VILNIUS UNIVERSITY
FACULTY OF MATHEMATICS AND INFORMATICS
INSTITUTE OF COMPUTER SCIENCE
INFORMATION TECHNOLOGIES STUDY PROGRAM

Problem-Based Project

Food Waste Reduction Platform

Done by:
Tadas Jokšas
Oleksii Morozov

Supervisor:
dr. Linas Butėnas

Vilnius
2025

Preface

This project was completed during the 4th semester of the *Information Technologies* study program, focusing on Innovative Studies. We chose the topic *ZeroWasteWagon: Reducing Food Waste reduction platform* from a list of suggestions in our "Problem-Based Project" course. The idea of using technology to tackle food waste intrigued us right away.

We were excited to explore how technology could connect consumers with surplus food from local venues to reduce waste. With guidance from our mentors, we worked hard to develop a practical and effective platform.

Teamwork and dedication were key to our project. Through research and collaboration, we overcame challenges and created a user-friendly system. The support from our teachers and the cooperation among team members were essential.

In the following pages, you will find what we learned and achieved. We hope our project showcases our skills and inspires new ideas in using technology for environmental sustainability.

June 23, 2025

Tadas Jokšas

Oleksii Morozov

Contents

Preface **2**

Abstract **4**

Santrauka **5**

Introduction **6**

1 Problem Analysis **7**
1.1 Analysis of the related projects . . . 7
1.2 Food Availability and Accessibility . . . 8
1.3 Platform Access Control and Partner Integration . . . 8
1.4 User Interaction and Geolocation Services . . . 8
1.5 Real-Time Data Handling and User Experience . . . 8
1.6 Payment Processing and Transaction Management . . . 9

2 Design **10**
2.1 System Architecture . . . 10
2.1.1 System workflow . . . 10
2.1.2 Use Case . . . 11
2.1.3 Database . . . 12
2.1.4 ER diagram . . . 12
2.1.5 Relational schema . . . 14
2.2 Functional requirement . . . 15
2.3 Non-functional requirements . . . 16
2.4 Wire frames . . . 17
2.4.1 Landing page . . . 17
2.4.2 User registration . . . 17
2.4.3 Partner registration . . . 18
2.4.4 User dashboard . . . 19
2.4.5 Partner working space . . . 20

3 Implementation **21**
3.1 Tools and Technologies . . . 21
3.2 User authentication . . . 21
3.3 Partner authentication . . . 23
3.4 Offers Creation System . . . 25
3.5 Map Integration . . . 25
3.6 Shopping Cart . . . 27
3.7 Payment System . . . 28
3.8 Screenshots of working system . . . 29
3.9 HCI practices used . . . 34

4 Conclusions and Future Work **34**

References **36**

Abstract

Zero Waste Wagon is a web-based platform designed to tackle food waste by connecting users with discounted surplus food from local businesses. Built on the Laravel framework, the platform offers secure user authentication and a user-friendly interface. Users can easily browse and filter available food offers based on their preferences, add items to their cart, and complete purchases seamlessly through Stripe integration. Partner businesses can efficiently manage their surplus food listings and access order history to optimize their operations. With features such as real-time updates, location-based services, and robust account management, ZeroWasteWagon provides a sustainable and convenient solution for both users and partner companies.

Santrauka

Maisto Atliekų Mažinimo Platforma

Zero Waste Wagon yra internetinė platforma, skirta kovoti su maisto švaistymo problema, jungiant vartotojus su vietinių įmonių siūlomu pertekliniu maistu už sumažintą kainą. Platforma, sukurta naudojant Laravel sistemą, siūlo saugią vartotojų autentifikaciją ir patogią naudotojo sąsają. Vartotojai gali lengvai naršyti ir filtruoti prieinamus maisto pasiūlymus pagal savo pageidavimus, pridėti prekes į krepšelį ir patogiai užbaigti pirkimus naudojantis Stripe mokėjimo integracija. Partneriai gali efektyviai valdyti perteklinio maisto sąrašus ir peržiūrėti užsakymų istoriją, kad optimizuotų savo veiklą. Su realaus laiko atnaujinimais, vietos nustatymo paslaugomis ir išsamia paskyros valdymo sistema, ZeroWasteWagon siūlo tvarią ir patogią sprendimą tiek vartotojams, tiek partnerių verslams.

Introduction

Reducing food losses and waste is essential in a world where the number of people affected by hunger has been slowly on the rise since 2014, and tons and tons of edible food are lost and/or wasted every day [10].

Globally, around 13.2 percent of food produced is lost between harvest and retail, while an estimated 19 percent of total global food production is wasted in households, in the food service, and in retail all together. Reducing food waste offers multiple benefits for people and the planet, contributing to improving food security, cutting pollution, saving money, reducing the pressures on nature and climate, and creating opportunities for the economy and society [12].

Despite the existence of numerous programs aimed at reducing food waste, participation is often dominated by large business chains, leaving small and medium-sized businesses with limited access to these initiatives. All these factors deprive opportunities to ensure food security and increase the environmental impact of food systems. As smaller venues are left out, they continue to face challenges in efficiently managing surplus food, contributing to an imbalance in the fight against food waste.

ZeroWasteWagon aims to address this problem by creating a platform where local venues can offer their surplus food to users at discounted prices. This helps reduce waste and provides affordable options for consumers. The core idea behind our platform revolves around connecting surplus food providers with organizations and individuals in need.

The platform is created using the Laravel web framework, which ensures the platform's performance and security for both users and partners. Customers can easily browse available food items through a range of filters, add items to their cart, and complete the checkout process efficiently. For business partners, managing inventory and orders is made simple, with easy-to-use tools that allow them to update listings, track orders, and modify venue details without hassle.

In addition to functionality, we have prioritized aesthetics and user experience by utilizing the Tailwind CSS framework. Tailwind's flexibility, combined with Human-Computer Interaction best practices, ensures that the platform is not only visually appealing but also intuitive and easy to navigate for users of all technical skill levels.

To further enhance convenience and security, ZeroWasteWagon integrates contactless payment options using Stripe, a trusted tool that facilitates fast, secure transactions. Whether consumers are purchasing from their phone or computer, they can feel confident that their personal and payment information is protected.

One of the standout features of ZeroWasteWagon is its integration of OpenStreetMap with AJAX technology, allowing users to explore participating establishments on an interactive map. This map-based search functionality is designed to give users a real-time view of their options, enabling them to filter venues by location and other criteria. As users scroll through the available listings, they can instantly see the exact location of each establishment on the map. After placing an order, users can go to their order history, where they will find a Google Maps link that directs them to the venue, providing additional convenience for planning their route.

1 Problem Analysis

1.1 Analysis of the related projects

ResQ club
ResQ Club is a Finnish platform designed to reduce food waste by allowing consumers to purchase surplus food from restaurants, cafes, and grocery stores. The service operates through location-based mobile and web applications, connecting users with nearby food that would otherwise go to waste.
Backend:

- ResQ Club uses APIs to facilitate communication between mobile and web platforms, as well as integration with partners. Its core system likely manages data related to inventory, user accounts, and orders.
- ZeroWasteWagon follows a similar approach in terms of managing food inventory and orders but focuses on offering partners flexibility in adding new food offers without needing predefined categories, which allows them to adapt to various types of food surplus.

Frontend:

- ResQ Club's platform supports both iOS and Android applications, developed using either native or cross-platform solutions. The web platform is designed for easy navigation, offering consumers a user-friendly experience when browsing and purchasing food offers.
- ZeroWasteWagon takes a similar approach by providing users with a clean interface to browse food offers and make reservations. However, at this point, ZeroWasteWagon's frontend is focused on web use, rather than native mobile apps.

ToGoodToGo
ToGoodToGo is another food waste reduction platform that operates in multiple countries. It allows consumers to buy surplus food at discounted rates, typically sold in "surprise bags." The platform works through mobile and web applications, but it has not yet launched in Lithuania.

Backend:

- ToGoodToGo likely handles business logic and stores transaction-related data through a structured database. It also uses geolocation APIs to help users find nearby food offers..
- ZeroWasteWagon similarly uses geolocation services to help users locate nearby venues offering surplus food. While ToGoodToGo focuses on larger geographical markets, ZeroWasteWagon is tailored to local businesses and emphasizes a simple system that allows partners to easily list and manage offers.

Frontend:

- ToGoodToGo's interface provides users with "surprise bags," where the exact contents of their purchase are unknown. It features a simple checkout process and a feedback system where users can rate their experiences with the purchase overall.

- In comparison, ZeroWasteWagon offers full transparency for items listed. Users know exactly what they are reserving, which provides a different purchasing experience compared to the "surprise bag" model. The platform's feedback system includes two distinct options: one for rating and describing the venue's service (e.g., how simple or fast the communication was) and another for rating the food received from the venue. This dual feedback system allows users to give more detailed reviews of their experience.

1.2 Food Availability and Accessibility

Food Waste Reduction Platform seeks to address the issue of food wastage by connecting consumers directly with sources of surplus food from local grocery stores and restaurants. Food waste primarily comes from overstocking and the perishability of items that do not sell before their expiry date. The platform enables businesses to list these items, providing detailed information including name, description, nutritional content, and price, alongside images of the food. The platform also facilitates real-time updates to ensure only currently available items are listed, minimizing the risk of advertising sold-out or expired items. This system improves food accessibility and reduces waste by redirecting surplus from potential disposal to consumption. [7]

1.3 Platform Access Control and Partner Integration

One of the key challenges in managing a platform that connects businesses with consumers is ensuring that only legitimate and trustworthy partners can list food offers. To address this, each business partner must register using an invitation code, which is verified and securely stored in a remote database. This verification process is crucial in authenticating partners and providing them access to the platform, where they can create and manage their food offers. Partners are able to input important details, such as dietary restrictions and caloric content, which are critical for consumers when making informed decisions. By implementing this controlled access, the platform prevents unauthorized or illegitimate businesses from posting offers, ensuring a reliable and secure environment for both consumers and partners. [5]

1.4 User Interaction and Geolocation Services

The challenge of connecting users with nearby food offers in real-time is crucial to the success of the platform. To address this, the platform utilizes the OpenStreetMap API for mapping and the HTML Geolocation API to accurately identify users' locations. This geolocation functionality is essential for real-time, location-based searches, allowing users to quickly find offers within a feasible distance. The interactive maps display business locations offering surplus food, making it especially effective in urban areas with a high concentration of options. By visualizing these locations geographically, users can better plan their visits, improving the likelihood of them taking up available offers and ultimately reducing food waste.

1.5 Real-Time Data Handling and User Experience

Real-time updates are crucial for a platform dealing with time-sensitive items like food [7]. The integration of real-time data handling ensures that the information displayed (e.g., availability of offers) is always current, preventing user disappointment and logistical issues. The platform also

features a user-friendly interface with functionalities like search, filter, and sort to enhance user experience. These tools are designed to work efficiently even during high traffic, supported by robust backend technologies that handle data requests and updates seamlessly.

