



**DALHOUSIE  
UNIVERSITY**

Master of Applied Computer Science

**CSCI 5709**

**Advanced Web Services**

**Assignment 1 (Group)**

**Group Members –**

1. Viraj Joshi

2. Lav Patel

3. Jay Kania

4. Neha Karkhanis

5. Arpit Ribadiya

## **A1.1. Project's Purpose, Goals and Intended Features**

### **1. Purpose**

Our goal is to end world hunger. We consider having access to food a fundamental human right, not a privilege. Nobody should go to bed hungry or struggle to get food each day. This initiative aims to create a community where restaurants can donate meals to the less fortunate.

Every night, restaurants with leftover food will update their public posting on our website with the amount they still have. In this manner, anyone in need of food can reserve a time slot and come pick up their portion of food for that evening. This will prevent food waste in restaurants and prevent people from going to bed hungry.

### **2. Goal**

The goal 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 tonnes of food is wasted each year, according to the most recent assessment [1]. In addition, it says that one-third of the food eaten was leftover. This project aims to distribute food that would otherwise be wasted and provide it to those in need.

### **3. Target user base**

Our project's goal is to reduce food waste and make it accessible to those in need. Our target demographic would be international students, the working class, or the homeless.

### **4. Intended Features**

#### **4.1 Features and tasks**

##### **1. User Profile Management**

- a. User Registration
- b. View User Profile
- c. Update User Profile
- d. Existing User Login

##### **2. Restaurant Profile Management**

- a. Restaurant Registration
- b. View Restaurant Profile
- c. Update Restaurant Profile
- d. Restaurant Login

##### **3. Food Donation Post Management**

- a. Create a new post
- b. View past Posts
- c. Edit active Posts

**4. Admin Content Moderation**

- a. Admin Dashboard visualization.
- b. Delete posts.
- c. Approve restaurants.

**5. Restaurant order management**

- a. View orders by user, their timeslots, and items.
- b. Change order status from pending to packed and picked.
- c. Get available volunteer information based on availability.

**6. User Order management**

- a. View all active donors with timings and items offered.
- b. Filter through all active donors based on location, restaurant name and type of food.
- c. Appointment slot booking for pickup.
- d. Email of confirmed order once an appointment is booked.

**7. Volunteer Management**

- a. Apply for volunteering with the application
- b. Volunteer application approval

**8. Subscription management**

- a. Users can subscribe to posts from a restaurant
- b. Users can unsubscribe to a restaurant.

**9. Donation Management**

- a. Users can donate money to restaurant
- b. Restaurant can see all donations received

**10. User order history Management**

- a. Users can look through their previous order history
- b. Users can check status of current order
- c. Cancel active orders

## A1.2. User Personas and Intended Scenarios

### User Personas

Following is the audience category that has been identified for our project:

**Student:** The type of user that will sign in to access the free meal service (see **Figure 1**).

**Manager of a restaurant:** This user will modify postings and add restaurant information, among other things (see **Figure 2**).

**Volunteer:** This user persona will utilise our website to sign up to work as a volunteer at a restaurant. (see **Figure 3**).

### 1.Student

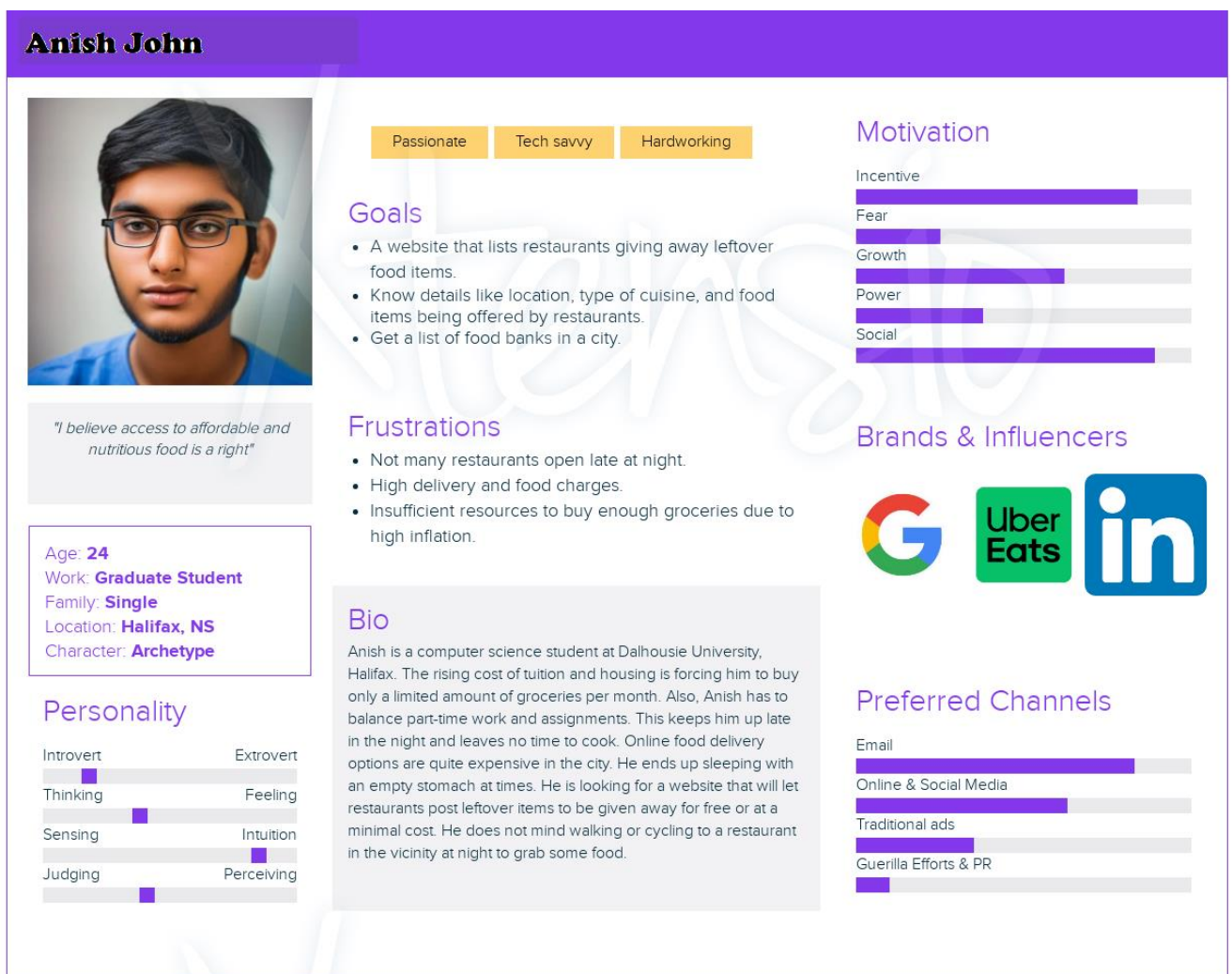
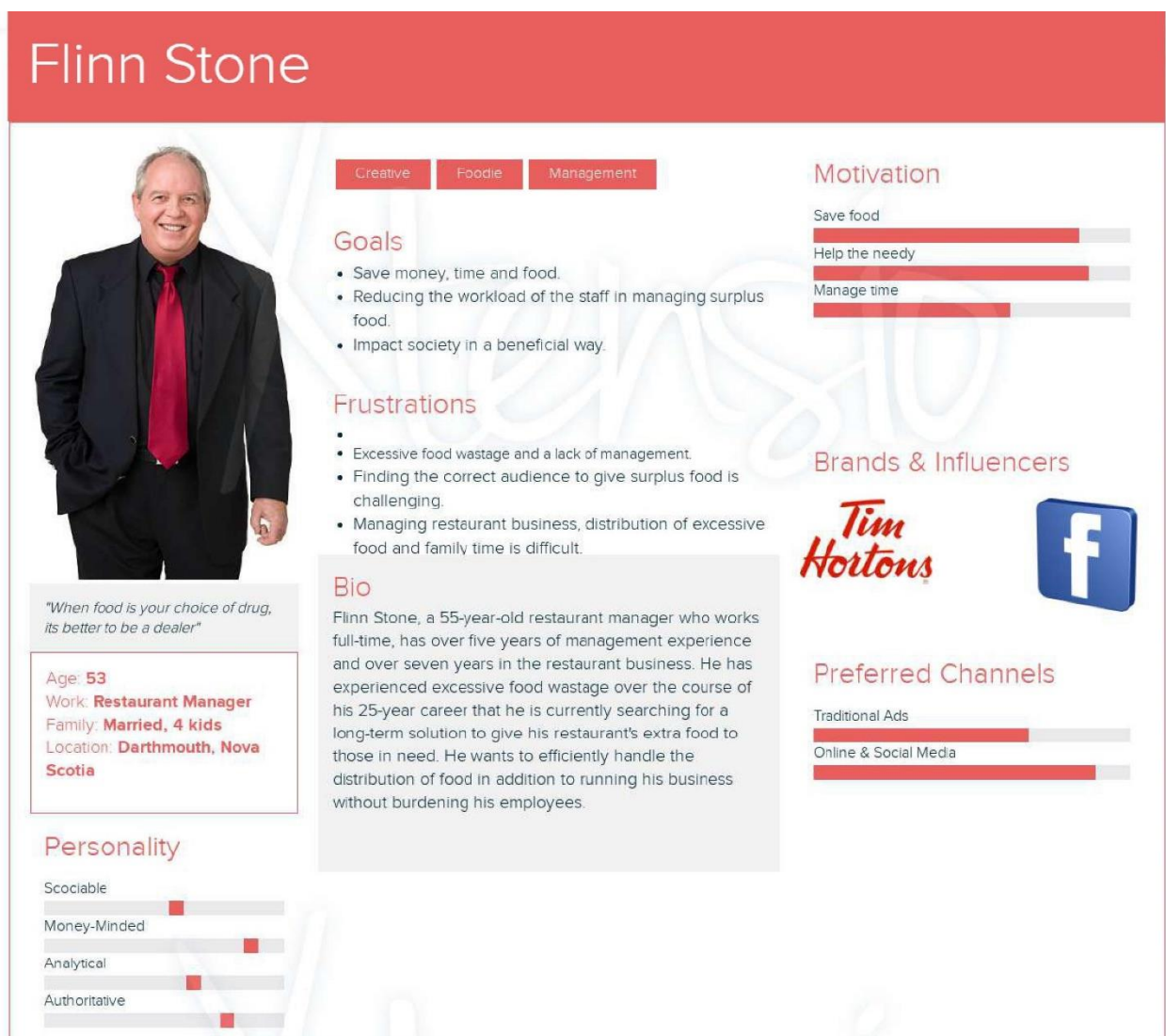


Figure 1: Anish John: Student, made using Xtensio.com [2]

