POLITEHNICA UNIVERSITY OF BUCHAREST

SOFTWARE ENGINEERING

Marketplace Business plan

Team:

- Esmaeil Torabazari
- Ionut TECU

Coordinator:

- Prof. Phd. Costin-Anton Boiangiu

Last Date Modification:

- **Thu, 22 Jan , 2026**

Delivery Date:

- **Thu, 22 Jan , 2026**

Table of Contents

1. Requirements Analysis

1.1 Introduction and purpose

The purpose of this document is to provide a detailed display of the system requirements for building a web-based marketplace platform. The primary objective is to clearly outline the features and system functionalities needed to create a platform where users can register, log in, and manage content such as business listings. This platform will provide businesses and merchants with the opportunity to introduce their products and services, sell them, receive feedback from customers, communicate for support and warranties, retain customers, track real product/service reviews (not fake reviews), and more. This document serves as a shared reference for developers, designers, and stakeholders to ensure alignment on the system's goals, design, and functionality.

By defining these requirements, we are establishing a roadmap for building a competitive and future-ready business directory platform, which could also constitute the basis for new feature development that enhance the software product and provide value to users.

1.2 History

In the past, small shops were common in every town and city, offering a variety of products and services to local communities. These small businesses played an essential role in daily life, allowing customers to buy directly from producers and sellers they trusted. However, with the rise of hypermarkets and large retail chains, these small shops have slowly disappeared. Large supermarkets provide convenience and lower prices, but they have also made it difficult for independent businesses to survive.

As a result, many small shop owners were forced to close their stores and seek employment elsewhere, losing their independence and entrepreneurial opportunities. The shift towards centralized retail has reduced diversity in the market and limited consumer choices. While digital transformation has created new opportunities, small businesses often struggle to enter the online marketplace due to financial and technical barriers.

This research aims to address this problem by creating an online marketplace that empowers small producers and merchants to compete in the digital economy. By providing a structured and supportive platform, this marketplace will help small businesses establish an online presence, regain visibility, and connect with customers who value unique and locally sourced products.

1.3 Scope

The business marketplace will be a platform designed to help small producers and sellers enter the world of e-commerce. It will provide them with an accessible space to sell their products, connect with customers, and gain visibility. A business directory will be included as part of the marketplace, allowing sellers to list their businesses and be discovered easily by potential buyers.

Beyond just offering an online store, the marketplace will also act as a stepping stone for small businesses, equipping them with the necessary tools, knowledge, and support to establish their own independent e-commerce presence. After a certain period, sellers will have the option to transition from the marketplace to their own standalone online stores, ensuring long-term sustainability and growth. This way, the marketplace will not only serve as a business hub but also as an incubator for future independent e-commerce businesses.

1.4 Definitions, Acronyms, and Abbreviations

- **API (Application Programming Interface)**: A set of protocols, tools, and definitions that enable different software applications to communicate with each other. It outlines the methods and data formats for requesting and exchanging information,

facilitating integration between various systems, software components, or external services.

- **GDPR (General Data Protection Regulation)**: A data protection and privacy regulation enforced by the European Union (EU) to safeguard personal data. Enacted on May 25, 2018, it aims to give individuals within the EU greater control over their personal data while establishing strict guidelines for organizations globally on how to process and protect such data when interacting with EU citizens.
- **JS (JavaScript)**: A versatile, high-level programming language primarily used for creating dynamic and interactive elements on websites. As one of the three core web technologies (along with HTML and CSS), JavaScript powers features like animations, real-time form validation, interactive maps, and more.
- **PHP (Hypertext Preprocessor)**: A widely-used, open-source server-side scripting language designed for web development, although it can also be used for general-purpose programming. PHP is especially effective for building dynamic, interactive websites and web applications.
- **UI (User Interface)**: The part of a system or application where users interact with a machine or software. It includes the visual elements (buttons, icons, menus, etc.) and functional components that allow users to operate the system smoothly. The primary objective of UI design is to ensure an intuitive, efficient, and user-friendly experience.

- **PHP SDK (Software Development Kit)**: A collection of tools, libraries, documentation, and code samples that help developers build software applications using PHP. It provides an interface to interact with a specific platform, service, or API. For eCommerce platforms, a PHP SDK typically enables developers to integrate third-party services (such as payment gateways, shipping providers, or analytics tools) with the platform's core functionality.

- **Plugin**: A software component that adds specific functionality to an existing software application. In the context of eCommerce platforms, plugins are used to extend or enhance the platform's features without altering the core code. For example, plugins

can add payment gateways, shipping options, SEO tools, or security enhancements to platforms like WordPress, PrestaShop, OpenCart, and Magento.

1.5 References

- No external legislation or additional references apply to the project.

1.6 Structure

The document is organized as follows:

- **Introduction:** serves as a foundational section, providing context, explaining the purpose of the document, and offering background information and relevant definitions.
- **General description**: provides a clear understanding of the product's purpose, the needs it addresses, and the boundaries within which it must operate.
- **System requirements**: define the specifications needed to ensure proper operation, encompassing external interfaces, core functionality, and performance standards, while ensuring the system achieves its objectives and aligns with user expectations and technical constraints.
- **Appendices**: designed to offer all necessary supporting details and provide supplementary information, including materials, references, and detailed explanations that enhance the understanding of the document's content.

1.7 Methodology

This project adopts a **structured systems engineering methodology,** combining principles from Requirements Engineering with a phased development approach, guided by the core logic of the V-Model — but adapted for practicality and flexibility based on the application's scope.

The proposed system is a **business-to-customer** (B2C) e-commerce marketplace tailored for small producers and sellers. Given the platform's evolving nature — from a centralized marketplace to enabling independent seller stores — the development approach must ensure clarity in requirements, progressive implementation, and **support for scalability**.

Key Methodology Elements:

- **Requirements Engineering** (RE) practices were used in the early phases to define the functional and non-functional needs of all users — including customers, sellers, and platform administrators.
    - These include features like product listings, business directory search, order handling, user management, and notifications.
    - RE ensured the scope was aligned with both the marketplace stage and the **seller graduation** phase (when sellers move to their own stores).
- A **phased lifecycle approach**, inspired by the V-Model, was used to:
    - Define clear relationships between requirements, design components, and testing procedures.
    - Support **validation** at each level: e.g., ensuring that marketplace registration works before introducing seller dashboards or APIs.
- While the V-Model structure informs the design and verification thinking, the project allows room for **Agile-like feedback cycles** — especially for user-facing components like UI workflows, marketplace onboarding, and feature prioritization. This ensures the solution stays user-focused and adaptable.

In this hybrid approach:

- **Structured documentation** supports long-term maintainability.
- **Iterative validation** helps refine seller onboarding, business directory structure, and independent store transition.
- **Scalability and modular design** ensure the system can grow alongside sellers and their businesses.

Ultimately, the chosen methodology reflects the **dual role of the application**:

1. As a **functional marketplace**, supporting daily commerce.
2. As an **incubator platform**, helping small businesses evolve into independent e-commerce entities.

