**Data Intelligence Low Code Inventory App for Personal and Corporate Settings**

**WasteDrop**

**Part 2 – Introduction and Design**

**Dr. Walker**

**Advanced Software Engineering**

**September 14, 2023**



Table of Contents

[Section 1 – Salient Characteristics 5](#_Toc148778330)

[Section 2 – Description of Salient Characteristics 5](#_Toc148778331)

[2.1 5](#_Toc148778332)

[2.2 6](#_Toc148778333)

[2.3 6](#_Toc148778334)

[Section 3 – Contextual Issues and Constraints 7](#_Toc148778335)

[3.1 7](#_Toc148778336)

[Section 4 – Proposed Schedule 8](#_Toc148778337)

[4.1.1 & 4.1.2 8](#_Toc148778338)

[(US001) Sign-up functionality 8](#_Toc148778339)

[(US002) Login Functionality 8](#_Toc148778340)

[(US003) Binz Creation 9](#_Toc148778341)

[(US004) Add Items to Binz Functionality 9](#_Toc148778342)

[(US005) “Consume” Binz Items Functionality 9](#_Toc148778343)

[(US006) UI enhancements 10](#_Toc148778344)

[(US007) Add Expiry Alerts 10](#_Toc148778345)

[(US008) User Personalized Alert Times 11](#_Toc148778346)

[(US009) Top Binz Items List 11](#_Toc148778347)

[(US010) Low Code User Friendly Query 12](#_Toc148778348)

[(US011) Logging Out 12](#_Toc148778349)

[(US012) Session State Handling 13](#_Toc148778350)

[(US0013) Create DB schema Design. 13](#_Toc148778351)

[(US0014) Page switching functionality. 14](#_Toc148778352)

[(US015) “Use one” functionality for items. 14](#_Toc148778353)

[(US016) “Use many” functionality for items 14](#_Toc148778354)

[(US017) Historical list 15](#_Toc148778355)

[(US018) Pre final release code tests 15](#_Toc148778356)

[(US019) Password Hashing 15](#_Toc148778357)

[4.2 – User Story Allocation 16](#_Toc148778358)

[4.3 – Sprint Period 16](#_Toc148778359)

[4.4 – Sprint Backlog Allocation 16](#_Toc148778360)

[4.5 – Burndown Chart 17](#_Toc148778361)

[Section 5 – Team & Roles 18](#_Toc148778362)

[Team Members 18](#_Toc148778363)

[Samuel Muvdi 18](#_Toc148778364)

[Ed 18](#_Toc148778365)

[Abel 18](#_Toc148778366)

[David Rice 19](#_Toc148778367)

[Ryan 19](#_Toc148778368)

[Section 6 – Project Design 20](#_Toc148778369)

[6.1 – Design Defined 20](#_Toc148778370)

[6.2 – Team Design Process 20](#_Toc148778371)

[6.2 – Graphical Design of WasteDrop 20](#_Toc148778372)

[Section 7 – Rationale Management 21](#_Toc148778373)

[7.1 – Issues Addressed 21](#_Toc148778374)

[7.2 – Alternatives 21](#_Toc148778375)

[7.3 – Decisions Made 21](#_Toc148778376)

[7.4 – Guiding Criteria 21](#_Toc148778377)

[7.5 – Developer Discourse 21](#_Toc148778378)

[Section 8 – Sources 22](#_Toc148778379)

Section 1 – Salient Characteristics

The salient characteristics of the customers will be individuals or companies looking to keep an organized and smart food inventory collection on their smart phones and computers where they can track all items from pantry items to freezer to fridge foods. This allows users to reduce food costs as well as have a positive environmental and social impact by reducing food waste which otherwise could have been donated or eaten before its expiration date. Our users will range from family households trying to manage multiple buyers putting in and taking out things from the fridge all the time to office managers who need to order large quantities of food for office events or just for their employees to restaurants, bakeries, catering companies, cafeterias, to food court venues and food trucks that just want to have a modern, visual, smart, and simple way of managing their food inventories without having to use the traditional paper and pencil.

# Section 2 – Description of Salient Characteristics

2.1

The background of WasteDrop stems from food waste, a common issue that is faced in many countries and areas around the world. According to the Food and Agriculture Organization of the United Nations, approximately one-third of all food produced for human consumption is lost or wasted (citation needed?). This significant waste can be partially attributed to consumers grappling with the management of expiration dates, as well as a lack of accessible information regarding optimal methods for storing and utilizing perishable items.

## 2.2

WasteDrop will stand out from other apps and tools in the market because we will be marketing and catering primarily to food inventories whether it’s dry, wet, or frozen food. Second, we will deploy our project online through the web and eventually as an app that can be downloaded and saved. Third, we will be using low-code user intelligence queries which will let the users get insights on their inventory without even having to know how to code. Finally, our app will implement sending notifications whenever food is expiring soon or otherwise according to the users’ preferences and settings.

