

WORKSHOP I

Zabala Serrano, Andruew Steven. Montoya Arredondo, Ruben David.

Ballen Triana, Hemerson Julian.

Proyecto Curricular de Ingeniería de Sistemas, Universidad Distrital Francisco José de Caldas

Bogotá D.C., Colombia

aszabalas@udistrital.edu.co

hjballed@udistrital.edu.co

rdmontoyaa@udistrital.edu.co

I. CANVAS

BogoGo

A. context

An ecommerce platform that connects multiple sellers and buyers at both national and international levels. The application must handle large volumes of data (orders, products, reviews, clicks, searches, payments, etc.) in real time.

1. Key Partners

Definition: The strategic relationships your business builds with other organizations or people to operate successfully.

- Partnerships with **product suppliers** (manufacturers, distributors, brands) to guarantee variety and availability
- **Logistics companies** (shipping services, warehouses) to manage fast and reliable deliveries.
- **Payment platforms** like PayPal, PayU, Stripe to process transactions securely.
- **Cloud providers** (AWS, Azure, GCP) to support distributed databases, BI, and scalability.

2. Key Activities

- Quick fulfillment processes
- Continuous **development and maintenance** of the ecommerce platform (website + mobile app).
- Creation of **BI dashboards** to provide managers with actionable insights.
- Providing **customer support** via chatbots, call centers, and self-service systems.

3. Key Resources

Definition: The assets (tangible and intangible) required to run the business and deliver the value proposition.

- **Technological infrastructure:** servers, distributed databases, cloud storage, analytics tools.
- **Skilled workforce:** developers, data scientists, marketing experts, and customer support agents.
- **Contracts and agreements** with suppliers and logistics partners

4. Value Proposition

Definition : The unique value your business offers to solve customer needs and stand out from competitors.

- **Personalized shopping experience.**

- **Fast and reliable purchases**, thanks to scalable infrastructure and optimized databases

- **Real-time insights** for sellers and managers to make better business decisions

- **Multi-region product availability** due to distributed storage and logistics networks

- **High availability** (minimal downtime, fast query execution) ensuring seamless shopping.

5. Customer Relationships

Definition: The type of interactions and support the business establishes with its customers.

- **Omnichannel support:** email, and phone.
- **Personalization** of promotions, product suggestions, and newsletters.
- **Loyalty programs** with rewards and discounts for recurring buyers.
- **Community engagement** through reviews, ratings, and user-generated content.

6. Channels

Definition: The ways the company communicates and delivers its value proposition to customers.

Applied to E-Shop360:

- **Website** as the main shopping platform
- **Social media** (Instagram, TikTok, Facebook, LinkedIn) to promote and interact with customers.
- **Marketplaces** (Amazon, MercadoLibre) as complementary sales channels.

7. Customer Segments

Definition: The groups of people or organizations the company targets and serves.

- **End consumers:** individuals buying products for personal use.
- **Small and medium-sized businesses (SMBs)** purchasing supplies or inventory.
- **Third-party sellers** who use the platform as a marketplace to sell their products.
- **Managers and administrators** who need BI insights for business decisions.

8. Cost Structure

Definition: The main costs involved in operating the

business.

Applied to E-Shop360:

- **Platform development and maintenance** (web, app, database).
- **Cloud services** for storage, processing, and distributed access.
- **Marketing expenses** (SEO, paid ads, affiliate marketing).
- **Logistics costs** (shipping, packaging, warehouses).
- **Personnel salaries** (developers, data scientists, customer support).
- **Licenses and tools** (BI dashboards, analytics, payment gateways).

9. Revenue Streams

Definition: The ways in which the business generates income.

- **Transaction commissions** on each sale made through the platform.
- **Advertising revenue** from brands that promote products within the platform.
- **Data analytics services** offered to vendors/suppliers for market insights.
- **Loyalty partnerships** where brands sponsor rewards and discounts.



Fig. 1. Canva model

II. REQUIREMENTS

A. Functional (FR)

User & Account Management

- The system shall allow clients to register, log in, and manage their accounts.
- The system shall allow providers (sellers) to create and manage their profiles.

Product & Catalog Management

- The system shall allow providers to create, update, and delete products with attributes that them select. (size, color, images, videos).