2. General Description

2.1 Product Description

The marketplace platform offers a comprehensive ecosystem for small businesses to engage with customers, sell products online, and grow their digital presence. It includes a fully interactive website that serves as the central hub for both merchants and customers. The platform integrates directly with merchants' websites, ensuring seamless synchronization of product data, inventory, and orders in real-time.

In addition to the web interface, a mobile application will be available to enhance user experience. This mobile app communicates with the marketplace's API, allowing customers to browse products, make purchases, and interact with the marketplace features on the go.

A key feature of the marketplace is its ability to provide merchants with data-driven recommendations based on machine learning algorithms. These recommendations will highlight best-selling products and offer insights to improve sales strategies. The platform will also generate reports to help merchants analyze business performance, identify trends, and make informed decisions.

For customers, the marketplace will incorporate blockchain technology to ensure transparency, security, and authenticity in reviews and feedback. Blockchain will help safeguard customer opinions and protect against fraudulent reviews, making feedback more trustworthy for both merchants and customers.

This product combines a dynamic e-commerce experience with advanced technologies such as machine learning and blockchain, offering merchants the tools they need to succeed while providing customers with a secure and personalized shopping experience.

The finalized product will include the following key components and interfaces, designed to deliver a seamless and efficient user experience:

User Interface (UI): The UI is designed to ensure ease of use, efficient navigation, and a visually appealing experience for diverse users. It provides web pages for registration, login, as well as filtering and search options that enables users quickly narrow down their choices and find relevant businesses within their community. To build credibility and transparency, the UI provides an effective feedback loop, allowing users to generate reviews and ratings. This system helps consumers make informed decisions by offering insights into the experiences of others.

External System Interfaces: To ensure seamless operation across various systems, the platform will include External System Interfaces that allow the marketplace to interact with external tools and services. These interfaces will integrate with third-party systems, such as merchant websites, payment gateways, and inventory management solutions, ensuring that product data and transactions are synchronized in real-time. This integration will streamline business processes for merchants and provide a smooth experience for customers, ensuring that all data is up to date and that orders are processed efficiently.

User Interaction: will focus on ensuring easy engagement for both customers and merchants. Customers will enjoy intuitive navigation, product recommendations, wishlists, and personalized offers. Merchants will manage their stores, track inventory, and access sales analytics through dedicated dashboards. The goal is to foster positive engagement and enhance the experience for both buyers and sellers.

Buyer Interaction: For Buyer Interaction, the marketplace will provide an engaging and secure environment for customers to explore, purchase, and provide feedback on products. Customers will be able to leave reviews and ratings on products, interact with merchants through messaging features, and receive product recommendations based on their browsing behavior. Blockchain technology will be used to secure customer reviews, ensuring that they are authentic,

tamper-proof, and transparent. This transparency will increase trust between customers and merchants and help build a reputation system that benefits both parties.

Merchant Interaction : Merchant Interaction with the platform will be facilitated through a robust Merchant Dashboard. This dashboard will allow sellers to upload products, track inventory, view order histories, and analyze sales data. Merchants will also be able to access data-driven insights powered by machine learning algorithms, which will recommend top-selling products, predict demand, and offer insights into customer behavior. This will help merchants make informed decisions, optimize their sales strategies, and grow their businesses. Additionally, the dashboard will feature tools for managing promotions, discounts, and advertising, helping merchants attract more customers.

Backend APIs: The platform will rely on Backend APIs to facilitate communication between the marketplace website, mobile app, and external systems. These APIs will ensure real-time data synchronization, allowing product information, pricing, and order details to be updated instantly across all platforms. The APIs will also enable secure transactions, data retrieval for reports, and smooth integration with external tools. For example, when a customer places an order, the API will instantly update the inventory, notify the merchant, and initiate the payment process, ensuring that the entire system runs consistently.

Analytics and Reporting Tools: To empower merchants to make data-driven decisions, the platform will include **Analytics and Reporting Tools**. These tools will allow merchants to track key performance indicators such as sales trends, customer behavior, and product performance. Merchants will be able to view detailed reports on their sales, returns, and customer feedback, as well as compare performance across different periods. These insights will help merchants optimize their product listings, improve their marketing efforts, and maximize profitability.

Security Features: Ensuring the Security of users and transactions will be a top priority. The platform will implement various security measures, including encrypted transactions, secure login protocols, and two-factor authentication (2FA) to protect user accounts and sensitive information. Blockchain technology will be used to secure reviews and feedback, ensuring that they cannot be altered or manipulated. This will enhance the trust and transparency between customers and merchants, as well as protect the integrity of the platform. Additionally, regular security audits and monitoring will be conducted to detect and mitigate potential vulnerabilities.

2.2 Product Functions

The online marketplace integrates a business directory as one of its central components, but also offers a wide range of other features that enhance the experience for customers, business owners, and administrators. The marketplace is designed to allow users to discover businesses, leave feedback, engage with services, and purchase products, while enabling business owners to manage their presence, attract customers, and grow their sales. Administrators have tools to moderate content, ensure security, and monitor platform performance for a seamless, secure, and efficient experience.

The marketplace platform provides the following key functions:

- **User Accounts and Personalization**: Users can create accounts to track their activity, save favorite businesses, write reviews, and manage applied offers. Users can also view and edit their review history, manage preferences, and receive personalized product recommendations based on past activity and browsing behavior.

- **Business Listing Management** : Business owners can create and manage detailed business profiles, including essential information like business name, contact information, services offered, hours of operation, and location. Business owners can also upload product listings, post offers, discounts, or promotions, and set up seasonal or special event-based deals that users can view and redeem.

  .

- **Search with Filters** : Users can search for businesses and products with an array of customizable filters such as categories, price ranges, ratings, location, availability, and more. This helps users narrow down their options to find the best matches based on their needs and preferences.
- **View Business Details and Contact Information:** Each business profile includes detailed information, including descriptions, contact details (phone, email, website), social media links, location, and working hours. This functionality ensures that users can

quickly access all the information they need to contact and engage with businesses directly.

- **User Management via API:** The API allows for the programmatic creation, editing, and deletion of user accounts. This makes it easier for both users and administrators to manage user profiles, ensuring consistency and efficiency in handling user data.

- **Company Management via API** : Similar to user management, the API enables the programmatic creation, editing, and deletion of business profiles. This feature streamlines the management of business listings, ensuring that all company-related data is accurate, up-to-date, and easily accessible.

- **Search by Filters via API :** The API allows users to search for businesses and products based on specific filters such as category, price, location, ratings, etc. It enables developers to retrieve relevant and precise data quickly, enhancing system performance and improving the user experience.

- **View Business Details by API** : Through the API, applications can access up-to-date business details such as name, address, phone number, email, website, and operating hours. This ensures that users always receive the most accurate and relevant information.

- **Product Recommendations and Personalization**: The platform will use machine learning algorithms to offer personalized product recommendations for users. Based on user behavior, preferences, and past purchases, the system will recommend products that match the user's interests, enhancing the user experience and boosting sales for merchants.