## 2. Restaurant Manager



**Figure 2:** Flinn Stone: Restaurant Manager, made using Xtensio.com [2]

### 3. Volunteer

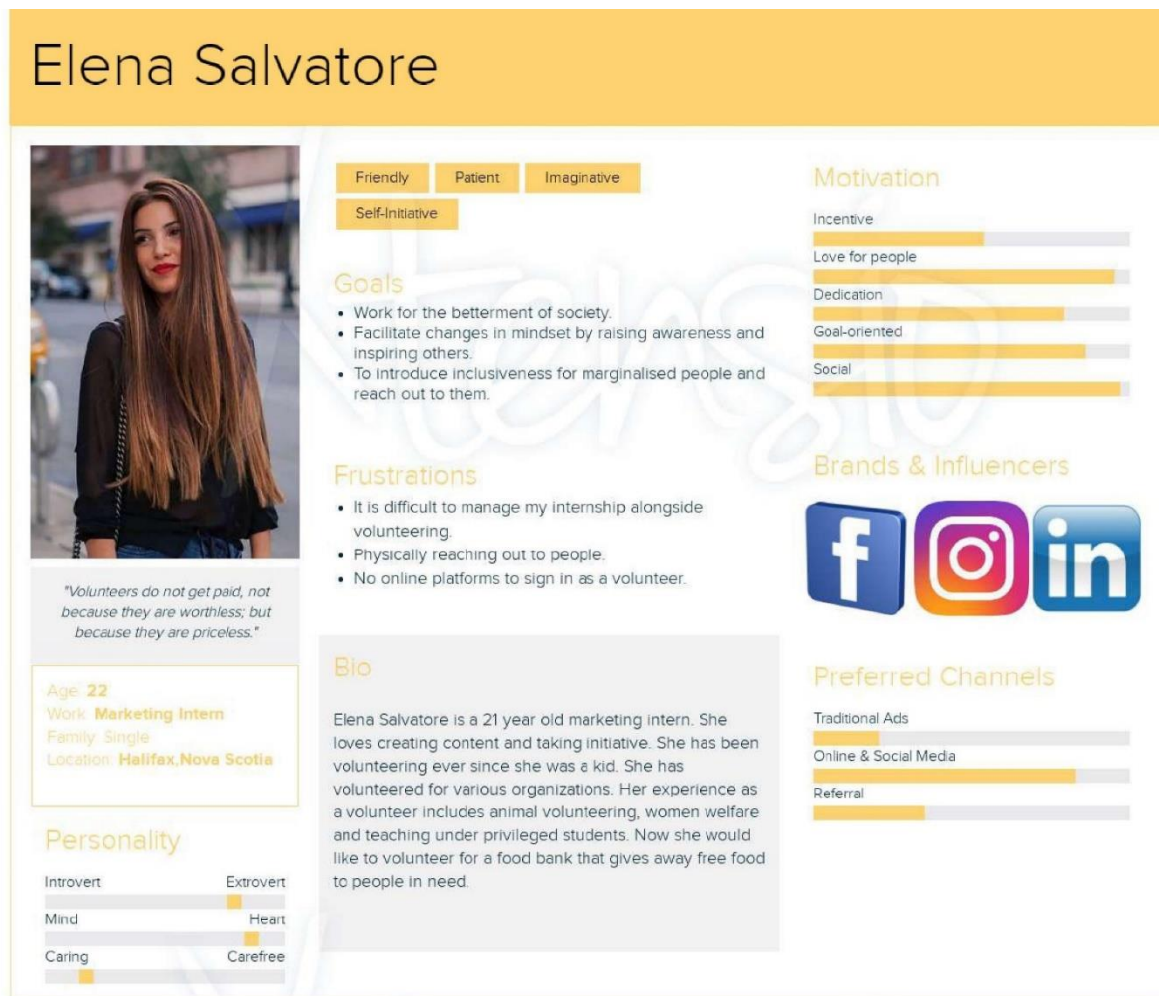


Figure 3: Elena Salvatore: Volunteer, made using Xtensio.com [2]

## Scenarios

We have chosen the following features from the list of features provided in A1.2. for Assignment 1

- User profile management
- Restaurant profile management
- Restaurant order management
- Volunteer management
- Subscription management

### 1. User registration on LastServe

A student needs food at midnight but is too tired to cook a meal. The student does not want to order food from a restaurant due to a tight budget. The student wants to create an account with a website that lists restaurants in the vicinity offering leftovers from the day at no cost.

**Persona:** Student

**Feature:** Create an account

**Need:** Find restaurants offering food for free

**Context:** Could not afford to order food from a restaurant

### 2. Log into account

A student is hungry and wants to check restaurants in the vicinity offering free leftover food on the LastServe website.

**Persona:** Student

**Feature:** Login account

**Need:** Find nearby restaurants offering free food

**Context:** Hungry student looking for free food

### 3. 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.

**Persona:** Student

**Feature:** View user profile

**Need:** See all the restaurants for which an email alert has been set

**Context:** The user is logged in and wants to view details

### 4. Edit User Profile

While creating an account, the student mis-spelled his name. Also, the student did not enter the last name as it was an optional field. The last time he went to a restaurant, he was not allowed to pick up food as the booking details had a different name than his ID. The student wants to change his first name and add last name to avoid any misunderstandings in the future.

**Persona:** Student

**Feature:** Edit user profile

**Need:** Update personal details

**Context:** Not allowed to pick up food as names did not match

### 5. Restaurant Registration

Every end of the day, a lot of food left so, the restaurant manager wonders what to do with this food. He is looking for an application where he can register to donate this remaining food.

**Persona:** Restaurant Manager

**Feature:** Restaurant Registration

**Need:** Looking for application which can help in donating food

**Context:** Restaurant Manager will use application to register restaurant for donating food items.



## 6. **Restaurant Login**

Restaurant manager wants to perform different action based on rights he has in the system and for that he needs to login into the system.

**Persona:** Restaurant Manager

**Feature:** Restaurant Login

**Need:** Wants to perform different operation in the system

**Context:** Restaurant Manager will use application to login into the system and then he can perform various task based on his role.

## 7. **View Restaurant Profile**

Restaurant manager wants to see all the required details he filled up during registering his restaurant.

**Persona:** Restaurant Manager

**Feature:** View Restaurant Profile

**Need:** Wants to see restaurant details

**Context:** Restaurant Manager will use application to view all information which he entered while registering his restaurant into our system.

## 8. **Update Restaurant Profile**

Restaurant manager want to edit some or all the details he filled up during registering his restaurant.

**Persona:** Restaurant Manager

**Feature:** Update Restaurant Profile

**Need:** Wants to edit restaurant details

**Context:** Restaurant Manager will use application to update any information about restaurant which he entered while registering his restaurant into our system.

## **9. 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.

**Persona:** Volunteer

**Feature:** Volunteer Registration

**Need:** Register to be a volunteer for food distribution

**Context:** The user is filling up the registration form.

## **10. 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.

**Persona:** LastServe Employee

**Feature:** Volunteer Registration

**Need:** Volunteer details are cross verified against requirements

**Context:** Registered volunteers need admin approval

## **11. View orders by the user, their timeslots, and items**

Restaurant Manager has posted the information on excess food available on the last serve. He wants to check the users and their appointment time slots to collect the food, and items and quantity ordered by them on the application so he can pre-pack the orders and allocate resources for order collection. He checks through the dashboard of all orders placed and their time slots.

**Persona:** Restaurant Manager

**Feature:** Restaurant order viewing

**Need:** Check the appointment and better manage the time and food

**Context:** They have placed the post and want to see the information on users who have ordered.

## **12. Change order status from pending to packed and picked**

The restaurant is viewing all the orders placed by users. He wants to change the order status once he has packed an order so other employees working on it can know to pack other orders. he also wants to mark an order as picked up once it is picked up by the user so it is easier to manage the order. He clicks on the packed and picked-up buttons once it is done.

**Persona:** Restaurant Manager

**Feature:** Change order status from pending to packed and picked

**Need:** keep track of orders that are packed and picked, so the pending orders can be worked on.

**Context:** Once the order is placed by the user Restaurant manager wants to track the order through different statuses.

## **13. Subscribe to a restaurant**

The student likes food from a particular restaurant and eagerly waits for the restaurant to 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 emails from the university or his manager. The student wants to receive a notification whenever his favourite restaurant posts on LastServe.

**Persona:** Student

**Feature:** Subscribe to a restaurant.

**Need:** Get notified when one of his favourite restaurants' posts on LastServe.

**Context:** Misses posts due to workload.

#### **14. 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.

**Persona:** Student.

**Feature:** Edit user profile.

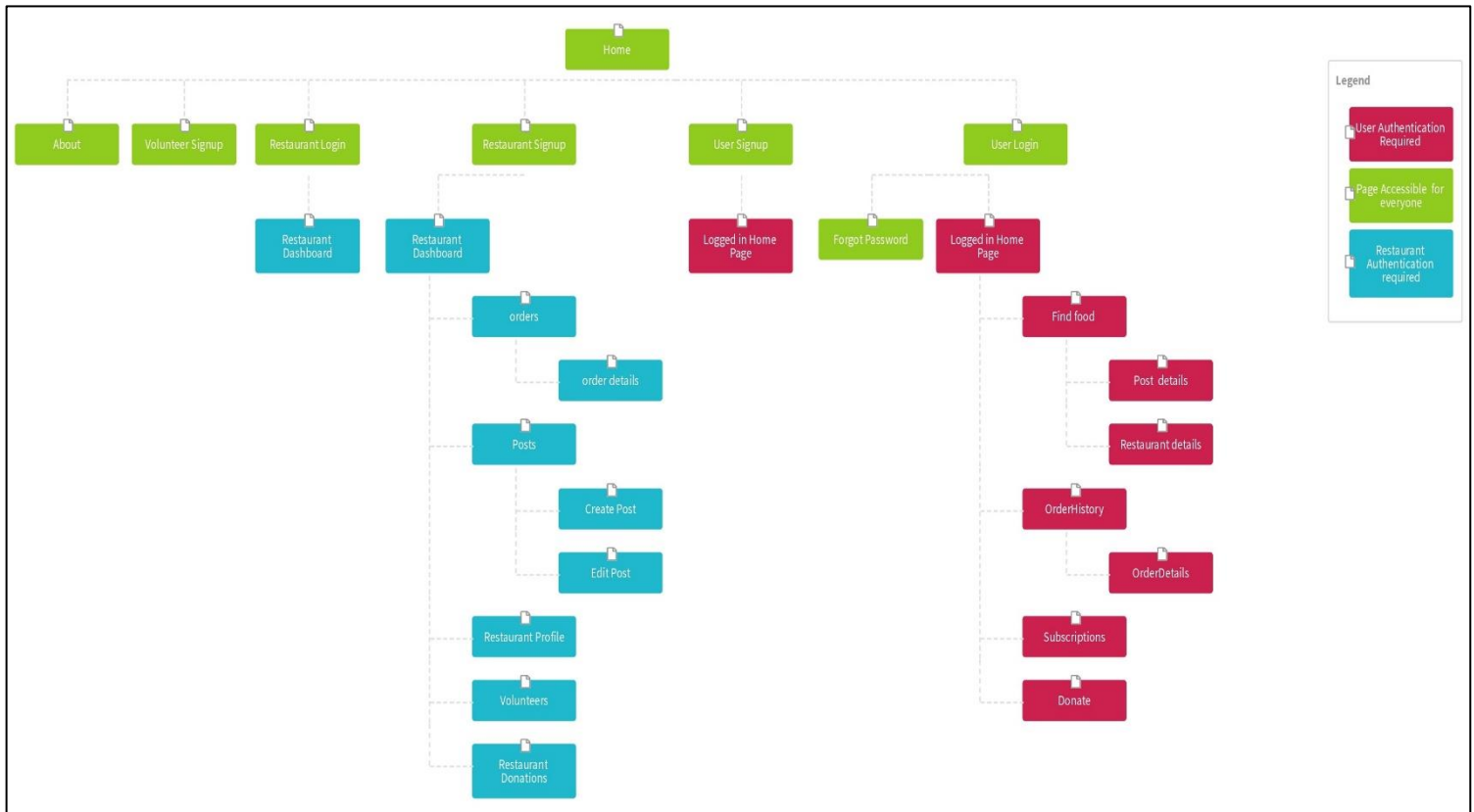
**Need:** No longer receive alters from a restaurant.

**Context:** safety concern forcing the student to avoid a restaurant.

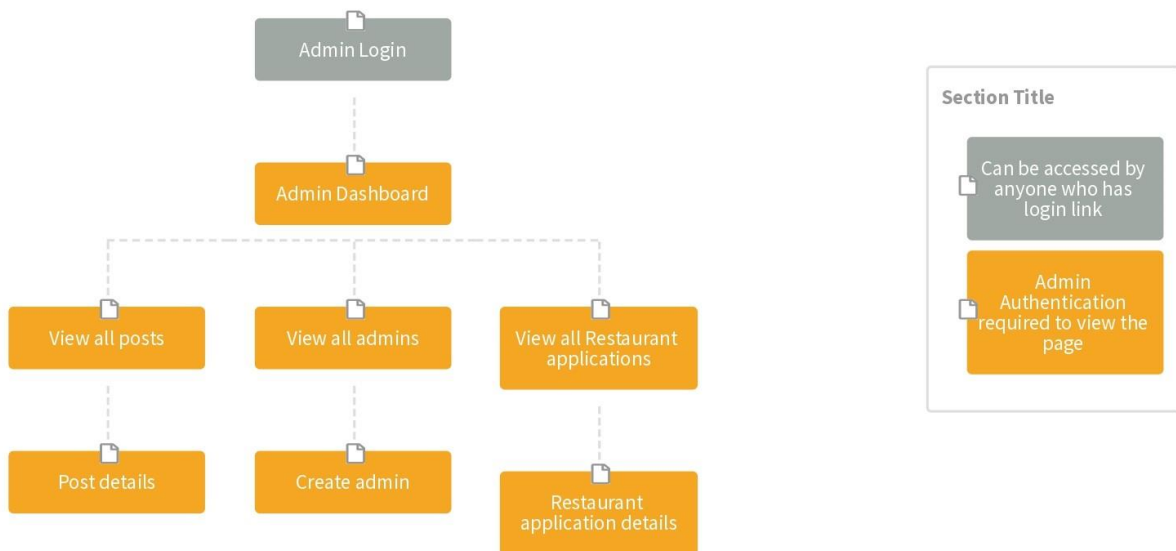
## A1.3. Sitemapping

### Sitemap

Here is the sitemap we created for our "LastServe" project (see **Figure 4**). A total of 31 pages makes up our website. We have also integrated several color-coded authentication locations.



