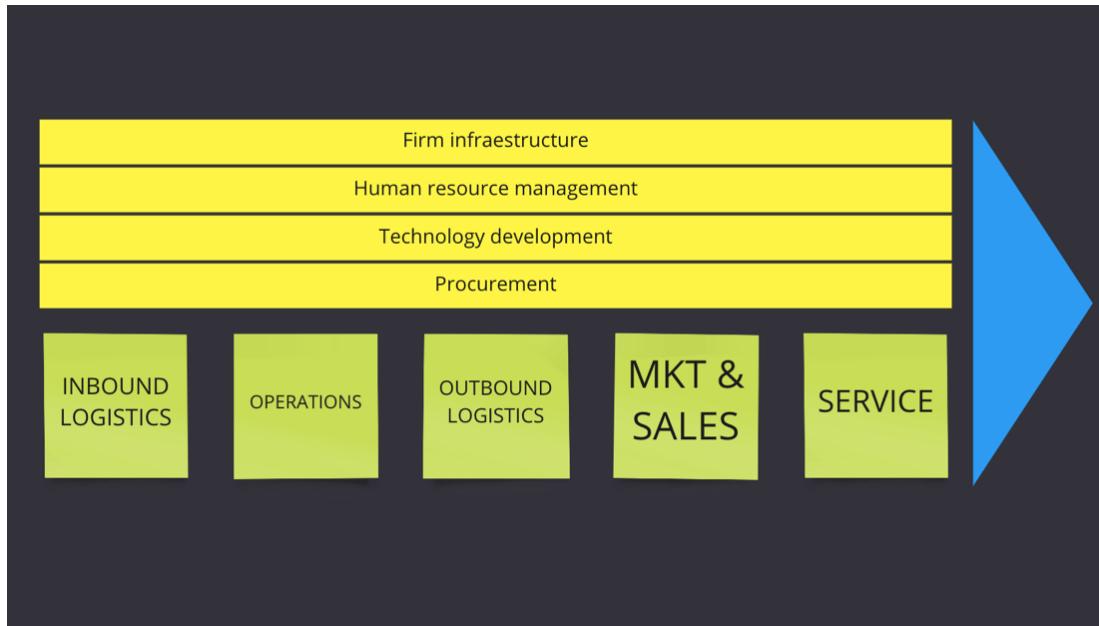
Figure 15: Lean canvas – Customer segments



Identifying the Business processes

The team members identified the processes which the system is supposed to handle, independently from the identified users. First, the team developed the value chain of the company in which the most important processes are presented.

Figure 16: Value chain



After that the team discussed which specifical parts would be handled by the system. The identified business processes are:

- Procurement
- Marketing and Sales
- Operations
- Business intelligence

Identifying functionalities

The members continued to identify the functionalities which the system must have. This was done by associating the identified roles with the identified business processes. This means that the possible involved users can execute their tasks and accomplish their objectives through the required functionalities. Then the team sorted the functionalities under an aggregated category of activities. Then those categories are used as a header for sorting and prioritizing the functionalities. Some of the identified functionalities are the following:

Figure 17: Functionalities



Figure 18: Aggregated functionalities

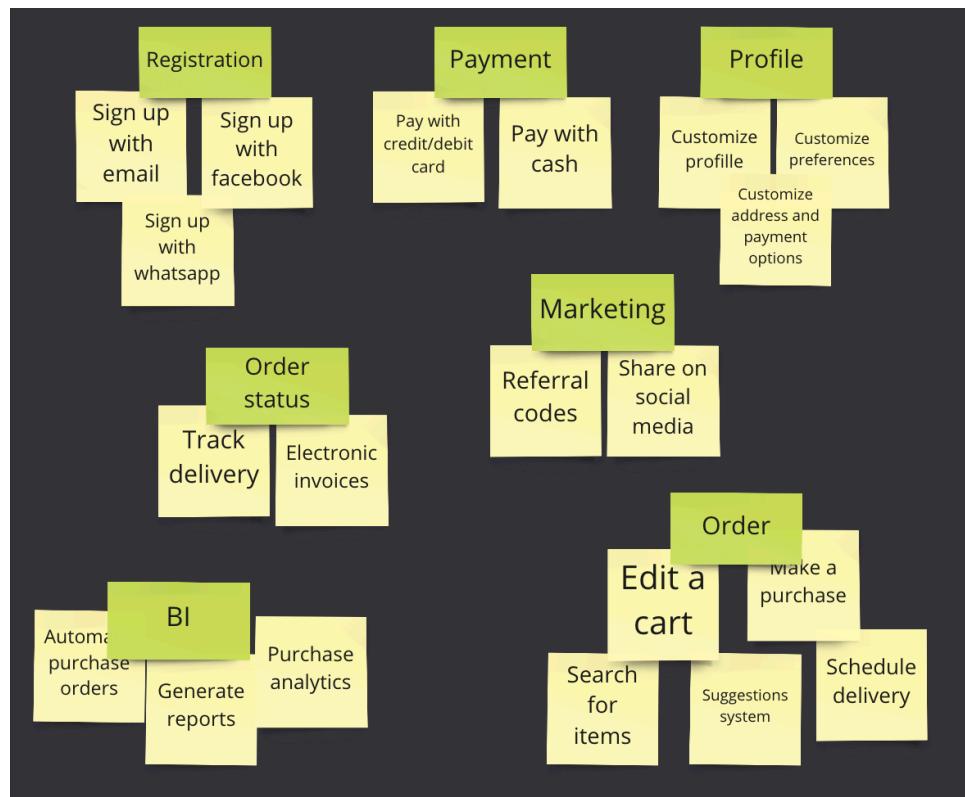
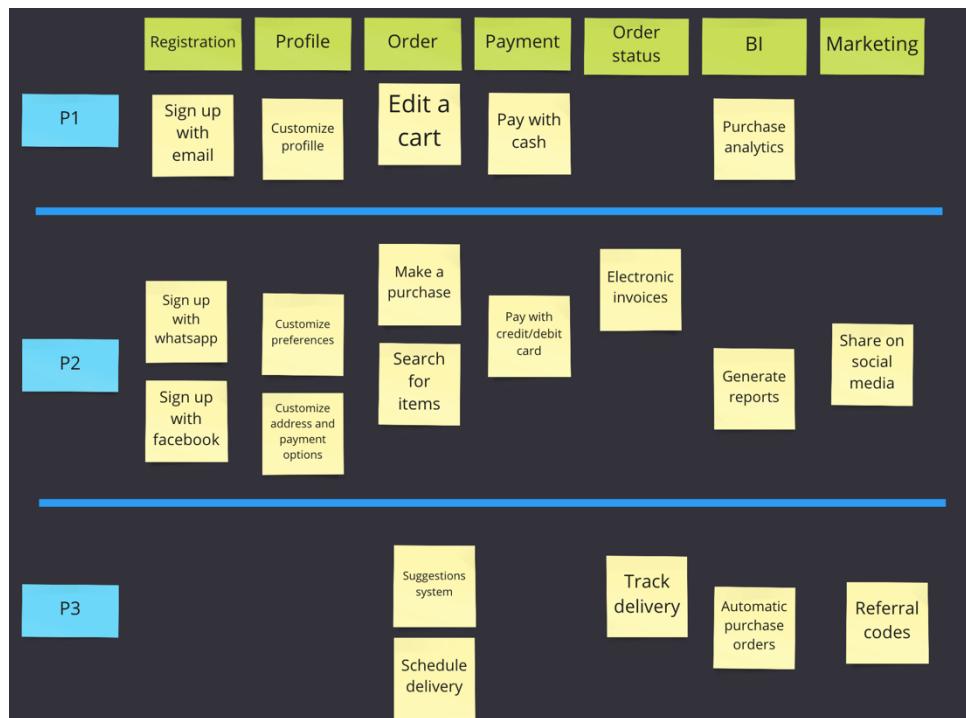


Figure 19: Prioritized functionalities



Product Backlog

The team analysed the before proposed functionalities and transformed them into a list of specific user stories. This list contains everything that is required from the system and is the only source of truth for the development regarding needs and modifications.

Table 1: Product Backlog

ID	User Story		
	As a/an ...	I want ...	so that...
1	Shopper	to quickly understand what the business is offering	I can further explore the page
2	Shopper	to be able to visualize the available products without the need of registering	I can decide if I want to go further
3	Shopper	to sign up to the page with Facebook	I can start shopping
4	Shopper	to sign up to the page with WhatsApp	I can start shopping
5	Shopper	to sign up to the page with my email	I can start shopping
6	Shopper	visualize my profile	I can check my data is correct

7	Shopper	to customize my profile	I can introduce my personal data
8	Shopper	to customize my preferences	I can get personalized offers
9	Shopper	to be able to recover my password	guarantee the safety of my account in case I forget it
10	Shopper	to add a credit/debit card	I can pay
11	Shopper	to add addresses	I can get the goods delivered
12	Shopper	to see what products are offered	I can decide what to buy
13	Shopper	search for a product	I can add it to my cart
14	Shopper	to see the detail of a product	I can decide if I want it
15	Shopper	to add products to a cart	I can purchase them
16	Shopper	edit the products of a cart	I can purchase what I want after changing my mind
17	Shopper	to checkout my cart	I can pay for the goods
18	Shopper	to schedule the delivery date	I can receive the products on a proper time
19	Shopper	to be able to pay with cash	I can use cash to pay when I get the goods
20	Shopper	to be able to pay with credit card	I can pay instantly online
21	Shopper	to verify the data of my purchase	I can be sure that everything on the purchase is correct
22	Shopper	to see a confirmation on screen	I can know that the purchase was successful
23	Client	to login or logout instantly	I do what I need
24	Client	to receive an invoice on my mail	I can check the result of my purchase
25	Client	to receive updates about my purchase	I can keep track of what is going on

26	Client	to track the state of my delivery	I know where it is
27	Client	to cancel the order	I receive my money back
28	Client	to receive suggestions when I make a purchase	I can find more options
29	Client	to receive a premade personalized option of purchase	I can shop with less effort
30	Client	to rate the purchase	I can transmit my opinion
31	Client	to message someone on the company	I can get support
32	Warehouse operator	to receive a notification every time an order is placed	I can start preparing the order
33	Warehouse operator	to receive a detail list of what the items of the order are	I can fulfill the order correctly
35	Warehouse operator	to print the details of the delivery	I can attach them to the package
34	Warehouse operator	to check the stock of the products	I can know if there are enough products
36	Warehouse operator	to confirm that I completed the order	I can do the next operation
37	Warehouse operator	to notify if there is a problem with the order	I can find a solution
38	Business operator	to add products to the database	they can be sold
39	Business operator	to edit the attributes of a product	they can be updated
40	Business operator	to apply discounts	the products are bought
41	Business operator	to run analytics of the purchases	I can make decisions
42	Business operator	to generate customized financial and stock reports	I can check the results
43	Business operator	to have predictions of the purchases	I can make better future decisions

44	Business operator	to get all the users' emails	I can run mail marketing campaigns
45	Business operator	to get an automatic report of the needed items for purchases	I can complete the procurement
46	Business operator	to receive enquiries from clients	I can solve them
47	Business operator	to receive enquiries from warehouse operators	I can solve them

2.1.3. Prioritizing the Product Backlog

After listing all the user stories, the list had to be prioritized. The prioritization helped the team to define which items were more important and were meant to be developed first. The product owner was tasked with prioritizing it and consulted the team to get more insight.

Table 2: Prioritized Product Backlog

ID	User Story			Priority	Risk
	As a/an ...	I want ...	so that...		
1	Shopper	to quickly understand what the business is offering	I can further explore the page	Medium	Medium
2	Shopper	to be able to visualize the available products without the need of registering	I can decide if I want to go further	Medium	Medium
5	Shopper	to sign up to the page with my email	I can start shopping	High	High
6	Shopper	visualize my profile	I can check my data is correct	Medium	Low
7	Shopper	to customize my profile	I can introduce my personal data	High	Medium
9	Shopper	to be able to recover my password	guarantee the safety of my account in case I forget it	High	High
10	Shopper	to add a credit/debit card	I can pay	Medium	High
11	Shopper	to add addresses	I can get the goods delivered	Medium	Medium
12	Shopper	to see what products are offered	I can decide what to buy	Medium	Low

13	Shopper	search for a product	I can add it to my cart	Medium	Low
14	Shopper	to see the detail of a product	I can decide if I want it	Medium	Low
15	Shopper	to add products to a cart	I can purchase them	High	High
16	Shopper	edit the products of a cart	I can purchase what I want after changing my mind	High	High
17	Shopper	to checkout my cart	I can pay for the goods	High	High
19	Shopper	to be able to pay with cash	I can use cash to pay when I get the goods	Medium	High
21	Shopper	to verify the data of my purchase	I can be sure that everything on the purchase is correct	Medium	Medium
22	Shopper	to see a confirmation on screen	I can know that the purchase was successful	Medium	Low
23	Client	to login or logout instantly	I do what I need	Medium	Medium
24	Client	to receive an invoice on my mail	I can check the result of my purchase	High	Medium
3	Shopper	to sign up to the page with Facebook	I can start shopping	Medium	Medium
8	Shopper	to customize my preferences	I can get personalized offers	High	Medium
20	Shopper	to be able to pay with credit card	I can pay instantly online	High	High
25	Client	to receive updates about my purchase	I can keep track of what's going on	Medium	Medium
26	Client	to track the state of my delivery	I know where it is	Medium	Medium
27	Client	to cancel the order	I receive my money back	High	High
4	Shopper	to sign up to the page with WhatsApp	I can start shopping	High	High
18	Shopper	to schedule the delivery date	I can receive the products on a proper time	High	Medium
28	Client	to receive suggestions when I make a purchase	I can find more options	High	High

30	Client	to rate the purchase	I can transmit my opinion	Medium	Low
32	Warehouse operator	to receive a notification every time an order is placed	I can start preparing the order	Medium	Low
33	Warehouse operator	to receive a detail list of what the items of the order are	I can fulfill the order correctly	Medium	Medium
35	Warehouse operator	to print the details of the delivery	I can attach them to the package	High	Medium
36	Warehouse operator	to confirm that I completed the order	I can do the next operation	High	Medium
44	Business operator	to get all the users' emails	I can run mail marketing campaigns	Medium	High
45	Business operator	to get an automatic report of the needed items for purchases	I can complete the procurement	Medium	High
29	Client	to receive a premade personalized option of purchase	I can shop with less effort	High	High
31	Client	to message someone on the company	I can get support	High	Medium
34	Warehouse operator	to check the stock of the products	I can know if there are enough products	Medium	Medium
37	Warehouse operator	to notify if there is a problem with the order	I can find a solution	Medium	Medium
38	Business operator	to add products to the database	they can be sold	High	High
39	Business operator	to edit the attributes of a product	they can be updated	High	High
40	Business operator	to apply discounts	the products are bought	Medium	Medium
41	Business operator	to run analytics of the purchases	I can make decisions	Medium	Medium
42	Business operator	to generate customized financial and stock reports	I can check the results	Medium	High
43	Business operator	to have predictions of the purchases	I can make better future decisions	High	High
46	Business operator	to receive enquiries from clients	I can solve them	Medium	Medium

47	Business operator	to receive enquiries from warehouse operators	I can solve them	Medium	Medium
----	-------------------	---	------------------	--------	--------

2.2. Minimum Viable Product release

Since there are a lot of identified user stories that can either be or not be useful for the final version of the system, the team had to set time boundaries and identified which user stories apparently are the most important and should be included on the MVP release.