1.6 Payment Processing and Transaction Management

To facilitate transactions, the platform integrates with Stripe API, allowing secure and reliable payment processing. This system is crucial for ensuring that financial transactions are handled smoothly and without errors, providing a trustworthy environment for users to make purchases. The Stripe API also supports the platform's feature of reserving selected items for a limited duration (10 minutes), during which users can complete their purchases. This mechanism helps manage the inventory effectively, ensuring that items are not held indefinitely without purchase.[6]

2 Design

2.1 System Architecture

2.1.1 System workflow

This MVC architecture ensures a clear separation, making the ZeroWasteWagon application easy to manage and extend. Each component handles a distinct aspect of the application, from user interactions and interface rendering to business logic and data management. This structure facilitates efficient development, testing, and maintenance of the application

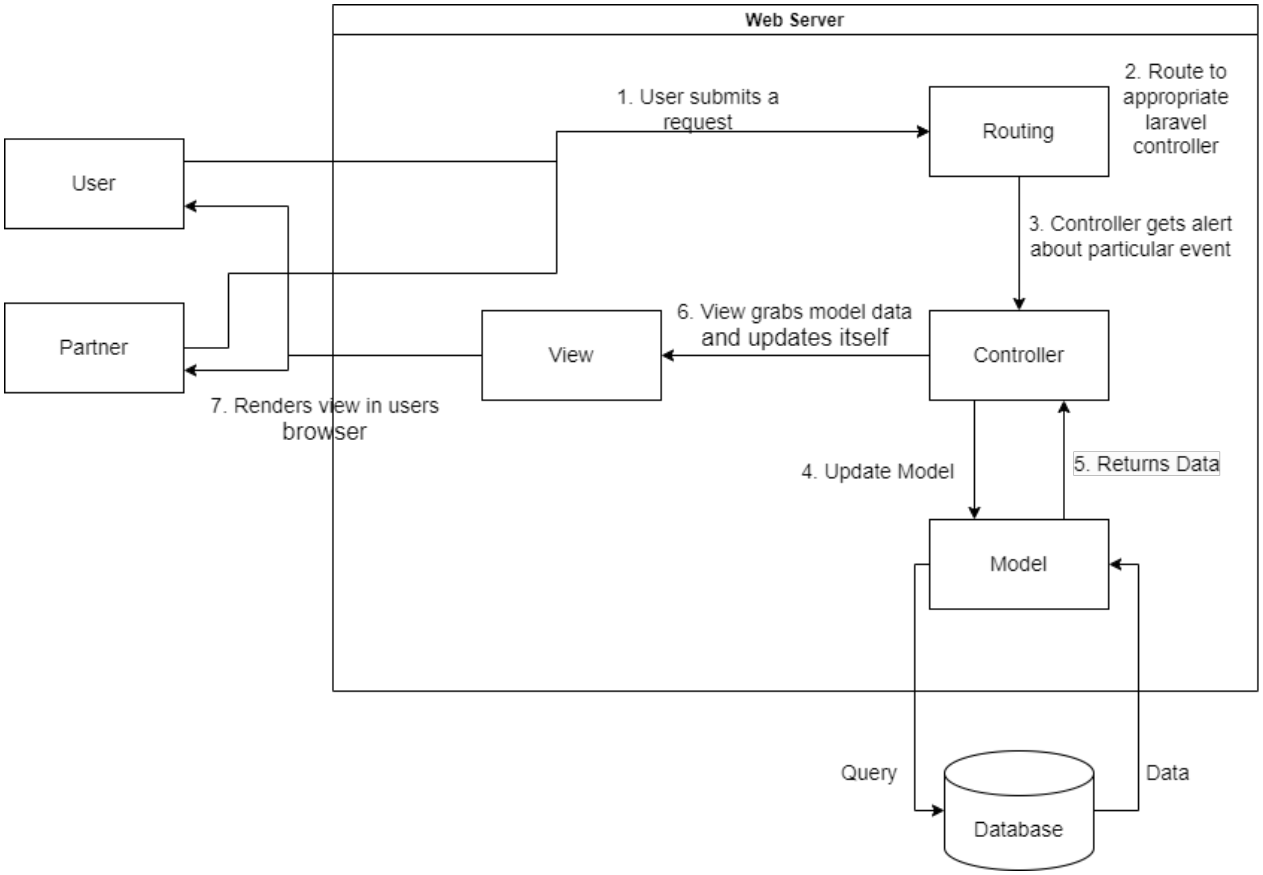

Figure 1. Model-View-Controller system

2.1.2 Use Case

The use case diagram provides a high-level overview of the functionality of the system.

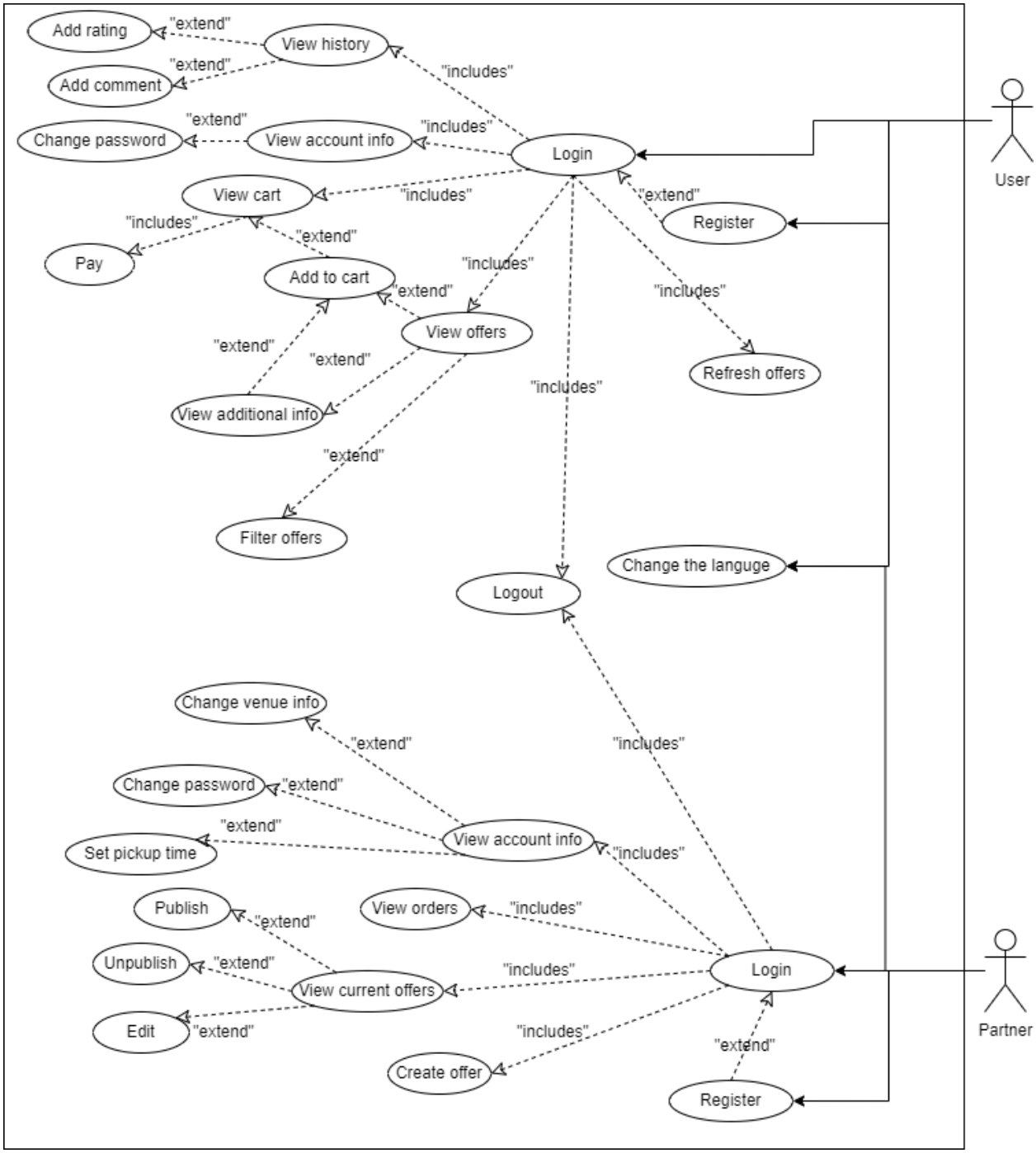

Figure 2. Use case diagram

The system provides two types of users:

- **User**: Individuals or organizations for one reason or another seeking free or low-cost food.
- **Partner**: Individuals, restaurants, or businesses with either surplus edible food they cannot consume or sell, or food that's about to expire.

2.1.3 Database

MySQL is used as the database in this project because it is a versatile and powerful database management system that offers a range of benefits, including open-source availability, scalability, performance, reliability, and security. The database is stored on a separate VM and is directly connected to the web server.