**Figure 4:** SiteMap1, made using Writemaps [3]



**Figure 5:** SiteMap 2, made using WriteMaps[3]

## A.1.4. Use Cases

### Use Cases

#### 1.User Profile Management

##### Use case 1: User registration on LastServe

1. User opens the browser and types [www.lastserve.ca](http://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
  - 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 confirm 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

## **Use case 2: Log into account**

1. User opens the browser and types [www.lastserve.ca](http://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
  - 6.5. User clicks on 'Reset password' link
  - 6.6. System displays reset password page, requesting the user to enter the registered email address
    - 6.6.1. User enters email id
    - 6.6.2. 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 code
    - 6.7.2. User enters the authentication code
    - 6.7.3. User clicks on '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 confirm password
    - 6.8.2. User clicks on 'Reset Password' button



6.9. System displays “New password must be minimum 8 characters” message

6.9.1. User enters a new password

6.9.2. User clicks on ‘Reset Password’ button

6.10. System displays ‘New Password’ and confirm password does not match”

6.10.1. User enters a new password and confirm password

6.10.2. User clicks on ‘Reset Password’ button

7. System resets the password and displays ‘Password reset successful’ message

8. System redirects user to login page

9. System authenticates the user

10. System redirects the user to the home page

### **Use Case 3: View User Profile**

(Assumption – User is logged in)

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

#### **Use Case 4: Edit User Profile**

(Assumption – User is logged in)

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
  - 3.1.1. User clicks on the 'Back' button
  - 3.1.2. 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
    - 5.1.1. User enters first name
    - 5.1.2. System displays 'First Name updated' message
6. System updates details
7. System redirects user to the home page

## **2.Restaurant Order Management**

### **Use case 1: Change order status from pending to packed and picked**

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 went wrong, 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
  - 9.1.1 If there is an error while changing the state of the order user is displayed the error message “Something went wrong, please try again”

### **Use Case 2: View orders by the user, their timeslots, and items**

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, Name of the user items ordered appointment time, 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 two buttons are shown, packed and picked

- 5.2 Order with status packed is the order that is packed but not picked up, in the actions one button is shown as picked
6. The restaurant manager wishes to see other orders than the ones displayed in the active orders table
  7. The restaurant manager clicks on the next page tab at the bottom of the table
  8. The restaurant manager wants to see the orders sorted based on appointment time, he clicks on the appointment time header
  9. The system sorts the orders and displays the sorted order
  10. The system displays past in the table below active order with the following columns order number, Name, items

### **3.Volnteer Registration**

#### **Use Case 1: Volunteer Management**

- 1.User visits website to register as a volunteer. He lands on the landing page
2. User clicks on the button labelled as “Register Volunteer”
- 3.System re-directs user to the Volunteer registration page
4. System prompts the user to fill the registration form
5. On the registration page the user will fill up a form to provide all the details
6. System firstly, requires the users first and last name along with their contact details
7. User will fill up the above details and hit enter
8. Secondly the user will be prompted to provide their gender and occupation details
- 9.User will provide the details and hit enter
- 10.Finally, the system will prompt the user to provide their availability and experience as a volunteer and hit enter
- 11.Once the user has filled up all the details they can hit enter

#### **Use Case 2: Providing volunteer list to restaurants in need of volunteers.**

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

#### **4. Restaurant Profile Management**

##### **Use Case 1: Register Restaurant**

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 will open 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 in the 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”.

##### **Use Case 2: Update Restaurant**

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 restaurants details against all validation added in the system.
4. On successfully updating restaurant details restaurant manager will be redirected to restaurant dashboards.

#### **5. Subscription Management**

##### **Use Case 1: Subscribe to a restaurant**

(Assumption – The user is logged in)

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.

**Use Case 2: Unsubscribe to a restaurant**

(Assumption – User is logged in)

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 which lists all subscribed restaurants.
  - 3.1. User clicks on the 'Back' button.
  - 3.2. System redirects user to the home page.
4. User clicks on the 'remove button' beside the restaurant's name.
5. System displays 'Restaurant unsubscribed successfully' message.
6. User clicks on the 'Back' button.
7. System redirects user to the home page.

## **A.1.5. User Experience and Task Flow**

### **1.User profile management**

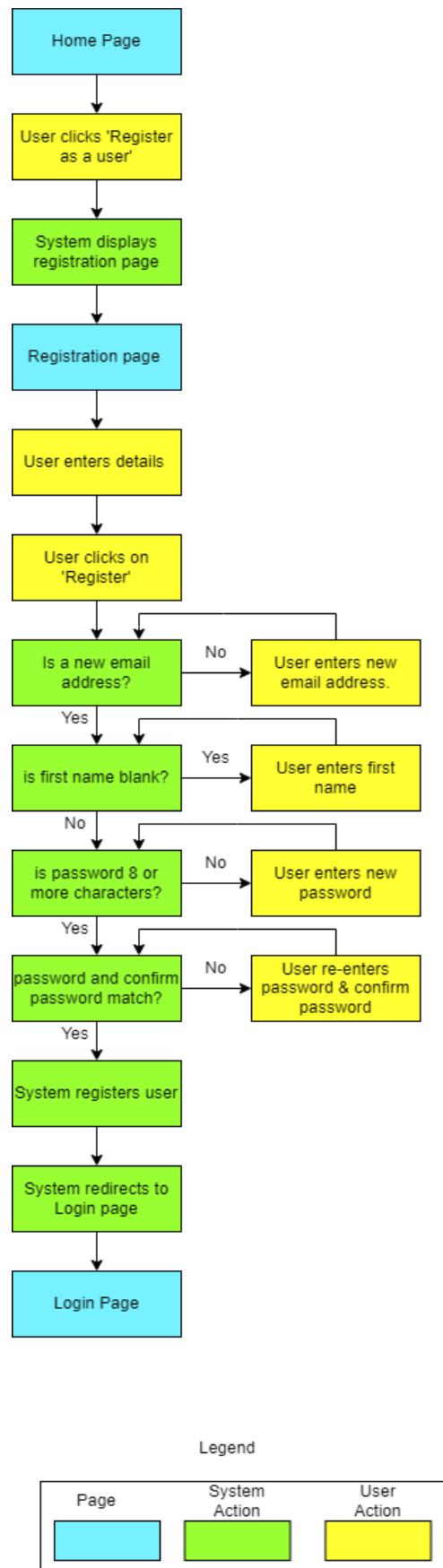
Feature Owner – Viraj Joshi (viraj.joshi@dal.ca)

#### **Tasks**

- a. User Registration
- b. Existing User Login.
- c. View User Profile
- d. Update User Profile

