
FRIGO INC.

**FRIGO
Vision**

Version 1.0

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

Revision History

Date	Version	Description	Author(s)
26/01/24	0.1	First draft	Tristan Kristoffersen Askvik, Ingunn Tonetta Erdal, Andreas Røed Tvette & Halvor Etterdal
04/04/24	0.2	Second draft	Ingunn Tonetta Erdal
22/04/24	1.0	Finished draft	Ingunn Tonetta Erdal

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

Table of Contents

1.	Introduction	5
1.1	Purpose and scope	5
1.2	Definitions, Acronyms, and Abbreviations	5
1.3	Overview	5
2.	Positioning	5
2.1	Business Opportunity	5
2.2	Problem Statement	6
2.3	Product Position Statement	6
3.	Project goals	6
3.1	Impact goals	6
3.2	Result goals	6
3.3	Process goals	7
4.	Stakeholder and User Descriptions	7
4.1	Market Demographics	7
4.2	Stakeholder Summary	7
4.3	User Summary	8
4.4	User Environment	9
4.5	Key Stakeholder or User Needs	9
5.	Product Overview	10
5.1	Product Perspective	10
5.2	Summary of Capabilities	10
5.3	Assumptions and Dependencies	10
5.4	Risk analysis	10
5.5	Quantifiable and non-quantifiable benefits	12
5.6	Estimated costs	12
5.7	Licensing and Installation	12
6.	Product Features	12
7.	Constraints	13
8.	Quality Ranges	13
9.	Precedence and Priority	14
10.	Other Product Requirements	14
10.1	Applicable Standards	14
10.2	System Requirements	14
10.3	Performance Requirements	14
11.	Documentation Requirements	15
11.1	User Manual	15
11.2	Online Help	15
11.3	Installation Guides, Configuration, and Read Me File	15

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

11.4	Labeling and Packaging	15
12.	A Feature Attributes	15
12.1	A.1 Status	16
12.2	A.2 Effort	16
12.3	A.3 Stability	16
12.4	A.4 Target Release	16
12.5	A.5 Assigned To	16
12.6	A.6 Reason	16

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

Vision

1. Introduction

This Vision document is prepared by Group 13, composed of students studying Digital Infrastructure and Cybersecurity, as part of the subject IDATT1005 System Development course at the Norwegian University of Science and Technology (NTNU). The purpose of this document is to collect, analyze, and define high-level needs and features of the “FRIGO” app. The introduction of the Vision document provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this Vision document.

1.1 Purpose and scope

This Vision document aims to outline the high-level needs and features of an at-home meal-planning tool, focusing on the capabilities required by stakeholders and target users, elucidating why these needs exist and how the at-home meal-planning tool will meet them. This tool is specifically designed to streamline the management of food inventory, provision of recipes, and generation of shopping lists, catering to the needs of private households and student collectives.

1.2 Definitions, Acronyms, and Abbreviations

Fridge – A user’s *fridge* is a function of the application that keeps track of the food the user has at home, such as food stored in the fridge or in their cabinet.

Household – A feature that allows multiple users living together to have a shared fridge, shopping lists and recipes.

MVP – Minimal Viable Product

1.3 Overview

This document is organized into sections detailing the project’s positioning, goals, stakeholder and user descriptions, product overview, and detailed feature set.

2. Positioning

2.1 Business Opportunity

The purpose of the product is to meet the business opportunity in the at-home meal-planning market. This is an incredibly large market, where we have chosen to have our focus on young adults, both those living alone and those with a shared food inventory. Young adults are often newer to cooking by themselves, have tighter food budgets and often have limited time (such as students during exam periods, or working on top of studies). Due to this, young adults have a lot to gain by using our product.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

2.2 Problem Statement

The problem of	Planning dinner and shopping based on current ingredients
affects	People who shop and cook at home, especially those living by themselves such as students
the impact of which is	Needing to spend a lot of time and energy every day to plan dinner and what to buy
a successful solution would be	A solution which saves time, money, and reduces food-waste

