

Food Bank Management Software System

Raunak Sandhu, David Oti-George, Finn Vamosi

ABSTRACT

Our Food Bank Management Software System addresses an important need identified within the SU Food Bank and various other Food Banks — the absence of a comprehensive software management system. This project aims to address the challenges faced in tracking orders, managing inventory, and coordinating between users and admins involved in the food bank's operations. The motivation behind this initiative stems from the positive impact it could have on the SU Food Bank by improving the organization's productivity and efficiency.

Our solution involves using MySQL as the relational database management system, and using Flask as the backend framework and React for the frontend library. MySQL serves to securely store and manage data for this project, and we used MySQL Workbench to create and manage the database locally. Flask was used to handle HTTP requests from the frontend and also to handle requests to the database. React was used to create a dynamic frontend that sent HTTP requests to the backend and displayed results to the webpages as well as setup the webpages navigation.

The project aims to provide a Food Bank with the ability to track orders, requests, administrators, users and inventory. This system caters to the different needs of donors, clients (those who request items from the foodbank), volunteers, and coordinators. Clients and donors can create accounts and view inventory, while volunteers and coordinators have tools for managing hamper requests, tracking fulfillment, and overseeing inventory.

INTRODUCTION

UCalgary's SU Food Bank provides an important service to the campus community, serving as a safety net to students and staff in need. However, they currently have no software system for tracking orders and inventory, handling incoming food requests, or organizing volunteers and clients, which causes inefficiencies and reduces the amount of people they can help in a day. Inventory is constantly in flux, making it difficult to see which items need to be ordered. Currently, clients request food hampers through virtual meetings with a volunteer, but this could just as easily be done with an online form, which would free up volunteers to pack more hampers and handle incoming orders and donations. Demand for the food bank's services is higher than ever within the campus community, and there is not enough funding from the university to hire developers for the creation of a system of this kind.

DESCRIPTION

To address this problem, we have developed a comprehensive website which connects with a database system. Through the website, volunteers at the FoodBank can view current inventory, place orders to suppliers, verify users, place orders from suppliers, as well as view volunteer shifts, while coordinators (highest level employees at the Foodbank) can do all the same and also add new admins. Clients can place hamper requests, and donors can view their previous donations. Clients and donors must sign up for accounts, inputting the necessary personal information, while volunteers and coordinators must be added by a coordinator. Clients can also make hamper requests, while donors can view past donations made. All users (clients and donors) are initially shown a login page, which will direct them to the correct dashboard (admin-volunteer/coordinator or user-donor/client) based on their stored account type. The website directly interacts with the MySQL database through the Flask backend, so all webpages are persistent and updated when database values change.

DESIGN

Donors:

- Account Creation: Donors can register accounts with minimal personal information.
 - Login.
- View Needed Items: View items in low quantity which are urgently needed by the food bank.
- View Previous Donations: Receive confirmation of their good deed and see what they donated.
- Logout.

Clients:

- Account Creation: Clients must register accounts with more in-depth personal information.
 - Login.
- Online Requests: Place orders for food hampers, specifying their desired items.
- Request History: View their past orders and track fulfillment.
- View Inventory: See if the items or alternatives they want are available before making a request.
- Logout.

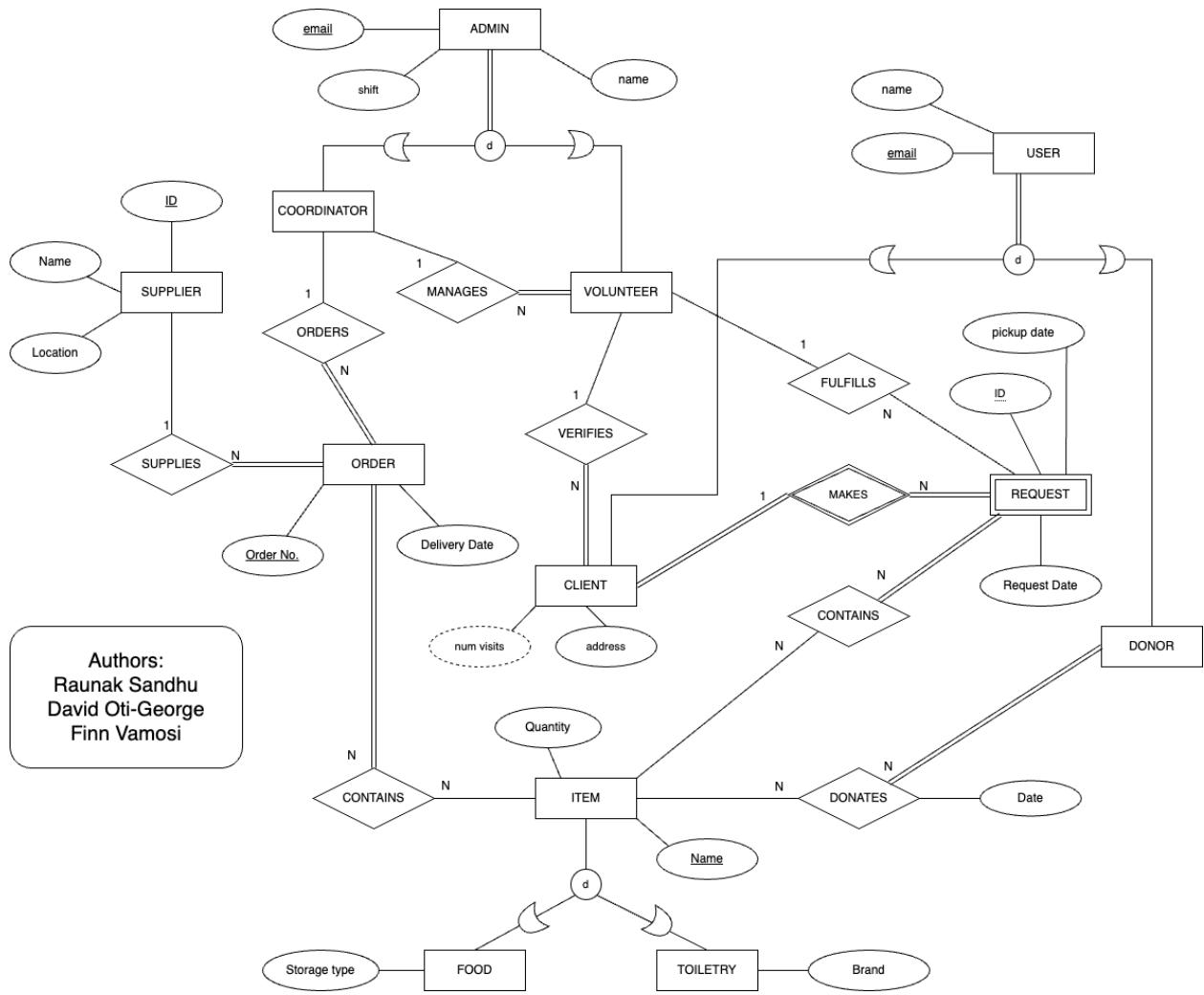
Volunteers:

- Login: Required to login using verified credentials.
- View Hamper Requests: Access details of hamper requests in order to pack them.
- Fulfill Requests: Mark orders as fulfilled and ready for pickup once packed.

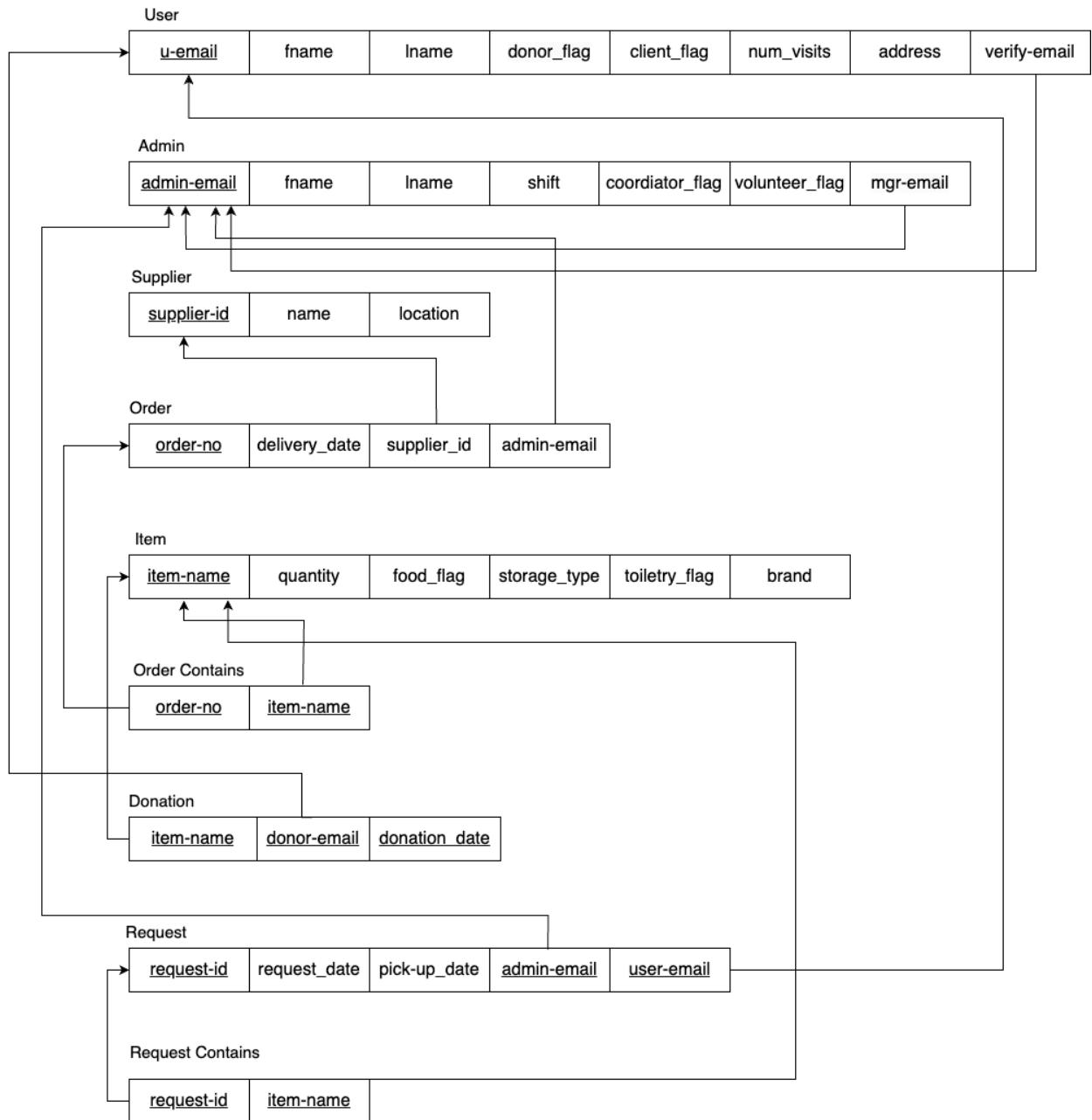
- Inventory Updates: Marking an order as fulfilled will remove its requested items from the inventory.
- Order Supplies: Place orders with wholesale suppliers to replenish stocks.
- Donation Creation: Create a donation from a donor in the system when they drop off items.
- View/Edit Inventory: Update quantity of items, or add new items that will be offered moving forward.
- Verify Users: Once users' information and enrollment is confirmed, they can be verified in the system.
 - Admin is notified if the user is already verified.
- Logout.