#### **Task Flow Diagrams**

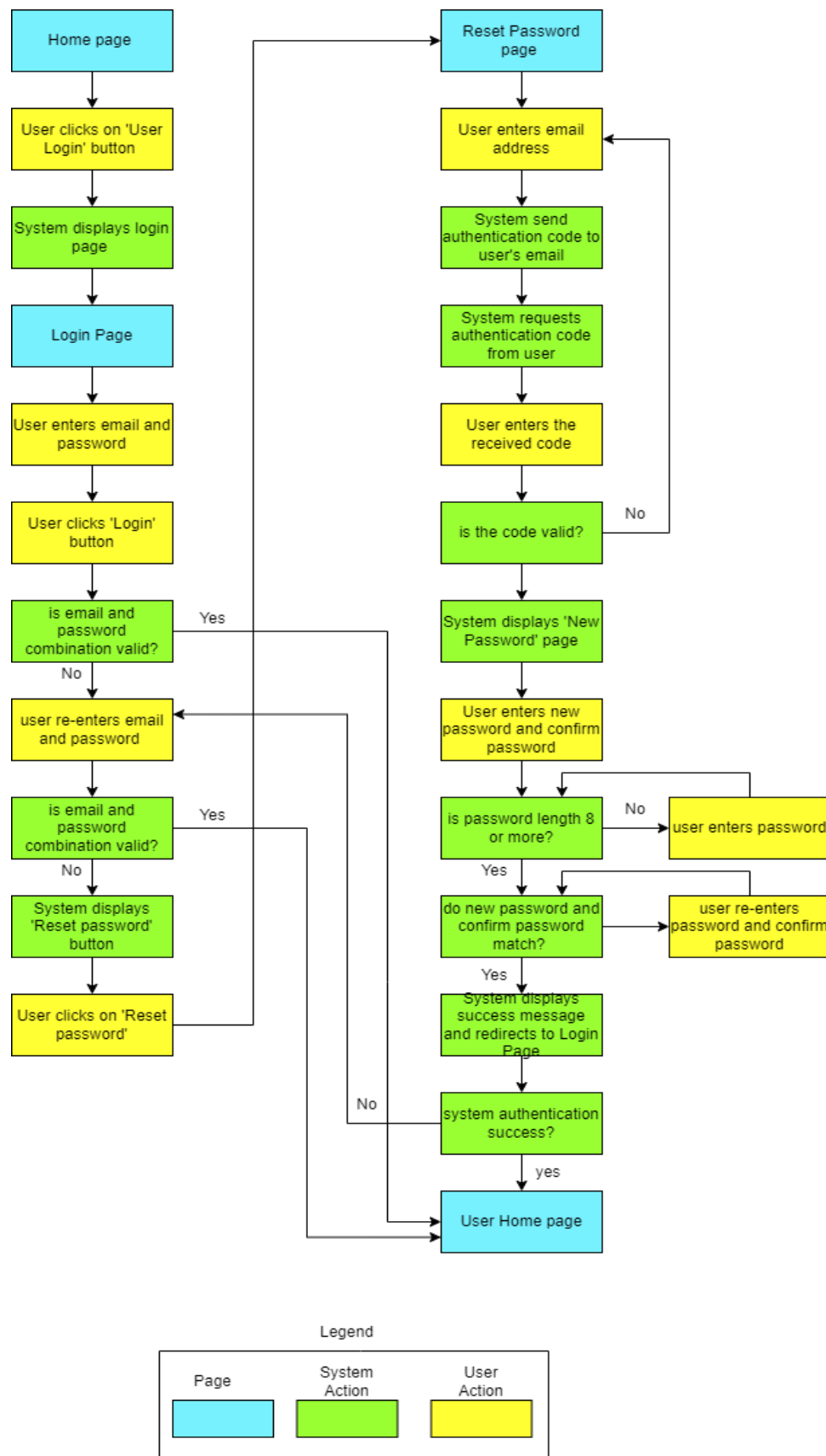
- a. User Registration



**Figure 6:** User registration task flow diagram developed by Viraj Joshi [4]

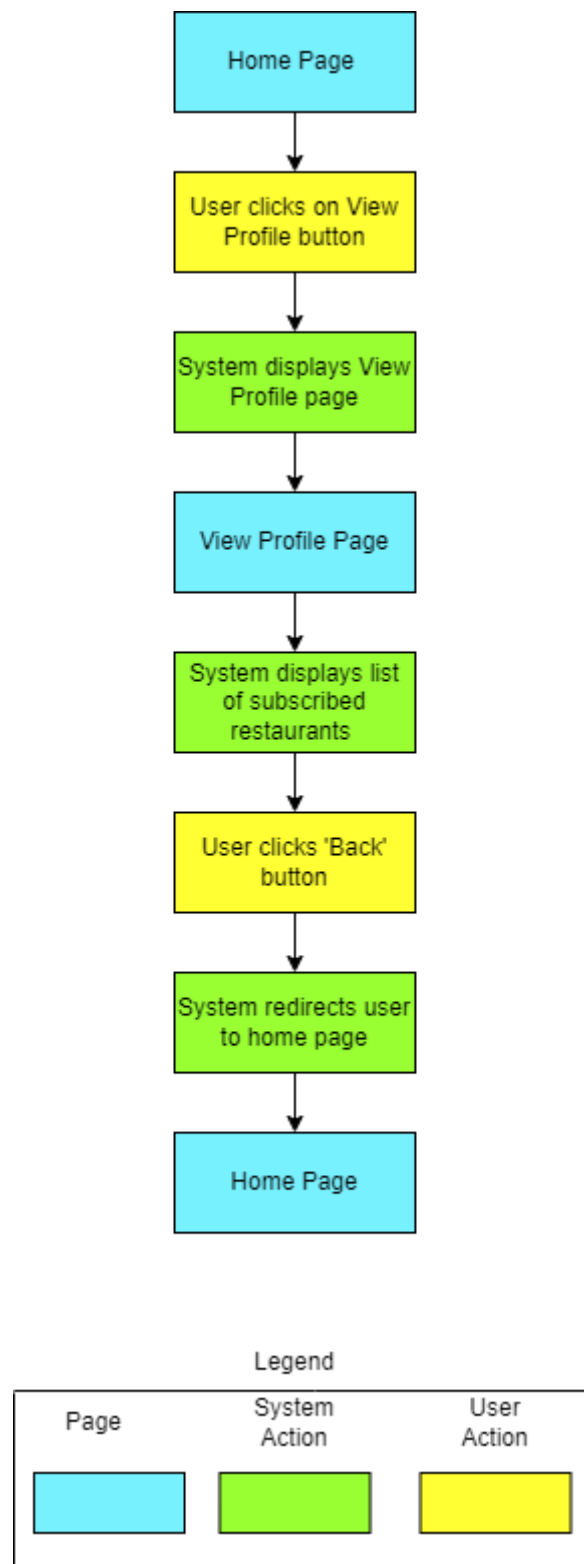


## b. Existing User Login



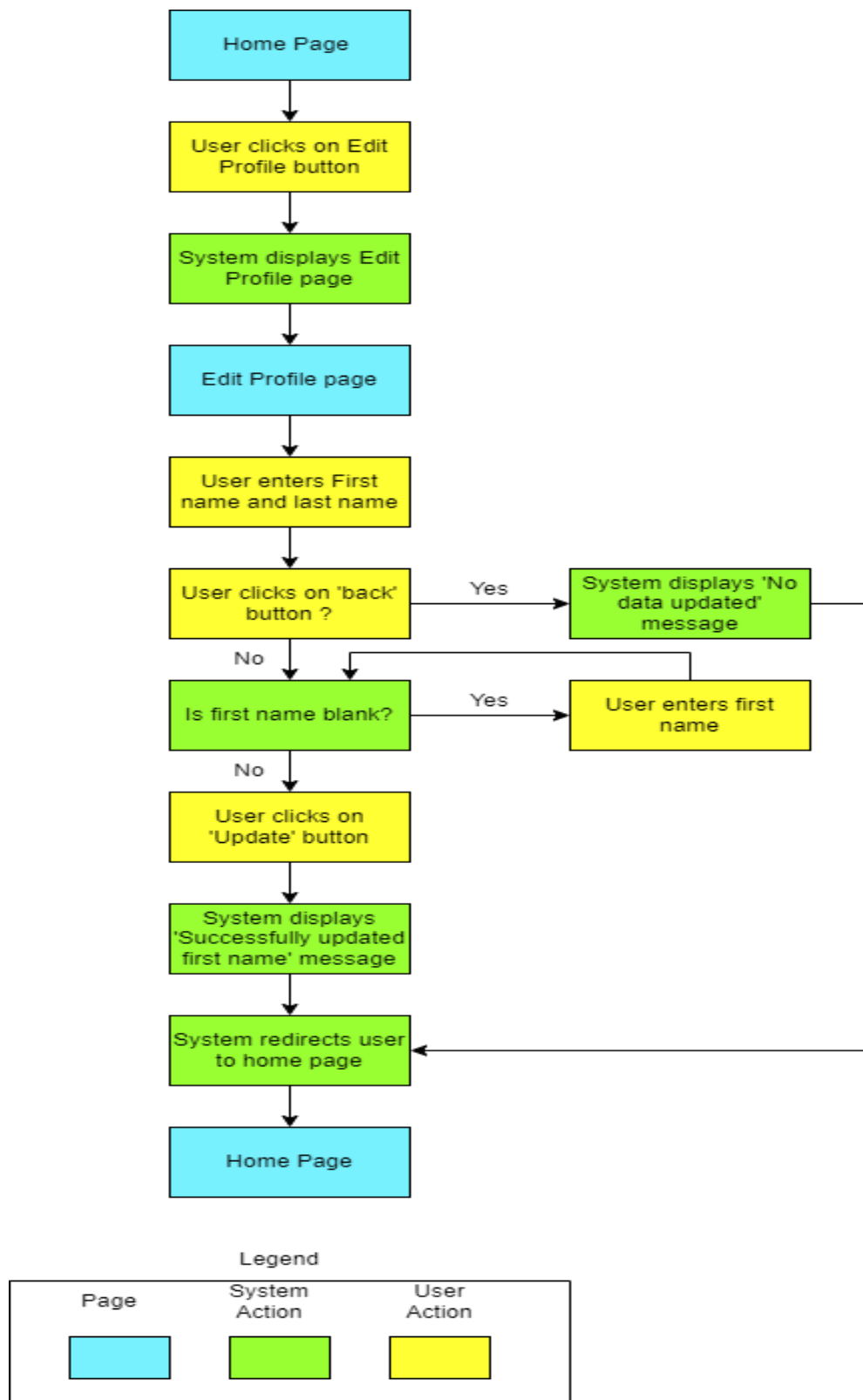
**Figure 7:** Task Flow Diagram for user login developed by Viraj Joshi [4]

c. View User Profile



**Figure 8:** Task Flow Diagram for View Profile developed by Viraj Joshi [4]

#### d. Edit User Profile



**Figure 9:** Task flow diagram to edit profile developed by Viraj Joshi [4]

## 2. Restaurant Profile Management

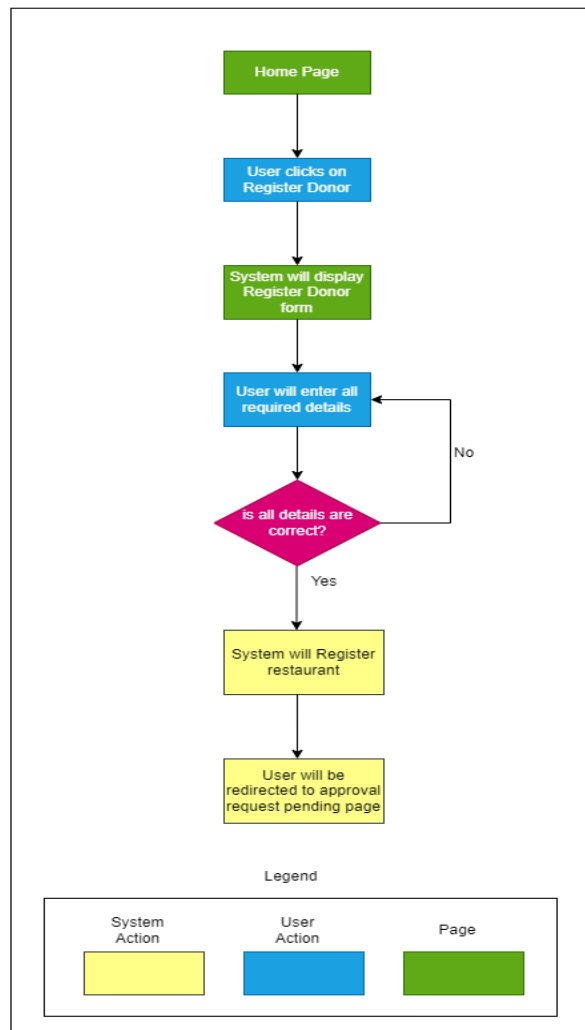
Feature Owner – Arpit ribadiya ([ar304626@dal.ca](mailto:ar304626@dal.ca))

Tasks –

- a. Restaurant Registration
- b. View Restaurant Profile
- c. Update Restaurant Profile
- d. Restaurant Login

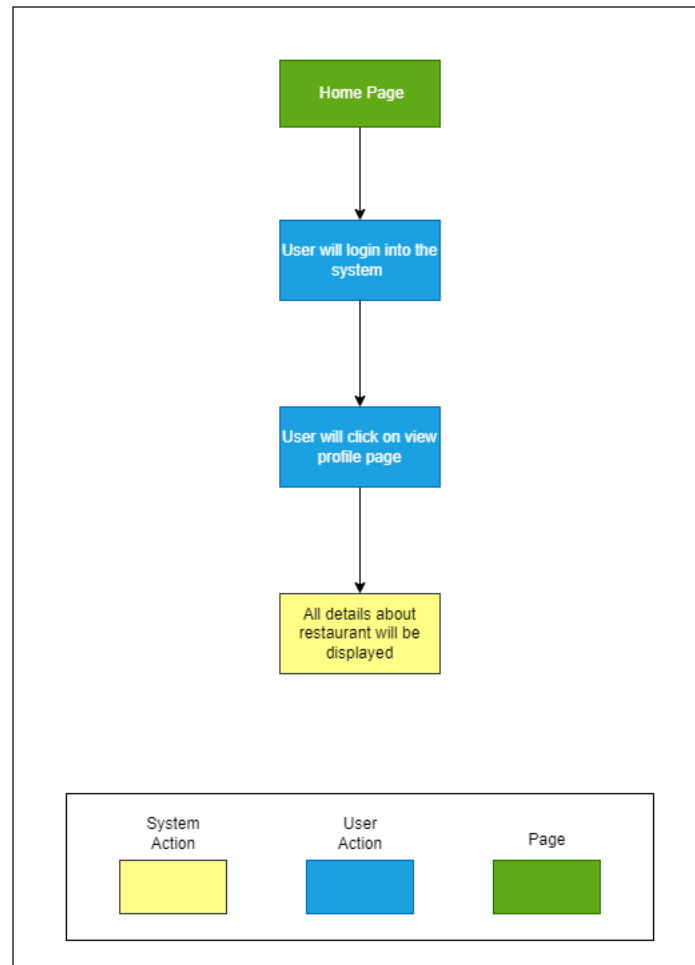
### Task flow diagrams

- a. Restaurant Registration



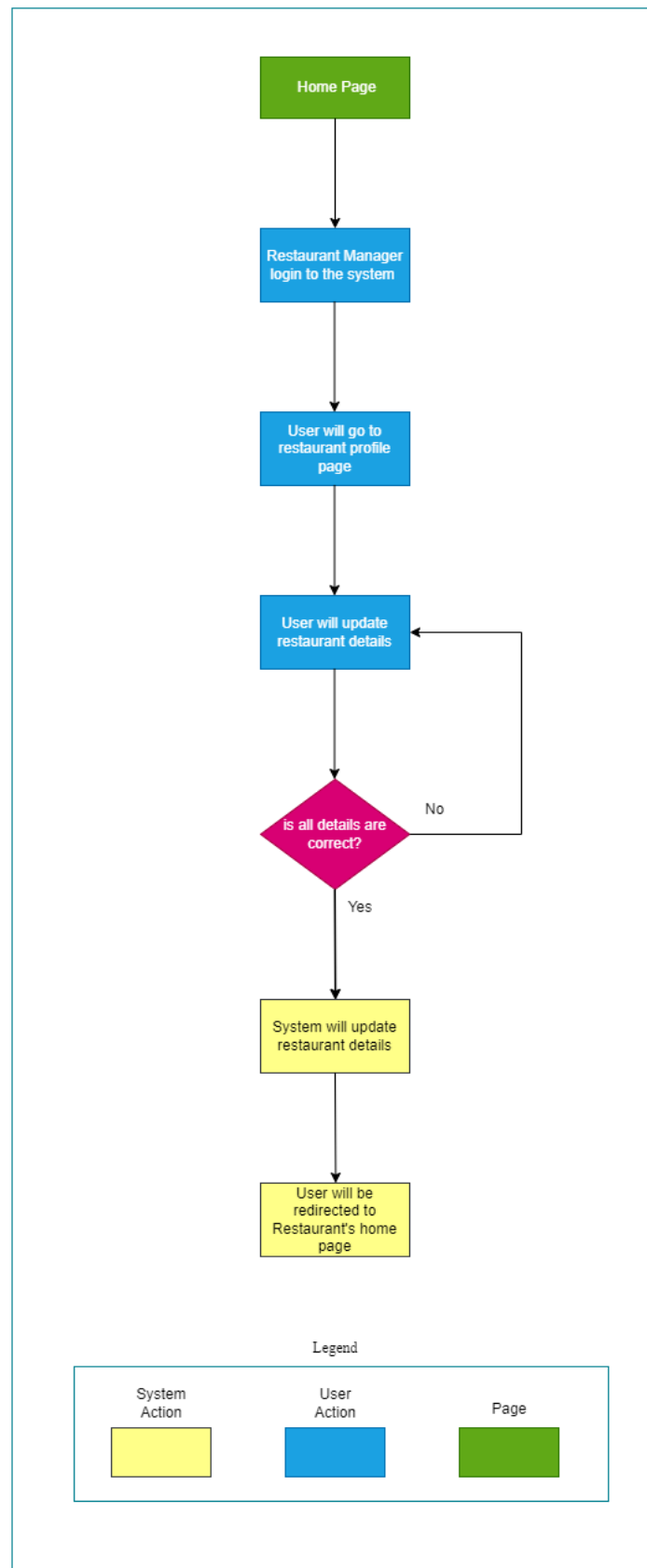
**Figure 10:** Taskflow diagram Restaurant registration developed by Arpit Ribadiya [4]