2.3 Product Position Statement

For	Young adults.
Who	Have little time to shop and plan dinner based on their ingredients.
FRIGO	Is a mobile-friendly, browser-based web app.
That	Gives you an overview of what is in your food inventory and make shopping lists based on the user's current ingredients.
Unlike	Many meal-planning and inventory management apps, FRIGO integrates a unique "household" feature that allows multiple users to manage a collective inventory in real time. This ensures that all users in a household have an up-to-date overview of available ingredients, avoiding duplicating purchases and reducing food waste.
Our Product	Is an efficient, accessible, including, user-friendly and dynamic app. It provides an all-in-one solution for managing food inventory and is available for everyone at any given time.

3. Project goals

3.1 Impact goals

Our vision for the FRIGO app is to serve as a tool for individuals and collectives, such as student households in Trondheim, enabling them to maintain a clear and efficient overview of their food inventory at any given moment. To this end, our impact goals for the FRIGO app are designed to eliminating redundant checks and uncertainty about food stocks because it provides a shared platform that allows collectives households to synchronize their food inventories, ensuring all members have real-time access to consolidated list of available ingredients. The app also not only enables users to track what is in their fridge to maximize their use of their ingredients. This helps with both simplifying shopping trips and reducing the time and commitment associated with meal planning. It also contributes to cost savings and reduces the amount of food waste and therefore creates a more sustainable living environment. The overall impact goal it to deliver an app experience that supports the lifestyle of our target users, who are often pressed for time, on a budget, and looking for ways to simplify their daily routines without compromising on the quality and enjoyment of their meals.

3.2 Result goals

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

By the end of the project on the 26th of April we wish to have developed an app that aims to revolutionize meal planning, making it more efficient, enjoyable and eco-friendly for our users. In the pursuit of our vision for the FRIGO app, our team is committed to delivering a set of precise, actionable results that directly meet our users 'needs and our clients' expectations. Our focus is on launching a browser-based web-application that is optimized for mobile devices, ensuring accessibility and ease of use for users on the go. One of the core functionalities of the FRIGO app is the ability to add currently owned food into the app and track their currently owned food items. Additionally, the app will feature a shopping list tool that not only allows users to create lists based on needed recipe ingredients but also facilitates the easy transfer of purchased items back into their virtual pantry upon the completion of a shopping trip. Lastly, we want the app to be designed with a simple, minimalistic aesthetic. The app's user interface will feature round borders and a bottom-expanded menu, enhancing usability and navigation.

3.3 Process goals

In the development of the FRIGO app, while the end product is our primary objective, we also emphasize the importance of the development journey we undertake as a team. This project is meant to give the participating students competence in project management, building strong teamwork, and most importantly, facilitating personal and professional growth.

Growth, both as a collective and individually, is a cornerstone of our project ethos. We anticipate that each team member will experience personal development through the challenges and triumphs of this project. This growth will be evaluated in the individual team members' reflection note, which is an essential part of our self-improvement journey.

By the project's conclusion, we envision a team that's not only proud of the product we've created but also the strides we've made in our capabilities, our collaborative efforts, and our self-awareness.

4. Stakeholder and User Descriptions

This section outlines the distinct roles and responsibilities of the stakeholders involved in the development of the FRIGO app. The Product Owner's role is to act as the liaison between stakeholders and the development team, communicating process, gathering feedback, and aligning expectations. The development team's is central to the app's development, transforming vision into a functional product. Teachers and the Student Assistant contribute by offering educational guidance and feedback. Lastly, our project is driven by the needs and requirements as outlined by our stakeholder and client, Callum Gran. It's important for the client to ensure that the product will be well-received by its intended audience and meets its requirements.

4.1 Market Demographics

The project is not profit-driven; therefore, our growth goal is based on the size of the userbase and the satisfaction of these users. Given the release of a product which fulfills our result goals, the potential number of users is quite high. If we focus on the Norwegian market, there's a high percentage of the population within our target audience – as described in section 2.1. As food prices have inflated there is also a higher general consciousness in the population regarding food-spending, which makes an app like this even more viable.

