Group Project Log

Group Name:	Group 6

Group Members:	Arpit Ribadiya, Jay Kania, Lav Patel, Neha Karkhanis, Viraj Joshi

Deliverable:	LastServe project proposal and Front-end code (Via Gitlab)

Group Member Name	**Work Done (%)**
Arpit Ribadiya	20%
Jay Kania	20%
Lav Patel	20%
Neha Karkhanis	20%
Viraj Joshi	20%
Total:	**100%**

PROJECT PROPOSAL

PROJECT GROUP 6

LastServe: An initiative to eradicate world hunger by establishing a conduit for restaurants to give extra food to the hungry.

Members and Contributors

Arpit Ribadiya
B00932018
ar304626@dal.ca

Jay Kania
B00923785
jy440982@dal.ca

Lav Patel
B00910579
lv842182@dal.ca

Neha Karkhanis
B00903157
nh601176@dal.ca

Viraj Joshi
B00924759
viraj.joshi@dal.ca

Faculty of Computer Science
Dalhousie University

28 February 2023

CONTENTS

1. PROJECT BACKGROUND

In Halifax, there are a disproportionately large number of underpreviledged individuals, students, and people in need. Access to food is perceived by us as a fundamental human right, not a privilege. No one ought to struggle to acquire enough food each day or go to bed hungry. With the help of this project, eateries will be able to feed the needy in their local neighborhood.

Restaurants that have leftover food will update their public posting on our website each night with how much food is still available. Anyone in need of food can do so by setting up a time slot and coming to pick up their meal for that evening. By doing this, food waste in restaurants will be reduced and people won't go to bed hungry.

1.1 Project Objectives

The purpose of this initiative is to lessen the amount of food that restaurants discard after use. Food that hasn't been sold or is about to expire falls under this category. 1.3 billion tons of food is wasted each year, according to the most recent assessment [6]. Recent studies reflect $162 billion in food waste-related costs and $2 billion in lost profits annually [4]. Out-of-home waste such as that produced at restaurants, eateries, canteens, snack bars, etc. is the second chief source of food waste [5]. Saving just a quarter of food lost or wasted globally each year would feed 870 million people. This project aims to distribute food that would otherwise be wasted and provide it to those in need.

Additionally, through our website, we aim to instill in people the mindset to donate and make a difference because we believe there is enough on the planet for everyone’s needs but not for everyone’s greed. LastServe makes it easy for restaurants to register and distribute leftover food that would otherwise be wasted.

1.2 Live Project URL

Git Lab repository for the LastServe code: https://git.cs.dal.ca/ribadiya/csci_5709_web_group6.git

Live project URL: https://last-serve.netlify.app/

Group#_6

2. APPLICATION DETAILS

2.1 Target User Insight

This website mostly caters to individuals on a restricted budget, restaurant owners, and volunteers. Those on a limited budget, such as students, the unemployed, and those with limited time are our potential users. Restaurant proprietors make up the second group of users. The restaurant would be listed on the website, and posts would be updated daily, with the amount of food still available. Those who have additional time on their hands and can assist restaurants in providing food to the hungry would make an additional group of users. The administrators are LastServe coordinators who oversee the site's user base and content. They will be provided with a unique set of tools that will allow them to do this by authorizing user registrations, examining, or deleting restaurant material as needed, and managing user engagement.

The website was designed to be very user-friendly and to take a short time to finish a task. The forms are created so that they are simple to read and just gather the necessary information without requesting excessive amounts of personal information. We kept the color scheme subdued to accommodate all types of users in order to appeal to emotions and communicate the essence of our website.

2.2 Brand Attributes

As an organization, LastServe wants to emphasize the fact that we support the underprivileged and rely on restaurants to provide food for the needy. The vast majority of the material is on the landing page to ensure that consumers understand our initiative. For our objectives to be understood, a new user wouldn't need to read in-depth. In order to further communicate our website's motto visually, we have also incorporated photos.

The development of trust and the instillation of the value of giving are crucial features because our website also asks visitors to donate money to restaurants so they can provide more meals for the hungry. Since students make up the majority of our user base, LastServe not only benefits the poor but also aids partner restaurants in developing a reputation among young people. It aids in both restaurant promotion and food waste prevention.

2.3 Competitive Landscape

The free web platform LastServe is available to all types of consumers that require food on the go. Providing free services to both sides, LastServe serves as an intermediary between restaurants and customers.

LastServe attempts to alleviate world hunger by collaborating with restaurants. Besides providing food for the underprivileged, LastServe provides various additional services.

Existing competitors for our brand would be **Food Rescue, Share the meal, Flash food, and Good to go.**

The features that make our website different are listed below:

- **Overall Layout and Design:** LastServe is user-friendly, with conveniently accessible components and straightforward signup processes. All of the forms are designed in such a manner that the user will feel secure in providing their information without fear of violating their confidentiality. Navigating between each component is simple since we implemented a consistent layout and design components to avoid confusing the user. Our website's color scheme is subdued to provide an already hungry consumer with a calm sensation.

- **Speed:** Our website administrators are adopting a NoSQL database that will be kept on-site, resulting in speedier query results. In addition, the site is constructed in such a way that users may reach the final point in three clicks, saving them time.

- **Mission-specific and service-oriented:** Our objective is to help alleviate global hunger by reducing food waste generated by restaurants. We think that even if we cannot help everyone, we can help someone. Unlike other restaurants, LastServe serves cooked cuisine rather than raw materials. This saves time and money for individuals who are short on time.

- **User base:** We have a very diverse group of users, from students to those who are underprivileged, and they come from different backgrounds and age ranges.
  Restaurant managers can help save food by feeding the hungry what would otherwise go to trash. LastServe also provides people with the opportunity to assist in food distribution programs and earn good deeds.

- **Posts are updated daily**: Participating restaurants will update their posts on a daily basis with the amount of food available for order and booking. The admin additionally reviews and approves these posts to ensure that the information is appropriate and available on time. Posts are also purged on a regular basis to avoid any form of cluster and misunderstanding.

2.4 Project Scope

2.4.1 Project Goals

The Goals of our project are as follows:

- To aid in the reduction of world hunger.
- To provide a platform for restaurants to reduce food waste by registering and posting about leftover food for individuals in need.
- To instill a sense of donation and the power of giving in those who can afford it.
- To provide a platform for gathering volunteers and matching volunteers with restaurants in need of help running the food distribution initiative.

2.4.2 Project Features

1. **User Profile Management**: To use the services offered by LastServe, a user must register on the website. They can check their user profile and, if necessary, amend it. After creating a profile, users can log in with a newly generated username and password to continue utilising the services as needed.
2. **Restaurant Profile Management:** Our website's main goal is to work with restaurants to provide food for those in need. A restaurant will register on LastServe.ca as a partner restaurant for this reason. After a restaurant registers, it can browse and update its profile as well as post on LastServe's public forum to let customers know what food is available and allow them to reserve slots. A restaurant management can also see customer orders, timeslots, and the menu items they have reserved. The status of the order will subsequently be changed by a restaurant manager from pending to packet to ready for pickup. On our website, a restaurant can also post a request for volunteers to assist with the meal service.
3. **User order management:** By logging in and registering, a user can now reserve slots based on their free time and the food's availability. All of the current postings and donors are visible to them. Consumers can schedule pick-up times and request emails after their orders have been confirmed.
4. **Admin Content Management:** An administrator gets a unique dashboard for visualising data. Through their dashboard, they can control every operation. Super powers allow an admin to remove any undesired posts. The user base may be managed, and they can view any recent donations made on LastServe. Depending on the restaurant's content, an administrator may also accept or reject a restaurant application. They can examine all user information, restaurant information, donor information, and volunteer information.

5. **Volunteer Management:** A volunteer is a person who signs up as a user on LastServe. They will be asked for their contact information so that the restaurant where they fit in can get in touch with them. An administrator will authorise a volunteer, and they will then be linked with suitable restaurants.
6. **Donation Management:** LastServe links hungry people and restaurants. With the goal of providing leftover food, we work with restaurants. Also, LastServe provides a platform for non-restaurant donors. Donors are at liberty to contribute any amount they like.
7. **Subscription Management**: Customers have the option of subscribing to restaurants nearby or that they enjoy. They also have the power to opt out if they so want.
8. **Order History Management:** The user has the option to view their prior orders and cancel any active orders. The order history dashboard also shows the status of the customer's current order.

2.5 Information Architecture

2.5.1 Proposed Sitemap

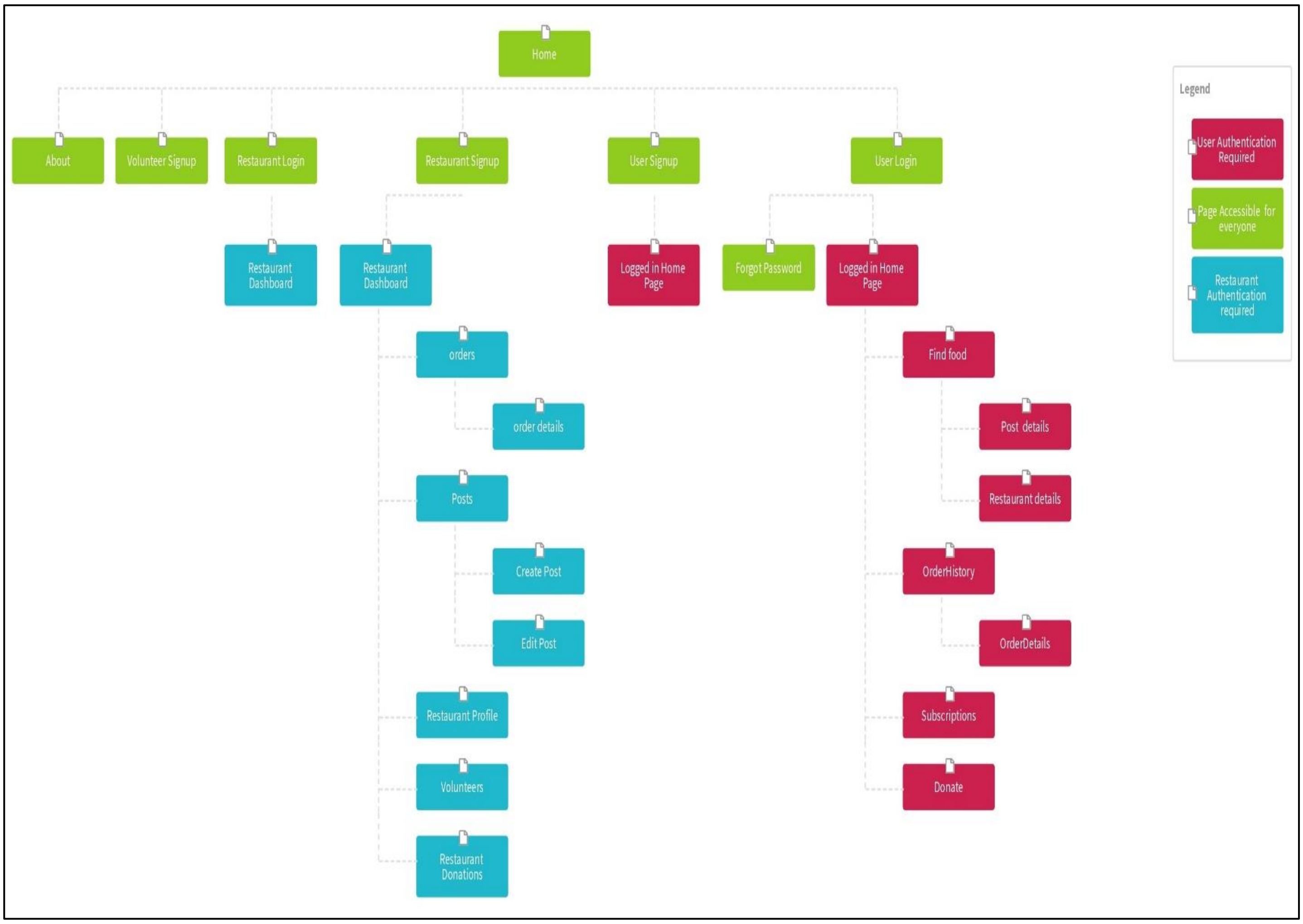
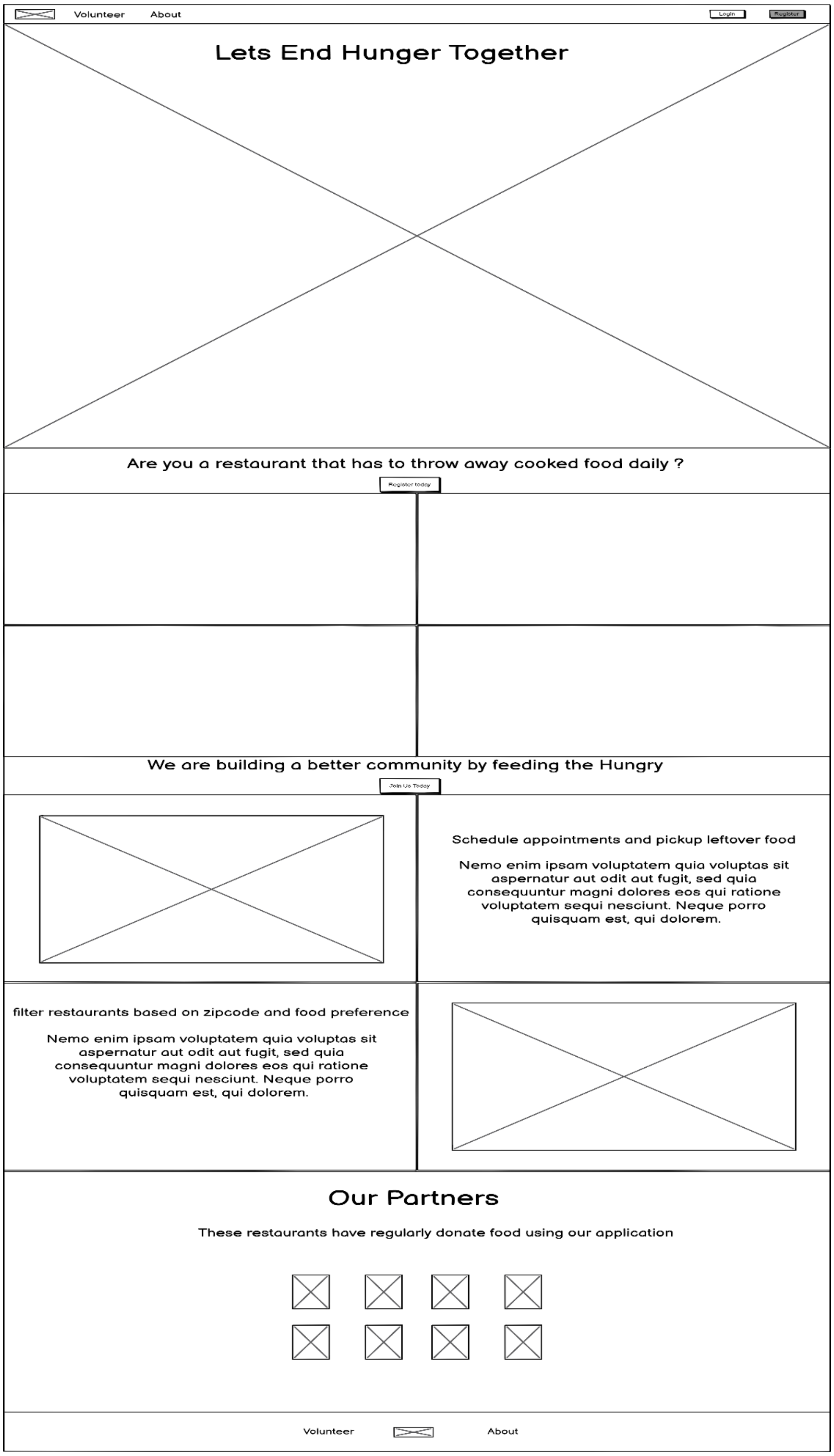


Figure 1: Sitemap for LastServe, Created using Writemaps [3].

2.5.2 Proposed Wireframes



Lets End Hunger Together

Are you a restaurant that has to throw away cooked food daily ?

We are building a better community by feeding the Hungry

Schedule appointments and pickup leftover food

Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem.

filter restaurants based on zipcode and food preference

Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem.

Our Partners

These restaurants have regularly donate food using our application

Figure 2: A wireframe for Home Page, Created using Balsamiq.cloud [4].

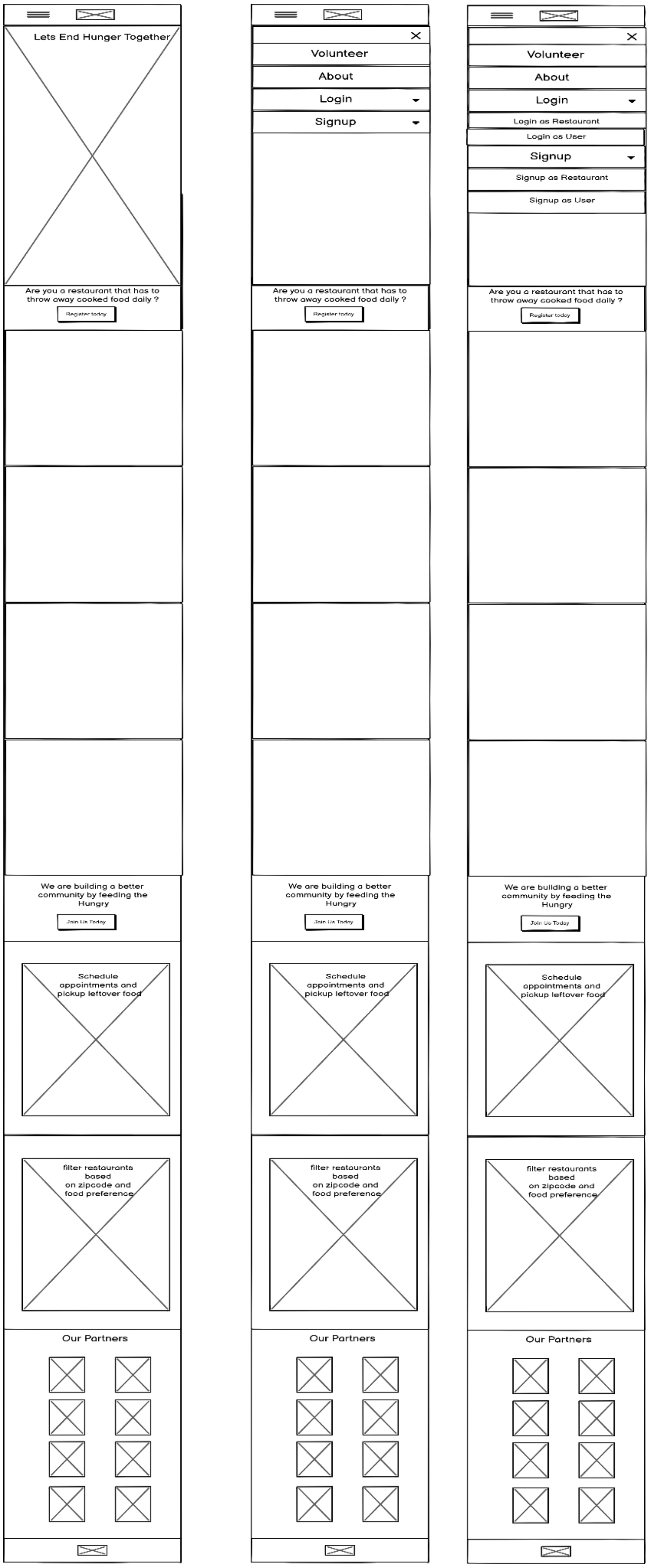

Lets End Hunger Together
Are you a restaurant that has to throw away cooked food daily ?
Register today
We are building a better community by feeding the Hungry
Join Us Today
Schedule appointments and pickup leftover food
filter restaurants based on zipcode and food preference
Our Partners
Volunteer
About
Login
Signup
Are you a restaurant that has to throw away cooked food daily ?
Register today
We are building a better community by feeding the Hungry
Join Us Today
Schedule appointments and pickup leftover food
filter restaurants based on zipcode and food preference
Our Partners
Volunteer
About
Login
Login as Restaurant
Login as User
Signup
Signup as Restaurant
Signup as User
Are you a restaurant that has to throw away cooked food daily ?
Register today
We are building a better community by feeding the Hungry
Join Us Today
Schedule appointments and pickup leftover food
filter restaurants based on zipcode and food preference
Our Partners

Figure 3: A wireframe for a Responsive mobile device, Created using Balsamiq.cloud [4].

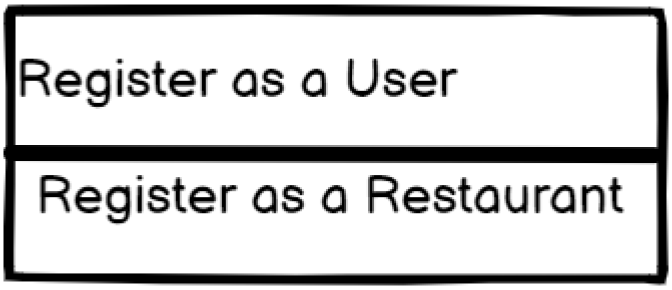


Figure 4: A wireframe for a user to choose which login do they want, Created using Balsamiq.cloud [4] .