## b. View Restaurant Profile



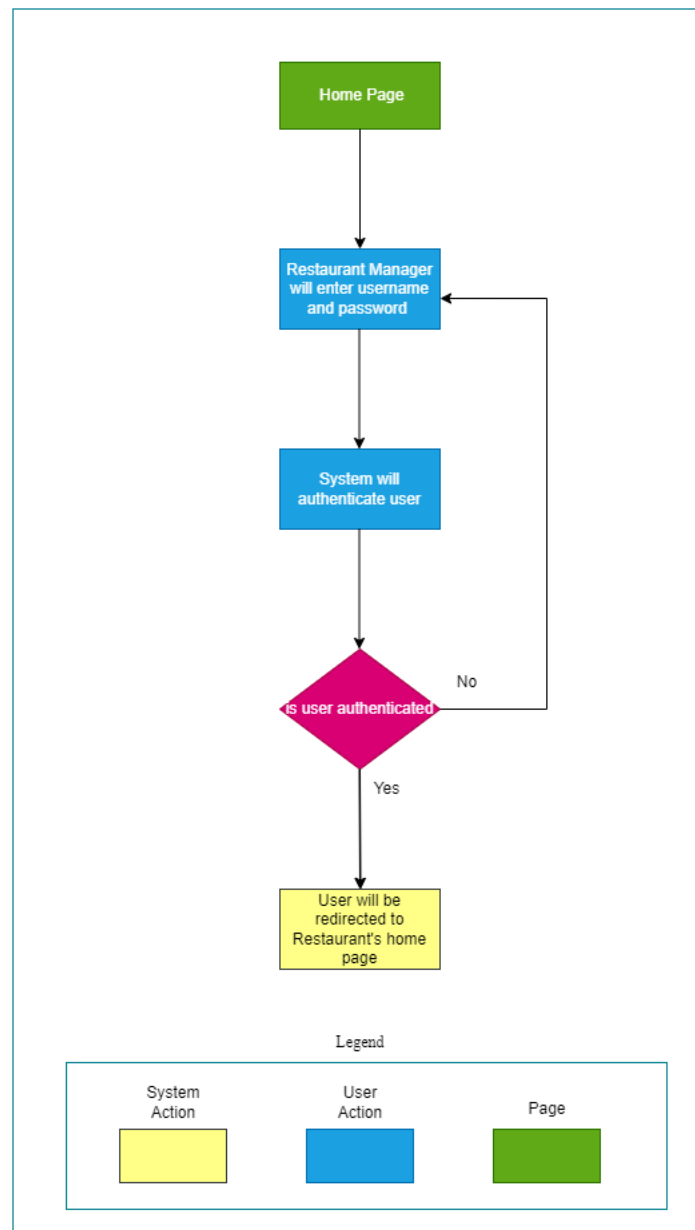
**Figure 11:** TaskFlow diagram, View restaurant profile developed by Arpit Ribadiya [4]

c. Update restaurant profile



**Figure 12:** Taskflow diagram for edit restaurant profile developed by Arpit Ribadiya [4]

#### d. Restaurant Login

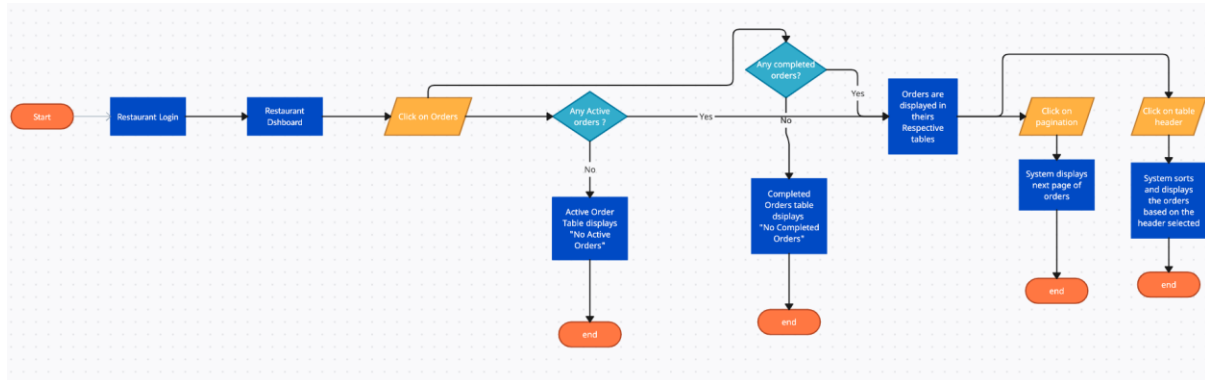


**Figure 13:** Taskflow diagram User Login developed by Arpit Ribadiya [4]

### 3. Restaurant order management

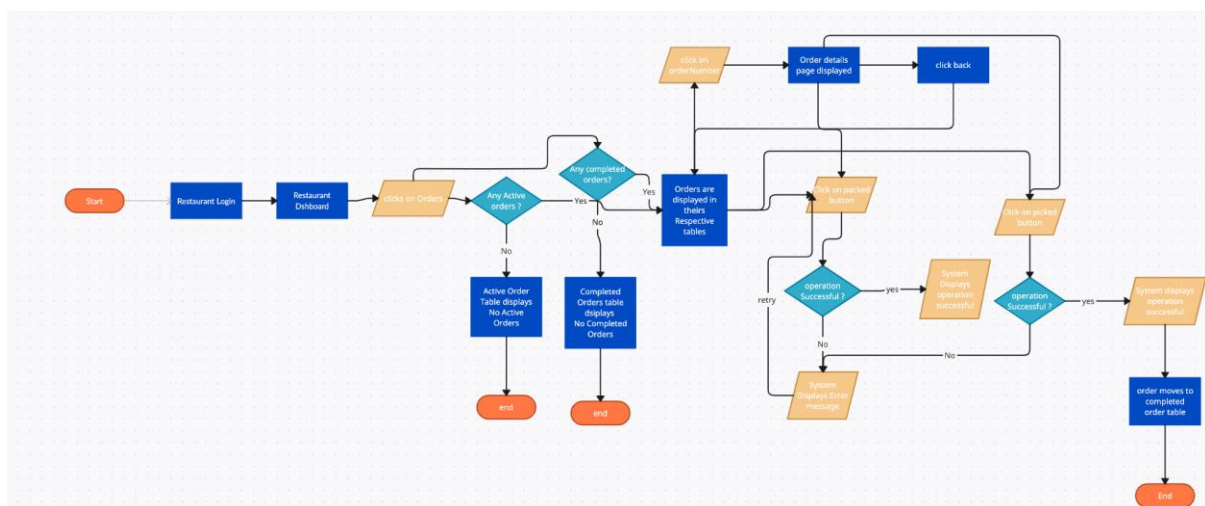
Feature Owner – Lav Patel ([lv842182@dal.ca](mailto:lv842182@dal.ca))

a) View orders by user, their timeslots, and items.



**Figure 14:** Taskflow diagram, View Orders developed by Lav Patel [5]

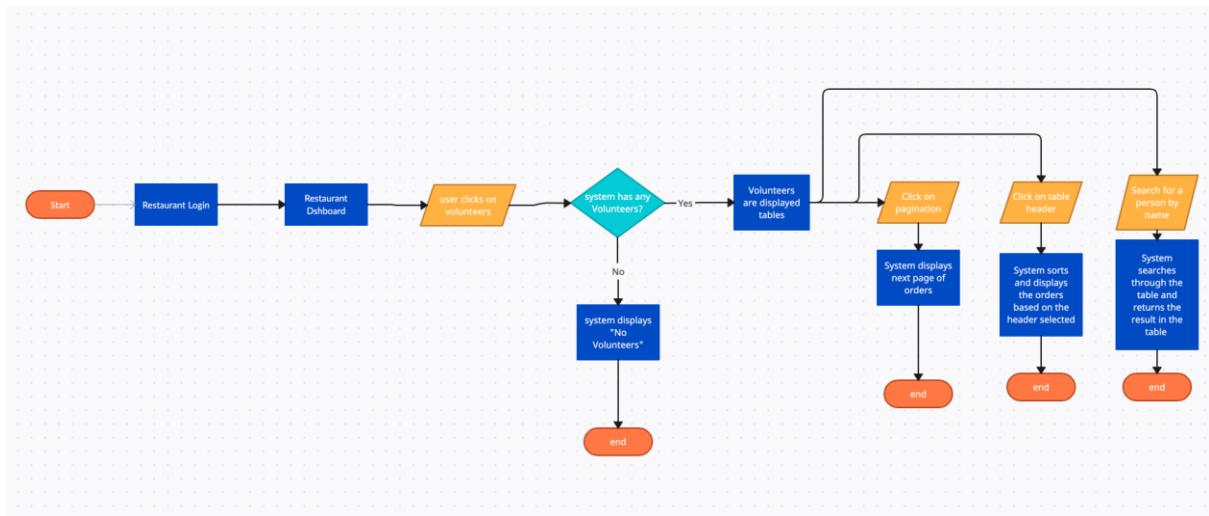
b) Change order status from pending to packed and picked.



**Figure 15:** Taskflow diagram, Change order status developed by Lav Patel [5]



c. Get available volunteer information based on availability.

