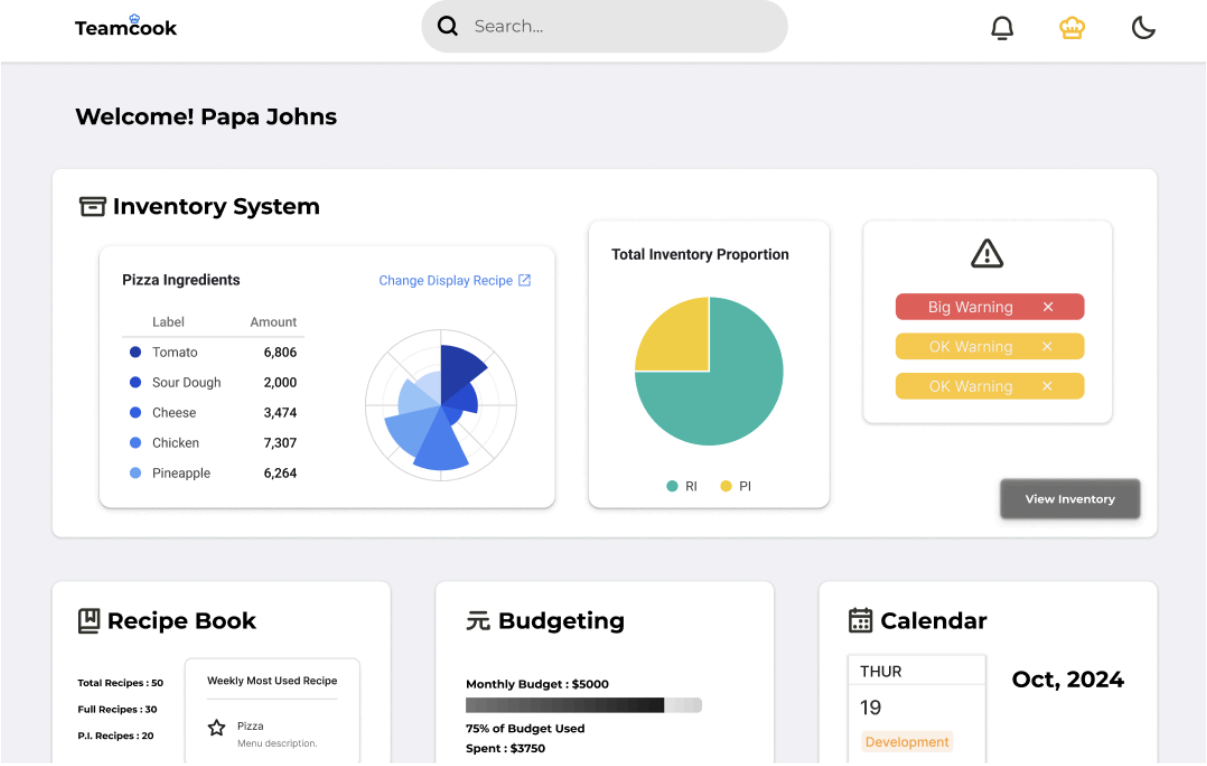
PROJECT: Teamcook

Author: Oscar Chow , zethtang@gmail.com , tatchan@berkeley.edu
Github id: ocWWp, ZethT, tchan102,

TL;DR: WEB APP for Monitoring Cooking Material + Budgeting + Team sharing/Input

With AI integrating



Resources

- ➢ GitHub **(MVP Implemented)** : Here
- ➢ Figma: Here
- ➢ Figma Prototype: Here

One-Line Vision Mission Statement

- Vision:
- Mission:

WHY? (Problem Space)

Inspired by the experience of a friend who is a professional cook, this app seeks to address the pain points faced by kitchen teams:

1. **Material Control:** Difficulty in tracking and controlling material purchases leads to potential loss of money.
2. **Team Conflicts:** Poor communication within the kitchen team results in misunderstandings and errors.
3. **Real-Time Tracking:** A lack of real-time data on material usage and expiration can lead to waste and inefficiency.

WHO?

- People worked in the food industry → **people in the kitchen**
- Restaurant owners/managers who need to have an overview of the kitchen's operation

User Research / Stories

We did user research with a group of kitchen owners and employees to consolidate our problem space, throughout the research, we know that:

Authentication System

- As a Chief, I want to have higher permission levels than cooks, so that I can manage all aspects of the kitchen operations.
- As a Chief, I want to grant cooks limited access to specific recipes, so that they only see the recipes they are assigned to prepare.
- As a Cook, I want to log in and access only the recipes assigned to me, so that I can focus on my tasks without distractions.