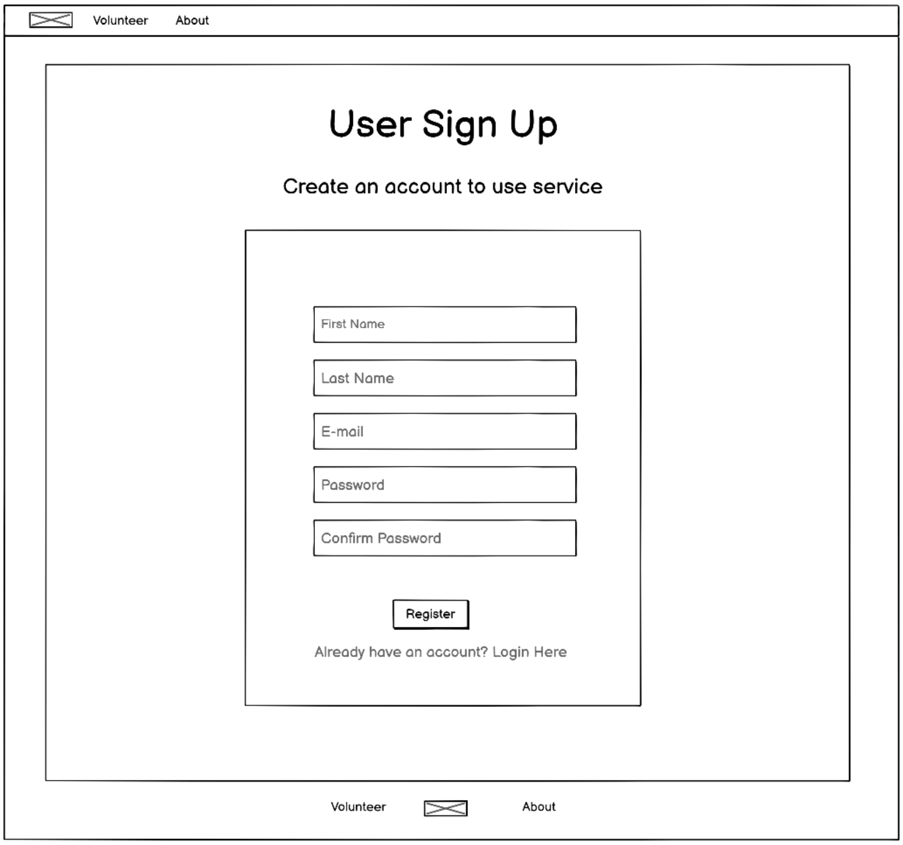
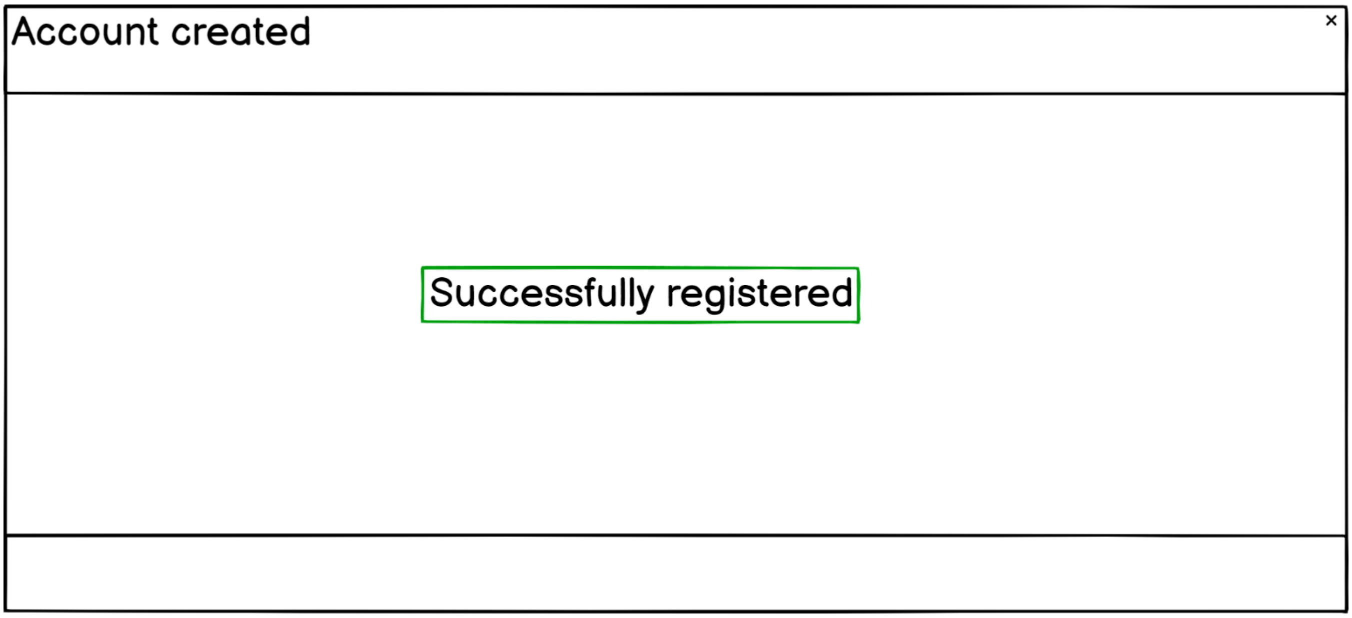


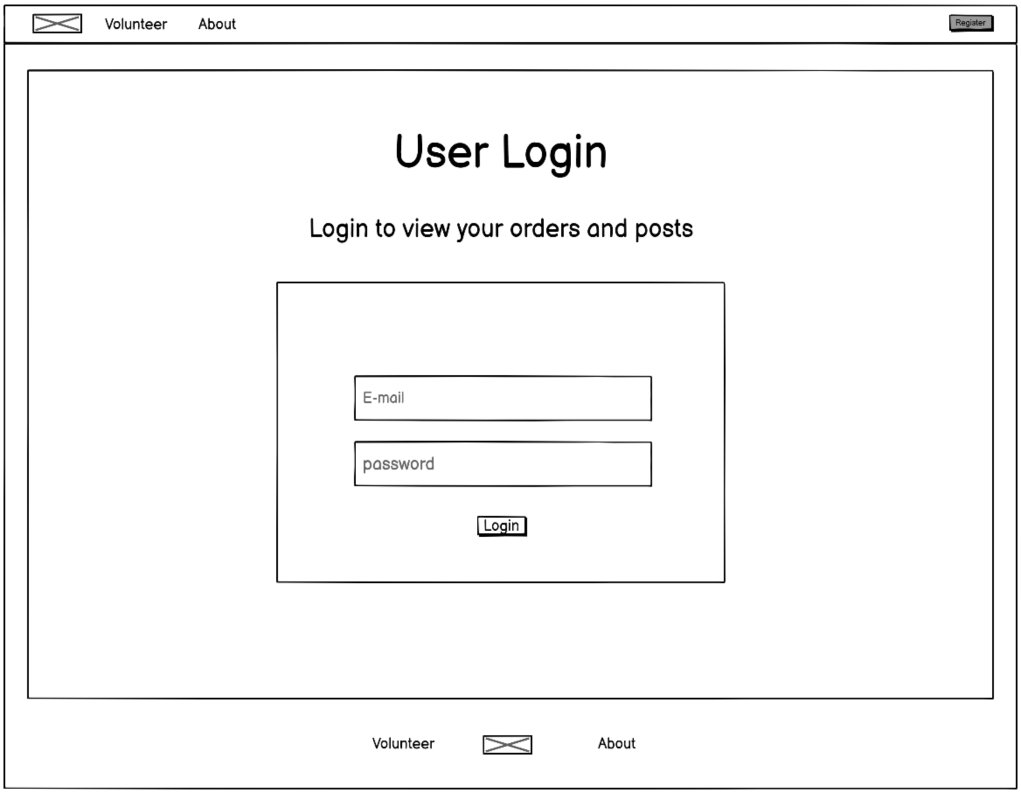
Figure 5: A wireframe for User sign up, which a user lands on after clicking on “Register as a User”, Created using Balsamiq.cloud [4].



Account created

Successfully registered

Figure 6: A wireframe for Successful login after a user successfully signs up on LastServe, Created using Balsamiq.cloud [4].



User Login

Login to view your orders and posts

Figure 7: A wireframe for user to login in into LastServe after successful registration, Created using Balsamiq.cloud [4].

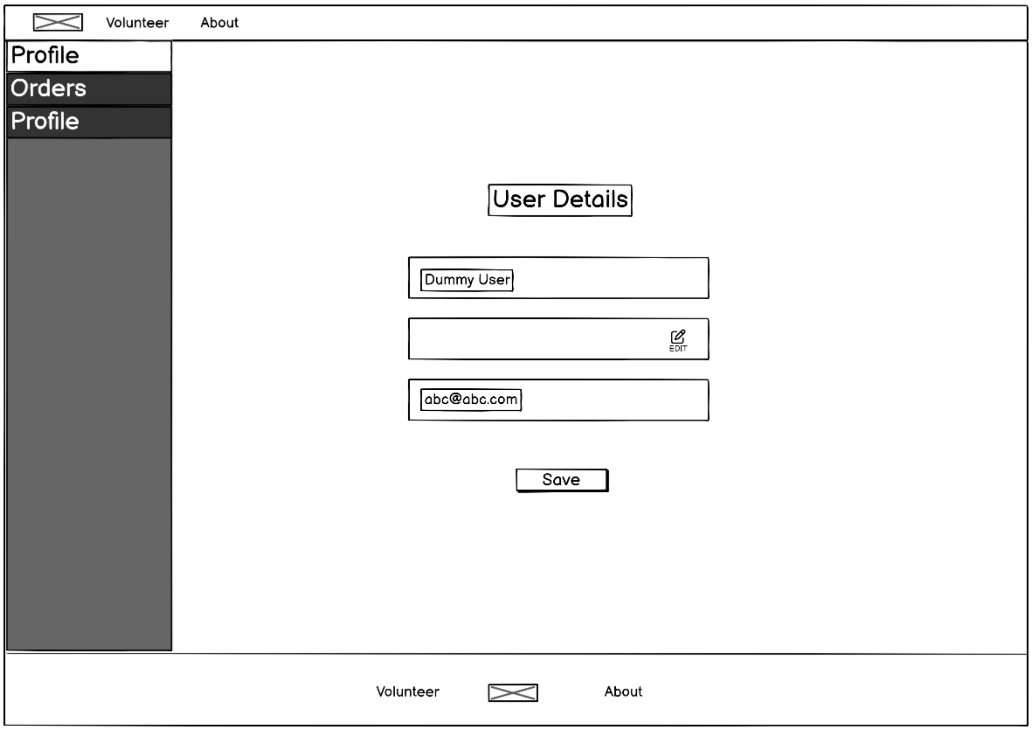


Figure 8: A wireframe for User details on the profile tab for user viewing, Created using Balsamiq.cloud [4].

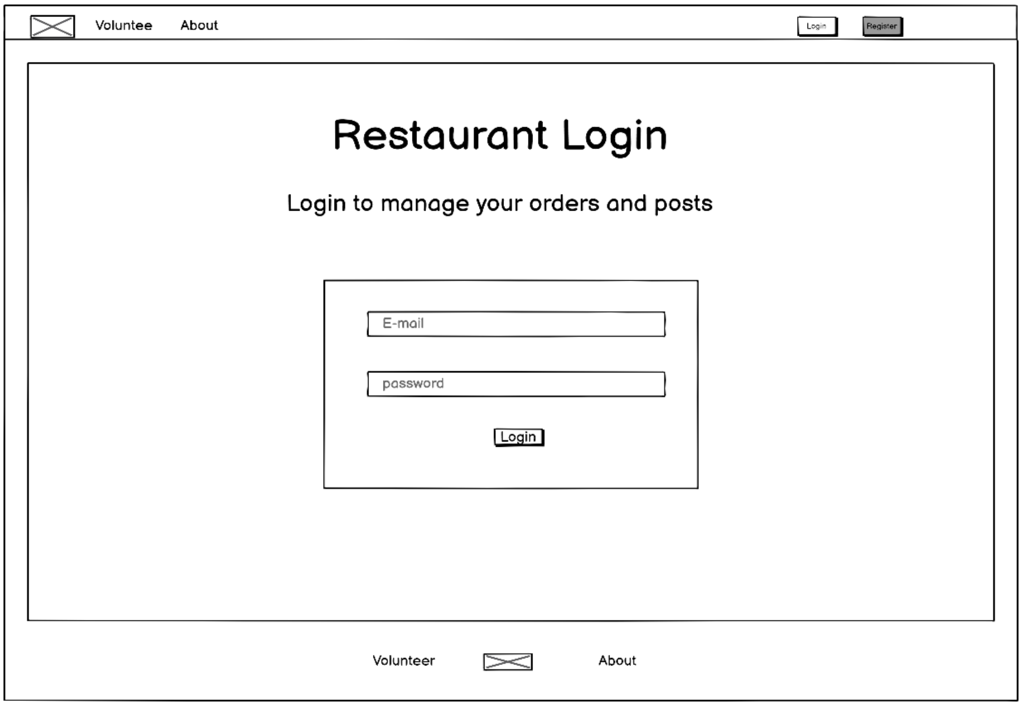
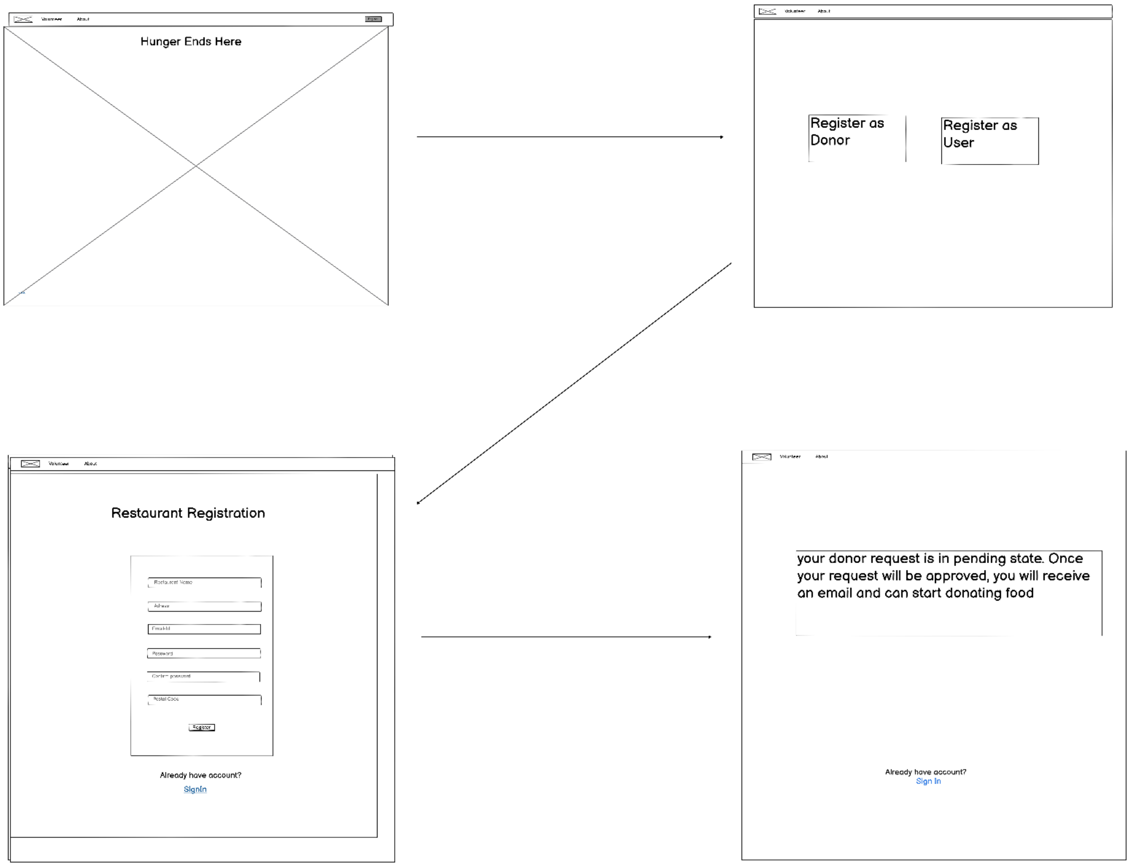


Figure 9: A wireframe for Restaurant login, after the user chooses to login as a Restaurant owner, Created using Balsamiq [4].



your donor request is in pending state. Once your request will be approved, you will receive an email and can start donating food

Figure 10: A wireframe for the entire registration process for a Restaurant, Created using Balsamiq.cloud [4].

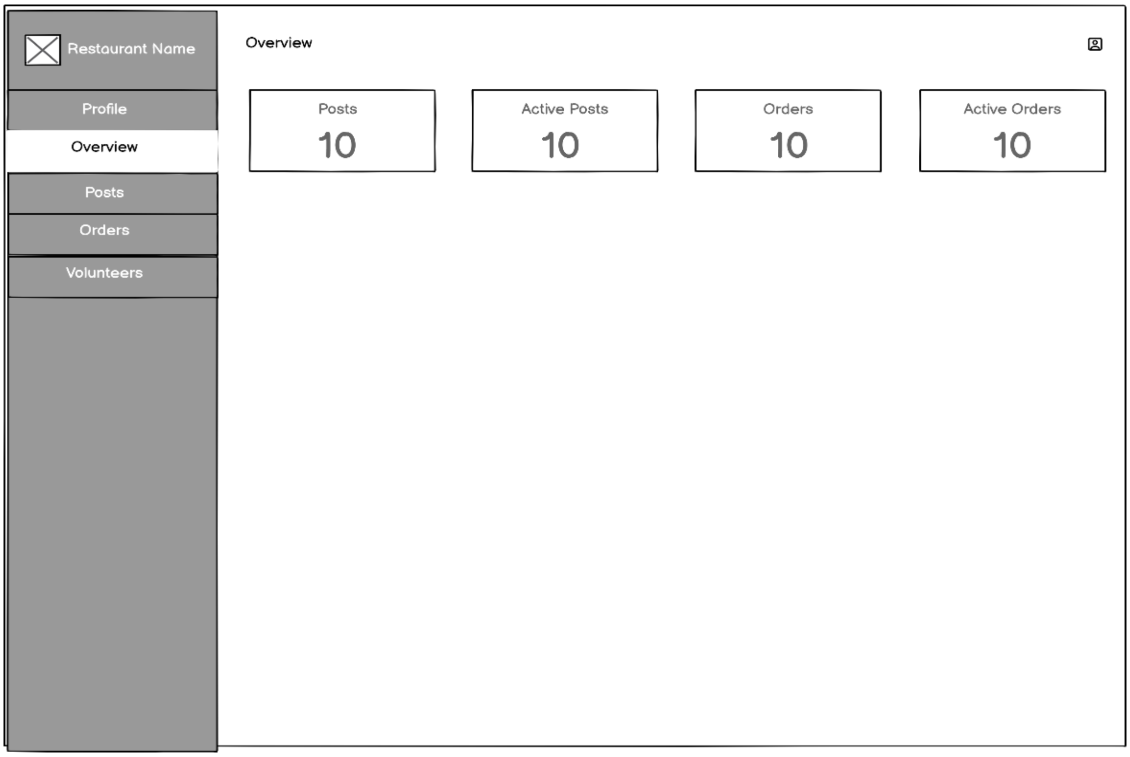


Figure 11: A wireframe for Restaurant Dashboard that provides an overview of all the details about active posts, orders and active orders, Created using Balsamiq.cloud [4].

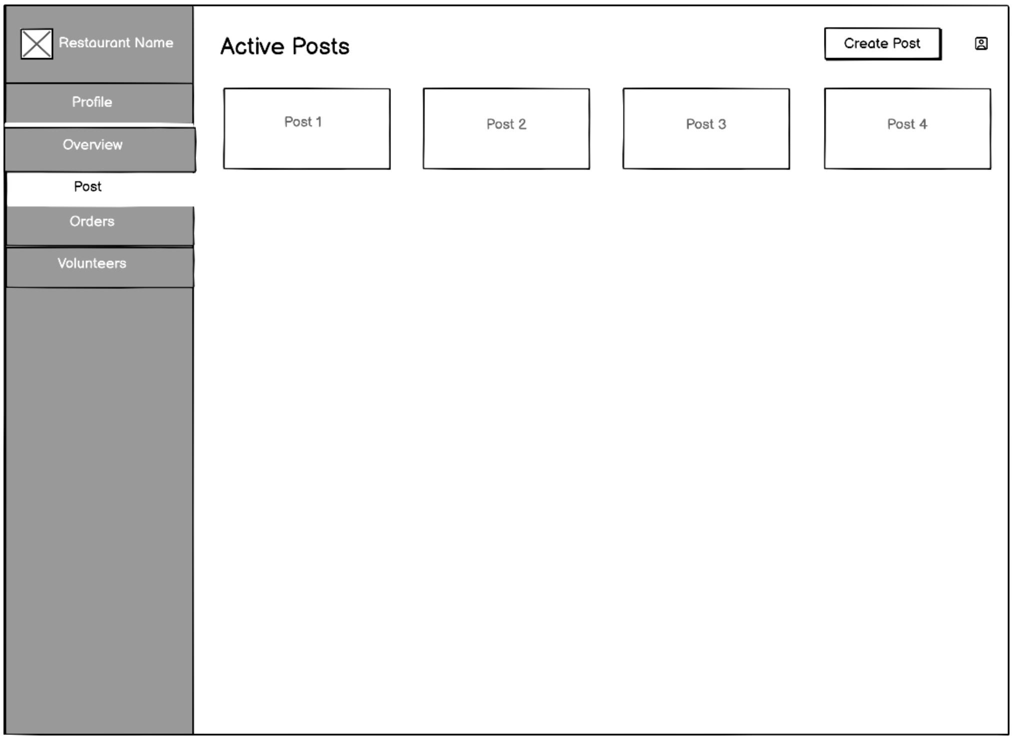
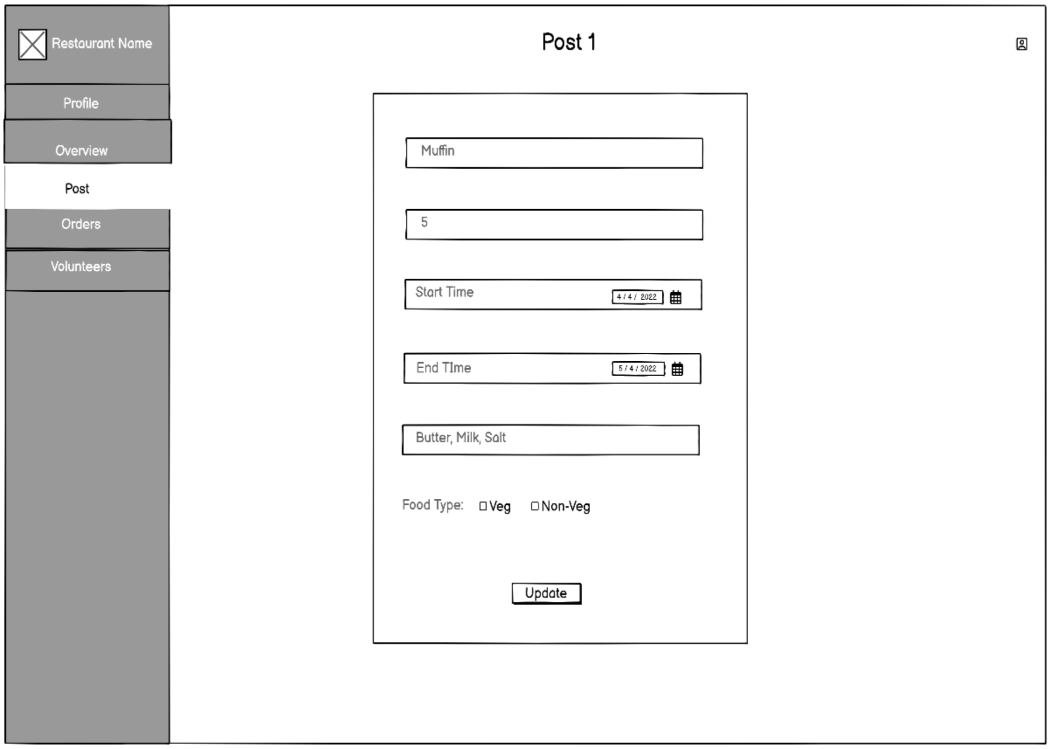


Figure 12: A wireframe for Active posts that can be viewed by the Restaurant owner after clicking on the post card, Created using Balsamiq.cloud [4].



