CHRIST

(DEEMED TO BE UNIVERSITY)

BANGALORE · INDIA

SustainLink

Database Design

Team Members:

1. Aaryan Singh - 2241102
2. Akshita Mathur - 2241109
3. Soravich Pinrat – 2241158
4. Jakob Rechberger
5. Felicia Schomig

This document outlines the database design for the SustainLink project. The database is designed to facilitate efficient storage and retrieval of information related to users, stores, organizations, inventory, requests, and locations.

Normalization:

- **1NF:** All fields are atomic and do not contain repeating groups in the tables.
- **2NF:** No partial dependencies are present in any of the tables. All non-prime attributes are fully functionally dependent on the primary key.
- **3NF:** No transitive dependencies. All fields depend only on the primary key in each table.

Database Tables

1. **Users Table –** Stores essential information about platform users, allowing for unique identification and user categorization.

- user_id ***(Primary Key)***: Unique identifier for each user.
- username: User's username.
- password: User's password.
- name: User's name.
- location_id ***(Foreign Key)***: Reference to the location of the user.
- contact_detail: Contact details of the user.
- role: Type of user (Store or Organization).
- image_file: Path to the user's image or logo.

Example:

user_id	username	password	name	location_id	contact_detail	role	Image_file
01	user1	xyz	Fresh Hub	101	fresh.h@gmail.com	store	/image/str1.png
02	user2	abc	Amit Kapoor	102	amit7@gmail.com	organization	/image/org1.png

2. **Stores Table –** Manages data specific to stores, linking each to a user profile and keeping count of donation and sale transactions.

- store_id ***(Primary Key)***: Unique identifier for each store.
- user_id ***(Foreign Key)***: Reference to the user associated with the store.
- donation_count: Counter for donation transactions (for reward system).
- sale_count: Counter for sale transactions (for reward system).

Example:

store_id	user_id	donation_count	sale_count
201	01	5	10
202	03	2	8

3. **Organizations Table -** Holds details pertinent to organizations or individuals interested in taking donations or purchasing items, establishing connections to user profiles for streamlined management.

- organization_id ***(Primary Key)***: Unique identifier for each organization.
- user_id ***(Foreign Key)***: Reference to the user associated with the organization.

Example:

organization_id	user_id
301	02
302	04

4. **Inventory Table -** Organizes product information within the inventory, indicating availability and associating products with respective stores.

- product_id ***(Primary Key)***: Unique identifier for each product.
- product_name: Name of the product.
- description: Description of the product.
- expiry_date: Expiry date of the product.
- category: Category of the product.
- brand: Brand of the product.

- quantity: Quantity of the product.
- unit: Unit of measurement for the product.
- image: Optional image for visual representation.
- availability: Type of availability (For Sale, For Donation, Not Available).
- store_id ***(Foreign Key)***: Reference to the store where the product is located.

Example:

product_id	product_ name	description	expiry_ date	category	brand	quantity	unit	image	availability	store_ id
401	Canned Beans	Canned Red beans with sauce	01/10/24	food	FreshIt	57	cans	/img/p 1.png	Donation	201
402	Unisex Moisturizer	Gel based face moisturizer	11/01/25	skincare	Ponds	120	box	/img/p 2.png	Sale	202

5. **Requests Table -** Tracks requests initiated by organizations for specific products, including quantity, status updates, and related associations.

- request_id ***(Primary Key)***: Unique identifier for each request.
- product_id ***(Foreign Key)***: Reference to the product being requested.
- quantity: Quantity of the requested product.
- request_date: Date when the request is made.
- organization_id ***(Foreign Key)***: Reference to the organization making the request.
- status: Status of the request (Pending, Approved, Rejected).

Example:

request_id	product_id	quantity	request_date	organization_id	status
501	402	40	02/09/24	301	Pending
502	402	100	11/12/24	302	Approved