2.1.4 ER diagram

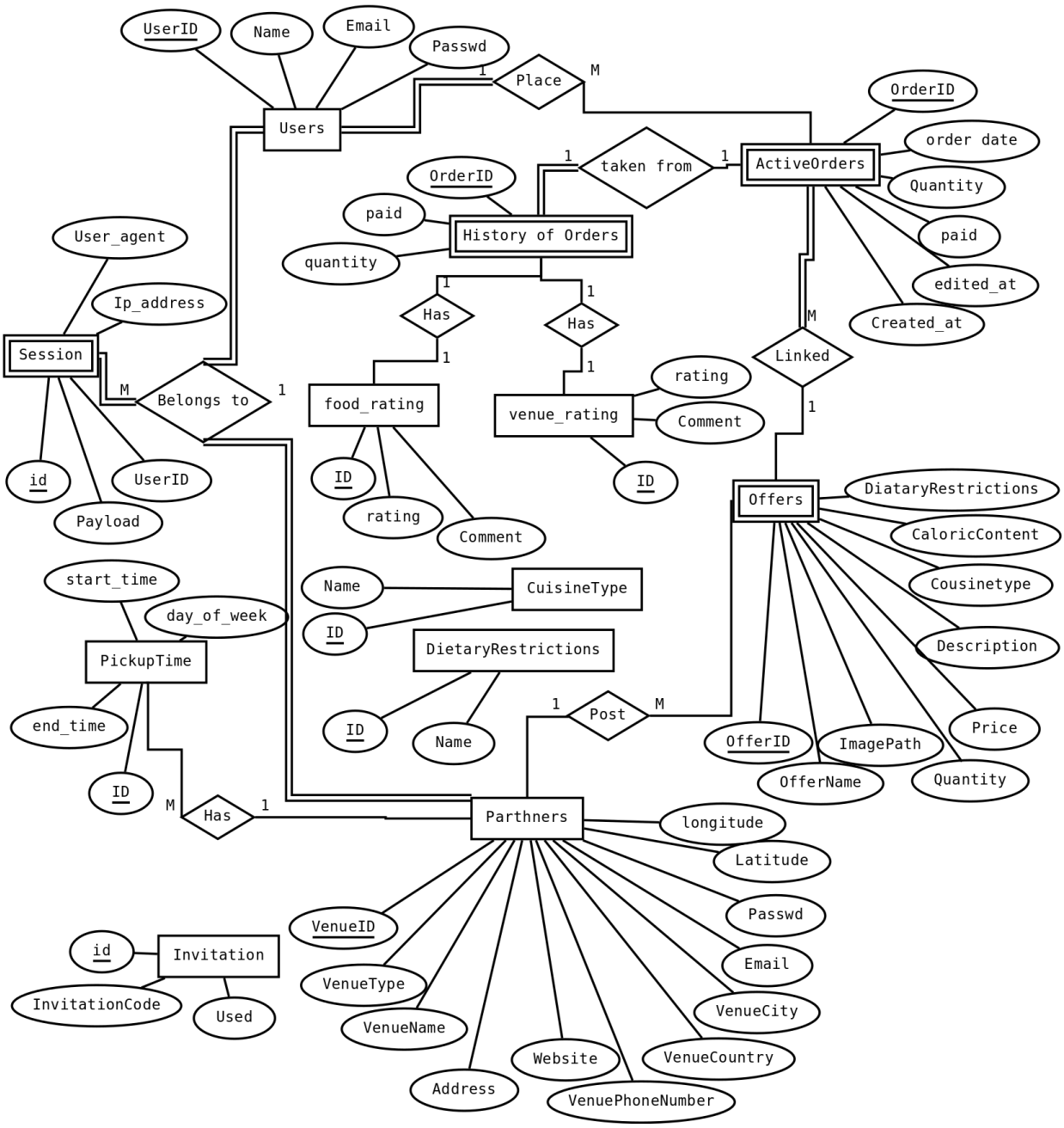

Figure 3. Entity Relationship Model

The ER diagram (Figure 3), represents the database structure of the food delivery platform, capturing key entities and their relationships. The primary entities are described as follows:

- **Users**: The User entity is central to the platform, as only authenticated users can access key features such as reserving food offers and leaving feedback. It contains attributes like

id (primary key), name, email, password, and location. Users have a one-to-many relationship with the **Orders** entity, meaning a user can place multiple orders.

- **Orders**: This entity captures completed orders made by users. It has attributes such as order_id (primary key), user_id (foreign key to Users), o er_id (foreign key to Offers), quantity, paid_status, and created_at. Orders represent finalized purchases.

- **Active Orders**: This entity represents ongoing or uncompleted purchases made by users. It includes attributes like active_order_id (primary key), user_id (foreign key to Users), o er_id (foreign key to Offers), quantity, created_at, and paid_status. This entity allows for real-time tracking of active orders.

- **History of Orders**: Once an order is completed, it is transferred to the History of Orders entity, which captures past transactions. It includes attributes such as history_order_id (primary key), user_id, o er_id, quantity, created_at, and paid_status, allowing users to review their order history and venues to track past sales.

- **Partners**: This entity represents businesses providing surplus food offers. It includes attributes like partner_id (primary key), venue_name, venue_type, address, email, phone_number, and password. Partners have a one-to-many relationship with **Offers**, meaning a partner can list multiple offers.

- **Offers**: This entity stores food offers provided by venues. It includes attributes such as offer_id (primary key), partner_id (foreign key to Partners), name, description, price, quantity, cuisine_type, dietary_restrictions, and image_path. Offers are linked to **Pickup Times**, ensuring users can see when they are available for collection.

- **Pickup Times**: This entity manages the available time slots for picking up offers. It contains attributes like pickup_time_id (primary key), partner_id, day_of_week, start_time, and end_time.

- **Food Ratings**: This entity captures feedback and ratings related to the food offers. Attributes include food_rating_id (primary key), order_id, food_o er_id, rating, and comment. Users can rate the food they receive from a venue.

- **Venue Ratings**: Similar to Food Ratings, this entity captures feedback related to the venue itself, focusing on the service and experience. It includes attributes like venue_rating_id (primary key), order_id, partner_id, rating, and comment.

- **Invitations**: This entity manages partner invitations for registration. It contains attributes like invitation_id (primary key), invitation_code, and used_status, ensuring only authorized partners can join.

- **Dietary Restrictions**: Stores predefined categories such as gluten-free or vegan, associated with food offers.

- **Cuisine Types**: Manages different cuisine classifications (e.g., Italian, Chinese) that partners can use when listing their food offers.

This entity-relationship structure allows for efficient management of user and partner data, real-time food offers, ongoing and historical orders, and user feedback, supporting a comprehensive system for reducing food waste. s.

2.1.5 Relational schema

The Relational Schema (Figure 4), developed based on the entity-relationship model in Figure 3, details the structure of the database, including tables, attributes, data types, keys, and relationships. Below are a few key highlights:

- **Storage of Images and Files**: Image paths, such as imagepath in the foodoffers table, are stored as strings representing the file location on the server. The actual image files are stored externally on the web server, while the database stores only their file paths.
- **Dynamic Attribute Values**: Attributes like cuisinetype in the foodoffers table are stored as actual values. This allows partners to choose from an extensive range of options when creating or modifying offers. This design provides flexibility, eliminating the need for developers to hardcode new options whenever the list of available choices expands..

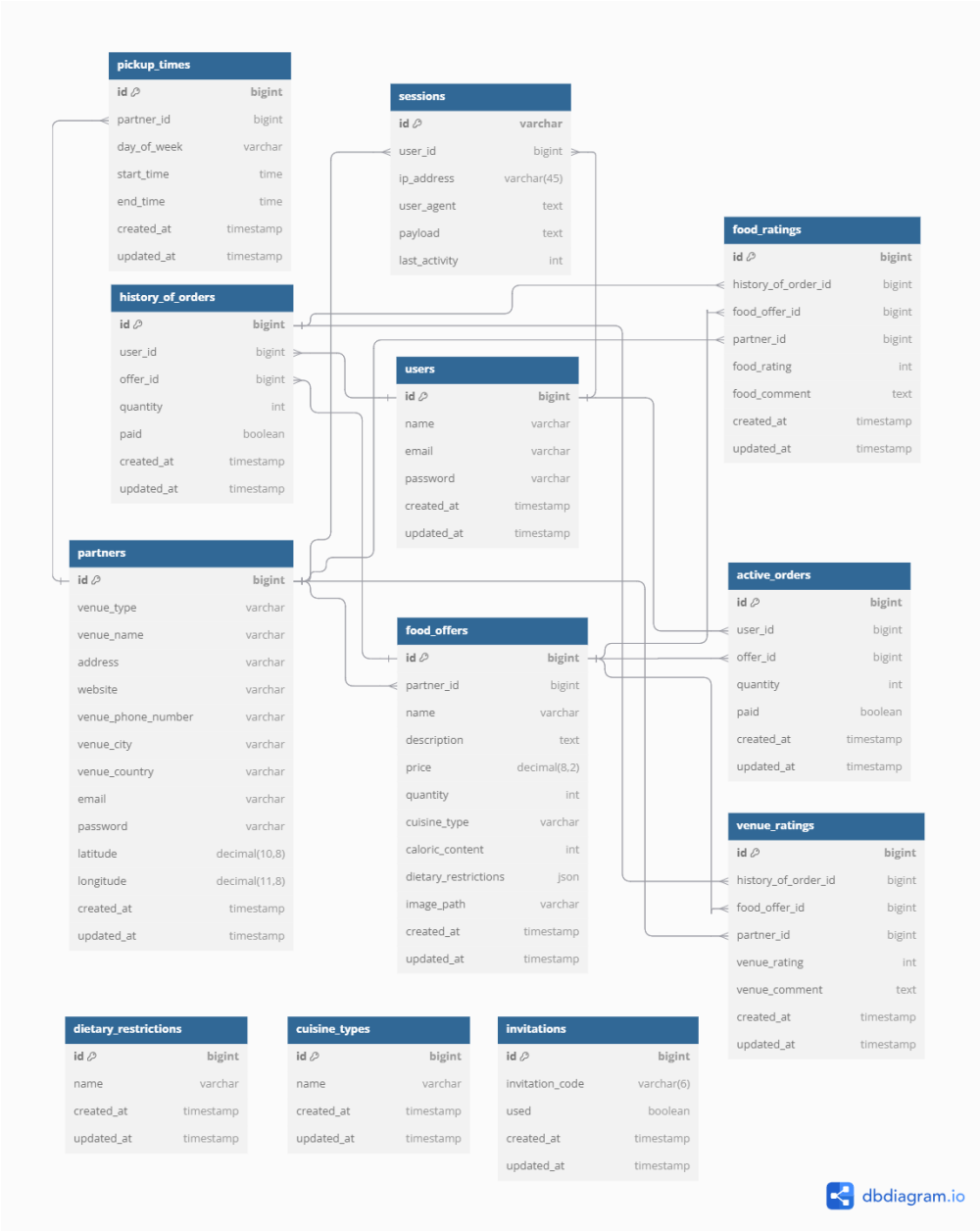

Figure 4. Relational schema

2.2 Functional requirement

User:

- Register an account: Users can create an account by providing basic information such as name, email and password.
- Login/Logout: Users can log in using their email and password, and log out when done.
- Profile Management: Users can update information such as password and see their profile information and see info such as email and name.
- Map Integration: Users can view an integrated map that shows venues with available food offers, enabling easy reservation or selection and distance calculation.
- Reservation Process: Users can reserve available food offers from listed venues

- Payment Process: Users can complete the payment process through a secure payment system integrated into the platform.
- Order History: Users can view their past orders made on the platform, see pickup times and google maps directions to venue location.
- Feedback: Users can leave feedback on the venues and products they have bought and users can see the average rating on venues and respective producs others have bought and left a review

Partner (Business):

- Registration process for businesses with relevant details (name, location, type of food offered, contact information).
- Business Registration: Businesses can register on the platform, providing essential details such as name, location, type of food offered, and contact information.
- Business Profile Management: Businesses can edit their profile information, including location, contact details, password and business pickup times .
- Offer Creation and Management: Businesses can create, update, or remove food offers, including setting prices, descriptions, and availability.
- Order Tracking: Businesses can view incoming sales through a dashboard to manage offers more effectively.
- Feedback Management: Businesses can access feedback from users to understand customer satisfaction and improve services.