Coordinators:

- Login: Required to login using verified credentials.
- View Hamper Requests: Access details of hamper requests in order to pack them.
- Fulfill Requests: Mark orders as fulfilled and ready for pickup once packed.
 - Inventory Updates: Marking an order as fulfilled will remove its requested items from the inventory.
- Order Supplies: Place orders with wholesale suppliers to replenish stocks.
- Donation Creation: Create a donation from a donor in the system when they drop off items.
- View/Edit Inventory: Update quantity of items, or add new items that will be offered moving forward.
- Verify Users: Once users' information and enrollment is confirmed, they can be verified in the system.
 - Admin is notified if the user is already verified.
- Logout.
- Personnel Management: Create accounts for new volunteers and coordinators with shift info and manager email.

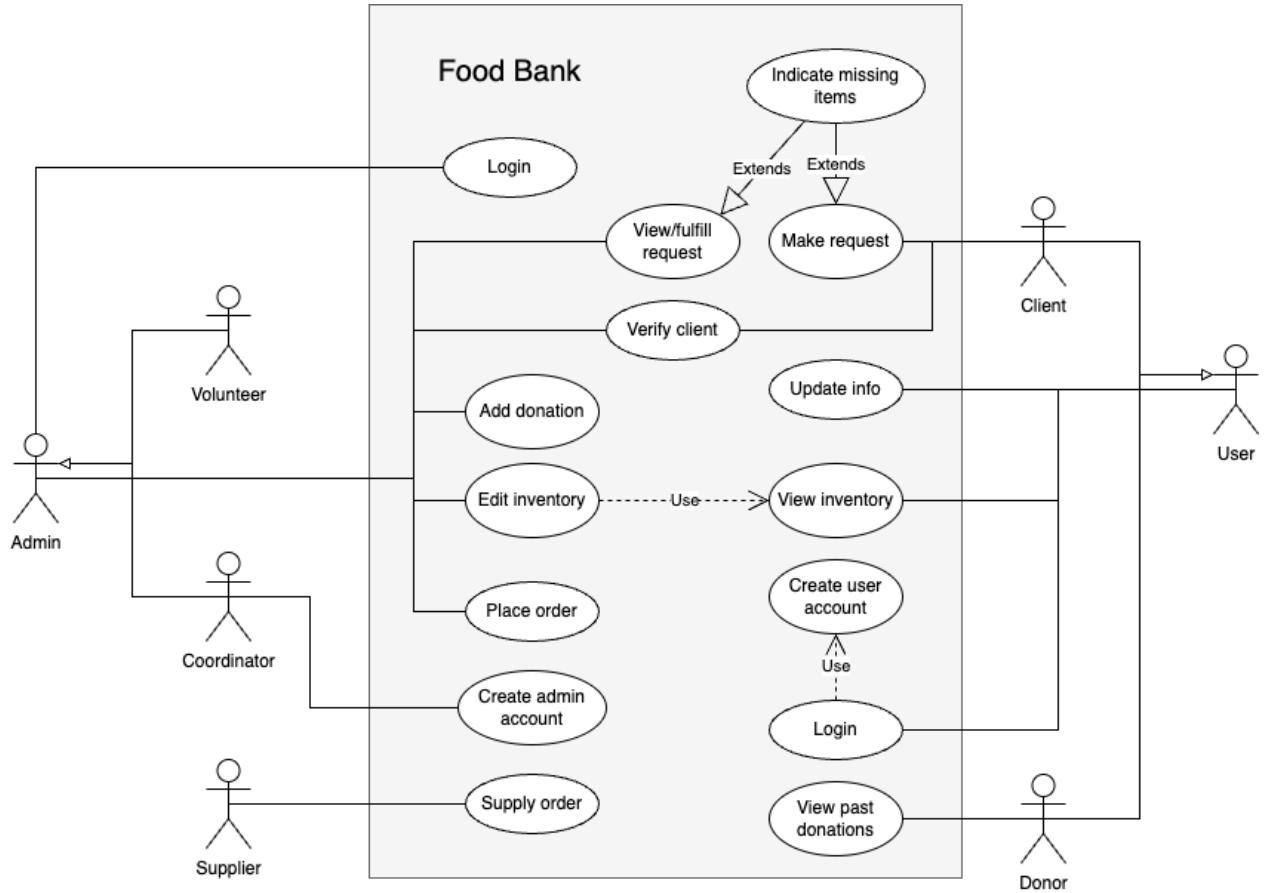


IMPLEMENTATION

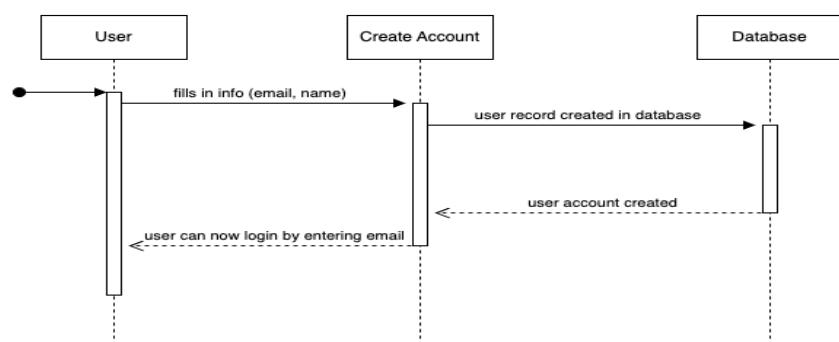
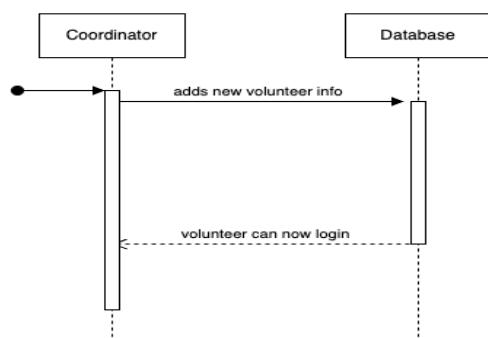
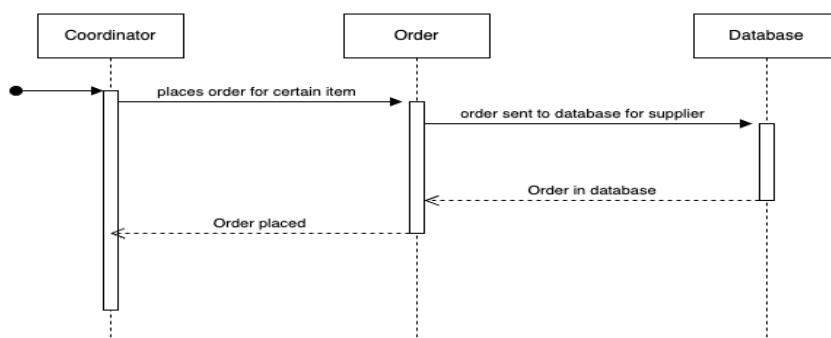
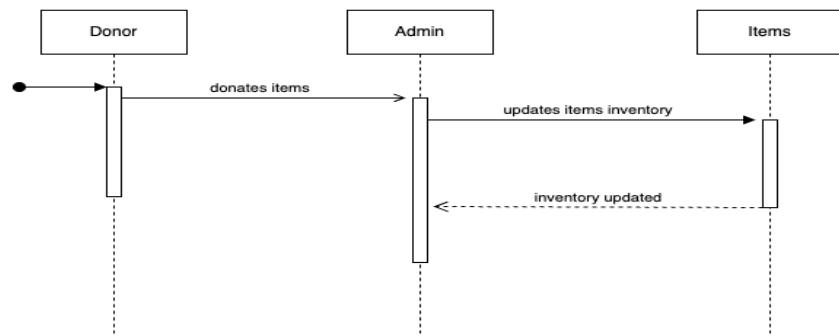
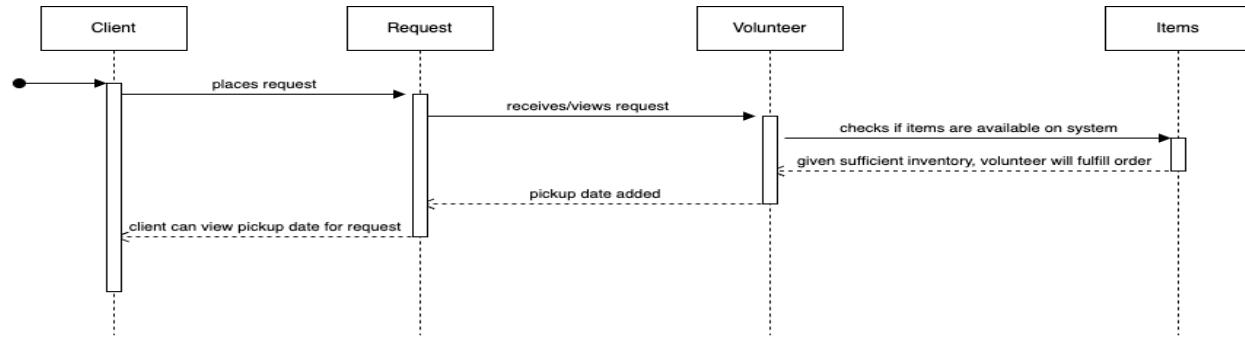


This relational model was created by following the standard algorithm for converting an entity relationship diagram to a relational schema diagram, with no changes made.

Use Case Diagram



Sequence Diagram



For the implementation of our Food Bank Management Software project, we opted for MySQL as our DBMS, alongside Flask for the backend and React for the frontend. MySQL Workbench was used for managing the database with ease during the initialization and testing phases of the project.

SQL Code (we have included an sql file in our project folder to import our database without having to run these commands):

Tables

CREATE TABLE Admin		
(a-email	VARCHAR(30)	NOT NULL,
fname	VARCHAR(10)	NOT NULL,
lname	VARCHAR(10)	NOT NULL,
shift	VARCHAR(45),	
coordinator-flag	CHAR(1),	NOT NULL,
volunteer-flag	CHAR(1),	NOT NULL,
mgr-email	VARCHAR(30),	
Primary Key (a-email))		
FOREIGN KEY (mgr-email) REFERENCES ADMIN (a-email)		
ON DELETE SET NULL,		
ON UPDATE CASCADE,);		
CREATE TABLE User		
(u-email	VARCHAR(30)	NOT NULL,
fname	VARCHAR(10)	NOT NULL,
lname	VARCHAR(10)	NOT NULL,
donor-flag	CHAR(1),	
client_flag	CHAR(1),	
num_visits	INT,	
address	VARCHAR(30),	
verify-email	VARCHAR(30),	
Primary KEY (u-email),		
FOREIGN KEY (verify-email) REFERENCES ADMIN (a-email)		
ON DELETE SET NULL,		
ON UPDATE CASCADE,);		
CREATE TABLE Supplier		
(supplier_id	CHAR(10)	NOT NULL,
name	VARCHAR(20)	NOT NULL,

location VARCHAR(30) **NOT NULL,**
Primary KEY (supplier_id));

CREATE TABLE `Order`
(order-no CHAR(10) **NOT NULL,**
delivery_date DATE,
admin-email VARCHAR(30),
supplier_id VARCHAR(10),
Primary KEY (order-no),
FOREIGN KEY (admin-email) **REFERENCES** ADMIN (a-email),
ON DELETE SET NULL,
 ON UPDATE CASCADE,
FOREIGN KEY (supplier_id) **REFERENCES** Supplier (supplier_id)
ON DELETE SET NULL,
 ON UPDATE CASCADE);