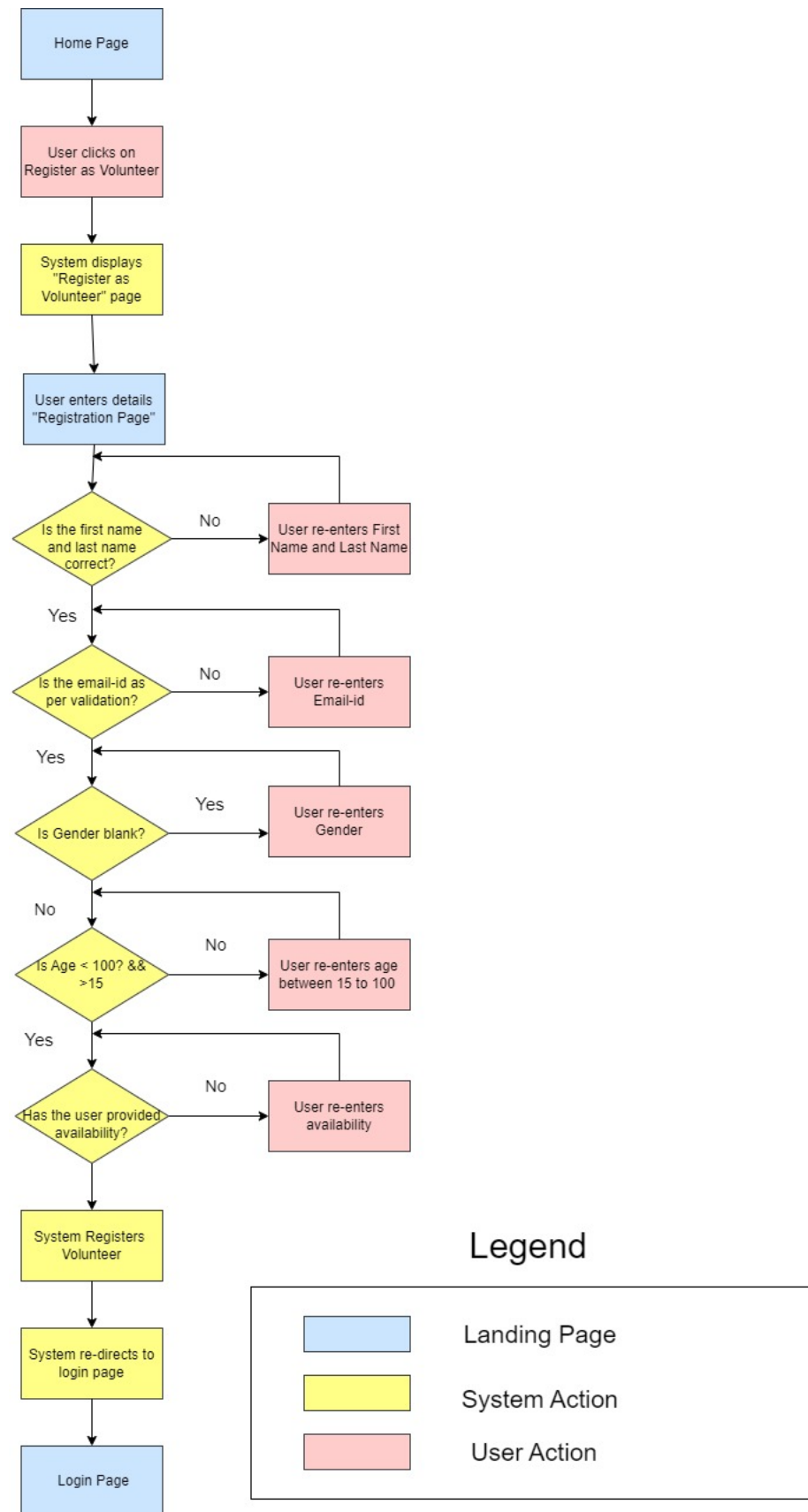


**Figure 16:** Task flow diagram Volunteer information developed by Lav Patel [5]

#### 4. Volunteer management

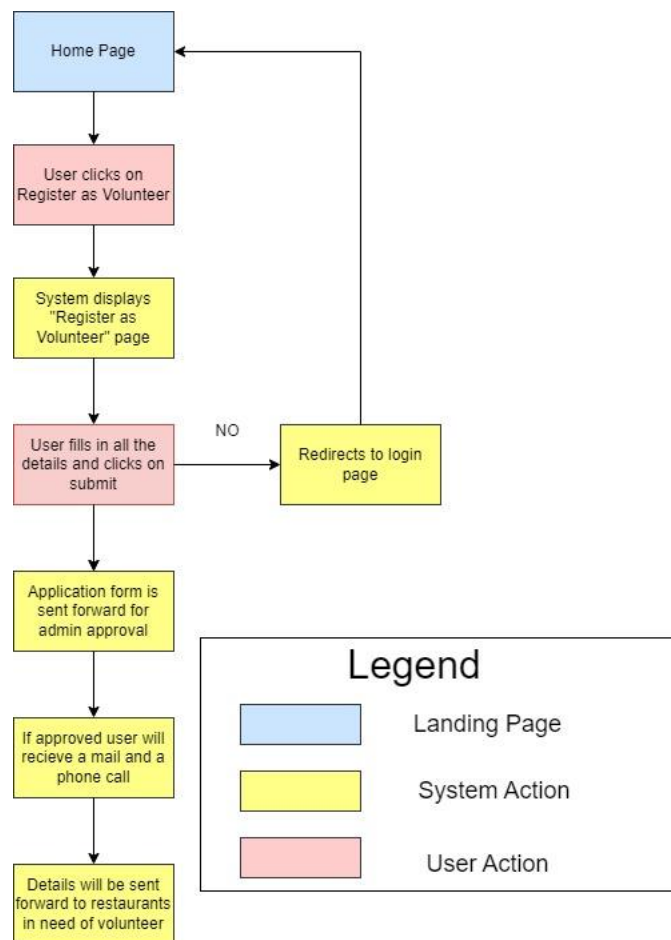
Feature Owner – Neha Karkhanis (nh601176@dal.ca)

a. Apply for volunteering with the application



**Figure 17:** Taskflow diagram for Volunteer registration developed by Neha Karkhanis [4]

b) Volunteer application approval management



**Figure 18:** Task flow diagram, Volunteer approval management developed by Neha Karkhanis [4]

## 5. Subscription management

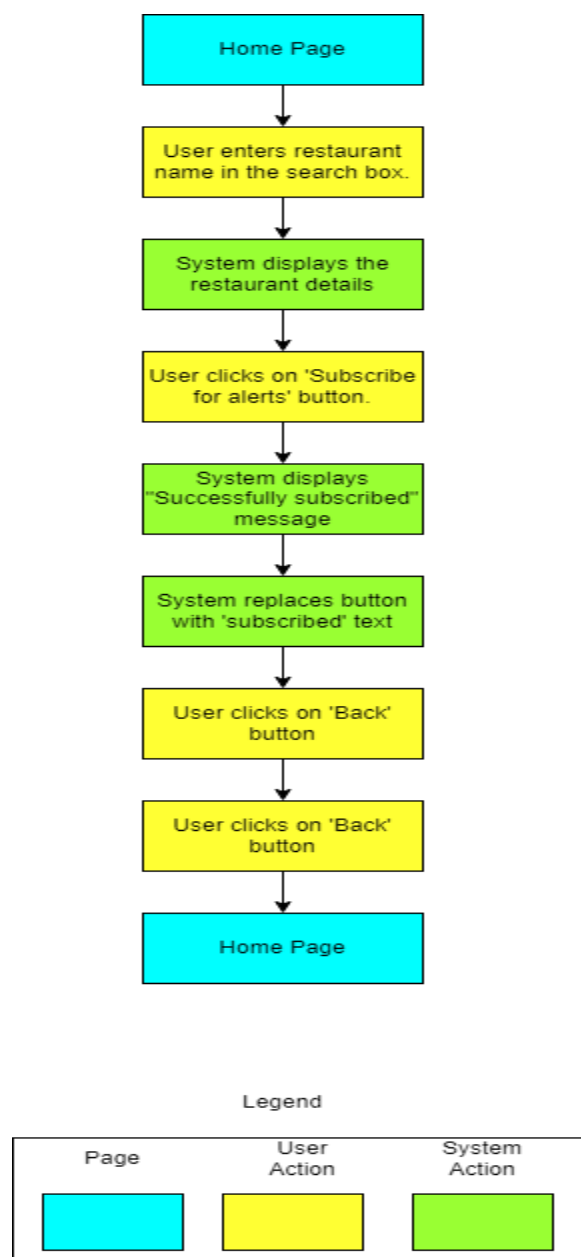
Feature Owner – Jay Kania ([jy440982@dal.ca](mailto:jy440982@dal.ca))

Tasks –

- a. Subscribe to posts from a restaurant
- b. Unsubscribe to a restaurant

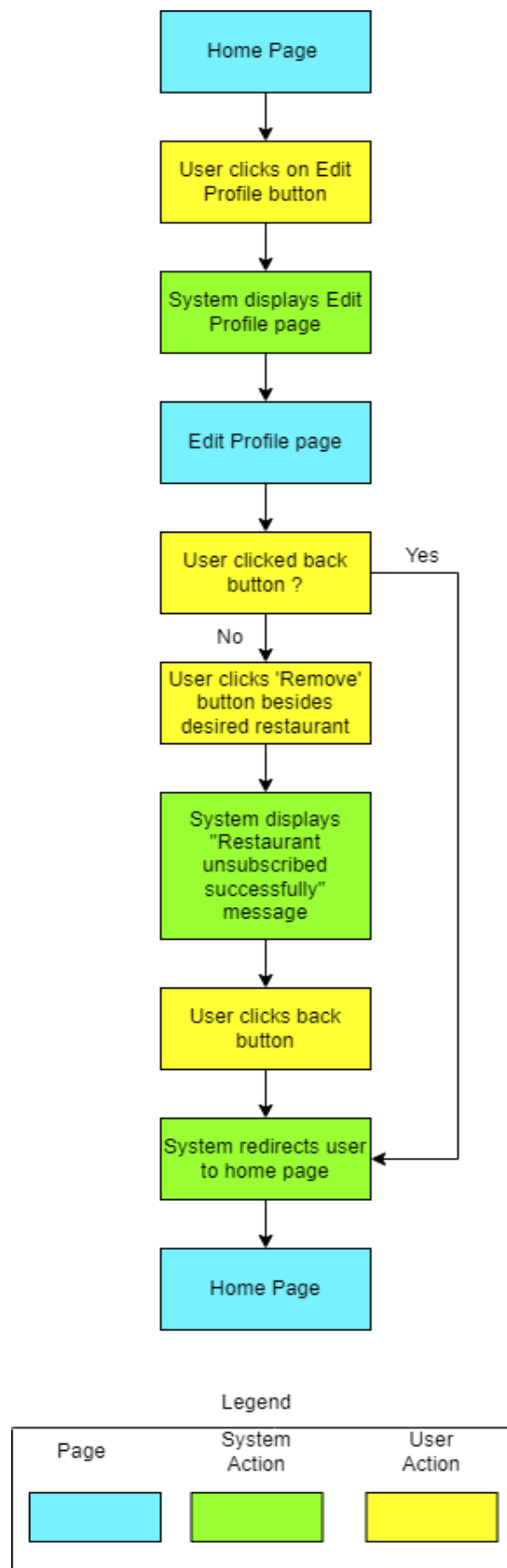
### Task flow diagrams

- a. Subscribe to posts from a restaurant



**Figure 19:** Task flow diagram to subscribe to restaurant developed by Jay Kania [4]

b. Unsubscribe to a restaurant



**Figure 20:** Task flow diagram to unsubscribe from restaurant developed by Jay Kania [4]

## A1.6. Lo-Fidelity Wireframes

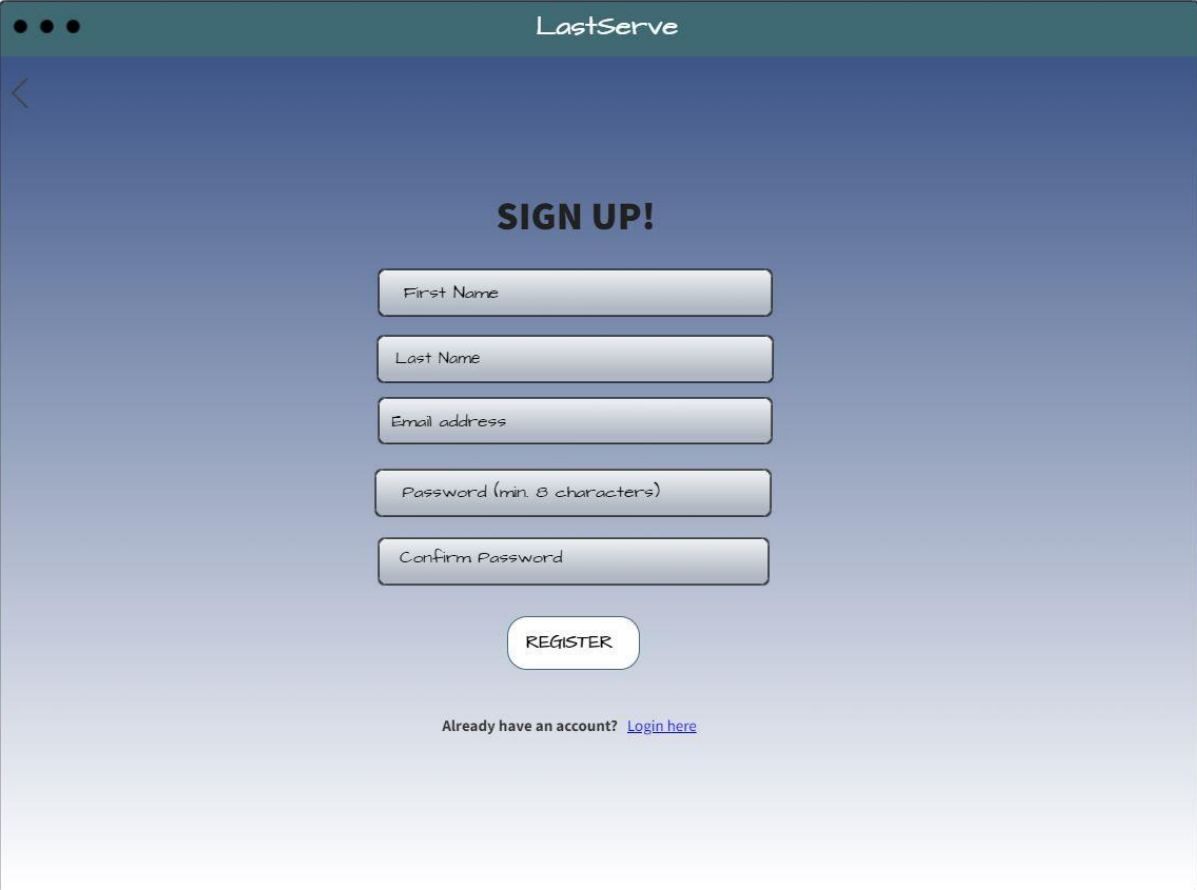
1. Wireframes for user signup task.

