MOBILE APPLICATION

FOR

AQUAPONICS SALES

**Submitted by:**

Edoo Jameel Mohammad Ibraheem

Daren Devakumaren Baloomoody

Yogheswar Surjoo

Vibahakarsingh Beeharry

**MODULE NAME: GENIE LOGICIEL**

**Université des Mascareignes**

**Faculty of ICT**

**FEBRUARY 2024**

Table of Contents

[**1** **| INTRODUCTION** 3](#_Toc159929229)

[**1.1** **INTRODUCTION** 3](#_Toc159929230)

[**1.2** **PROBLEM STATEMENT** 4](#_Toc159929231)

[**1.3** **AIMS AND OBJECTIVES** 5](#_Toc159929232)

[**2** **| SYSTEM DEVELOPMENT** 6](#_Toc159929233)

[**2.1** **UML ANALYSIS AND MODELLING** 6](#_Toc159929234)

[**2.1.1** **Definition of Actors** 6](#_Toc159929235)

[**2.1.2** **Definition of Use Cases** 7](#_Toc159929236)

[**2.1.3** **Use Case Diagram** **Error! Bookmark not defined.**](#_Toc159929237)

[**2.1.4** **Detailed Description of Use Cases** **Error! Bookmark not defined.**](#_Toc159929238)

# **1.0 | INTRODUCTION**

# **INTRODUCTION**

Maurice a vu une augmentation significative du nombre de petits agriculteurs en aquaponie ces dernières années. L'aquaponie, une méthode de bio-agriculture durable qui combine l'aquaculture et l'hydroponie, a suscité l'intérêt de nombreux Mauriciens cherchant à adopter un meilleur mode de vie alimentaire.

Des études ont montré que Maurice est l'un des principaux pays d'Afrique en ce qui concerne l'utilisation de pesticides et d'engrais dans l'industrie agricole. L'aquaponie est souvent la solution privilégiée pour de nombreuses personnes en matière de bio-agriculture, car elle offre une approche holistique de l'agriculture grâce à la combinaison de l'aquaculture et de l'hydroponie pour créer un système autosuffisant qui ne dépend pas de sources externes telles que les engrais chimiques pour stimuler la croissance des plantes.

# **1.2 PROBLEM STATEMENT**

Tout d'abord, les pratiquants amateurs d'aquaponie sont confrontés à de nombreux défis lorsqu'il s'agit de mettre en place et de faire fonctionner un système d'aquaponie. Une des difficultés auxquelles beaucoup d'entre eux sont confrontés est qu'ils peuvent ne pas être en mesure de fournir un approvisionnement constant en produits aux revendeurs en raison de leur production à petite échelle. En conséquence, ils sont obligés de trouver des acheteurs du grand public pour vendre leurs produits.

Cela peut être une tâche difficile car les seules options disponibles sont la publicité payante, qui est coûteuse et peu rentable, et la publicité sur les réseaux sociaux, qui peut être une tâche fastidieuse pour beaucoup, en supposant qu'ils aient le temps d'y participer.

D'autre part, le nombre de personnes optant pour un choix alimentaire plus sain a augmenté au fil des ans en raison des préoccupations concernant l'utilisation élevée de pesticides dans l'industrie agricole à Maurice. Et l'un des principaux obstacles pour les membres du grand public à cet égard est la difficulté à localiser les commerçants qui vendent des produits biologiques dans leur quartier.

Sans oublier que, dans la plupart des cas, vérifier si ces produits sont effectivement biologiques et bien cultivés n'est pas possible pour les acheteurs de confirmer par eux-mêmes sans l'aide d'une organisation tierce.

# **1.3 AIMS AND OBJECTIVES**

L'objectif principal de ce projet est de créer une application mobile dans le but d'établir une connexion entre les pratiquants amateurs ou à petite échelle de l'aquaponie à travers Maurice et les membres du grand public à la recherche de vendeurs de produits biologiques dans leur région.

L'application mobile sera affiliée à une organisation dédiée à la promotion et à l'éducation soit de l'aquaponie soit de l'agriculture biologique en général à Maurice. La principale raison à cela est que cette organisation aura la responsabilité de permettre uniquement aux commerçants vendant des produits biologiques d'aquaponie d'accéder au côté marchand de l'application.

En ce qui concerne les problèmes rencontrés par les petits commerçants en aquaponie à Maurice, en établissant une connexion entre les commerçants et les acheteurs intéressés par leurs produits, qu'il s'agisse de producteurs à petite échelle ou d'amateurs, les praticiens de l'aquaponie n'auront plus à se soucier de trouver des clients par eux-mêmes.