Muffin

5

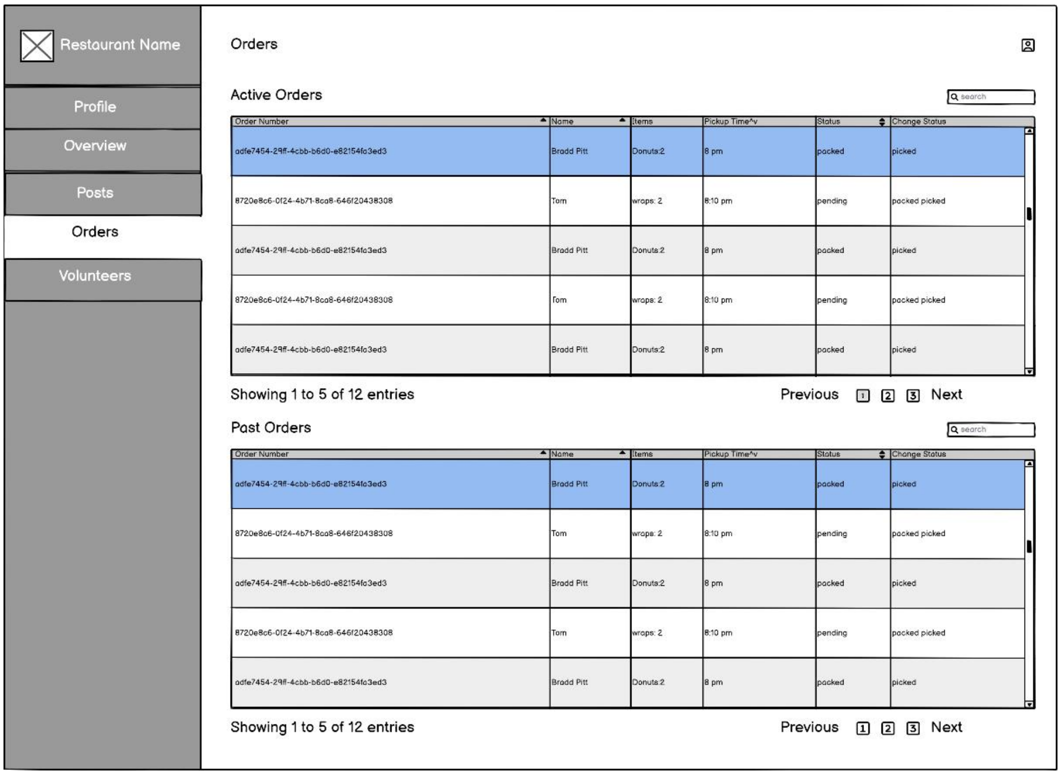
Start Time 4 / 4 / 2022

End TIme 5 / 4 / 2022

Butter, Milk, Salt

Food Type: □ Veg □ Non-Veg

Figure 13: A wireframe for Restaurant owner to view details of a particular post, after clicking on the active posts card, Created using Balsamiq.cloud [4].



Orders

Active Orders

Order Number	Name	Items	Pickup Time^v	Status	Change Status
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps: 2	8:10 pm	pending	packed picked
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps: 2	8:10 pm	pending	packed picked
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked

Showing 1 to 5 of 12 entries

Past Orders

Order Number	Name	Items	Pickup Time^v	Status	Change Status
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps: 2	8:10 pm	pending	packed picked
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps: 2	8:10 pm	pending	packed picked
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm	packed	picked

Showing 1 to 5 of 12 entries

Figure 14: A wireframe for currently active orders and past orders which can only be viewed by the restaurant owner to keep a track, Created using Balsamiq.cloud [4].

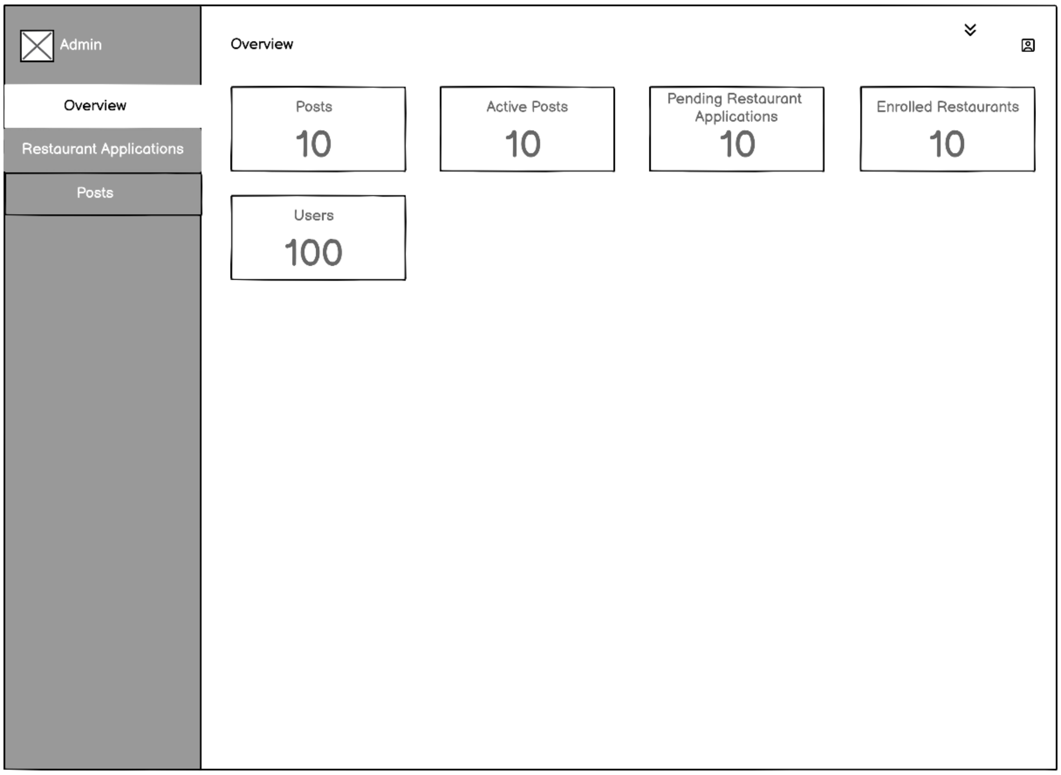
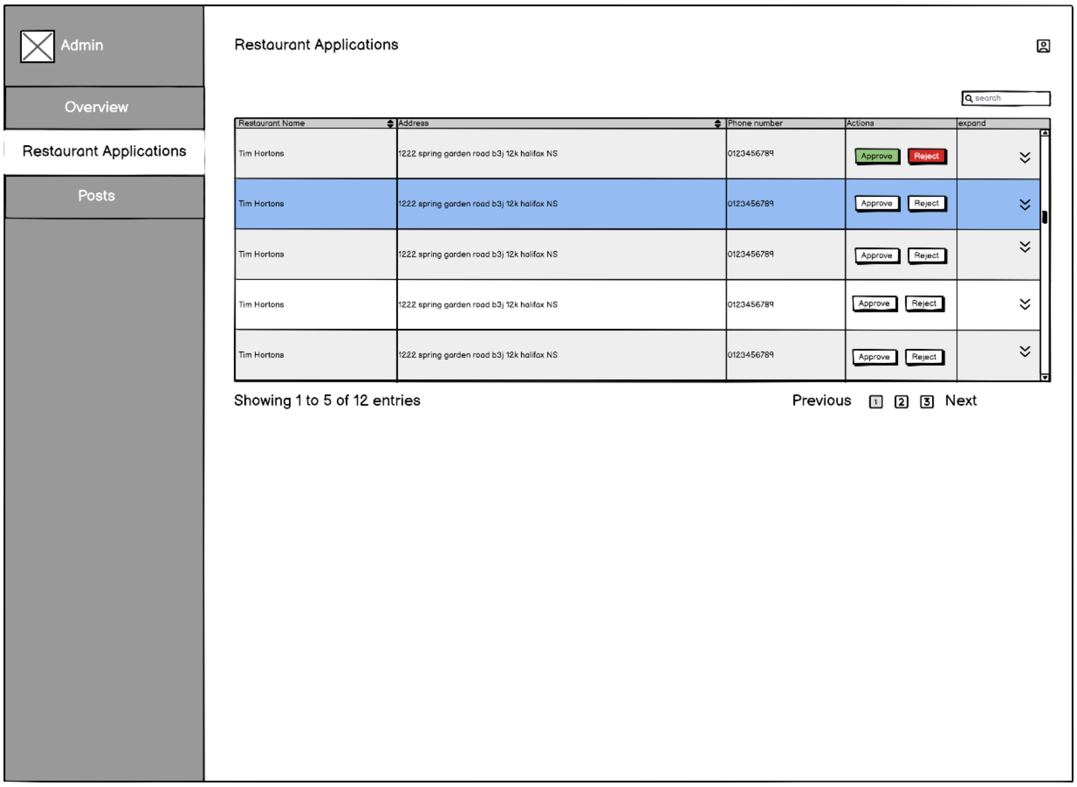
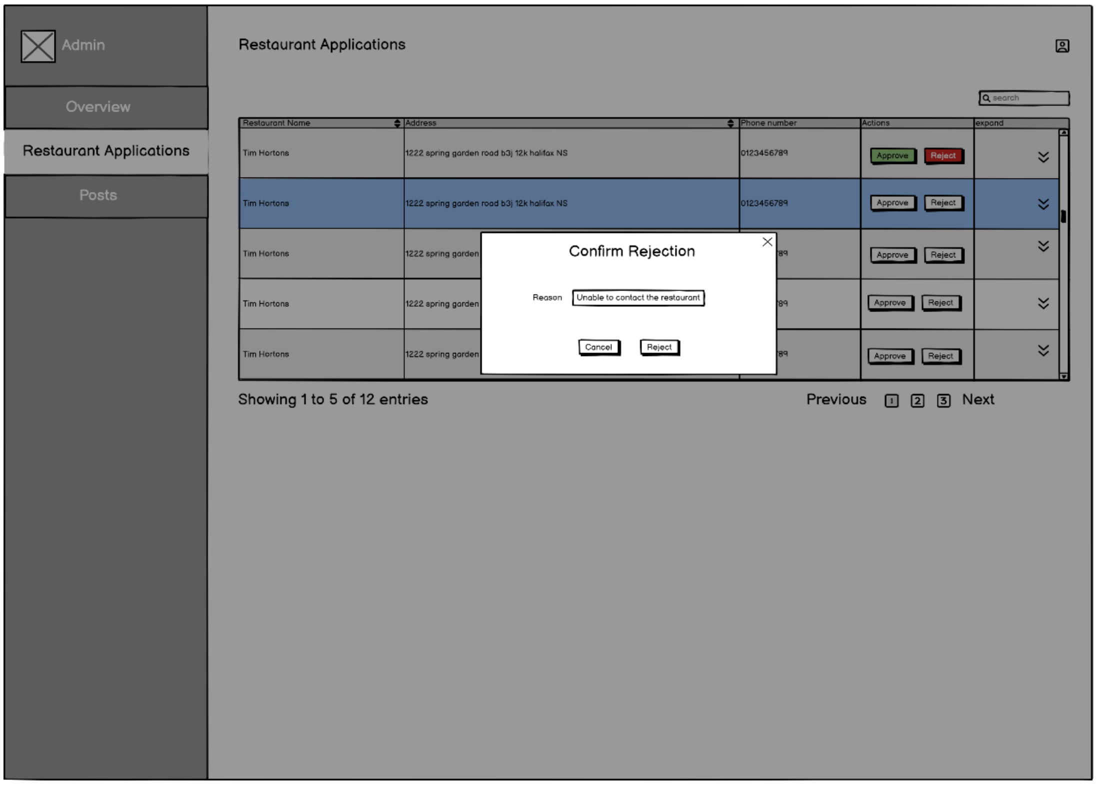


Figure 15: A wireframe for Admin dashboard, to manage all admit related tasks i.e to view all posts, to approve/reject restaurant application and to manage user base, Created using Balsamiq.cloud [4].



Restaurant Name	Address	Phone number	Actions	expand
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	

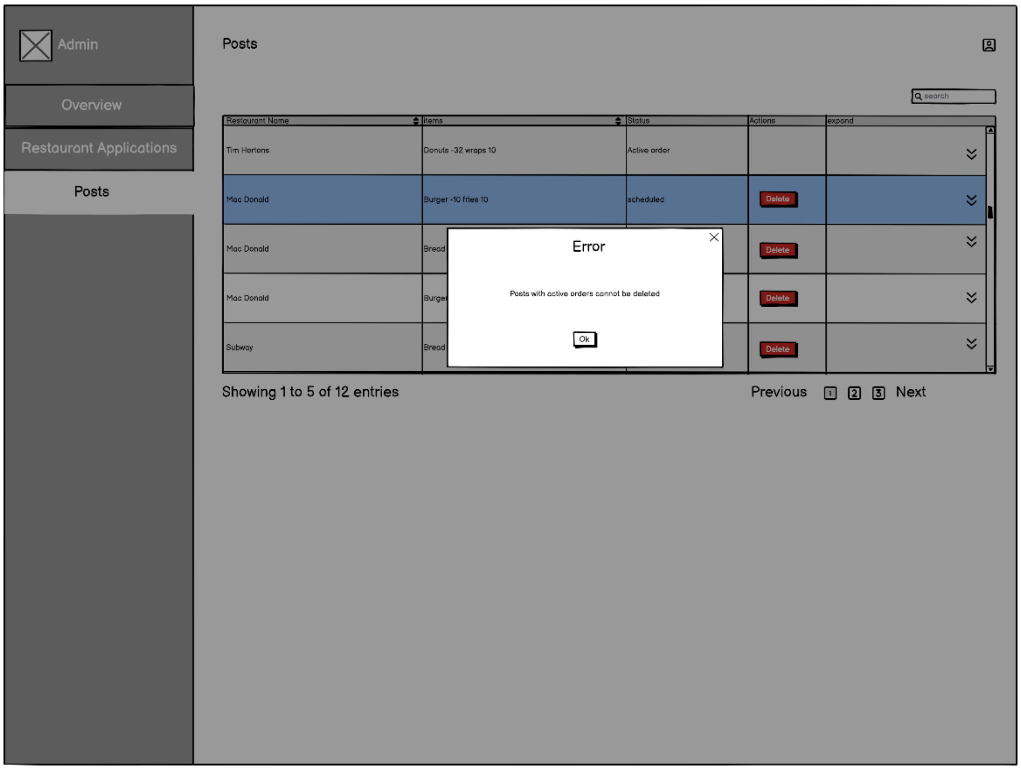
Figure 16: A wireframe of Admin dashboard to approve/reject Restaurant applications, Created using Balsamiq.cloud [4].



Reason: Unable to contact the restaurant

Restaurant Name	Address	Phone number	Actions	expand
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden road b3j 12k halifax NS	0123456789	Approve Reject	
Tim Hortons	1222 spring garden	89	Approve Reject	
Tim Hortons	1222 spring garden	89	Approve Reject	
Tim Hortons	1222 spring garden	89	Approve Reject	

Figure 17: A wireframe of a restaurant being rejected by admin due to certain reasons, Created using Balsamiq.cloud [4].



Restaurant Name	items	Status	Actions	expand
Tim Hortons	Donuts -32 wraps 10	Active order		⌄
Mac Donald	Burger -10 fries 10	scheduled	Delete	⌄
Mac Donald	Bread		Delete	⌄
Mac Donald	Burger		Delete	⌄
Subway	Bread		Delete	⌄

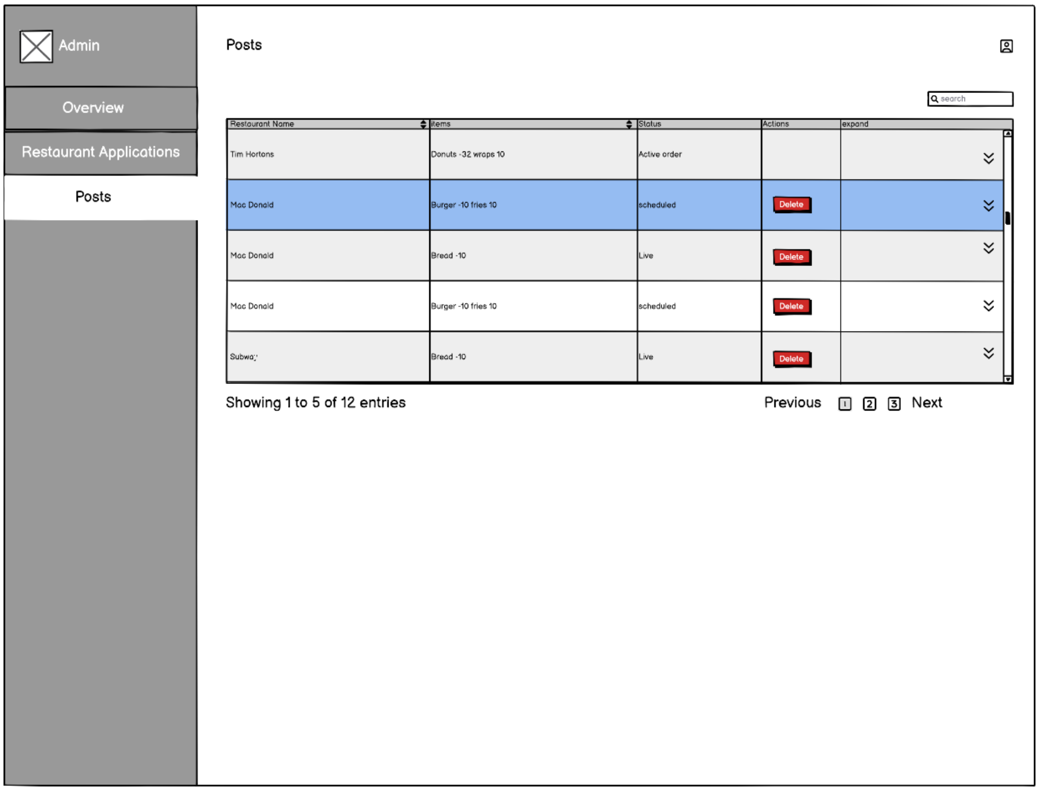
Error

Posts with active orders cannot be deleted

Ok

Showing 1 to 5 of 12 entries

Figure 18: A wireframe of the admin trying to delete an active post; the system displays an error message and restricts the admin to delete it, Created using Balsamiq.cloud [4].



Restaurant Name	items	Status	Actions	expand
Tim Hortons	Donuts -32 wraps 10	Active order		⌄
Mac Donald	Burger -10 fries 10	scheduled	Delete	⌄
Mac Donald	Bread -10	Live	Delete	⌄
Mac Donald	Burger -10 fries 10	scheduled	Delete	⌄
Subwa;	Bread -10	Live	Delete	⌄

Showing 1 to 5 of 12 entries

Figure 19: A wireframe of the admin viewing the posts on LastServe.ca, Created using Balsamiq.cloud [4].

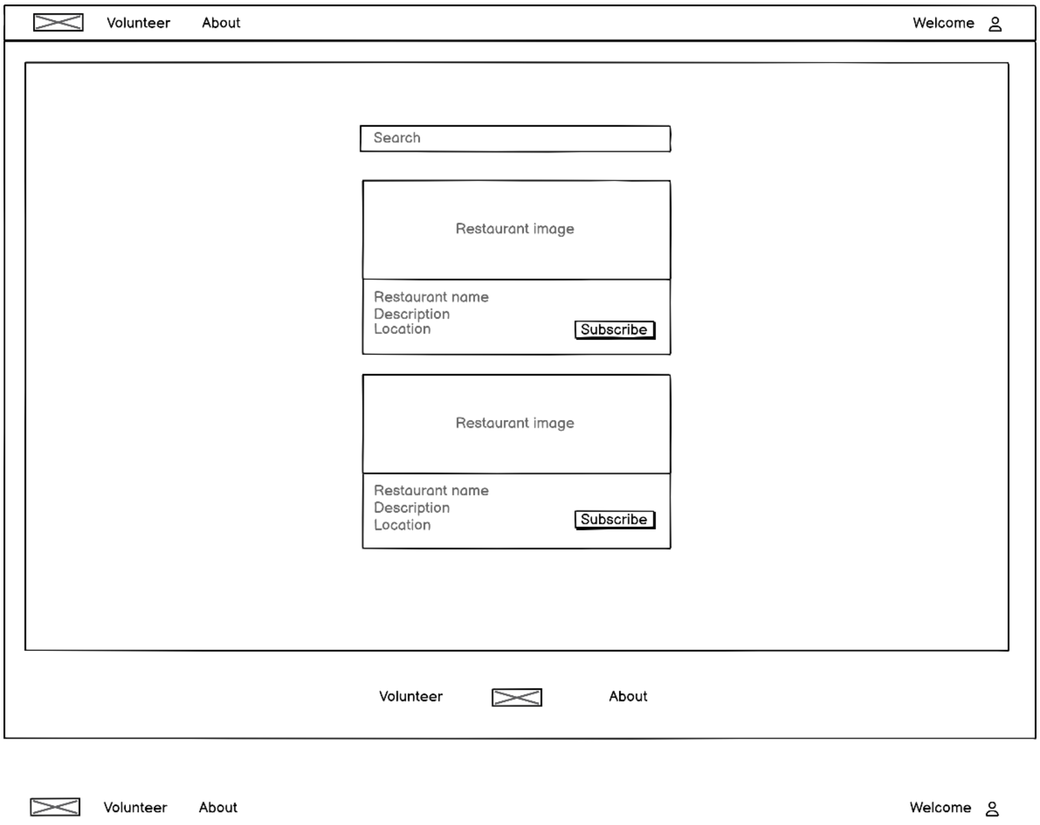


Figure 20: A wireframe for subscription page, Created using Balsamiq.cloud [4].