Feature : User Profile Management

Task : User signup

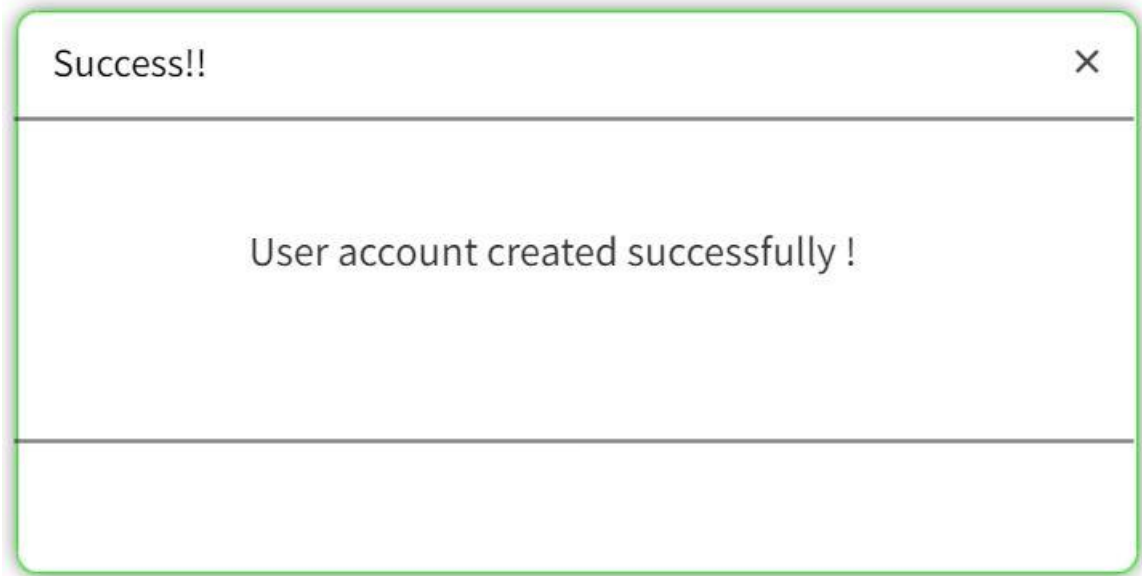
Owner : Viraj Joshi (vr862603@dal.ca)

Tool used: Mockflow [6]



The wireframe shows a web browser window titled "LastServe". Inside the browser, there is a back arrow icon in the top left corner. The main content area has a dark blue gradient background. At the top center, the text "SIGN UP!" is displayed in bold white capital letters. Below this, there are five stacked input fields with light gray backgrounds and rounded corners. The labels for these fields are "First Name", "Last Name", "Email address", "Password (min. 8 characters)", and "Confirm Password", all in a small, light gray font. Below the input fields is a white rounded rectangular button with the word "REGISTER" in black capital letters. At the bottom center, there is a line of text: "Already have an account? [Login here](#)", where "Login here" is a blue hyperlink.

**Figure 21:** Wireframe of sign-up page developed by Viraj Joshi.



**Figure 22:** Wireframe for successful account creation developed by Viraj Joshi

## 2. Wireframes

**Feature :** Restaurant Order Management

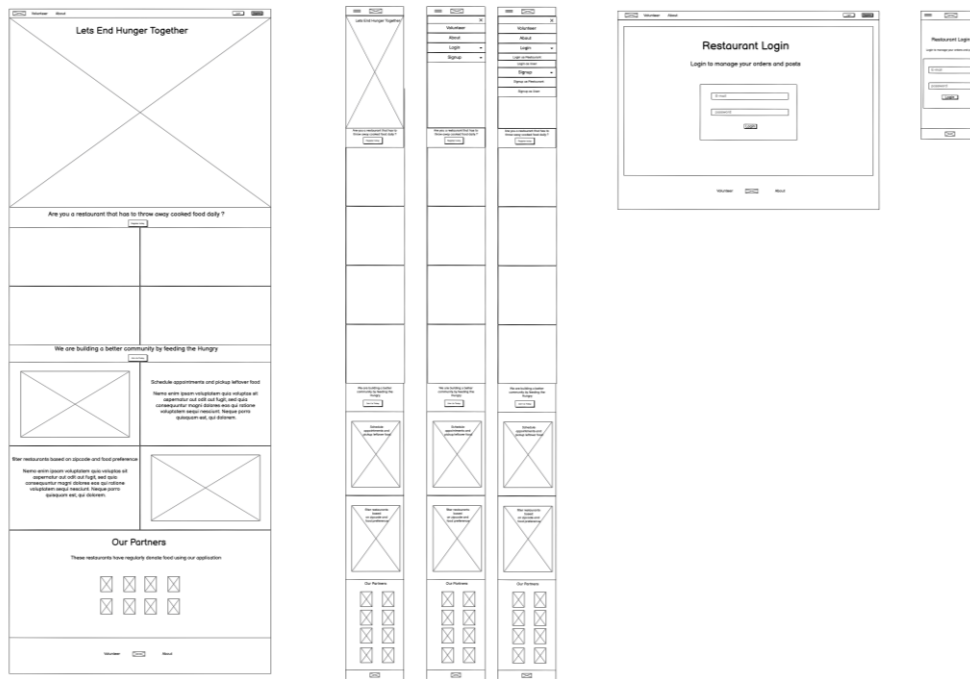
**Task :** Volunteer Availability information

**Owner :** Lav Patel ([lv842182@dal.ca](mailto:lv842182@dal.ca))

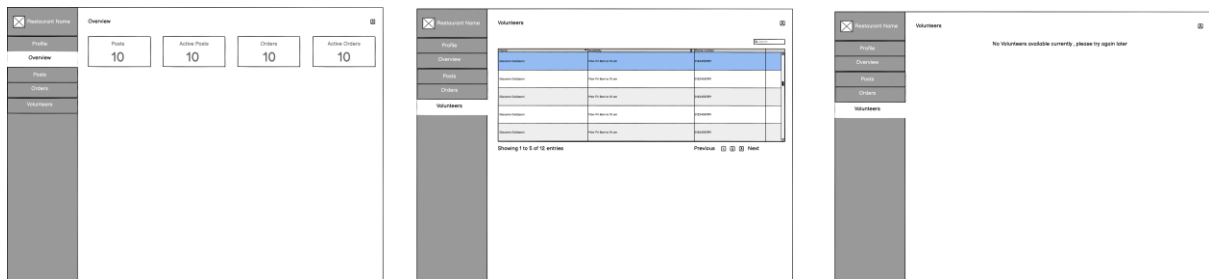
**Tool used :** Balsamiq [7]

**Icons used :** Balsamiq[7] and stickpng [8]

**inspiration for Table user interface :** datatables[9]



**Figure 23:** Wireframe for home page and login for task above developed by Lav Patel



**Figure 24:** Wireframes for Restaurant Dashboard post login for the task above developed by Lav Patel



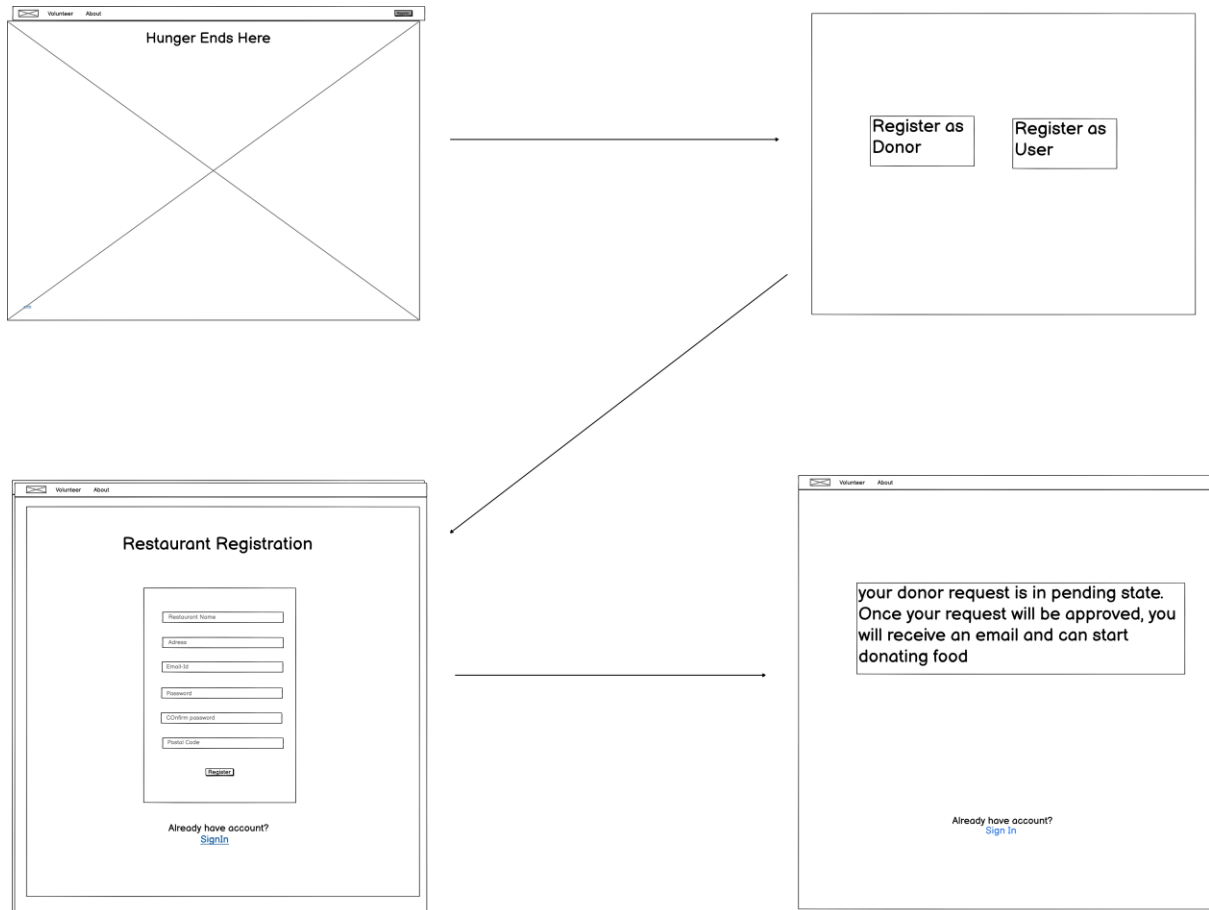
### 3. Wireframes for

Feature : Restaurant Profile Management

Task : Restaurant registration

Owner : Arpit ribadiya (ar304626@dal.ca)

Tool used : balsamiq[7]