En ce qui concerne les membres du public, trouver des commerçants vendant des produits certifiés biologiques, en particulier des produits d'aquaponie, sera beaucoup plus facile.

L'application mobile permettra aux commerçants d'afficher leurs produits ainsi que leur emplacement. Les membres du public pourront ensuite localiser les commerçants les plus proches d'eux ou trouver des commerçants en recherchant des produits ou des emplacements spécifiques directement.

# **2.0 | Analysi****s**

# **2.1 UML ANALYSIS AND MODELLING**

La phase de développement du système débutera par une analyse des exigences du système afin d'accomplir les tâches définies dans le système.

Cela sera réalisé en utilisant les principes du langage de modélisation unifié (UML) ; plus précisément, nous définirons les acteurs et les cas d'utilisation dans le système ainsi que des diagrammes UML pour fournir une représentation visuelle.

**2.2 Definition of Actors**

"Les acteurs peuvent être définis comme quelque chose qui interagit avec le système. Les acteurs peuvent être des utilisateurs humains, certaines applications internes ou peut-être certaines applications externes" (Waykar, 2015).

Les acteurs au sein du système peuvent être divisés en deux catégories, à savoir les acteurs principaux et les acteurs secondaires..

**2.3 Actors.**

Acteur primaire est celui qui propose des services a gens afin qu’ils puissent arriver à leur but.

Acteur secondaire est celui qui bénéfices de ces service-là. Dans le cas de notre projet l’acteur primaire est le gérant du business car ce lui qui va proposer ces services aux clients par conte notre acteur secondaire serons leur client car lui vont bénéfice des différents services qui seront à leur disposition.

L’acteur externe par conte est celui qui n’est pas notre client mais qui car même bénéfices notre service par l’aide d’un client secondaire.

**2.4 Les Cas d’utilisation et les Relation.**

On a deux types de relation entre cas d’utilisation:

* Include
* Extend

**• Include**

En terme simple, include signifie un cas spéciale qui est obligatoire dans les cas d’utilisation. Sans son exécution, le cas d’utilisation principale ne s’éxecute pas.

**• Exclude**

Exclude signifie un cas optionnel qui n’a pas d’influence sur les cas de base du cas d’utilisation.

**2.5 Definition of Use Cases.**

La définition d'un cas d'utilisation dans le cadre de ce projet est une description écrite d'une tâche ou action spécifique qui sera effectuée par les acteurs principaux du système au sein de l'application mobile.

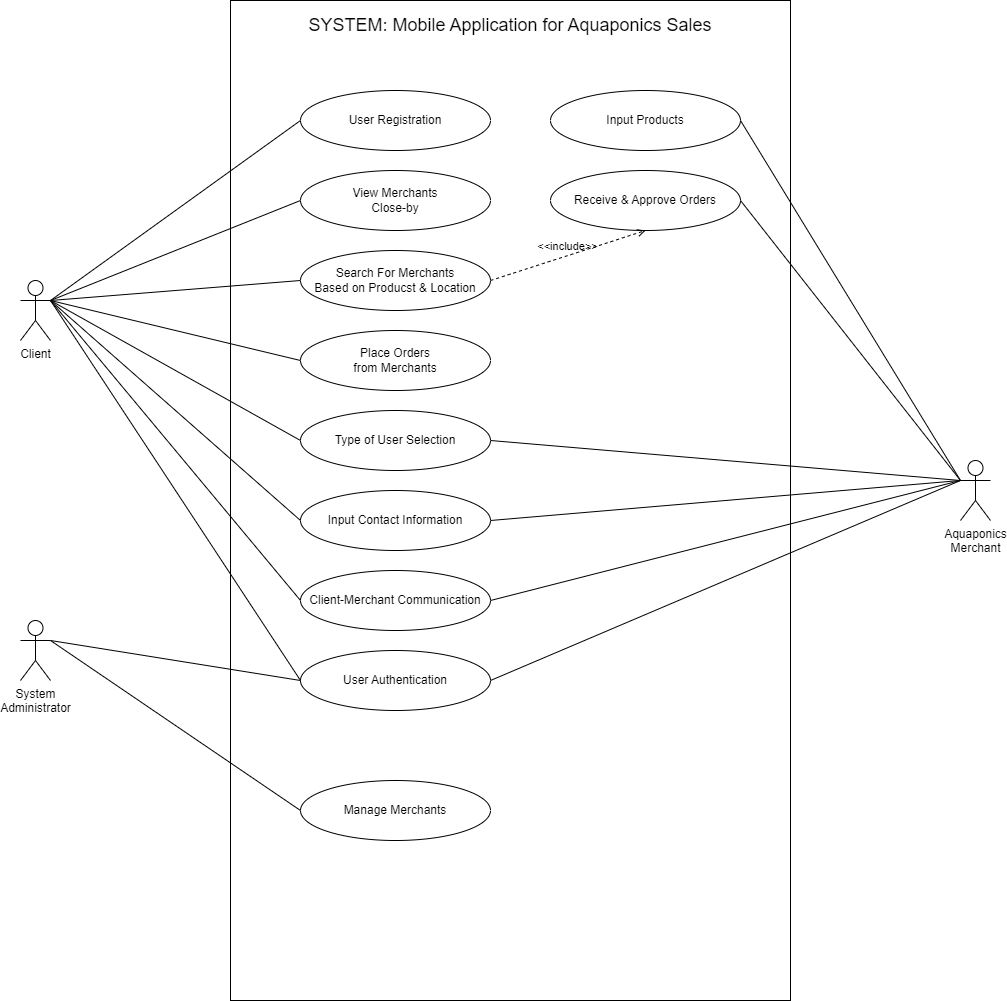
**La liste des cas d'utilisation à effectuer par les commerçants d'aquaponie pré-enregistrés est la suivante :**