- The system shall allow clients to search and filter products and providers by multiple attributes (name, category, price, rating, availability).

Shopping & Transactions

- The system shall allow customers to add/remove products from the shopping cart.
- The system shall allow secure checkout with multiple payment gateways.
- The system shall show order statuses: confirmed, processing, shipped, delivered, returned.
- The system shall allow both registered users and guest checkout.
- The system shall allow customers to apply discounts, promo codes, and warranties.

Data Ingestion & Analytics

- The system shall continuously ingest user activity (clicks, searches, orders, stock updates) throughout the day.
- The system shall process ingested data in real time for analytical use.
- The system shall generate business intelligence dashboards for managers, showing sales trends, popular products, customer segmentation, and delivery KPIs.

Recommendations & Personalization

- The system shall show product recommendations based on collaborative filtering, content similarity, and contextual data (zone, history, seasonality).
- The system shall display targeted advertisements.

Third-Party Integrations

- The system shall integrate with ERP, CRM, PIM systems for business operations.
- The system shall integrate with payment gateways (e.g., PayPal, Stripe).
- The system shall allow integration with shipping/logistics APIs.
- The system shall allow social media sharing of products and promotions.

Mobile Friendliness

- The system shall provide a mobile-responsive interface accessible on smartphones and tablets.
- The system shall support a native or hybrid mobile app version.

Multi-Location Data Access

- The system shall replicate data across multiple regions to reduce latency.
- The system shall automatically direct users to the nearest node.

B. Non-Functional Requirements (NFR)

Performance & Fast Queries

- Analytical queries must return results in < 2 seconds for aggregated data.
- The homepage shall load in < 4 seconds on mobile (4G).

Scalability

- The system shall handle 10,000 concurrent users.

- The architecture shall support horizontal scaling (adding servers as needed).

Availability & Reliability

- The system shall maintain 99.9% uptime.
- The system shall tolerate node failures with automatic failover.
- The system shall ensure no data loss in case of pipeline failure.

Security

- All sensitive data must be encrypted in transit and at rest.
- Only system administrators can assign roles and change access permissions.
- The system must comply with GDPR and local data regulations.
- The system must be resilient to DDoS, SQL injection, and XSS attacks.

Usability

- Customers must be able to complete purchases within 3 steps (browse → cart → checkout).
- The system shall provide an intuitive user interface with clear calls-to-action.
- BI dashboards must provide interactive and drill-down visualizations.

Maintainability

- The system shall use a microservices architecture for modular updates.
- The backend must be documented to simplify future modifications.

III. USER STORIES

A. Customer Stories

- As a customer, I want to create an account so that I can securely access my purchase history and saved preferences.
 - o AC1: Given I am on the registration page, when I enter valid personal data (name, email, password), then my account is created and I receive a confirmation email.
 - o AC2: Given I try to register with an already used email, when I submit the form, then I see an error message.
- As a customer, I want to browse and search for products so that I can quickly find what I need.
 - o AC1: Given I am on the search bar, when I enter a keyword, then the system returns matching products within 2 seconds.
 - o AC2: Given I apply filters (price range, category), when I search, then only products matching those filters appear.
- As a customer, I want to add products to a shopping cart so that I can purchase multiple items at once.
 - o AC1: Given I select a product, when I click “Add to cart,” then the item appears in my cart.
 - o AC2: Given I have items in my cart, when I update quantity, then the total is recalculated instantly.

- As a customer, I want to check out and pay securely so that I can complete my order successfully.
 - o AC1: Given I am in the checkout flow, when I choose a valid payment method and confirm, then the order is created, and I receive confirmation.
 - o AC2: Given I enter invalid card details, when I confirm, then the system rejects the payment and shows an error.
- As a customer, I want to track the status of my order so that I know when it will arrive.
 - o AC1: Given I am logged in, when I view “My Orders,” then I can see the current status (Confirmed, Processing, Shipped, Delivered).
- As a customer, I want to receive personalized recommendations so that I can discover relevant products.
 - o AC1: Given I am logged in, when I visit the homepage, then I see at least 5 recommended products based on my past activity.