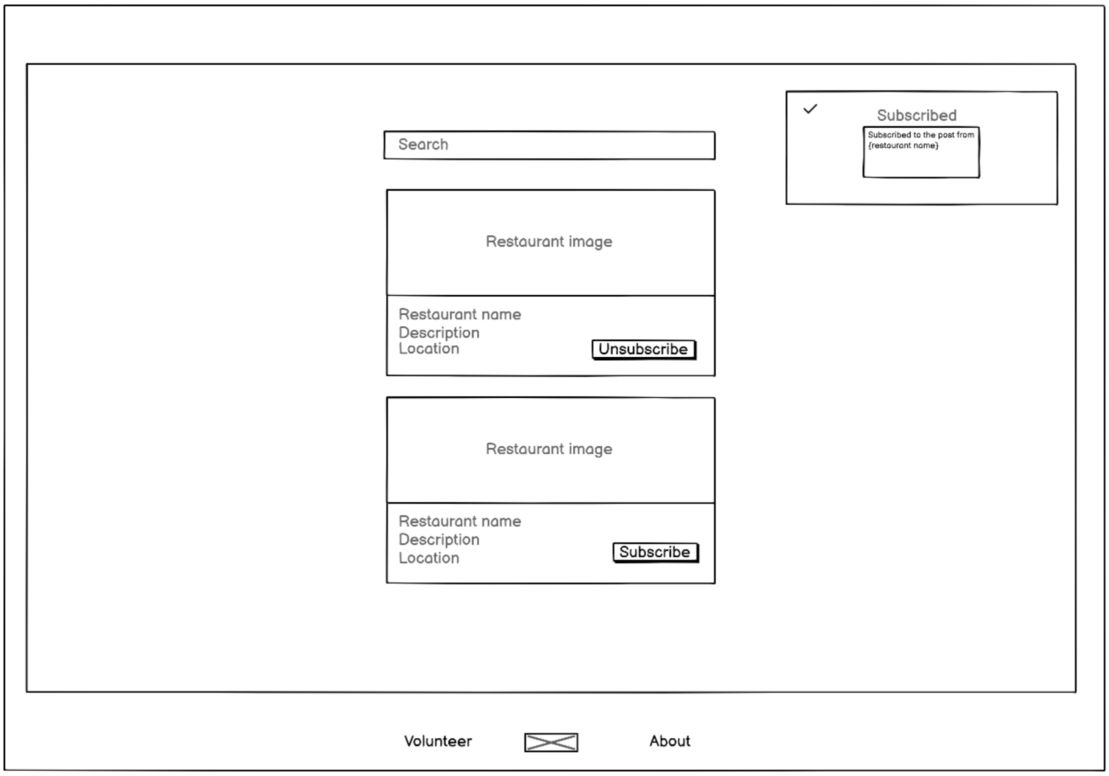


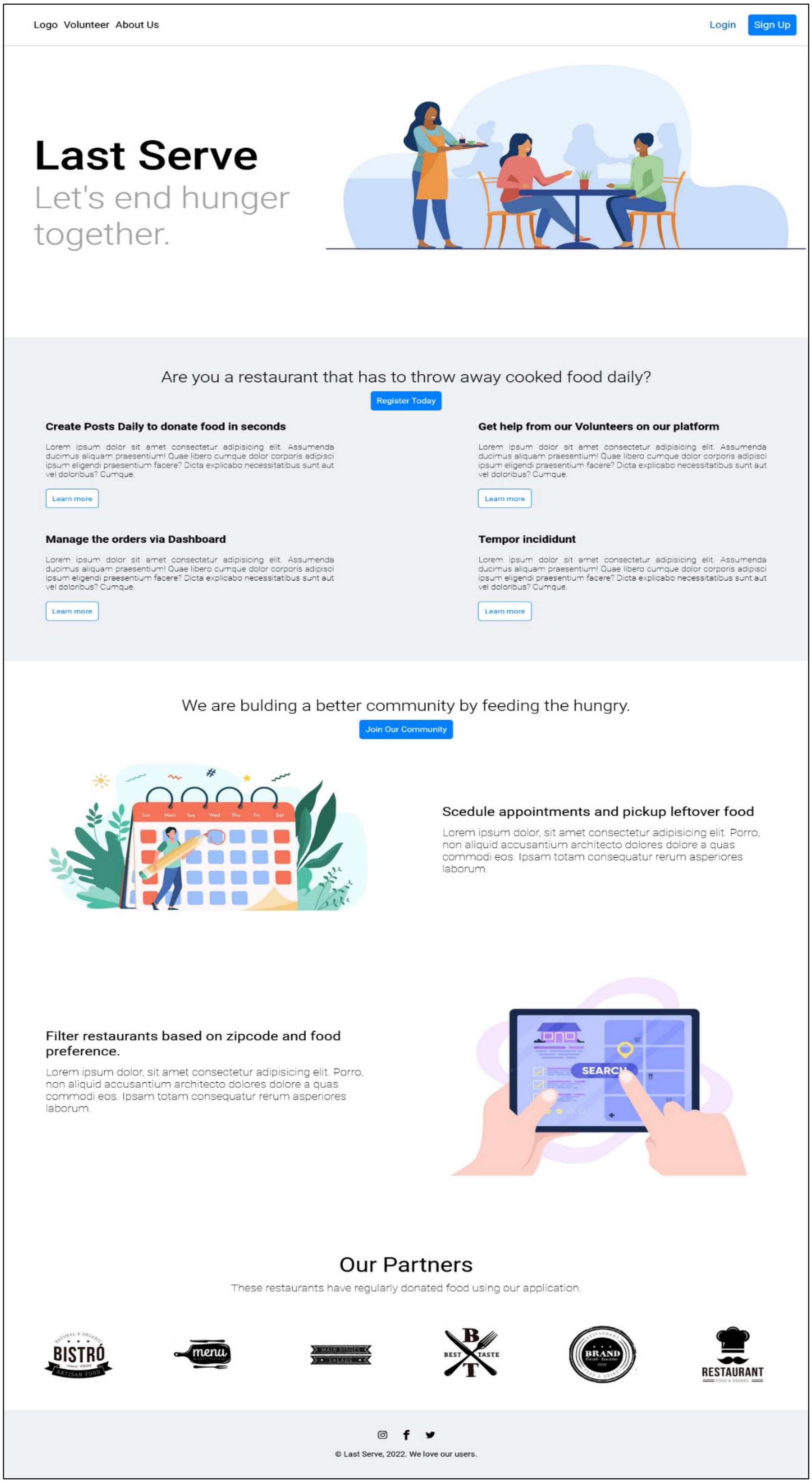
Figure 21: A wireframe for subscribed pages; a user can unsubscribe when required, Created using Balsamiq [4].

2.5.3 Website Design

The design we created was centered on enhancing user experience and making navigation simple. The layout is adaptable and takes a simple approach with a soft colour scheme. All form-filling procedures are clear and straightforward.

The design choices for several pages are listed below:

Home page: The fundamental idea of our website was intended to be easily communicated on the home page. The message is presented using large fonts, and the accompanying graphics help the reader understand our goal. It contains all the requisite links to point users in the direction of all important pages. A new user can establish trust by learning names of our partner restaurants. Our homepage is completely transparent to show what we do (see **Figure 22**).



Last Serve

Let's end hunger together.

Are you a restaurant that has to throw away cooked food daily?

Create Posts Daily to donate food in seconds

Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda ducimus aliquam praesentium! Quae libero cumque dolor corporis adipisci ipsum eligendi praesentium facere? Dicta explicabo necessitatibus sunt aut vel doloribus? Cumque.

Get help from our Volunteers on our platform

Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda ducimus aliquam praesentium! Quae libero cumque dolor corporis adipisci ipsum eligendi praesentium facere? Dicta explicabo necessitatibus sunt aut vel doloribus? Cumque.

Manage the orders via Dashboard

Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda ducimus aliquam praesentium! Quae libero cumque dolor corporis adipisci ipsum eligendi praesentium facere? Dicta explicabo necessitatibus sunt aut vel doloribus? Cumque.

Tempor incididunt

Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda ducimus aliquam praesentium! Quae libero cumque dolor corporis adipisci ipsum eligendi praesentium facere? Dicta explicabo necessitatibus sunt aut vel doloribus? Cumque.

We are bulding a better community by feeding the hungry.

Scedule appointments and pickup leftover food

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Porro, non aliquid accusantium architecto dolores dolore a quas commodi eos. Ipsam totam consequatur rerum asperiores laborum.

Filter restaurants based on zipcode and food preference.

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Porro, non aliquid accusantium architecto dolores dolore a quas commodi eos. Ipsam totam consequatur rerum asperiores laborum.

Our Partners

These restaurants have regularly donated food using our application.

© Last Serve, 2022. We love our users.

Figure 22: Screenshot for LastServe Home Page

User Registration: The user registration form is simple and simply requires the bare minimum of data. To prevent any problems in the database, it includes all the validations. A user can generate a unique password once and use it to log in each time (see **Figure 23**).

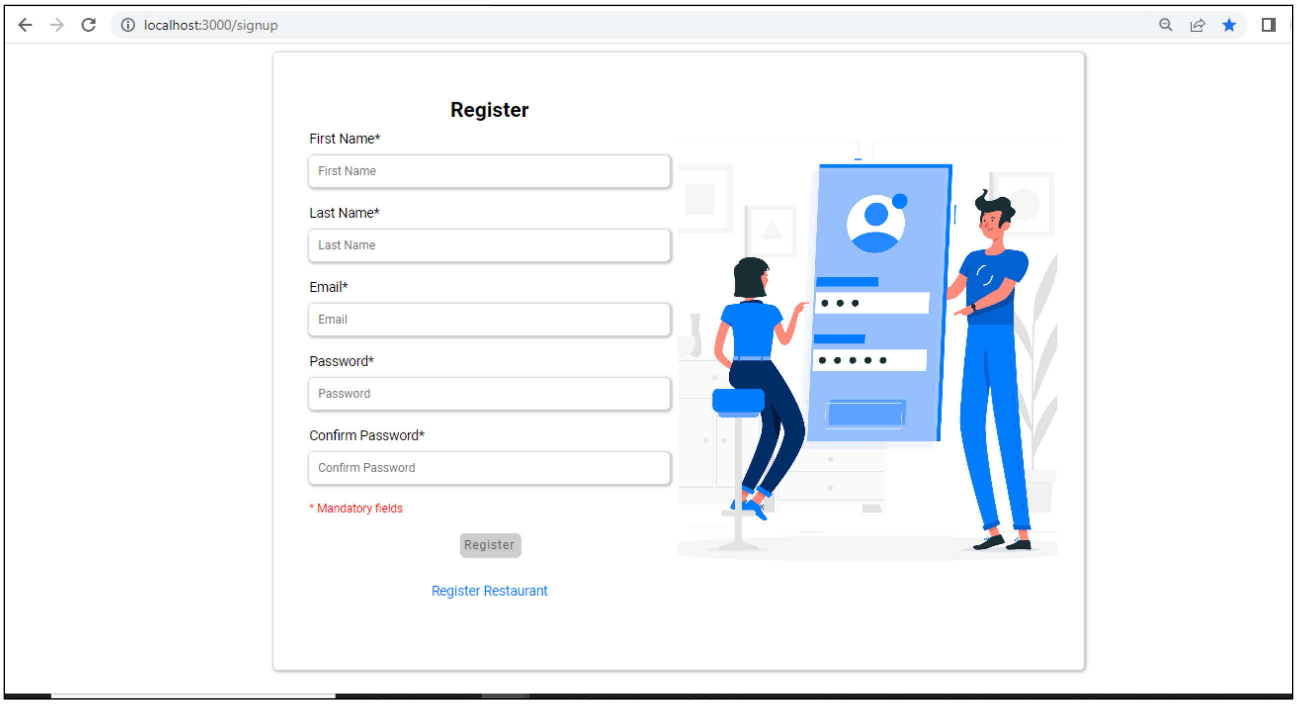


Figure 23: Screenshot for user registration

Restaurant Registration Page: (see **Figure 24**)

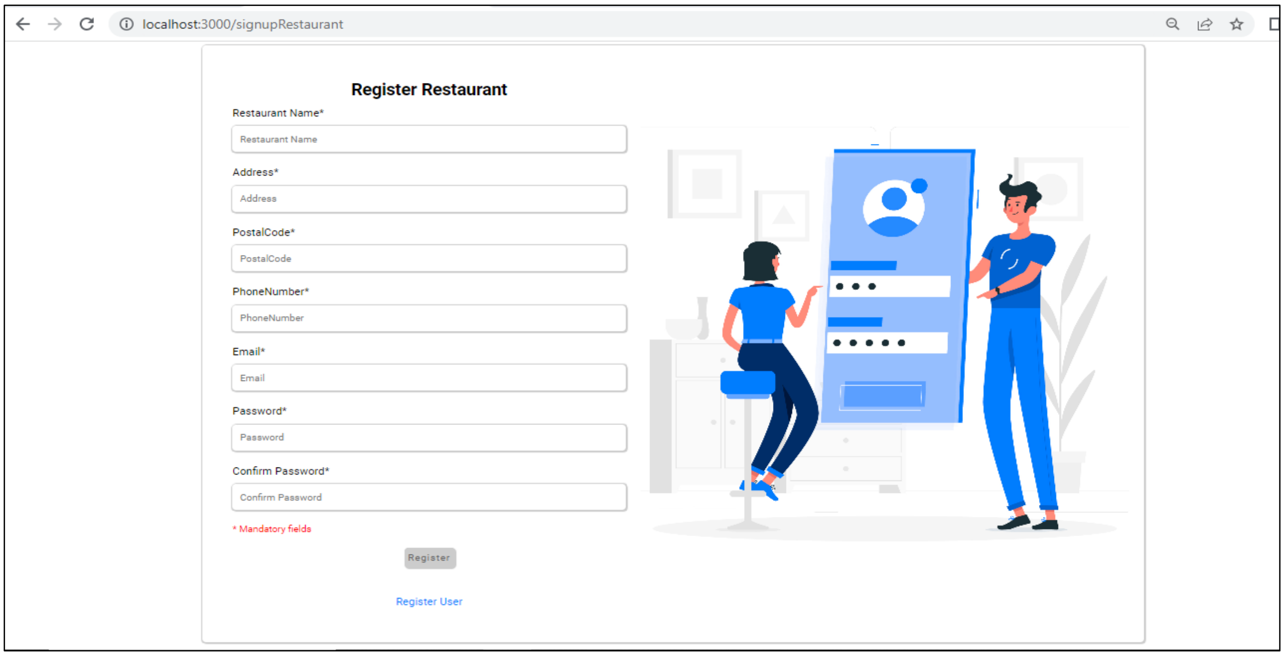
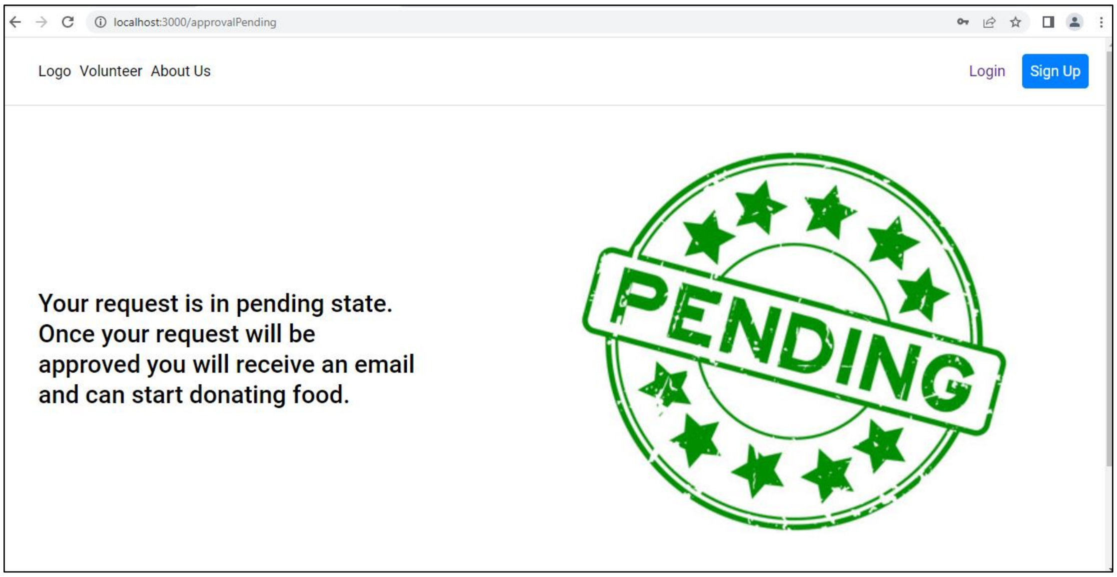


Figure 24: Screenshot for Restaurant Registration Page

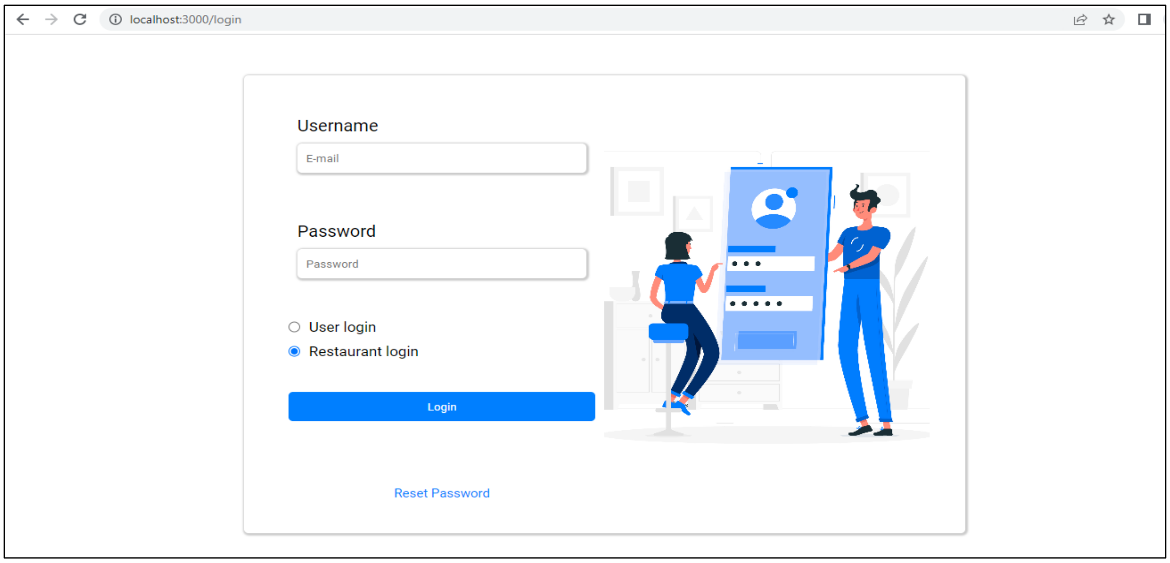
Restaurant Pending Approval Page: The restaurant will be sent to this page after successfully registering on our website. The user will see a notification about pending requests. The business will receive an email once their request has been granted, at which point they can start donating food through our website (see **Figure 25**).



Your request is in pending state. Once your request will be approved you will receive an email and can start donating food.

Figure 25: Screenshot for pending approval page

Restaurant Login: The restaurant manager will be able to log in to our website once their restaurant has been approved and carry out specific duties dependent on permission. He will see the above form after clicking the login link. After choosing the "Restaurant Login" radio button and entering his login information, he will be taken to the restaurant dashboard page once his login has been successful (see **Figure 26**).



Username

E-mail

Password

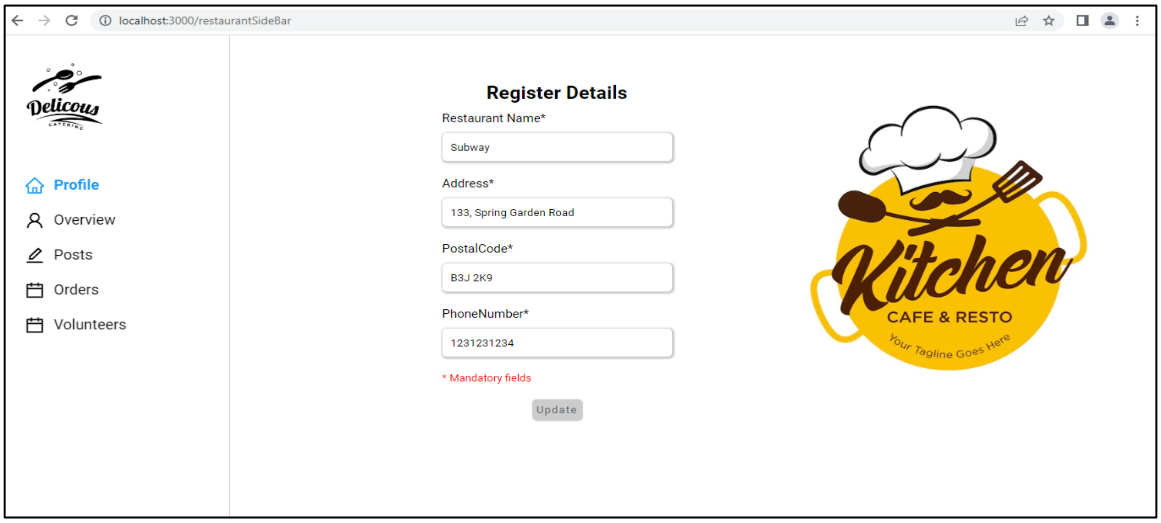
Password

○ User login

● Restaurant login

Figure 26: Screenshot for Restaurant Login

Restaurant View/Edit Profile: The manager can log in to the system and view and modify the restaurant's details once it has been approved. After logging in, the restaurant manager will be taken to the restaurant dashboard, where he can access the "Profile" option on the left. When he clicks on it, all of the most recent information about the restaurant, including the logo, is presented. He can adjust this information to suit his needs, and the system will update to reflect the changes successful (see **Figure 27**).



