User Management & Authentication

1. Sign up - MARA

Description:

The system shall allow users to create an account. The application shall display a sign-up form on the home page. The user interface must present a clean and intuitive layout with the following components: five clearly labeled input fields: **Username:** The user must input their desired username (username must be *unique*). **Email Address:** The user must input their desired email address; **Password:** The user must input a secure password that meets the security criteria; **Phone Number:** The user must input their phone number for account recovery or authentication purposes; **Type of user** field that has only 2 options: seller or buyer; a '**Submit' Button** to submit the registration details. All input fields should be properly labeled for accessibility, using clear and concise text, such as "Email Address," "Password," and "Phone Number".

The sign-up form must ensure that the user's input is valid and correct before submission for: **Usernames:** ensure that usernames are unique and at least 4 characters long; provide alternative suggestions if the desired username is taken. **Email Validation:** ensure that the email input field is not empty; validate the email format using a regular expression to ensure it follows a standard email pattern; display an error message if the email format is invalid, e.g., "Please enter a valid email address.";check if the email is already in use and display an appropriate message, e.g., "This email is already registered.". If an account with the same email already exists, the system must not allow duplicate registration. **Password Validation:** ensure the password field is not empty; enforce a password policy that requires a minimum of 8 characters, at least one uppercase letter, at least one number, at least one special character. Display an error message if the password does not meet the criteria, e.g., "Password must be at least 8 characters long and include an uppercase letter, a number, and a special character.". **Phone Number Validation:** ensure the phone number is in a valid format (the phone number should have the +40 prefix (romania area code), followed by 9 numerical characters); display an error message if an invalid format is entered, e.g., "Please enter a valid phone number.". **Type of User validation**: ensure that the user chooses one of the two types (buyer/seller). **Submission:** the system shall validate the input data, show error messages for the fields that are invalid. The system shall proceed with the storing of the data only after all the validations are made.

The registration system must be feasible to implement within the given resources and technology constraints: the registration functionality must be able to handle up to **10 concurrent users** without performance degradation, the system must complete authentication requests within **2 seconds**, the email validation and uniqueness checks must be optimized to minimize delays, the system must securely store user data with **hashed passwords.**

After a successful sign-up, the user gets redirected to the login page.

2. Log in - ARIANA

Description:

Secure Authentication System

Login Task Description

The secure authentication system must provide a login functionality that is clear and unambiguous. The user interface should present a clean and intuitive layout with three clearly labeled input fields: one for the email address, where the user must input their registered email, another for the password, which should correspond to the entered email address, and a third for the CAPTCHA text that was generated. A CAPTCHA mechanism is requiring the user to enter the correct text before submitting their credentials. A LogIN button must be included to allow users to proceed with their login attempt, along with a sign-up button that redirects them to the registration page. A SIgnUp button should be displayed, which will redirect the users that do not have an account yet to the Sign up page.All input fields should be properly labeled to ensure accessibility, using clear and concise text such as "Email Address," "Password," and "Enter CAPTCHA.", “”Don t have an account yet? Click here!”

To ensure validity and correctness, the login form must check that the user's input is properly formatted before submission. The email input field should not be empty, and its format must be validated using a regular expression to confirm it follows a standard email pattern. If the format is invalid, an error message such as "Please enter a valid email address" must be displayed. Similarly, the password field should not be empty, and an appropriate error message like "Password cannot be empty" should be shown if left blank. The CAPTCHA field must also be filled, and the text entered by the user must match the generated CAPTCHA. If the user fails to provide the correct CAPTCHA, an error message such as "CAPTCHA verification failed" must be displayed. When the submit button(LogIn) is clicked, all necessary field validations must be performed, and any errors should be clearly displayed to the user.Moreover it should open the correct page(for buyer/seller)

The login system must comprehensively handle all relevant use cases and validations to ensure completeness. A registered user should be able to log in by entering a valid email address and password, along with the correct CAPTCHA text. If the login attempt fails due to incorrect credentials, an error message such as "Invalid email or password" must be displayed. To enhance security, the CAPTCHA mechanism must always be required to ensure that login attempts are performed by a real user rather than an automated script. Additionally, the system should implement an account-locking feature that temporarily disables login attempts after a user exceeds a predefined number of failed attempts, such as five. In such cases, the system should display a message informing the user, stating, "Too many failed login attempts. Please try again later."