4.2 Stakeholder Summary

Name	Description	Responsibilities
Product owner	Andreas Røed Tvette	- Prioritize the product backlog.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

		<ul style="list-style-type: none"> - Communicate with stakeholders. - Monitoring the project's progress.
Developer team	Tristan Kristoffersen Askvik, Ingunn Tonetta Erdal, Andreas Røed Tvette and Halvor Etterdal	<ul style="list-style-type: none"> - Develop the FRIGO app. - Attend weekly meetings and give feedback. - Reporting and documentation. - Provide tests for the customer.
Teachers	Surya Bahadur Kathayat, Muhammed Ali Norozi and Grethe Sandtrak	<ul style="list-style-type: none"> - Main head of the project. - Provide feedback. - Grade the project.
Student Assistant	Anna Jansdatter Bjørgo	<ul style="list-style-type: none"> - Attends weekly meetings. - Provide feedback during meetings and answer questions.
Client	Callum Gran	<ul style="list-style-type: none"> - Represents the interests and needs of the end-users. - Providing the primary vision and requirements for the project. - Offers feedback and approval on project milestones. - Provides clear and actionable requirements for the system development. - Reviews and test the product.

4.3 User Summary

Name	Description	Responsibilities	Stakeholder
Young adult purchasing food only for themselves	An individual who uses the system for personal grocery shopping, managing their own food inventory and meal planning.	<ul style="list-style-type: none"> - Inputs personal food inventory - Selects recipes based on personal preferences - Generate personal shopping lists 	Callum Gran
Young adults purchasing food together	A group of users such as roommates, friends and families who use the system collectively for joint grocery shopping and shared meal	<p>In addition to the responsibilities of the above user:</p> <ul style="list-style-type: none"> - Coordinates the above points with one or more other users 	Callum Gran

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

	planning.		
--	-----------	--	--

4.4 User Environment

The target users of our system, which include families and young adults purchasing food both individually and collectively, operate in a diverse set of environments:

- Anywhere from 1 person living alone, to multiple people living together.
- The application will be used in every timespan from just a quick glance (quickly checking what they have in the fridge), to longer timespan such as while using a recipe to create a meal and creating a recipe, or on a shopping trip.
- The application will rarely be used while completely stationary, and mostly when on the move in some way: at the grocery store, in the kitchen or anywhere else.
- Due to this quick and on-the-move usage the users will want the main platform to be a mobile phone.

4.5 Key Stakeholder or User Needs

Need	Priority	Concerns	Current Solution	Proposed Solutions
Ability to add food items to the app.	High	Items are not removed when making a recipe recommended by the system.	User must manually change the inventory.	When making food from a recipe, the items used are automatically subtracted from inventory.
Real-time inventory updating.	High	Keeping the inventory of items is labor intensive.	Manually enter the items you have bought.	When finishing a shopping-list the items are automatically added to inventory.
Simplification of recipe and ingredient matching.	Medium	Recipes require specific items which causes them to not be recommended.	No solution (Coop chicken breast is not the same as Gilde chicken breast).	Create a feature that recognizes the ingredient categories (for example chicken) rather than specific brands to widen recipe recommendations.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

5. Product Overview

5.1 Product Perspective

The system that will be designed is self-contained and has no other dependencies than an internet connection to make use of the products' storage and collaboration tools. The setting for which the product should be used is on a mobile device, such as a smartphone, and will mainly be in use either in the user's home or supermarket of choice. At the time of writing, there are no other related systems or products that directly influence the development of this system.

5.2 Summary of Capabilities

Table 4-1 Customer Support System

Customer Benefit	Supporting Features
Users will spend less time planning meals, reduce food-waste, and save money.	The overview of current ingredients in their fridge.
Users will be able to better plan their grocery shopping.	The application will allow the user to manually add items to their shopping list, as well as being able to automatically add items which they lack for a specific recipe.
Multiple users living together will be able to synchronize their grocery shopping and meal-planning.	Multiple users can join a <i>household</i> , allowing them to see their shared fridge and shopping lists.
Users can easily know what they have in their fridge while not at home, without having to physically check their fridge.	The application keeps track of all items the user has in their fridge and is therefore always accessible (given a phone and internet).