* Sélection du type d'utilisateur (vendeur ou acheteur) - Sélectionner vendeur
* Authentification de l'utilisateur uniquement (connexion)
* Saisie de leurs coordonnées de contact, adresse et emplacement dans l'application
* Saisie de leurs produits et des informations connexes
* Recevoir et approuver les commandes passées par les clients ainsi que les notifications associées
* Recevoir/envoyer des messages directs des clients via l'application

**2.6 Use case diagramme.**

Le but d'un diagramme de cas d'utilisation est de produire une représentation visuelle du mécanisme opérationnel de l'application web et de ses cas d'utilisation en relation avec les acteurs du système.

Ce diagramme facilite le processus de développement car le développeur peut clairement voir les interactions entre les différentes parties du système.



**Figure 1: Use Case Diagram.**

**2.7 Diagramme de classe.**

Le but d'un diagramme de classe est de représenter la structure statique d'un système logiciel en mettant en évidence les classes, les attributs, les méthodes et les relations entre les classes. Il permet de visualiser la manière dont les différentes entités du système sont organisées et interagissent les unes avec les autres. Les class diagrammes sont utiles pour la conception et la modélisation des systèmes orientés objet, permettant aux développeurs de mieux comprendre l'architecture du système et de faciliter la communication entre les membres de l'équipe de développement.

A diagram of a database

Description automatically generated with medium confidence

**Figure 2: Class Diagram.**

**2.8 Description détaillée des cas d'utilisation.**

**Client.**

|  |
| --- |
| Case n°1 |
| **Use Case Name:** Place Order  **Actor:** Client  **Description** : Client browses, selects products, checks out, and confirms order.   * **Pre-conditions** : * Client is registered and logged into the system. * Merchant has an active account with products listed. * Client has chosen desired products from the merchant's listing. |

**Basic Flow:**

1. **Browse Products:**

* The client browses the merchant's product listing, searching by category, name, or using other available filters.
* The system displays product details, including images, descriptions, prices, and available quantities.

1. **Select Products:**

* The client selects the desired product(s) by clicking on "Add to Cart" buttons.
* The system adds the chosen product(s) to the client's virtual shopping cart, potentially displaying a confirmation message and updating the cart total.

1. **Review Cart:**

* The client accesses the shopping cart to review the selected items, quantities, and total cost.
* The system displays a summary of all products in the cart, including options to edit quantities or remove items.

1. **Choose Delivery/Pickup:**

* The client selects their preferred delivery method (e.g., home delivery, in-store pickup) if applicable.
* The system may display associated delivery fees or estimated delivery times based on the chosen method.

1. **Enter Payment Information:**

* The client enters their payment information securely (e.g., credit card details).
* The system employs secure payment processing to capture and verify the information.

1. **Place Order:**

* The client confirms the order details and clicks a "Place Order" button.
* The system transmits the order information to the merchant, including product details, delivery preferences, and client contact information.

1. **Order Confirmation:**

* The system displays a confirmation message to the client, including order details and an estimated delivery timeline.