B. Vendor Stories

- As a vendor, I want to create and update product listings so that customers always see accurate information.
 - o AC1: Given I am a vendor, when I add a new product, then it appears in the product catalog.
 - o AC2: Given I update stock to “0,” when customers browse, then the product is marked as “Out of Stock.”
- As a vendor, I want to manage my inventory so that products out of stock are not sold.
 - o AC1: Given inventory level is zero, when a customer tries to add the product to cart, then they are prevented from doing so.
- As a vendor, I want to view my sales analytics so that I can make better business decisions.
 - o AC1: Given I am logged in as a vendor, when I open the analytics dashboard, then I see daily/weekly/monthly sales charts.

C. Administrator Stories

- As an administrator, I want to generate sales and customer behavior reports so that I can evaluate performance.
 - o AC1: Given I select a time range, when I generate a report, then I receive a PDF/CSV file with sales metrics.
- As an administrator, I want to manage vendors and customers so that I can maintain a secure and trusted marketplace.
 - o AC1: Given I am an admin, when I deactivate a vendor, then their products become unavailable.
- As an administrator, I want to ensure high availability and monitor system performance so that the platform is always reliable.
 - o AC1: Given the system reaches 80% capacity, when load increases, then autoscaling is triggered.

D. System-Level / Technical Stories (to reflect your advanced requirements)

- As a system, I want to ingest data continuously so that real-time analytics are always up to date.
 - AC1: Given new user clicks/orders arrive, when data is streamed, then it is processed within <5 seconds.
- As a system, I want to execute fast queries on large datasets so that dashboards load instantly for managers.
 - AC1: Given a query over >1M records, when executed, then results return in <3 seconds.
- As a system, I want to replicate data across multiple locations so that global customers experience low latency.
 - AC1: Given a customer in Europe, when they request product data, then response time is <200ms due to regional replication.
- As a system, I want to scale horizontally so that the platform can handle growing traffic and transactions.
 - AC1: Given 10x traffic increase, when load testing runs, then the system handles requests without downtime.

IV. INITIAL DATABASE ARCHITECTURE

A. Architecture

A hybrid architecture is proposed, specifically utilizing PostgreSQL and MongoDB. PostgreSQL provides the capability to store critical data necessary for executing transactions, as well as to deliver analytics on sales, products, users, and other relevant information. MongoDB, on the other hand, offers the flexibility required to store documents and content such as product images, reviews, ratings, and even feedback on affiliated sellers.

B. ER Diagram

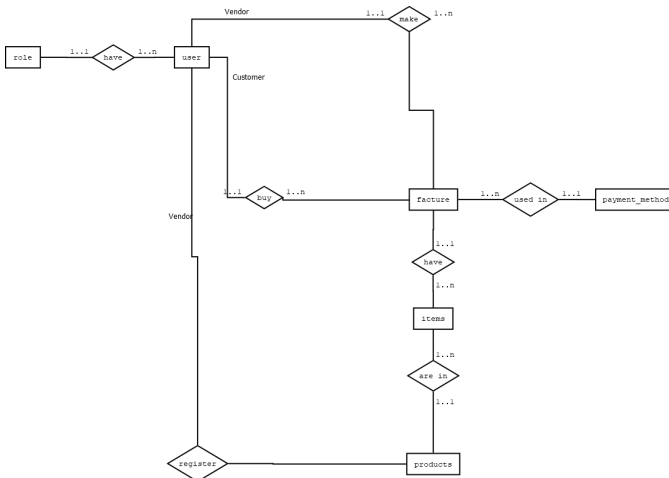


Fig. 2. ER model developed at DIA

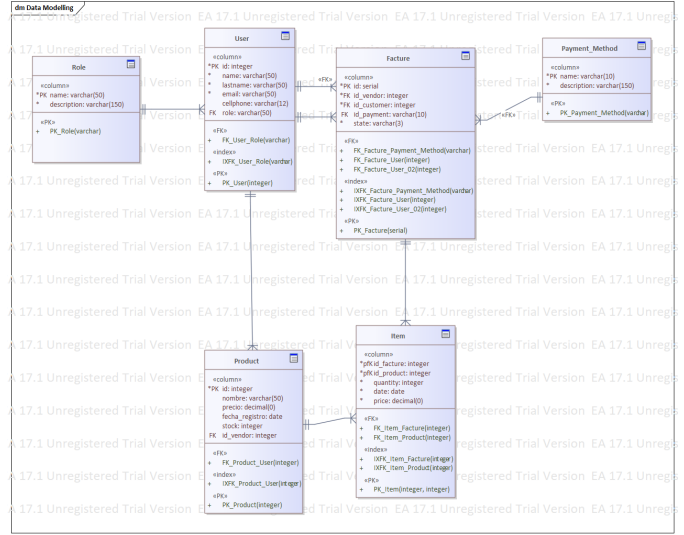


Fig. 3. ER model developed at Enterprise Architect (EA)

