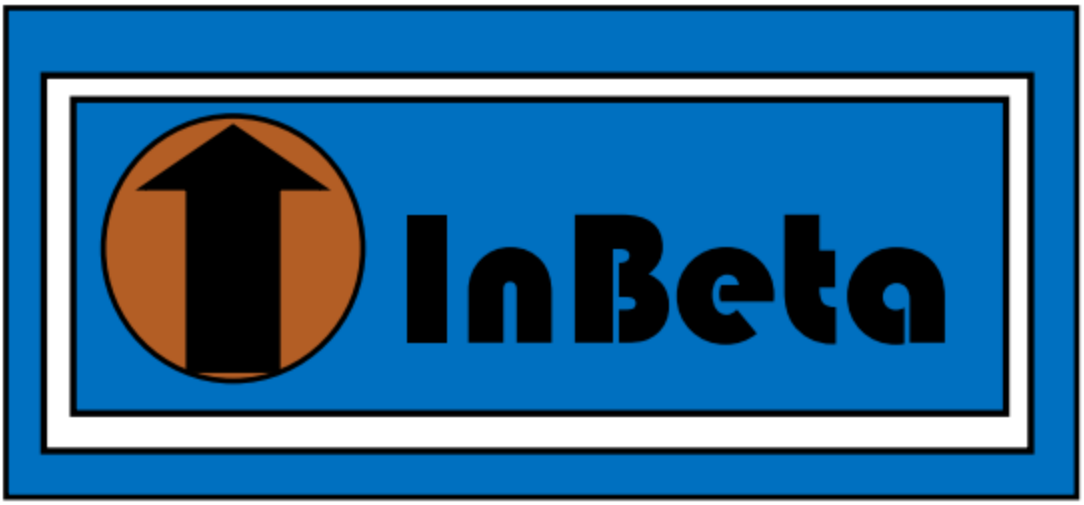


Pantry

By InBeta

Project Plan

Cole Pendergraft, Piper Sheldon Young, Ethan Pressley, Colton Lobdell, Joe Cates
Professor Juan Flores
November 8, 2021

Pantry Application

Pantry is an application that helps food based businesses track and manage food items, as well as prevent unnecessary food waste. Most food based businesses have a massive influx of food items that must be stored for later use, making it a difficult task to keep track of on a day to day basis. The Pantry application will provide a solution to this problem as it gives businesses an efficient method of keeping track of their inventories. Pantry tracks expiration dates and warns users when food items will go bad, allowing businesses to better coordinate when to replenish stock. While Pantry was designed with businesses in mind, any person who wants help managing food storage and waste can benefit from this easy to use application. Pantry will initially be designed for Android, but there is nothing limiting the development of an IOS or desktop application.

Management Plan

Our team took inspiration from the Scrum framework when deciding how we would organize the production of Pantry. In our initial meetings we divided the development of Pantry into different

modules and stages, creating specific tasks that needed to be completed. On a weekly basis we determine what must be completed in that week, using ClickUp to track who is assigned to what tasks. Rather than assign entire development portions to certain team members, each member chooses specific tasks they can complete. This manner of team organization has allowed us to be more productive, and helps each team member to have a more comprehensive understanding of how our application is designed.

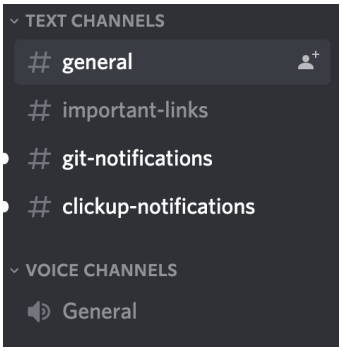
Architecturally, we have separated the development of Pantry into frontend and backend functionality. Each section is then broken into modules, as seen below:

- **User Side (Seen by user)**
    - Pantry List
        - The Pantry List stores a log of all the food items currently inside the user's pantry and the best-by date associated with each food item.
    - Shopping List
        - The Shopping List contains two distinct types of food objects, those that the user has manually written into their shopping list, and the elements within the Pantry List that the user has selected to automatically populate the shopping list when that item expires or is consumed.
    - Input
        - The Input module contains all the functionality that enables the user to enter in food items into either their Pantry List or their Shopping List.
    - Search/Sort/Filter
        - The Search/Sort/Filter module enables the user to rapidly access items stored in either their Pantry List or their Shopping List either by searching it up directly or employing filters or sorting tags that restrict the elements displayed.
    - Menu/Navigation
        - The Menu module handles transitioning between the different model views, Pantry List, Shopping List, and Setting. The Menu will be accessible from anywhere in the app through a button that opens and closes the view.

- **Hidden Side (not seen by user)**
    - Notifications
        - The Notification System will be designed to communicate directly with the Pantry List, reading the associated best-by dates and notifying the user when something is about to go bad. Ideally, this notification will also include tips about how to extend the shelf life of the food item.
    - Settings
        - The Settings System will enable the user to interface directly with modular system features and change the application's operations to suit their liking. This will include such features as selecting whether to use system default best by dates or enable custom best-by date entry.
    - Database