- **Customer Reviews and Feedback (Blockchain Integration) :** Customers can leave reviews and feedback for businesses, products, and services. To ensure transparency and trust, reviews will be stored using blockchain technology, making them tamper-proof and

verifiable. This ensures that customers can rely on authentic reviews, building credibility for the marketplace.

- **Order Management and Tracking:** Users can place orders for products listed by businesses, and both customers and business owners will have access to real-time order tracking. Merchants will be able to manage and update order statuses, while customers can track the shipping and delivery progress of their purchases.

- **Mobile Application:** In addition to the web platform, the marketplace will offer a mobile application that allows users to access the marketplace on-the-go. The app will synchronize with the marketplace API, providing the same functionalities, including browsing products, viewing business listings, placing orders, writing reviews, and managing account details, directly from their smartphones.

- **Promotions and Discounts:** Business owners will be able to create and manage special promotions, discounts, and seasonal offers. These can be targeted to specific customer segments or time frames, helping businesses increase sales and attract new customers. Customers will be notified of these promotions through push notifications or email alerts.

- **Admin Dashboard and Content Moderation :** Administrators will have access to a robust admin dashboard that allows them to oversee platform activity. This includes content moderation tools to review and approve new business listings, promotions, and customer reviews. Admins will also monitor user activity for compliance with platform policies and address any inappropriate content.

- **Analytics and Reporting Tools:** Merchants will have access to detailed analytics tools that provide insights into sales trends, customer demographics, and product performance. These reports will help business owners optimize their strategies, identify top-selling products, and refine marketing efforts to drive growth.

- **Payment Gateway Integration:** The platform will support secure payment processing through integrated payment gateways. Users will be able to make secure purchases with various payment options, including credit cards, digital wallets, and bank transfers. The payment system will ensure secure transactions, and merchants will be able to track payments through their dashboards.

- **Delivery:** The marketplace offers a seamless delivery process, where merchants are responsible for dispatching orders. Customers can track their shipments in real-time and receive notifications at key stages, such as dispatch, out for delivery, and successful delivery. Various delivery methods, including standard, express, and local pick-up, are available depending on the merchant's preferences, ensuring a reliable and efficient service for customers.

- **Security and Privacy Features :** The platform will implement robust security measures, including encrypted data transfer, secure payment processing, two-factor authentication (2FA), and regular system audits to protect both user and business data. Blockchain technology will be employed to ensure the integrity of reviews and feedback, providing transparency and trust within the marketplace.

These features ensure that the marketplace is comprehensive, offering a smooth and seamless experience for both users and merchants. The system is designed to be accessible for users with varying levels of technical expertise while maintaining the flexibility and functionality needed for advanced users and administrators.

2.3 User Description

The platform is designed to serve a variety of users, each with distinct roles and needs:

- **Business Owners -** Business owners are the core participants, using the platform to promote their businesses, manage their online presence, and connect with potential customers. They can create and update business profiles, post offers and promotions, track customer interactions, and manage product listings. Business owners also have access to analytics tools to monitor their sales performance and make data-driven decisions. Additionally, they can respond to customer reviews, handle orders, and customize their storefronts to enhance engagement.

- **General Users (Customers)** : Customers are individuals looking to discover and interact with businesses that meet their needs. They may be searching for services, products, or professionals in a specific location or industry. Customers can browse business listings, filter search results, read reviews, and engage directly with business owners. They can also save favorite businesses, track applied offers, leave feedback, and make purchases directly through the platform. The platform provides a personalized experience by offering product recommendations based on customer behavior and preferences.

- **Administrators:** Administrators are responsible for overseeing the platform to ensure it operates smoothly, securely, and efficiently. They monitor user activities, moderate content such as business listings and customer reviews, and enforce platform policies. Administrators also manage security protocols, user access, and ensure that the platform's infrastructure is maintained. They generate performance reports and work to continuously improve the platform's user experience and performance.

The platform's diverse functionalities cater to the unique needs of each user group. It enables seamless interaction between customers and business owners, empowers business owners with the tools to grow their online presence, and provides administrators with control and oversight to ensure a smooth, secure, and efficient environment for all users.

2.4.1 Constraints

While building and maintaining the online marketplace, there are several constraints and challenges that need to be addressed to ensure the platform's efficiency, security, and scalability. These challenges include technical, operational, and legal aspects that need careful attention:

- **Hardware Limitations**

  The platform must be accessible across a wide range of devices, including desktops, laptops, tablets, and smartphones, to ensure it reaches a broad user base. The system should be optimized to run smoothly on devices with varying levels of processing power and memory. Compatibility with major web browsers, including older versions, is essential for supporting users with diverse setups. On the server side, the infrastructure

must be capable of handling concurrent requests efficiently while implementing backup and recovery systems to prevent data loss.

- **Performance Requirements**

  The platform must deliver a responsive user experience. Tasks like searching, profile registration, and updating listings should complete within two to three seconds under normal conditions. The system needs to support up to 1,000 users simultaneously, with scalability to accommodate up to 10,000 users as demand grows. To maintain optimal performance as the data expands, efficient database indexing, caching, and query optimization techniques are necessary.

- **Compliance with Legislation**

  The platform must comply with privacy and data protection regulations, including the General Data Protection Regulation (GDPR). Personal data must be securely encrypted during transmission (via HTTPS) and storage. Users should have control over their data, including the ability to view, edit, and delete their personal information. Strict data retention policies must be in place to ensure personal data is not stored longer than necessary for the platform's purposes.

- **External Integrations**

  The system will rely on third-party services for functionalities such as authentication, cloud storage, payment processing, and mapping. These integrations must be secure and reliable, with fallback mechanisms in place in case of service failures. Furthermore, all third-party services used must comply with data privacy standards to ensure the responsible handling of user data.

- **Security Standards**

  Protecting user data is paramount for maintaining trust in the platform. The system should use secure authentication methods like JSON Web Tokens (JWT) and hash passwords using algorithms like bcrypt. All data exchanges must be encrypted via HTTPS, and the platform should guard against threats such as SQL injection, cross-site

scripting (XSS), and cross-site request forgery (CSRF). Regular security audits should be conducted to identify and fix any vulnerabilities in the system.

- **User Scalability**

  As the platform grows, it must scale to accommodate an increasing number of users and business data entries. Cloud-based hosting will provide the flexibility to scale resources as needed. Load balancing should be implemented to distribute user traffic across multiple servers, especially during peak times. Additionally, the database should be able to handle large datasets efficiently through methods like sharding or replication to ensure continued speed and reliability.

- **Deployment Constraints**

  The platform must be deployed on reliable servers that support the necessary technologies, such as PHP, MySQL, and Apache. The hosting environment should include automated backups to safeguard against data loss, and a disaster recovery plan should be in place to restore the system quickly in case of failure. The deployment process should prioritize reliability, security, and ease of ongoing maintenance.