**Alternative Flows:**

* **Insufficient product quantity:** The system informs the client and allows them to adjust the quantity or choose alternative options.
* **Payment processing failure:** The client is notified and given the option to retry payment or choose a different payment method.
* **Order cancellation:** The client can choose to cancel the order before the merchant confirms it. The system allows the client to remove items from the cart or cancel the entire order.

**Post-conditions:**

* An order is placed with the merchant.
* The client receives confirmation and order details.
* The merchant receives notification and order details.

|  |
| --- |
| Case n°2 |
| **Use Case Name:** User registration  **Actor:** Client  **Description** : Client browses, selects products, checks out, and confirms order.   * **Pre-conditions** : User is new to the system and has not previously registered. |

**Basic Flow:**

1. **Access Registration Page:**

The user accesses the registration page through the system's interface, often labeled "Sign Up" or "Create Account".

1. **Provide Personal Details:** The user enters their personal information, typically including:

**Required:**

* Name
* Id number
* Phone number
* Address

**Optional:**

* Email address
* Birthday

1. **Create Login Credentials:**

The user sets a username and password for future logins. The system may enforce password complexity requirements.

1. **Agree to Terms and Conditions:**

The user reviews and agrees to the system's terms of service and privacy policy.

1. **Submit Registration:**

The user clicks a "Submit" or "Register" button to send the information to the system.

1. **Verification (optional):**

Depending on the system, the user might need to verify their email address through a link sent to their inbox.

**Alternative Flows:**

* **Missing required information:** The system prompts the user to enter any missing mandatory details.
* **Invalid email address:** The system informs the user of the error and allows them to correct the email address.
* **Weak password:** The system enforces password complexity rules and prompts the user to create a stronger password.
* **Terms and conditions not accepted:** The user cannot complete registration without agreeing to the terms and conditions.

**Post-conditions:**

* Upon successful registration, the user receives a confirmation message and is either:
  + Logged in automatically.
  + Directed to a login page to use their newly created credentials.
* The user's information is stored securely in the system's database.

|  |
| --- |
| Case n°3 |
| **Use Case Name:** View Merchants Close-By  **Actor:** Client  **Description** : Client browses, selects products, checks out, and confirms order.  **Pre-conditions** :   * Client is registered and logged into the system. * Location services are enabled on the client's device (optional). |

**Basic Flow:**

1. **Request Location (optional):**

* If location services are enabled, the system retrieves the client's current location (latitude and longitude).
* Alternatively, the client can manually enter their desired location (e.g., zip code, city name).

1. **Search for Merchants:**

* The system uses the client's location (either retrieved or provided) to search for merchants within a specified radius (e.g., 5km).

1. **Display Results:**

* The system displays a list of nearby merchants, including:
  + - Merchant names and addresses.
    - Distance from the client's location.
    - Category or product type (optional).
    - Ratings and reviews (optional).

1. **View Merchant Details (optional):**

* The client can click on a specific merchant from the list to view detailed information, such as:
* Full address and contact information.
* Operating hours.
* Additional product details or images.

**Alternative Flows:**

* **Location services disabled:** The system prompts the client to enable location services or allows them to manually enter their location.
* **No merchants found:** Informs the client that no merchants are currently available within the specified radius.

**Post-conditions:**

* The client has access to a list of nearby merchants based on their location.
* They can choose to view individual merchants for further details.

**Additional Considerations:**

* This is a simplified example, and the actual use case might involve additional functionalities like filtering based on specific categories or sorting by distance.
* The use case diagram would also need to consider the actors involved (e.g., merchants updating their location information).

|  |
| --- |
| Case n°4 |
| **Use Case Name:** Search for Merchants based on Products & Location  **Actor:** Client  **Description** : Client searches products and location, system finds nearby stores.   * **Pre-conditions** : Client is registered and logged into the system. |

**Basic Flow:**

1. **Enter Search Criteria:**

* The client interacts with the search interface, potentially including:
* Entering keywords or selecting product categories.
* Specifying a location (e.g., zip code, city name) or using their current location (optional).
* Applying additional filters (optional), such as price range, ratings, or specific features.

1. **Submit Search:**

The client clicks a "Search" or "Find" button to submit their search criteria.

1. **Process Search Query:**

* The system searches the database based on the provided criteria:
* Matching product names, descriptions, or categories.
* Filtering by location (if provided) or using the client's current location (if enabled).
* Applying any additional filters selected by the client.

