

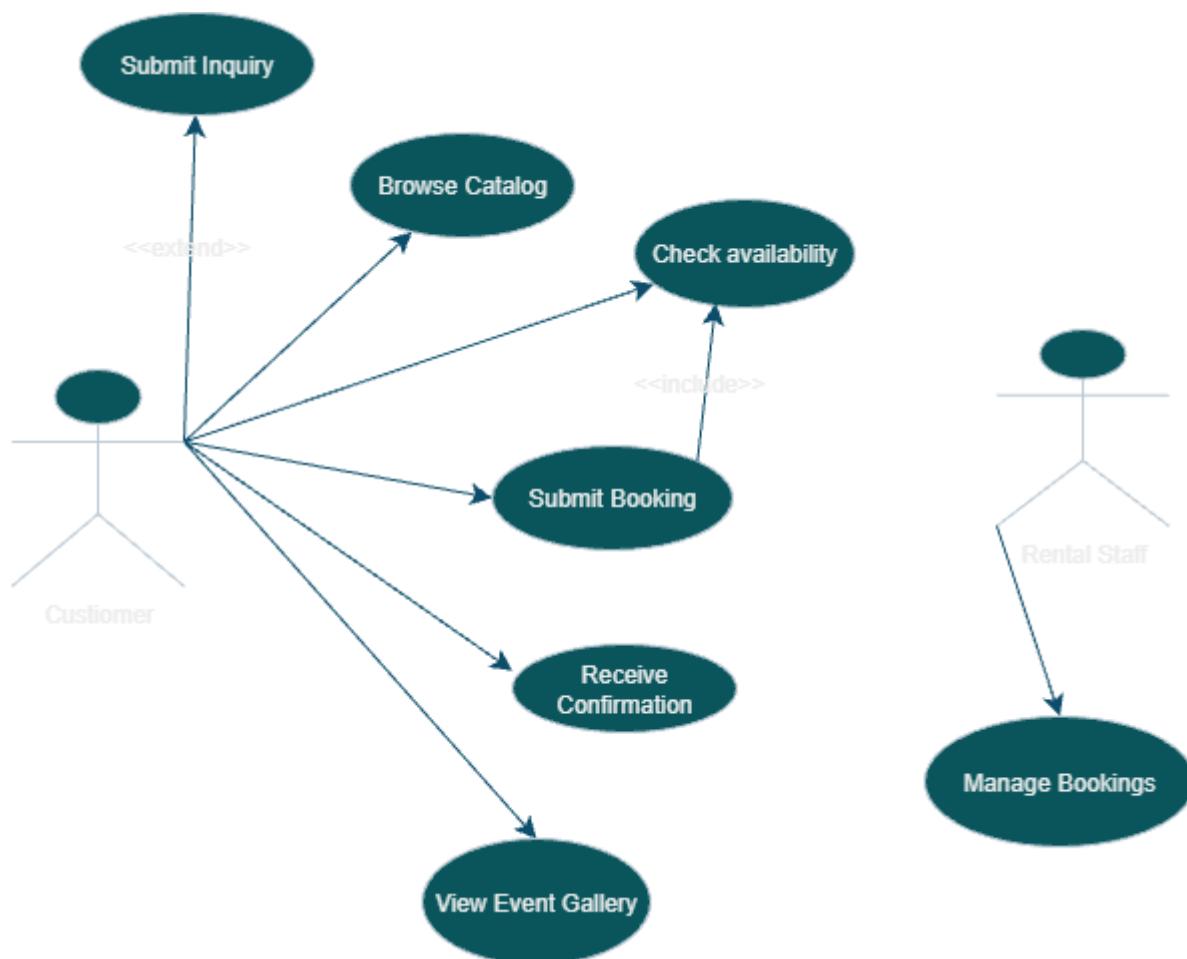
# SYSTEM REQUIREMENTS SPECIFICATION

## 1. Functional Overview & Use-Case Model

- Problem Statement

Currently, Papela manages bookings and inventory manually using physical notebooks, mobile calls, and messaging apps like WhatsApp. This outdated process often results in duplicated reservations, missing orders, and inefficient stock tracking. Delays in communication and limited visibility into item availability have led to poor customer experience and reduced operational efficiency, especially during peak event periods.