Table 3: Release MVP Backlog

ID	User Story			Priority	Risk	Estimation
	As a/an ...	I want ...	so that...			
MVP						
1	Shopper	to quickly understand what the business is offering	I can further explore the page	Medium	Medium	20
2	Shopper	to be able to visualize the available products without the need of registering	I can decide if I want to go further	Medium	Medium	20
5	Shopper	to sign up to the page with my email	I can start shopping	High	High	8
6	Shopper	visualize my profile	I can check my data is correct	Medium	Low	8
7	Shopper	to customize my profile	I can introduce my personal data	High	Medium	13
9	Shopper	to be able to recover my password	I can guarantee the safety of my account in case I forget it	High	High	8
10	Shopper	to add a credit/debit card	I can pay	Medium	High	5
11	Shopper	to add addresses	I can get the goods delivered	Medium	Medium	5
12	Shopper	to see what products are offered	I can decide what to buy	Medium	Low	5
13	Shopper	search for a product	I can add it to my cart	Medium	Low	8

14	Shopper	to see the detail of a product	I can decide if I want it	Medium	Low	13
15	Shopper	to add products to a cart	I can purchase them	High	High	8
16	Shopper	edit the products of a cart	I can purchase what I want after changing my mind	High	High	13
17	Shopper	to checkout my cart	I can pay for the goods	High	High	8
19	Shopper	to be able to pay with cash	I can use cash to pay when I get the goods	Medium	High	5
21	Shopper	to verify the data of my purchase	I can be sure that everything on the purchase is correct	Medium	Medium	5
22	Shopper	to see a confirmation on screen	I can know that the purchase was successful	Medium	Low	3
23	Client	to login or logout instantly	I do what I need	Medium	Medium	5
24	Client	to receive an invoice on my mail	I can check the result of my purchase	High	Medium	5
3	Shopper	to sign up to the page with Facebook	I can start shopping	Medium	Medium	8
8	Shopper	to customize my preferences	I can get personalized offers	High	Medium	13
20	Shopper	to be able to pay with credit card	I can pay instantly online	High	High	8
25	Client	to receive updates about my purchase	I can keep track of what's going on	Medium	Medium	5
26	Client	to track the state of my delivery	I know where it is	Medium	Medium	5

27	Client	to cancel the order	I receive my money back	High	High	5
28	Client	to receive suggestions when I make a purchase	I can find more options	High	High	20
30	Client	to rate the purchase	I can transmit my opinion	Medium	Low	3
32	Warehouse operator	to receive a notification every time an order is placed	I can start preparing the order	Medium	Low	3
35	Warehouse operator	to print the details of the delivery	I can attach them to the package	High	Medium	3
36	Warehouse operator	to confirm that I completed the order	I can do the next operation	High	Medium	3
38	Business operator	to add products to the database	they can be sold	High	High	5
39	Business operator	to edit the attributes of a product	they can be updated	High	High	8
45	Business operator	to get an automatic report of the needed items for purchases	I can complete the procurement	Medium	High	13

2.2.1. Sprint 1

This Sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the print. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 4: Sprint 1 Backlog

ID	User Story			Estimation	Acceptance criteria
	As a/an ...	I want ...	so that...		
Sprint 1					
1	Shopper	to quickly understand what the business is offers	I can further explore the page	20	Have an aesthetic landing page Explain the shopping process in the landing page Have a good UI on

					the landing page Add some contact details on the landing page
2	Shopper	to be able to visualize the available products without the need of registering	I can decide if I want to go further	20	Show some available products Show the product prices Show the product image Show if the product is on stock
5	Shopper	to sign up to the page with my email	I can start shopping	8	Show the sign-up screen Show mail and password spaces User can successfully sign up

Table 5: Sprint 1 Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Design initial use cases	x					
Design initial data structure	x					
Design Wireframes	x					
Design class structure	x					
Create a repository	x					
Design initial database	x					
Create initial database	x					
Create initial back-end skeleton	x					
Create initial CRUD	x					
Create landing page	x					
Insert products into DB	x					
Create products back-end	x					
Create products page view	x					
Create DB triggers	x					

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the Sprint.

System Design (UML)

Figure 20: Initial used case diagram

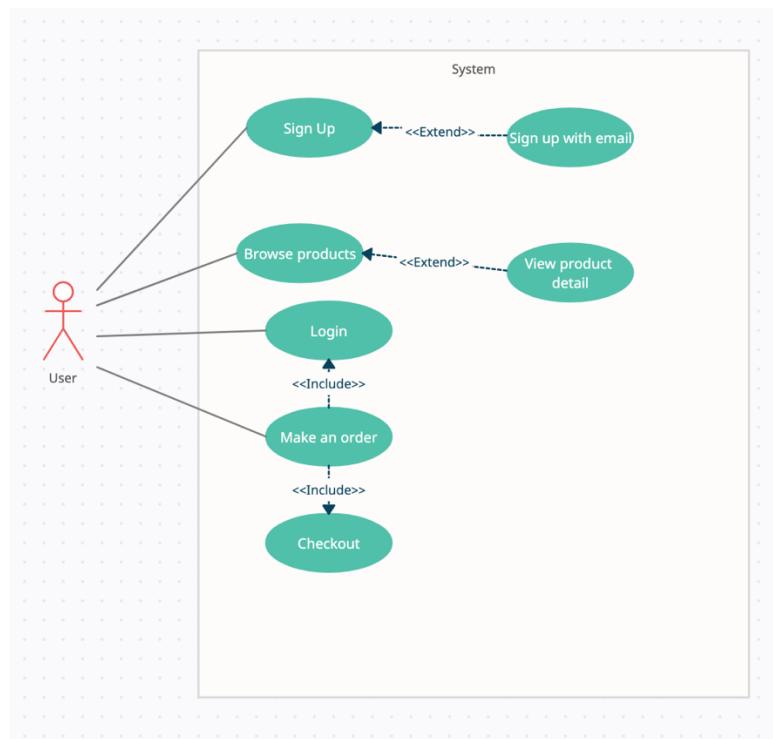
UI/UX Design

Figure 21: Landing page wireframe

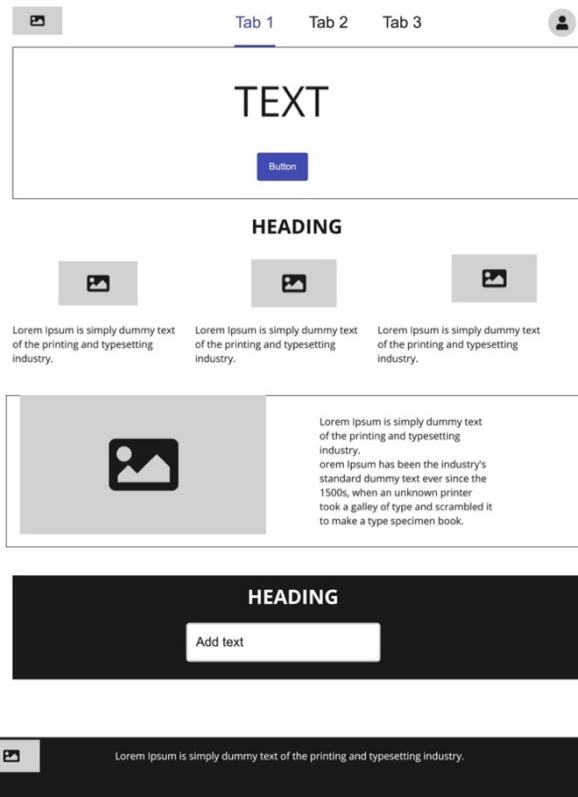


Figure 22: Products wireframe

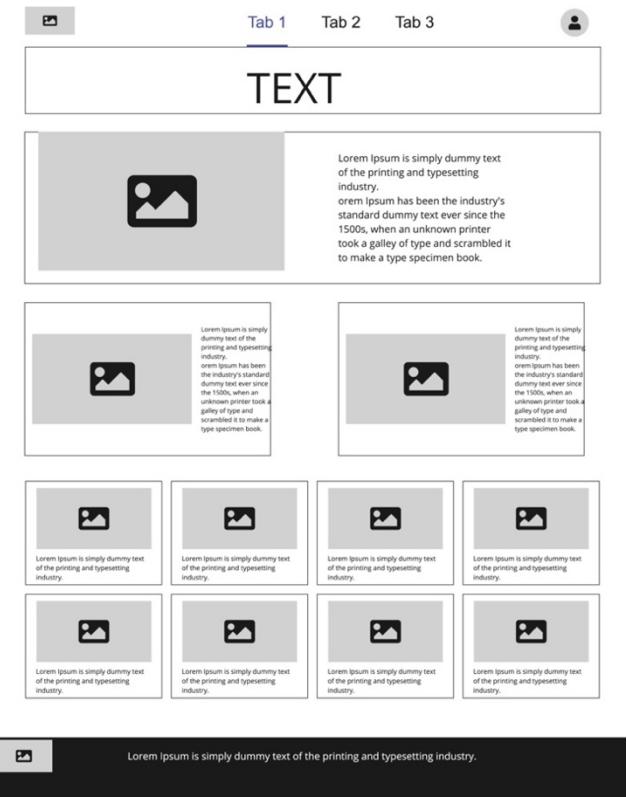
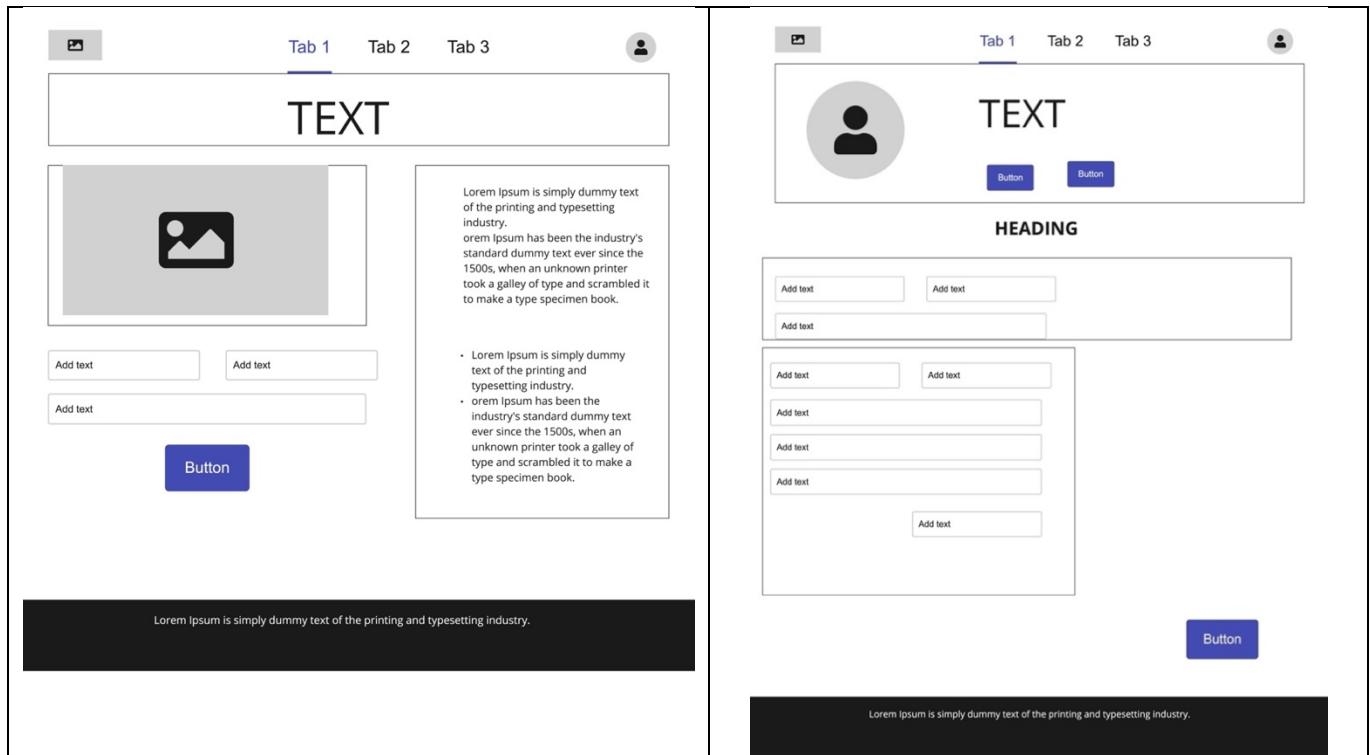