5.3 Assumptions and Dependencies

- The user has a smartphone to use the product.
- The user has an internet connection to access the website the service is hosted on.
- The user will access the website through a modern web browser.

5.4 Risk analysis

Description of the risk	Mitigations	Probability of the risk occurring. Scale: 1-5	Consequences of the risk. Scale: 1-5
A. The chosen technologies end up being too complex.	Technologies being used have been carefully reviewed to make sure that they are simultaneously capable while not being too complex.	<p>1</p> <p>It is low because we've made sure they are suitable and not too complicated.</p>	<p>4</p> <p>If the risk occurs, we could be unable to deliver a product which fulfills the customers' needs.</p>

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

B. Sickness or inability to work by a project member.	We keep the project's bus-factor above 1 member by having a good flow of information. If this risk were to occur, someone else could pick up their work without issues.	4 All members are currently healthy with no complications; however, sickness does sometimes occur – especially over a longer period such as this project.	3 The loss of a working member could lead to delays or overwork and burn-out for the other members.
C. Loss of data such as code, designs, or other documentation.	We have protocols defined in our employment contract for document and code handling. Everything is stored in the cloud, but with regular local backups.	2 There is a small possibility of project members not following protocol AND a loss to occur.	5 Depending on what is lost and during which stage of development, loss of data could be catastrophic.
D. Conflicts within the group	All project members are conscious of keeping a good dynamic. In the case of a conflict protocols are defined in the employment contract.	3 Smaller conflicts and disagreements can occur, but larger ones have a low probability.	2 Unless there's a very large conflict, it will most likely only lead to temporary delays or bad group dynamics.
E. School assignments and projects may require time that the dev-team intended to use on this project.	Implement a weekly structured schedule to manage time efficiently on each task, ensuring consistent progress throughout the project's duration.	4 The probability of team members conflicting school assignments is relatively high, as we have numerous classes requiring group work, which makes our schedules heavily independent on the availability of others.	2 Unless significant time is diverted to other assignments and projects, then it will most likely only result in temporary delays.
F. Stakeholders' availability may not always be guaranteed, potentially leading to missed critical feedback.	Regular meetings with stakeholder and distribute meeting invitations well in advance to ensure their availability.	2 Given the agreed-upon regular meeting times, the risk of stakeholders being unavailable is low.	3 Should the team miss critical feedback and risk losing our stakeholder, we are prepared to promptly secure a replacement to ensure minimal project disruption.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

Catastrophic					
Critical		E	B		
Hazardous		D			
Marginal			F		C
Negligible				A	
Probability/Consequence	Very low	Low	Medium	High	Very high

5.5 Quantifiable and non-quantifiable benefits

Our project is not designed for any specific business, which inherently limits our ability to quantify benefits due to the absence of baseline data for comparison. Consequently, we contend that the section to ‘Quantifiable and non-quantifiable benefits’ does not apply to the scope of our project and recommend its exclusion from the vision document.

5.6 Estimated costs

The estimated development cost is not relevant due to the project's nature. It is a project for a university class and does not have a budget other than the hours put in by team members. Regarding the number of hours, the estimated cost is 80 hours total per team member. Operational costs will exist but are low. These costs stem from licensing fees and server rental. Management costs are not relevant to the product due to the project's nature.

5.7 Licensing and Installation

The product will require no installation as it will be hosted on a website. Internal security, such as encryption for passwords, will be handled when a user registers an account and will be stored securely in a database on our servers.

If it is fit, the product should support TLS – Transport Layer Security – which will require a certificate from a certified issuer. The additional development requirement is negligible due to services such as Cloudflare.