- Use-Case Diagram



- Use-Case Descriptions

### **UC1: Browse Catalog**

- **Actor:** Customer
- **Trigger:** Customer navigates to the catalog page on the website.
- **Normal Flow:** System displays a categorized list of rental items (e.g., chairs, tables) with images, descriptions, prices, and filters for category, price, or event type.

## **UC2: Check Availability**

- **Actor:** Customer
- **Trigger:** Customer inputs event dates on the catalog or availability page.
- **Normal Flow:** System queries the inventory database and displays available items for the selected dates in real time.

## **UC3: Submit Booking**

- **Actor:** Customer
- **Trigger:** Customer completes a booking form with event details (date, location, items).
- **Normal Flow:** System validates item availability, saves the booking request, and notifies the customer of submission.

## **UC4: Manage Bookings**

- **Actor:** Rental Staff
- **Trigger:** Staff accesses the secure admin dashboard.
- **Normal Flow:** System displays pending bookings; staff reviews, approvals, or rejects requests, updating the booking status.

## **UC5: Receive Confirmation**

- **Actor:** Customer

- **Trigger:** System processes a successful booking request.
- **Normal Flow:** The System sends an automated email to the customer confirming the booking details.

### **UC6: View Event Gallery**

- **Actor:** Customer
- **Trigger:** Customer navigates to the event gallery page.
- **Normal Flow:** System displays a gallery of past events with images and descriptions to showcase Papela's services.

### **UC7: Submit Inquiry**

- **Actor:** Customer
- **Trigger:** Customer submits a contact form for non-booking inquiries.
- **Normal Flow:** The System saves the inquiry and notifies staff for follow-up.

## 2. USER STORIES

- US-1: As a customer, I want to browse available rental items.

Traceability: UC-1 → US-1

- US-2: As a customer, I want to search for items by category.

Traceability: UC-1 → US-2

- US-3: As a customer, I want to view item details (description, price, availability).

Traceability: UC-6 → US-3

- US-4: As a customer, I want to add items to a rental cart.

Traceability: UC-4 → US-4

- US-5: As a customer, I want to select rental dates.

Traceability: UC-5 → US-5

- US-6: As a customer, I want to submit a rental inquiry/order.

Traceability: UC-7 → US-6

- US-7: As a customer, I want to receive order confirmation.

Traceability: UC-5 → US-7

- US-8: As an admin, I want to add new rental items with details like name, quantity and price.

Traceability: UC-4 → US-8

- US-9: As an admin, I want to edit existing rental item details.

Traceability: UC-4 → US-9

- US-10: As an admin, I want to view the current inventory status of all items.

Traceability: UC-4 → US-10

- US-11: As an admin, I want to update an item's availability whether marked as rented, returned, damaged).

Traceability: UC-4 → US-11

- US-12: As an admin, I want to categorize rental items.

Traceability: UC-4 → US-12

- US-13: As an admin, I want to review and approve/deny rental inquiries.

Traceability: UC-4 → US-13

- US-14: As an admin, I want to schedule deliveries and pickups for orders.

Traceability: UC-4 → US-14

- US-15: As an admin, I want to assign delivery routes to drivers.

Traceability: UC-4 → US-15

- US-16: As an admin, I want to generate invoices for completed rentals.

Traceability: UC-4 → US-16

- US-17: As an admin, I want to view financial summaries (e.g., daily/weekly/monthly revenue).

Traceability: UC-17 → US-17

- US-18: As an admin, I want to generate reports on popular rental items.