CREATE TABLE Item
(item-name VARCHAR(20) **NOT NULL,**
quantity INT **DEFAULT** 0 **NOT NULL,**
storage_type VARCHAR(10) **NOT NULL,**
brand VARCHAR(10) **NOT NULL,**
food_flag CHAR(1),
toiletry_flag CHAR(1),
Primary KEY (item-name));

CREATE TABLE Order_Contains(
order-no CHAR(10) **NOT NULL,**
item-name VARCHAR(20) **NOT NULL,**
Primary KEY (order-no),
Primary KEY (item-name),
FOREIGN KEY (order-no) **REFERENCES** `Order` (order-no),
ON DELETE CASCADE,
 ON UPDATE CASCADE,
FOREIGN KEY (item-name) **REFERENCES** Item (item-name),
ON DELETE CASCADE,
 ON UPDATE CASCADE);

CREATE TABLE Donation(

```

admin-email          VARCHAR(30),           NOT NULL,
item-name           VARCHAR(20),
donation_date      DATE                  NOT NULL,
Primary KEY (admin-email),
Primary KEY (item-name),/
FOREIGN KEY (admin-email) REFERENCES Admin (a-email),
ON DELETE CASCADE,
    ON UPDATE CASCADE,
FOREIGN KEY (item-name) REFERENCES Item (item-name)
ON DELETE SET NULL,
    ON UPDATE CASCADE );

```

```

CREATE TABLE Request(
    request-id        CHAR(10)            NOT NULL,
    admin-email       VARCHAR(30),
    user-email        VARCHAR(30),           NOT NULL,
    pick-up_date     DATE,
    request_date     DATE,
Primary KEY (request-id ),
FOREIGN KEY (admin-email) REFERENCES Admin (a-email),
ON DELETE SET NULL,
    ON UPDATE CASCADE,
FOREIGN KEY (user-email) REFERENCES User (u-email),
ON DELETE CASCADE,
    ON UPDATE CASCADE);

```

```

CREATE TABLE Request_Contains(
    request-id        CHAR(10),           NOT NULL,
    item-name         VARCHAR(20),          NOT NULL,
Primary KEY (request-id),
Primary KEY (item-name),
FOREIGN KEY (request-id) REFERENCES Request (request-id),
ON DELETE CASCADE,
    ON UPDATE CASCADE,
FOREIGN KEY (item-name) REFERENCES Item (item-name),
ON DELETE CASCADE,
    ON UPDATE CASCADE);

```

Example INSERT and UPDATE Statements

Admin Table

```
INSERT INTO Admin (a-email, fname, lname, shift, coordination-flag, volunteer_flag, mgr-email)
VALUES ('admin@example.com', 'John', 'Doe', 'Morning', 'Y', 'N', 'manager@example.com');
```

```
UPDATE Admin
```

```
SET shift = 'Evening'
WHERE a-email = 'admin@example.com';
```

User Table

```
INSERT INTO User (u-email, fname, lname, donor-flag, client_flag, num_visits, address,
verify-email)
VALUES ('user@example.com', 'Alice', 'Smith', 'Y', 'N', 3, '123 Main St', 'admin@example.com');
```

```
UPDATE User
```

```
SET num_visits = 4
WHERE u-email = 'user@example.com';
```

Supplier Table

```
INSERT INTO Supplier (supplier_id, name, location)
VALUES ('supp001', 'ABC Suppliers', 'Cityville');
```

```
UPDATE Supplier
```

```
SET location = 'Townsville'
WHERE supplier_id = 'supp001';
```

Order Table

```
INSERT INTO Order (order-no, delivery_date, admin-email, supplier_id)
VALUES ('order001', '2023-12-01', 'admin@example.com', 'supp001');
```

```
UPDATE Order
```

```
SET delivery_date = '2023-12-02'
WHERE order-no = 'order001';
```

Item Table

```
INSERT INTO Item (item-name, quantity, storage_type, brand, food_flag, toiletry_flag)
VALUES ('Shampoo', 50, 'Shelf', 'BrandX', 'N', 'Y');
```

```
UPDATE Item
```

```
SET quantity = 55  
WHERE item-name = 'Shampoo';
```

Order_Contains Table

```
INSERT INTO Order_Contains (order-no, item-name)  
VALUES ('order001', 'Shampoo');
```

Donation Table

```
INSERT INTO Donation (admin-email, item-name, donation_date)  
VALUES ('admin@example.com', 'Shampoo', '2023-12-03');
```

Request Table

```
INSERT INTO Request (request-id, admin-email, user-email, pick-up_date, request_date)  
VALUES ('request001', 'admin@example.com', 'user@example.com', '2023-12-05',  
'2023-12-01');
```

```
UPDATE Request  
SET pick-up_date = '2023-12-06'  
WHERE request-id = 'request001';
```

Request_Contains Table

```
INSERT INTO Request_Contains (request-id, item-name)  
VALUES ('request001', 'Shampoo');
```

Additional Statements Used

Admin

```
"SELECT * FROM Admin"
```

```
"SELECT fname FROM Admin"
```

```
"SELECT a_email FROM Admin WHERE a_email = '%s'"
```

User

```
"UPDATE User SET verify_email = %s WHERE u_email = %s"
```

```
"SELECT * FROM User"
```

```
"SELECT fname FROM User"
```

"SELECT u_email FROM User WHERE u_email = '%s'"

"SELECT * FROM User WHERE u_email = '%s'"

Supplier

"SELECT * FROM Supplier"

Request

"SELECT * FROM Request"

"SELECT * FROM Request WHERE request_user = '%s'"

"Delete FROM Request WHERE request_id = %s"

"UPDATE Request SET request_admin = %s, pickup_date = %s WHERE request_id = %s"

Request_Contains

"SELECT request_item FROM Request_Contains WHERE id_request = '%s'"

Order

"SELECT * FROM Order"

Order_Contains

"SELECT item_name FROM Order_Contains WHERE order_no = '%s'"

Item

"SELECT * FROM Item"

"SELECT item_name, quantity FROM Item"

"SELECT item_name FROM Item WHERE item_name = '%s'"

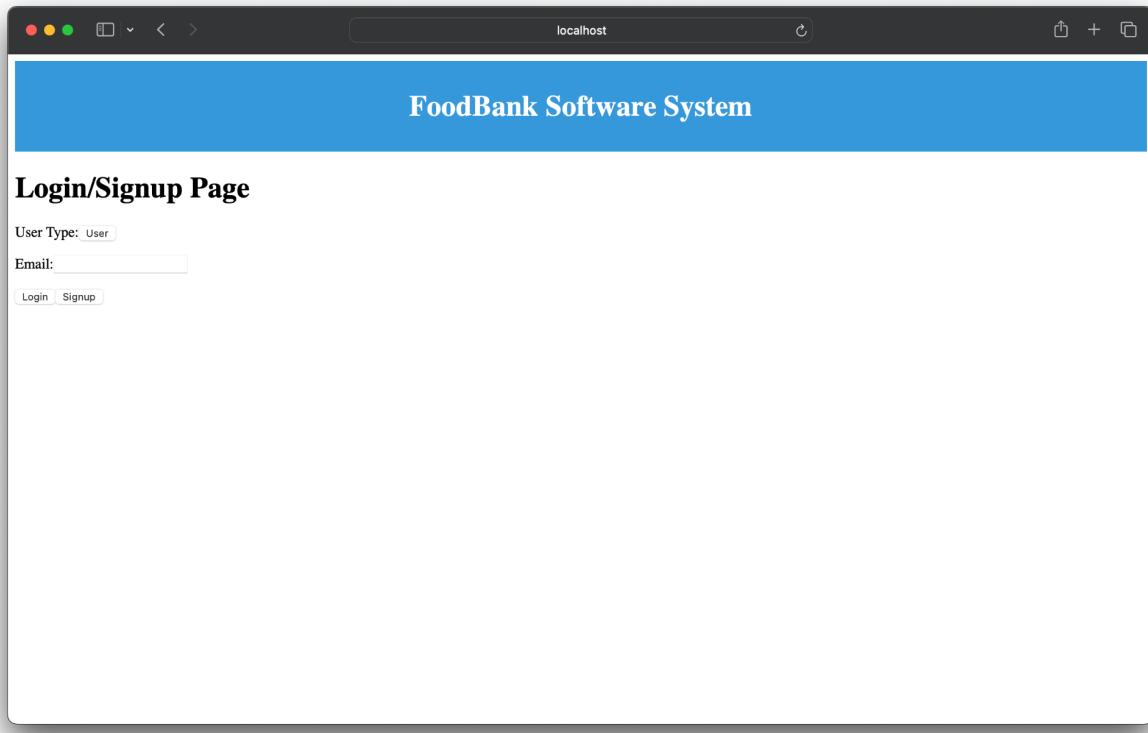
"UPDATE Item SET quantity = %s WHERE item_name = %s"

"DELETE FROM Item WHERE item_name = %s"

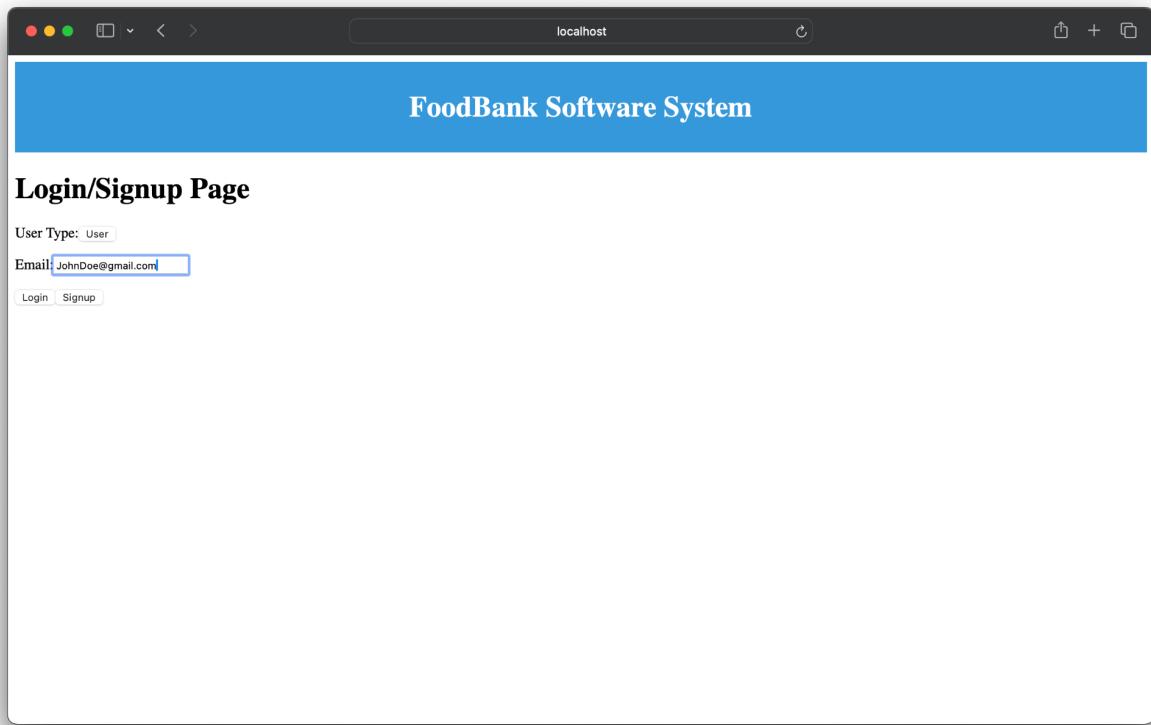
Donation

```
"SELECT item, donation_date FROM Donation WHERE donor_email = '%s'"
```

USER MANUAL



This landing page is the first thing any user sees upon accessing the food bank website. From here, existing users can login using their registered email address. Below, the existing client John Doe enters his email, then presses the “Login” button.