## 2.3

WasteDrop is designed with the primary goal of substantially reducing food waste by assisting individuals in effectively handling their perishable items. Additionally, the application can act as a valuable tool for cost-conscious consumers, helping them cut unnecessary expenses. By offering essential insights, reminders concerning expiration dates, practical tips on food storage and usage, and streamlined inventory management, WasteDrop promotes cultural awareness and education, encouraging responsible food consumption practices. In this way, WasteDrop holds the potential to bring about a significant positive impact on both consumer behavior and the environmental repercussions linked to food waste.

# Section 3 – Contextual Issues and Constraints

## 3.1

Depending on user-provided input, printed expiration dates may pose potential issues. To enhance the application's efficacy, it may be necessary to establish a system for validating or cross-referencing expiration dates specific to various products. Additionally, compliance with data protection laws is a crucial external factor. Users should have confidence that their personal information and food inventory details are handled securely.

# Section 4 – Proposed Schedule

## 4.1.1 & 4.1.2

### (US001) Sign-up functionality

**Description**

As a user, I want to be able to sign up for an account on WasteDrop so that I may login and use the app.

**Criteria**

Users will be able to create accounts on WasteDrop, that will contain their first name, last name, email address, and a hashed password. This sign-up functionality will allow users to have accounts which they can then use to access WasteDrop.

### (US002) Login Functionality

**Description**

As a user, I want to be able to login to the account I created on WasteDrop and access their and only their accounts, so that I can access my own account

**Criteria**

Users will be able to login to the accounts they made on WasteDrop and retrieve their data.

### (US003) Binz Creation

**Description**

As a user, I want to be able to create a Binz (unit of storage for inventory, ex: fridge\_house, fridge\_garage, cabinet\_1) so that I can access my items.

**Criteria**

A user can create a Binz that is saved to their account, and they can access it freely.

### (US004) Add Items to Binz Functionality

**Description**

As a user, I want to be able to add items to my Binz so that I can track my inventory.

**Criteria**

Binz will have a feature where the user can add items with their information.

### (US005) “Consume” Binz Items Functionality

**Description**

As a user, I want to be able to “Consume” (delete from my binz) items in my Binz, so that items that no longer are in my real-life inventory can be removed from my Binz.

**Criteria**

Ability to take things off from Binz inventory if they were eaten or thrown away etc.…

### (US006) UI enhancements

**Description**

As a UI developer, I want users to be able to have a simple yet modern look for WasteDrop and be able to enjoy how the app looks so they are more inclined to continue using it.

**Criteria**

The app is beautified the most it can be so that users see it as more likeable and pretty.

### (US007) Add Expiry Alerts

**Description**

As a user, I want to be able to be alerted any time my Binz items are about to expire so that I may take proper action.

**Criteria**

WasteDrop lets users know when their items are about to expire (users add expiry dates using the sell by and best by dates on food, if for some reason the sell by or expiry dates are not accurate on the food, maybe it’s just the date the food should be sold by and not necessarily expired, the user should choose a given date to be the expiry date)

### (US008) User Personalized Alert Times

**Description**

As a user, I want to pick how many days before an item expires to be alerted that the item is expiring so that I can have enough time to take proper action.

**Criteria**

The user can pick how long before an item is about to expire to be notified that an item is expiring soon.

### (US009) Top Binz Items List

**Description**

As a user, I want my most added items to be added to a short list so that I may be able to see how many times I have added it and which items are most frequent in my Binz.

**Criteria**

Users have a top items list where they can see what items they added the most and how many times they have added it.

### (US010) Low Code User Friendly Query

**Description**

As a user, I want to be able to create my own searches (queries) on my Binz to get information on my items, like which items expire tomorrow, or how many items I have currently in my Binz, so that users who aren’t experienced in data analytics can still be able to search their Binz and find insights on it.

**Criteria**

Users will have the option to make no code / low code queries using drop down lists to get insights on their Binz data.

### (US011) Logging Out

**Description**

As a user, I want to be able to sign out of my account so that, if someone were to have access to the same device as me, they would not be able to access the information on my account without my login credentials.

**Criteria**

Users will have the option to log out of their account when they are finished using the app, closing their session and ensuring security of their account information.

### (US012) Session State Handling

**Description**

As a user, I do not want to have to repeatedly enter the same information to continue accessing my account. I want to have indefinite access to my account until if/when I choose to log out.

**Criteria**

Users’ account data will be stored while they use browse the application by maintaining the data on the server-side to persist across many requests from a particular client. Once the user closes their session, this temporary data is deleted.

## (US0013) Create DB schema Design.

**Description**

As a developer, I want to be able to design a functional and normalized database schema so that the users can have a smooth experience when querying or inserting.