Inventory System

- As a Chief, I want to set up a calendar for cooks to prepare dish components or record inventory, so that kitchen activities are well-coordinated.
- As a Chief, I want to check current inventory levels, so that I can make informed decisions about ordering ingredients.
- As a Cook, I want to check the inventory levels, so that I know which ingredients are available for cooking.
- As a Cook, I want to input the quantities of raw or processed ingredients used, so that the inventory remains accurate and up-to-date.

Recipe Book

- As a Chief, I want to input complete recipes and component recipes for each dish, so that cooks have precise instructions for preparation.
- As a Chief, I want to use a measurement conversion function, so that I can easily adjust recipes as needed.
- As a Cook, I want to read the recipes assigned to me, so that I can prepare dishes correctly.

Notification System

- As a Chief, I want to request updates from cooks on specific items, so that I stay informed about kitchen operations.
- As a Cook, I want to respond to the Chief's requests promptly, so that communication is efficient and clear.

Budgeting

- As a Chief, I want to view prices for each ingredient, so that I can manage the kitchen's budget effectively.
- As a Chief, I want to receive prompts when ingredient costs change, so that I can adjust menu prices or order quantities accordingly.
- As a Chief, I want to see visualized data on ingredient costs, usage, and stock trends based on items sold, so that I can make informed financial decisions.

Feature Defining

Feature Prioritize for MVP Oscar Chow

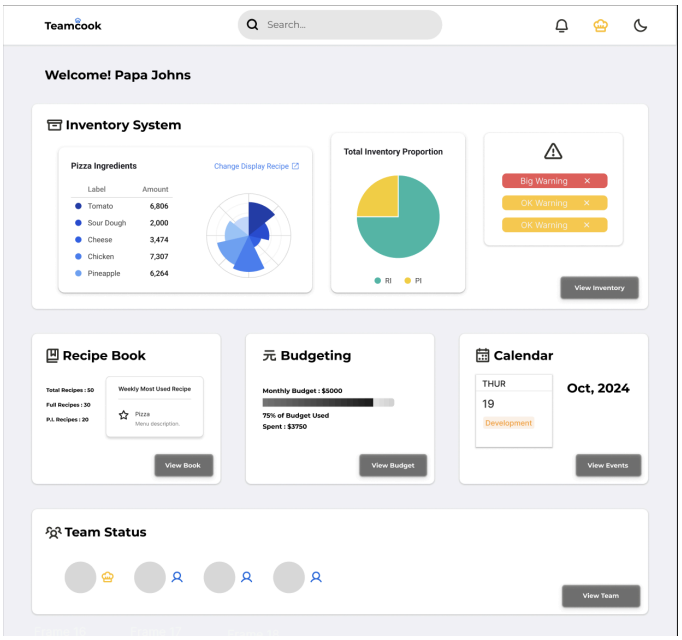
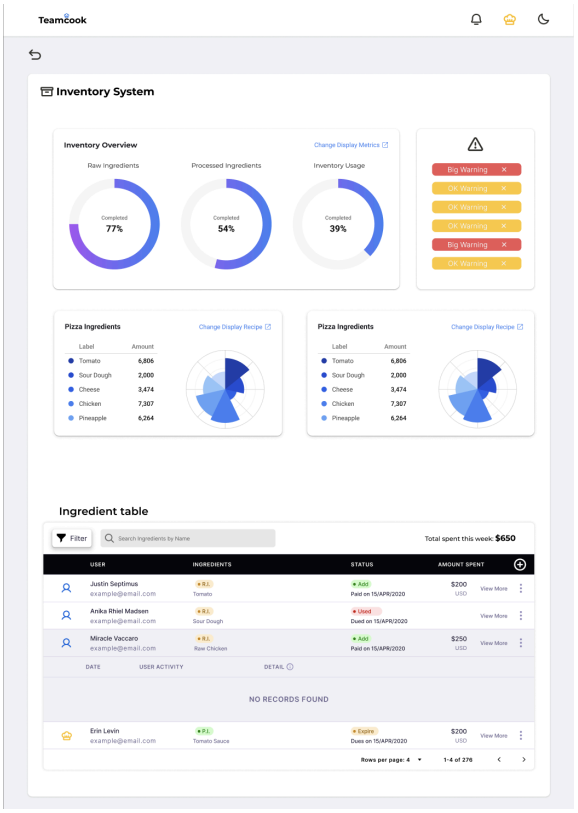
Priority 0	**Home / Landing Page: A dashboard including the overview of all features**
	• **Inventory System**
	  ○ The inventory system will have two types of data:
	  ■ **(Raw) ingredients:** Show the number of fresh bought ingredients
	  ■ **(Processed) ingredients:** Show the number of prepared parts for a dish
	  ○ **Calendar**
	  ■ The calendar feature in this system is designed to facilitate organizational tasks for both the Chief user and the Cooks
	  ■ **Chief User(s)**
	  • **Setup Calendar Alerts**: Users can create calendar events to alert (Cook) about specific tasks, such as preparing ingredients or checking inventory on particular dates on a calendar interface.
	  ■ **Cook User(s)**
	  • **View the calendar:** Users can view the calendar (read-only), and receive alerts when it's close to the particular events.
	  ○ **Storage Tracking**
	  ■ One page interface that allows users to manage and monitor inventory levels efficiently, ensuring that both cooks and the chief have real-time access to the current stock status.
	  ■ **Both User(s)**
	  • **Check Inventory:** Users can access the inventory data to see the current stock levels
	  • **Input Ingredient Quantities**: Users can enter the numbers of ingredients into the inventory dataset
	  ■ **Live-time tracking**
	  • The system keep track on menu sold and compute ingredient used in real-time using APIs
	  ■ **AI Integration (Optional):**
	  • **Restock Prediction:** Predict optimal restock times based on historical usage patterns and economic order quantity principles.
	  • **Demand Analysis:** Identify which materials are being used frequently to help plan future purchases.
	• **Recipe Book**