6. Product Features

No.	Feature	Description
1.	Managing food inventory	Overview of current items the user has in the refrigerator. The necessary sub features are the following: Adding an item. <ul style="list-style-type: none"> Adding one or multiple items Deleting one or multiple items
2.	Cookbook with recipes	Overview of different recipes. The necessary sub features are the following: Adding a title. <ul style="list-style-type: none"> Filtering on relevant metrics Searching by keywords Present a short overview of relevant information such as time, price etc. for each of the listed recipes Opening a given recipe
3.	Creating recipes	Allowing the user to create a recipe of their choice. After completing the recipe, the items are automatically removed from the inventory. The necessary sub features are the following: <ul style="list-style-type: none"> Adding a title, cost etc. Adding ingredients

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

		<ul style="list-style-type: none"> • Adding a picture • Adding a step-by-step description of how to make the recipe
4.	Receive recommended recipes	Recommended recipes based on the user's current pantry inventory.
5.	View shopping lists	<p>Present an overview of the users' current and older lists. The necessary sub features are the following: Brief information of what the list contains, and which list it is.</p> <ul style="list-style-type: none"> • Toggle feature to switch between active and archived shopping lists. • Open and view the different shopping lists
6.	Creating a shopping list	<p>Allow the user to create a new shopping list. After completing the shopping list, the items are automatically added to the inventory. The necessary sub features are the following:</p> <ul style="list-style-type: none"> • Add a title • Add a description to the list • Add items to the list • Remove items from the list
7.	Managing shopping lists	<p>The user can modify and change the shopping list. The necessary sub features are the following:</p> <ul style="list-style-type: none"> • Add one or multiple items • Remove one or multiple items • Keep or remove shopping lists
8.	Joining and leaving a household	<p>The user is automatically a part of a household and can join other households by entering the code from the person that created the household. The necessary sub features are the following:</p> <ul style="list-style-type: none"> • Generates a code for others to join • Add a title for the household • Leave household • Join another household • Delete household • View members
9.	Oauth login	<p>Ability to log in with your Google or Oauth technology.</p> <ul style="list-style-type: none"> • Google login • Linking Oauth accounts

7. Constraints

The main constraint is based on how the application will be used. As the application is mobile-first our main constraint is the available screen real-estate. The product should also be fast enough to be responsive to user input and site download should be fast enough to not affect the user in any way. The design should be uniform and logical to make sense to a user with no previous experience with the product.

8. Quality Ranges

Uniform design: the design should be easy to understand and similar throughout the whole application.

Easy to navigate it should be easy for new users to understand how to get around and interact with the application, as well as efficient for experienced users.

Responsive inputs and actions: the application should not have unreasonably long loading times when interacted with, achieved through optimizing code.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

Automated actions for usability: common tasks should have “automation” features to provide the user with an easy choice which would otherwise be a labor-intensive DIY process. This encapsulates items such as moving newly shopped items from the shopping list to the refrigerator automatically. This would save 100s of potential clicks.

Robustness: the availability of the website is dependent on how it is hosted, and therefore not something which is decided by our development. However, upon errors we should always give the user proper feedback, so they know that the application is not in the expected state.

9. Precedence and Priority

Overview of features and their priority.

No.	Feature	Priority
1.	Manage food inventory	High
2.	Cookbook with recipes	High
3.	Creating recipes	High
4.	Receive recommended recipes	Medium
5.	View shopping lists	High
6.	Creating a shopping list	High
7.	Managing shopping lists	High
8.	Joining and leaving a household	Medium
9.	Oauth login	Medium

10. Other Product Requirements

10.1 Applicable Standards

The product must comply with the TCP/IP protocol stack. It will be platform compliant with any operating system that has access to a modern web browser. Regarding safety standards, the product will follow the newest guidelines and standards for password protection and encryption.

10.2 System Requirements

The only system requirement for the application is that the end device has a modern web browser. There are no hardware or software specific requirements outside of this.

10.3 Performance Requirements

The performance requirements are speed and responsiveness for the end user. There will be data transfers every time the user navigates to a new page or when they alter their data. The data transfer to and from the server needs to be fast and reliable to decrease the user's perceived latency to a minimum, to increase user satisfaction. To provide this, the database must be optimized, and most processing should be server side. This way, all end users will have the same experience regardless of their device's capabilities.

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

11. Documentation Requirements

This section describes the documentation we are going to have in our project. Including user manual, online help, and other resources. This way our application will be easy to use for everyone.