Figure 23: Checkout wireframe

Figure 24: Profile wireframe



Development environment

HTML5

HTML is the acronym for Hyper Text Markup Language which refers to the hypertext markup language, a standard language in charge of the World Wide Consortium or W3C that is used when creating web pages. The latest version that exists, and the one that will be used in the development of the application, is HTML5, implementing the semantic web paradigm by introducing new tags, which themselves offer information about the content that they are going to show. (HTML Tutorial, 2021)

CSS3

Cascading Style Sheets (or CSS) are cascading style sheets which refer to the language that describes the presentation of documents written in HTML. CSS is designed mainly to separate the content of the document from the way it is displayed, that is, it is used to shape the graphical interfaces of web pages. (CSS Tutorial, 2021)

JavaScript

JavaScript is a client-side programming language, although it also has server-side functions due to technologies like NodeJS. It is interpreted (not compiled), object-oriented, weakly typed and allows to give dynamism to web pages, thus improving the user interface. The way to use JavaScript together with the HTML files of the web pages is like that seen previously with the CSS files, that is, the JavaScript files will be created separately and must be referenced from the corresponding HTML files. (JavaScript Tutorial, 2021)

Python

It is an interpreted, non-compiled programming language that uses dynamic, strongly typed, cross-platform, and multi-paradigm typing. We can find it running on servers, in iOS, Android, Linux, Windows or Mac applications. It is the language that is normally chosen to develop Big Data and Artificial Intelligence (AI) applications, and since these practices are currently on the rise, we wanted to add this versatile language to our project. (Python For Beginners, 2021)

MySQL

MySQL is an open-source relational database management system (RDBMS) with a client-server model. RDBMS is a software or service used to create and manage databases based on a relational model. (Hostinger Tutorials, 2021)

Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open-source. (The Web framework for perfectionists with deadlines | Django, 2021)

Amazon Web Services

Amazon Web Services is a provider of cloud services, it allows us to have storage, computing resources, mobile applications, databases, and a long etcetera in cloud computing mode. (AWS, 2021)

Git

Git is a version control software designed by Linus Torvalds, with the efficiency and reliability of application versioning in mind when they have a large deal of source code file. With it we can perform all the tasks that correspond to a version administration and management software: (Git, 2021)

- Create and clone repositories. These repositories can be stored locally or remotely.
- Perform the basic actions: commit, push, pull and merge of our files.
- Detect and resolve conflicts.
- Consult the history of the changes that have been made in our repositories.

Gitlab

Gitlab is a Git-based collaborative software development and version control web service. In addition to a repository manager, the service also offers wiki hosting and a bug tracking system, all of which are published under an open-source license. (Gitlab, 2021)

Sprint Review

On the sprint review meeting the compliance of the user stories and the deliverables are verified by the Product Owner and other stakeholders.

Table 6: Sprint I review

User Story	Delivered?	Acceptance criteria	Accepted?
1	NO	Have an aesthetic landing page Explain the shopping process in the landing page Have a good UI on the landing page Add some contact details on the landing page	NO
2	NO	Show some available products Show the product prices Show the product image Show if the product is on stock	NO

5	NO	Show the sign-up screen spaces Show mail and password User can successfully sign up	NO
---	----	---	----

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint did not follow a good rhythm. The accomplishment of the story points was below the average expected velocity. As a result, none of the user stories was successfully delivered. During the first half, the team spent a lot of time trying to decide which was the best path to follow. The main discussion was about which back-end framework was more appropriate and whether it was more convenient to try a microservice architecture from the beginning. Provided that the nature of the system is meant at a long term to evolve to a microservice architecture, the team got stuck discussing and designing what the MVP architecture could be on both monolithic and microservices architecture. On parallel there was some notorious advancement on the design part with the basic design on wireframes. In the end, any effort related to microservices architecture was discarded and the team decided to go safe with monolithic approach. Furthermore, the team also had some conflict with which programming language and framework was better suited for the system. While some of the members were more comfortable with Java, or PHP, the problem remained on how most of the members were familiarized with Python and did not find optimal to catch up with other programming languages.

Figure 25: Sprint 1 Burndown Chart

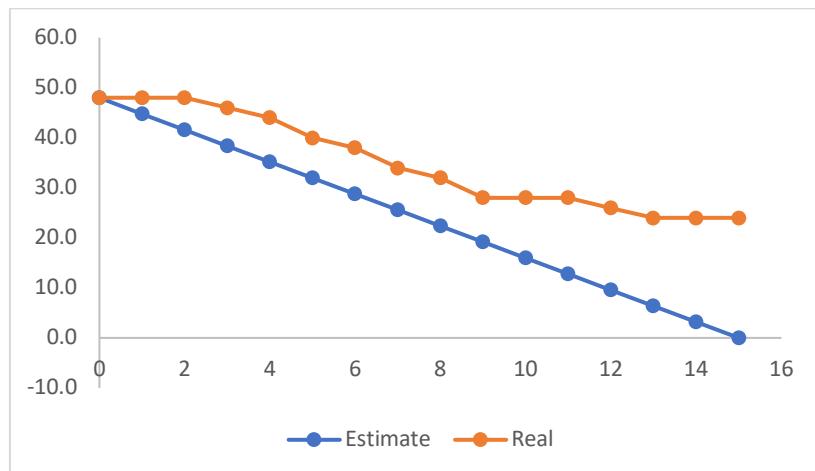


Table 7: Sprint 1 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Design initial use cases						x
Design initial data structure						x
Design Wireframes						x
Design class structure						x
Create a repository						x
Design initial database						x
Create initial database			x			
Create initial back-end skeleton			x			
Create initial CRUD		x				
Create landing page	x					

Insert products into DB	x						
Create products backend	x						
Create products page initial view	x						
Create DB triggers	x						

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

Table 8: Sprint 1 Retrospective

What happened?		Future actions
What went well?	Discussion helped to get a middle ground Team managed to learn more about the tools they are using	
What did not go well?	Coordinating who is doing what Could not complete the Sprint objectives	Define who can contribute beyond their titles Complete what is missing on next Sprint
What puzzles us?	Team members have different opinions for the architecture Business unit is behind schedule verifying business model	

2.2.2. Sprint 2

This sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the Sprint. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 9: Sprint 2 Backlog

ID	User Story			Estimation	Acceptance criteria
	As a/an ...	I want ...	so that...		
Sprint 2					
1	Shopper	to quickly understand what the business is offers	I can further explore the page	6	Have an aesthetic landing page Explain the shopping process in the landing page Have a good UI on the landing page Add some contact details on the landing page

					Show some available products Show the product prices Show the product image Show if the product is on stock
2	Shopper	to be able to visualize the available products without the need of registering	I can decide if I want to go further	6	
5	Shopper	to sign up to the page with my email	I can start shopping	8	Show the sign-up screen Show mail and password spaces User can successfully sign up
6	Shopper	visualize my profile	I can check my data is correct	8	Show the profile page Show all the fields Verify the data displayed matches the user data
7	Shopper	to customize my profile	I can introduce my personal data	13	Show an option to edit the profile Have available fields to edit the profile Have some only non-key fields available for edition Verify the edited fields are correctly saved

Table 10: Sprint 2 Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Create initial database			x			
Create initial backend skeleton			x			
Create initial CRUD		x				
Create landing page	x					
Insert products into DB	x					
Create products backend	x					
Create products page initial view	x					
Create DB triggers	x					
Design mockup for landing page	x					
Design mockups for product pages	x					
Design mockup for customization	x					
Create the profile page backend	x					
Create the profile page initial view	x					

Connect DB with profile page	x					
------------------------------	---	--	--	--	--	--

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the Sprint.

Team assignations

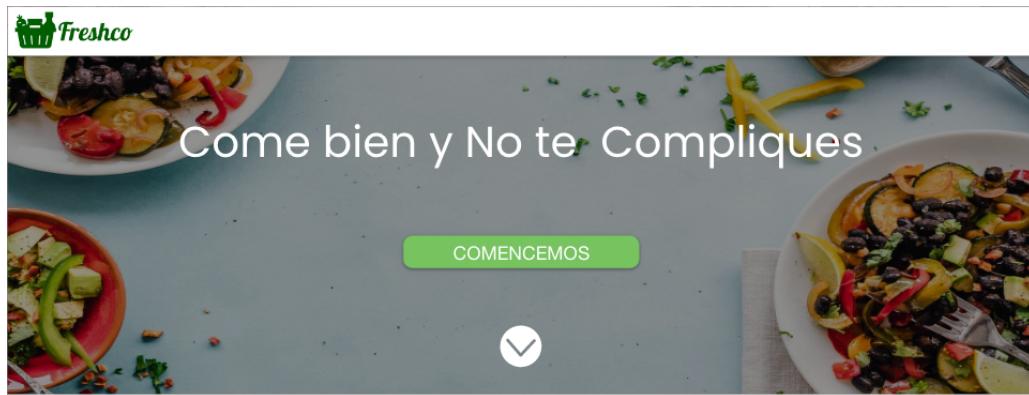
As above mentioned, the team decided that despite that having specific job titles or qualifications, the team would contribute on the main needed areas.

Table 11: Dev team assigned hours

Team	Daily work	Days/week	Hrs/week	Week/Sprint	Hrs/Sprint	Dedicated time		
						Back-end	50%	30
Carles Almendez	4	5	20	3	60	Architecture	30%	18
						Infrastructure Management	20%	12
						Design UX	30%	36
Cami Xiom Fernan	8	5	40	3	120	Design UI	30%	36
						Front-end	40%	48
						Backend	60%	72
Jose Elchaman	8	5	40	3	120	Architecture	30%	36
						Front-end	10%	12
						Database	50%	60
Total	28	20	140	12		Data Science	50%	60
								420

UI/UX Design

Figure 26: Landing page mockup



Por qué Freshco ?