For the login process to be verifiable, the backend system must accurately verify the email, password, and CAPTCHA text entered by the user. This should be achieved by comparing the provided credentials against those stored securely in the Users table. Passwords must be stored using a secure hashing algorithm, ensuring that the system verifies hashed passwords rather than plain-text credentials. The CAPTCHA system must generate distorted text and require the user to input the correct characters before allowing login attempts to proceed. Additionally, failed login attempts should be tracked in the database, and if the limit of five attempts is exceeded, the account must be locked for a predefined duration. Once the lockout period expires, the failed attempt counter should reset, allowing the user to attempt logging in again. If incorrect credentials are provided, the system must display a clear and specific error message, such as "Incorrect password" or "Invalid email."

The login system must be feasible to implement within the available resources and technology constraints. It should be capable of handling up to 100 concurrent users efficiently. The database should be optimized for quick lookups, ensuring that searches for matching credentials are performed using secure password hashing techniques and indexing methods to enhance performance.

All together :

All components of the login system must be seamlessly integrated to provide a complete and secure user experience. The frontend must feature a clean and responsive login page with proper form validation for email, password, and CAPTCHA input fields. The backend should effectively handle login requests, verify credentials, and manage user sessions while using secure password hashing techniques. The database should securely store user information, ensuring that passwords are hashed and fields are indexed for optimal search performance. The CAPTCHA system must be incorporated into the login process and always required for successful authentication to prevent automated attacks. The system must also include an account lockout mechanism that temporarily blocks a user's account after excessive failed login attempts, providing an additional layer of security. Finally, clear and specific error messages must be displayed whenever authentication issues arise, ensuring that users receive precise feedback if they enter invalid credentials, fail the CAPTCHA, or reach the account lockout threshold.

3. Buyer Profile & Settings - EDI

Description:

The system shall allow buyers to manage their profile, wishlist while tracking their loyalty and benefits through a badge system. Buyers will have access to a dedicated profile page where they can update personal information, add addresses for shipping and billing and choose the default address. Also they will be able to sync wishlists and merge their badge progress. The system shall ensure security, validation, and a seamless user experience, while supporting at least **10 concurrent users** and ensuring that all functionalities execute within **2 seconds**.

Functional Requirements:

1. Profile Management

The system shall provide a profile page with sections for personal details. Buyers shall have access to an **"Edit Profile"** section where they can update their personal details, including their full name, email address, phone number, password, and shipping and billing addresses(including postal code, which is made of 6 digits, city: any option from a list of given cities, and street: any option from a given list of cities). These changes shall be reflected immediately after saving to ensure that all transactions and communications use the most up-to-date information. The system shall validate **all profile inputs** based on **existing sign-up validation rules(from Sign Up)**, If incorrect information is entered, an error message shall be displayed next to the affected field.Profile updates shall be reflected in real-time. Buyers can manage separate **billing and shipping addresses** or opt to use the same address for both. Buyers can save multiple **shipping locations** (e.g., home, work) and set a default.