1. **Display Search Results:**

The system displays a list of merchants that meet the search criteria, including:

* Merchant names and addresses.
* Product information or images (relevant to the search query).
* Distance from the client's location (if applicable).
* Additional details like ratings or reviews (optional).

1. **Refine Search (optional):**

* The client can further refine their search by:
  + Modifying search keywords or filters.
  + Adjusting the location radius (if applicable).
* The system updates the results based on the revised criteria.

**Alternative Flows:**

* **No results found:**

The system informs the client that no merchants currently meet their search criteria and offers suggestions to broaden their search.

* **Location services disabled:**

The system prompts the client to enable location services for more accurate results.

**Post-conditions:**

* The client has access to a list of merchants based on their search criteria.
* They can choose to view individual merchants for further details.
* They can refine their search further to narrow down the results.

|  |
| --- |
| **Case n°5** |
| **Use Case Name:** Client-Merchant Communication  **Actor -Client:** Initiates communication with the merchant.  **Description** : Client browses, selects products, checks out, and confirms order.  **Pre-conditions** :   * Both client and merchant are registered and logged into the system. * The client has identified a specific merchant they want to communicate with. |

**Basic Flow:**

1. **Client Initiates Communication:**

* The client accesses a dedicated communication channel within the system for the chosen merchant. This could involve:
  + Sending a direct message.
  + Initiating a chat conversation.
  + Using a built-in messaging system within the platform.

1. **Client Sends Message:**

* The client composes their message, potentially including:
  + - Inquiries about products or services.
    - Order-related questions or requests.
    - General communication regarding their needs.

1. **System Sends Message:**

* The system securely transmits the client's message to the merchant through the designated communication channel.

1. **Merchant Receives Message:**

* The merchant receives a notification about the new message from the client.

1. **Merchant Reviews Message:**

* The merchant reviews the message content and understands the client's request.

1. **Merchant Sends Response:**

* The merchant composes a response to the client's message, potentially including:
* Answers to inquiries.
* Updates on order status or other relevant information.
* Proposals or clarifications regarding the client's needs.

1. **System Sends Response:**

* The system securely transmits the merchant's response back to the client.

1. **Client Receives Response:**

* The client receives a notification and can access the merchant's response through the designated communication channel.

**Alternative Flows:**

* **Merchant unavailable:**

If the merchant is offline or unavailable at the time, the client may be notified and offered options to leave a message or try again later.

* **Technical issues:**

In case of system errors or technical issues, both client and merchant might receive error messages or experience delays in communication.

**Post-conditions:**

* Client and merchant have established communication through the system's designated channels.
* They can continue exchanging messages to address inquiries, resolve issues, or facilitate transactions.

|  |
| --- |
| Case n°6 |
| **Use Case Name:** User Authentication  **Actor:** User (Client or Merchant)  **Description** : User enters credentials, system verifies and grants access or denies.  **Pre-conditions** : User is on the login page of the system. |

**Basic Flow:**

1. **Enter Credentials:**

* The user enters their login credentials, typically:
  + - Username or email address
    - Password

1. **Submit Login:**

* The user clicks a "Login" or "Sign In" button to submit their credentials.

1. **System Authenticates:**

* The system verifies the user's credentials against its database, checking:
  + - Username/email exists
    - Password matches the user's account

**Alternative Flows:**

1. **Invalid Credentials:**

* If the username or password is incorrect, the system:
  + - Displays an error message informing the user of invalid credentials.
    - Offers options to retry login or reset password.

1. **Account Locked (optional):**

* If the user enters incorrect credentials multiple times, the system might:
  + - Lock the account temporarily for security reasons.
    - Require additional verification (e.g., security questions) to unlock.

**Post-conditions:**

* **Successful Login:**

If authentication is successful, the system:

* + - Grants the user access to their account dashboard.
    - Sets a session cookie or token for further secure access within the system.
* **Unsuccessful Login:**

If authentication fails due to invalid credentials or account lock, the user remains on the login page and needs to address the issue before proceeding.

**Additional Considerations:**

* This use case assumes a single-factor authentication mechanism. Multi-factor authentication (MFA) might be implemented as an additional step, requiring further details in the diagram.
* The use case could be further extended to include functionalities like password reset, account recovery, and session management (e.g., logout, session timeout).