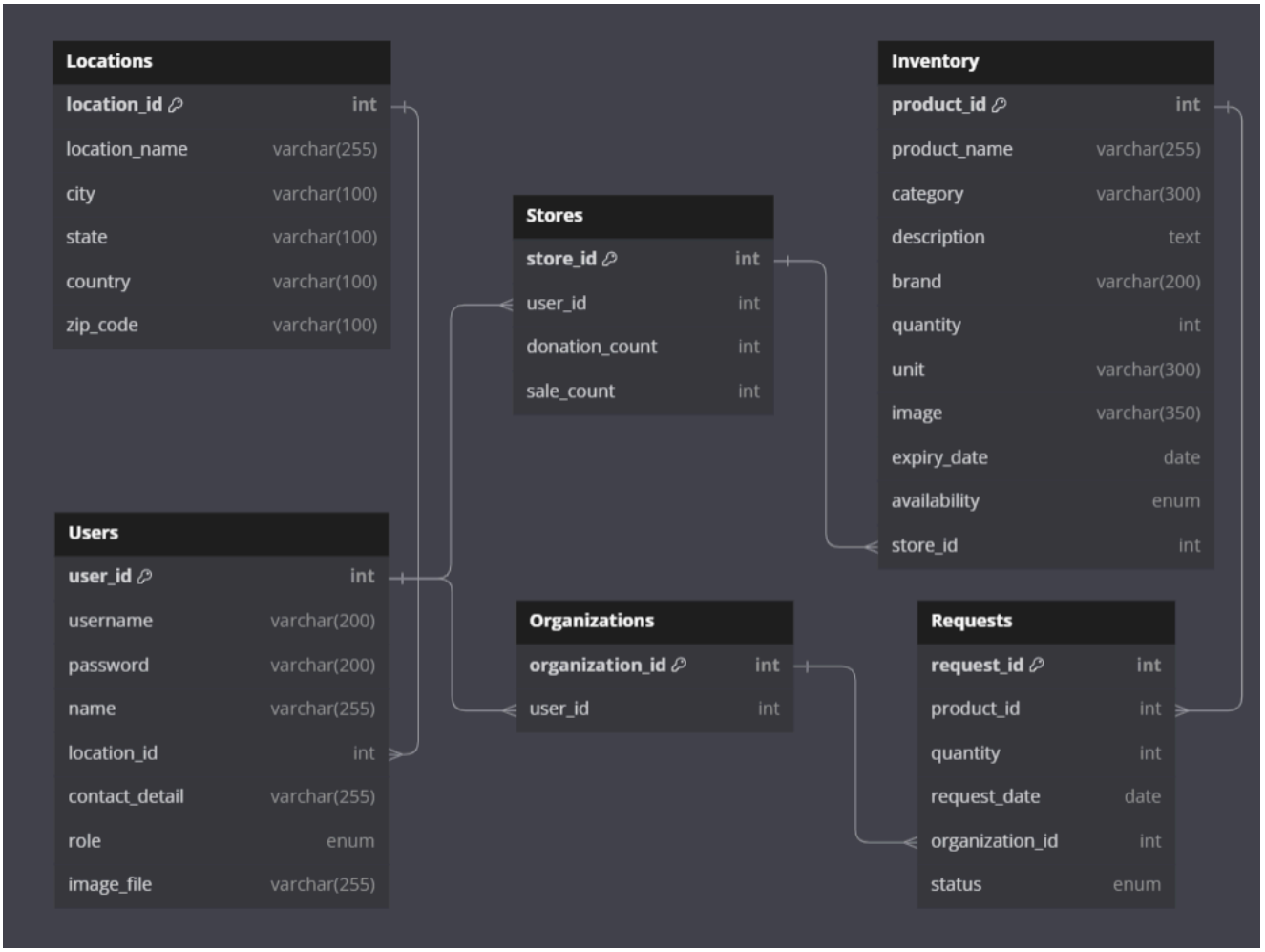
6. **Locations Table -** Provides a geographical context for the system, storing location-specific details to enhance data clarity and organization.

- location_id ***(Primary Key)***: Unique identifier for each location.
- location_name: Name of the location.
- city: City of the location.
- state: State of the location.
- country: Country of the location.
- zip_code: Zip code of the location.

Example:

location_id	location_name	city	state	country	zip_code
601	SG Palya	Bangalore	Karnataka	India	560029
602	Kurbanagar	Amritsar	Punjab	India	560021

Database Design


Locations
location_id int
location_name varchar(255)
city varchar(100)
state varchar(100)
country varchar(100)
zip_code varchar(100)
Stores
store_id int
user_id int
donation_count int
sale_count int
Inventory
product_id int
product_name varchar(255)
category varchar(300)
description text
brand varchar(200)
quantity int
unit varchar(300)
image varchar(350)
expiry_date date
availability enum
store_id int
Users
user_id int
username varchar(200)
password varchar(200)
name varchar(255)
location_id int
contact_detail varchar(255)
role enum
image_file varchar(255)
Organizations
organization_id int
user_id int
Requests
request_id int
product_id int
quantity int
request_date date
organization_id int
status enum

DBML Code

```
Table Users {
  user_id int [pk]
  username varchar(200)
  password varchar(200)
  name varchar(255)
  location_id int [ref: > Locations.location_id]
  contact_detail varchar(255)
  role enum('Store', 'Organization')
  image_file varchar(255) // Example for images or logos
}

Table Stores {
  store_id int [pk]
  user_id int [ref: > Users.user_id]
  donation_count int //for reward system
  sale_count int
}

Table Organizations {
  organization_id int [pk]
  user_id int [ref: > Users.user_id]
  // Add other organization-specific fields here
}

Table Inventory {
  product_id int [pk]
  product_name varchar(255)
  category varchar (300)
  description text
  brand varchar(200)
  quantity int
  unit varchar(300)
  image varchar (350)
  expiry_date date
  availability enum('For Sale', 'For Donation' 'Not Available')
  store_id int [ref: > Stores.store_id]
}

Table Requests {
  request_id int [pk]
  product_id int [ref: > Inventory.product_id]
  quantity int
  request_date date
  organization_id int [ref: > Organizations.organization_id]
  status enum('Pending', 'Approved', 'Rejected')
}

Table Locations {
  location_id int [pk]
  location_name varchar(255)
  city varchar(100)
  state varchar(100)
  country varchar(100)
  zip_code varchar(100)
  // Add other location-specific fields here
}
```