No la Pienses

Recogemos tus preferencias
y te damos increíbles ideas



No pierdas Tiempo

Realizamos las compras por
ti y te las llevamos a casa



No te expongas

Cumplimos con todos los
protocolos de bioseguridad



Figure 27: Login mockup

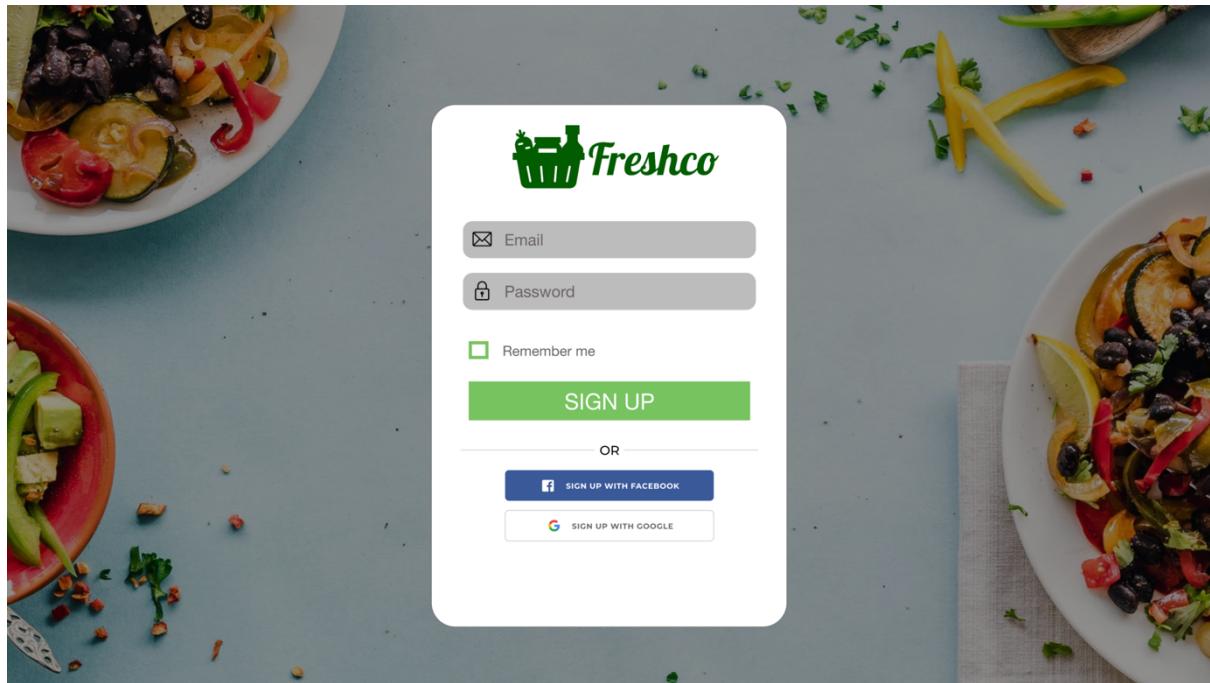


Figure 28: Customization mockup



Figure 29: Products mockup

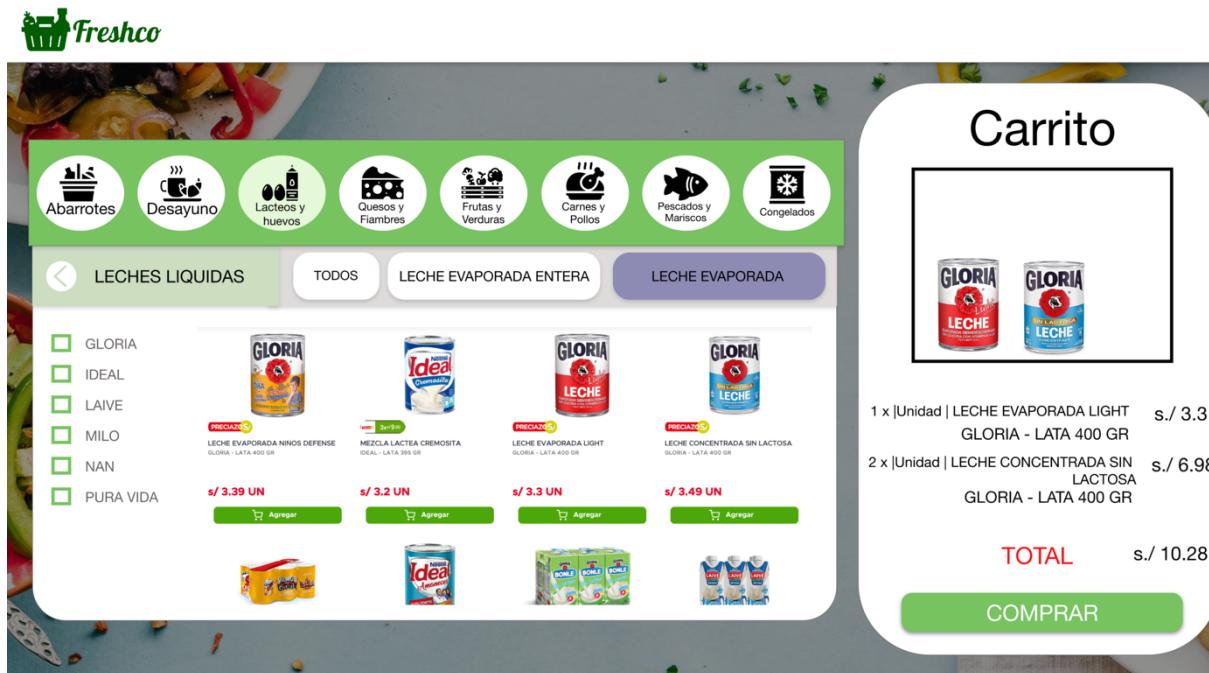


Figure 30: Product detail mockup



Figure 31: Checkout mockup

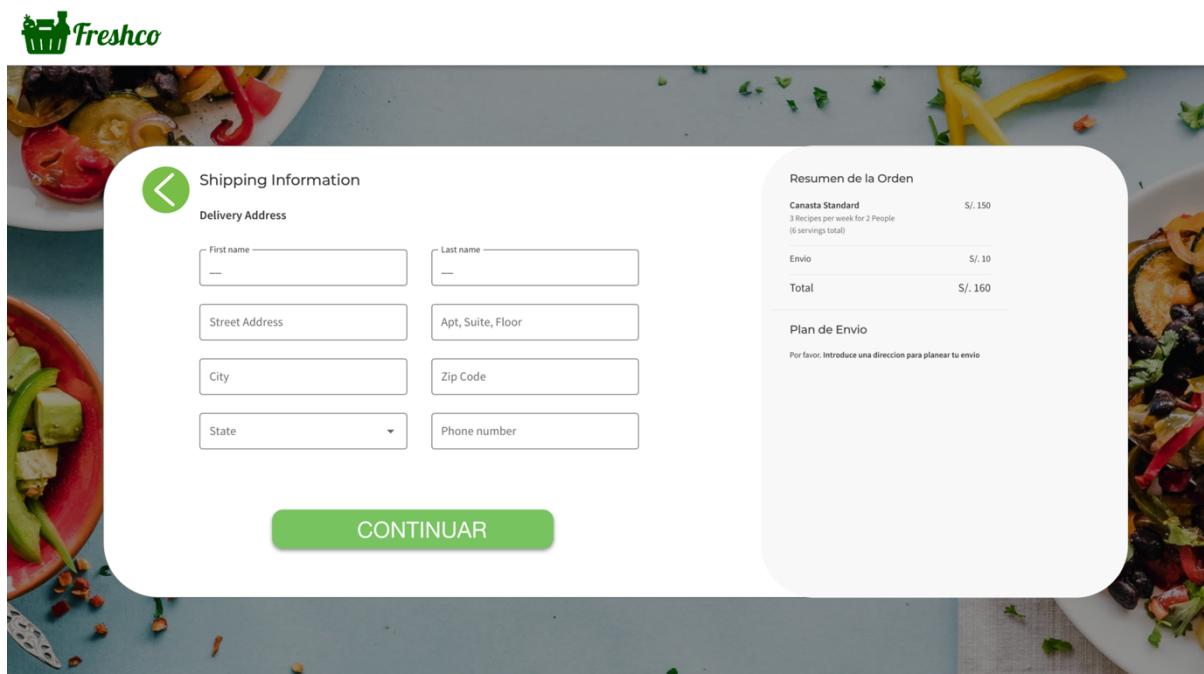
Testing

Table 12: Sprint 2 testing

Test scenario	User story	Test cases	Expected result
Verify the landing page	1	Click all buttons Try different browsers	Everything works fine
Verify the products page	2	Click all products Show multiple products in page	All products can be seen and selected

		Load all product details Switch on different views	
Verify the sign-up page	5	Register with different mails Try fake emails Try complicated passwords	Successful sign up
Verify the profile page	6	Visit page with different accounts Click all the buttons	Everything works fine
Verify changing data in profile page	7	Try all the inputs Enter different values Try entering different kind of values Enter incorrect data	Data can be successfully edited

Sprint Review

On the Sprint review meeting the compliance of the user stories and the deliverables are verified by the Product Owner and other stakeholders.

Table 13: Sprint 2 review

User Story	Delivered?	Acceptance criteria	Accepted?
1	YES	Have an aesthetic landing page Explain the shopping process in the landing page Have a good UI on the landing page Add some contact details on the landing page	YES. Barebones remains to be upgraded
2	NO	Show some available products Show the product prices Show the product image Show if the product is on stock	YES Barebones remains to be upgraded
5	YES	Show the sign-up screen Show mail and password spaces User can successfully sign up	YES Barebones remains to be upgraded
6	NO	Show the profile page Show all the fields Verify the data displayed matches the user data	NO
7	NO	Show an option to edit the profile Have available fields to edit the profile Have some only non-key fields available for edition Verify the edited fields are correctly saved	NO

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint followed a better rhythm. The accomplishment of the story points was below the average expected velocity on a better rate. There remained some story points left on the Sprint. Regarding the design for the front-end, it is getting behind schedule

because the business unit still has not gotten feedback and the designer is working without confirmation. There remains to define what will be the brand assets. Beyond that, the implementation of the latest feature remained on development.

Figure 32: Sprint 2 Burndown Chart

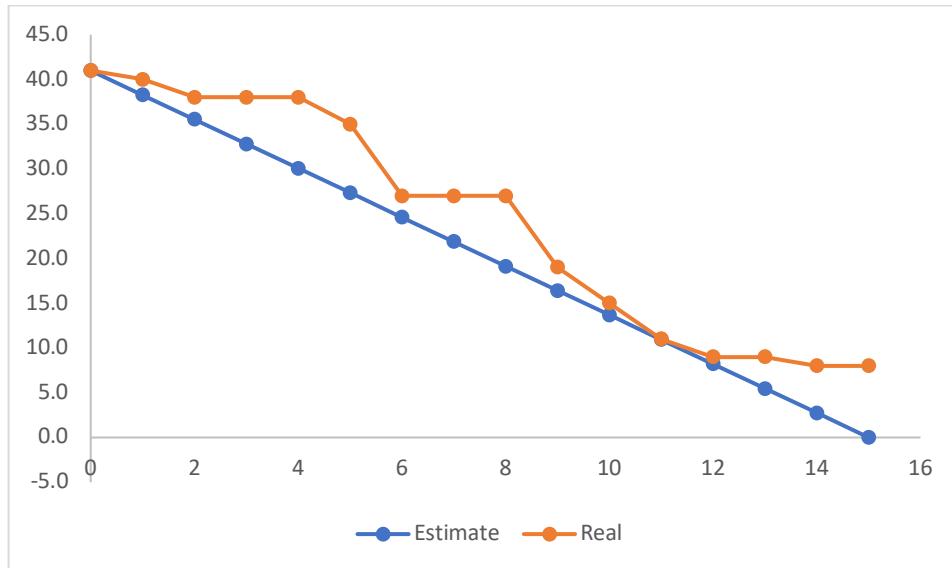


Table 14: Sprint 2 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Create initial database						x
Create initial backend skeleton						x
Create initial CRUD						x
Create landing page						x
Insert products into DB						x
Create products backend						x
Create products page initial view						x
Create DB triggers						x
Design mockup for landing page						x
Design mockup for product page						x
Design mockup for profile page						x
Create the profile page backend				x		
Create the profile page initial view			x			
Connect DB with profile page			x			

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

Table 15: Sprint 2 Retrospective

What happened?	Future actions
----------------	----------------

What went well?	Team managed to complete previous sprint goals	
	Team decided to all cooperate on backend	
What did not go well?	Underestimated the difficulty of completing an additional feature	Compromise to less story points for next sprints
	No confirmation with tested users	Go on with what remains
What puzzles us?	Design tasks are getting behind schedule	

2.2.3. Sprint 3

This Sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the Sprint. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 16: Sprint 3 Backlog

ID	User Story			Estimation	Acceptance criteria
	As a/an ...	I want ...	so that...		
Sprint 3					
6	Shopper	visualize my profile	I can check my data is correct	2	Show the profile page Show all the fields Verify the data displayed matches the user data
7	Shopper	to customize my profile	I can introduce my personal data	6	Show an option to edit the profile Have available fields to edit the profile Have some only non-key fields available for edition Verify the edited fields are correctly saved
9	Shopper	to be able to recover my password	guarantee the safety of my account in case I forget it	8	Have option to request account recovery by mail Have an available link to recover password on mail Successfully introduce and verify new password Redirect to login page

10	Shopper	to add a credit/debit card	I can pay	8	Have the option to add a credit card on the profile page Successfully integrate the Culqui API Deduct 1 PEN from the credit card to verify payment
11	Shopper	to add addresses	I can get the goods delivered	5	Have the option to add an address to the profile Successfully integrate the google geocoding API Successfully identify the client house on google maps
3	Shopper	to sign up to the page with Facebook	I can start shopping	8	Have the option to sign up by clicking Facebook button Integrate the Facebook login for the web

Table 17: Sprint 3 Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Create the profile page backend				x		
Create the profile page initial view			x			
Connect DB with profile page			x			
Setup mail server	x					
Implement recovery by mail	x					
Implement card fields	x					
Implement integration with Culqui	x					
Implement payment page	x					
Implement address book	x					
Implement Google geocoding API	x					
Implement address page	x					
Implement Facebook sign up API	x					
Implement Facebook button	x					
Integrate DB triggers for user	x					

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the Sprint.

UI/UX Design

Figure 33: Landing page



Figure 34: Sign up page

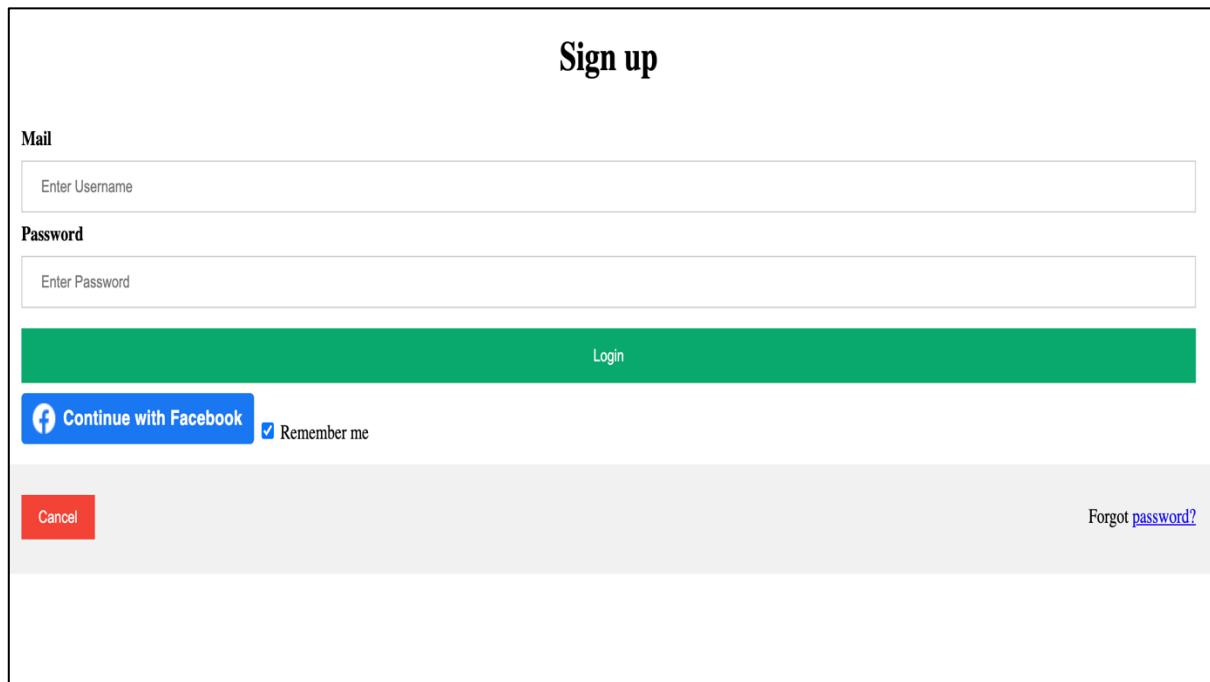
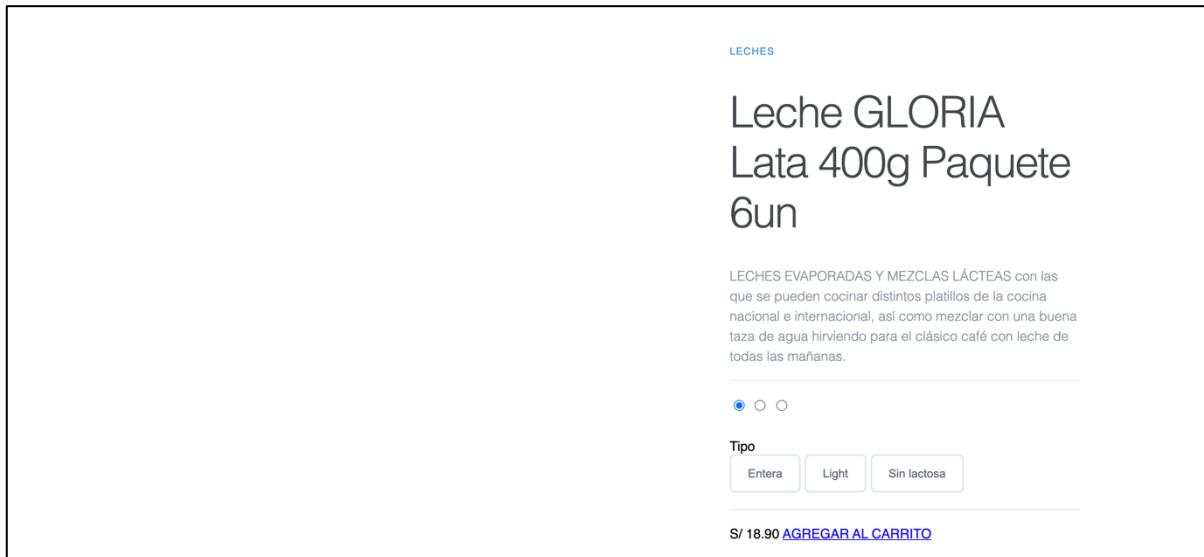