C. Entities and Relationships

- Entities:
 - User: Entity that represents a person registered in the system. A user can have different roles (customer, seller, or administrator) and perform actions according to the assigned role, such as purchasing products, registering them for sale, or managing the application.
 - Role: Entity that defines the types of permissions and responsibilities within the system. The available roles are:
 - Customer: Can browse and purchase products.
 - Seller: Can register, manage, and sell products.
 - Administrator: Can manage users and access system metrics or analytical reports.
 - Product: Entity that represents the goods or services available for sale in the system. It includes information such as name, description, category, price, stock, and the associated seller
 - Facture: Entity that stores the details of a transaction carried out in the system. It includes the purchasing user, the seller of the product, the selected payment method and the facture status.
 - Payment Method: Entity that records the payment options used by users to complete transactions (e.g., credit card, debit card, bank transfer, online payment).
 - Item: Entity that records the products included in a Facture. It stores the product ID, facture ID, quantity, price, and transaction date.
- Relationships
 - User – Role
 - Relationship: A User must have a single Role, and a Role can be assigned to many Users.
 - Cardinality: N (users) → 1 (role).
 - Description: Each user belongs to exactly one role (customer, seller, or administrator), but each role

can apply to multiple users.

o Seller – Product

- * Relationship: A Seller can register multiple Products.
- * Cardinality: 1 (seller) \rightarrow N (products).
- * Description: Each product must be associated with a single seller, but a seller can manage many products.

o Seller – Facture

- * Relationship: Relationship: A Seller can be linked to multiple factures.
- * Cardinality: 1 (seller) \rightarrow N (factures).
- * Description: Each facture records the seller of the purchased product.

o Customer – Facture

- * Relationship: A Customer can generate multiple Factures.
- * Cardinality: 1 (customer) \rightarrow N (factures).
- * Description: Each facture is associated with a single customer, but a customer may have many invoices over time.

o Facture – Item

- * Relationship: A Facture contains multiple Items.
- * Cardinality: 1 (facture) \rightarrow N (items).
- * Description: Each item details a purchased product, including quantity, price, and date.

o Product – Item

- * Relationship: A Product can appear in multiple Items.
- * Cardinality: 1 (product) \rightarrow N (items).
- * Description: A product may be purchased in different factures and, therefore, appear in multiple items.

o Facture – Payment Method

- * Relationship: A Facture is paid with a single Payment Method, while a Payment Method can be associated with many factures.
- * Cardinality: N (factures) \rightarrow 1 (payment method).
- * Description: Each facture must register the payment method used, although the same method (e.g., credit card) can be used in multiple factures.

D. Data flows and Application's needs

Considering the application requirements, transactional and relational data (e.g., users, roles, products, invoices, items, and payment methods) should be stored in a relational database to ensure integrity and consistency. Unstructured data such as product images or logs can be managed with a NoSQL database or cloud storage, enabling a hybrid approach that balances scalability, flexibility, and performance.

REFERENCES

[1] CFI Team, "Business Model Canvas Examples," corporate finance institute. [Online]. Available: <https://corporatefinanceinstitute.com/resources/management/business-model-canvas-examples/>

[2] Y. Shvetsova, "Functional and Non-Functional Requirements of Online Shopping System: List & Examples," elogic, Jun. 8, 2023. [Online]. Available: <https://elogic.co/blog/functional-and-non-functional-requirements-for-ecommerce-websites/>

[3] M. Rehkopf, "Historias de usuario con ejemplos y plantilla," atlassian. [Online]. Available: <https://www.atlassian.com/es/agile/project-management/user-stories>

[4] Team Asana, "Historias de usuario: 3 ejemplos para generar valor para el usuario," asana, Jan. 21, 2025. [Online]. Available: <https://asana.com/es/resources/user-stories>