**Figure 25:** Wireframes for the task above developed by Arpit Ribadiya

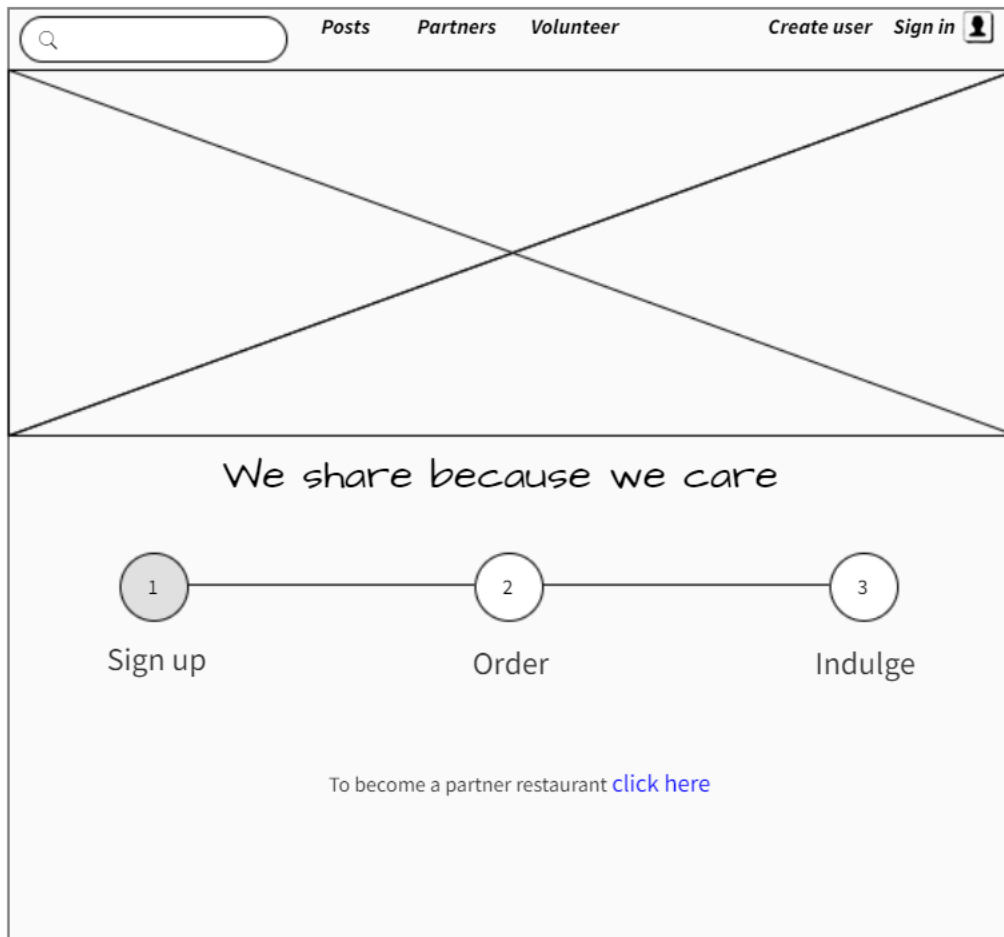
#### 4. Wireframes for

**Feature** : Volunteer application Management


**Task** : Apply for Volunteering with the application

Tool used: mockflow[6]

**Owner** : Neha karkhanis ([nh601176@dal.ca](mailto:nh601176@dal.ca))



**Figure 26:** Wireframe for home page for the task above developed by Neha Karkhanis

[Posts](#) [Partners](#) [Volunteer](#) [Create user](#) [Sign in](#) 

Without volunteers we'd be a nation without a soul

First Name

Last Name

Emai Id

Phone no.

Gender


☐ Male  
☐ Female  
☐ Other

Occupation

☐ Student  
☐ Working


Provide Availability

12 May 2016



SUBMIT

**Figure 27:** Wireframe for Volunteer registration form for the task above developed by Neha Karkhanis



SUCCESSFUL REGISTRATION

**Figure 28:** Wireframe for alert for the task above developed by Neha Karkhanis

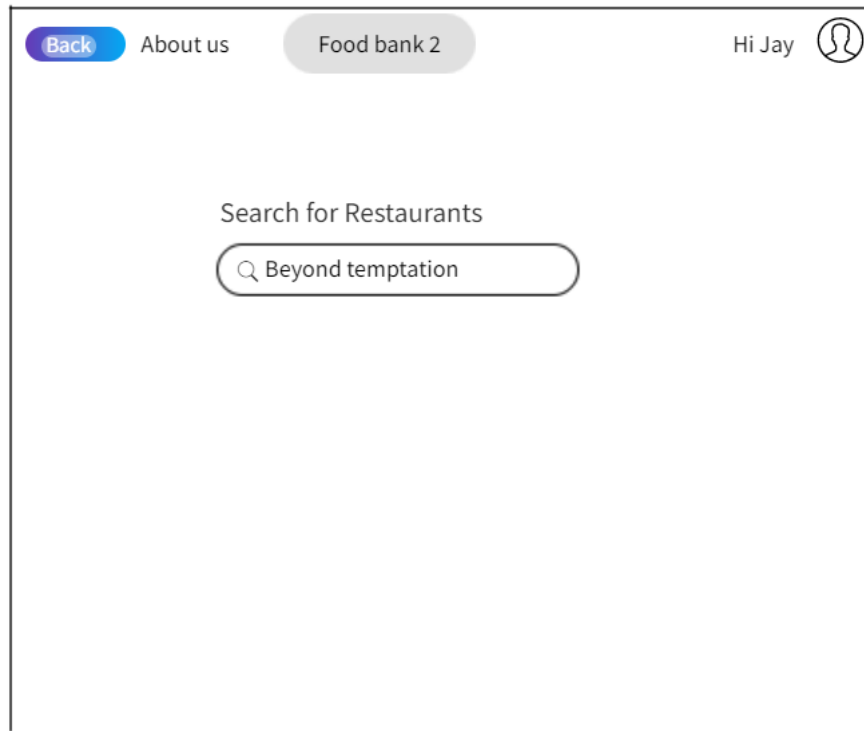
## 5. Wireframes for

Feature: Subscription Management

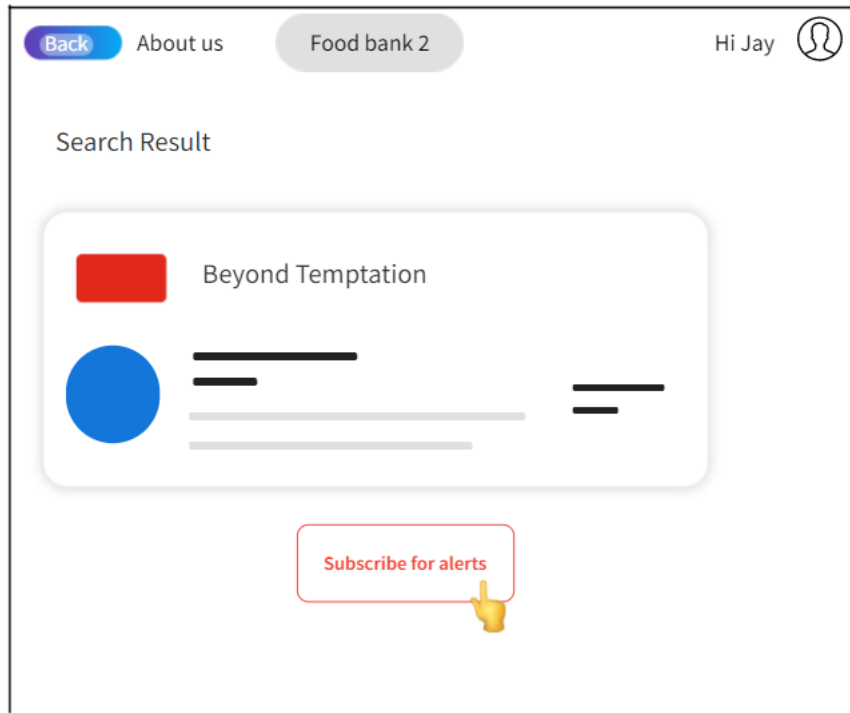
Task: Subscribe to a restaurant

Tool used: Mockflow [4]

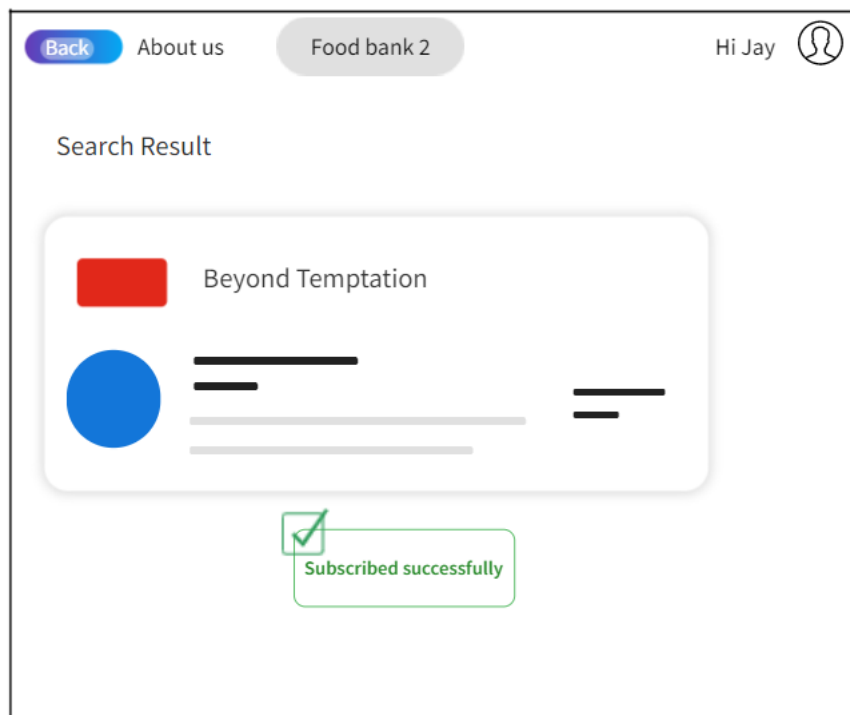
Owner: Jay Kania ([jy440982@dal.ca](mailto:jy440982@dal.ca))



**Figure 29:** Wireframe for searching a restaurant to be subscribed developed by Jay Kania



**Figure 29:** Wireframe for subscribing to a restaurant to be subscribed developed by Jay Kania



**Figure 30:** Wireframe for subscribed successfully message developed by Jay Kania.

## References

- [1] “WWF Sustainable Food Initiatives,” *WWF*. [Online]. Available: <https://www.worldwildlife.org/initiatives/food#:~:text=About%201.3%20billion%20tons%20of%20food%20are%20wasted,the%20same%20amount%20of%20land%20we%20use%20now>. [Accessed: 06-Feb-2023].
- [2] “Create powerful business content together with xtensio,” *Create powerful business content together with Xtensio*, 02-Oct-2022. [Online]. Available: <https://xtensio.com/>. [Accessed: 06-Feb-2023].
- [3] A. P. in S. Basics, A. Procter, Written By Alex Procter Alex left a corporate career to jump into all things web in 2011, Alex left a corporate career to jump into all things web in 2011, How to create a sitemap in 2019, Sitemaps Explained Simply: Sitemap Generator Shortcuts, and Making a Website: The Absolute Basics, “Site mapping guide: All your sitemap questions answered,” *WriteMaps*, 19-Nov-2019. [Online]. Available: <https://writemaps.com/blog/site-mapping-guide/>. [Accessed: 06-Feb-2023].
- [4] “Draw.io - free flowchart maker and diagrams online,” *Flowchart Maker & Online Diagram Software*. [Online]. Available: <https://app.diagrams.net/>. [Accessed: 06-Feb-2023].
- [5] Author Amanda Athuraliya Amanda Athuraliya is the communication specialist/content writer at Creately, Author, A. Athuraliya, Amanda Athuraliya is the communication specialist/content writer at Creately, and View all posts by Amanda Athuraliya →, “How to make a user flow diagram,” *Creately*, 09-Dec-2022. [Online]. Available: <https://creately.com/guides/user-flow-diagram/>. [Accessed: 06-Feb-2023].
- [6] Produte, “Wireframe tools, prototyping tools, UI mockups, UX suite, remote designing,” *MockFlow*. [Online]. Available: <https://mockflow.com/>. [Accessed: 06-Feb-2023].
- [7] “Balsamiq. rapid, effective and fun wireframing software: Balsamiq,” . *Rapid, Effective and Fun Wireframing Software*. [Online]. Available: <https://balsamiq.com/>. [Accessed: 06-Feb-2023].
- [8] “3 line menu icon transparent PNG - Stickpng.” [Online]. Available: <https://www.stickpng.com/img/icons-logos-emojis/menu-icons/3-line-menu-icon>. [Accessed: 07-Feb-2023].
- [9] “Add advanced interaction CONTROLSTO YOUR HTML tables the Free & Easy Way,” *DataTables*. [Online]. Available: <https://datatables.net/>. [Accessed: 06-Feb-2023].