Figure 35: Product view page



Testing

Table 18: Sprint 3 testing

Test scenario	User story	Test cases	Expected result
Verify the profile page	6	Visit page with different accounts Click all the buttons	Everything works fine
Verify changing data in profile page	7	Try all the inputs Enter different values Try entering different kind of values Enter incorrect data	Data can be successfully edited
Verify the password recovery	9	Try recovery with different accounts Try using recovery links with wrong user Timeout the recovery page	Successful recover password
Verify the cards can be added	10	Try all the inputs Enter different values Try entering different kind of values Enter incorrect data	Card is successfully added
Verify the addresses can be added	11	Try all the inputs Enter different values Try entering different kind of values Enter incorrect data	Address is successfully added
Verify sign up with Facebook works	3	Try different Facebook users Timeout sign up page Enter wrong data Try with a registered user	Successfully registered with Facebook

Sprint Review

Table 19: Sprint 3 review

User Story	Delivered?	Acceptance criteria	Accepted?
6	YES	Show the profile page Show all the fields Verify the data displayed matches the user data	YES Barebones remains to be upgraded
7	YES	Show an option to edit the profile Have available fields to edit the profile Have some only non-key fields available for edition Verify the edited fields are correctly saved	YES Barebones remains to be upgraded
9	YES	Have option to request account recovery by mail Have an available link to recover password on mail Successfully introduce and verify new password Redirect to login page	YES Barebones remains to be upgraded
10	YES	Have the option to add a credit card on the profile page Successfully integrate the Culqui API Deduct 1 PEN from the credit card to verify payment	YES Barebones remains to be upgraded
11	YES	Have the option to add an address to the profile Successfully integrate the google geocoding API Successfully identify the client house on google maps	YES Barebones remains to be upgraded
3	YES	Have the option to sign up by clicking Facebook button Integrate the Facebook login for the web	YES Barebones remains to be upgraded

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint followed the expected rhythm. The accomplishment of the story points was below near the average expected velocity. At some point it was ahead of schedule and worked for further testing. The design of the front-end of the pages got to a functional barebones stage but remains to get ahead with the visuals. As for the backend, triggers and database, the team has managed to catch up by having already a stable skeleton. The integration of Culqui, Google geocoding and Facebook were successful and got developed on schedule.

Figure 36: Sprint 3 Burndown Chart

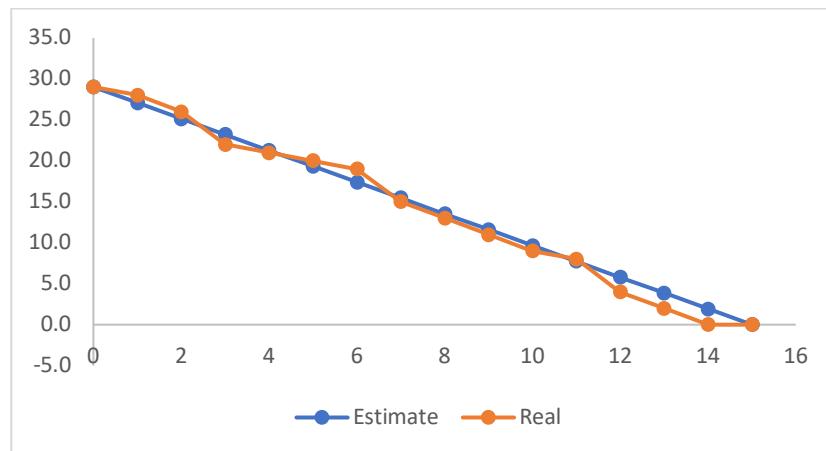


Table 20: Sprint 3 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Create the profile page backend						x
Create the profile page initial view						x
Connect DB with profile page						x
Setup mail server						x
Implement recovery by mail						x
Implement card fields						x
Implement integration with Culqui						x
Implement payment page						x
Implement address book						x
Implement Google geocoding API						x
Implement address page						x
Implement Facebook sign up API						x
Implement Facebook button						x
Integrate DB triggers for user						x

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

Table 21: Sprint 3 Retrospective

What happened?	Future actions
What went well?	Team managed to complete all sprint goals Team took less story points for this sprint
What did not go well?	Design is getting too ambitious with the frontend Develop a more basic version with less JavaScript complexity

What puzzles us?	The issue of microservices still arises for long term	
	Business unit is getting some insights to verify the system but still cannot be tested by clients	

Mid-Project Analysis

Having completed already 3 Sprints, half of the project has been completed. At the beginning of the project there were some difficulties related to the inception of the project. There remain issues with dependencies related to design and the business unit. So far, the business unit has relied on testing the business model with a non-technical solution. The system has gotten to the point in which it has barebones functionality but still needs a more refined front-end. The business unit already managed to get more insights about the market and the ideal solution. For example, the value of the industry in terms of sale of food online goes as high as 25 million PEN. The addressable market size for the city of Lima is about 10.91 million PEN. The team identified 5 customer profiles which are related to different kinds of households in Lima, and each has its own characteristics. Also, 2 kinds of decision profiles were identified: people who want to decide what to cook and get the shops made and people who shop standard items and later need cooking recommendations.

Figure 37: Lean canvas completed

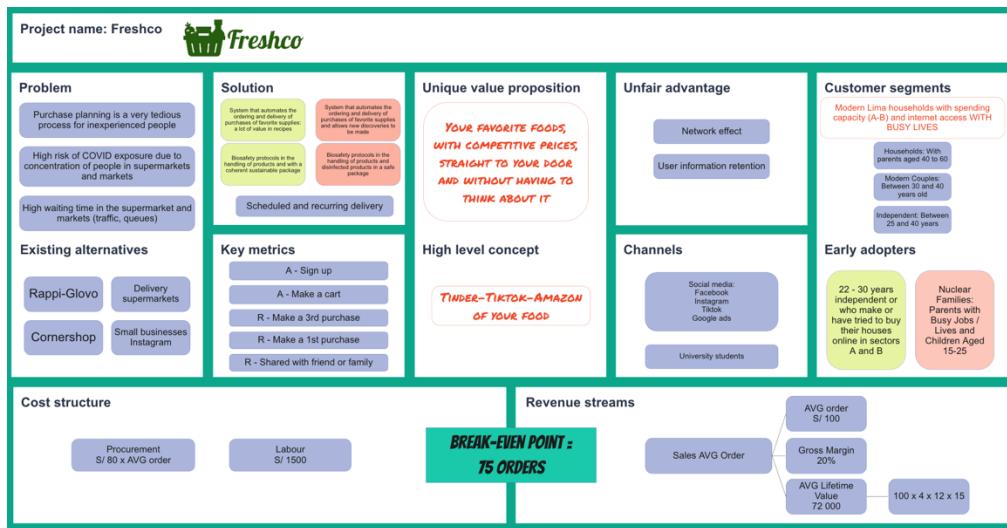


Figure 38: Industry value

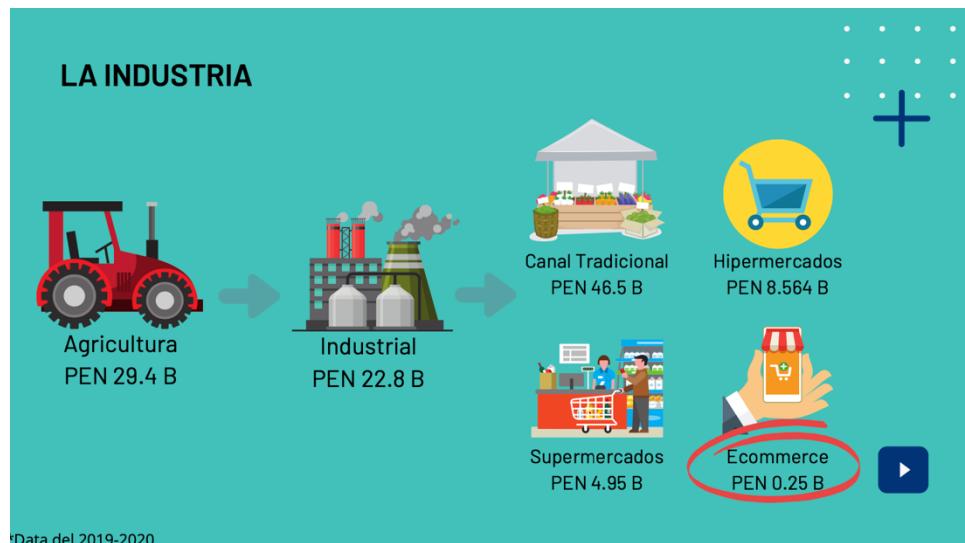


Figure 39: Market size



Figure 40: Customer profiles

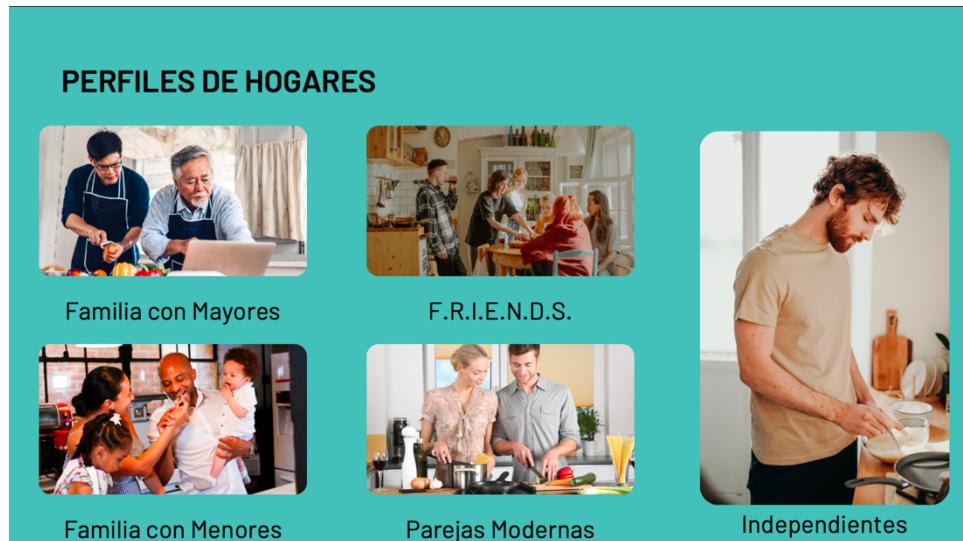
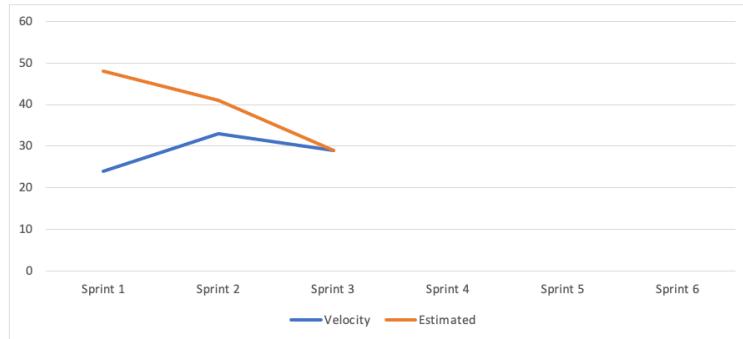


Figure 41: Custom decision profiles



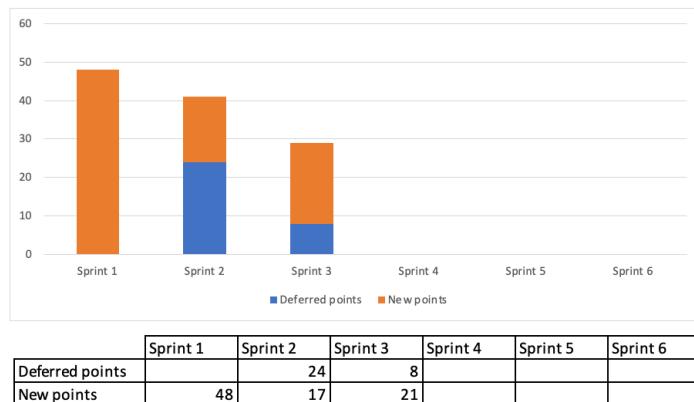
By analysing the performance of the team some findings were also made. First, at the beginning the real velocity was considerably lower than was expected, but by now it has stabilized. Also, since the Sprints goals were not met at the beginning, some tasks were deferred to the next Sprint and slowed down the general progress.