2.3 Non-functional requirements

- The platform should be intuitive and easy to navigate for all users
- Simple, clean user interface with clear instructions and feedback
- Ensure the platform can handle a moderate number of requests simultaneously without significant delays.
- The system should maintain accurate and consistent data, ensuring proper handling of user and business information.
- Implement basic security measures to protect user data

2.4 Wire frames

To achieve the final look of the site and check usability, cruishal practice of HCI was used. Surface basic and alternative wireframes were created. During testing and interviewing independent users, the prototypes below were selected to imlement on the final version.

2.4.1 Landing page

For landing page was used one of the Nielsen Heuristics practise called Match Between the System and the Real World, to name the regular "Registration" buttons that makes the decision process of choosing a platform for the user is more tempting. Also, to avoid the need to go to an additional page to obtain information about the platform, it was decided to place the “About Us” section directly on the landing page.

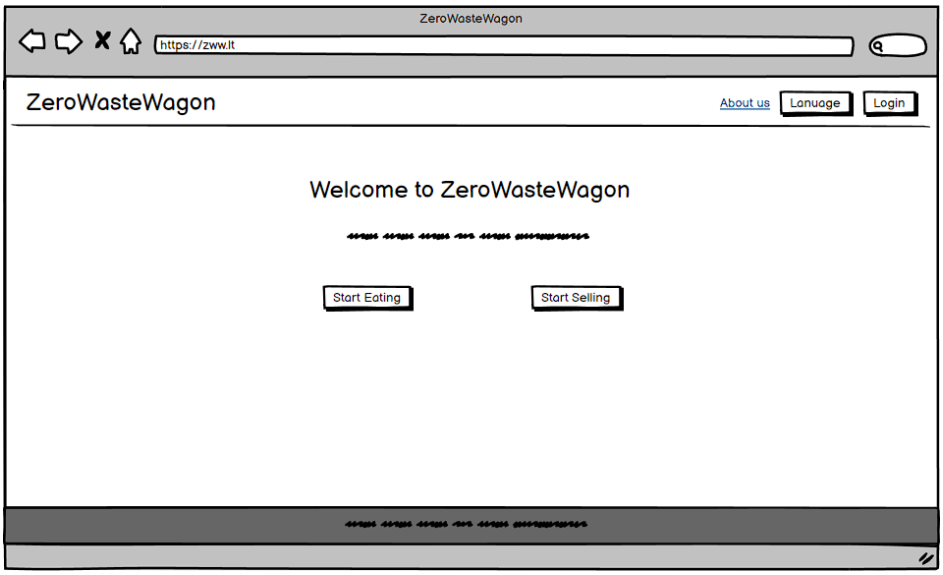

Figure 5. Landing page wire frame

Description of wire frame:
The "Start Eating" button redirect to user registration form.
The "Start Selling" button redirect to partner registration form.
Language selector are provided on the header and will follow user on all pages of the website.

2.4.2 User registration

The simplest and most discreet registration form was created to collect user information for creating an account.

Description of the wire frame:

- Name: This text field allows users to enter their full name.
- Email: This text field is where users enter their valid email address, which will be used for account verification and potentially for future communication.
- Password: This text field is where users create a secure password for their account.
- Finish Registration Button: Once all required information is filled out, users can click this button to submit the form and complete their account registration.

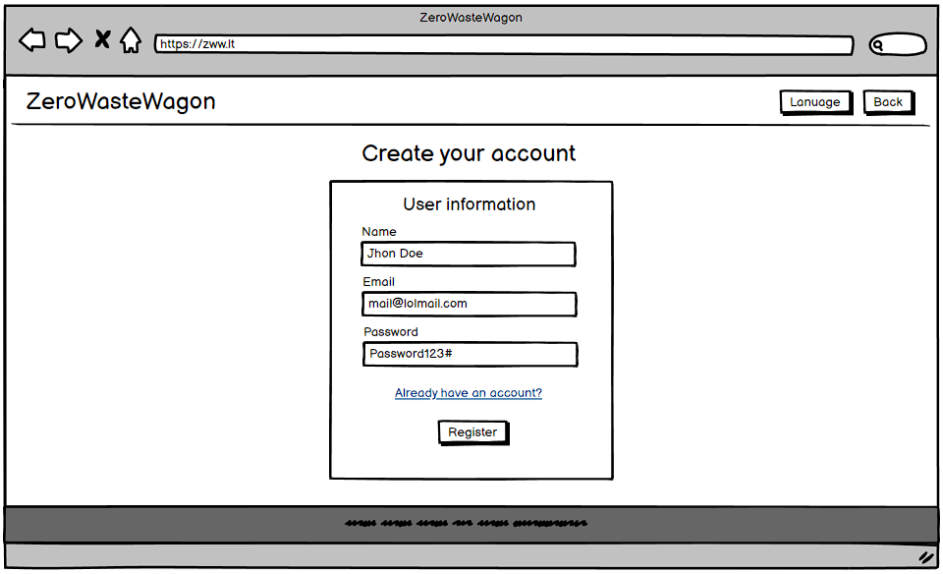

Figure 6. User registration wire frame

2.4.3 Partner registration

The form aim to collect the partner's business detail seating an account. The registration form includes fields for all the necessary information about the establishment, and was divided into two parts to avoid overloading the interface.

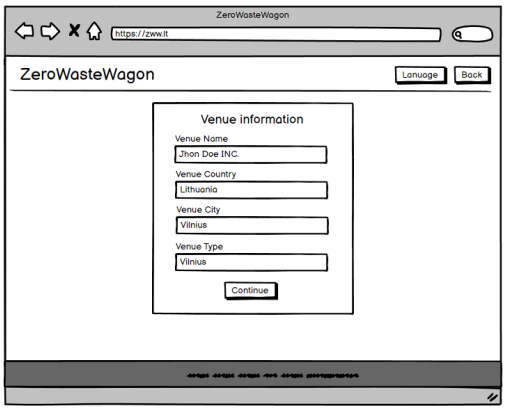

Figure 7. Partner registration (1 step)

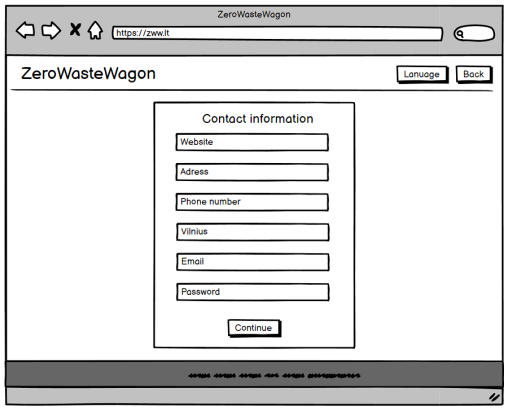

Figure 8. Partner registration (2 step)

Description of the wire frame "Partner registration (1 step)":

- Venue Name: This text field allows a partner to enter the name of the venue.
- Venue Country: This text field is where users specify the country where the venue is located.
- Venue City: This text field allows users to enter the city where the venue is located.
- Venue Type: This text field should allow users to define the type of venue (e.g., conference hall, restaurant).
- Continue Button: Once all the required venue information is filled out, users can click this button to proceed to the next step in the process.

HCI practice was used, to simplify navigation between the steps of registration. In such way partner will easily understand, that on the first step he provides venue information, and contact information on the second step.

Description of the wire frame "Partner registration (2 step)":

- Website: This text field allows users to enter the URL of the venue's official website.
- Address: This text field is where users input the physical address of the venue.
- Phone Number: This text field allows users to enter a valid contact phone number for the venue.
- City: The field is pre-filled with "Vilnius" and allows users to specify the city where the venue is located.
- Email: This text field is where users input the venue's official email address for communication purposes.
- Password: This text field allows users to set a secure password for account access or management.
- Register Button: Final step of registration. After pressing the button, form will be validated and registration process will be finished.

2.4.4 User dashboard

User dashboard was built with using Bottom-up Design Approach. This practice commonly used for websites with similar elements, like products. The page can be separated on two parts tools bar and main content area.

Tools bar description:

- Search Venues: A search field for users to find specific venues or items.
- Sort: A dropdown menu or button to allow users to sort the results by various criteria (e.g., price, distance, rating).
- Filters: A dropdown menu or button to apply filters to the search results (e.g., dietary restrictions, food type).
- Refresh: A button to refresh the search results.

Main content part is a list of establishments and available offers.

Main content description:

- Name: The name of the venue.
- Rating: A star rating indicating the venue's popularity or customer satisfaction.
- Image: A placeholder image for the venue.
- Price: The price of the item.
- Quantity: The number of items available.

- Add to Cart: A button to add the item to the user's shopping cart.

The main part of the content is supplemented with a map to simplify studying the list of establishments and viewing offers near you.

2.4.5 Partner working space

The platform should support a variety of tools for effective offer management, allowing offers to be created and managed in real time. Thus, the functionality will be divided into several pages.

List of offers description:
The page is divided into two parts: a list of current offers and unpublished offers. They have similar structure, but all of products, that was placed in the first list are visible for users and can be unpublished at any time.

- Image: A placeholder image for the offer.
- Offer Name: The name of the offer.
- Description: A brief description of the offer.
- Price: The price of the offer.
- Quantity: The available quantity of the offer.
- Actions: Buttons for managing the offer.
- Edit: A button to edit the offer details.
- Unpublish: A button for published offers to remove the offer from the public listing.
- Publish: A button for unblished offers to add the offer to the public listing.

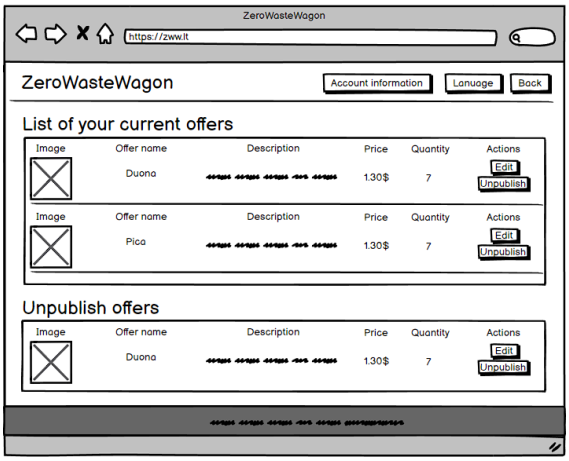

Figure 9. Partner side list of offers

Figure 10. Create the new offer page

To add new offers there is a page for creating new offers. All attributes are filled manually, there is also the possibility to add attributes for filters, such as Type of cuisine and Dietary Restrictions and caloric composition.

3 Implementation

3.1 Tools and Technologies

For the Food Waste Reduction Platform, the choice of `Laravel` [9] as the primary web development framework is driven by its robust architecture, extensive community support, and comprehensive features that facilitate rapid application development. Laravel's elegant syntax and built-in functionalities streamline tasks such as routing, sessions, and caching, making it highly efficient for building complex applications. [8]

- **Frontend Design:** The platform utilizes a combination of CSS frameworks and custom styling for its design. While `Tailwind CSS` [14] is used in certain areas for its utility-first approach, providing flexibility in creating responsive and customizable elements, additional CSS and JavaScript components are integrated for more complex layouts and dynamic features. This combination ensures a user-friendly interface that adapts well to different devices and screen sizes, improving user experience and accessibility across the platform.

