# Acme Food Bank Inventory Tracking System

*First Draft*

Acme Corporation

**T-Nine**

Yaru Gao;

Dante Meyer;

# *Table of Contents*

## I. Introduction

The Acme Food Bank faces significant challenges in managing frequent and varied donations of food and hygiene products. To address these challenges, our project aims to develop a reliable inventory tracking system. This system will improve the management of donations, ensuring accurate tracking of quantities and expiration dates, and enhance overall operational efficiency.

## II. Background and Related Work

The Acme Food Bank employs multiple workers who manage a large number of assorted products, tracking donations and stock to ensure they don' t run out and can distribute products evenly. Manual data management and entry are cumbersome and tedious due to the volume and similarity of the product entries. This manual process is not scalable and requires significant effort to process data. Implementing automation and user-friendly stock management would enable more efficient tracking, allowing associates to edit and monitor products in bulk.

This solution requires a frontend for multiple users to easily interact with a centralized set of data, so we will be using a **Python/Flask/HTML/SQL** stack to develop the project, by providing a web application for users to interact with the backend, storing inventory data securely in SQL. The users of the application need to be able to quickly and accurately be able to track quantities and attributes of catalogued products and easily produce statistical analyses of sold products. The frontend should be intuitive and communicate responsively with the hosted database, providing the right amount of abstraction to prevent the user from having to manually manage the database.

Fulfilling these objectives, this project will fall under the categories of web development and database management. A centralized database would allow for all users to operate in sync with the same database without worry of inconsistencies, and a web frontend would be simple to iterate on and very portable. Because of this, most inventory tracking programs use similar tech stacks. We'll be using SQLite [1] as the database backend, as it is an open source and in the public domain, and it has a reliable and well-documented Python module [2] that can directly be used with Flask. Unlike backends like MySQL, SQLite doesn't require a perpetually running database server, and instead utilizes embedded database files.

A prominent example of state-of-the-art inventory tracking is Square, responsible for processing roughly 4 billion sales transactions in 2023 [3]. Square is provided as a SaaS (Software-as-a-Service) business management platform, handling statistics, inventory tracking, and online transactions, using a streamlined frontend that can also be accessed via an API. There are also open-source, locally hostable alternatives, such as Iventree, which can be deployed on a local machine. For this project, our software will be more similar to the latter, to allow the food bank to fulfill its specific needs without relying on external servers. It will also provide access to data points specific to food and hygiene, such as expiration dates, and provide simple ways to catalog and analyze that data.

## III. Project Overview

### Goals and Desired Outcomes:

Our goal is to build a comprehensive inventory management solution for Acme Corp's food bank. The system will address the key challenges of managing frequent and varied donations of food and hygiene products, tracking quantities and expiration dates, and generating efficient data reports from multiple data sources.

To achieve this, we will use a combination of Python, Flask, HTML, and SQLite, as aforementioned. Flask will be utilized to facilitate interaction between backend functions and the HTML-based user interface. Python will be utilized to handle backend functionalities, including database management and data processing, while HTML will be used to create a user-friendly web interface. SQLite will serve as the database management system, providing solution for storing and retrieving data.

The final outcome will feature a web interface where users can view inventory information, donation details and donor information, and receive notifications about upcoming expiration dates. Additionally, the system will allow users to check other key information such as tracking inventory quantity of any specific product and the respective expiration date, viewing related transactions, and managing inventory inflow and outflow details. Moreover, the system will enable security measures, such as accounts-based access control with two-factor authentication (2FA) and TLS encryption.

### Functionality design:

In detail, the system will feature the following functionalities:

*Dashboard*: A user-friendly dashboard providing an overview of inventory levels, recent donations, and upcoming expirations. The dashboard will serve as the control center for all inventory management activities. It will display key metrics using graphs, charts, and tables. Users will be able to quickly see the status of the inventory, including total items in stock, categories of items, item location, brand name, purchase date, and recipient’s info and demand. The dashboard will also feature a search functionality to allow users to easily find specific items or categories. Recent donations will be highlighted, showing donor names, donation dates, and items donated.

*Account Data Storage*: Users will be able to register for new accounts through a secure registration process. This functionality will involve storing account data in the SQLite database with encrypted passwords and other sensitive information.

*User Authentication*: Secure login with 2FA to protect user accounts and sensitive information. The system will use TLS encryption to ensure secure data transmission.

*Inventory Management*: Functions for adding new donations, updating existing inventory, and marking or removing items that have been distributed or expired. Using Python, the system will support real-time updates and implement full CRUD operations to maintain accurate and current inventory records.

*Notifications*: Alerts and notifications for items nearing their expiration dates to ensure timely distribution. Upcoming expirations will be highlighted with marks (e.g., color-coded alerts) to draw attention to items that need to be distributed soon.

*Reporting*: Generation of customized reports based on data from multiple entities, with options to export reports in various formats (e.g., PDF, CSV). The system will record and integrate related data about inventory from the database, enabling it to generate reports required for monthly analysis.

*Recipient Orders*: Allow recipients to send their demand requests to the system similar to purchase orders. This information will be stored in the database for transaction processing and data analysis.

*Predictive Analytics*: Use and analyze existing inventory data from the database to predict inventory patterns, trends, and needs within a specific time frame. This will help the food bank anticipate future demands and manage resources more effectively.

By implementing these functionalities, the project will provide Acme Corp's food bank with a powerful tool to manage inventories more efficiently and enhance operational effectiveness. This solution will address the current limitations of existing inventory management systems and offer an approach to meet the specific needs of the food bank.

## IV. Client and Stakeholder Identification and Preferences

The primary client for this project is Acme Corp, specifically its food bank division. The food bank staff need a straightforward, user-friendly system to manage donations, track inventory, and monitor expiration dates. They require the system to work on the tablet they recently purchased and prefer a mobile app or web interface for easy access.

Other stakeholders include donors, recipients, and IT support staff. Donors need to feel confident that their contributions are managed efficiently. Recipients rely on timely and unexpired products from the food bank, so improved inventory management directly benefits them. IT support staff need the system to be secure and easy to maintain. Features like two-factor authentication and TLS encryption will help protect sensitive information and ensure that only authorized users have access to the system.

## V. Glossary

2FA (Two-Factor Authentication): An extra layer of security used to ensure that people trying to gain access to an online account are who they say they are.

TLS (Transport Layer Security): A cryptographic protocol designed to provide communications security over a computer network.

Web/Mobile App Development: The process of creating applications for mobile devices and web platforms.

User Interface (UI): The space where interactions between humans and machines occur, aimed at effective operation and control of the machine.

CRUD Operations: CRUD: An acronym for Create, Read, Update, Delete, representing the four basic operations performed on database records or data in a software application.

## VI. References

[1] SQLite, "SQLite Copyright," 5 May 2024. [Online]. Available: https://www.sqlite.org/copyright.html. [Accessed 17 May 2024].

[2] Python Software Foundation, "sqlite3 — DB-API 2.0 interface for SQLite databases — Python 3.8.2 documentation," 2024. [Online]. Available: https://docs.python.org/3/library/sqlite3.html.

[3] Square, "squareup.com," 2024. [Online]. Available: https://squareup.com/us/en/about.