- **Interface Constraints**

  The platform's user interface (UI) must be responsive, accessible, and intuitive. The design should work seamlessly across various screen sizes, from desktop monitors to mobile devices, to ensure usability for all users. The platform should adhere to the Web Content Accessibility Guidelines (WCAG), supporting users with disabilities through features like screen reader compatibility and keyboard navigation. The interface should be clean and modern, compatible with major browsers, to provide a smooth and engaging experience for users.

These constraints and challenges must be carefully considered throughout the development and deployment of the platform to ensure it meets the expectations of users and operates efficiently in a secure and scalable environment.

2.4.2 Challenges

While the platform is designed to overcome many barriers for small businesses, there are several challenges that will need to be addressed during its development, launch, and ongoing operation. These challenges span various technical, operational, and market factors that could impact the success of the platform.

User Adoption: One of the key challenges is ensuring both business owners and customers adopt and effectively use the platform. Many small business owners may be hesitant to transition from traditional storefronts to e-commerce, especially if they lack technical expertise. Effective onboarding, user training, and continuous support will be crucial to ensuring high adoption rates.

Market Competition: The online marketplace space is highly competitive, with numerous established players already serving both customers and businesses. The challenge will be to differentiate the platform by offering unique value propositions, such as tailored business support, robust analytics, or a seamless user experience. Establishing trust and building a user base will require a solid marketing strategy and continuous improvements.

Data Privacy and Security: Ensuring the protection of sensitive user data is one of the greatest challenges, especially when handling personal, payment, and business information. Regular security audits, implementing strong encryption methods, and staying updated on data protection regulations (such as GDPR) will be essential to maintaining a secure platform and building trust with users.

Scalability and Performance: As user demand grows, ensuring that the platform can scale effectively without compromising performance will be a major challenge. The system must be able to handle increasing traffic, more product listings, and growing data volumes without slowdowns or crashes. Load balancing, database optimization, and cloud-based solutions will be essential to managing these growing demands.

Cross-Platform Consistency: Delivering a consistent experience across web and mobile platforms can be challenging, as different devices and screen sizes may present layout, performance, and usability issues. Achieving uniform functionality and design while optimizing performance across platforms will require careful planning and testing.

Third-Party Integrations: Reliance on third-party services for authentication, payments, mapping, and other critical functionalities poses challenges in terms of reliability and data security. If any third-party service experiences issues, it could affect the entire platform's operation. Ensuring these integrations work smoothly and have proper backup plans will be crucial to avoid service disruptions.

Continuous Updates and Maintenance: To keep the platform running smoothly and securely, ongoing maintenance and regular updates are essential. This includes bug fixes, performance enhancements, and new feature releases. However, these updates must be managed carefully to avoid downtime or negative impacts on user experience.

User Support and Engagement: Maintaining high levels of customer and merchant support is vital to ensuring satisfaction. Providing quick, helpful responses to user issues and feedback, and offering educational resources to assist businesses in using the platform effectively, are ongoing challenges that require dedicated attention.

2.5 Assumptions and Dependencies

This section outlines the key assumptions and external dependencies considered during the development of the business directory platform. These elements are essential to ensure the platform's functionality, reliability, and successful deployment.

- **User Access Assumptions**
  It is assumed that users will access the platform via modern web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. Users are expected to have a basic understanding of web applications and a stable internet connection to ensure smooth interaction with the platform. Business owners and administrators are assumed to have access to desktops, laptops, or tablets for managing their business profiles

- **Hosting Environment Assumptions**
  The platform assumes that the hosting environment will support key technologies such as PHP, MySQL, and web servers like Apache or Nginx. Hosting providers should ensure reliable uptime, adequate server resources (CPU, RAM, and storage), and automated backup solutions to guarantee system stability and protect user data.

- **Third-Party Service Dependencies**
  Several third-party services are integral to the platform's core functionality. These include authentication services like OAuth for social login, cloud storage solutions such as Google Cloud Storage for managing user-uploaded files, and mapping services like Google Maps API for location-based searches. The platform’s performance depends on these services remaining secure and operational, and disruptions to these services could affect user experience.
- **Database Dependencies**
  The platform relies on MySQL for storing business listings, user profiles, transactions, and analytics. Regular maintenance, indexing, and optimization ensure performance, while sharding or replication supports scalability.

- Additionally, Google Cloud Datastore will handle real-time marketplace functions like product recommendations and machine learning insights. This hybrid approach ensures MySQL provides transactional reliability, while Google Datastore enhances scalability for dynamic data processing.

- **Internet Connectivity Dependencies**
  A stable and reliable internet connection is required for both users and the platform's server infrastructure. The platform relies on constant connectivity for real-time interactions, such as searching, registration, and updating profiles. On the server side, the hosting provider must ensure high network availability to minimize service disruptions.

- **Legal Compliance Dependencies**
  The platform must adhere to the General Data Protection Regulation (GDPR) and other relevant privacy laws. This includes secure data handling practices, such as encrypting user data during transmission and storage. Ensuring compliance with these regulations is critical to protecting user privacy, avoiding legal penalties, and maintaining trust with users.

3. System Requirements

3.1 External Interface Requirements

The marketplace platform provides an intuitive and secure interface for different user roles:

- Business Owners: Register businesses, create listings, manage profiles, and interact with customers.
- Customers: Search for businesses, browse products or services, leave reviews, and complete transactions.
- Administrators: Monitor platform activity, moderate content, and manage disputes.

3.1.1 User Interfaces

- Registration & Login: Secure sign-up via email or OAuth authentication.
- Business Profile Management: Create, update, and manage business listings.
- Search & Filter System: Search using keywords, location, category, and rating filters.
- Marketplace Transactions: Customers can purchase services or products, and business owners can process orders.
- Admin Dashboard: Tools for user management, content moderation, and analytics.

3.1.2 Hardware Interfaces

- Client Devices: Supports desktops, tablets, and smartphones with modern web browsers.
- Server Infrastructure: Scalable cloud-based hosting with automated backups.

3.1.3 Software Interfaces

- Frontend: HTML, CSS, JavaScript (React or Vue.js for a dynamic user experience).
- Backend: PHP with Laravel or Node.js for efficient data handling.
- Database: MySQL for structured data, Google Datastore for high-speed marketplace transactions.
- APIs & Integrations:
    - OAuth authentication for secure user logins.
    - Google Maps API for location-based business searches.
    - Payment Gateways (Stripe, PayPal) for secure transactions.
    - Cloud Storage (Amazon S3 or Google Cloud) for business assets.

3.1.4 Communication Interfaces

- HTTPS Encryption: All data transmissions are secured with SSL/TLS.
- RESTful APIs: Data exchange between frontend and backend using JSON format.

3.2 Functional Requirements

The platform offers key functionalities for seamless business and customer interactions:

3.2.1 User Registration & Authentication

- Secure user registration with email verification and password encryption.
- Multi-role authentication (business owners, customers, admins).

3.2.2 Business Listings & Management

- Businesses can create, update, and delete listings.
- Upload images, videos, and business details (location, services, hours).

