

Charitaste: Reducing food waste

The goal of this document is to describe a digital solution that allows retailer stores to allocate their excess expiring consumer items, mainly food from the FMCG category to charities to lower food waste.

Authors: Alma Maka Eszter, Bäck András, Klucsik Krisztián Pál, Hermecz Vajk

Brief

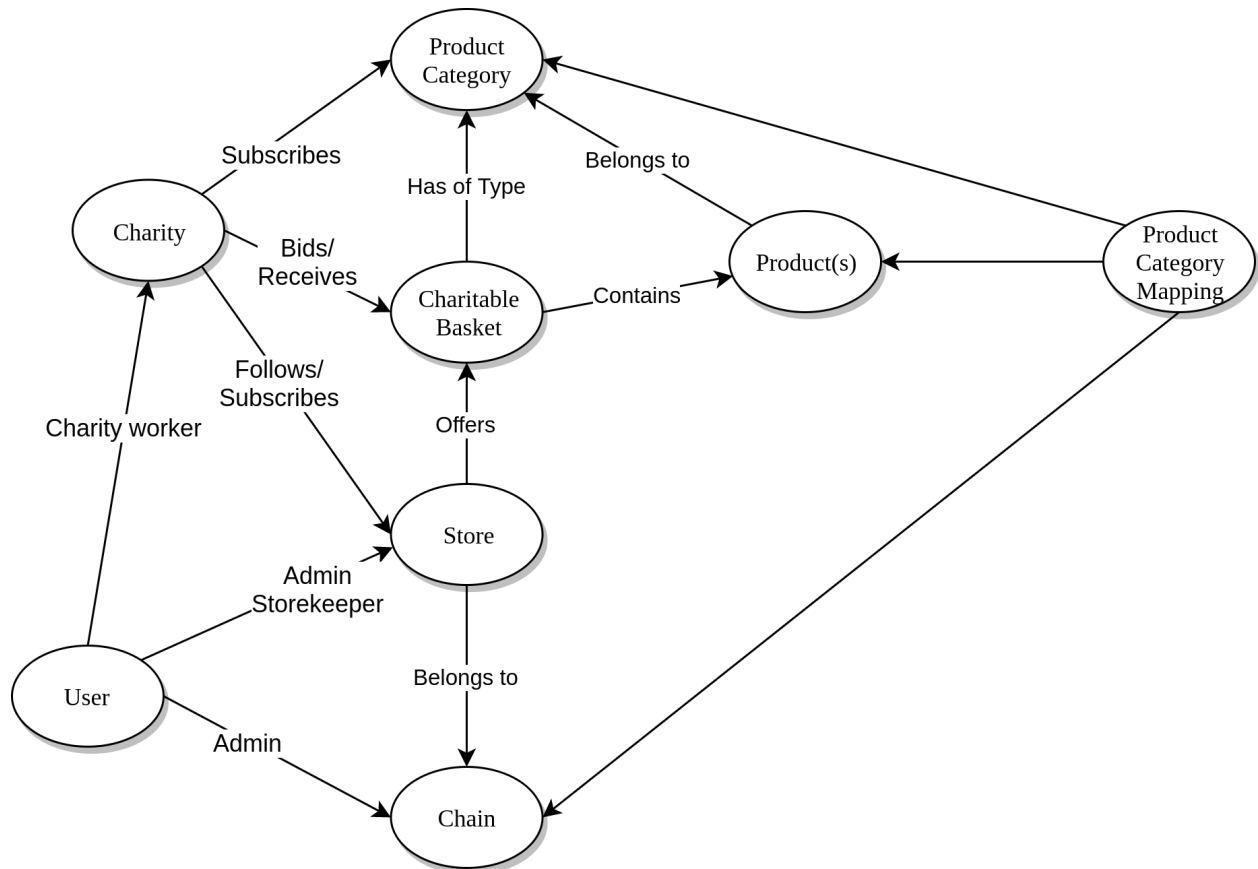
The initial problem we got introduced to, is that Tesco produces 26k tons of food waste each year. In some countries they managed to decrease this amount by 82%. Our goal was to ideate and design a system that would allow achieving such a state.

Preface

During the design process, we realized that a single Retailer is only representing a small portion of the market (<20%). Marketing and advertising a system to raise awareness of charities about this option can be expensive. It does not make sense to have such a system per retailer. These systems would actually replicate the same functionality. The base unit of organization should not be a retailer, but a locality, where producers and consumers of charitable products can find each other, within close distances. A reasonable boundary for such a system would be boundaries of countries, as linguistic, cultural, and juridical boundaries usually align with these. Such boundaries strongly influence the operations of the system. As a result of this thinking, we decided not to design the system for the needs of Tesco solely, but to come up with a platform that multiple retailer chains/brands/stores can use. This change in approach, while keeps the charity side of the system the same, also requires a higher level of configurability to the system.

Problem Specification

Domain model



Entities

Chain: A retailer. Can have from a single up to many stores. The main goal of this level is to allow for multi-store retailers to administer their stores, product mapping data, and integration to the system. We assume, that all stores in a given chain are managing the same set of products, aka. product codes within a chain are unique across all stores.

Store: A single physical location managing consumer goods, having excess consumer goods that could be donated to charities.

Charity: A non-profit organization seeking food/consumer goods donations.

Charitable basket: A collection of consumer goods up for donation.

Product Category Mapping: Chain level assignment of the chain's product codes to product categories.

Product Category: We specify three orthogonal categorizations, see later. These categories are used both by the chains/stores to identify the products, and charities to seek for them.