---

2. Wishlist Functionality

The system shall display a **"View Wishlist"** button within the buyer's profile, allowing quick access. The wishlist shall present a **clear and visually structured layout**, displaying product names, images, and prices. The system shall allow buyers to generate a **unique code** for their wishlist. This code can be shared with others so they can **view the buyer's wishlist** without needing direct access to the buyer's account. When the code is entered on a dedicated page, the system will retrieve the wishlist contents linked to that code, allowing the recipient to see the same list of products, including names, images, and prices, and take actions like adding items to their own cart. Each wishlist item shall have a **"Remove" button**, allowing buyers to delete individual products. Before removing a product, a **confirmation prompt** shall appear to prevent accidental deletion. Buyers shall be able to transfer items from the wishlist to the shopping cart using a **"Move to Cart" button**. If a product is successfully added to the cart, It shall be **automatically removed from the wishlist**. A confirmation message shall be displayed. If the product is **out of stock**, an error message shall be displayed, preventing it from being added to the cart. Wishlist updates shall be reflected in real-time. Buyers shall be able to **organize their wishlist using sorting and filtering options**: **Sort By:** Date Added (Newest/Oldest), Price (Low to High/High to Low), Popularity. **Filter By:** Category, Stock Availability, Discounted Items.

---

3. Badge System & Discounts

The system will feature a **badge system** designed to reward buyers based on their cumulative activity, specifically focusing on **Total Spending** and **Number of Purchases**. These metrics will be used to calculate a buyer's badge level using the formula: 0.4 * max(**Number of Purchases/40, 1)**.+ 0.6 * max(**Total Spending** /800,1), which will be updated dynamically based on ongoing engagement with the platform. There will be four **Badge Levels**, each offering progressively greater benefits and value: **Bronze(- a discount of 5%**, **Silver- a discount of 10%**, **Gold - a discount of 15%**, and **Platinum - a discount of 20%**. Each badge level will be awarded according to a **predefined formula** that takes these

factors into account, ensuring that buyers who engage more deeply with the platform receive higher-level badges. The result computed from the formula will determine which Badge the buyer owns: 0-0.25: Bronze, 0.25-0.5: Silver, 0.5-0.75-Gold, 0.75-1-Platinum. On the **Profile Page**, the badge system will be clearly visible in the right corner, displaying the following key information: **Current Badge Level**: The active badge the buyer has earned, with a clear indication of whether they hold a Bronze, Silver, Gold, or Platinum badge. **Progress Tracker**: A visual representation of the buyer's progress toward the next badge level. This tracker will provide: a numerical view of the buyer's current standing, An estimated prediction of the effort (spending, purchases) required to reach the next badge level, meaning the necessary values of **Total Spending** and **Number of Purchases in order to reach the next Badge**. Real-time updates based on the buyer's ongoing activity. The system will automatically display the discount in the buyer's profile section after each purchase, under the Badge Level. The discount details will be stored in the database.The percentage of the discount will be determined by the buyer's current badge level at the time of the purchase, ensuring that the rewards grow as the buyer's engagement with the system increases.

---

4. "Family Sync" Functionality

The system shall allow buyers to synchronize their profiles with other accounts sharing the same home address through a feature called **"Family Sync."** This functionality enables buyers to merge their wishlists and badge information into a single, unified profile. On the buyer's profile page, a permanent list will display all eligible accounts for synchronization, along with two buttons: **"Request Sync"** and **"Request Unsync."** The **"Request Unsync"** button will be disabled if the profiles are already merged.

When a buyer presses **"Request Sync,"** the other account owner will receive a notification upon their next login, displaying the requesting buyer's username along with two options: **"Accept"** and **"Deny."** Selecting **"Deny"** will cancel the request without merging the profile. Selecting **"Accept"** will complete the synchronization, merging both accounts.

This streamlined process ensures seamless profile integration while allowing users to retain control over their account synchronization. Once the sync is completed, the wishlist and badge progress from all synced accounts will be shared across the profiles. The buyer will have a consolidated wishlist, which includes all items saved across all synced profiles. The system will automatically detect and remove any duplicate products from the combined wishlist, ensuring no duplicates remain. Additionally, the system will calculate the highest badge level based on the cumulative activity across all synced profiles. The buyer's badge status and discount history will reflect the total activity from all linked profiles. After syncing, the system will display a confirmation message to the buyer, and the profile, wishlist, and badge progress will be updated immediately, ensuring a seamless transition. Buyers will be able to view their synced profiles and any pending sync requests within the permanent list in their profile. Syncing will ensure that all progress, whether related to the wishlist or badge levels, is merged into a unified experience.To ensure the process remains clear and user-friendly, the system will allow buyers to opt in or out of the **Family Sync** functionality at any time, using the 2 buttons mentioned above,. The option to unlink profiles will also be available, ensuring that buyers have control over their sync preferences. This feature will

provide a more unified shopping experience for families, while still allowing individual buyers to maintain control over their own account details and settings.