3.2.3 Marketplace Transactions

- Businesses can list products and services with pricing.
- Customers can purchase or book services through secure payment integration.

3.2.4 Search & Filtering System

- Search businesses by keywords, categories, ratings, and location.
- AI-powered recommendations for relevant businesses.

3.2.5 User Engagement & Reviews

- Customers can rate and review businesses.
- Business owners can respond to reviews and engage with customers.

3.2.6 Payment & Delivery System

- Secure checkout process with multiple payment gateways.
- Integration with third-party delivery services for logistics.

3.2.7 Admin Dashboard & Moderation Tools

- Manage business listings, user accounts, and reported content.
- Monitor transactions and platform analytics.

3.3 Performance Requirements

3.3.1 Response Time

- Search queries & data retrieval: ≤ 2 seconds.
- Page loads & profile updates: ≤ 3 seconds.
- Transactions & payments: ≤ 3 seconds.

3.3.2 Scalability & Reliability

- Handles 1,000 concurrent users, scalable up to 10,000 users.
- Cloud-based hosting with auto-scaling & load balancing.
- 99.9% uptime with automated backups for disaster recovery.

3.4 Design Constraints

3.4.1 Technology Stack

- The platform must be built using web technologies, with:
  - Frontend: React (or another modern JavaScript framework)
  - Backend: PHP. or Golang
  - Database: MySql

- APIs must follow RESTful principles and be documented using OpenAPI specifications.
- The platform must be cloud-deployable (e.g., AWS, Azure, or similar).

3.4.2 Deployment & Scalability

- The system must support containerized deployment (e.g., Docker) and be ready for horizontal scaling.

- It should be compatible with CI/CD pipelines for continuous integration and deployment.

3.4.3 Legal and Regulatory Compliance

- Must follow applicable data protection laws (e.g., GDPR) regarding user consent, data access, and deletion.

- If payments are involved, the system must integrate with PCI DSS-compliant payment gateways.

3.4.4 Localization

- The platform must support multi-language content for marketplace expansion (e.g., English and local language).
- Date, currency, and formatting must be region-aware.

3.4.5 Third-Party Dependencies

- The system must integrate with:
  - External email/SMS services for notifications.
  - Payment providers (e.g., Stripe, NETOPIA).
  - Possibly map/geolocation APIs for business directory search.

3.5 Software System Attributes

3.5.1 Reliability

- The system must maintain consistent availability, targeting 99.9% uptime.
- Failures (e.g., network issues, server downtime) must be logged and handled gracefully.
- Critical actions (e.g., placing orders, updating profiles) must be transaction-safe and recoverable.

3.5.2 Availability

- The system must be available 24/7, with automatic restart/recovery in case of failure.

- A load-balanced setup must be planned to maintain service availability under growing traffic.

3.5.3 Maintainability

- Codebase must be modular, documented, and follow clean architecture principles to simplify bug fixing and updates.
- System updates should be deployable with zero downtime where possible.
- Version control must be enforced (e.g., using Git).

3.5.4 Scalability

- The system must support growth in:
  - Number of users (customers and sellers)
  - Volume of products and transactions
  - Number of independent seller stores

- Database and backend must allow for horizontal scaling as needed.

3.6 Security & Compliance

The business marketplace platform must ensure robust security measures to protect sensitive data, user identities, and transactional integrity. Given that the system handles customer information, seller business data, and financial transactions, the following security requirements are mandatory

3.6.1 Data Protection

- End-to-end encryption (HTTPS, SSL/TLS).
- GDPR compliance: Users can access, modify, and delete personal data.
- JWT authentication for secure API requests.

3.6.2 Threat Protection

- Protection against SQL injection, XSS, and CSRF attacks.
- Secure password hashing.

3.6.3 Authentication & Authorization

- The system must enforce secure **user authentication**, supporting:
  - Email/password login with hashed credentials.
  - (Optional) Two-Factor Authentication (2FA) for admin or seller accounts.

- Access to platform features should be **role-based**, with distinct access levels for:
    - Admins
    - Sellers
    - Customers

- API endpoints must enforce **token-based access control** (e.g., JWT or OAuth2).

3.6.4 Secure Communication

- All communication between users, services, and the database must occur over **HTTPS/TLS**.
- The platform should **reject unencrypted (HTTP) requests**.

3.6.5 Secure APIs

- All API endpoints must include:
    - **Authentication tokens**
    - **Rate limiting** to prevent abuse
    - **CORS configuration** to control access

- Public APIs (e.g., business directory search) must have **read-only permissions**.

3.7.5 Input Validation & Protection Against Attacks

- The system must validate and sanitize all inputs to prevent:
    - SQL Injection
    - Cross-Site Scripting (XSS)
    - Cross-Site Request Forgery (CSRF)

- Uploads (e.g., product images) must be scanned and size-limited.

3.7.6 Session Management

- Sessions must expire after a defined period of inactivity.
- Session tokens must be **stored securely** (e.g., HTTP-only cookies).

3.7.7 Logging & Monitoring

- The platform must log key activities (logins, failed attempts, product updates).

- Security-related events must be monitored and auditable for forensic purposes.

3.7.8 Administrative Controls

- Admin dashboards must be accessible only to authorized personnel.
- Admin-level changes (e.g., banning users, editing listings) must trigger audit logs.

4. Design Constraints

4.1 Technology Constraints

- Backend: PHP (Laravel) & Go Lang (Node.js if is necessary).
- Frontend: React or Vue.js.
- Database: MySQL (primary), Google Datastore (for fast transactions).
- Web Server: Apache or Nginx.

4.2 Compliance Constraints

- GDPR-compliant data handling.
- Adherence to accessibility standards (WCAG).

5. Software System Attributes

5.1 Reliability & Availability

- 99.9% uptime guarantee.
- Automated failover mechanisms.

5.2 Maintainability & Scalability

- Modular codebase with API-driven architecture.
- Scalable storage and computing resources.

5.3 Usability Attributes

The system must offer a high degree of usability to support diverse user groups, including small business owners, end customers, and platform administrators

5.3.1 Ease of Learning

- New users should be able to **register, list products, and manage their accounts** with minimal training or support.

- Sellers should be guided through onboarding using:
    - Tooltips
    - Setup wizards
    - Tutorials or guided tours

5.3.2 User Interface Clarity

- The platform must have a **clean and modern UI**, following common usability standards
- Icons, buttons, and text labels must be:
    - **Clearly visible**
    - **Self-explanatory**
    - **Consistently placed**
- Use of **visual hierarchy** and **responsive design** is mandatory.

5.3.3 Accessibility

- The system should be **usable by people with disabilities**, following **WCAG 2.1** Level AA guidelines.
- Features must include:
    - Keyboard navigation
    - Text alternatives for images
    - Color contrast for readability

5.3.4 Multilingual Support

- The interface must support at least **two languages** initially (English and local language).
- Language selection should be user-configurable and persist across sessions.

5.3.5 Consistency Across Devices

- The system must provide a **responsive design** that works on:
    - Desktops
    - Tablets
    - Smartphones
- Navigation and layout must remain **functionally consistent** across devices.