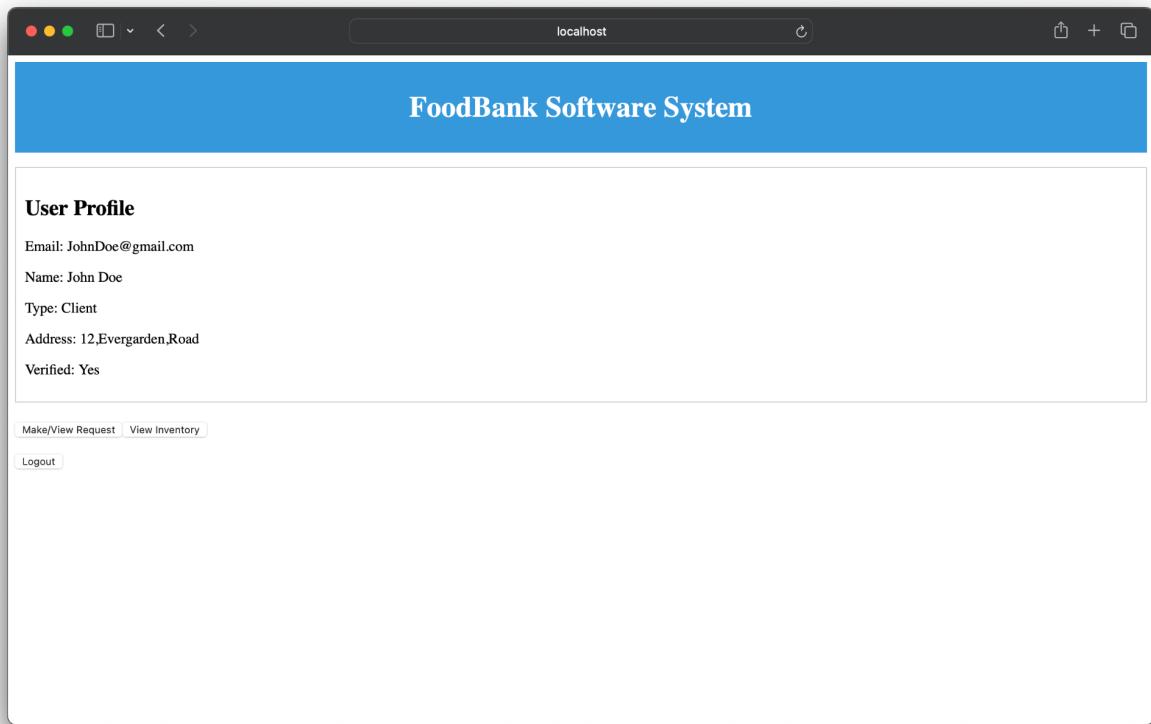


The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main title "FoodBank Software System" is centered at the top in a blue header bar. Below it, the page title "Login/Signup Page" is displayed in bold black text. A sub-header "User Type:" is followed by a dropdown menu showing "User". An input field for "Email" contains "JohnDoe@gmail.com". At the bottom of the form are two buttons: "Login" and "Signup".

User Type: User

Email: JohnDoe@gmail.com

Login Signup



The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main title "FoodBank Software System" is centered at the top in a blue header bar. Below it, the page title "User Profile" is displayed in bold black text. The user profile information is listed as follows:

- Email: JohnDoe@gmail.com
- Name: John Doe
- Type: Client
- Address: 12,Evergarden,Road
- Verified: Yes

At the bottom of the page are three buttons: "Make/View Request", "View Inventory", and "Logout".

User Profile

Email: JohnDoe@gmail.com

Name: John Doe

Type: Client

Address: 12,Evergarden,Road

Verified: Yes

Make/View Request View Inventory

Logout

Upon doing so, John is brought to his user profile. From here, since he registered as a client, he can view the current inventory, make a request, or view his previous requests. He can also log out at any time by pressing the “Logout” button at the bottom of the page, which will return him to the landing page. Clicking on the “View Inventory” button brings up a table showing the names, quantities, storage types, brands, and item types of each item the food bank offers, as shown below.

The screenshot shows a web browser window titled "FoodBank Software System" on the localhost. The main content area is divided into two sections: "User Profile" and "Inventory".

User Profile:

- Email: JohnDoe@gmail.com
- Name: John Doe
- Type: Client
- Address: 12,Evergarden,Road
- Verified: Yes

Inventory:

Name	Quantity	Storage Type	Brand	Type
Dad Cookies	33	Shelf	Mondelez	food
Kellogs Corn Flakes	13	Shelf	Kellogs	food
Ultra Comfort Toilet Paper	2	Shelf	Charmin	toiletry
Ultra Soft Toilet Paper	5	Shelf	Charmin	toiletry

At the bottom left, there is a "Logout" button.

Suppose John then decides to place a hamper request. He clicks on the “Make/View Request”, bringing up the following page, where makes a selection of items he would like to request: Dad’s Cookies and Ultra Comfort Toilet Paper. Once done, he presses “Make Request”.

Name: John Doe
Type: Client
Address: 12,Evergarden,Road
Verified: Yes

Make/View Request | View Inventory

Hamper Request

Make Request:

Unique RequestID: 4b626bfdac

Select Items

- Dad Cookies
- Kelloggs Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

Selected Items:

Past Requests:

Request ID	Request User	Ready Date	Request Date
be68305bb3	johndoe@gmail.com	December 10, 2023	December 10, 2023

Name: John Doe
Type: Client
Address: 12,Evergarden,Road
Verified: Yes

Make/View Request | View Inventory

Hamper Request

Make Request:

Unique RequestID: 4b626bfdac

Select Items

- Dad Cookies
- Kelloggs Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

Selected Items: Dad Cookies, Ultra Comfort Toilet Paper

Past Requests:

Request ID	Request User	Ready Date	Request Date
be68305bb3	johndoe@gmail.com	December 10, 2023	December 10, 2023

Request and Request_Contains inserted successfully!

The request has been successfully made! John dismisses the popup and is returned to the Make/View Request screen, where we can see that the new request with ID 08d0933923 has been added to his list of past requests:

Type: Client
Address: 12, Evergarden, Road
Verified: Yes

[Make/View Request](#) | [View Inventory](#)

Hamper Request

Make Request:

Unique RequestID: e8b5547710

Select Items

- Dad Cookies
- Kellogg's Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

Selected Items:

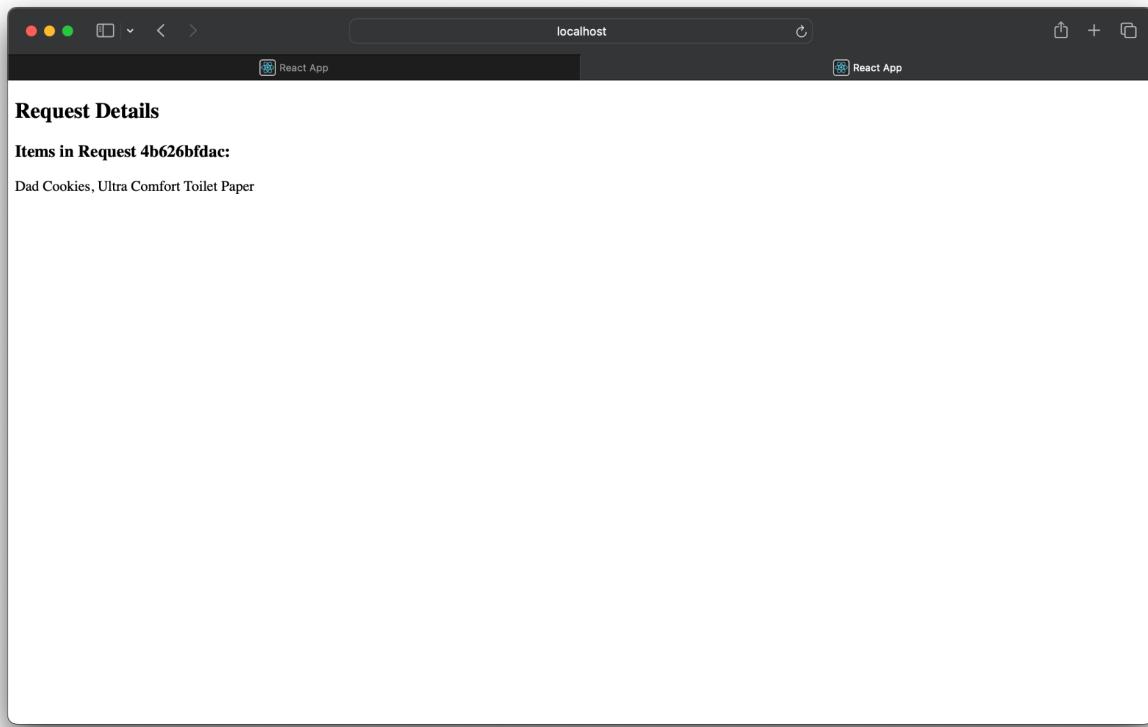
[Make Request](#)

Past Requests:

Request ID	Request User	Ready Date	Request Date
4b626bfdac	johndoe@gmail.com		December 11, 2023
be68305bb3	johndoe@gmail.com	December 10, 2023	December 10, 2023

[Logout](#)

If John wants to check what he ordered, he can click on the highlighted “Request ID”, which will open the following page in a new tab:



Now suppose a new user comes to the site and wants to create an account. They press the “Signup” button, and are brought to the account creation page, as shown below. They then enter their email address, first and last name, account type, and home address.

A screenshot of a web browser window titled "localhost". The main title bar says "FoodBank Software System". Below it, the sub-page title is "User Signup". The form contains fields for Email, First Name, Last Name, User Type (set to "Client"), and Address. At the bottom are "Signup" and "Back" buttons.

Email:

First Name:

Last Name:

User Type: Client

Address:

[Signup](#) [Back](#)

A screenshot of a web browser window titled "localhost". The main title bar says "FoodBank Software System". Below it, the sub-page title is "User Signup". The form fields have been populated with sample data: Email ("SomeName@gmail.com"), First Name ("Some"), Last Name ("Name"), User Type ("Client" radio button selected), and Address ("123 Lives Here"). At the bottom are "Signup" and "Back" buttons.