4. Seller profile + change settings - PATRI

Description:

The system shall allow sellers to manage their listed products and update their profile information. Sellers will have access to a dedicated dashboard where they can view and modify their product listings, edit their business information, and track their reliability through a **user score** metric. The system shall ensure security, validation, and a seamless user experience.

Functional Requirements:

- Implement a **dashboard** where sellers can view their listed products.
- Allow sellers to **edit product details**, including price, description, and stock quantity.
- Implement a **user score** **system** based on sales history, reviews, and reliability.
- Allow sellers to **update their business information** (e.g., store name, contact info).
- User should have a followers count (not a list, just a number).

1. Seller Dashboard & Product Management

The system shall provide a **dashboard** where sellers can view all their listed products with essential details (name, price, stock quantity). Each of these products can be clicked on to view the details. Sellers shall have options to **filter, search, and sort** their products based on name, category, availability or sort by price (ascending or descending). The system shall allow sellers to perform **actions**, such as updating stock levels or deleting listings. Sellers shall have the ability to add **new products**. Profile and product update operations shall **complete within 2 seconds** on a broadband connection (minimum 25 Mbps).

2. Profile & Business Information Updates

Sellers shall be able to update their **business profile** information, including: Store Name, Business Email, Business Phone Number, Business Address, Store Description. The system shall enforce **input validation**, ensuring:

  - The email follows a valid format (example@domain.com).
  - The phone number contains only digits and follows a valid format.
  - Required fields (store name, business email, business address) are not left empty.

If incorrect information is entered, an error message shall be displayed next to the affected field (“Invalid phone number format”, “Invalid email format”). The system *shall*

prevent duplicate business names within the platform. The system shall handle up to **10 concurrent sellers** updating their profiles without degradation in performance.

3. User Score System

Each seller shall have a **user score** based on all reviews of all products of a specific user. This is the average of all reviews that the seller received. For example, for a seller with product listings with ***x*** total reviews. When adding all scores together we get ***y.*** Therefore, the user score will be **z = *y* / *x***. The user score shall be a ***read-only field*** displayed on the seller's profile (format **z / 5**). The score shall be updated automatically based on predefined criteria.

4. Sales Analytics

The system shall provide sellers with an **analytics dashboard** displaying key business metrics, including:

- **Total revenue** over customisable time periods (daily, weekly, monthly).
- **Number of sales** per product.
- **Best-selling products**, ranked by sales volume.

The dashboard shall include a visual **linear graph** to represent trends over time, a field that displays total revenue (since the user registered) and revenue for the specific time period selected. Sellers shall be able to **export sales reports** in CSV format. The system shall allow filtering of sales data based on date range. Sellers shall receive **performance insights**, such as low-stock warnings.

5. Admin profile - COSMIN

Description:

The system shall provide an admin dashboard that enables administrators to efficiently manage users, sellers and new admins while ensuring security and access control. The dashboard will allow admins to create new admin accounts, enforce user bans or suspensions in cases of fraudulent activity, and generate reports on platform activity. To maintain security, an authentication system will be implemented, initially using hardcoded data until the login functionality is fully integrated. Middleware will restrict dashboard access based on roles, and API endpoints will be secured to ensure that only authorized admins can access sensitive data.

For user management, the system will include functionality to track banned users by storing their ban status in the database. The admin panel will provide a user interface for suspending users and preventing banned users from logging in. Additionally, admins will have the ability to reverse bans and restore user access when necessary.