**Acceptance criteria**

A working and normalized database schema design is created.

## (US0014) Page switching functionality.

**Description**

As a user, I want to be able to switch pages whenever I log in so that I may access my items.

**Acceptance** **criteria**:

When users log in or need to access items, the switch page functionality works correctly.

## (US015) “Use one” functionality for items.

**Description**

As a user, I want to be able to reduce my inventory quantity by 1 for a specific item each time by a button so that I may track updated quantity.

**Acceptance criteria**

A button will allow for the reduction of 1 for inventory quantity for item.

## (US016) “Use many” functionality for items

**Description**

As a user, I want to be able to reduce the count of my inventory quantity by a specified amount so that I may track my consumption and updated inventory.

**Acceptance criteria**

Users will be able to reduce inventory count quantity of an item by more than just 1

## (US017) Historical list

**Description**

As a user, I want to be able to see all items I ever added so that I may keep track of past inventory for analysis.

**Acceptance criteria**

Trigger moves consumed items to historical items table.

## (US018) Pre final release code tests

**Description**

As a developer, I want to do full app testing to pick up any bugs before the final release so that users wont experience any last-minute issues.

**Acceptance criteria**

No final bugs are found in last check before final code deployment on release.

## (US019) Password Hashing

**Description**

As a developer, I want to hash passwords before they are inserted into the database so that users’ data is protected.

**Acceptance Criteria**

User passwords are hashed before inserted to the database.

## 4.2 – User Story Allocation

Team developers will have their user stories assigned to them based on their past coding experiences, developing experiences, and what they have been able to learn through research this semester. Each developer will be assigned a reasonable amount of stories each sprint to ensure that the project moves smoothly and on time.

In the case that the project needs a new story for more functionalities or additions, a new story will be created for it and then will be added to the next sprint or started in the current sprint and if not completed, rolled over to the next sprint to complete. This will allow for all user stories to be tracked and the ones that are higher priority to be done as soon as possible.

## 4.3 – Sprint Period

Each sprint will last for 2 weeks. Each sprint will start on Wednesdays and end on Wednesdays. Sprints will be tracked via Trello.

## 4.4 – Sprint Backlog Allocation

The allocation of User Backlogs to sprints will be as follows:

* Sprint 1 9/13-9/27: (US001), (US002), (US011), (US013), (US014), (US019)
* Sprint 2 9/27-10/11: (US003), (US004), (US005), (US006), (US007)
* Sprint 3 10/11-10/25: (US008), (US009)
* Sprint 4 10/25-11/8: (US010), (US012), (US015)
* Sprint 5 11/8-11/22 (US016), (US017)
* Sprint 6 11/22-12/6 (US018)

## 4.5 – Burndown Chart

The burndown chart for the schedule that was outline in section 4.4 is shown below:

# Section 5 – Team & Roles

**Team #7 WasteDrop Developers**

## Team Members

Samuel Muvdi

* Lead developer
* BA
* Project Manager
* Oversee and develop code on full stack.
* Ensure data and Python code and scripts are working as expected.
* Create new user stories & approve tentative stories.

Ed

* Developer / Backend Data-Ops
* Assist with creation and testing of Python scripts and Postgres implementation.
* Ensure Python scripts are handling data as expected.

Abel

* Developer
* Assist with python coding to ensure functionalities are being properly created, implemented, and documented.
* Additionally, assist with testing of various components of the application.

### David Rice

* Scrum Master
* Maintaining Trello board and overseeing documentation of each rollout.
* Documenting project progress and completion of deliverables.

### Ryan

* Developer
* Help with python coding and making sure functionalities are being added and tested.

# Section 6 – Project Design

## 6.1 – Design Defined

Define design //using scholarly literature to support your definition; following APA citation. NOTE FOR US: BE SURE TO INCLUDE SOURCE CITATIONS IN SECTION 8 - SOURCES

## 6.2 – Team Design Process

Introduce your design process – the process invoked by your team

## 6.2 – Graphical Design of WasteDrop

Demonstrate the design graphically

* Use case Diagrams
* Activity Diagrams (if applicable)
* Class Diagrams
* Sequence diagrams

# Section 7 – Rationale Management

be sure to provide a justification for design decisions made during the project (hint\* - take notes as you are going through the process, journal even…this will help you with this portion of the project). NOTE FOR US: DO NOT FILL ANYTHING IN HERE, JUST PUTTING THE GENERAL DIRECTIONS FOR SECTION 7 HERE

## 7.1 – Issues Addressed

The issues that were addressed

## 7.2 – Alternatives

The alternatives that were considered

## 7.3 – Decisions Made

The decisions that were made to resolve the issues

## 7.4 – Guiding Criteria

The criteria used to guide decisions

## 7.5 – Developer Discourse

The debate developers went through to reach a decision

# Section 8 – Sources