Categorization

We identified three main, orthogonal categories for consumer goods. They are detailed hereinafter.

Handling category

Defines what handling or permissions a specific product needs.

On the store side this allows for marking products what special handling they require, and on the other side, the charities can specify if they have the required equipment / knowhow / authorization / legal permission to do so.

E.g.:

- Normal
- Cooling is required
- Freezing is required
- Permission X is required (e.g.: for drugs, hazardous materials)

Reuse category

Specifies the potential reuse area of the consumer good.

E.g.:

- **Human consumption:** The quality of the product is good for human consumption.
- **Animal feed:** The product is not suitable for human consumption but it's quality is good for animal feed.
- **Composting:** The product is not safe for consumption.

Product category

This is the traditional hierarchical categorization, one can find in an online store.

We find it unnecessary and overly complex to allow for search on the product level. The product hierarchy gives the flexibility both for the stores and the charities to decide if they want to track products (e.g. milk) on a low level (e.g.: as milk category) or higher level (e.g.: as dairy, or even drink).

Allocation/Distribution models

There are many different approaches to organize allocation and distribution of the product baskets. These should be validated on the market to see which one of them are ideal for stores / charities, as they require a different level of effort from the actors. Also the available models can vary from store-to-store / chain-to-chain.

Collection models

These are basically similar to how people can shop:

- **Shopping:** The charity worker goes to the store, collects the goods like others in the store. Identifies him-/herself as acting on behalf of a charity, so the cashier gives away to goods for free or at a discounted price. In this case, the published expiring products are still out in the store shelves. This means that normal consumers can still collect them, thus the actual number of items for pickup cannot be guaranteed. To compensate for the fluctuation in the available number of products, ideally bigger batches should be offered.
- **Store pickup:** The store employee pulls the products from the shelves of the store, and places them to a container and into the pickup area. The package is still collected by the charity worker.
- **Delivery:** Starts similarly to the store pickup model, but the store also organizes the delivery of the good. This is the only possible collection model for customer fulfillment centers, where there is no physical access for the customers to the actual location, but there is still a stock of products that can expire.
 - Store employee collects goods into packages
 - Store delivers to charity location

Offering models

- **First come first served:** Once the product is identified by the stores as charitable, subscribed charities receive a notification through app or email, and can claim the item.
- **Random:** To increase fairness, this one lets charities apply for a charitable item within a time window, afterwards the system randomly assigns the items to one of them.
- **Auction based:** Each charity can bid for the charitable basket/good, the one with the highest bid can collect the items. This approach allows for the introduction of store credits, and lets chains/stores to control which charities they favour. This model also allows non-charities to participate in the system. While non-charities pay with actual money, charities can use their virtual credits, not actually paying money.
- **Subscription:** Probably one of the most sustainable approaches. Once a local charity and store learns about each others needs/offers, they can get into a long term deal about charitable food supply. Ideally most of the donations should run under this model, while excess/surge/ontime offers would fall into the other model bins.

Actors/Roles

- **Charity Worker:** A user managing a charity in the system.
- **Storekeeper:** A user working at a store location, handling the food items.
- **Storeadmin:** A user overlooking and administrating a single store location.
- **Chainadmin:** A user overlooking and administrating a chain with all its stores and product mappings.

- **Systemadmin:** An operator of the whole system. Evaluates charity/chain applications, defines core categories.

User stories

Charity

- Apply for registration.
- Login
- Search for nearby stores.
- Browse historical/current offers of stores.
- Subscribe to stores.
- Register to product/reuse/handling categories they are interested in.
- Search/browse offers of nearby / subscribed stores.
- Select items which they would like to get to a basket
- Selects desired expiration time (N days) for product category
- Select the rough amount of items from bulk items.
- Make a request.
- Get notified on the change of the order status.
- Make an approximate order for a future date.
- Statistics?

Store: Store employee(s)

- Scan data about expiring products (in few days in advance) through the system.
- Scan data about expiring products (in few days in advance) through the retailer's system. The retailer's system forwards this information through a system API.
- Collect products for charities
- Pack expired products for charities
- Publish product pack offering for charities

Chain: Administrator

- Configuration of product code to category assignments
- Create new stores of the chain.
- Statistics by stores monthly/yearly

System admin

- Inviting charities to the platform
- Evaluate charity registrations
- Administer charities
- Evaluate chain registrations
- Administer chains/stores

- Manage reuse categorization (Human, Animal, Compost)
- Manage handling categorization (Requires cooler, freezer, permit X)
- Manage product categorization (Food, Drink, Dairy, Meat, ...)

Implementation Specification

The solution needs to be publicly available, ideally on a PaaS platform, to allow for low operational costs.

Modules

Web application

The majority of the solution should be implemented as a single web application, that would allow every actor to perform their operations in the system.

Most administrative operations would be performed from desktops, but some key aspects should support mobile based usage. These are:

- **Charity side:** All the operations a charity can perform in the system after registration: Charities might have nothing else just a mobile phone. Specifying their needs and handling order pickups should be supported in a mobile accessible way.
- **Store worker:** Scanning the expiring items and updating their data in the system calls for a mobile solution. It is also important for this operation to be extremely streamlined, as in each store, this task might be performed from hundreds to thousand times daily, thus a suboptimal solution can waste a lot of time, and can also lead to rejection of the usage of the platform.

Mobile application

The charity and store worker centric operations might be better implemented within a mobile application. Such an application can have built-in barcode scanning, which could drastically speed up registration of expiring items. A minor other advantage could be a more streamlined way to send notifications to the charities about offer availability/updates.