The dashboard will also feature analytics and reporting capabilities, displaying insights through charts and graphs. It will provide a summary of total users, as well as a breakdown of buyers and sellers, enabling easy comparison of user types. The system will retrieve this data through database queries and dynamically visualize trends using pie charts and other graphical elements. The dashboard's operations will be optimized to ensure that all functionalities complete within two seconds on a broadband connection, maintaining a smooth user experience for administrators.

Tasks:

Authentication & Access Control

1. **Create Admin Role in Authentication System** – Use hardcoded data until login is implemented (Blocked).
2. **Implement Middleware to Restrict Dashboard Access** – Based on role.
3. **Secure API Endpoints for Admin Access** – Ensure only authorized admins can access sensitive data.

New Admin Account Management

4. **Investigate possible libraries for form display**
5. **Build UI Component to Display a Form for New Admin Account Creation**
6. **Replicate registering logic for admin**
7. **Connect the New Admin form to registration logic**

User Ban Management

8. **Add Database Field to Store Ban Status** – Track user ban state.
9. **Develop Admin UI to Suspend Users** – Enable suspension actions from the admin panel.
10. **Implement Logic to Prevent Banned Users from Logging In** – Depends on login UI/task.
11. **Create Option for Admins to Reverse Bans** – Allow restoring banned accounts.

Analytics & Reporting

12. **Choose a Charting Library for Analytics – Select a suitable charting solution.**
13. **Design UI Section to Display Total Users**
14. **Design UI Section to Display Buyer/Seller Counts** – Compare user types.
15. **Write Database Query to Retrieve Total Users**
16. **Write Database Query to Retrieve Total Buyers and Sellers Count -** Track user distribution.
17. **Implement Pie Chart for Total Buyers/Sellers** – Show user trends dynamically.

6. My Market - ARIS

Description:

The **"My Market"** feature enhances the user experience by allowing buyers to view detailed seller profiles, follow sellers, filter products, and sellers will receive notifications for reaching follower goals. The system efficiently handles buyer-seller relationships, supports real-time updates, and ensures users can manage their followed sellers seamlessly. This comprehensive, user-friendly, and efficient functionality offers an enjoyable shopping experience while maintaining system integrity and reliability.

Buyers can access a dedicated **Seller Profile** page that provides key information about a seller and their products. The profile includes the seller's name, logo, e-mail, address, phone number, number of followers and a list of available products. A prominently displayed **"Follow Seller"** button allows buyers to follow or unfollow the seller instantly, with changes being updated in real time in the database. The profile page is designed to align with the marketplace's UI guidelines, ensuring a clear and structured layout. Buyers can also filter and sort the seller's products within the profile page for easier navigation, and the profile loads within 2 seconds under standard network conditions to provide a seamless user experience.

Buyers have access to a dedicated section called **My Market**, which contains both the My Market Feed and a list of followed sellers alongside management tools for viewing profiles and unfollowing.

The **My Market Feed** displays products exclusively from followed sellers and is sorted based on the most recent updates, ensuring buyers receive relevant information. The feed remains consistently updated with the latest product offerings.

My Market allows users to search for a seller, view profiles of sellers by pressing the "View Profile" button and manage followed sellers by directly unfollowing them with the press of the button "Unfollow". Buyers can filter the feed to view products only from a specific seller by pressing the logo of the seller. Changes made in this section are immediately reflected in the database, ensuring real-time updates while maintaining a smooth and efficient user experience. The layout and interactions in My Market follow UI principles to provide intuitive navigation and seamless usability.

Sellers receive notifications related to their follower count, including alerts when they gain a new follower or reach a predefined milestone such as 100, 500, or 1,000 followers. These notifications appear when sellers access the notifications section within their account, keeping them informed about their follower growth in an organized manner. The notifications section is designed following UI standards to ensure clarity and ease of access.