Email: SomeName@gmail.com

First Name: Some

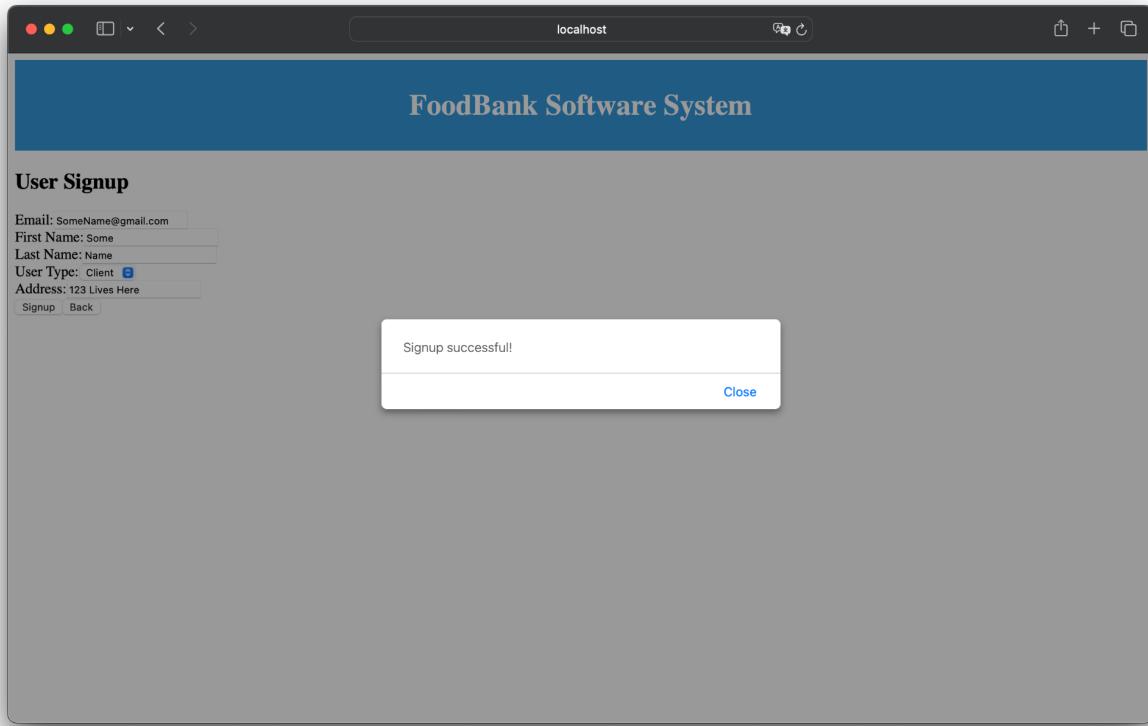
Last Name: Name

User Type: Client

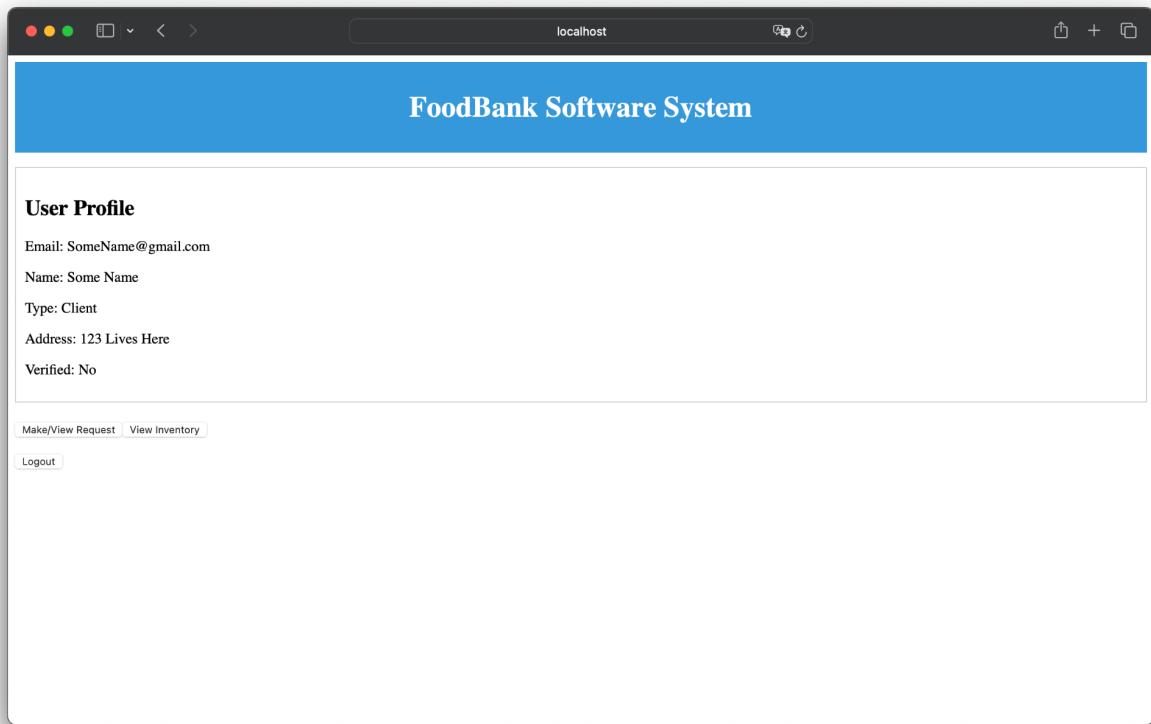
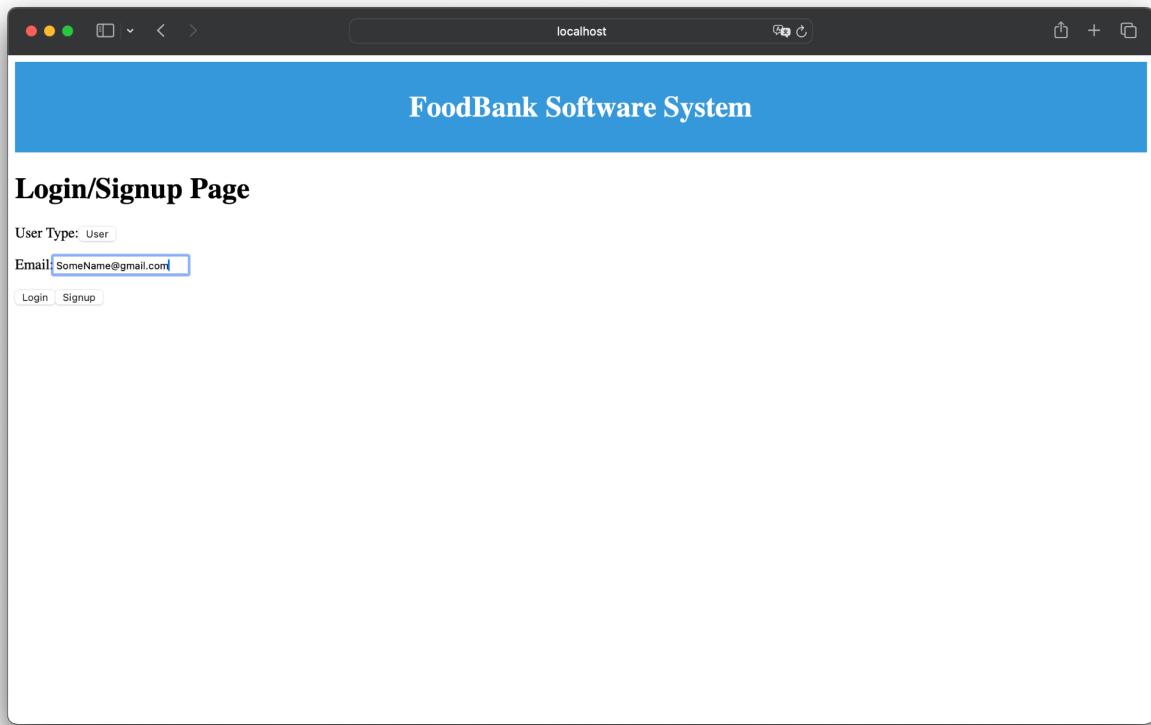
Address: 123 Lives Here

[Signup](#) [Back](#)

An example of a new client's information is shown above. This new client then presses "Signup", and their account is created and entered in the database.



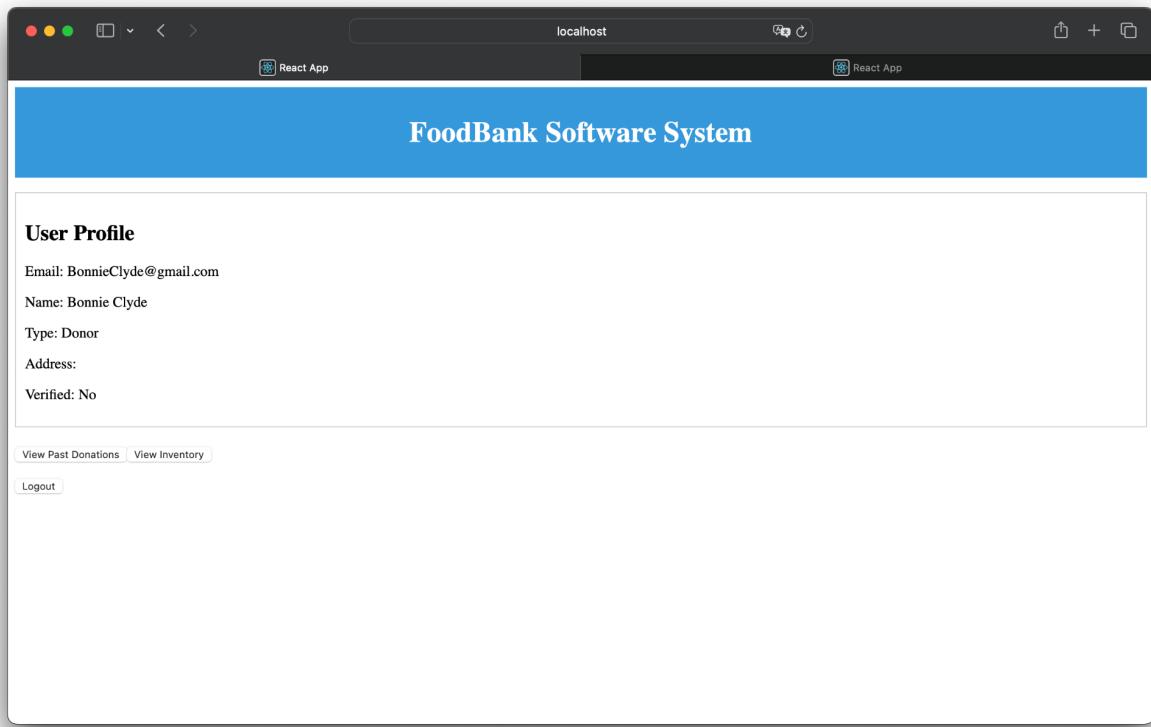
Then the new client returns to the landing page, where they can enter their email to login, as below.



Here we can see that all their information was correctly added to the database. They can now view inventory or make/view requests, as shown earlier. Clicking on “Make/View Request”, we see that this new user has no past requests, as expected.

The screenshot shows a web application running on localhost. At the top, there is a header bar with the URL 'localhost'. Below the header, a sidebar displays account information: Email: SomeName@gmail.com, Name: Some Name, Type: Client, Address: 123 Lives Here, and Verified: No. At the bottom of this sidebar are two buttons: 'Make/View Request' and 'View Inventory'. The main content area is titled 'Hamper Request' and contains a section for 'Make Request'. It shows a 'Unique RequestID: 909699b8ac'. Below this is a 'Select Items' section with a list of items: Dad Cookies, Kellogg's Corn Flakes, Ultra Comfort Toilet Paper, and Ultra Soft Toilet Paper. Underneath the list is a 'Selected Items:' section which is currently empty. A 'Make Request' button is located at the bottom of this section. Further down, there is a 'Past Requests:' section with a table header row containing 'Request ID', 'Request User', 'Ready Date', and 'Request Date'. At the very bottom of the page is a 'Logout' button.

Next, suppose we have an existing donor named Bonnie Clyde. Bonnie logs in by entering their email address, as usual, but since her account type is “Donor”, she is brought to the donor page instead. This page is different from those “Clients” are shown, in that instead of being able to make or view requests, donors have the option to view their past donations, or view the current inventory. Pressing the “View Inventory” brings up the same table as shown before.