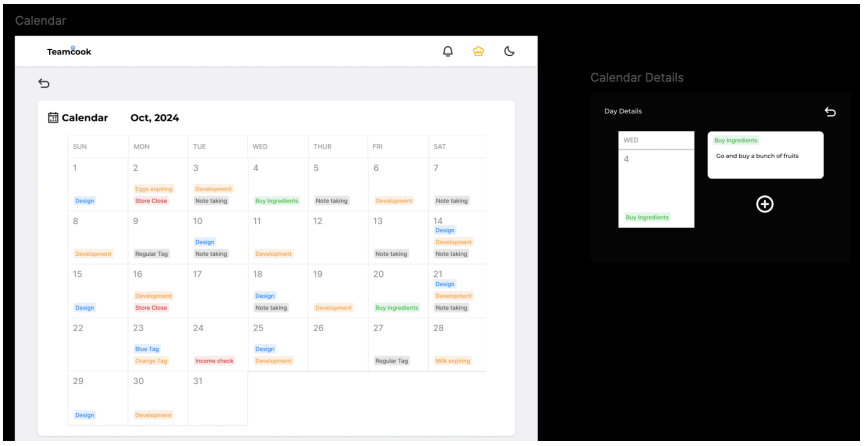
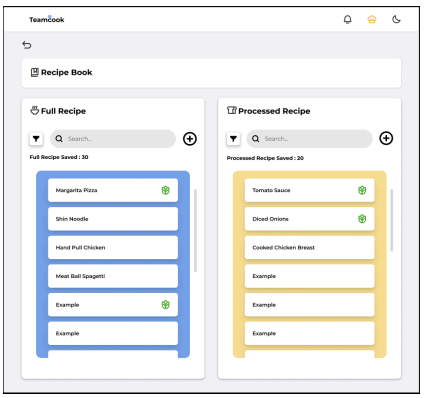
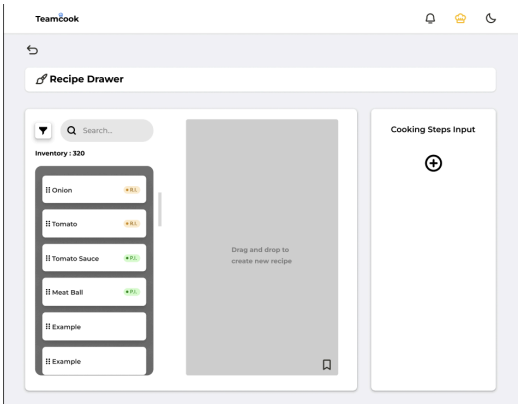
	• The Recipe Book feature is designed to centralize and manage recipes, ensuring that both the Chief user and the cooks have access to the necessary information to prepare dishes accurately and efficiently
	  ○ **Recipe Type:**
	  ○ There are two type of recipes, complete recipes and processed recipes
	  ■ **Complete recipes**: A recipes for a dish (raw ingredients + processed ingredients)
	  ■ **PI recipes**: A recipe for the processed ingredients (how much raw ingredients used)
	  ○ **The Book**
	  ○ Both User(s) can review saved recipes here
	  ○ **Recipe Builder:**
	  ○ This feature is **only** for (Chief) users to build-up their recipe, features allows for the flexible use of the same components in multiple recipes.
	  ■ **Recipe Building**
	  • Build new dish by selecting Raw/Processed Ingredients
	  • Drag-and-Drop builder
	  ○ Can drag and drop PI recipes into a new dish.
	  ■ **PI Recipe Building**
	  • Build new PI recipe by selecting Raw Ingredients
	  • Reusable PI recipes
	  ○ Can create and reuse saved PI recipes
	  ■ **Measurement Conversion feature**
	  • A measurement conversion feature, allowing the (Chief) users to convert between different units along with the recipe (e.g., how many cups of rice equal how many milliliters).
	• **Budgeting**
	  ○ The Budgeting feature is designed to help users manage costs, track spending, and optimize inventory based on financial constraints and sales data.
	  ○ **Trend Visualization**
	  ■ **Ingredient usage trend**: Display graphs showing ingredient usage patterns + keep tacks on the cost of Ingredient Cost Trends
	  ■ **Stock Trend Analysis**:Visualize stock levels, highlighting shortages and excesses based on sales data.
	  ○ **Spending Tracker:** Record all expenditures related to cooking materials. Users can view and analyze spending over different periods (weekly, monthly).