- **Database Management:** `MySQL` [2] is used as the database management system, chosen for its reliability and support within the Laravel ecosystem. Integration with Laravel's `Eloquent ORM` provides a powerful yet simple ActiveRecord implementation for working with the database. This allows for fluent and expressive database queries and is instrumental in managing complex data relationships efficiently.

- **Mapping and Geolocation:** Integration with `Leaflet.js` for interactive maps, and use of the `OpenStreetMap API` [3] for geocoding addresses and the `HTML Geolocation API`[1] for user location pin point and for calculating the distance between user location and each venues location using the `Haversine formula`. These tools are essential for facilitating the geographic features of the platform, enabling users to easily find food offers based on their location utilizing offer sorting based on distance.

- **Authentication and Security:** The platform employs Laravel's native session-based authentication mechanism to secure partner logins and manage sessions. A specialized `middleware` authentication guard for partners distinctly from other users, ensuring a secure and isolated access environment for business partners.

- **Payment Processing:** The integration of `Stripe` [4] for payment processing is implemented through Laravel's service container and package ecosystem, providing a robust and secure environment for handling transactions. This allows the platform to manage financial transactions efficiently, with minimal setup and maintenance overhead.

These technologies collectively support the platform's need for a robust, scalable, and user-friendly application, addressing the core functionalities of reducing food waste through an accessible digital solution.

3.2 User authentication

User registration on the platform is managed through a secure and robust process using Laravel's web framework functionalities. Users are prompted to provide essential information such as name,

email, and password, all of which are validated to meet specific criteria to ensure data integrity and security.

- **Form Submission and Validation:** The registration process begins with the user completing a form that captures their name, email, and password. Laravel's validation mechanism is employed to ensure that all entries meet the required standards:
  - Names must be a string with a maximum length of 255 characters.
  - Email addresses are required to be unique within the `users` table, ensuring no duplicate entries. They must also conform to standard email formatting.
  - Passwords must be at least 8 characters long and confirmed via a matching confirmation field to prevent typing errors.

  Custom error messages are provided to guide users in case of validation failures, enhancing the user experience by making error notifications clear and specific.

- **Account Creation:** Upon successful validation, the user's details are stored in the database. The password is securely hashed using Laravel's `Hash` facade, which utilizes the Bcrypt hashing algorithm. This step ensures that sensitive user information is securely stored and safeguarded against unauthorized access.

- **Redirect and Session Handling:** After the account is successfully created, users are redirected to the login page where they can begin using their credentials to access the platform. A session flash message confirms the successful registration, providing immediate feedback to the user about the registration status.

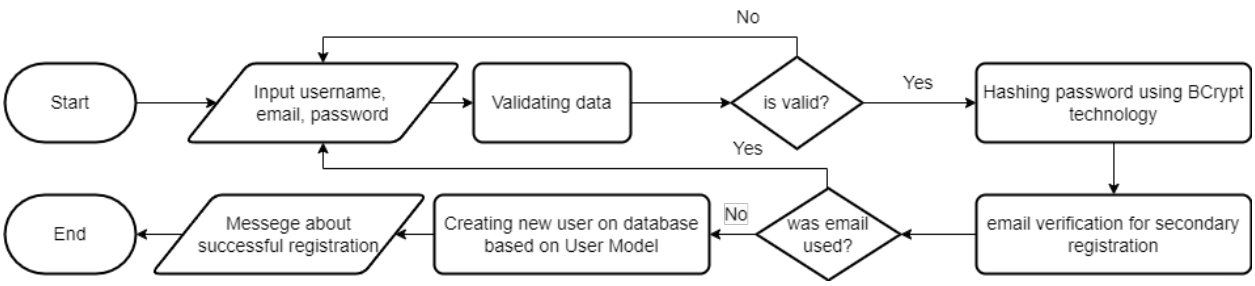

Figure 11. User registration flow

This methodical approach not only secures user information but also streamlines the process of integrating new users into the platform, making it accessible and user-friendly.

For the user login process, the system will ask you to enter your email and password. After clicking the "Login" button, data validation will occur, namely checking the availability of input data, compliance with the email format and password length. After this, the process of checking the presence of credentials in the database and their compliance will begin. If validation fails, the user will return to the same page with appropriate messages about the problem. If the validation and verification of credentials is successful, the partner will be redirected to the user dashboard.

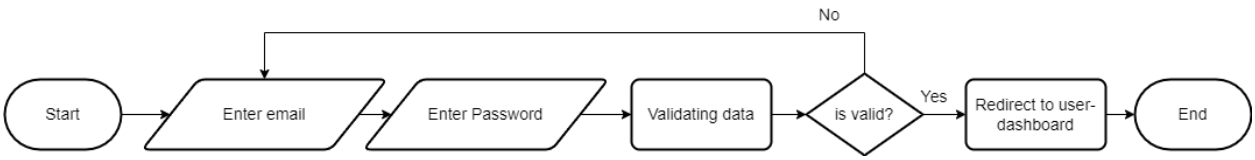

Figure 12. User login flow

3.3 Partner authentication

Partner registration in the Food Waste Reduction Platform is a multi-step process designed to ensure that only authorized and verified businesses can create and manage food offers. This process is secured by an invitation system and is detailed across several registration stages, each handling specific aspects of the partner profile and business details.

- **Invitation Code Validation:** The initial step requires partners to enter a unique invitation code, which is verified against the database for existence and unused status. Upon successful validation, the code is marked as used, and the partner progresses to the next step of registration. This step ensures that only invitees can register, maintaining platform exclusivity and integrity.

- **Venue Information Submission:** In the second step, partners are prompted to input detailed information about their venue, including country, city, name, and type. This information is crucial for categorizing and locating the venue on the platform. The data is temporarily stored in the session to facilitate a seamless registration process.

- **Geolocation and Final Details:** The final registration step involves collecting contact information and attempting to geocode the venue address using a `GeocodingService`. Successful geocoding is essential for accurate mapping on the platform.

```
// Pseudo code for using Nominatim OpenStreetMap API to geocode an address

function geocode_address(address) {
    // Send a request to OpenStreetMap API with the address
    response = requests.get("https://nominatim.openstreetmap.org/search", {
        params: {
            "q": address,
            "format": "json"
        }
    });

    // Parse the response to extract latitude and longitude
    location = response.json()[0];
    latitude = location["lat"];
    longitude = location["lon"];

    return {latitude, longitude};
}
```

Partners are required to provide a website URL, address, phone number, email, and a secure password. The email must be unique among registered partners, and the password is hashed for security before being stored in the database.

- **Completion and Session Clearing:** Once all steps are successfully completed, the partner's record is created in the database, including geolocation coordinates (latitude and longitude). After registration, session variables related to the process are cleared to maintain security and privacy..

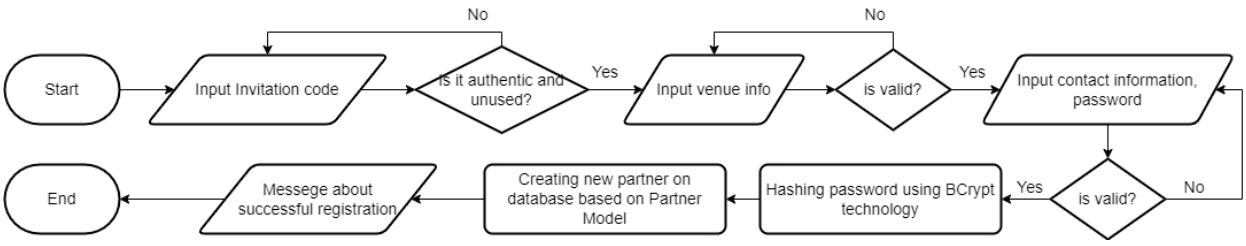

Figure 13. Partner registration flow

This structured and phased approach not only secures the registration process but also ensures that all necessary information is accurately captured and validated, setting a strong foundation for partner activities on the platform.

For the partner login process, the system will ask you to enter your email and password. After clicking the "Login" button, data validation will occur, namely checking the availability of input data, compliance with the email format and password length. After this, the process of checking the presence of credentials in the database and their compliance will begin. If validation fails, the partner will return to the same page with appropriate messages about the problem. If the validation and verification of credentials is successful, the partner will be redirected to the partner dashboard.

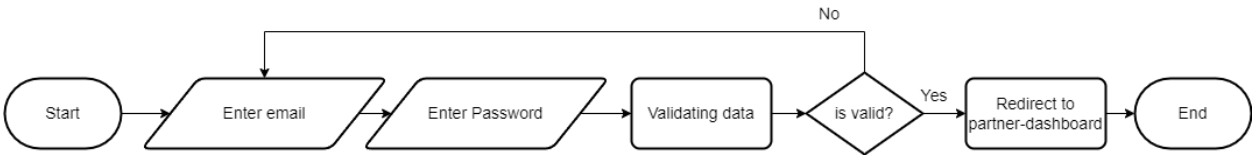

Figure 14. Partner login flow

3.4 Offers Creation System

The platform provides a straightforward yet effective system for partners to manage food offers, giving them control over the items they wish to sell. The system includes essential features such as creating, updating, and managing offers, all through the partner dashboard.

- **Offer Creation:** Partners can create new offers through a simple form that captures basic information like the offer name, description, price, quantity, and image. Additional fields like dietary restrictions, cuisine type, and caloric content are optional. When creating an offer, the system validates the required fields, ensuring that all entries meet the necessary standards. The image is uploaded to the server, and only its path is stored in the database, keeping the process lightweight. Once validated, the offer is saved to the database and becomes visible to users..

- **Offer Editing:** Partners can edit existing offers, including updating the name, price, description, and quantity. They can also update the image associated with the offer. When a new image is uploaded, it is stored on the server, and the database is updated with the new image path.

- **Offer Management:** Offers can be unpublished by setting their quantity to zero, making them unavailable for reservation. This method ensures that offers are not deleted but can be reactivated later..

These features are integrated into the platform's partner dashboard, providing a seamless experience for managing food offers. This system not only empowers partners to effectively control their listings but also supports the platform's goal of reducing food waste by facilitating the timely management of offers.

3.5 Map Integration

The Food Waste Reduction Platform incorporates an interactive map to facilitate location-based services, significantly enhancing the user experience by visually representing venue locations and offers. This map is powered by Leaflet, an open-source JavaScript library for mobile-friendly interactive maps, and utilizes OpenStreetMap for its tile layers. [13]

- **Map Initialization and Venue Representation:** Upon loading the user dashboard, the map is initialized using OpenStreetMap API, Venues are marked on the map using custom icons, and clicking these icons reveals detailed information about the venue, such as name and address, enhancing user interaction and providing context for the offers available at each location.