Upon pressing the “View Past Donations” button, a table appears with the desired results. Note that donors DO NOT place donations online - they drop items off in person, and an admin adds their donations to the donation through the website. The way this is done is described in the Admin Manual.

The screenshot shows a web application window titled "FoodBank Software System" running on "localhost". The main content area is titled "User Profile" and displays the following information:

- Email: BonnieClyde@gmail.com
- Name: Bonnie Clyde
- Type: Donor
- Address: (empty)
- Verified: No

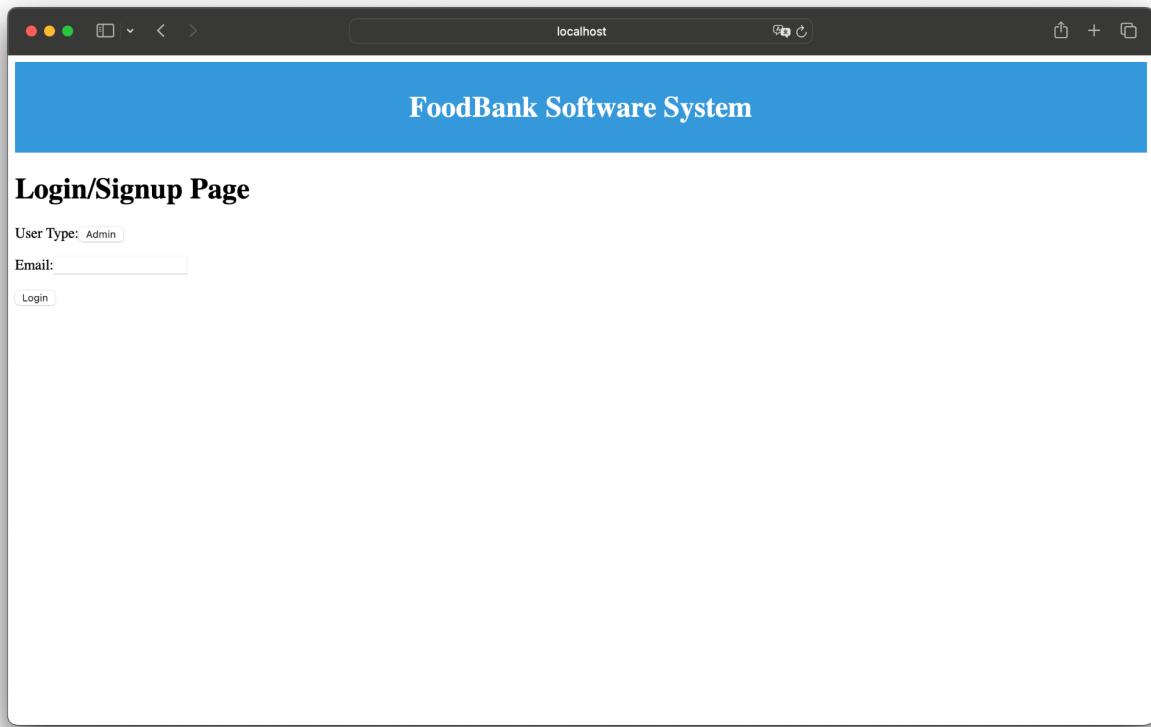
Below the profile section are two links: "View Past Donations" and "View Inventory".

The "View Past Donations" section contains a table showing past donations:

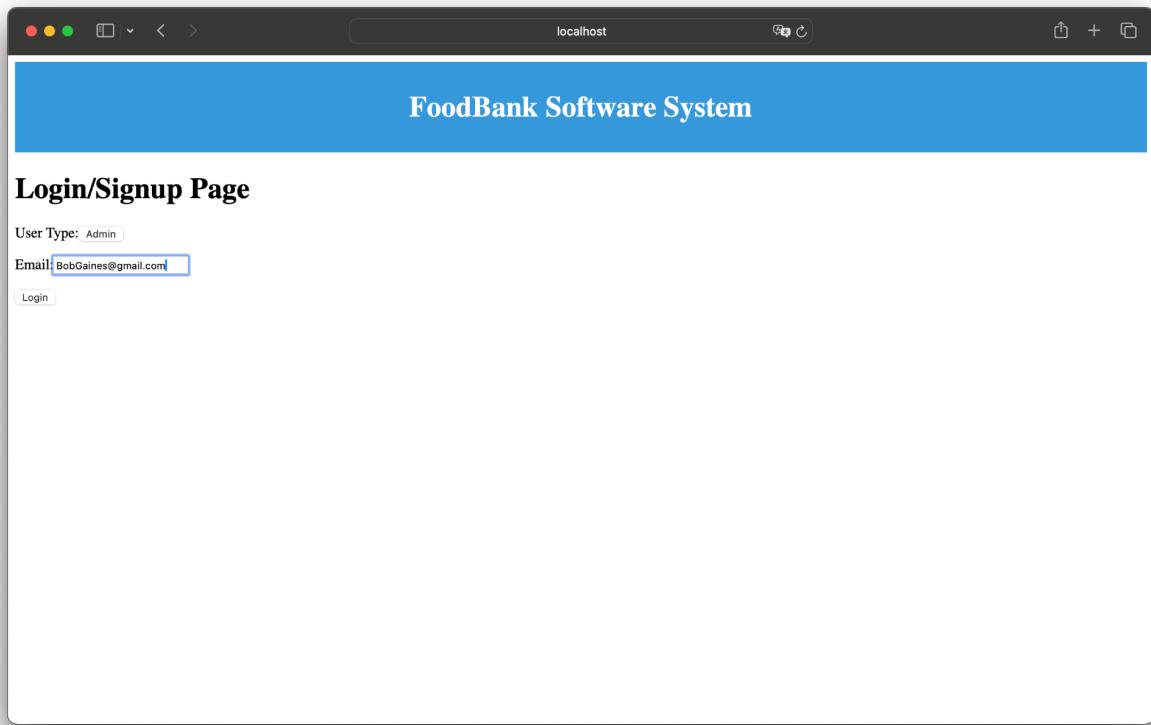
Item	Donation Date
Ultra Comfort Toilet Paper	2023-12-10
Ultra Soft Toilet Paper	2023-12-10

A "Logout" button is located at the bottom left of the main content area.

ADMIN MANUAL



When an admin (either a volunteer or coordinator) wants to login, they press the toggle button next to "User Type", setting it to "Admin". They then enter their email, as would a user.

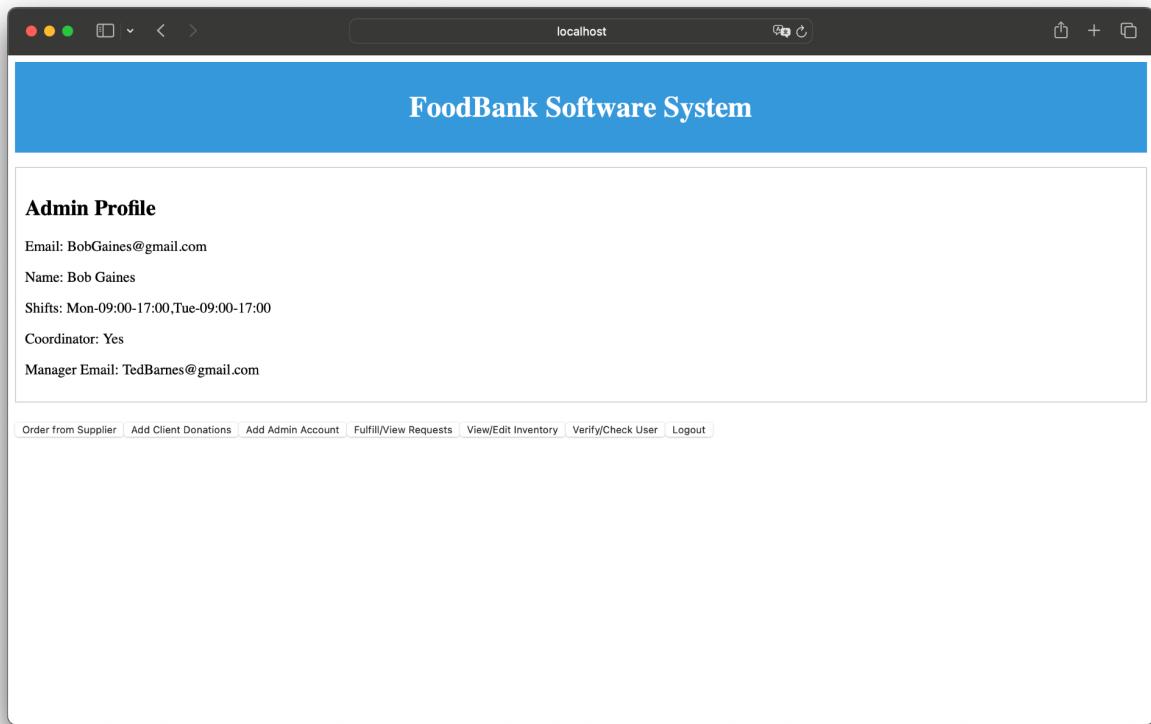


The screenshot shows a web browser window titled "FoodBank Software System" on the localhost. The page is titled "Login/Signup Page". It has two input fields: "User Type" set to "Admin" and "Email" set to "BobGaines@gmail.com". A "Login" button is located below the fields.

User Type: Admin

Email: BobGaines@gmail.com

Login



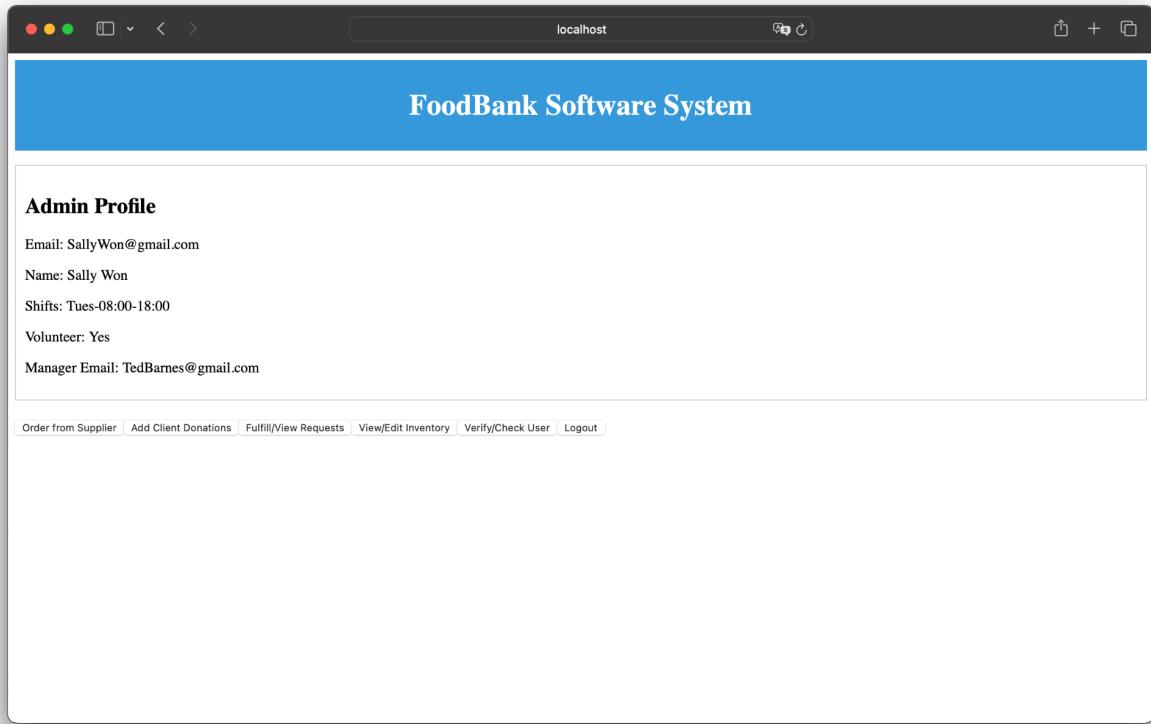
The screenshot shows a web browser window titled "FoodBank Software System" on the localhost. The page displays an "Admin Profile" section with the following information:

