Lab 2 – LivelyShelfs Product Specification

Colin Thoe

Old Dominion University

CS411W

Dr. Sarah Hosni

March 9, 2025

Version 2

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose

This document is meant to outline the purpose, scope and capabilities of the LivelyShelfs prototype. The document will explain the structure and composition of the prototype while also listing the necessary tools to build it.

1.2 Scope

LivelyShelfs is a web application designed to help users efficiently manage their food inventory and reduce waste. The system is accessible on both desktop and mobile devices, allowing individuals, households, restaurants, and small businesses to track their stored food items, receive expiration alerts, and analyze consumption patterns. By integrating a predictive waste analysis feature, the system notifies users of items nearing expiration and offers suggestions to optimize food usage. Additionally, LivelyShelfs facilitates shared food management through its Shelf Friends feature, enabling multiple users to track shared inventory across different locations.

The LivelyShelfs prototype will include a three-tier architecture, consisting of a frontend interface developed with React-Vite, a backend API layer using Node.js, and a MySQL database for storing user information and food inventory data. The prototype will demonstrate the core features using manufactured test data rather than real data. The Prototype will also not include

many of the more advanced features displayed in the Real World Product, like camera barcode scanning and the web-crawling recipe recommendation API.

1.3 Definitions, Acronyms, and Abbreviations

API: “Application Programming Interface”, allows different software applications to communicate with each other.

Database: An electronic storage system.

Web-Crawler: An application that automatically sifts through the internet looking for data.

1.4 References

[1]C. Thoe, *Lab 1 – LivelyShelfs Product Description*, 1st ed. 2025.

1.5 Overview

This product specification details the LivelyShelfs system, including its purpose, functionalities, hardware and software requirements. It will describe the three tiered system that LivelyShelfs is constructed with, as well as the specific tools and frameworks that it uses.

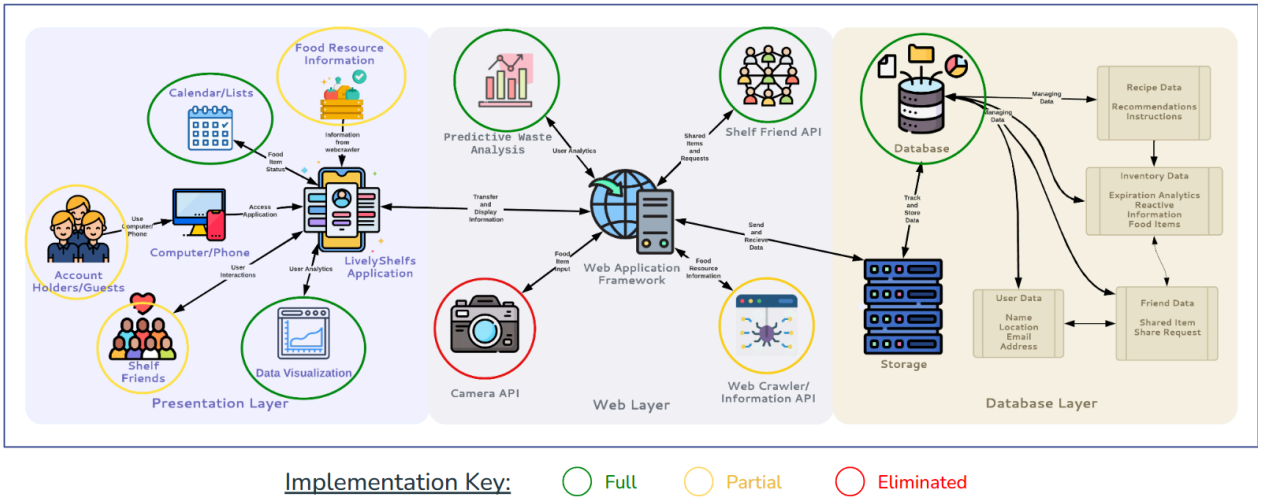
2 Overall Description

2.1 Product Perspective

The architecture of the LivelyShelfs prototype will have three tiers, as shown in Figure 1. The Presentation Layer will be the main area for frontend development, using a React-Vite structure. This is where the user will interact with the product and use its features. The next layer is the Web Layer, where the various API's that back up the frontend will interact with each other and the database. The most notable API is the Shelf Friends API, that manages the user's Shelf Friends and Shared items. The other important feature associated with the Web Layer is the Predictive Waste Analysis feature, which records the user's potentially spoiled items and alerts the user to avoid wasting nearly expired items. The Database Layer will contain the database, which is being developed in MYSQL, and will hold all the information associated with the app, including the user's information and food inventory.