Figure 42: Expected vs Real velocity



	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6
Velocity	24	33	29			
Estimated	48	41	29			

Figure 43: Deferred points vs New points



	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6
Deferred points		24	8			
New points	48	17	21			

2.2.4. Sprint 4

This Sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the Sprint. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 22: Sprint 4 Backlog

ID	User Story			Estimation	Acceptance criteria
	As a/an ...	I want ...	so that...		
Sprint 4					
13	Shopper	search for a product	I can add it to my cart	8	Product can be searched on a browser Non existing products show similar products Queries are direct

					coincidences Brands shorten the searches
15	Shopper	to add products to a cart	I can purchase them	8	Products are added to cart by actioning buttons Several units can be added at the same time
16	Shopper	edit the products of a cart	I can purchase what I want after changing my mind	8	Products in cart can be deleted Units of products can be edited
17	Shopper	to checkout my cart	I can pay for the goods	8	The temporal cart is registered and triggers payment page Cart issues a unique ID
19	Shopper	to be able to pay with cash	I can use cash to pay when I get the goods	5	Button allows to pay by cash Issue unique ID of transaction
22	Shopper	to see a confirmation on screen	I can know that the purchase was successful	3	Issue unique order ID Register order to be fulfilled Lower stock with products Show in screen confirmation

Table 23: Sprint 4 Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement backend and queries for search	x					
Implement cart's back-end	x					
Implement cart's view	x					
Implement temporal state cart	x					
Implements checkout back-end	x					

Implement checkout view	x					
Implement cash payment triggers	x					
Implement card payment triggers	x					
Implement payment view	x					
Implement order registration	x					
Implement confirmation page	x					
Implement confirmation view	x					

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the sprint.

System Design (UML)

Figure 44: Use case diagram sprint 4

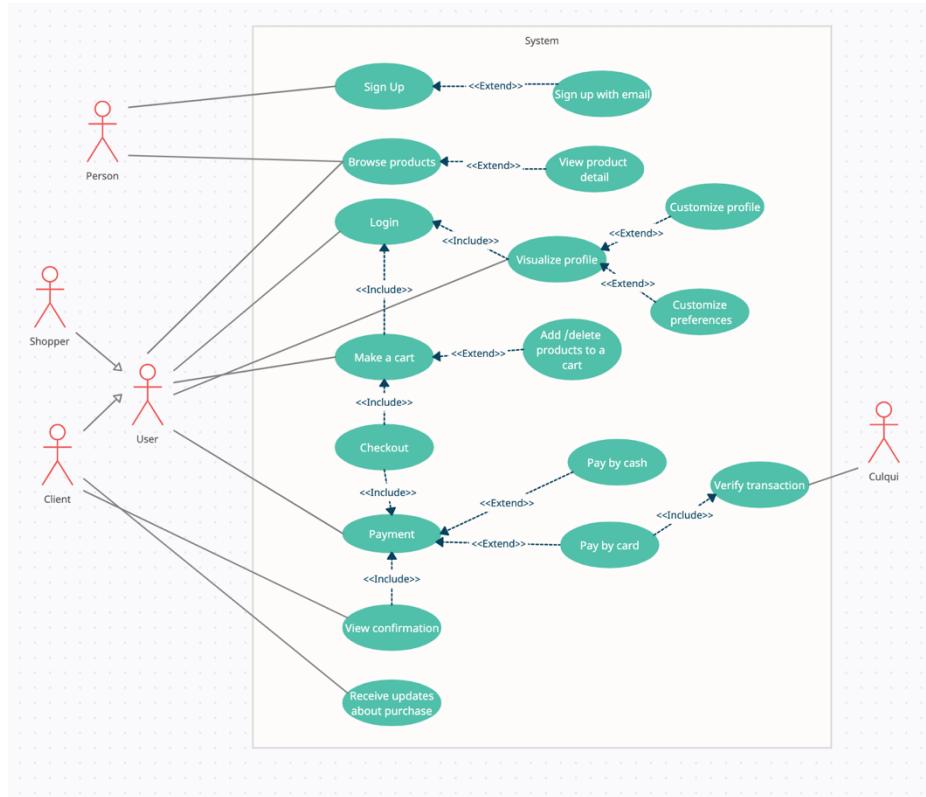
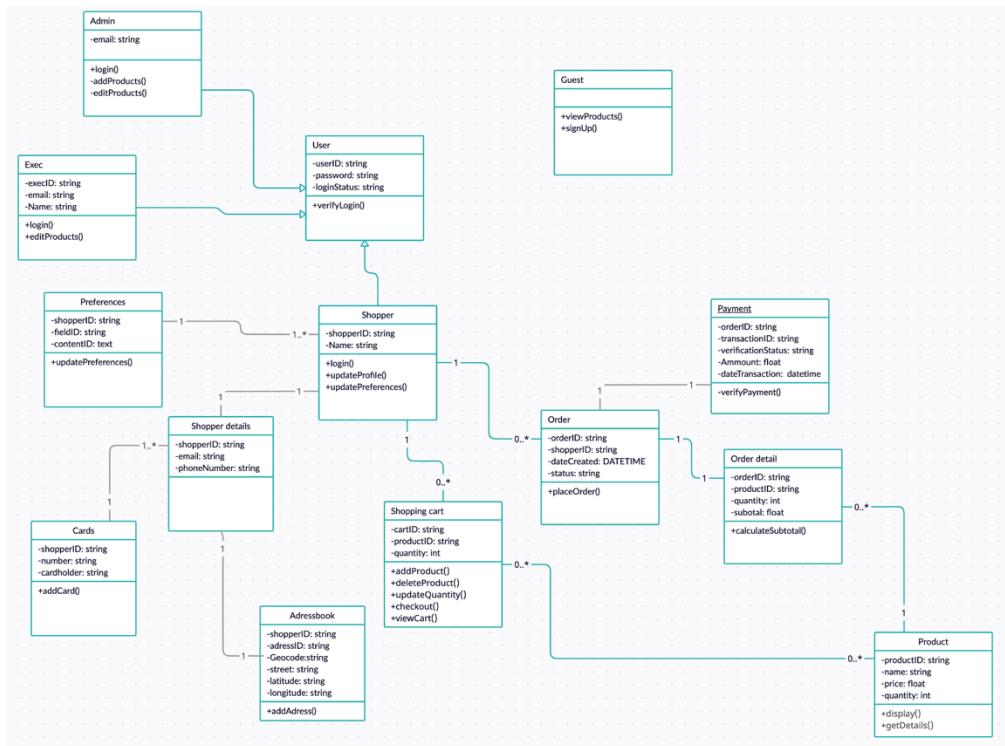


Figure 45: Class diagram sprint 4 (Based on: People.cs.ksu.edu, 2021.)



UI/UX Design

Figure 46: Landing page 1



Figure 47: Landing page 2

Por qué FreshCo?



No te angusties

Recogemos tus preferencias y te damos increíbles ideas



Mantente seguro

Cumplimos con todos los protocolos de Bioseguridad



Ahorra tiempo

Realizamos las compras por ti y te las llevamos a casa



En cada Compra

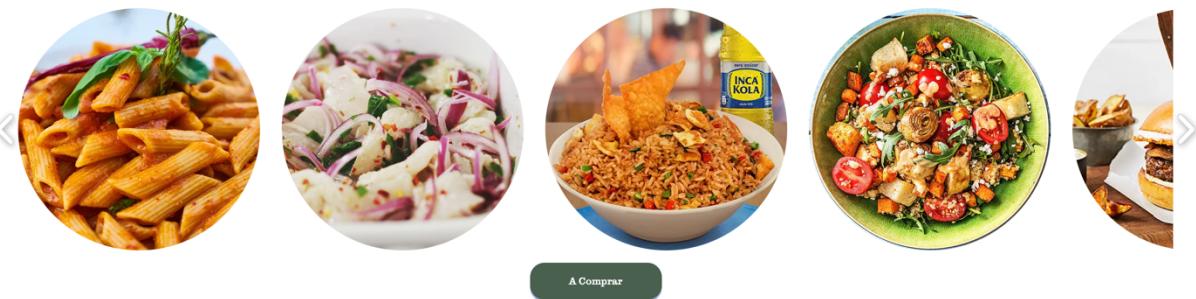
- Ingredientes de alta calidad
- Ingredientes convenientes para cocinar
- Nuevas sugerencias después de tus compras

[A Comprar](#)



Figure 48: Landing page 3

Variedad de Tipos de Cocina



[A Comprar](#)



Flexibilidad

Distintas posibilidades para gestionar tus compras

Figure 49: Landing page 4

FAQ

¿Las compras son seguras?

Si, cumplimos con todos los protocolos de Bioseguridad

¿Los productos son frescos?

Si, todos los productos son adquiridos en la mejor fecha y son tratados para que obtengas lo mejor

¿Cuándo es la entrega?

Al momento de la compra tu decides el marco de tiempo para recibir tu entrega

¿Mis datos son privados?

Si, tu perfil es generado bajo protocolo de encriptación donde el sistema aprende de ti

CONTACTO

Queremos oír más de tí



Testing

Table 24: Sprint 4 testing

Test scenario	User story	Test cases	Expected result
Verify the product search page	13	Search for different products Try different text combinations Apply filters Search for nonexistent products	Successful search
Verify products can be added to the cart	15	Try manually adding all the products Try adding a same product several times Make different product combinations	Products are added successfully
Verify the cart can be edited	16	Change products quantities Delete products Add products Make different combinations	Successfully edited cart
Verify the checkout of the card	17	Edited different times the cart Try checkout different times Cancel checkout Checkout again	Successfully checked out cart
Verify the cash payment option	19	Try entering payment several times Pay with a legit card Pay with a fake card Cancel payment	Successful payment
Verify the confirmation screen	22	Try different checkouts Try different cart combination Try multiple operations at the same time	Confirmation screen appears successfully

Sprint Review

Table 25: Sprint 4 review

User Story	Delivered?	Acceptance criteria	Accepted?
13	YES	Product can be searched on a browser Non existing products show similar products Queries are direct coincidences Brands shorten the searches	YES Barebones remains to be upgraded
15	YES	Products are added to cart by actioning buttons Several units can be added at the same time	YES Barebones remains to be upgraded
16	YES	Products in cart can be deleted Units of products can be edited	YES Barebones remains to be upgraded
17	YES	The temporal cart is registered and triggers payment page Cart issues a unique ID	YES Barebones remains to be upgraded
19	YES	Button allows to pay by cash Issue unique ID of transaction	YES Barebones remains to be upgraded
22	YES	Issue unique order ID Register order to be fulfilled Lower stock with products Show in screen confirmation	YES Barebones remains to be upgraded

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint followed the expected rhythm. The accomplishment of the story points was near the average expected velocity. The team seemed to had found an ideal rhythm. During this Sprint all the work was related to the carts, checkout, and confirmation. Since all of these units are related, the work and integration were smoother.

Figure 50: Sprint 4 Burndown Chart

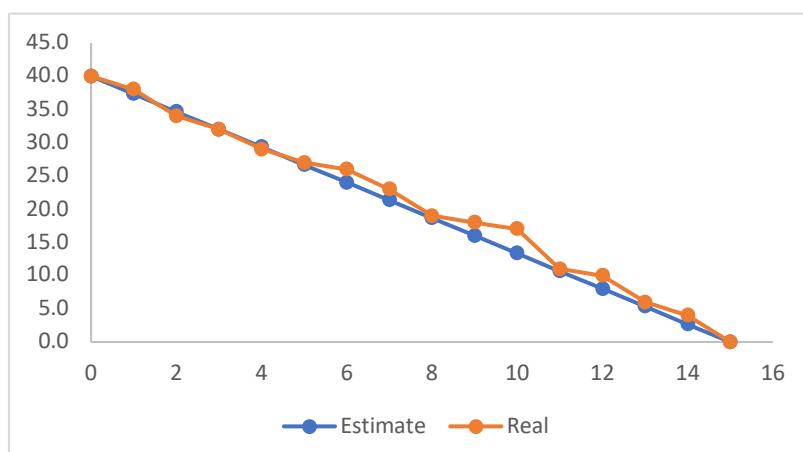


Table 26: Sprint 4 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement backend and queries for search						x
Implement cart's back end						x
Implement cart's view						x
Implement temporal state cart						x
Implements checkout backend						x
Implement checkout view						x
Implement cash payment triggers						x
Implement card payment triggers						x
Implement payment view						x
Implement order registration						x
Implement confirmation page						x
Implement confirmation view						x

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

Table 27: Sprint 4 Retrospective

What happened?		Future actions
What went well?	Team managed to complete sprint goals Test customer could use the page with no problem	
What did not go well?	It was identified that technical debt remains not discussed	Discuss refactoring options at the end of the project
What puzzles us?	Further Deployment on cloud might get difficult	

2.2.5. Sprint 5

This Sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the Sprint. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 28: Sprint 5 Backlog

As ID	User Story			Estimation	Acceptance criteria
	A/a/n ...	I want ...	so that...		
Sprint 5					

24	Client	to receive an invoice on my mail	I can check the result of my purchase	5	Send a pdf invoice to the client Template includes always same elements Mail arrives in less than 5 minutes
8	Shopper	to customize my preferences	I can get personalized offers	13	Tagging system identifies customer preferences Tag's associate profile user with certain food Preference customization shows up as an option Preference form is interactive
26	Client	to track the state of my delivery	I know where it is	5	State is set by triggers or confirmations State can be notified by email Email is short and graphic
28	Client	to receive suggestions when I make a purchase	I can find more options	20	Shopping suggestion appears when searching Shopping option has preset recurrent items Customer can customize the products System will compare suggestion with result and learn
30	Client	to rate the purchase	I can transmit my opinion	3	Receive email with link to rating Rating form is short Rating form is associated with preferences

Table 29: Sprint 5 Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement triggers for sending invoice mail	x					
Design invoice template	x					
Implement invoice and DB requests	x					
Implement tags on NOSQL DB	x					
Make tags associations	x					
Join form with tags on DB triggers	x					
Implement status triggers for sending email	x					
Design status template	x					
Implement Recombee API	x					
Connect 3 rd party with RDBS and NOSQL DB	x					
Implement recommendation on product pages	x					
Implement trigger for rating email	x					
Design rating template	x					
Implement storage of rating	x					

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the Sprint.