	○ **Budget Reports:** Automatically generate budget reports summarizing total spending, categorized by material type.
Priority 1	**Login / User identification** User can register as either chief or cook to join a restaurant, or create a new restaurant ● **Chief:** When login as a chief, user can have full access and control + administrative settings ● **Cook:** When login as a cook, user can have Role-Specific access + granted by the chief ● **Creating a new restaurant:** User can create a restaurant (automatically a chief) and other users can join as a cook
Priority 2	**Inventory System Feature AI Integration:** ● **Restock Prediction:** Predict optimal restock times based on historical usage patterns and economic order quantity principles. ● **Demand Analysis:** Identify which materials are being used frequently to help plan future purchases. **Budgeting Feature AI Integration:** ○ **Budget Optimization:** Automatically update inventory suggestions based on user inputted budgets. Recommend materials and quantities that align with the current budget constraints. ○ **Inventory Recommendations:** Utilize AI to provide recommendations on material purchases that optimize inventory levels without exceeding budget. ○ **Pricing Recommendations:** Prompt for raising/lower the price base on the ingredients cost
Edge Cases	

UI/UX Design (subject to change)

- ➢ Figma: Here
- ➢ Figma Prototype: Here

Home Page	
Inventory Page	

Calendar Page	
Recipe Book Page	 
Budget Page	

Metrics

Further Define

MVP TDD

tatchan@berkeley.edu

- Tech stack:
    - Web APP
        - FE
            - Framework/Library: React (for building interactive user interfaces)
            - UI Framework: CoreUI (for pre-built UI components)
            - Styling: SCSS, CSS Modules, or styled-components
            - Build Tool: Vite (for bundling and building)
        - BE
            - Toast POS API (https://doc.toasttab.com/doc/devguide/apiOverview.html)
            - SQL
    - AI:
        - GPT API???

Object we're using in this project

QTY:
Unit: str
Qty: float

STOCK:
Name: str
Purchase date: timestamp
Expiry date: timestamp
cost: float
quantity: QTY

RI (Raw Ingredient):
Qty: sum of QTY of stock
unit: str (like cups, mL)
Name: str
categories: str
Stocks: Array of STOCK

PI (Processed Ingredient):
Name:
Raw ingredients:???
Create date:timestamp
Expiry date:timestamp
Cost: sum of the RI cost

RECIPE:
Type: Processed/ Full Recipe
Name: str
Create time: timestamp
Steps: str
Processed Ingredient:PI
Raw Ingredient(not include in Processed): RI

RESTAURANT:
Name: str
Address: str
Phone: str
Recipes: RECIPE

USER:
Login_id: str
Pw: ?
name: str
Post:str

EVENT:
Name:
Who created:USER
Restaurant: RESTAURANT
Time: Timestamp

Page Content

Home Page

- **Design: One Column roll-down**
  - Log-in/register button
  - Nav bar

DashBoard:
- Inventory
- Budgeting
- Notifcation

Inventory Page
- **Sub-page**
  - **Calendar:**

Reicpe Book
- **Design**: Two columns
  - Frist column: Recipe
  - Second column: PI recipe
- **UI features**:
  - **Create recipe**: User can create a new (PI)recipe
  - **Drag-drop**:user can drag item form PI recipe
  - **Edit:** Each recipe has an edit button

Budgeting

Team
- **Design:** One page
- **Info section:**
  - Restaurant info
  - Chief contacts
  - Cooks contacts