Figure 1

LivelyShelfs Prototype Major Functional Component Diagram



2.1.1 Hardware Interfaces

The hardware requirements will be a computer capable of running Windows, Mac or Linux.

2.1.2 Software Interfaces

The software required for LivelyShelfs will include a React-Vite framework for the frontend, a Node.js framework for the backend, Mocha for unit testing, and a database created using MYSQL.

2.2 Product Functions

As seen in Figure 1, the LivelyShelfs prototype will implement a decent amount of the Real World Product's features, with a focus on demonstrating novel features. Account management will be partially implemented, allowing users to interact with test data to simulate

login and profile management. The add and remove friend functionality will be fully implemented to showcase how users can connect and share inventory. The add and remove member feature, which allows for more complex multi-user management, will not be included in the prototype. The inventory management system will be mostly functional, with some limitations. Users will be able to add and remove items manually, but barcode scanning or camera-based input will not be included unless time allows for implementation. Additionally, the purchase history feature will be omitted, as it is not considered essential for demonstrating the innovative aspects of the prototype. For proactive waste management, the predictive waste analysis and Shelf Friends sharing features will be fully implemented, though the predictive analysis will rely on test data rather than real-time inputs. The recipe recommendations feature will be partially implemented, as the prototype will not include web crawling to fetch external recipes. Lastly, the data visualization feature will also be available in a limited capacity, relying on test data rather than real user statistics.

Table 1

Real World Vs Prototype Product Feature Table

Category	Features	Real World Product	Prototype	Reasoning
Account Management	Login/ Authenticate	Fully Functional	Partially Implemented	Limited time will not be dedicated to basic functionalities
	Location Usage	Fully Functional	Partially Implemented	Limited time will not be dedicated to basic functionalities
	Account Creation / Deletion	Fully Functional	Partially Implemented	Limited time will not be dedicated to basic functionalities
	Add / Remove Friend	Fully Functional	Fully Functional	
	Add / Remove Member	Fully Functional	Eliminated	Limited time will not be dedicated to basic functionalities
Inventory Management	Add / Remove Item	Fully Functional	Partially Implemented	Implement manual input, implement camera if we have time
	Track Item Expiration	Fully Functional	Fully Functional	
	Mark Items Shareable	Fully Functional	Fully Functional	
	Quantity Viewing	Fully Functional	Fully Functional	
	Purchase History	Fully Functional	Eliminated	Limited time and not an innovated feature
	Inventory History	Fully Functional	Eliminated	Limited time and not an innovated feature
Proactive Waste Management	Predictive Waste Analysis	Fully Functional	Fully Functional	Limited test data
	Shelf Friends Sharing	Fully Functional	Fully Functional	
	Recipe Recommendations	Fully Functional	Partially Implemented	Limited time will not be dedicated to web crawler functionalities
	Incentives	Fully Functional	Partially Implemented	Limited time will not allow for full reward
	Data Visualization	Fully Functional	Partially Implemented	Limited test data
	Sharing Analytics	Fully Functional	Partially Implemented	Limited test data

2.3 User Characteristics

Livelyshelfs is mostly focused towards busy household owners and parents who don't have time to constantly manage their kitchen inventory. It will also be useful for people wishing to reduce their ecological and financial footprint by keeping track of wasted food, which will be used to inform the user to make smarter decisions when purchasing commonly wasted items.

2.4 Constraints

N/A

2.5 Assumptions and Dependencies

The dependencies for the LivelyShelfs Prototype will be: React, Vite, Node.js, Mocha, and MYSQL.