UI/UX Design

Figure 51: Preferences form page



Testing

Table 30: Sprint 5 testing

Test scenario	User story	Test cases	Expected result
Verify the mail invoice	24	Try different accounts Try different orders Compare all the products	Invoice sent to mail
Verify the preferences customization	8	Try all the inputs Enter different values Try entering different kind of values Enter incorrect data	Data can be successfully edited
Verify the order status	26	Try different accounts Simulate different status changes	Order status modified and reported
Verify the suggestions	28	Make custom preferences Try different accounts Make different orders Build different fake purchase histories	Suggestions are presented
Verify the rating option	30	Try different accounts Try different orders Match the products in orders with rating	Rating is completed

Sprint Review

Table 31: Sprint 5 review

User Story	Delivered?	Acceptance criteria	Accepted?

24	YES	Send a pdf invoice to the client Template includes always same elements Mail arrives in less than 5 minutes	YES Barebones remains to be upgraded
8	YES	Tagging system identifies customer preferences Tag's associate profile user with certain food Preference customization shows up as an option Preference form is interactive	YES Barebones remains to be upgraded
26	YES	State is set by triggers or confirmations State can be notified by email Email is short and graphic	YES Barebones remains to be upgraded
28	YES	Shopping suggestion appears when searching Shopping option has preset recurrent items Customer can customize the products System will compare suggestion with result and learn	NO Upgrades remain to be done so it works properly
30	YES	Receive email with link to rating Rating form is short Rating form is associated with preferences	YES Barebones remains to be upgraded

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint followed the expected rhythm but was a little behind schedule. The accomplishment of the story points was near the average expected velocity. Most of the developments of this sprints were related to mailing the content of the order objects. The recommendation system caused some issues because the chosen option was to use a 3rd party to make the recommendation work. The inputs will flow from the internal DBs and the output of the 3rd part will come back again to it. Even though the system was completely integrated, the recommendations weren't so satisfying.

Figure 52: Sprint 5 Burndown chart

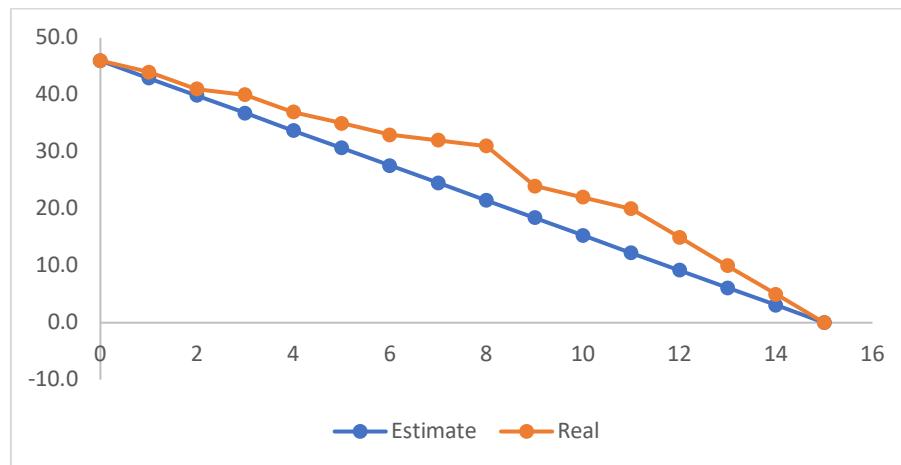


Table 32: Sprint 5 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement triggers for sending invoice mail						x
Design invoice template						x
Implement invoice and DB requests						x
Implement tags on NOSQL DB						x
Make tags associations						x
Join form with tags on DB triggers						x
Implement status triggers for sending email						x
Design status template						x
Implement Recombee API						x
Connect 3 rd party with RDBS and NOSQL DB						x
Implement recommendation on product pages						x
Implement trigger for rating email						x
Design rating template						x
Implement storage of rating						x

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

Table 33: Sprint 5 Retrospective

What happened?	Future actions
What went well?	Team managed to complete sprint goals

What did not go well?	The implementation of the 3 rd party system API did not product the expected result	Fix on the next sprint
What puzzles us?	Dependence on 3 rd parties might be complicated on long term scalability	

2.2.6. Sprint 6

This Sprint was developed for 3 weeks. The initial planning meeting took place on the first day of the Sprint. The most important user stories were chosen, and they were incorporated into the Sprint Backlog. Then, from the user stories, some tasks were defined for the development of the Sprint.

Sprint Backlog

Table 34: Sprint 6 Backlog

ID	User Story			Estimation	Acceptance criteria
	As a/an ...	I want ...	so that...		
Sprint 6					
28	Client	to receive suggestions when I make a purchase	I can find more options	4	Shopping suggestion appears when searching Shopping option has preset recurrent items Customer can customize the products System will compare suggestion with result and learn
32	Warehouse operator	to receive a notification every time an order is placed	I can start preparing the order	3	Notification is sent by email to worker email to worker Notification is sent before packing day Each order placed is stacked on worker dashboard
35	Warehouse operator	to print the details of the delivery	I can attach them to the package	3	Order details pdf is attached on notification email Each order details file is attached on the stack of the dashboard
36	Warehouse operator	to confirm that I completed the order	I can do the next operation	3	Option to confirm that order was completed

38	Business operator	to add products to the database	they can be sold	5	Admin limited privileges for exec user Exec dashboard shows the option to add products	
39	Business operator	to edit the attributes of a product	they can be updated	8	Admin limited privileges for exec user Exec dashboard shows the option to customize products attributes	
45	Business operator	to get an automatic report of the needed items for purchases	I can complete the procurement	13	Exec dashboard can produce a pdf with the procurement list A reminder with the pdf attached is sent by email	

Table 35: Sprint 6 Kanban board

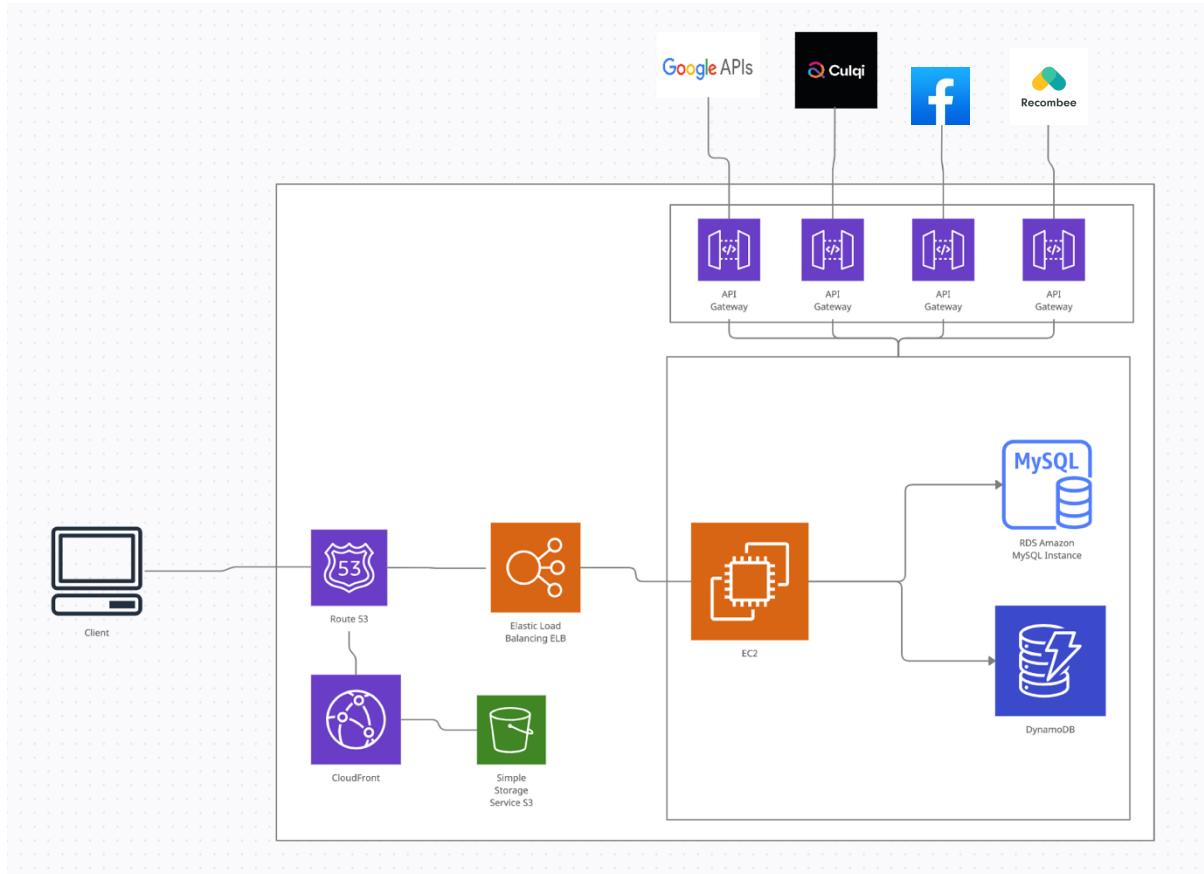
Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement recommendation on product pages			x			
Setup trigger for mailing orders details	x					
Design order details template	x					
Implement worker dashboard	x					
Implement orders view on worker dashboard	x					
Implement order completion button on worker dashboard	x					
Implement executive dashboard	x					
Give new privileges to exec user	x					
Implement product catalogue customization view to exec dashboard	x					
Setup trigger for procurement list on mail	x					
Implement procurement view on exec dashboard	x					
Make connections to DB on worker dashboard	x					
Make connections to DB on exec dashboard	x					
Create login page for workers and execs	x					

Development

Even though for Scrum documentation is not a priority, the team still considered relevant to present some intermediate products that were made during the development of the Sprint.

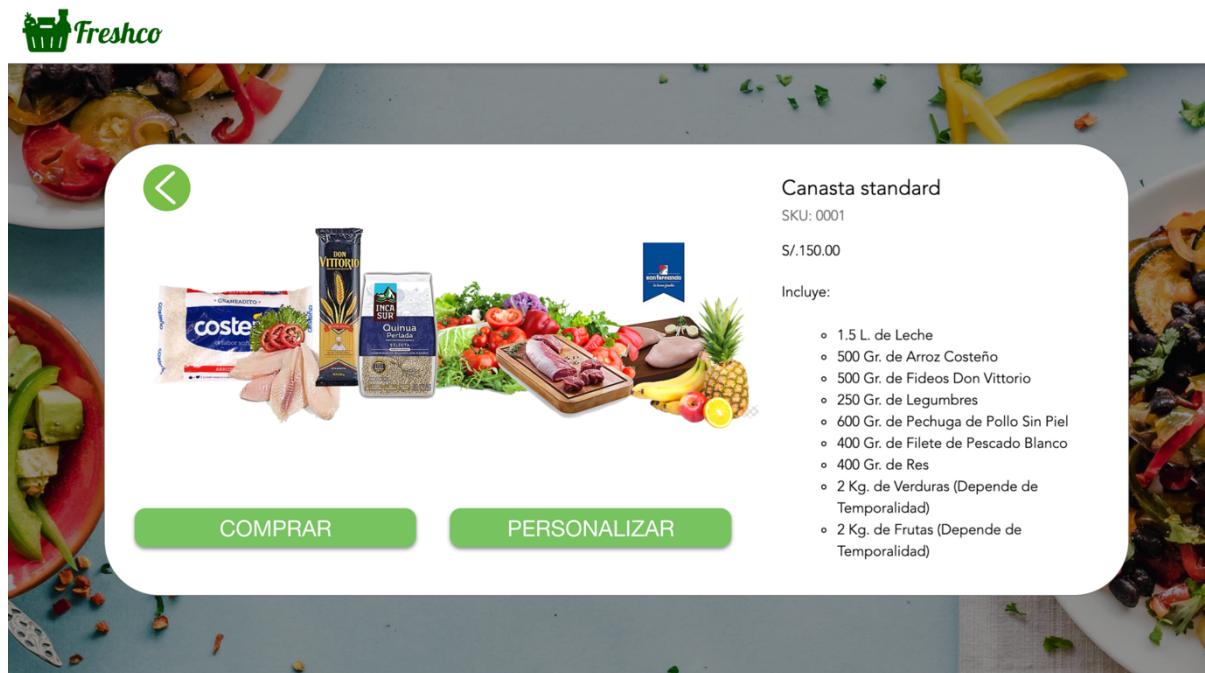
System Design (UML)

Figure 53: System architecture on AWS



UI/UX Design

Figure 54: Customized suggestions page



Testing

Table 36: Sprint 6 testing

Test scenario	User story	Test cases	Expected result
Verify the suggestions	28	Make custom preferences Try different accounts Make different orders Build different fake purchase histories	Suggestions are presented
Verify the workers mail notification	32	Try different orders and combinations Try different clients	Worker successfully notified
Verify the worker dashboard	35,36	Try login with appropriate and inappropriate accounts Try several orders Try confirming orders	Worker dashboard works properly
Verify the exec dashboard	38,39	Try login with appropriate and inappropriate accounts Modify the catalogue Change the available stock Try incorrect operations	Exec dashboard works properly
Verify the automated reports	45	Print a report Try different filters Export the report	Reports are created successfully

Sprint Review

Table 37: Sprint 6 review

User Story	Delivered?	Acceptance criteria	Accepted?