**System Administration.**

|  |
| --- |
| Case n°7 |
| **Use Case Name:** Manage Merchants (Admin)  **Actor:** System Administration  **Description:** User enters credentials, system verifies and grants access or denies.   * **Pre-conditions:** Administrator is logged in to the system. |

1. **Access Merchant Management:**

* The administrator navigates to a dedicated section within the system for managing merchants.

1. **View Merchant List:**

* The system displays a list of all registered merchants, potentially including:
  + - Merchant names.
    - Account statuses (active, pending, suspended).
    - Additional details (optional, e.g., category, location).

1. **Select Merchant:**

* The administrator chooses a specific merchant from the list to manage their account.

1. **Manage Merchant Details:**

* Based on the selected merchant, the system allows the administrator to perform various actions, potentially including:
  + - **View/Edit Merchant Information:** Access and modify contact details, business information, and other relevant data associated with the merchant's account.
    - **Approve/Reject Registration:** Review and decide on pending merchant registration requests based on specific criteria.
    - **Suspend/Activate Account:** Manage the active status of the merchant's account.
    - **View Performance Data (optional):** Access data related to the merchant's performance within the system (e.g., transaction history, customer reviews).

**Post-conditions:**

* The administrator has performed desired actions to manage the selected merchant's account.
* The system reflects the updated information or changes made by the administrator.

**Alternative Flows:**

* **Merchant not found:** Informs the administrator if the selected merchant doesn't exist or cannot be located.
* **Insufficient permissions:** If the administrator attempts actions beyond their designated permissions, the system prompts them with an access denied message.

**Aquaponics Merchants.**

|  |
| --- |
| Case n°8 |
| **Use Case Name:** Input Product  **Actor:** Aquaponies Merchant  **Description:** User enters credentials, system verifies and grants access or denies.  **Pre-conditions:**   * Aquaponics merchant is registered and logged in to the system. * Merchant has an active account with the system. |

**Basic Flow:**

1. **Access Product Input Interface:**

* The merchant navigates to a dedicated section within the system for adding or managing their products.

1. **Create New Product:**

* The system provides an interface for the merchant to enter new product details, potentially including:
  + - Product name and description.
    - Category selection (e.g., fish, plants, equipment).
    - Images or multimedia displaying the product (optional).
    - Price and availability information.
    - Stock level or inventory management options (optional).

1. **Submit Product Information:**

* The merchant reviews and submits the entered product details.

1. **System Processes Information:**

* The system validates the provided information and stores it in its database.

1. **Product Added (Optional Confirmation):**

* The system confirms successful product addition and displays the newly added product information.

**Alternative Flows:**

* **Missing required information:** The system prompts the merchant to enter any missing mandatory details.
* **Invalid information:** The system informs the merchant of any invalid data (e.g., incorrect pricing format) and allows them to correct it.
* **Image upload error (optional):** If image upload fails due to technical issues, the system informs the merchant and allows them to retry or continue without images.

**Post-conditions:**

* The Aquaponics merchant has successfully added a new product to their listings within the system.
* The product information is available for potential clients to view and browse.

**Additional Considerations:**

* This is a simplified example, and the actual use case might involve additional functionalities like:
  + Editing existing product information.
  + Managing product variations or options (e.g., different sizes, colors).
  + Setting up product promotions or discounts.
* The use case diagram could be further extended to show interactions with the system regarding managing product orders or responding to customer inquiries.

**Ordering system.**

|  |
| --- |
| Case n°9 |
| **Use Case Name:** Receive and Approve Orders  **Actor:** Aquaponics Merchant  **Description:** User enters credentials, system verifies and grants access or denies.  **Pre-conditions:**   * Aquaponics merchant is registered and logged in to the system. * Merchant has an active account with product listings. |

**Basic Flow:**

1. **Access Order Management:**

* The merchant navigates to a dedicated section within the system for managing orders.

1. **View Orders:**

* The system displays a list of orders placed by clients, potentially including:
  + - Order details (product names, quantities, prices).
    - Customer information (name, contact details).
    - Order status (pending, awaiting approval, processing, completed).

1. **Select Order:**

* The merchant chooses a specific order to review and process.

1. **Review Order Details:**

* The system displays detailed information about the selected order, including:
  + - Ordered products and quantities.
    - Customer information and delivery preferences.
    - Total order amount and any applicable fees.

1. **Approve Order (optional):**

* If necessary, the merchant approves the order, indicating they can fulfill it based on availability and stock.
* This step might be automatic depending on system functionalities.

1. **Process Order:**

* The merchant initiates the process to fulfill the order, potentially involving:
* Confirming product availability.
* Initiating packaging or preparing the order for delivery/pickup.
* Communicating with the client regarding any updates or estimated delivery timeline (optional).