- **User Location and Interaction:** The user's position is marked on the map with a distinct icon, differentiating it from the venue markers. This visual distinction helps users orient themselves in relation to the venues. Additionally, users can interact with the map to explore different areas, zoom in and out, and obtain a broader or more detailed view of food offers in various parts of the city.

```
// Pseudo code for using HTML Geolocation API to get user's location

```

```
function getUserLocation() {
    if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(function(position) {
            var latitude = position.coords.latitude;
            var longitude = position.coords.longitude;
            // Use latitude and longitude as needed
        });
    } else {
        // Geolocation is not supported by this browser
    }
}
```

- **Dynamic Venue and Offer Data:** As users navigate the platform, venue data, including geographic coordinates and related offers, are dynamically retrieved and updated on the map. This feature ensures that the information displayed is current and accurate, reflecting real-time changes in offers and venue statuses.[15]

- **Distance calculation between user's location and each venue's location:** The platform calculates the distance between a user's location and various venue locations to provide relevant and proximate food offers. This is achieved using the Haversine formula, which computes the distance between two points on the Earth's surface given their latitude and longitude coordinates. This calculation is crucial for offering the ability to sort venues and offers by ascending distance while presenting the most accessible food offers first.

```
// Pseudo code for calculating distance using the Haversine formula

function haversine_distance(lat1, lon1, lat2, lon2) {
    var R = 6371;  // Earth radius in kilometers

    var dlat = Math.radians(lat2 - lat1);
    var dlon = Math.radians(lon2 - lon1);

    var a = Math.sin(dlat / 2) ** 2 + Math.cos(Math.radians(lat1)) *
        Math.cos(Math.radians(lat2)) * Math.sin(dlon / 2) ** 2;
    var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));

    var distance = R * c;
    return distance;
}
```

- **Accessibility and Responsiveness:** The map interface is designed to be fully responsive, ensuring a seamless experience across various devices and screen sizes. Accessibility is further supported by intuitive controls and clear, distinguishable icons, making the map easy to use for all users, regardless of their tech-ability.

This sophisticated map integration not only aids in the geographical visualization of venues and offers but also plays a crucial role in the user's decision-making process by providing a spatial context to the available food offers.

3.6 Shopping Cart

The shopping cart on the Food Waste Reduction Platform is a critical component that facilitates the temporary reservation and eventual purchase of food offers. It is designed to provide a responsive and intuitive user experience, allowing customers to manage their selected items before checkout.

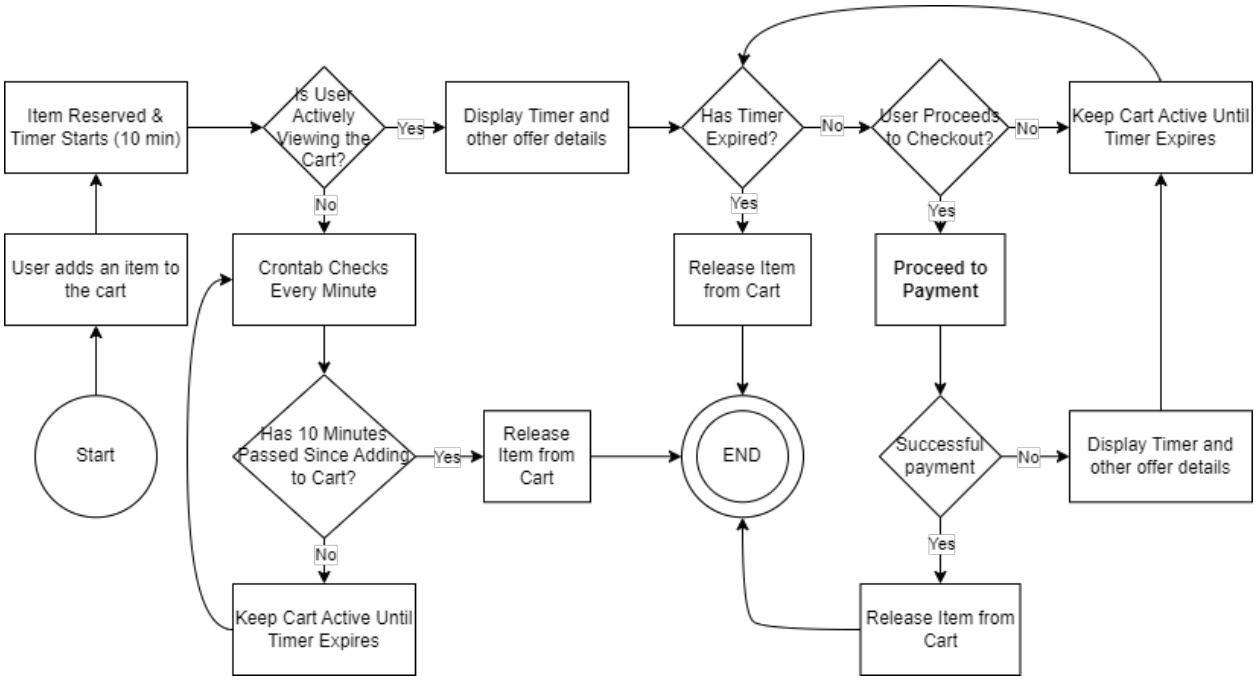

Figure 15. Shopping cart flow

- **Adding Items to Cart:** When users choose to add offers to their cart, the system verifies the available stock against the requested quantity. If sufficient stock is available, the quantity is reserved by decrementing the stock and creating an active order record associated with the user's account. This operation ensures that the stock levels are immediately updated to reflect the reserved items, preventing overbooking of limited offers.

- **Cart Display and Management:** The shopping cart interface displays all active orders for the logged-in user. Each item's name, quantity, price, and a calculated subtotal are shown. The cart also integrates a real-time countdown timer that tracks the reserved duration of each item, set to 10 minutes. This feature helps manage the urgency of the purchasing decision and ensures that the items are either purchased in a timely manner or returned to the stock if the timer expires.

- **Checkout Process:** The checkout button in the cart initiates the payment process through Stripe, a secure online payment system. Upon clicking the button, a server-side request creates a Stripe checkout session, and the user is redirected to complete the payment. This integration not only simplifies the payment process but also enhances security by handling sensitive financial information outside the platform's direct infrastructure.

```
// Pseudo code for creating a Stripe checkout session

function createCheckoutSession(request) {
```

```
    \Stripe\Stripe::setApiKey(config('services.stripe.secret'));

    var session = \Stripe\Checkout\Session::create({
        payment_method_types: ['card'],
        line_items: [{
            price_data: {
                currency: 'eur',
                product_data: {
                    name: 'Total Cart Value'
                },
                unit_amount: request.total * 100  // amount in cents
            },
            quantity: 1
        }],
        mode: 'payment',
        success_url: route('payment.success'),
        cancel_url: route('payment.cancel')
    });

    return response.json({id: session.id});
}
```

- **User Feedback and Notifications:** The platform provides immediate feedback via alert messages for various cart actions, such as successful addition to the cart and expiration notices if the cart's reserved timer runs out. This feedback mechanism is essential for maintaining clear communication with the user, enhancing the overall shopping experience.

This shopping cart system supports the platform's goals by ensuring effective management of food offer reservations, facilitating quick turnover of perishable goods, and reducing food waste through timely purchases.

3.7 Payment System

The payment system on the Food Waste Reduction Platform is a crucial component that facilitates secure and efficient financial transactions. It leverages Stripe, a widely recognized payment processing service, to handle payments securely and ensure the integrity of transaction data. [15]

- **Stripe Integration:** The integration with Stripe begins with the creation of a checkout session using Stripe's API. This session configures the payment details, including the total amount, currency, and the type of payment method accepted (e.g., credit or debit cards). The amount charged is dynamically calculated based on the total value of the cart, which is passed to the Stripe session in cents (as required by Stripe's API).

- **Checkout Session:** When a user decides to checkout, a Stripe checkout session is initiated. This session specifies the payment attributes, including a unique session ID returned by the Stripe API, which is then used to redirect the user to Stripe's secure checkout page. The checkout process is configured to redirect users to either a success or a cancellation URL based on the transaction's outcome.

- **Transaction Completion:** Upon successful payment, the user is redirected to a success URL where the system processes the conversion of active orders into historical records.

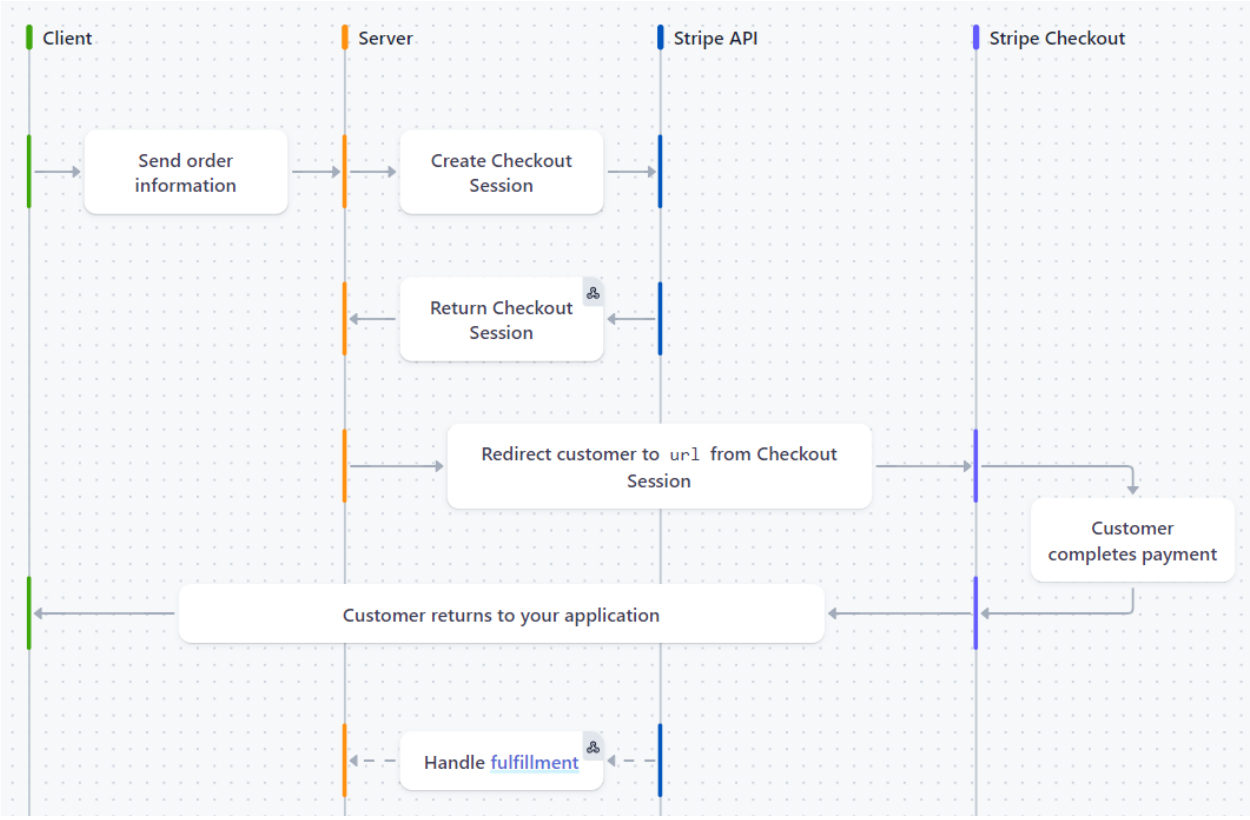

Figure 16. Stripe checkout flow.