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

At the bottom of the page, there is a navigation bar with links: Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

This brings them to the admin page, which has many more options than a user account. This admin, Bob Gaines, is a coordinator, but the sole difference between volunteers and coordinators is that the latter may add other admin accounts. See below to compare with the volunteer Sally Won's dashboard:



Moving forward, we will work with Bob's coordinator account in order to explore all the options. First, we press the "Order from Supplier" button. This brings up the following page, which contains several options. The first is the desired delivery date; clicking the text box brings up a calendar widget which makes selecting the date easy.

localhost

Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Order from Supplier | Add Client Donations | Add Admin Account | Fulfill/View Requests | View/Edit Inventory | Verify/Check User | Logout

Orders Section

Add Order

Delivery Date: 2023-12-10

Supplier: Select a supplier

Items:

- Dad Cookies
- Kellogs Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

Add Order

Orders

Order No	Delivery Date	Admin Email	Supplier ID
3d5af866b4	December 29, 2023	SallyWon@gmail.com	0000000001

localhost

Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Order from Supplier | Add Client Donations | Add Admin Account | Fulfill/View Requests | View/Edit Inventory | Verify/Check User | Logout

Orders Section

Add Order

Delivery Date: 2023-12-14

Supplier: Select a supplier

Items:

- Dad Cook
- Kellogg C
- Ultra Con
- Ultra Soft Toilet Paper

Add Order

Orders

Order No	Delivery Date	Admin Email	Supplier ID
3d5af866b4	December 29, 2023	SallyWon@gmail.com	0000000001

Next, the desired supplier is selected from a drop down widget with each partnered supplier, and the needed items are selected. Once this is done, Bob presses the “Add Order” button, and a message appears confirming that the order was placed successfully. When we refresh the page, we see that the order has been added to the list of placed orders.

The screenshot shows a web application interface for managing orders. At the top, there are system status messages: "Shifts: Mon-09:00-17:00,Tue-09:00-17:00", "Coordinator: Yes", and "Manager Email: TedBarnes@gmail.com". Below these are navigation links: "Order from Supplier", "Add Client Donations", "Add Admin Account", "Fulfill/View Requests", "View/Edit Inventory", "Verify/Check User", and "Logout".

Orders Section

Add Order

Delivery Date: 2023-12-14

Supplier: Walmart, id=0000000001

Items:

- Dad Cookies
- Kellogs Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

Add Order

Order added successfully!

Orders

Order No	Delivery Date	Admin Email	Supplier ID
3d5af866b4	December 29, 2023	SallyWon@gmail.com	0000000001

Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Order from Supplier | Add Client Donations | Add Admin Account | Fulfill/View Requests | View/Edit Inventory | Verify/Check User | Logout

Orders Section

Add Order

Delivery Date: 2023-12-11

Supplier: Select a supplier

Items:

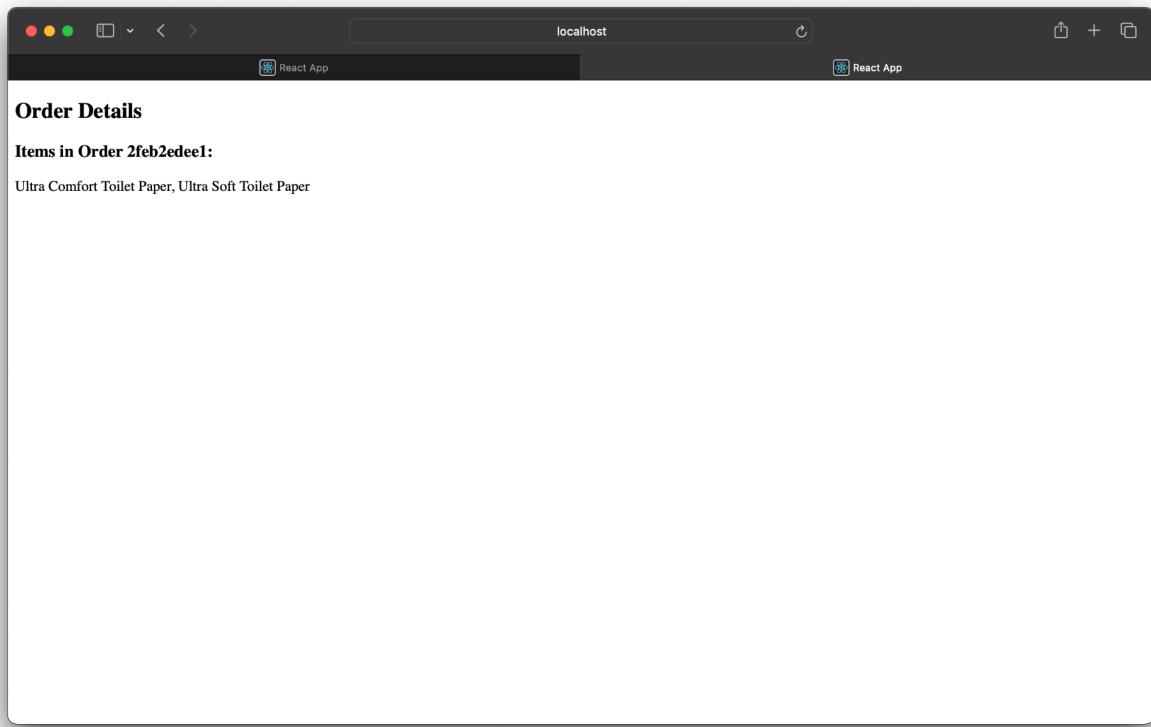
- Dad Cookies
- Kellogg's Corn Flakes
- Ultra Comfort Toilet Paper
- Ultra Soft Toilet Paper

[Add Order](#)

Orders

Order No	Delivery Date	Admin Email	Supplier ID
2feb2edee1	December 14, 2023	BobGaines@gmail.com	0000000001
3d5af866b4	December 29, 2023	SallyWon@gmail.com	0000000001

When we click on the highlighted “Order No”, a page is opened in a new tab which shows the correct items were added.



Returning to the admin dashboard, Bob can handle donations by clicking the “Add Client Donations” button. This brings up options as shown below. As before, the donation date features a convenient calendar widget. Once all the information is entered, pressing the “Add Donation” button registers the donation on the donor’s account.

The screenshot shows a web application running on localhost. The title bar says "localhost" and "React App".

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

[Order from Supplier](#) | [Add Client Donations](#) | [Add Admin Account](#) | [Fulfill/View Requests](#) | [View/Edit Inventory](#) | [Verify/Check User](#) | [Logout](#)

Client Donations Section

Donor Email:

Donation Date:

Select Item:

The screenshot shows the same web application on localhost.

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

[Order from Supplier](#) | [Add Client Donations](#) | [Add Admin Account](#) | [Fulfill/View Requests](#) | [View/Edit Inventory](#) | [Verify/Check User](#) | [Logout](#)

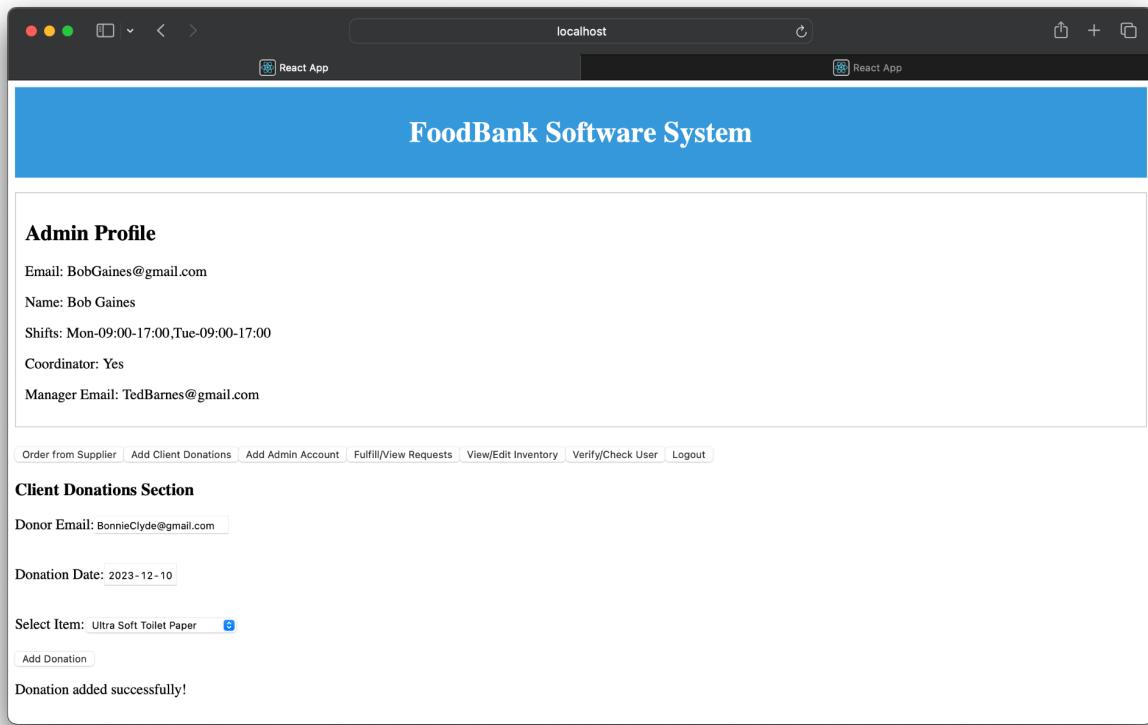
Client Donations Section

Donor Email:

Donation Date:

Select Item:

A date picker calendar for December 2023 is displayed, showing the days from 1 to 31. The 10th is highlighted in blue, indicating it is the selected date.



Now, suppose Bob wants to add a new volunteer to the system, so he clicks on “Add Admin Account”. He fills out all the fields of the following form, then presses submit, and a popup confirms that the account creation was a success.

localhost

FoodBank Software System

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

[Order from Supplier](#) | [Add Client Donations](#) | [Add Admin Account](#) | [Fulfill/View Requests](#) | [View/Edit Inventory](#) | [Verify/Check User](#) | [Logout](#)

Add Admin Account Section

Email: First Name: Last Name: Admin Type: Volunteer Shift Time: Manager Email:

localhost

FoodBank Software System

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

[Order from Supplier](#) | [Add Client Donations](#) | [Add Admin Account](#) | [Fulfill/View Requests](#) | [View/Edit Inventory](#) | [Verify/Check User](#) | [Logout](#)

Add Admin Account Section

Email: First Name: Last Name: Admin Type: Volunteer Shift Time: Manager Email:

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main content area is titled "Admin Profile" and lists the following information:

- Email: BobGaines@gmail.com
- Name: Bob Gaines
- Shifts: Mon-09:00-17:00,Tue-09:00-17:00
- Coordinator: Yes
- Manager Email: TedBarnes@gmail.com

A modal dialog box is open in the center-right, displaying the message "Add new admin successful!" with a "Close" button at the bottom.