Traceability: UC-18 → US-18

### 3. PRECONDITIONS AND POSTCONDITIONS

#### **US-1: Browse available rental items**

Preconditions

1. The customer must be on the UI page containing the rental items.
2. The database must have a column showing the status of rental items.  
Whether they are “available” or not.
3. The customer must have a reliable internet connection.

Postconditions

1. The items are displayed with their information.

#### **US-2: Search for items by category**

Preconditions

1. The database has a column that categorizes items.
2. The customer has access to the search feature with a category filter.

Postconditions

1. The search results list the items in that category.

#### **US-3: View item details**

Preconditions

1. The item is in the database.
2. The customer selects that item.

Postconditions

1. A detailed description of the item appears.

## **US-4: Add items to the rental cart**

Preconditions

1. The items are in the database.
2. The customer has an active cart.

Postconditions

1. The items are added to the cart.
2. The UI shows the number of items in the cart.

## **US-5: Select rental dates**

Preconditions

1. The customer is allowed to pick a date on the cart page.
2. The items are in the cart.

Postconditions

1. The dates are linked to the cart and stored.

## **US-6: Submit rental inquiry/order.**

Preconditions

1. There is at least one item in the cart.
2. The customer fills in an inquiry form with necessary details.

Postconditions

1. UI displays submission confirmation.
2. The order is labelled in the database as pending.
3. An SMS or email is sent to confirm submission.

## **US-7: Receive order confirmation**

Preconditions

1. A successful order was made.
2. The SMS or email integration is working.

## Postconditions

1. Email or SMS is sent to the customer.
2. A record of the order is made in the database.

## **US-8: Add new rental items**

### Preconditions

1. The admin is authorized to do so.
2. The UI provides an item entry form.

### Postconditions

1. The items are stored in the database.
2. The items are made visible to the customers.

## **US-9: Edit existing item details**

### Preconditions

1. The admin has authorization.
2. The items are in the database.

### Postconditions

1. The item details are edited and updated.
2. The changes are made visible to the customers.

## **US-10: View inventory status**

### Preconditions

1. The admin is authorized to do so.
2. There are items in the inventory.

### Postconditions

1. The inventory status is displayed.

## **US-11: Update item availability**

#### Preconditions

1. The admin has authorization.
2. The items are in the database.
3. The admin selects the update action.

#### Postconditions

1. The database updates the item's availability status.
2. The customers see the updated status in the UI.

### **US-12: Categorize rental items**

#### Preconditions

1. The admin has authorization
2. The database contains existing categories and can create new ones.

#### Postconditions

1. The items are updated with their category ID.
2. The updated categories are visible to the customer.

### **US-13: Approve or deny rental inquiries**

#### Preconditions

1. The admin is authorized.
2. There are inquiries in the database.

#### Postconditions

1. The status is changed to show whether the order is approved or denied.
2. The customer receives an email or SMS to let them know what was decided.

### **US-14: Schedule deliveries and pickups**

#### Preconditions

1. The admin is logged in.
2. There are approved orders.

3. The scheduling module is active in the UI.

#### **Postconditions**

1. The database contains delivery and pickup schedules.
2. The schedule is communicated to drivers and clients.
3. The admin dashboard's user interface displays scheduled tasks.

### **US-15: Assign Delivery Routes to Drivers**

#### **Preconditions**

1. Admin is authenticated.
2. Drivers and delivery orders exist in the database.
3. The routing module is available in the UI.

#### **Postconditions**

1. Assigned routes are saved in the database for each driver.
2. Drivers receive notifications of their assigned routes.
3. UI updates the delivery board with route assignments.

### **US-16: Generate Invoices for Completed Rentals**

#### **Preconditions**

1. Admin is authenticated.
2. Completed rental orders exist in the database.
3. The invoice template system is configured.

#### **Postconditions**

1. An invoice record is generated and saved in the database.
2. Customer receives the invoice via email.
3. UI displays confirmation and links to download the invoice.