5.3.6 User Feedback & Error Messages

- The system must provide clear and helpful **error and success messages**, using plain language.
- Forms must include:
  - Real-time validation (e.g., “password too short”)
  - Highlighting of missing/incorrect fields

- Users should **never be left guessing** what went wrong.

5.3.7 User Support Tools

- Include access to:
  - Help center or FAQ
  - Contact support form
  - Chatbot (optional for future version)

- Admins must be able to respond to user inquiries through a **moderation dashboard**.

6. Enterprise Architecture Mapping

The business marketplace platform is not just a web application—it is an integrated system supporting a wide range of **business, information, application, and technology** functions. This section provides a high-level mapping of the system using the **Enterprise Architecture (EA) framework**, ensuring alignment between business goals and IT capabilities.

The goal is to ensure that the platform is **strategically structured**, **scalable**, and **capable of evolving** alongside the businesses it supports.

6.1 Overview of EA Domains

The architecture follows the **four-layer enterprise architecture model**, reflecting both **technical and business** concerns:

6.1.1 Business Architecture

- **Purpose**: Support small producers entering e-commerce and empower them to become independent sellers.

- **Core Business Functions**:
  - Seller registration and onboarding

- Product listing and sales
- Customer account management
- Order handling and payments
- Business directory for discoverability
- Admin and moderation functions

- **Stakeholders**:
  - Marketplace admin
  - Sellers (small businesses)
  - End customers
  - Technical operations team

6.1.2 Information/Data Architecture

- **Key Data Entities**:
  - Users (sellers, customers, admins)
  - Products and Categories
  - Orders & Transactions
  - Business profiles (directory)

- **Data Models**:
  - Relational model for structured data (e.g., users, orders)
  - Document-oriented APIs using JSON/OpenAPI

- **Considerations**:
  - Data validation and consistency
  - Privacy controls (user data visibility)
  - GDPR-ready architecture

6.1.3 Application Architecture

- **Core Applications**:
  - Web-based user interfaces (for customers and sellers)
  - Admin dashboard
  - RESTful APIs for modular interaction

- **Application Components**:
  - Authentication service
  - Product management module
  - Order & payment service
  - Notification system
  - Business directory engine

- **Architecture Style**: Modular, service-oriented (not full SOA, but components are decoupled for maintainability)

6.1.4 Technology Architecture

- **Frontend**: React (SPA), Tailwind UI, i18n library
- **Backend**: Node.js (Express) or Django REST Framework
- **Database**: PostgreSQL for structured data; optional Redis or MongoDB for caching/session handling
- **APIs**: OpenAPI-standard REST APIs with secure access
- **Deployment**:
  - Dockerized services
  - Cloud-hosted (e.g., AWS, Azure, or DigitalOcean)
  - CI/CD pipelines for agile delivery

- **Security Infrastructure**:

  - HTTPS, JWT for auth
  - Role-based access control
  - Monitoring and logging tools (e.g., Prometheus, ELK)

6.2 Alignment with Business Vision

- Technical design **enables growth**, allowing sellers to graduate to standalone shops.
- Modularity supports **scalability** and **future integration** (e.g., third-party logistics, analytics).
- The system can evolve from a **marketplace hub** into an **e-commerce incubator**.
- Platform design supports **enterprise-wide visibility**, governance, and customer trust.

7. Risk Analysis

Effective system development must include identification and mitigation of risks that could impact the success of the platform. This risk analysis focuses on **technical**, **operational**, and **business** aspects, considering the specific context of a business marketplace platform for small producers.

Each risk is categorized by **type**, **probability**, **impact**, and potential **mitigation strategy**.

7.1 Risk Categories and Mitigation Plans

R1 – Data breach or security vulnerability

- **Category:** Technical / Security
- **Probability:** Medium
- **Impact:** High
- **Mitigation Strategies:**
  - Use HTTPS and token-based auth (JWT)
  - Encrypt sensitive data at rest and in transit
  - Conduct regular security audits and penetration testing

R2 – System downtime or availability issues

- **Category:** Technical / Operational
- **Probability:** Medium
- **Impact:** High
- **Mitigation Strategies:**
  - Deploy on cloud with auto-scaling
  - Implement monitoring & alerting tools (e.g., Prometheus, ELK)
  - Set up automated backups and failover systems

R3 – Critical bugs in payment or checkout process

- **Category:** Functional / Technical
- **Probability:** High
- **Impact:** High
- **Mitigation Strategies:**
  - Write unit and integration tests for key components
  - Implement CI/CD pipeline with automated testing
  - Prepare rollback strategies for failed deployments

R4 – Low adoption by small sellers

- **Category:** Business / Strategic
- **Probability:** Medium
- **Impact:** High
- **Mitigation Strategies:**
  - Offer a simple onboarding process
  - Provide tutorials, support, and community guidance
  - Use surveys and interviews to identify barriers and iterate

R5 – Fraudulent behavior or abuse by users

- **Category:** Operational / Security
- **Probability:** Medium
- **Impact:** Medium
- **Mitigation Strategies:**

- Implement user behavior monitoring
- Use role-based permissions and moderation tools
- Add a reporting/flagging system for user-generated content

R6 – Changing project requirements or scope creep

- **Category:** Business / Management
- **Probability:** High
- **Impact:** Medium
- **Mitigation Strategies:**
  - Use agile planning and iterative development
  - Prioritize features using a product roadmap
  - Document changes with version-controlled requirement specs

R7 – Failure in third-party service integration (e.g., payments, SMS)

- **Category:** Technical / External
- **Probability:** Low
- **Impact:** High
- **Mitigation Strategies:**
  - Choose reliable providers with SLAs
  - Add retry logic and error handling
  - Monitor external service health with fallback alerts

R8 – Poor usability on mobile devices

- **Category:** Usability / Design
- **Probability:** Medium
- **Impact:** Medium
- **Mitigation Strategies:**
  - Implement responsive UI testing
  - Prioritize mobile-first layout in front-end development
  - Continuously gather user feedback for UI improvements

7.2 Risk Monitoring Approach

- Risks will be logged and tracked using a **risk register**.
- High-impact risks will be reviewed at each development phase.
- Automated monitoring tools will be used for **real-time alerts** (e.g., system errors, unusual usage patterns).
- Feedback loops from users will help catch emerging usability or adoption risks

Appendices

Milestones

Marketplace Gannt

Task	Start (Week)	Duration (Weeks)	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12
Project Initiation & Planning	1	2												
Requirements Analysis	2	2												
Architecture Design	3	3												
Backend Development	5	4												
Frontend Development	6	4												
Integration	8	2												
Testing & Validation	7	4												
Deployment Preparation	11	2												
Go-Live & Closure	12	1												

RACI

RACI Matrix – Marketplace Platform Project

The RACI matrix defines roles and responsibilities for key project activities. It ensures accountability, transparency, and effective coordination across the project lifecycle.