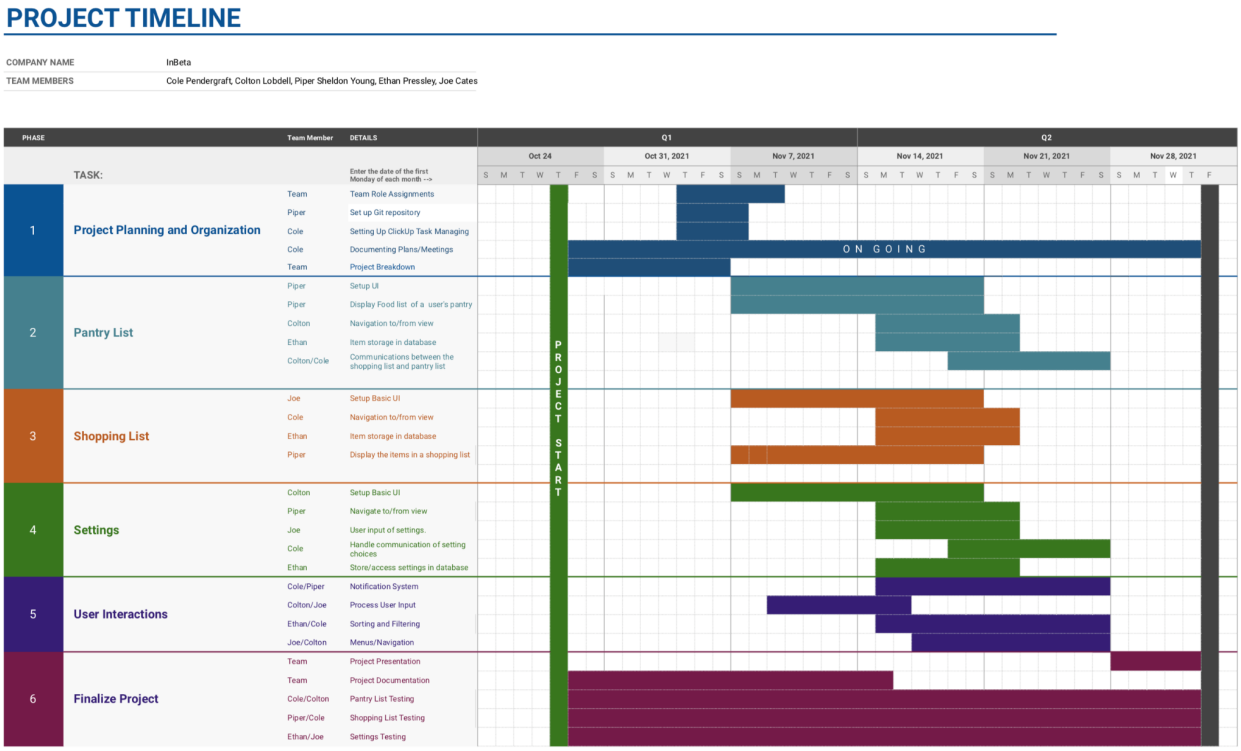
- A database with different tables to store relevant information. Examples of tables include a shopping list table and a pantry list table.

User side development is separated into the different operational views of the application. The Pantry List, Shopping List, Menu, and Settings will all have an associated interactable view. Other modulus such as Input, Search/Sort/Filter, and Notifications provide important functionality for user interaction and application functionality. In our first sprint we prioritized completing the main user interface portions of the application.



As mentioned, our team meets at least once a week, where planning for the next week's sprint takes place. While we do not have daily meetings, we are in constant communication through discord, so any problems or ideas can be addressed to the team at any time. We have created webhooks for both Github and ClickUp, which feed updates from both websites into a discord channel. Github allows us to use version control as each team member develops through their own repository branch. We also use Google Drive to share notes and collaborate on documentation.

Project Timeline



We determined a general project schedule when first approaching the project. Modules and functionality have been roughly mapped to specific time slots. Since we are taking an agile

approach to this project, this schedule is evolving on a weekly basis, depending on the progress made in each sprint.

Building Plan

Each week tasks are determined and distributed to each team member using ClickUp. Evaluation of project progress and development priority is also determined, and evolves our project plan and schedule. Initial priority was given to developing the user interface portion of the application. Providing complete documentation for project requirements and design was also a main focus before continuing with further development. Each team member is actively contributing to the current development stage.

Once the basic user interface elements are complete, more advanced functionality can be implemented. This stage will involve developing backend components such as database storage and searching, user input and interaction, and integration between different user interface views. The user interface will remain a key component for our development focus but will have lower priority in this stage. When all required functionality is complete, we will move to a final stage in which testing, features, and user interaction are at the highest priority. All functionality will be thoroughly tested and optimized to best facilitate a desirable user experience. While debugging and testing are an important component of each stage, they will have higher priority in later stages of development. Frequent meetings with the client throughout each stage will ensure that the product meets expectations, and that team efforts are optimally allocated.

Monitoring and Reporting

Monitoring our work is done through ClickUp, Discord, and through regular meetings. While progress on each main module is associated with our project schedule, ClickUp is our main scheduling tool as it allows for the assigning of tasks and active reporting of progress. All other project monitoring is done through discord and through in person meetings.

Rationale

Our team decided that an agile approach would be the best fit for this project because each of us has similar experience and capabilities when considering developing each module. For this reason, breaking modules into tasks and completing them as a team seemed more appropriate and efficient than assigning a team member to each module. This method of development has also proven to provide each team member with a stronger understanding of each application component. We chose to give priority to developing the user interface because as a mobile application, almost all functionality of Pantry will be linked to user interaction in some way. Because of this it seemed most appropriate to build the user interface first, then implement the remaining functionality on top of it. When designing the Pantry architecture, functionality was the main motivating factor when separating modules and tasks. Each interactable view, as well as each application function, has been placed into its own development section.

When the Pantry application was in its initial stages, we were still unsure of the ideal target market. We presented Pantry as a tool to be used by the average person or family to prevent food waste. We have since realized that Pantry serves primarily as a useful managing and organizing system for an entity with high throughput of food items. To account for this we have changed Pantry's focus from individual users, to food based businesses such as restaurants and grocery stores. These businesses purchase immense quantities of food on a weekly basis, storing it in walk-in fridges, freezers, and pantries. Restaurants must already keep track of inventory and expiration dates for food safety purposes, and Pantry will provide them with an efficient tool for doing so.