1. **Update Order Status:**

* The merchant updates the order status within the system to reflect the progress, such as "processing" or "shipped."

1. **(Optional) Mark Order as Complete:**

* Once the order is fulfilled and delivered/picked up by the client, the merchant can mark it as "completed" within the system.

**Alternative Flows:**

* **Order cancellation request:** The client might request to cancel the order before it is shipped. The merchant can choose to accept the cancellation and update the order status accordingly.
* **Insufficient stock:** If the merchant discovers insufficient stock to fulfill the order, they can:
  + Contact the client to discuss options (e.g., partial order fulfillment, alternative products).
  + Cancel the order and inform the client.
* **Technical issues:** In case of system errors, the merchant might experience difficulty updating order status or accessing order details.

**Post-conditions:**

* The Aquaponics merchant has reviewed, processed, and updated the status of an order placed by a client.
* The system reflects the updated order status and potential communication with the client (optional).

**Additional Considerations:**

* This is a simplified example, and the actual use case might involve additional functionalities like:
* Managing order returns or refunds.
* Tracking shipments and providing delivery updates to clients.
* Communicating with clients regarding order-related issues or questions.

# **3.0 | Design And Conception.**

**3.1 Wireframes.**

Un wireframe est une représentation visuelle ou un schéma d'un site web, d'une application ou d'une interface logicielle. Il esquisse la structure de base et la disposition de l'interface utilisateur, en se concentrant sur l'agencement des éléments tels que les boutons, les menus, les champs de texte et les images, sans inclure les détails de conception tels que les couleurs, les polices ou les graphiques. Les wireframes sont souvent utilisés aux premières étapes du processus de conception pour communiquer des idées, établir la fonctionnalité et recueillir des commentaires des parties prenantes avant de passer à des conceptions de plus haute-fidélité. Ils aident les concepteurs et les développeurs à cartographier l'expérience utilisateur et à s'assurer que l'interface répond aux besoins de ses utilisateurs prévus.

A screenshot of a mobile phone

Description automatically generated

Figure 3 : Wireframes 1

A screenshot of a login form

Description automatically generated

Figure 4 : Wireframes 2

**3.2 Wireflow.**

Un Wireflow est une combinaison de wireframes et de flux de tâches dans une conception d'interface utilisateur. Cela signifie qu'il intègre à la fois la structure visuelle des wireframes et la séquence d'actions ou de tâches que les utilisateurs accomplissent à travers ces wireframes. En d'autres termes, un Wireflow combine la représentation visuelle de l'interface avec le flux narratif des interactions utilisateur, offrant ainsi une vue plus complète et détaillée de l'expérience utilisateur à travers différentes étapes.

En pratique, un Wireflow peut inclure des wireframes individuels pour chaque écran ou étape de l'interface utilisateur, ainsi que des flèches ou des lignes pour indiquer le flux de navigation ou les transitions entre ces étapes. Cela permet aux concepteurs de visualiser non seulement la disposition et l'apparence des éléments de l'interface, mais aussi la manière dont les utilisateurs interagissent avec ces éléments tout au long de leur parcours. Les Wireflow sont souvent utilisés pour planifier et optimiser les processus interactifs dans les applications et les sites web.