## **US-17: View Financial Summaries**

### **Preconditions**

1. Admin is logged in.
2. Financial transactions exist in the database.

### **Postconditions**

1. UI displays summarized data: daily, weekly, and monthly revenue.
2. No modifications to the database.

## **US-18: Generate Reports on Popular Rental Items**

### **Preconditions**

1. Admin is authenticated.
2. The system has tracked item rental counts in the database.
3. The reporting module is accessible in the UI.

### **Postconditions**

1. A report is generated showing the most popular items.
2. Report data is displayed in the UI and can be exported.
3. Database remains unchanged.

## **4. STORY SIZING AND EPICS**

An **epic** is a large body of work that can be broken down into smaller, more manageable tasks called user stories. If the user stories are ones that will take up to two (2) days of effort to complete, they become epics. Below are some epics and their decomposition.

- **Epic E1: Customer Rental Workflow**

➤ **Reason for Epic:** This comprises the entire end-to-end process for a customer from browsing to placing an order. This would involve significant UI/UX design, database interactions, and complex logic, which would most definitely exceed two (2) days.

➤ **Proposed Decomposition:**

- US-1: As a customer, I want to browse available rental items.
- US-2: As a customer, I want to search for items by category.
- US-3: As a customer, I want to view item details (description, price, availability).
- US-4: As a customer, I want to add items to a rental cart.
- US-5: As a customer, I want to select rental dates.
- US-6: As a customer, I want to submit a rental inquiry/order.
- US-7: As a customer, I want to receive order confirmation.

• **Epic E2: Inventory Management System**

➤ **Reason for Epic:** This involves all the backend functionality for managing rental items, their status, quantities, and potentially maintenance records. This is a core system component that has multiple interfaces and data models.

➤ **Proposed Decomposition:**

- US-8: As an admin, I want to add new rental items with details like name, quantity and price.
- US-9: As an admin, I want to edit existing rental item details.
- US-10: As an admin, I want to view the current inventory status of all items.
- US-11: As an admin, I want to update an item's availability whether marked as rented, returned, damaged).

- US-12: As an admin, I want to categorize rental items.
- **Epic E3: Order Fulfillment & Logistics**
  - **Reason for Epic:** This involves everything from processing an order after it's placed, scheduling deliveries, tracking items, and managing returns. This includes complex scheduling, potentially mapping integrations, and communication tools.
  - **Proposed Decomposition:**
    - US-13: As an admin, I want to review and approve/deny rental inquiries.
    - US-14: As an admin, I want to schedule deliveries and pickups for orders.
    - US-15: As an admin, I want to assign delivery routes to drivers.
- **Epic E4: Financial Management & Reporting**
  - **Reason for Epic:** This involves handling payments, tracking revenue, and creating various financial reports. This requires integration with accounting systems and reporting features.
  - **Proposed Decomposition:**
    - US-16: As an admin, I want to generate invoices for completed rentals.
    - US-17: As an admin, I want to view financial summaries (e.g., daily/weekly/monthly).
    - US-18: As an admin, I want to generate reports on popular rental items.

## 5. NON-FUNCTIONAL REQUIREMENTS

ID	QUALITY ATTRIBUTE	REQUIREMENT	RATIONALE	MEASUREMENT/TEST
NFR-SEC-01	Security	System shall enforce passwords of minimum	To mitigate risk of unauthorized access	Automated password strength