Register Details

Restaurant Name*: Subway

Address*: 133, Spring Garden Road

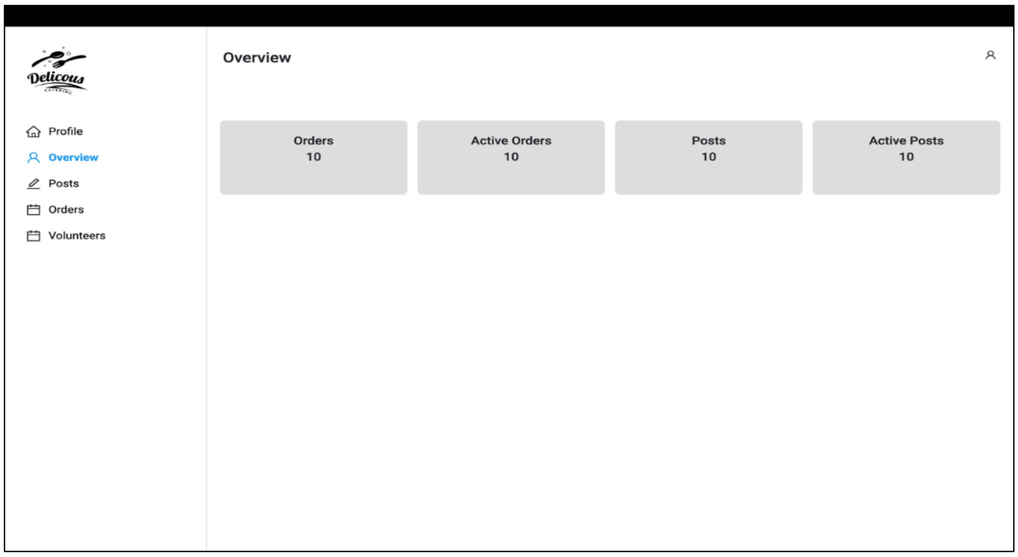
PostalCode*: B3J 2K9

PhoneNumber*: 1231231234

* Mandatory fields

Figure 27: Screenshot for View/Edit Restaurant Profile

Restaurant Dashboard overview: As soon as the restaurant manager logs in, they are taken to the Overview page, where they can see a quick summary of all the restaurant's statistics, including the total number of orders, the number of active orders, the number of posts the restaurant has made, and the number of active posts. To make the dashboard's interface easy to use and understand, the color scheme has been kept minimalistic successful (see **Figure 28**).

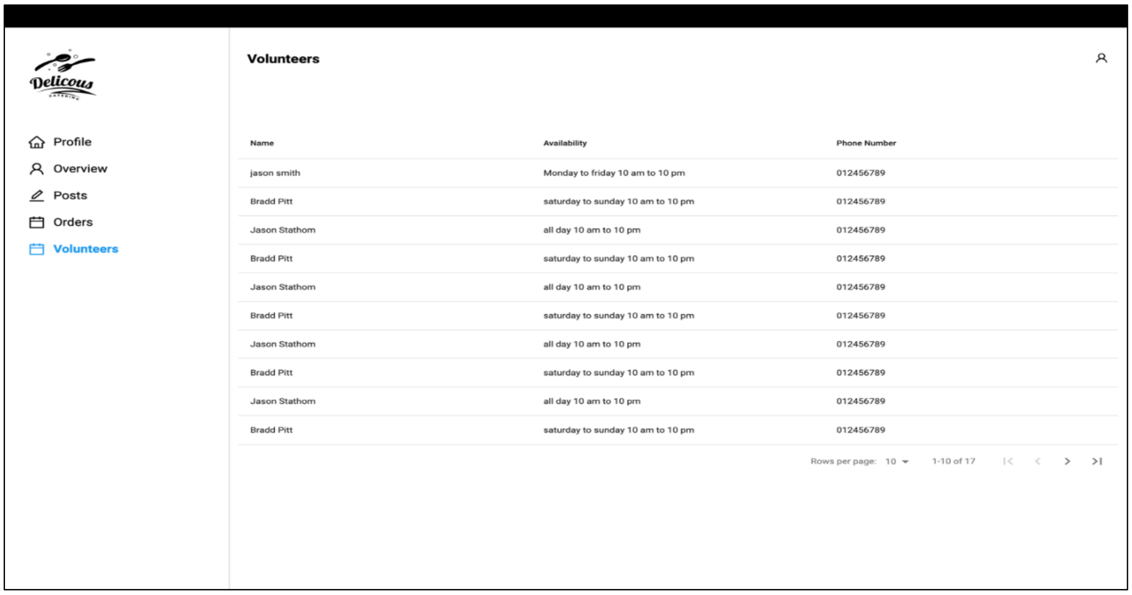


Overview

Orders	Active Orders	Posts	Active Posts
10	10	10	10

Figure 28: Screenshot for Restaurant Dashboard Overview Page

Volunteer Display: When the restaurant management needs assistance from a volunteer, they can go to the volunteers page to get a list of everyone in the system who is willing to help out and their availability. To help the restaurant identify a volunteer more quickly, the table can be arranged according to availability. The volunteer's phone number is also provided for the management to call. The tables' pagination will assist in breaking up the material into manageable portions successful (see **Figure 29**).

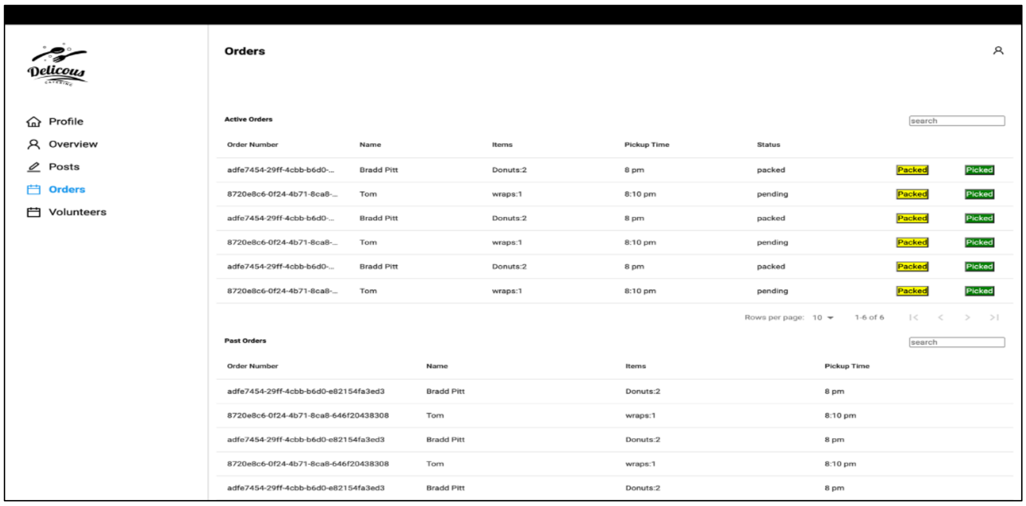


Volunteers

Name	Availability	Phone Number
jason smith	Monday to friday 10 am to 10 pm	012456789
Bradd Pitt	saturday to sunday 10 am to 10 pm	012456789
Jason Stathom	all day 10 am to 10 pm	012456789
Bradd Pitt	saturday to sunday 10 am to 10 pm	012456789
Jason Stathom	all day 10 am to 10 pm	012456789
Bradd Pitt	saturday to sunday 10 am to 10 pm	012456789
Jason Stathom	all day 10 am to 10 pm	012456789
Bradd Pitt	saturday to sunday 10 am to 10 pm	012456789
Jason Stathom	all day 10 am to 10 pm	012456789
Bradd Pitt	saturday to sunday 10 am to 10 pm	012456789

Figure 29: Screenshot for Volunteer viewing Page on Admin dashboard

Restaurant orders page: When the restaurant manager wishes to manage the orders on the system they can navigate to the orders screen. The page is divided into 2 table, active orders, and past orders, to organize the data into logical groups and make information retrieval easier for the restaurant manager. Search bars are provided on top of the table to easily search for an order. For active order buttons are provided to change the status of the order to packed and picked to manage the orders successful (see **Figure 30**).



Orders

Active Orders

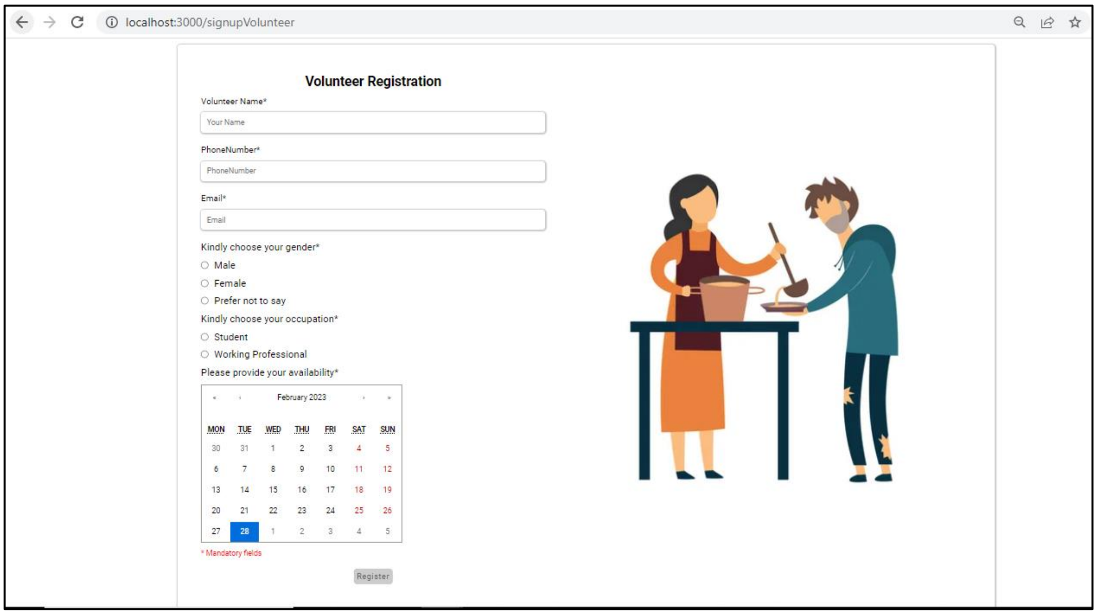
Order Number	Name	Items	Pickup Time	Status		
adfe7454-29ff-4cbb-b6d0-...	Bradd Pitt	Donuts:2	8 pm	packed	Packed	Picked
8720e8c6-0f24-4b71-8ca8-...	Tom	wraps:1	8:10 pm	pending	Packed	Picked
adfe7454-29ff-4cbb-b6d0-...	Bradd Pitt	Donuts:2	8 pm	packed	Packed	Picked
8720e8c6-0f24-4b71-8ca8-...	Tom	wraps:1	8:10 pm	pending	Packed	Picked
adfe7454-29ff-4cbb-b6d0-...	Bradd Pitt	Donuts:2	8 pm	packed	Packed	Picked
8720e8c6-0f24-4b71-8ca8-...	Tom	wraps:1	8:10 pm	pending	Packed	Picked

Past Orders

Order Number	Name	Items	Pickup Time
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps:1	8:10 pm
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm
8720e8c6-0f24-4b71-8ca8-646f20438308	Tom	wraps:1	8:10 pm
adfe7454-29ff-4cbb-b6d0-e82154fa3ed3	Bradd Pitt	Donuts:2	8 pm

Figure 30: Screenshot for Volunteer display Page

Volunteer Registration: All of the fields needed for a volunteer to register are visible on the volunteer page. They must complete all the necessary fields so that a restaurant can find the best match. The form is kept straightforward and doesn't ask for a lot of personal data successful (see **Figure 31**).



localhost:3000/signupVolunteer

Volunteer Registration

Volunteer Name*

Your Name

PhoneNumber*

PhoneNumber

Email*

Email

Kindly choose your gender*

○ Male
○ Female
○ Prefer not to say

Kindly choose your occupation*

○ Student
○ Working Professional

Please provide your availability*

« ‹ February 2023 › »

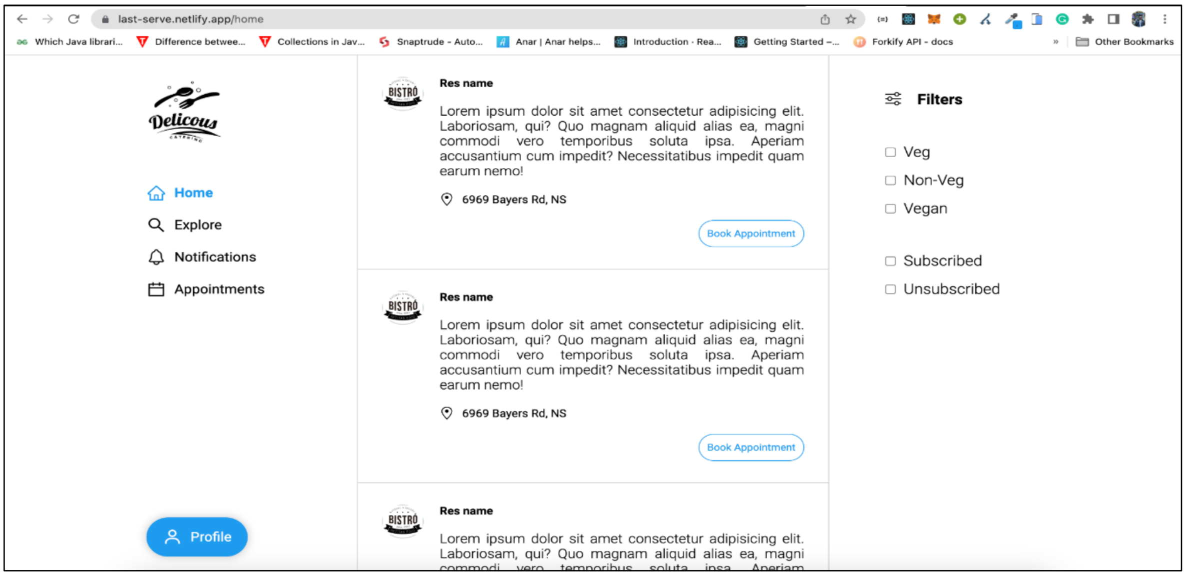
MON	TUE	WED	THU	FRI	SAT	SUN
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	1	2	3	4	5

* Mandatory fields

Register

Figure 31: Screenshot for Volunteer registration

Appointment booking: Users will see a list of posts from restaurants that are now offering free food on the home page after logging into their accounts. Depending on the food category and whether they have a subscription to a certain restaurant, users can filter the establishments successful (see **Figure 32**).



last-serve.netlify.app/home

Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni commodi vero temporibus soluta ipsa. Aperiam accusantium cum impedit? Necessitatibus impedit quam earum nemo!

6969 Bayers Rd, NS

Book Appointment

Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni commodi vero temporibus soluta ipsa. Aperiam accusantium cum impedit? Necessitatibus impedit quam earum nemo!

6969 Bayers Rd, NS

Book Appointment

Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni

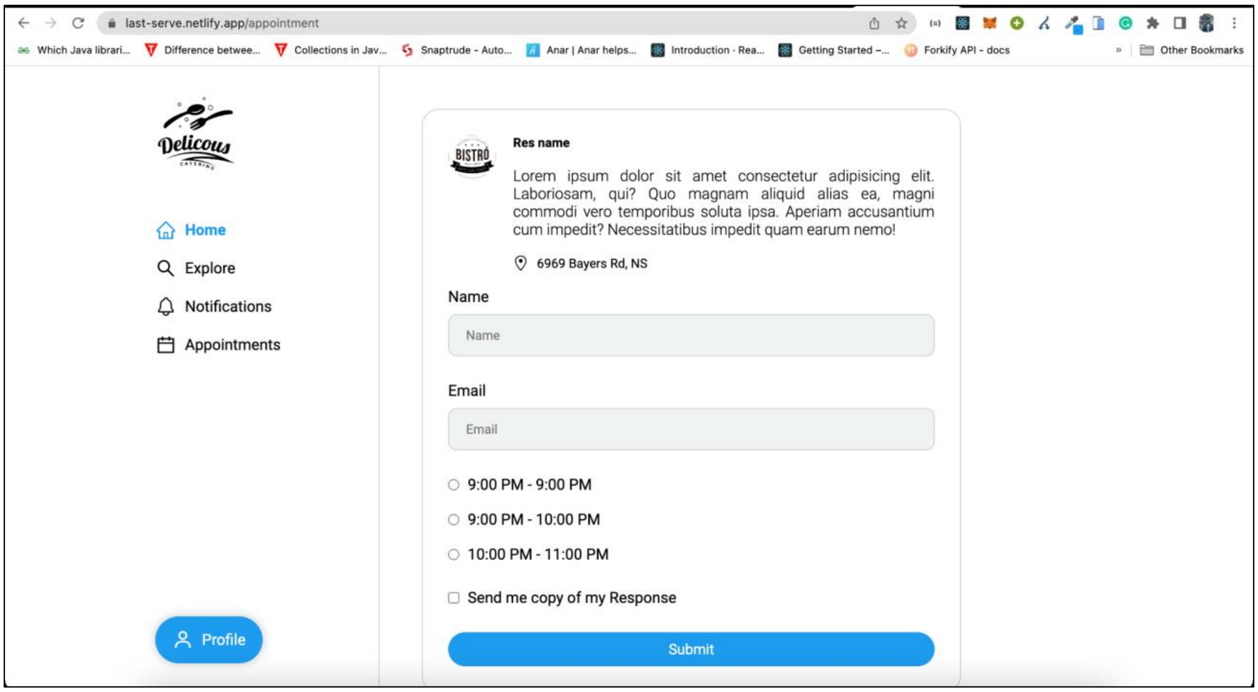
Filters

□ Veg
□ Non-Veg
□ Vegan

□ Subscribed
□ Unsubscribed

Figure 32: Screenshot for appointment booking Page

When users select the "Book Appointment" button on the main page, the above page will be displayed. On this page, a form asking for the user's name, email address, desired time slot, and whether they want a copy of their response via email will be present (see **Figure 33**).



Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni commodi vero temporibus soluta ipsa. Aperiam accusantium cum impedit? Necessitatibus impedit quam earum nemo!

6969 Bayers Rd, NS

Name

Name

Email

Email

○ 9:00 PM - 9:00 PM

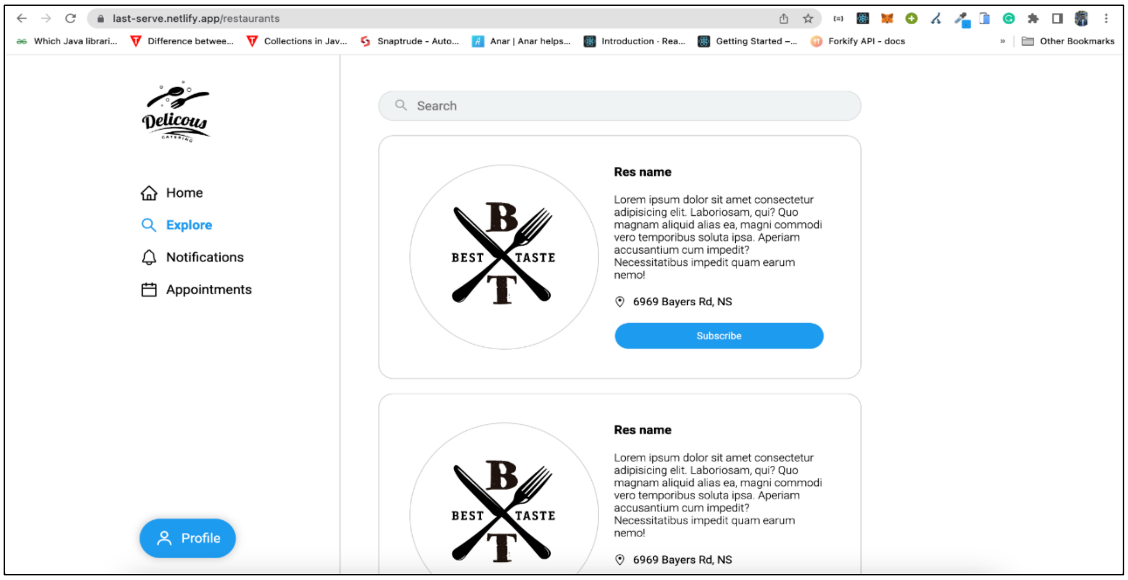
○ 9:00 PM - 10:00 PM

○ 10:00 PM - 11:00 PM

□ Send me copy of my Response

Figure 33: Screenshot for appointment booking screen

On the explore page users will be able to see a list of restaurants. Users can click on the subscribe button to get notifications whenever the restaurant posts something (see **Figure 34**).



Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni commodi vero temporibus soluta ipsa. Aperiam accusantium cum impedit? Necessitatibus impedit quam earum nemo!

6969 Bayers Rd, NS

Res name

Lorem ipsum dolor sit amet consectetur adipisicing elit. Laboriosam, qui? Quo magnam aliquid alias ea, magni commodi vero temporibus soluta ipsa. Aperiam accusantium cum impedit? Necessitatibus impedit quam earum nemo!

6969 Bayers Rd, NS

Figure 34: Screenshot for Subscription page

2.6 User Experience

User Scenario- User Registration: At midnight, a student is starving but too worn out to prepare supper. Due to a limited budget, the student does not want to order food at a restaurant. The student wishes to register with a website that lists local restaurants that provide leftovers from the day at no charge. To finish the procedure, he will do the following.

1. User opens the browser and types www.lastserve.ca

2. System displays the home page.
3. User clicks on the 'Register as a user' button on the home page.
4. System displays the registration page, requesting first name, last name, email address and password.
5. User enters the details.
6. User clicks the 'register' button.
  6.1 System displays 'First Name cannot be blank' error message.
    6.1.1 User enters the first name.
    6.1.2 User clicks the 'register' button.
  6.2 System displays 'Email already exists' error message.
    6.2.1 User enters a new email address.
    6.2.2 User clicks the 'register' button.
  6.3 System displays 'Password must be minimum 8 characters long' message.
    6.3.1 User enters a password with 8 or more characters.
    6.3.2 User clicks the 'register' button.
  6.4 System clears password and confirms password field and displays 'Password and confirm password fields do not match error message.
    6.4.1 User enters matching passwords.
    6.4.2 User clicks the 'register' button.
7. System registers the user.
8. System redirects the user to the login page.