Activity	Project Manager	Software Architect	Backend Developer	Frontend Developer	QA	DevOps	Security	Admin
Project Planning	A	C	I	I	I	I	I	I
Requirements Analysis	A	R	C	C	I	I	C	C
Architecture Design	C	A/R	C	I	I	C	C	I
Backend Development	I	C	A/R	I	C	I	C	I
Frontend Development	I	C	I	A/R	C	I	I	C
Testing & Validation	I	I	C	C	A/R	I	I	I
Deployment	I	C	I	I	I	A/R	C	I
Security & Compliance	I	I	C	I	C	I	A/R	I

Go-Live Approval	A	C	I	I	C	C	C	I

RACI Legend

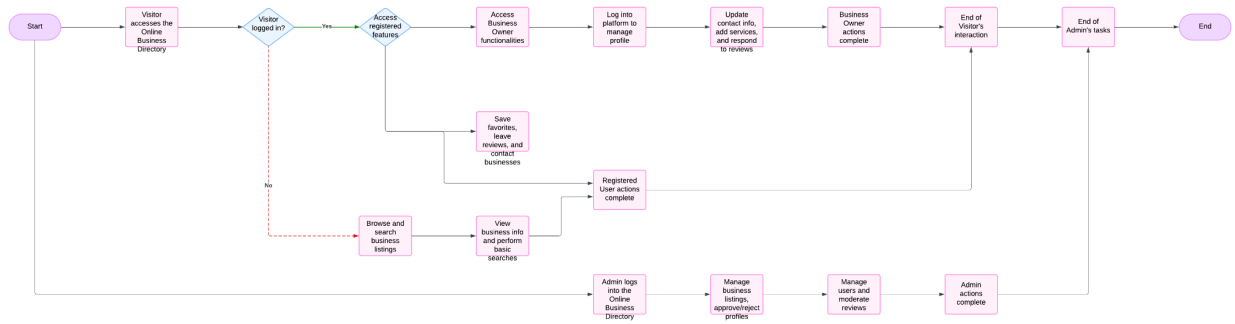
R – Responsible: Performs the work

A – Accountable: Ultimately accountable for the outcome

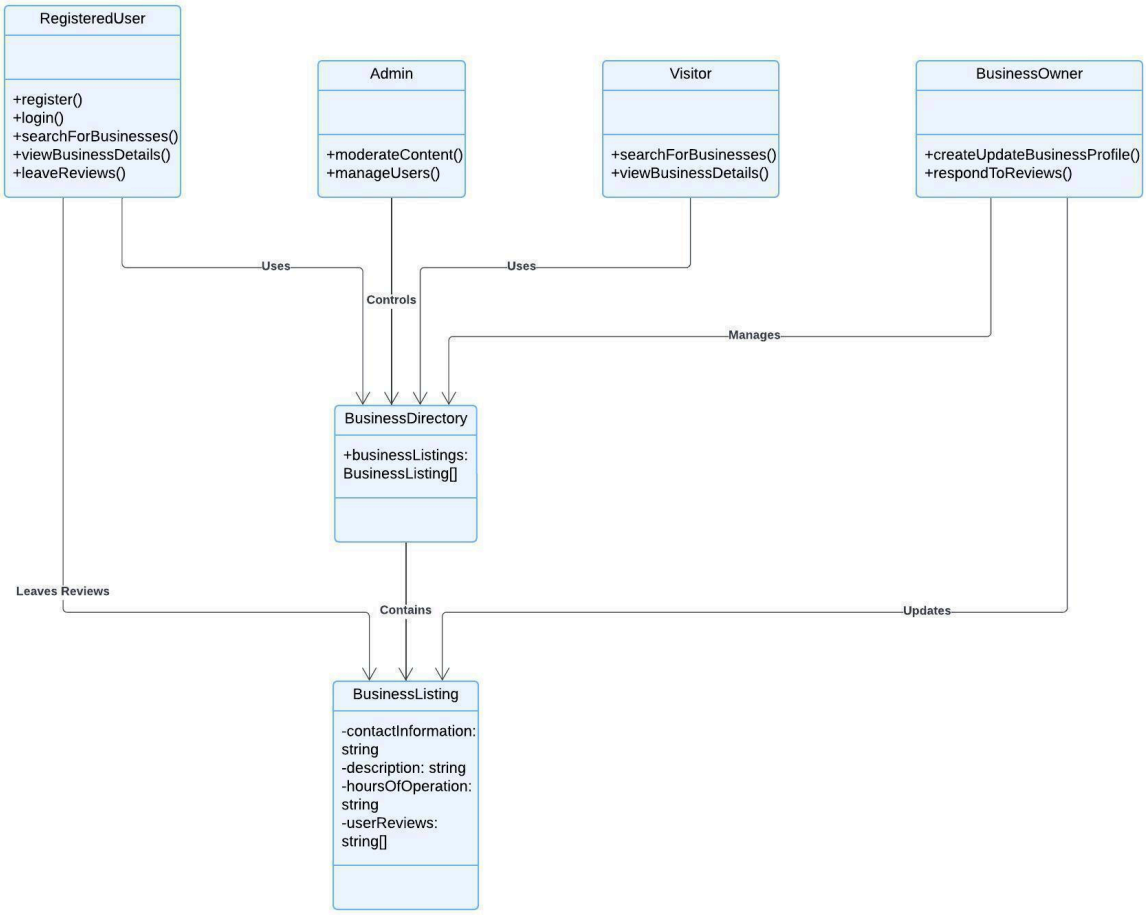
C – Consulted: Provides input or expertise