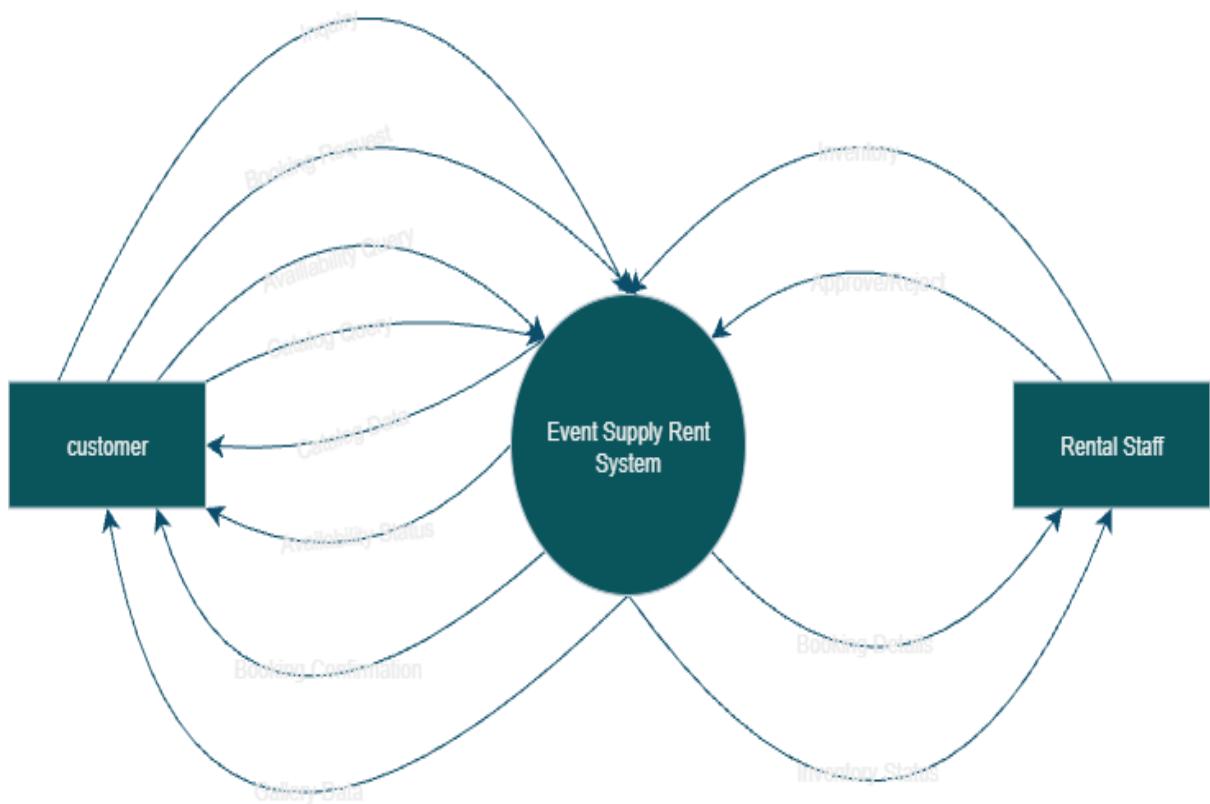
		12 characters, including uppercase, lowercase, digits, and symbols.	and brute-force attacks to sensitive data.	validation; quarterly penetration tests.
NFR-PERF-01	Performance	System shall handle at least 200 concurrent user requests with an average response time under 2 seconds.	To ensure responsiveness and a smooth user experience under load.	Load testing tool simulating 200+ users; monitor response-time logs.
NFR-AVAIL-01	Availability	System shall achieve 99.9% uptime per calendar month, excluding scheduled maintenance windows.	To maintain service reliability and meet SLAs and to ensure user trust in the system.	Uptime monitoring dashboard; monthly availability report.
NFR-USAB-01	Usability	New users shall complete core workflows (e.g., placing an order) without assistance in under 3 minutes.	To reduce training overhead and improve user satisfaction and to assist users in navigating the interface intuitively.	Usability testing sessions; time-on-task measurements and user surveys.
NFR-MNT-01	Maintainability	Codebase shall maintain a unit-test coverage of at least 85% and average cyclomatic complexity $\leq 10$ .	To facilitate easier troubleshooting and future enhancements and to simplify future maintenance and reduce risk of bugs.	Automated coverage reports; static analysis of code complexity.
NFR-REL-01	Reliability	The system shall achieve at least 99.9% uptime per calendar month.	Ensure service is available and trustworthy for end users.	Monitor service health continuously; calculate monthly uptime percentage from logs.
NFR-SCAL-01	Scalability	The system shall scale horizontally to handle at least 1,000 requests per second without degradation in throughput or latency.	Accommodate traffic growth and peak demand seamlessly.	Perform stress tests ramping to $\geq 1,000$ RPS; monitor CPU, memory, and response times under load.