Each active order associated with the user is marked as paid and transferred to the history of orders to maintain a record of transactions. This step ensures that the order lifecycle is properly managed and provides a traceable history of purchases for both auditing and user reference.

- **Cancellation Handling:** If a user cancels the transaction or if the payment fails, the user is redirected to a cancellation page. This allows users to attempt the payment again or to modify their cart before proceeding with another checkout attempt, enhancing the user experience and flexibility in handling payment issues.

This robust payment system not only ensures financial security but also enhances user confidence through reliable processing of transactions and clear communication of the payment outcomes.

3.8 Screenshots of working system

Previously prepared wireframes were used to create the final look of the platform. It helped map out content hierarchy and placement before coding begins, ensuring the right information is highlighted and organized logically for users. Also it emphasized the user journey and interaction without getting bogged down by design elements like color schemes or typography. They prioritize functionality and navigation, ensuring a user-friendly website. In final, wireframes designed with user-centered layouts, optimized the flow before visual design began.

Landing page provides clear and concise information about the platform's purpose and offers two primary actions for users: "Start Eating" and "Start Selling." The page also includes basic navigation elements, such as links to "About Us" and "Login."

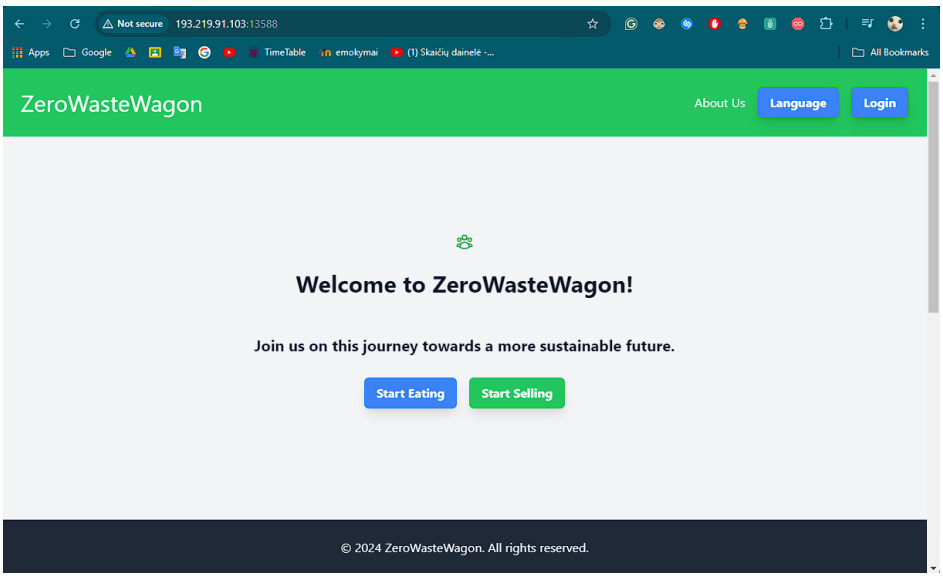

Figure 17. Landing page

Interacting with the landing page, the user will be able to register or log in to an existing account. After the registration process, the user immediately gets to the dashboard, where he can view the available offers.

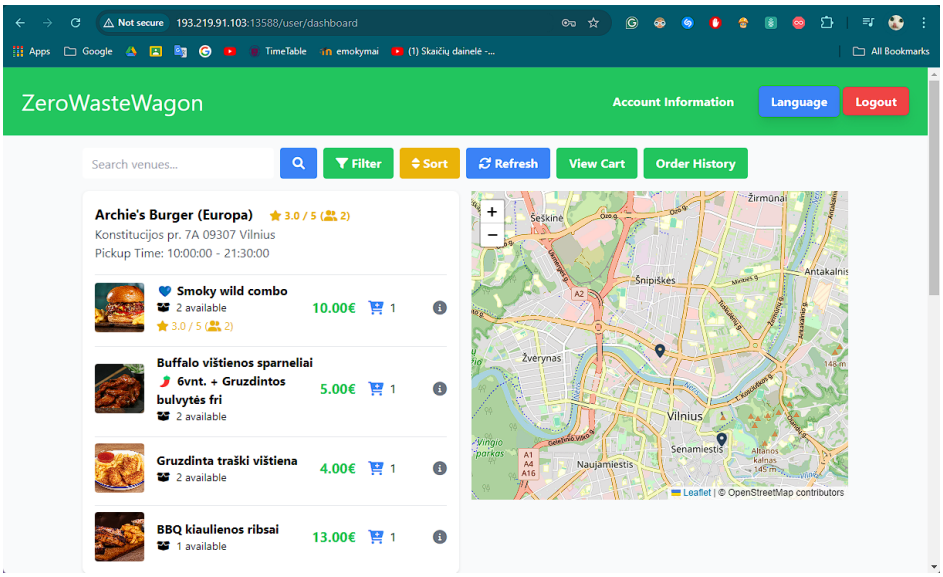

Figure 18. User dashboard

Dashboard allows users to search for nearby zero-waste venues, view their menu items, place orders, and track their order history. Key features include a map-based search function, filters to refine search results, and a clear display of menu items with availability and pricing information. Additionally, the dashboard provides a convenient way for users to manage their cart, view their order history, and access account information.

The user can add an order to the cart in two ways: immediately, by clicking the cart button, or after viewing detailed information about the offer.

The offer details window provides a detailed view of a specific menu item. It displays the name of the item, a description of its components, the current availability, the price, the cuisine type, the caloric content, and any dietary restrictions. The window also includes a high-quality image of the

dish and a button to add it to the user's cart.

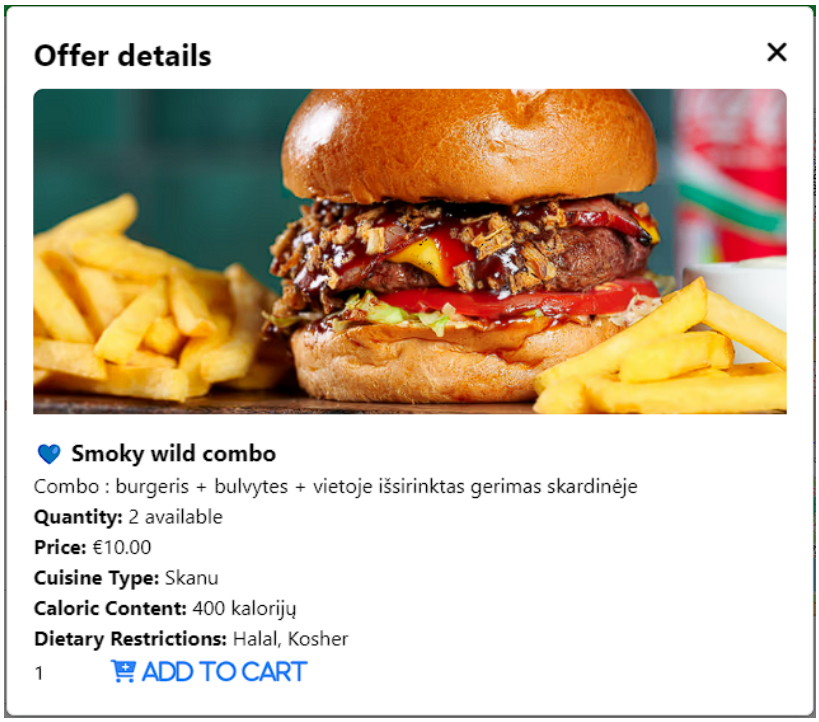

Figure 19. Product info

The user also has the ability to view the cart and purchase history. Cart lists each product, its quantity, unit price, and subtotal. Additionally, the page displays the total order amount and provides two buttons: "Proceed to Checkout" and "Back to Shop." This allows users to review their selections, make any necessary changes, and proceed to the checkout process when they are ready to complete their order. There is also a product reservation timer in the basket.

Order history page provides users with a record of their past orders. It displays a list of orders, with each order containing details such as the order date, items purchased, quantity, price, and venue information. Users can expand each order to view more specific details, including the product description, image, and order time. This page allows users to track their past orders and review their purchase history.

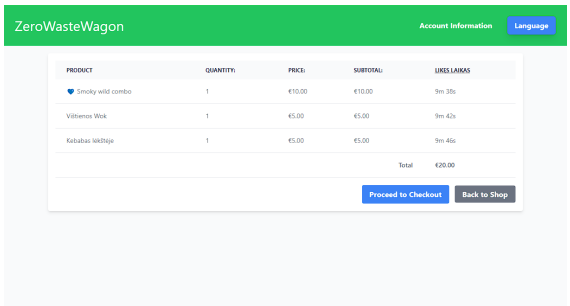

Figure 20. User cart

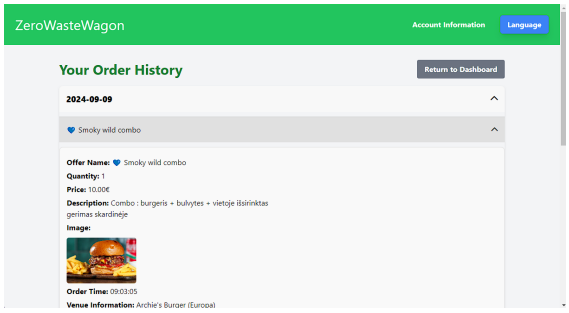

Figure 21. User history of orders

Partners also have the option to create an account and log into an existing one using the landing page. After authorization, partners get to their dashboard, where they can effectively manage offers, view order history, and create new offers.