I – Informed: Kept informed of progress

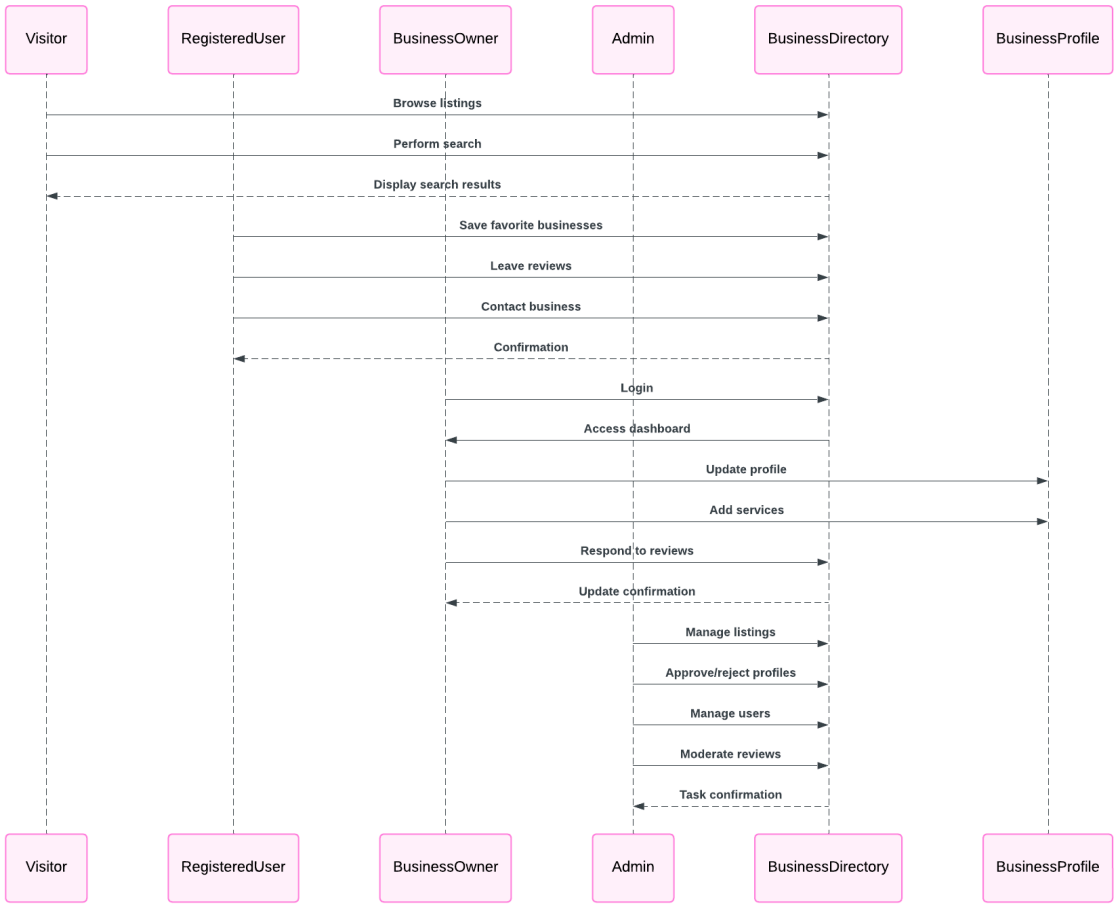
A1. System Diagram



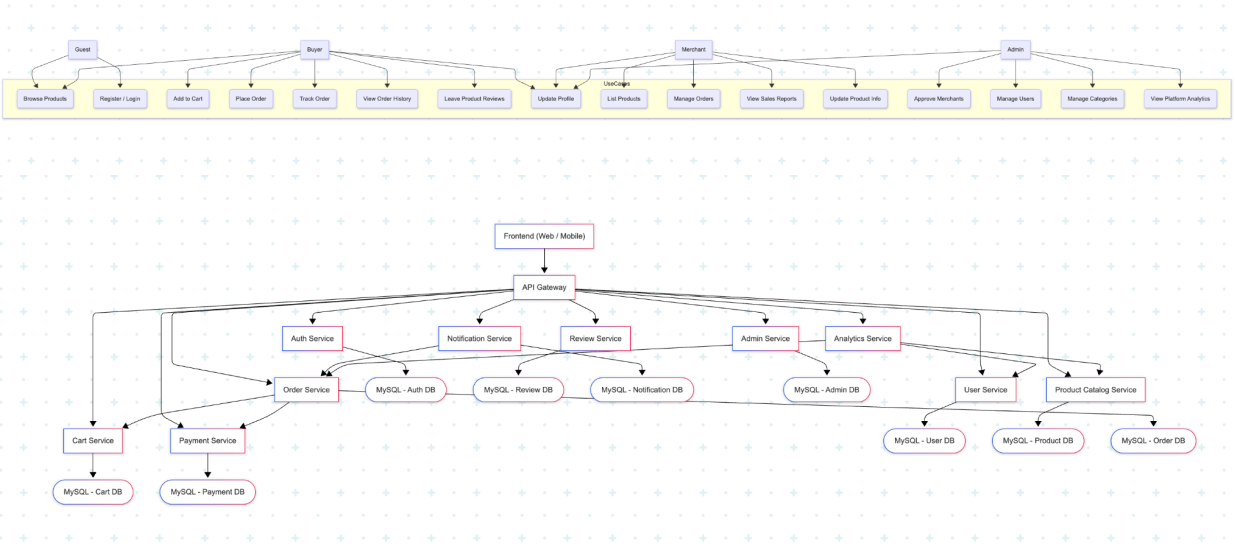
A3. Class Diagrams



A4. Sequence Diagrams


Visitor
RegisteredUser
BusinessOwner
Admin
BusinessDirectory
BusinessProfile
Browse listings
Perform search
Display search results
Save favorite businesses
Leave reviews
Contact business
Confirmation
Login
Access dashboard
Update profile
Add services
Respond to reviews
Update confirmation
Manage listings
Approve/reject profiles
Manage users
Moderate reviews
Task confirmation
Visitor
RegisteredUser
BusinessOwner
Admin
BusinessDirectory
BusinessProfile

A5. Other Diagrams



API Specification (OpenAPI 3.0)

This section provides the API structure for the Marketplace platform, covering key endpoints, authentication, request/response formats, and necessary operations.

B.1 Existing YAML Definitions

B.1.1 Companies API

- Register, update, delete companies
- Fetch company details, search & filtering
- Company verification & approval
- OpenAPI YAML: Companies
- URL : **https://spec.mdir.eu/Companies.yaml**

B.1.2 Products API

- Add, update, delete products

- Product details, pricing, availability
- Search & filter products
- OpenAPI YAML: Products
- URL : **https://spec.mdir.eu/Products.yaml**

B.1.3 Orders & Transactions API

- Create, update, cancel orders
- Payment processing & status tracking
- Order fulfillment & delivery status
- OpenAPI YAML: Orders
- URL : **https://spec.mdir.eu/Orders.yaml**

B.1.4 Categories API

- Manage product/service categories
- Hierarchical category structure (parent-child relationships)
- OpenAPI YAML: Categories
- URL : **https://spec.mdir.eu/Categories.yaml**

C.2 Additional Necessary API Endpoints

These are the endpoints that are essential for the full functionality of the Marketplace but are currently not defined yet.

C.2.1 Authentication & Security API

- User authentication (JWT, OAuth 2.0)
- Role-based access control (Admin, Business Owner, Customer)
- Password recovery & account verification
- OpenAPI YAML: Authentication & Security

C.2.2 Reviews & Ratings API

- Submit & manage product/business reviews
- Rating calculation & moderation
- OpenAPI YAML: Reviews & Ratings

C.2.3 Notifications API

- Email & push notifications
- Order updates & promotional messages
- OpenAPI YAML: Notifications

C.2.4 Admin & Moderation API

- Manage users & businesses
- Approve or reject business listings
- Moderate reviews & reported content
- OpenAPI YAML: Admin & Moderation

C.2.5 Search & Filtering API

- Advanced search for businesses, products, and categories
- Filtering based on price, ratings, and availability
- OpenAPI YAML: Search & Filtering

C.2.6 Error Handling & Response Codes

- Standardized API error responses
- HTTP status codes & messages
- OpenAPI YAML: Error Handling

C.2.7 Rate Limiting & Performance Considerations

- API request limits & throttling
- Performance optimization (caching, indexing)
- OpenAPI YAML: Rate Limiting & Performance