At the bottom of the page, there is a navigation bar with links: Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

We can then log in to the website using Jack Black's email and verify his information:

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main content area is titled "Admin Profile" and lists the following information:

- Email: JackBlack@gmail.com
- Name: Jack Black
- Shifts: Wed-12:00-14:00
- Volunteer: Yes
- Manager Email: BobGaines@gmail.com

At the bottom of the page, there is a navigation bar with links: Order from Supplier, Add Client Donations, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

Returning to Bob's account, we move on to the fulfillment of requests by clicking "Fulfill/View Requests". Here we can see that John Doe has placed a request. We can click on the highlighted "Request ID" to see the items contained in the request in a new tab. When the hamper has been packed, we can click "Fulfill", and the hamper will be marked as ready as of the current date, and the admin account in use (Bob) will be noted.

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System".

Admin Profile:

- Email: BobGaines@gmail.com
- Name: Bob Gaines
- Shifts: Mon-09:00-17:00,Tue-09:00-17:00
- Coordinator: Yes
- Manager Email: TedBarnes@gmail.com

Request List:

Request ID	Request Admin	Request User	Ready Date	Request Date	Actions
be68305bb3		johndoe@gmail.com		December 9, 2023	<button>Fulfill</button>

Request Details

Items in Request be68305bb3:

Kellogg's Corn Flakes, Ultra Comfort Toilet Paper

FoodBank Software System

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

[Order from Supplier](#) | [Add Client Donations](#) | [Add Admin Account](#) | [Fulfill/View Requests](#) | [View/Edit Inventory](#) | [Verify/Check User](#) | [Logout](#)

Request List

Request ID	Request Admin	Request User	Ready Date	Request Date	Actions
be68305bb3	BobGaines@gmail.com	johndoe@gmail.com	December 9, 2023	December 9, 2023	Already Fulfilled

The screenshot shows a web application interface for the FoodBank Software System. At the top, a blue header bar displays the title "FoodBank Software System". Below the header, the main content area is divided into two sections: "Admin Profile" and "Item Data".

Admin Profile:

- Email: BobGaines@gmail.com
- Name: Bob Gaines
- Shifts: Mon-09:00-17:00,Tue-09:00-17:00
- Coordinator: Yes
- Manager Email: TedBarnes@gmail.com

Item Data:

Name	Quantity	Storage Type	Brand	Type	Action
Dad Cookies	35	Shelf	Mondelez	food	<input type="button" value="Update Quantity"/>
Kellogs Corn Flakes	13	Shelf	Kellogs	food	<input type="button" value="Update Quantity"/>
Ultra Comfort Toilet Paper	2	Shelf	Charmin	toiletry	<input type="button" value="Update Quantity"/>
Ultra Soft Toilet Paper	5	Shelf	Charmin	toiletry	<input type="button" value="Update Quantity"/>
				Toiletry <input checked="" type="checkbox"/>	<input type="button" value="Add New Item"/>

We now turn our attention once more to the inventory screen. Since Bob is an admin, he can also edit the inventory, instead of only viewing it. Let's say both kinds of toilet paper have run out. We can set both their quantities to 0 as below, then press the "Update Quantity" button.

localhost

FoodBank Software System

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Order from Supplier | Add Client Donations | Add Admin Account | Fulfill/View Requests | View/Edit Inventory | Verify/Check User | Logout

Item Data

Name	Quantity	Storage Type	Brand	Type	Action
Dad Cookies	35	Shelf	Mondelez	food	<button>Update Quantity</button>
Kellogs Corn Flakes	13	Shelf	Kellogg	food	<button>Update Quantity</button>
Ultra Comfort Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
Ultra Soft Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
				Toiletry	<button>Add New Item</button>

localhost

FoodBank Software System

Admin Profile

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Item quantity update successful!

[Close](#)

Order from Supplier | Add Client Donations | Add Admin Account | Fulfill

Item Data

Name	Quantity	Storage Type	Brand	Type	Action
Dad Cookies	35	Shelf	Mondelez	food	<button>Update Quantity</button>
Kellogs Corn Flakes	13	Shelf	Kellogg	food	<button>Update Quantity</button>
Ultra Comfort Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
Ultra Soft Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
				Toiletry	<button>Add New Item</button>

If we want to add an item, we can easily do so as follows:

The screenshot shows a web application window titled "FoodBank Software System" running on "localhost".

Admin Profile:

- Email: BobGaines@gmail.com
- Name: Bob Gaines
- Shifts: Mon-09:00-17:00,Tue-09:00-17:00
- Coordinator: Yes
- Manager Email: TedBarnes@gmail.com

Item Data:

Name	Quantity	Storage Type	Brand	Type	Action
Dad Cookies	35	Shelf	Mondelez	food	<input type="button" value="Update Quantity"/>
Kellogs Corn Flakes	13	Shelf	Kellogs	food	<input type="button" value="Update Quantity"/>
Ultra Comfort Toilet Paper	0	Shelf	Charmin	toiletry	<input type="button" value="Update Quantity"/>
Ultra Soft Toilet Paper	0	Shelf	Charmin	toiletry	<input type="button" value="Update Quantity"/>
Apples	40	Fridge	Farm	Food	<input type="button" value="Add New Item"/>

The screenshot shows a web application interface titled "localhost". At the top, there's a blue header bar. Below it, the main content area has a white background. On the left, a sidebar contains the title "Admin Profile" and several user details: Email: BobGaines@gmail.com, Name: Bob Gaines, Shifts: Mon-09:00-17:00,Tue-09:00-17:00, Coordinator: Yes, and Manager Email: TedBarnes@gmail.com. Below the sidebar is a horizontal navigation bar with links: Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout. The main content area is titled "Item Data" and features a table with columns: Name, Quantity, Storage Type, Brand, Type, and Action. The table lists items like Apples, Dad Cookies, Kellogg's Corn Flakes, Ultra Comfort Toilet Paper, and Ultra Soft Toilet Paper. Each row includes an "Update Quantity" button and an "Add New Item" button. The "Action" column for the last row shows a "Toiletry" category with a small icon.

Name	Quantity	Storage Type	Brand	Type	Action
Apples	40	Fridge	Farm	food	<button>Update Quantity</button>
Dad Cookies	35	Shelf	Mondelez	food	<button>Update Quantity</button>
Kellogg's Corn Flakes	0	Shelf	Kellogg's	food	<button>Update Quantity</button>
Ultra Comfort Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
Ultra Soft Toilet Paper	0	Shelf	Charmin	toiletry	<button>Update Quantity</button>
				Toiletry	<button>Add New Item</button>

Lastly, we can verify a user with the “Verify/Check User” button. This allows admins to verify client accounts or check if they are already verified.

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main content area is titled "Admin Profile". It contains the following information:

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Below this, a navigation bar includes links for Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

A bold header "Enter User Email to verify their account or check if they are already verified." is displayed. Below it is a form with a "User Email:" input field and a "Verify User" button.

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main content area is titled "Admin Profile". It contains the same information as the first screenshot:

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Below this, a navigation bar includes links for Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

A bold header "Enter User Email to verify their account or check if they are already verified." is displayed. Below it is a form with a "User Email:" input field containing "JaneDoe@gmail.com" and a "Verify User" button.

A message "User is already verified." is displayed below the form.

The screenshot shows a web browser window titled "localhost" displaying the "FoodBank Software System". The main content area is titled "Admin Profile". It contains the following information:

Email: BobGaines@gmail.com
Name: Bob Gaines
Shifts: Mon-09:00-17:00,Tue-09:00-17:00
Coordinator: Yes
Manager Email: TedBarnes@gmail.com

Below this, a navigation bar includes links for Order from Supplier, Add Client Donations, Add Admin Account, Fulfill/View Requests, View/Edit Inventory, Verify/Check User, and Logout.

A bold header "Enter User Email to verify their account or check if they are already verified." is followed by a text input field containing "User Email: BonnieClyde@gmail.com" and a "Verify User" button.

A success message "Verification updated successfully!" is displayed below the button.

APPENDIX

User

u_email	fname	lname	client_flag	address	num_visits	verify_email	donor_flag
BonnieClyde@gmail.com	Bonnie	Clyde	0	NULL	0	NULL	1
DonaldFrump@gmail.com	Donald	Frump	0	NULL	NULL	SallyWon@gmail.com	1
JaneDoe@gmail.com	Jane	Doe	1	12,Evergarden,Road	3	TedBarnes@gmail.com	0
JimJong@gmail.com	Jim	Jong	0	NULL	0	NULL	1
JohnDoe@gmail.com	John	Doe	1	12,Evergarden,Road	2	SallyWon@gmail.com	0
MattLow@gmail.com	Matt	Low	0	NULL	0	NULL	1
TomJerry@gmail.com	Tom	Jerry	1	15,Grandville,Crescent	0	NULL	0

Admin

a_email	fname	lname	shift	coordinator_flag	volunteer_flag	mgr_email
BobGaines@gmail.com	Bob	Gaines	Mon-09:00-17:00,Tue-09:00-17:00	1	0	TedBarnes@gmail.com
SallyWon@gmail.com	Sally	Won	Tues-08:00-18:00	0	1	TedBarnes@gmail.com
TedBarnes@gmail.com	Ted	Barnes	Mon-09:00-17:00,Tue-09:00-17:00	1	0	NULL

Item

item_name	quantity	storage_type	brand	food_flag	toiletry_flag
Dad Cookies	31	Shelf	Mondelez	1	0
Kellogs Corn Flakes	13	Shelf	Kellogs	1	0
Ultra Comfort Toilet Paper	3	Shelf	Charmin	0	1
Ultra Soft Toilet Paper	5	Shelf	Charmin	0	1

Request

request_id	request_admin	request_user	pickup_date	request_date
be68305bb3	NULL	iohn doe@gmail.com	NULL	2023-12-10

Request_contains

id_request	request_item
be68305bb3	Kellogs Corn Flakes
be68305bb3	Ultra Comfort Toilet Paper

Donation

donor_email	item	donation_date
DonaldFrump@gmail.com	Dad Cookies	2023-12-01
DonaldFrump@gmail.com	Dad Cookies	2023-12-05
BonnieClyde@gmail.com	Ultra Comfort Toilet Paper	2023-12-10
DonaldFrump@gmail.com	Ultra Comfort Toilet Paper	2023-12-01
BonnieClyde@gmail.com	Ultra Soft Toilet Paper	2023-12-10

Order

order_no	delivery_date	admin_email	supplier_id
3d5af866b4	2023-12-29	SallyWon@gmail.com	1

Order_contains

order_no	item_name
3d5af866b4	Dad Cookies
3d5af866b4	Kellogs Corn Flakes
3d5af866b4	Ultra Comfort Toilet Paper

Supplier

supplier_id	name	location
0000000001	Wal mart	Calgary
0000000002	Cos tco	Airdrie
0000000003	Sob eys	Okotoks

References

How to Create a Flask + React Project | Python Backend + React Frontend (Tutorial)

<https://www.youtube.com/watch?v=7LNI2JIZKHA>

<https://legacy.reactjs.org/docs/getting-started.html>

<https://www.askpython.com/python-modules/flask/flask-mysql-database>

<https://flask.palletsprojects.com/en/3.0.x/>