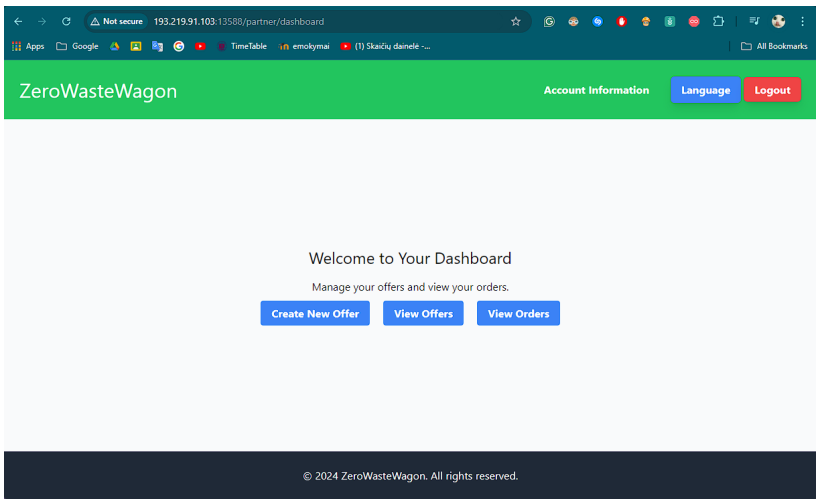

Figure 22. Partner dashboard

Create new food offer page is a form designed for venues to add their menu items to the platform. It requires users to input information such as the offer name, description, cuisine type, dietary restrictions, price, quantity, and an image of the dish. This page allows venues to easily create and manage their offerings on the ZeroWasteWagon platform, making it accessible to customers.

Additionally, the page includes a "Return to Dashboard" button, allowing venues to navigate back to their main dashboard after submitting their offer. The form also includes a "Choose File" button for uploading an image of the dish. This user-friendly interface simplifies the process of adding new menu items to the platform, making it convenient for venues of all sizes.

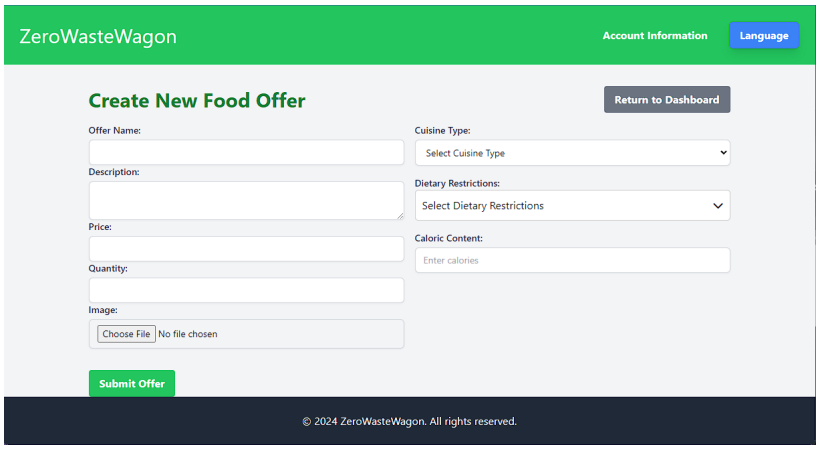

Figure 23. Creating offer page

After creating an offer, the partner will be able to check the list of realized offers and manage them. List of offers page provides a comprehensive overview of a venue's current menu offerings. It displays a list of all active offers, including the image, offer name, description, price, quantity, and available actions. Venues can use this page to manage their menu items, edit existing offers, or unpublish them if necessary. This page is divided into two parts, where offers that are not publicly available are also presented. They have the same structure, but users will not be able to view and order them. The partner has the opportunity to publish an offer at any time, indicating the available quantity.

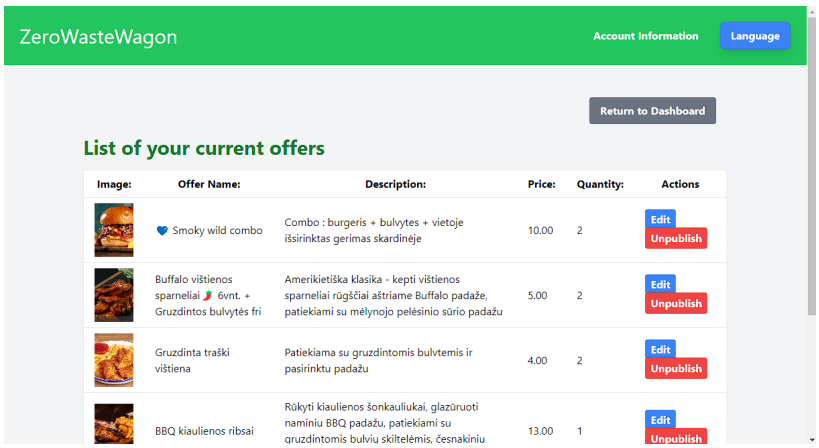

Figure 24. Partner list of offer

Order history page provides a detailed overview of a venue's past orders. It displays a list of orders, with each order containing information such as the order date, customer details, items purchased, quantity, price, payment status, and feedback ratings. Venues can use this page to track their order history, analyze customer preferences, and identify areas for improvement.

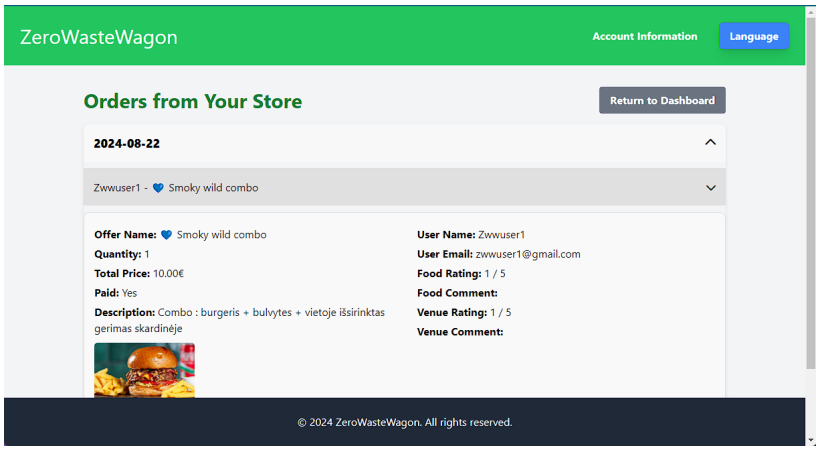

Figure 25. Partner order history

This interface simplifies the process of reviewing order history and gaining valuable insights into customer behavior.

3.9 HCI practices used

The use of HCI practices was fundamental when creating the platform. Nielson Usability Heuristics [11] chose the main set of rules for User Interface Design, which number 10 essential principles for creating a quickly mastered user-friendly design. So we Hence, the platform does not violate the following principles:

1. Visibility of System Status;
2. Match Between the System and the Real World;
3. User Control and Freedom;
4. Consistency and Standards;
5. Error Prevention;
6. Recognition Rather than Recall;
7. Flexibility and Efficiency of Use;
8. Aesthetic and Minimalist Design;
9. Help Users Recognize, Diagnose, and Recover from Errors;
10. Help and Documentation.

Visibility of system status follows users at every step on the platform. By pressing any button user won't be missed on the site, also in any case regarding to user Control and freedom principle it's possible to leave any page and explore the website as the user wants.

Studying the new system won't take a lot of time because of the implemented "Match Between the System and the Real World" principle, because of the use of SVG icons for buttons where there could be misunderstanding, like dietary restrictions. Also, the platform was created using basic standards of UX/UI design such as color and font selection.

One of the most important practices was the use of information architecture solutions, such as fine-grained ontology. For dashboards, a Bottom-up Design Approach which ideally meets the requirements of websites with the big amount of pretty similar data.

4 Conclusions and Future Work

In this project, we introduced a platform designed to connect users with food offers from various venues, promoting sustainability and reducing waste. Users can explore offers from different venues, reserve and pay for them, and provide feedback, while partners can manage their listings, track orders, and respond to user feedback. The platform integrates key features such as user registration, an interactive map to locate venues, filtering of offers, and a secure payment system, all built to ensure a seamless and user-friendly experience.

Moving forward, there are several areas where the platform can be further enhanced. A key improvement would be the introduction of an email service, which could provide users with tailored special offers, follow-up details after purchases, and important updates related to their orders.

Additionally, the order management system for partners could be improved, enabling businesses to more efficiently track and manage reservations and sales.

Another potential enhancement would be reconsidering the login process. Currently, users must log in to see the available offers. However, the platform could be adapted so that users only need to log in when they decide to make a purchase, improving the overall user experience by allowing anonymous browsing.

Lastly, creating a mobile app version of the platform could significantly expand its accessibility, providing users with a more convenient and optimized experience on their mobile devices. This, coupled with push notifications, could offer users real-time updates on new offers and important order information, enhancing user engagement.

These improvements would help ensure the platform remains competitive, user-friendly, and scalable as it continues to grow.

References

[1] Geolocation API Documentation, 2024. Accessed: May 22, 2024.

[2] MySQL Documentation, 2024. Accessed: May 22, 2024.

[3] OpenStreetMap API v0.6 Documentation, 2024. Accessed: May 22, 2024.

[4] Stripe Documentation, 2024. Accessed: May 22, 2024.

[5] Diego Gama Amaral and Renato J Orsato. Digital platforms for food waste reduction: The value for business users. *Business Strategy and the Environment*, 32(4):1373--1387, 2023.

[6] Thang Ho. Stripe payments in web application. 2019.

[7] Jiequan Hong, Anicia Jaegler, and Olivier Gergaud. Mobile applications to reduce food waste in supply chains: a systematic literature review. *British Food Journal*, 126(2):509--530, 2024.

[8] Krisna Wira Indrawan, Sena Kristiawan, Timothy Micha Zoe, Steffi Tania, Febrian Ivan Song, Jeremy Nathaniel Saputra, Islam Nur Alam, and Nur Afny Catur Andryani. Digital marketplace design development for creative recycle product. In *2023 IEEE 9th Information Technology International Seminar (ITIS)*, pages 1--6, 2023.

[9] Laravel Team. Laravel Documentation, 2024. Accessed: May 22, 2024.

[10] United Nations. International day of awareness of food loss and waste 2024, 2024.

[11] Jakob Nielsen. Enhancing the explanatory power of usability heuristics. In *Proc. ACM CHI'94 Conf.*, pages 152--158, Boston, MA, April 1994.

[12] United Nations Environment Programme. Food waste index report 2024. think eat save: Tracking progress to halve global food waste.

[13] Bharani S, E. Mohanraj, Sujith M, and Vinothkumar P. Optimizing surplus food management: A user-centric application approach. In *2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE)*, pages 1--6, 2024.

[14] Tailwind CSS Team. Tailwind CSS Documentation, 2024. Accessed: May 22, 2024.

[15] Erik Wittern. Web apis - challenges, design points, and research opportunities: Invited talk at the 2nd international workshop on api usage and evolution (wapi '18). In *2018 IEEE/ACM 2nd International Workshop on API Usage and Evolution (WAPI)*, pages 18--18, 2018.