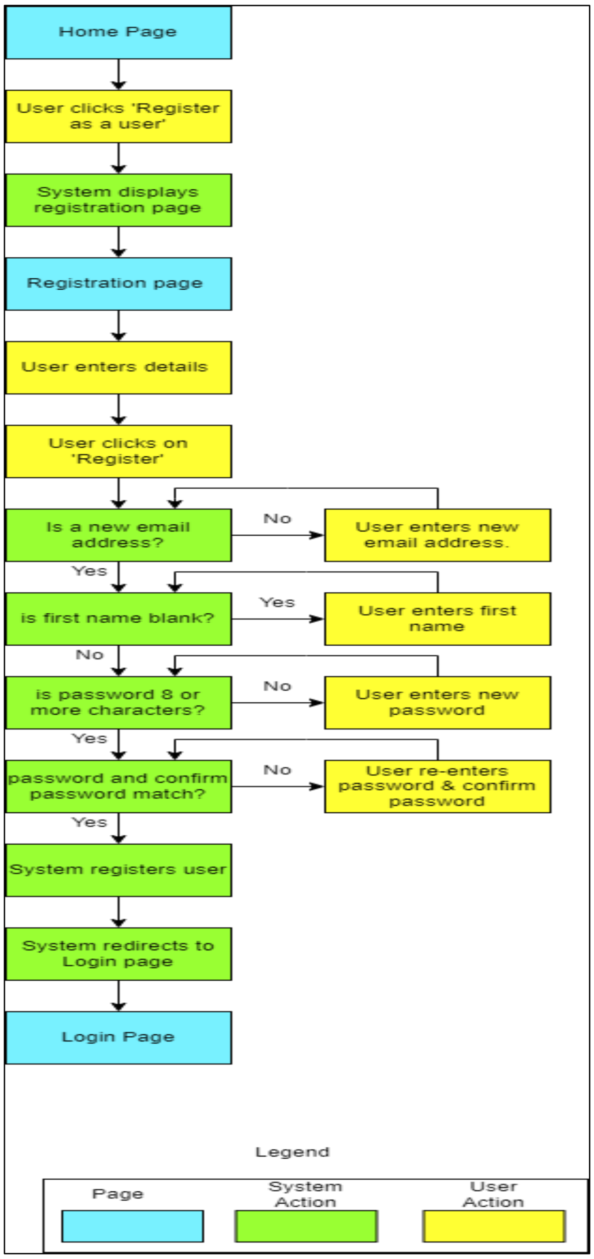

Home Page
User clicks 'Register as a user'
System displays registration page
Registration page
User enters details
User clicks on 'Register'
Is a new email address?
No
User enters new email address.
Yes
is first name blank?
Yes
User enters first name
No
is password 8 or more characters?
No
User enters new password
Yes
password and confirm password match?
No
User re-enters password & confirm password
Yes
System registers user
System redirects to Login page
Login Page
Legend
Page
System Action
User Action

Figure 35: Task flow diagram for User Registration, Created using Draw.io [5].

User Scenario- Login into account: A student is hungry and wants to check restaurants in the vicinity offering free leftover food on the LastServe website.

1. User opens the browser and types www.lastserve.ca
2. System displays the home page.
3. User clicks the 'User Login' button.
4. System displays the login page, requesting email and password.
5. User enters the email address and password.
6. Users clicks the 'Login' button.
6.1. System displays an 'Invalid email and/or password' error message.
6.1.1. System requests email and password.
6.1.2. User enters details.
6.1.3. User clicks 'Login' button.
6.2. System displays 'Invalid email and/or password' error message.

6.3. System displays ‘Reset password’ message.
6.4. User clicks on ‘Reset password’ link.
6.5. System displays reset password page, requesting the user to enter the registered email address.
6.6. User enters email id.
    6.6.1. User clicks on ‘Submit’ button.
6.7. System sends an authentication code to the entered email address.
    6.7.1 System displays a field to enter the code.
    6.7.2 User enters the authentication code.
    6.7.3 User clicks on the ‘Submit’ button.
6.8 System displays the reset password page, requesting a new password and confirm password.
    6.8.1 User enters a new password and confirms password.
    6.8.2 User clicks on ‘Reset Password’ button.
6.9 System displays a “New password must be minimum 8 characters” message.
    6.9.1 User enters a new password.
    6.9.2 User clicks on the ‘Reset Password’ button.
6.10 System displays ‘New Password’ and confirm password does not match”.
    6.10.1 User enters a new password confirms password.
    6.10.2 User clicks on the ‘Reset Password’ button.
    6.10.3 System resets the password and displays the ‘Password reset successful’ message.

9. System redirects the user to the login page.
10. System authenticates the user.
11. System redirects the user to the home page.

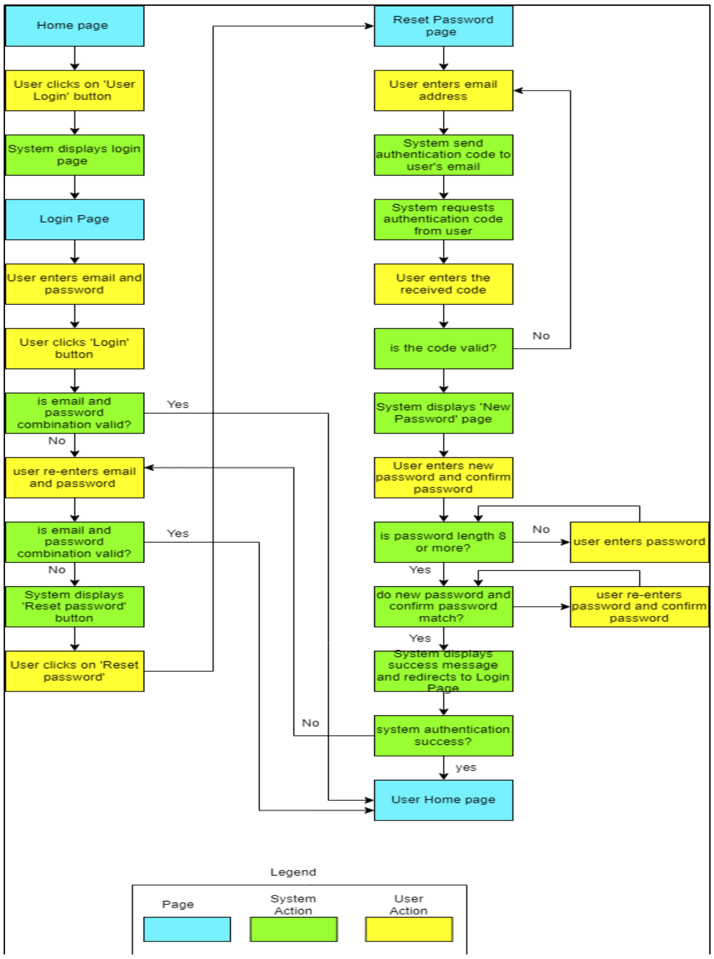

Home page
User clicks on 'User Login' button
System displays login page
Login Page
User enters email and password
User clicks 'Login' button
is email and password combination valid?
Yes
No
user re-enters email and password
is email and password combination valid?
Yes
No
System displays 'Reset password' button
User clicks on 'Reset password'
Reset Password page
User enters email address
System send authentication code to user's email
System requests authentication code from user
User enters the received code
is the code valid?
No
System displays 'New Password' page
User enters new password and confirm password
is password length 8 or more?
No
user enters password
Yes
do new password and confirm password match?
user re-enters password and confirm password
Yes
System displays success message and redirects to Login Page
system authentication success?
No
yes
User Home page
Legend
Page
System Action
User Action

Figure 36: Task flow diagram for User Login, Created using Draw.io [5].

User Scenario- View User Profile: The student has subscribed to get an email alert when a subscribed creates a post. The student wants to see a list of subscribed restaurants.

1. System displays the home page.
2. Student clicks on the ‘View Profile’ icon on the home page.
3. System displays the view profile details page.
4. User scrolls to the ‘Subscriptions’ section on the view details page.
5. System displays a list of subscribed restaurants.
6. User clicks on the ‘Back’ button
7. System redirects user to the home page.

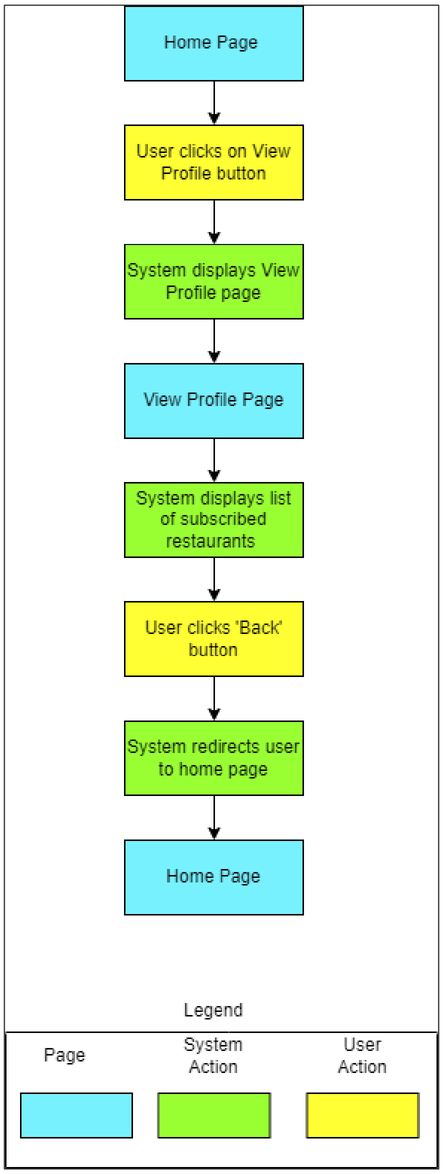

Home Page
User clicks on View Profile button
System displays View Profile page
View Profile Page
System displays list of subscribed restaurants
User clicks 'Back' button
System redirects user to home page
Home Page
Legend
Page
System Action
User Action

Figure 37: Task flow diagram for User View Profile, Created using draw.io [5].

User Scenario- Edit User Profile: The student typed his name incorrectly when setting up an account. Also, even though it was an optional field, the student omitted the last name. The last time he visited a restaurant, he was denied access to pick up food since the name on the reservation was different from the name on his ID. In order to prevent future misunderstandings, the student wishes to modify his first name and add a last name.

1. System displays the home page.
2. User clicks on the 'Edit Profile' button on the home page.
3. System displays the edit profile page with first name and last name.
  3.1 User clicks on the 'Back' button.
  3.2 System redirects user to the home page.
4. User enters first name and last name.
    4.1 User clicks 'back' button.
    4.2 System displays 'No data updated' message.
5 User clicks on 'Update' button.
    5.1 System displays 'First Name cannot be blank' error message.
    5.2 User enters first name.
    5.3 System displays 'First Name updated' message.
6 System updates details.
7 System redirects user to the home page.

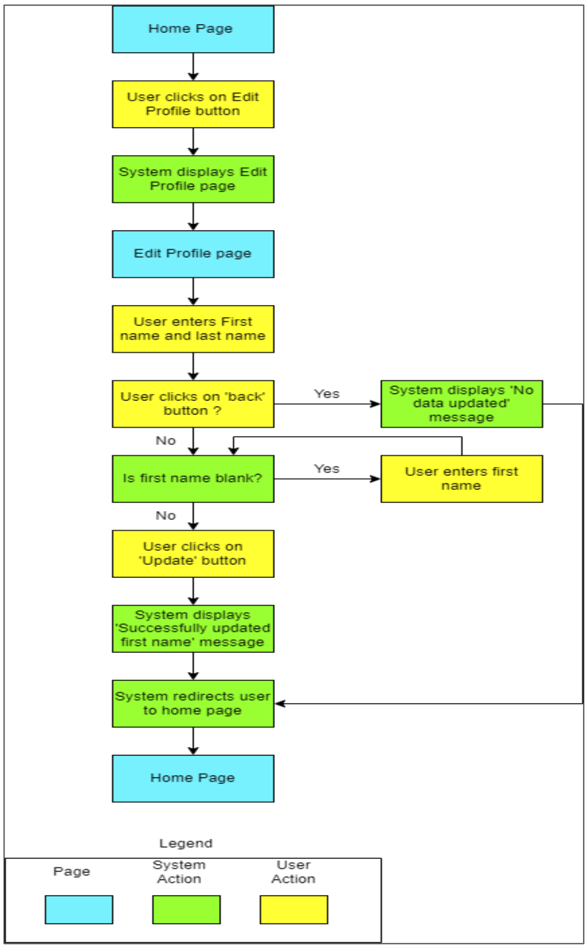

Home Page
User clicks on Edit Profile button
System displays Edit Profile page
Edit Profile page
User enters First name and last name
User clicks on 'back' button ?
Yes
System displays 'No data updated' message
No
Is first name blank?
Yes
User enters first name
No
User clicks on 'Update' button
System displays 'Successfully updated first name' message
System redirects user to home page
Home Page
Legend
Page
System Action
User Action

Figure 38: Task flow diagram to edit User Profile, Created using Draw.io [5].

User Scenario- Restaurant Registration: The restaurant manager is always unsure of what to do with the excess food at the end of the day. He is searching for a website where he can sign up to donate this leftover food.

1. User go to home page of website and click on Register button. So, he will get two options and one of them will be register as Donor. So, when user clicks on it, it willopen Register Restaurant(donor) page.
2. User will enter all required details in the form such as restaurant name, address, phone number, email Id, password etc. and clicks on Register button.
3. System will authenticate restaurants registration details against all validation added inthe system.
4. On successful registration user will be redirected to a page where one message will be displayed that "your donor request is in pending state. Once your request will be approved, you will receive an email and can start donating food".

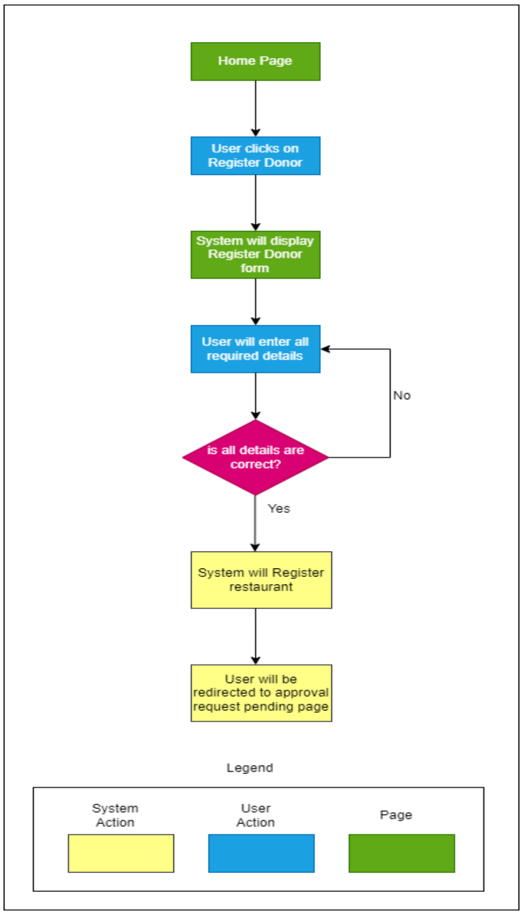

Home Page
User clicks on Register Donor
System will display Register Donor form
User will enter all required details
No
is all details are correct?
Yes
System will Register restaurant
User will be redirected to approval request pending page
Legend
System Action
User Action
Page

Figure 39: Task flow diagram for Restaurant Registration, Created using Draw.io [5].

User Scenario- Restaurant Login: The restaurant manager needs to log into the system to do different actions depending on the access he has in the system.

1. Restaurant manager will go to the restaurant login page.
2. Restaurant manager will enter username and password and hits enter key.
3. System will authenticate restaurants username and password.
4. On successfully login restaurant home page will be displayed to restaurant manager.

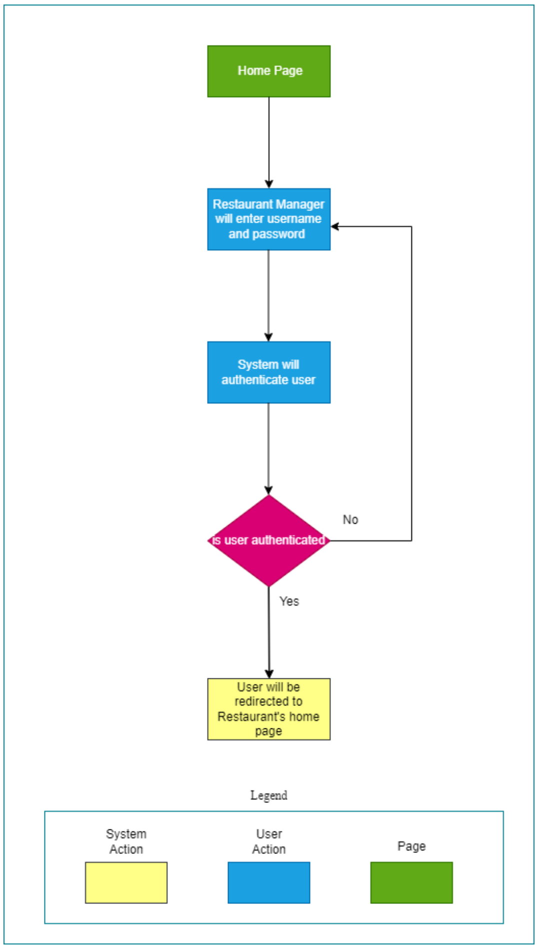

Home Page
Restaurant Manager will enter username and password
System will authenticate user
Is user authenticated
No
Yes
User will be redirected to Restaurant's home page
Legend
System Action
User Action
Page

Figure 40: Task flow diagram for Restaurant Login, Created using Draw.io [5].

User Scenario- Update Restaurant: A restaurant manager feels the need to update his restaurant's information because the phone number or address have changed in some way.

1. Restaurant manager will go to the restaurant profile page.
2. Restaurant manager will edit fields whatever he wants to update for example phone number, address, pin code etc. and click on update.
3. System will authenticate restaurant details against all validation added in the system.
4. On successfully updating restaurant details the restaurant manager will be redirected to restaurant dashboards.

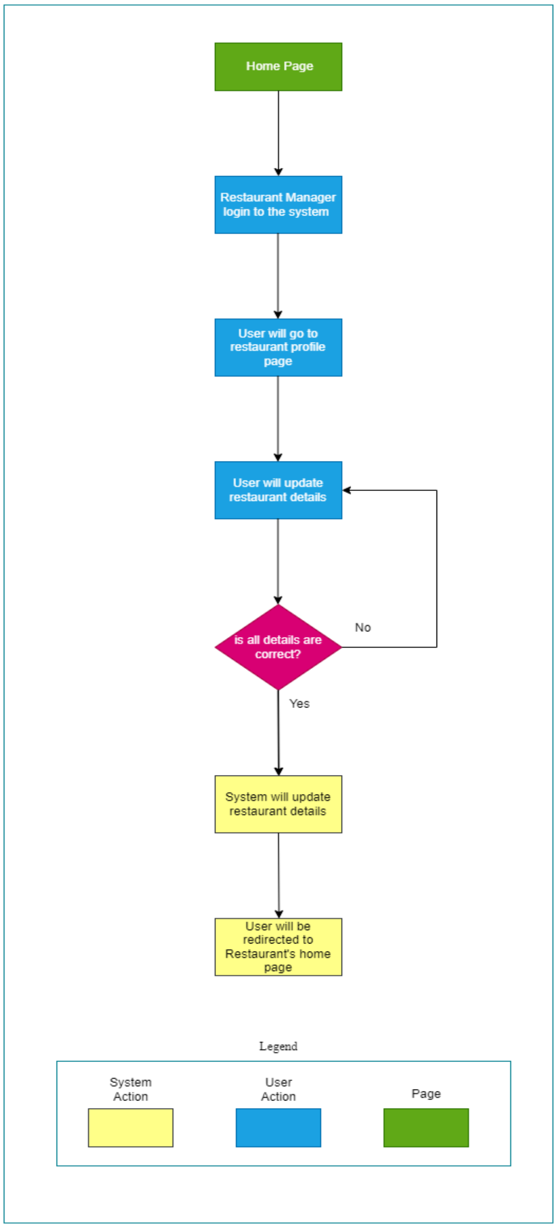

Home Page
Restaurant Manager login to the system
User will go to restaurant profile page
User will update restaurant details
is all details are correct?
No
Yes
System will update restaurant details
User will be redirected to Restaurant's home page
Legend
System Action
User Action
Page

Figure 41:Task flow diagram to update Restaurant Profile, Created using Draw.io [5].

User Scenario- View Restaurant: A restaurant manager wants to view the details of his restaurant to make sure everything is perfect and to determine if any of the details require change.

1. Restaurant manager will go to the restaurant profile page.
2. Restaurant manager will see all the details of his restaurants.

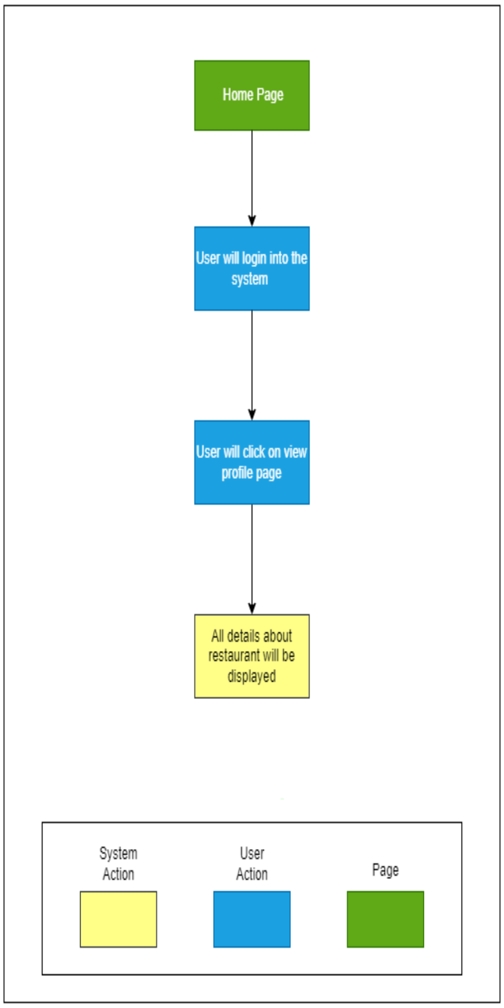

Home Page
User will login into the system
User will click on view profile page
All details about restaurant will be displayed
System Action
User Action
Page

Figure 42: Task flow diagram to View Restaurant Profile, Created using Draw.io [5].