28	YES	Shopping suggestion appears when searching Shopping option has preset recurrent items Customer can customize the products System will compare suggestion with result and learn	YES Barebones remains to be upgraded
32	YES	Notification is sent by email to worker email to worker email to worker Notification is sent before packing day Each order placed is stacked on worker dashboard	YES Barebones remains to be upgraded
35	YES	Order details pdf is attached on notification email Each order details file is attached on the stack of the dashboard	YES Barebones remains to be upgraded
36	YES	Option to confirm that order was completed	YES Barebones remains to be upgraded
38	YES	Admin limited privileges for exec user Exec. dashboard shows the option to add products	YES Barebones remains to be upgraded
39	YES	Admin limited privileges for exec user Exec. dashboard shows the option to customize products attributes	YES Barebones remains to be upgraded
45	YES	Exec. dashboard can produce a pdf with the procurement list A reminder with the pdf attached is sent by email	YES Barebones remains to be upgraded

Burndown Chart

As it can be seen on the graph, generally the development of this Sprint followed the expected rhythm. The accomplishment of the story points was near the average expected velocity. For this Sprint, the main concern of the team was directed to the development of mail notifications, worker dashboards, executive dashboards, and reporting. The team lagged a little while developing the new dashboards, however, managed to catch up at the end of the Sprint. In general, the last implementations were related to already existing functionalities, so it was not so difficult in the end.

Figure 55: Sprint 6 Burndown chart

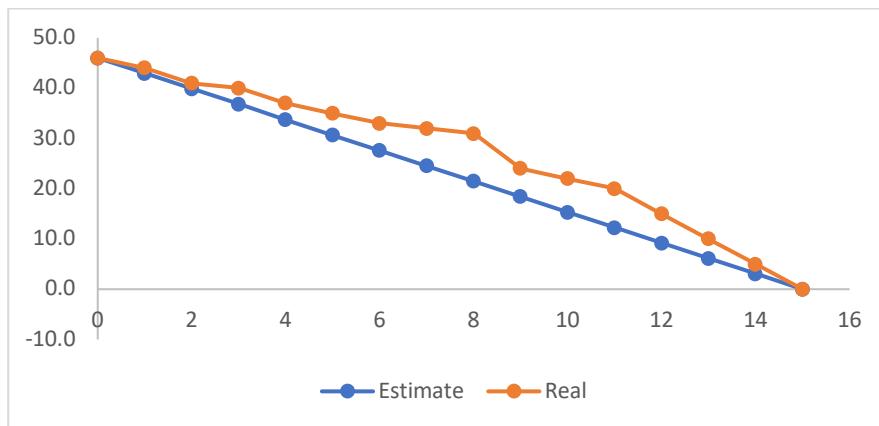


Table 38: Sprint 6 final Kanban board

Tasks	To Do	Analysis	Dev	Test	Deploy	Done
Implement recommendation on product pages						x
Setup trigger for mailing orders details						x
Design order details template						x
Implement worker dashboard						x
Implement orders view on worker dashboard						x
Implement order completion button on worker dashboard						x
Implement executive dashboard						x
Give new privileges to exec user						x
Implement product catalogue customization view to exec dashboard						x
Setup trigger for procurement list on mail						x
Implement procurement view on exec dashboard						x
Make connections to DB on worker dashboard						x
Make connections to DB on exec dashboard						x
Create login page for workers and execs						x

Sprint Retrospective

The last day of the Sprint. The team got together to discuss about their performance during the Sprint in general. The main conclusions were written down.

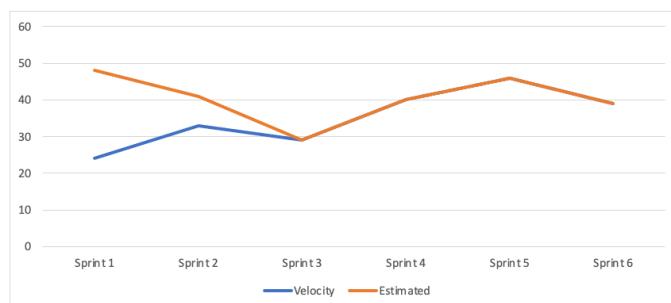
Table 39: Sprint 6 Retrospective

What happened?		Future actions
What went well?	Team managed to complete sprint goals Integral deployment on AWS was successful	
What did not go well?	Refactoring requires a high-tech debt	Start refactoring on next release
What puzzles us?	Long term orientation discussion	

Final Performance Review

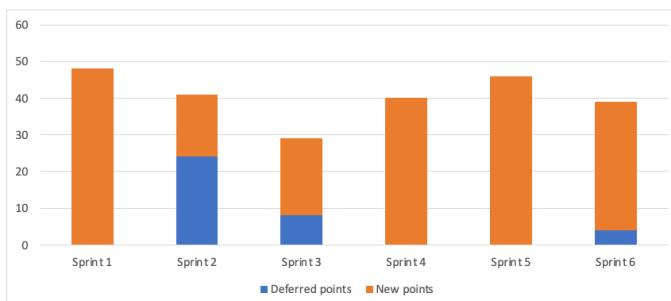
By analysing the performance of the team some findings were also made. First, at the beginning of the project, the team was not matching the expected velocity, but after some point in the middle they managed to catch up and increase their average velocity. Also, in general, the work developed on all the Sprints after the middle was composed of new points with no deferred tasks.

Figure 56: Final Expected vs real velocity



	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6
Velocity	24	33	29	40	46	39
Estimated	48	41	29	40	46	39

Figure 57: Final Deferred vs new story points



	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6
Deferred points		24	8	0	0	4
New points	48	17	21	40	46	35

Section 3: Final Project Report

What changed?

The development process of the system since day 1 was expected to face change. At the beginning the system had a high-level approach that was meant to distance itself from the traditional supermarket online shopping approach. However, as the product was being developed, the business side revealed that it was uncharted territory. It is only natural that for the development of a product and business model for a start-up that many times what is required will change. This was frustrating and sometimes a little difficult for the team to overcome.

First, the high-level requirements contemplated 3 modules: Ecommerce, Supply Chain Management and Business Intelligence. This was discarded during the pre-project. For an approximately 5-month development project, such requirements would not be realistic. So, the team decided to go deep into the core of what was the need of the system and produced the list of most important features required. This was only accomplished by identifying the users and the stories each of them had to accomplish. The most important identified stories were centered more in the ecommerce requirements rather than the business intelligence ones. The investors were not so pleased with this because they were pitched an advance solution with predicting machine learning properties. However, as the correct fit of problem/solution was more important, the store stories ended being prioritized. This also affected the work of the data scientist because her specialty was not being fully employed and she had to focus on database engineering and management.

Second, during the first Sprint the Scrum team could not decide the programming language, the architecture, the development framework, and the tools. The CTO suggested going with well-known powerful approaches that are employed on the most popular tech unicorns. However, the team lacked qualifications to even be able to begin with them. The solution was to choose the path that offered enough common ground for everyone's abilities. For example, they experimented trying to sketch a microservice architecture based on NodeJS and Kubernetes. This was discarded because it was too ambitious and simply not feasible for the available time. Most of the team had only junior experience. Python was the chosen programming language and Django was the chosen framework under a monolithic architecture on AWS. The complete process of discussing, sketching, and speculating slowed down the processes as a whole and left the first Sprint with unaccomplished objectives.

After the third Sprint it was evident that the feedback was not so valuable. The business team on their side got behind schedule and could not act on par with the Scrum team. They tested the hypothesis with a non-technical solution that emulated what was being developed. The most important insights were written down. Clients wanted the business to offer delivery instead of outsourcing it. Clients wanted more options to customize their own and their family members' preferences. There was evidence to suggest that some clients prefer to fully control what they bought and get suggestions of what to cook, other clients wanted to get the suggestions first and with that adapt to it. It was assumed that maybe nutrition service of planning and purchasing could be added, however, as it involved a completely new service, it was rejected for the current release. Furthermore, as some stories were redundant, they were deleted or merged with other stories.

During the second half of the development, some selected clients were chosen to experiment with the MVP being built. In general, the feedback was positive, however, as it was a small sample, the long-term concerns were still raised. The UX was fine and there was no major problem with the usability. In the end, some parts of the other original modules were included as basic functionalities embedded on the main architecture. The biggest change was that the data science and machine learning parts were expected to be developed in house, but it ended up being outsourced to a 3rd party. As the deadline would not allow it, the data scientist focused on integrating the external platform with the own system architecture and databases. She also managed to develop a non-relational database.

Finally, the system was acceptable to release as an MVP to the general public. It still needs some refactoring and maintenance and is expected to do so during the next release.

The software process

The Scrum development had both advantages and disadvantages. Overall, the experience was satisfying with the flexibility issue. However, it was challenging to adopt it with the team. None of the team members had previous experience with it and everyone discovered what worked best during development. Many of the tasks were based on assumptions and not on feedback from real clients because it was not early accessible. The process had to be adapted to leave some documentation behind to help rework later because the development left considerable technical debt and is expected to be redone on other architecture with other technological components. Many trade-offs had to be made on areas in which there were some non-negotiable issues. This only demonstrates that the start-up experience is not as fashionable as the “Social Network” movie presents.

Developing the system

As it was stated before, the first half of the development was chaotic and many sacrifices had to be made. The documentation provides a decent guide to the development of the same system but with a more experienced team and broader schedule. As the team strictly adhered the principle of focusing on code rather than documentation, there are many “holes” left in between. The interpretation for the backlog and other artifacts can only be perfectly understood if one was part of the team. A different team would need some extent of support of the original team to interpret the documentation and adapt it. Many times, the physical artifacts would be altered or discarded in such manner that the digital documentation cannot be interpreted without context. It indeed requires some forensic work.

Management:

When I started the project, I expected to complete it in a shorter time, namely in 3 months. However, since the project was completing in parallel with the learning process, the time I spent on the entire project was 5 months. I expected that with an Agile approach work would go smoother, however, I discovered later that for this particular approach chaos was the rule. In my opinion, in the future, the solution to the problem of the actual and predicted time mismatch will be careful planning, allocation of resources and detailed scheduling before starting work.

Lessons learnt

The things I learned working on my project are:

- No matter how direct or clear the examples and lessons in the textbooks or other materials are, each case is unique, there is no source material that can guide you step-by-step and you must discover what works best for your team.
- It does not matter if what you design is not perfect or you do not get to represent what you imagine; usually while coding, testing, and failing you get to discover what works and what does not. And many times, you get something to work on and you never figure out why.
- In real life you face many situations that is not under your control and scope and the best thing you can do is take a step back and adapt to take two steps forward.

Final comments

It is almost impossible to complete such projects without help, collaboration, and mentoring. Yet somehow students with no prior experience in software engineering are expected to develop it on their own and by self-teaching. The hustle is indeed exiting, however, failing in an isolated environment without the ability to get feedback from instructors or to work with your peers exceeds the learning experience. In my honest opinion, this learning experience must be upgraded. And if you ever must do this project, find a way to get the most real-life experience you can get.

Bibliography

- Amazon Web Services, Inc. 2021. About AWS. [online] Available at: <<https://aws.amazon.com/about-aws/>>
- Alaimo, M. and Salias, M., 2015. Proyectos Ágiles con Scrum: Flexibilidad, aprendizaje, innovación y colaboración en contextos complejos. 2nd ed. Kleer
- Cohn, M., 2004. User Stories Applied: For Agile Software Development. Addison-Wesley Professional
- Cohn, M., 2005. Agile Estimating and Planning. Prentice Hall
- Djangoproject.com. 2021. The Web framework for perfectionists with deadlines | Django. [online] Available at: <<https://www.djangoproject.com/>>
- Grinberg, M., 2018. Flask web development: developing web applications with Python. 2nd ed. O'Reilly Media. Addison-Wesley Professional
- Forcier, J., 2008. Python Web Development with Django.
- Git-scm.com. 2021. Git. [online] Available at: <<https://git-scm.com/>>
- GitLab. 2021. Iterate faster, innovate together. [online] Available at: <<https://about.gitlab.com/>>
- Hostinger Tutorials. 2021. What is MySQL: MySQL Explained for Beginners. [online] Available at: <<https://www.hostinger.com/tutorials/what-is-mysql>>
- Maurya, A., 2018. Running lean: Iterate from Plan A to a Plan That Works. Beijing: O'Reilly.
- Patton, J., Treseler, M. and Troutman, E., 2014. User story mapping: Building Better Products Using Agile Software Design. O'Reilly Media
- People.cs.ksu.edu. 2021. Class diagram Version 1. [online] Available at: <http://people.cs.ksu.edu/~reshma/798_ClassDiagram.htm>
- PMI, 2017. A guide to the project management body of knowledge. Newtown Square, PA: Project Management Institute.
- Pressman, R. and Maxim, B., 2019. Software engineering: A Practitioner's Approach. 9th ed. McGraw-Hill Education.
- Python.org. 2021. Python For Beginners. [online] Available at: <<https://www.python.org/about/gettingstarted/>>
- Rasmussen, J., 2013. The Agile Samurai: How Agile Masters Deliver Great Software. Pragmatic Bookshelf
- Ries, E., 2011, The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown Business
- Stellman, A. and Greene, J., 2014. Learning Agile: Understanding Scrum, XP, Lean, and Kanban. O'Reilly
- W3schools.com. 2021. CSS Tutorial. [online] Available at: <<https://www.w3schools.com/css/>>

W3schools.com. 2021. HTML Tutorial. [online] Available at: <<https://www.w3schools.com/html/>>

W3schools.com. 2021. JavaScript Tutorial. [online] Available at: <<https://www.w3schools.com/js/default.asp>>

Software tools used

Codepen: <https://codepen.io>

Creately: <https://creately.com/es/home/>

Figma: <https://www.figma.com/>

Microsoft Excel: <https://www.office.com/>

Miro: <https://miro.com/>

Wix: <https://wix.com/>