11.1 User Manual

The application will be supported by a user manual to explain the functionalities of the app and guide users through every aspect of the product.

11.2 Online Help

To help the user of the product in different situations throughout normal use of the application we have decided to implement a basic “question mark” system. There will be a question mark icon throughout the application's pages which will describe the basic functionality of said page in normal language, and how the page should be interacted with.

11.3 Installation Guides, Configuration, and Read Me File

The product requires no installation, it is web based. There will however be a link to our Gitlab repository if one wants to host the service on their own. On our Gitlab there will be a complete Wiki page which details how to interact with and use the application. There will also be a README file which will describe the steps to host the application on a server, with information such as the necessary commands and dependencies. It will also provide a short overview of the basic functions of the application. Comments in the code will provide insight into how the generated information is processed and how the general dataflow through the system is implemented.

11.4 Labeling and Packaging

To provide end-users with the best experience possible, the team has decided that we will build our application based on an opensource React component-library called ShadCN/ui. This will lead to a uniform design throughout the entire application. The benefits of this compared to a standard dependency-based approach are manyfold. There are no licensing or copyright notices for using the library as it is free to use. Once the code is written, it is a direct part of our source code, which means that there are no dependencies introduced by the library, just code, which is now ours.

12. A Feature Attributes

In this chapter, we will detail the features to be implemented to our meal-planning tool for purchasing and managing food inventory for households, along with their specific attributes and assigned priorities. These attributes are crucial for evaluating each feature within our system's architecture. Initially, a core set of features has been identified as essential to the program's foundational functionality.

Example for when implementing a new feature:

No.	Feature	Description and reason	Priority
x	“New Feature”	A descriptive text for the new feature and an explanation of why it was necessary to implement.	High/Medium/Low

FRIGO	Version: 1.0
Vision	Date: 26/01/24
frigovision	

12.1 A.1 Status

As development proceeds, updates or additions to the feature set may be necessary to enhance the applications utility or respond to user's feedback. Discussions about potential new features or significant changes to the project scope will be thoroughly documented in this section. The frame keeps track of new features from when they're first suggested (proposed), to when we decide to add them (approved), and then they're finished (incorporated). It's a clear, up-to-date record of why we're adding features and how the project is moving forward.

12.2 A.2 Effort

Each team member is expected to contribute approximately 80 hours to the project. Our team will work hard to finish every part of the project, making sure to plan and follow our schedule. For bigger tasks that are more complicated, we will dedicate more time and work together to get them done.

To maintain a steady workflow and meet our milestones, we have established clear issues and deadlines, and we maintain open communication with our course instructors. This ensures that the entire team has a shared understanding of our objectives and knows what needs to be accomplished at every stage of the project.

12.3 A.3 Stability

In the initial stages of development, it's important to recognize that the feature list may be dynamic, with a high probability of changes. As we lay the groundwork and proceed with development, we anticipate that new insights, user feedback, and innovative solutions may lead to additional or removal of features. Not every feature we initially consider may prove to be as beneficial as originally thought, and better alternatives might arise as we delve deeper into the project.

12.4 A.4 Target Release

We have two releases for the product. The first release is to have minimal viable product target ready by 22th of March. This MVP will be a fully functional version of our product that includes only the core features necessary to meet the primary needs of our users. The second release will be the final product with all the planned features and is set to be released by 26th of April.

12.5 A.5 Assigned To

In our project, features will be allocated to individual or multiple team members based on how complicated they are and how much effort they need to be completed. The project management is streamlined through GitLab, where we employ an issue board to effectively delegate and monitor responsibilities. Each member of our five-person team is entrusted with the ownership of projects elements, ranging from individual tasks to larger, collaborative efforts. We encourage a supportive atmosphere where team members are empowered to seek assistance, ensuring that challenges are met collectively rather than individually.

12.6 A.6 Reason

Every feature included in our program must have a clear purpose. This is to make sure that we avoid cluttering our product with unnecessary elements that could make it complicated to use or navigate. By requiring a solid reason for each feature, we ensure that the product remains user friendly and focused on delivering value.