User Scenario- Create a new post: The restaurant manager wants to update the information about the remaining food each day so that visitors can make an appointment.

1. User will go to LastServe website and login as Restaurant Owner.

2. User will click on the Post tab from all available tabs in Hamburger menu.

3. Users will see all cards (1. Number of past posts, 2. Number of active posts, 3. Number of all orders, 4. Number of active orders).

4. User will click on the “Number of active posts” card.

5. User will be able to see all active posts of that restaurant along with the “Create Post” button on the top right side of the page.

6. User will click on this “Create Post” button.

7. New Form for creating posts will be displayed to user.

8. User will enter all required details for creating a post such as item name, item quantity, availability time for item to be picked up from restaurant etc.

9. After filling all required details, the user will click on the “Create” button so it will create this post.

  9.1. Users forget to enter some required fields while creating posts and clicks on the “Create” button.

9.2. User will get an error message “Please, enter a value for this field”.

9.3. User enters an invalid date for the “availability for item to be picked up” field and clicks on the “Create” button.

9.4. User will get an error message “please, enter valid date”.

10. User will be redirected to a page showing all active posts.

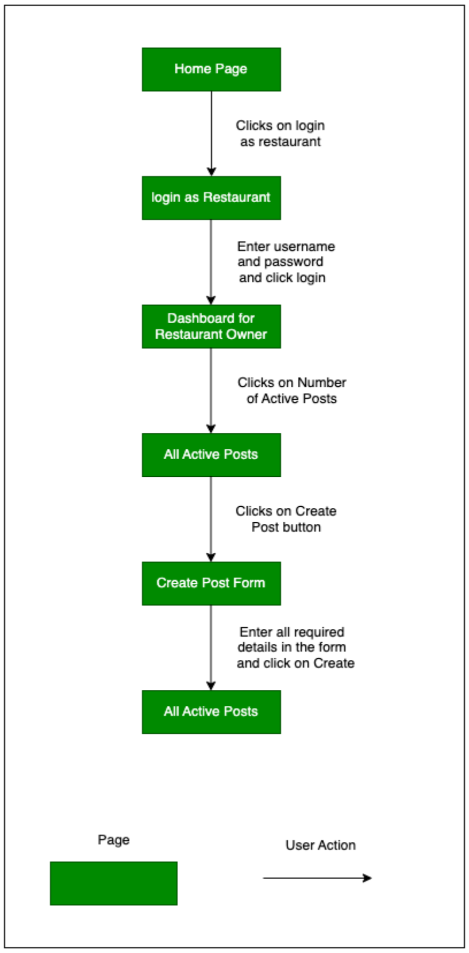

Home Page
Clicks on login as restaurant
login as Restaurant
Enter username and password and click login
Dashboard for Restaurant Owner
Clicks on Number of Active Posts
All Active Posts
Clicks on Create Post button
Create Post Form
Enter all required details in the form and click on Create
All Active Posts
Page
User Action

Figure 43: Task flow diagram to Create a new post, Created using Draw.io [5].

User Scenario- View Past Post: The restaurant manager needs access to both the active and archived posts so that they can edit or delete them as needed.

1. User will go to LastServe website and login as Restaurant Owner.
2. User will click on the Post tab from all available tabs in Hamburger menu.
3. Users will see all cards (1. Number of past posts, 2. Number of active posts, 3. Number of all orders, 4. Number of active orders).
4. User will click on the “Number of past posts” card.
5. User will be able to see all past posts of that particular restaurant along with the “Back” button on the page.

6. User will click on any particular past post and it will open all details of that particular post.

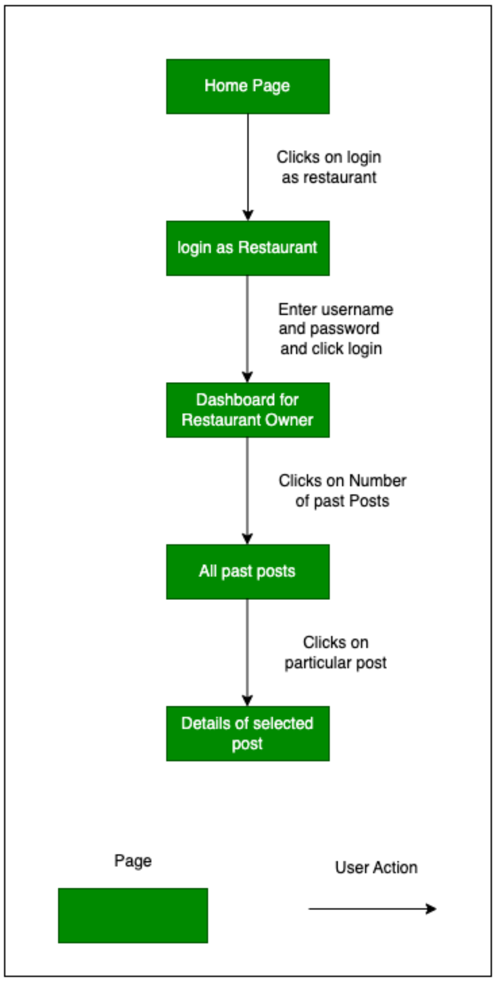

Home Page
Clicks on login as restaurant
login as Restaurant
Enter username and password and click login
Dashboard for Restaurant Owner
Clicks on Number of past Posts
All past posts
Clicks on particular post
Details of selected post
Page
User Action

Figure 44: Task flow diagram to view Past Posts, Created using Draw.io [5].

User Scenario- Edit active posts: To alter meal quantities or slot timings, the restaurant manager wants to amend the current post.

1. User will go to LastServe website and login as Restaurant Owner.
2. User will click on the Post tab from all available tabs in Hamburger menu.
3. Users will see all cards (1. Number of past posts, 2. Number of active posts, 3. Number of all orders, 4. Number of active orders).
4. User will click on the “Number of active posts” card.
5. User will be able to see all active posts of that restaurant.
6. User will click on any particular active post, and it will open all details of that active post along with the “Update” button on the page.

7. User will update required details of the post such as item name, item quantity, availability time to be picked up etc.

7.1. User forget to enter some required fields while updating posts and clicks on the “Update” button.

7.2. User will get an error message “Please, enter a value for this field”.

7.3. User enters an invalid date for the field “availability for the item to be picked up” and clicks on the “Update” button.

7.4. User will get an error message “please, enter a valid date”.

8. User will click on the “Update” button.

9. User will be redirected to a page where all active posts were shown.

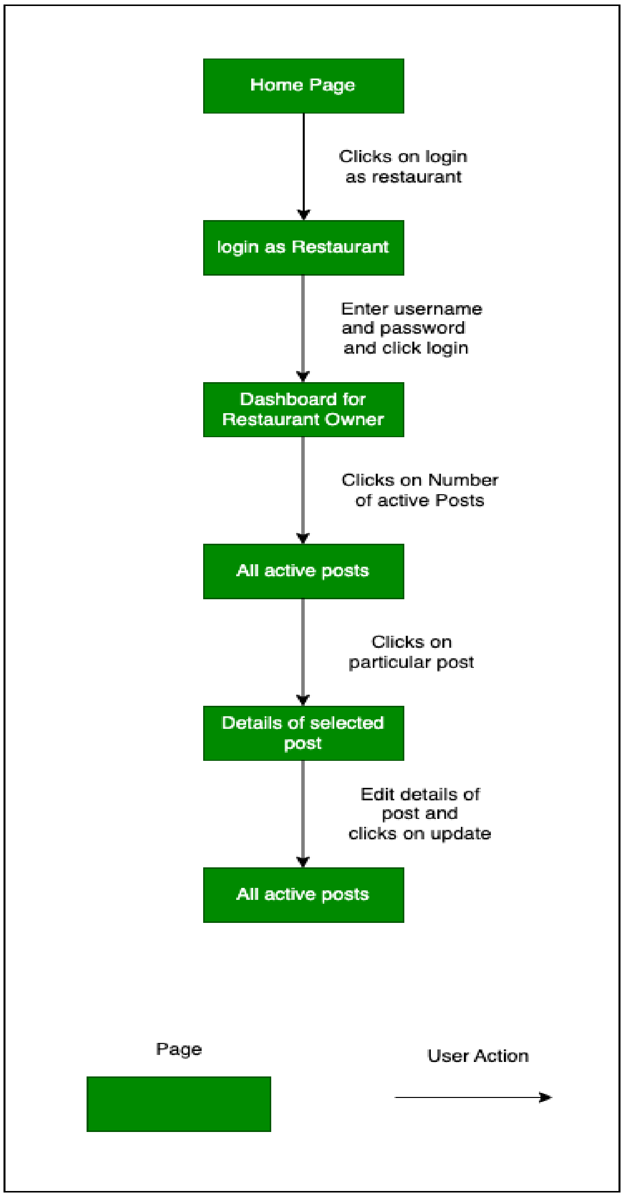

Home Page
Clicks on login as restaurant
login as Restaurant
Enter username and password and click login
Dashboard for Restaurant Owner
Clicks on Number of active Posts
All active posts
Clicks on particular post
Details of selected post
Edit details of post and clicks on update
All active posts
Page
User Action

Figure 45: Task flow diagram to edit Active Posts, Created using Draw.io [5].

User Scenario- Change order status from pending to packed and picked: The restaurant can see every order that customers have placed. After packing one order, he wants to modify the order status so that other workers working on it are aware of how to pack additional orders. Additionally, he wants to mark an order as picked up after the user picks it up so that the order can be managed more easily. When it's done, he clicks the buttons for packaged and picked-up.

1. The restaurant manager has logged in using his credentials.
2. The restaurant manager clicks on the orders.
3. The system displays all the active and past orders on tables.
  3.1 The system displays no active orders if there are no active orders for the restaurant.
4. The restaurant manager clicks on the packed button once an order is packed.
5. The system changes the state of the order to pack.
6. The system displays a message saying the order status is changed successfully.
  6.1 If there is an error while changing the state of the order "error something wentwrong, please try again is displayed to the user".
7. The restaurant manager clicks on picked up once the order is picked up.
8. The system changes the state of the order.
9. The system displays saying "Order status changed successfully", System removes the order from active orders and adds it to the past orders table.
10. If there is an error while changing the state of the order user is displayed the error message "Something went wrong, please try again".

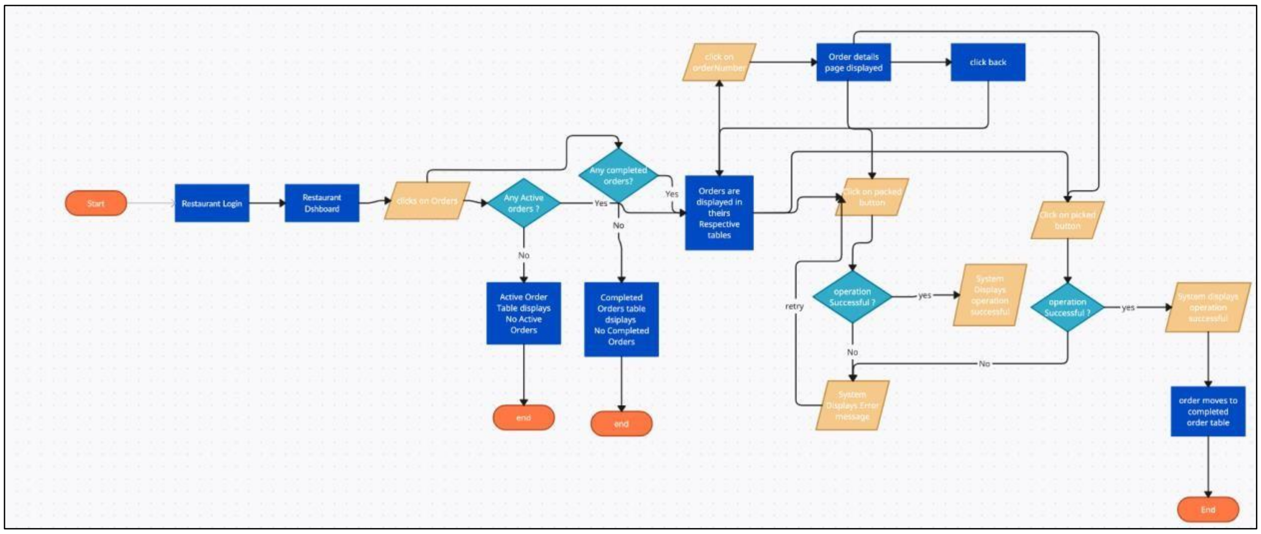

Start
Restaurant Login
Restaurant Dshboard
clicks on Orders
Any Active orders ?
Yes
No
Any completed orders?
Yes
No
Active Order Table displays No Active Orders
Completed Orders table dsiplays No Completed Orders
end
end
Orders are displayed in theirs Respective tables
click on orderNumber
Order details page displayed
click back
Click on packed button
Click on picked button
operation Successful ?
yes
System Displays operation successful
No
retry
System Displays Error message
operation Successful ?
yes
No
System displays operation successful
order moves to completed order table
End

Figure 46: Task flow diagram for Order Change Status, Created using Xtensio [6].

User Scenario- View orders by user, their timeslots, and items: The information regarding the extra food offered at the LastServe has been posted by the restaurant manager. In order to pre-pack the orders and allocate resources for order collection, he wants to check the users and their appointment time slots to pick up the food, goods, and quantity they ordered on the application. He looks at the dashboard to see all orders that have been placed and their time slots.

1. The restaurant manager has successfully logged in.
2. Restaurant managers see the Restaurant dashboard with various functions.

3. The restaurant manager clicks on the orders.
4. Two tables are displayed One for the active orders and completed past orders.
   4.1 There are no active orders user is displayed “there are no active orders”.
5. There are no past orders for the restaurant. The active orders table has the information on the order number, the Name of the user items ordered appointmenttime, the status of the order, and actions to be performed on the order.

   5.1 Order with status pending is the order that is just placed, in the actions twobuttons are shown, packed and picked. Order with status packed is the order that is packed but not picked up, in theactions one button is shown as picked.

   5.2 The restaurant manager wishes to see other orders than the ones displayed in the active orders table.
6. The restaurant manager clicks on the next page tab at the bottom of the table.
7. The restaurant manager wants to see the orders sorted based on appointment time,he clicks on the appointment time header.
8. The system sorts the orders and displays the sorted order.
9. The system displays past in the table below active order with the following columns order number, name, items.

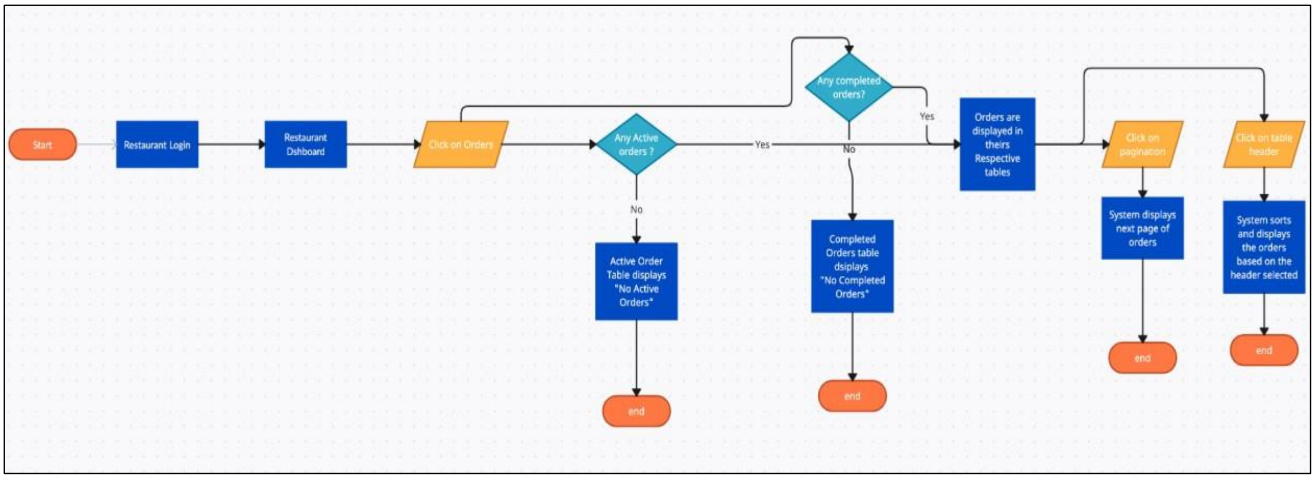

Start
Restaurant Login
Restaurant Dshboard
Click on Orders
Any Active orders ?
Yes
No
Active Order Table displays "No Active Orders"
end
Any completed orders?
Yes
No
Completed Orders table dsiplays "No Completed Orders"
end
Orders are displayed in theirs Respective tables
Click on pagination
System displays next page of orders
end
Click on table header
System sorts and displays the orders based on the header selected
end

Figure 47: Task flow diagram to view timeslots and items, Created using Xtensio [6].

User Scenario- View Order History: A student needs to view the number of days in the past week where he/she picked leftovers from restaurants. The student needs to view the order history on LastServe.

1. System displays the user home page.
2. User clicks on ‘Profile’ icon.
3. System displays ‘User Profile’ page.
4. User clicks on the ‘Orders’ button.
5. System displays a list of past orders.
6. User clicks on ‘Home’ button.
7. System redirects user to the home page.

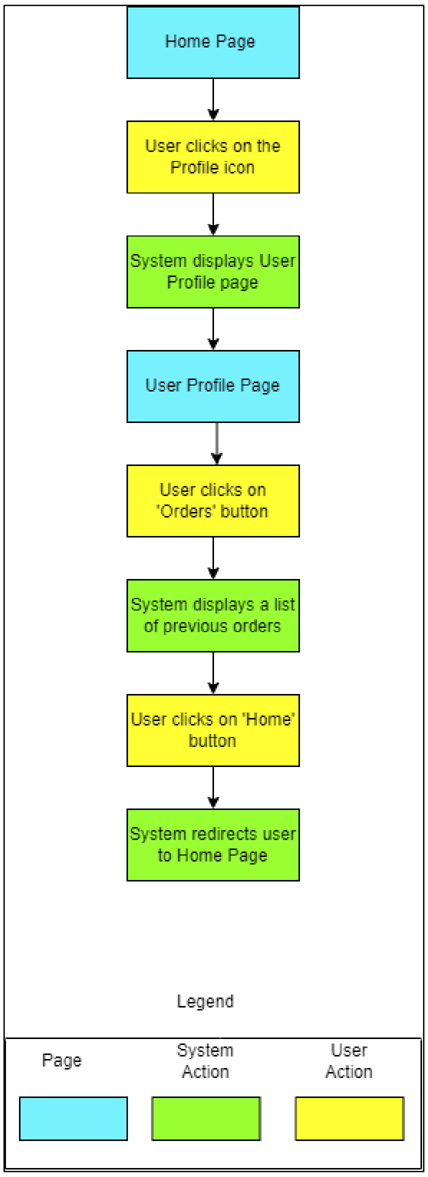

Home Page
User clicks on the Profile icon
System displays User Profile page
User Profile Page
User clicks on 'Orders' button
System displays a list of previous orders
User clicks on 'Home' button
System redirects user to Home Page
Legend
Page
System Action
User Action

Figure 48: A wireframe for User order history, Created using Draw.io [5].

User Scenario- View Current Order Status: The student has booked an appointment to pick up food at a nearby restaurant. Before leaving for the pick-up the student needs to verify is the restaurant has approved the order because at times restaurants dump posted food items that turn stale before it was picked up.

1. Assumption – User is logged in
2. System displays the user home page.
3. User clicks on ‘Profile’ icon.
4. System displays ‘User Profile’ page.
5. User clicks on the ‘Orders’ button.
6. System displays a list of current orders and its status.
7. User views order status.
8. User clicks on ‘Home’ button.
9. System redirects user to the ‘home’ page.

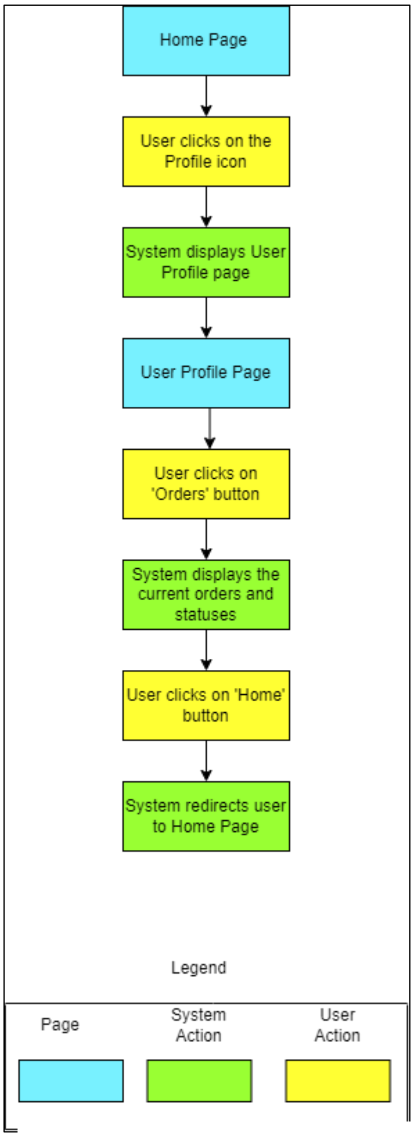

Home Page
User clicks on the Profile icon
System displays User Profile page
User Profile Page
User clicks on 'Orders' button
System displays the current orders and statuses
User clicks on 'Home' button
System redirects user to Home Page
Legend
Page
System Action
User Action

Figure 49: A wireframe for Current Order Status, Created using Draw.io [5].

User Scenario- Cancel Current Order- A student has scheduled an appointment to pick up food from a restaurant. Sometime after booking the appointment the weather conditions deteriorate and the student decides to step out for the pickup. To enable another person to pick up the food, the user wants to cancel the scheduled appointment.

1. System displays the user home page.
2. User clicks on ‘Profile’ icon.
3. System displays ‘User Profile’ page.
4. User clicks on the ‘Orders’ button.
5. System displays a list of current orders and its status.
6. User finds the current order to be cancelled.
7. User clicks the ‘Cancel’ button.
8. System displays ‘Confirm cancellation: Yes or No’ message.
  8.1 User clicks on ‘No’ button.
  8.2 System removes the message.
9 User clicks on ‘Yes’ button.
10 System displays ‘Cancelled successfully’ message.