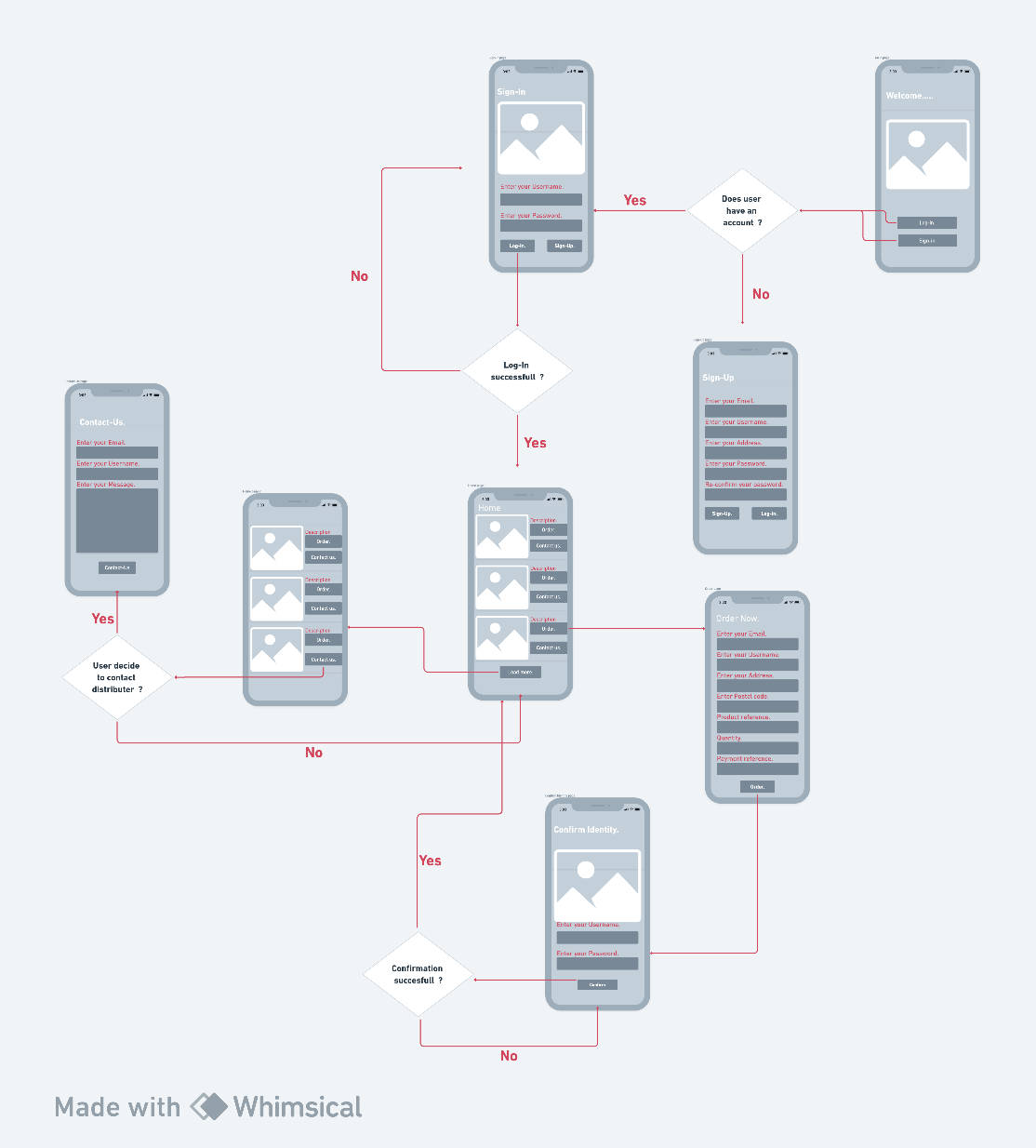


Figure 5 : Wireflow

**3.3 Prototype.**

Un prototype est une représentation fonctionnelle ou partiellement fonctionnelle d'un produit ou d'un système, généralement utilisé à des fins de test, de démonstration et de validation. Dans le contexte du développement de logiciels, un prototype peut prendre la forme d'une maquette interactive, d'une version simplifiée d'une application ou d'un site web, ou même d'un modèle physique dans certains cas.

L'objectif principal d'un prototype est de permettre aux concepteurs, développeurs et autres parties prenantes de visualiser et d'interagir avec une version concrète du produit afin de recueillir des commentaires, de repérer les problèmes potentiels et d'itérer rapidement sur les fonctionnalités et les aspects de conception. Les prototypes peuvent varier en termes de fidélité, allant des prototypes basse fidélité (simples croquis ou wireframes) aux prototypes haute-fidélité (plus proches de la version finale du produit, avec des interactions et des fonctionnalités avancées).

En résumé, un prototype est un outil crucial dans le processus de développement de produits, aidant à valider les concepts, à itérer sur les designs et à s'assurer que le produit final répond aux besoins et aux attentes des utilisateurs.

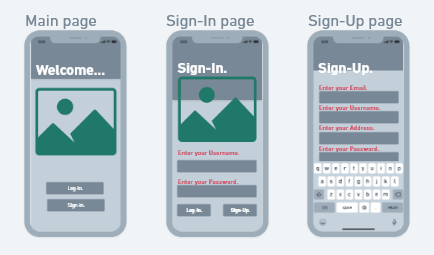


Figure 6 : Prototype 1



Figure 7 : Prototype 2

A screenshot of a mobile phone

Description automatically generated

Figure 8 : Prototype 3

**3.4 Database diagramme.**

**3.4 Diagramme séquence.**