11 System displays the list current orders and its status.

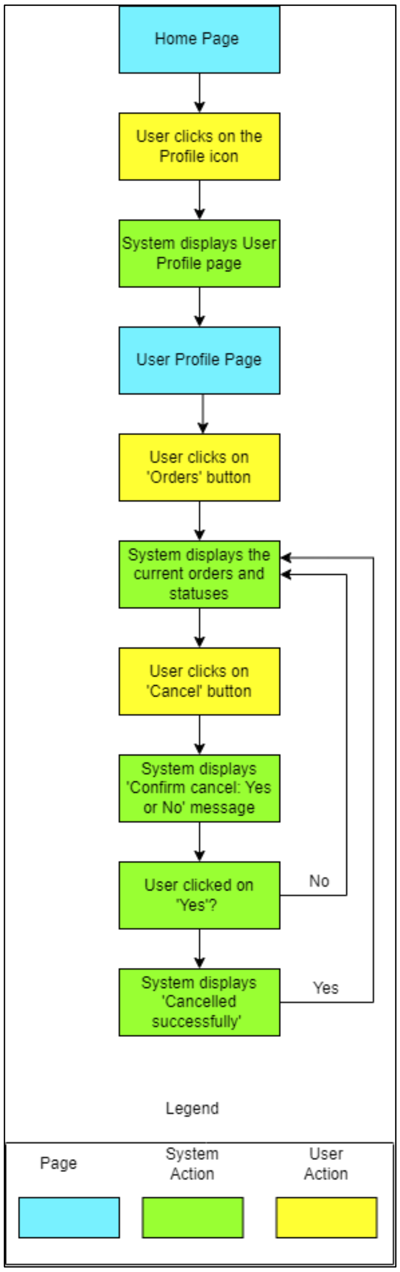

Home Page
User clicks on the Profile icon
System displays User Profile page
User Profile Page
User clicks on 'Orders' button
System displays the current orders and statuses
User clicks on 'Cancel' button
System displays 'Confirm cancel: Yes or No' message
User clicked on 'Yes'?
No
System displays 'Cancelled successfully'
Yes
Legend
Page
System Action
User Action

Figure 50: A wireframe to cancel order, Created using Draw.io [5].

User Scenario- Volunteer Registration: User wants to work as a volunteer for a food donation chain in her free time. She came across a website called LastServe; who register restaurants that have excess food. She begins her registration as a volunteer on LastServe website who in turn will forward her details to the restaurants who need volunteers.

1. User visits website “www.lastServe.ca” to register as a volunteer.
2. System displays the home page.

3. User clicks on the button labelled as “Register Volunteer”.

4. System re-directs user to the Volunteer registration page.

5. System prompts the user to fill the registration form,

6. On the registration page the user will fill up a form to provide all the details.

6.1 System firstly, requires the user’s first and last name.

6.1.1 User enters the First name and not the Last name.

6.1.2 Systems prompts “Please enter Last Name”.

6.1.3 User enters First name and Last name again.

6.2 System requires the user to enter an email id.

6.2.1 User enters their email id.

6.2.2 System prompts the user to enter email id in the correct format.

6.2.3 User enters email id again.

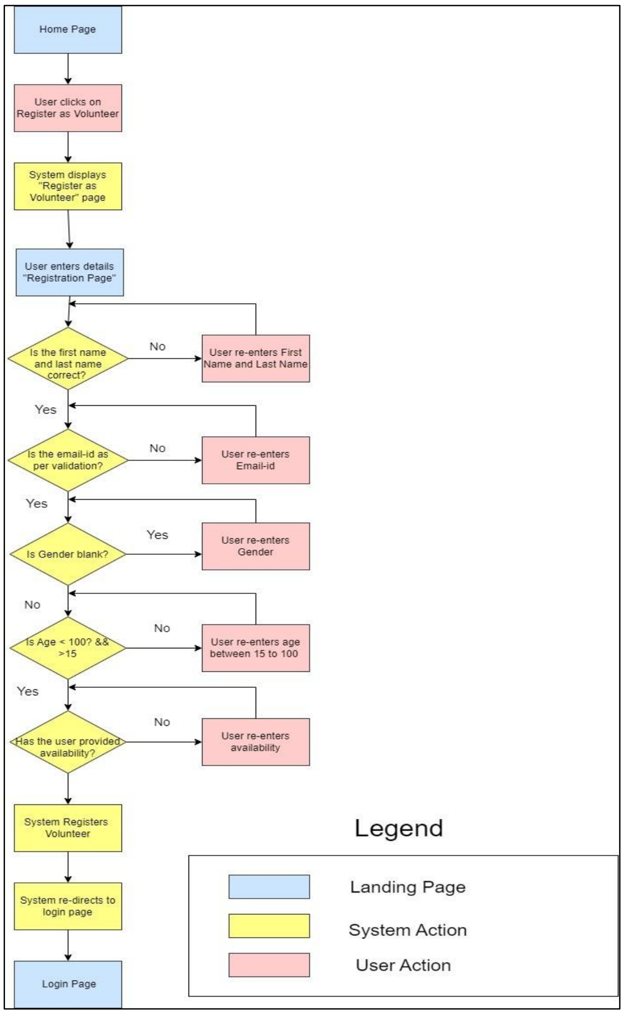
6.3 System requires the user to enter their phone number.

6.3.1 User enters a 10-digit phone number.

6.4 System requires the user to choose their gender and availability.

7. User will fill up the above details and hit submit.

8. The system will send the details for approval.



Home Page
User clicks on Register as Volunteer
System displays "Register as Volunteer" page
User enters details "Registration Page"
Is the first name and last name correct?
No
User re-enters First Name and Last Name
Yes
Is the email-id as per validation?
No
User re-enters Email-id
Yes
Is Gender blank?
Yes
User re-enters Gender
No
Is Age < 100? && >15
No
User re-enters age between 15 to 100
Yes
Has the user provided availability?
No
User re-enters availability
System Registers Volunteer
System re-directs to login page
Login Page
Legend
Landing Page
System Action
User Action

Figure 51: Task flow diagram for Volunteer Registration, Created using Draw.io [5].

User Scenario- Volunteer Approval: Users will provide their details to LastServe by filling up a registration form. Only the approved volunteer details will be sent forward to restaurants for hiring.

1. User submitted the registration form on LastServe website.
2. The registration details are now forwarded to the admin for approval.
3. The registration details of the user are now open on the admin page.
4. Admin verifies the details against all the requirements.
5. Based on approval or rejection the admin will notify the user about their selection.
6. Depending on requirements from restaurants the admin will sort the application and send it further.

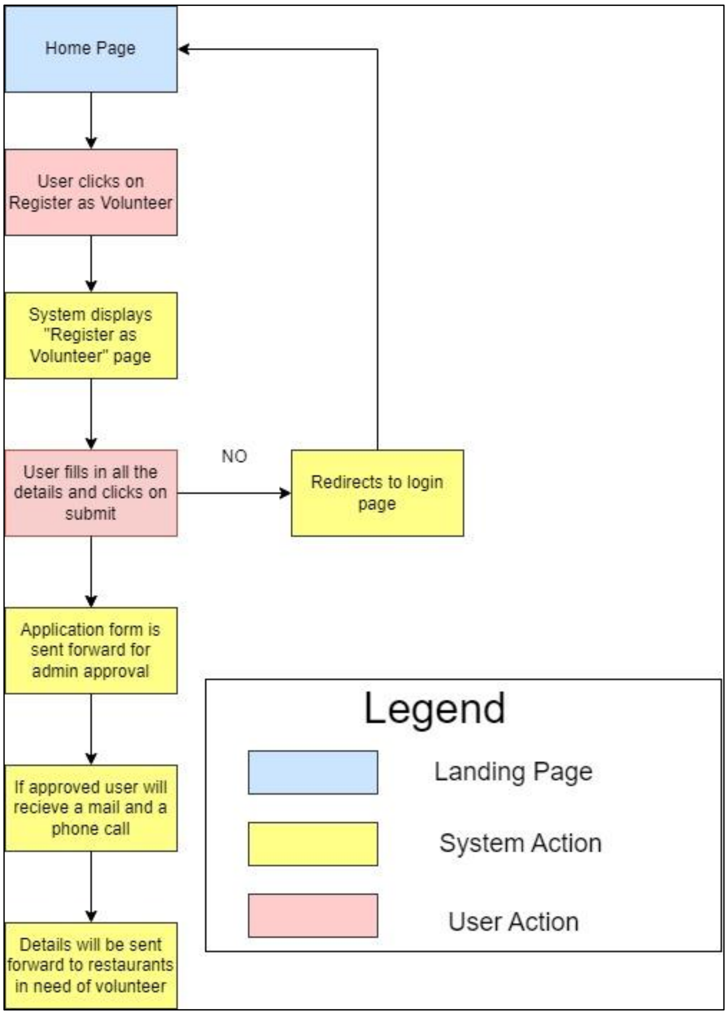

Home Page
User clicks on Register as Volunteer
System displays "Register as Volunteer" page
User fills in all the details and clicks on submit
NO
Redirects to login page
Application form is sent forward for admin approval
If approved user will recieve a mail and a phone call
Details will be sent forward to restaurants in need of volunteer
Legend
Landing Page
System Action
User Action

Figure 52: Task flow diagram for Volunteer Approval, Created using Draw.io [5].

User Scenario- Subscribe to a Restaurant: The student likes food from a particular restaurant and eagerly waits for the restaurantto post about leftover food. At times, he gets too busy completing assignments and misses posts from the restaurant. He frequently checks his inbox for important emailsfrom the university or his manager. The student wants to receive a notification whenever his favorite restaurant posts on LastServe.

1. System displays the home page.
2. User enters the restaurant name in the search bar.
3. System displays the search results with the restaurant details.
4. User clicks on the ‘subscribe for alerts’ button.
5. System displays a ‘Subscribed successfully’ message.
6. System replaces subscribe button with a ‘subscribed’ text.
7. User clicks on the back button.
8. System redirects to the home page.

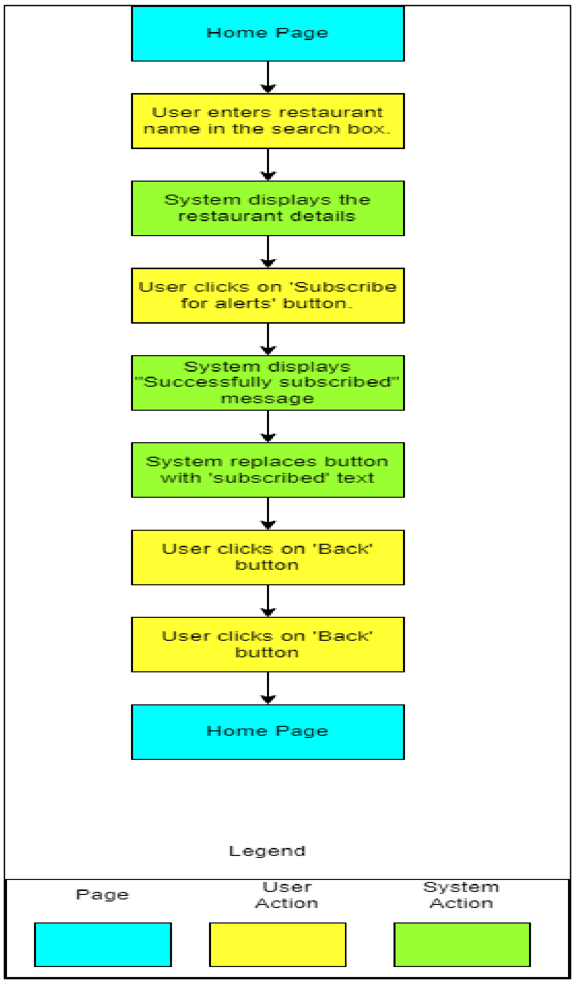

Home Page
User enters restaurant name in the search box.
System displays the restaurant details
User clicks on 'Subscribe for alerts' button.
System displays "Successfully subscribed" message
System replaces button with 'subscribed' text
User clicks on 'Back' button
User clicks on 'Back' button
Home Page
Legend
Page
User Action
System Action

Figure 53: Task flow diagram to Subscribe to a Restaurant, Created using Draw.io [5].

User Scenario- Unsubscribe to a Restaurant: The student gets email alerts when a subscribed restaurant posts about food availability. One such restaurant is located on a street that has witnessed crimes during the night in recent weeks. The student fears his/her safety and no longer wishes to pick up food from the restaurant. Hence, wants to unsubscribe to the restaurant.

1.User clicks on the ‘Edit Profile’ button on the home page.

2.System displays the edit profile page which lists all subscribed restaurants.

3.User clicks on the ‘Back’ button.

4.System redirects the user to the home page.

5.User clicks on the ‘remove button’ beside the restaurant’s name.

6.System displays a ‘Restaurant unsubscribed successfully’ message.

7.User clicks on the ‘Back’ button.

8.System redirects user to the home page.

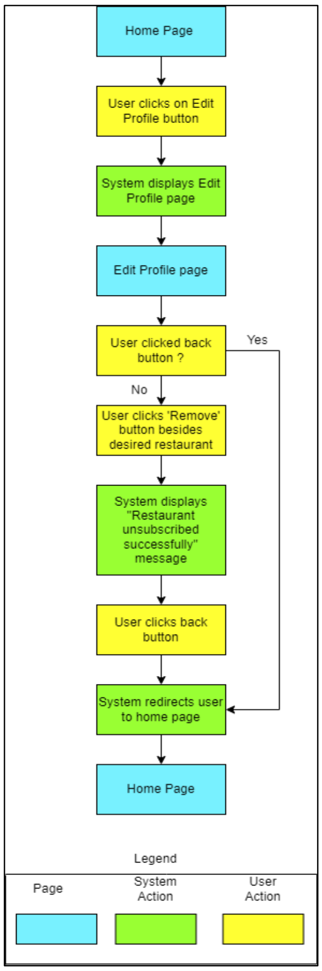

Home Page
User clicks on Edit Profile button
System displays Edit Profile page
Edit Profile page
User clicked back button ?
Yes
No
User clicks 'Remove' button besides desired restaurant
System displays "Restaurant unsubscribed successfully" message
User clicks back button
System redirects user to home page
Home Page
Legend
Page
System Action
User Action

Figure 54: Task flow diagram to Unsubscribe to a Restaurant, Created using Draw.io [5].

User Scenario- Donating money on LastServe: The notion of LastServe working with restaurants to feed the needy and raise awareness of food waste was praised by a user. They make the decision to give some money to the participating restaurants.

1. User opens the browser and types www.lastserve.ca
2. System displays the home page.
3. User clicks on ‘Register as a user’ button on the home page.

4. System displays the registration page, requesting first name, last name, email address and password.
5. User enters the details.
6. User clicks the ‘register’ button.
7. If all the criteria are not met, then the user is prompted to type in the details again.
8. If all the criteria are met a successfully signed-up message is displayed on the screen.
9. The navigation bar displays various tabs one of which is the “Donation Tab”.
10. User clicks on the donation tab.
  10.1 System displays the Donation page.
  10.2 System firstly, requires the user’s first and last name.
    10.2.1 User enters the First name and no Last name.
    10.2.2 Systems prompts “Please enter Last Name”.
    10.2.3 User enters First name and Last name again.
  10.3 System prompts user to enter the desired amount for donation.
  10.4 System requires the user to enter email id.
    10.4.1 User enters their email id.
    10.4.2 System prompts user to enter email id in the correct format.
    10.4.3 User enters email id again.
  10.5 System prompts user to choose a restaurant to donate money to from the drop down box.
  10.6 System then prompts the user to enter card details.
11. Once all the details are filled correctly the user clicks on Donate.

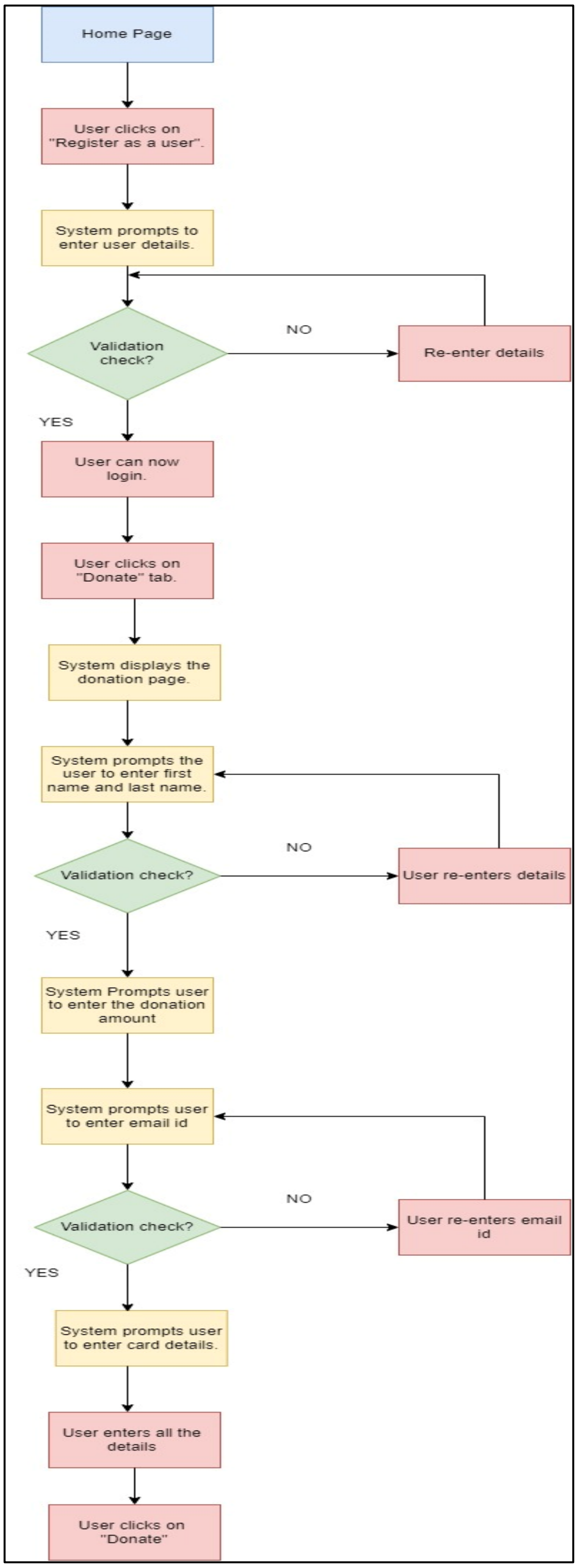

Home Page
User clicks on "Register as a user".
System prompts to enter user details.
Validation check?
NO
Re-enter details
YES
User can now login.
User clicks on "Donate" tab.
System displays the donation page.
System prompts the user to enter first name and last name.
Validation check?
NO
User re-enters details
YES
System Prompts user to enter the donation amount
System prompts user to enter email id
Validation check?
NO
User re-enters email id
YES
System prompts user to enter card details.
User enters all the details
User clicks on "Donate"

Figure 55: Task flow diagram to Donate to a Restaurant, Created using Draw.io [5].

User Scenario- Viewing Donation History:

1. User submits donation form on the LastServe website.
2. User clicks on history.
3. The history tab reveals all the donations made by the user.
4. The history tab contains details about the amount the doner has donated in the past and to which restaurant.
5. System also displays the donation history to restaurants.

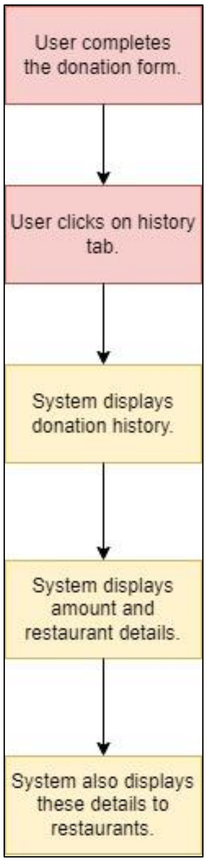

User completes the donation form.
User clicks on history tab.
System displays donation history.
System displays amount and restaurant details.
System also displays these details to restaurants.

Figure 56: Task flow diagram to View Donation History, Created using Draw.io [5].

3. ASSET INVENTORY

a) Client-side languages:

The list of client-side technologies used in developing the LastServe website are:

- **HTML 5:** Used to create templates for the user interface on the LastServe website.
- **CSS:** Used to adjust HTML templates' styles and align elements.
- **React**: Used for server-side API calls as well as client-side API validation.

b) Server-side languages:

The list of server-side technologies used in developing the LastServe website are:

- **Node.js**: Used to fetch the data that the client side requests from the database as well as for server-side validations.
- **NoSQL (MongoDB)**: Used to create database queries that would retrieve the necessary data..

c) API:

The list of APIs used in developing the application is as follows:

- REST API: Used to make API calls from the client side to the server side.

d) Images:

The list of image formats we used in developing our application are:

- jpeg
- jpg
- png

4. GROUP ROLES

The group roles are shown in the following table (see **Table 1**).

Table 1: Group role

Name	**Primary role**	**Secondary role**
Arpit Ribadiya	Full Stack Developer	Backend developer
Jay Kania	Full Stack Developer	UI Designer
Lav Patel	Full Stack Developer	Database manager
Neha Karkhanis	Full Stack Developer	Document Manager
Viraj Joshi	Full Stack Developer	Scrum Master

5. REFERENCES

[1] “Food Waste Statistics in Canada for 2023 - Made in CA,” *Made in CA*, Sep. 15, 2022. https://madeinca.ca/food-waste-canada-statistics/ (accessed Feb. 28, 2023).

[2] MOONSHOT COMPOSTING, “Restaurant Food Waste – How Much Food Is Thrown Away by Restaurants & Why?,” *MOONSHOT COMPOSTING*, Jun. 03, 2020. https://www.moonshotcompost.com/restaurant-food-waste-how-much-why/ (accessed Feb. 28, 2023).

[3] writemaps.com, “WriteMaps | Create Sitemaps Online,” *Writemaps.com*, 2019. https://writemaps.com/ (accessed Feb. 28, 2023).

[4] “Balsamiq. Rapid, Effective and Fun Wireframing Software | Balsamiq,” *Balsamiq.com*, 2023. https://balsamiq.com/ (accessed Feb. 28, 2023).

[5] “Flowchart Maker & Online Diagram Software,” *Diagrams.net*, 2023. https://app.diagrams.net/ (accessed Feb. 28, 2023).

[6]“Create powerful business content together with Xtensio,” *Xtensio.com*, Oct. 02, 2022. https://xtensio.com/ (accessed Feb. 28, 2023).