## 6. GLOSSARY AND REFERENCES

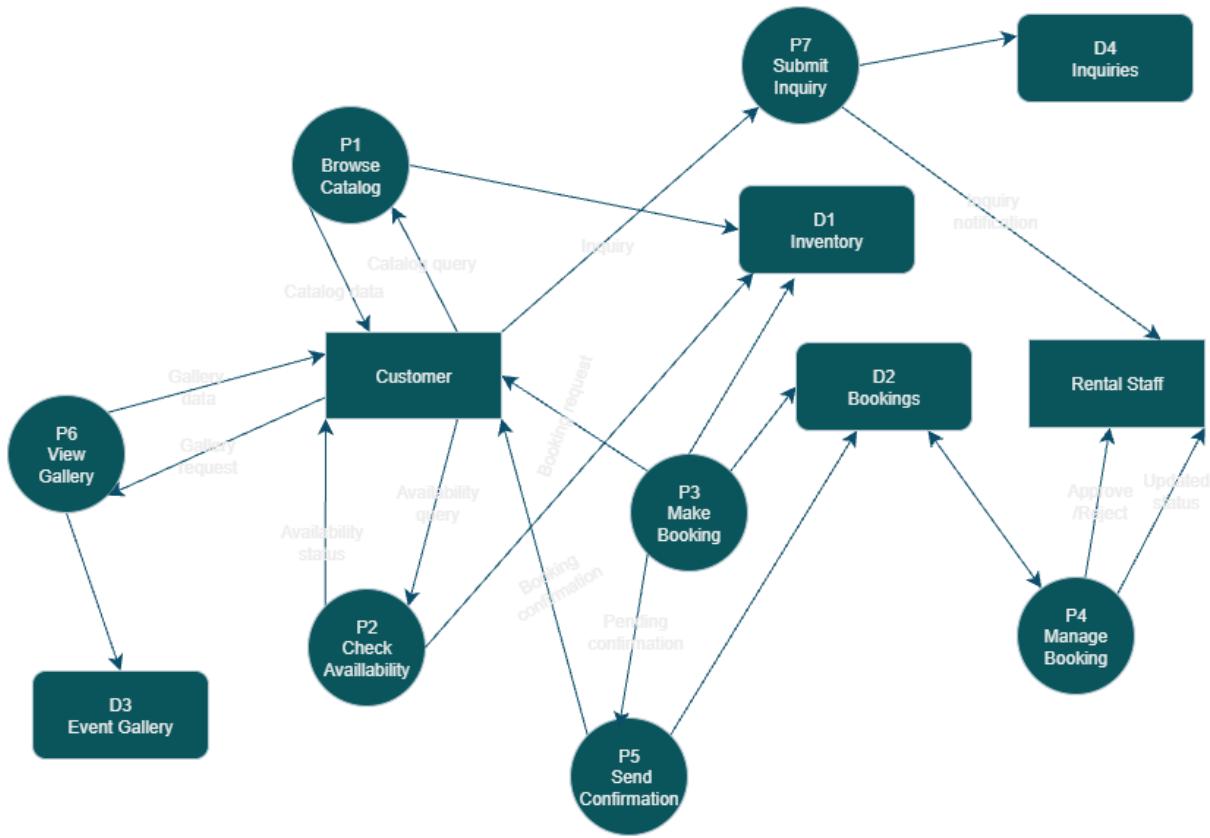
- **Admin:** The person in charge of managing the website, including overseeing products, processing orders, and handling user accounts.

- **API (Application Programming Interface):** A set of rules and protocols that allows different software systems to communicate and share data effectively.
- **Cart:** A virtual basket where customers can add items they want to buy before checking out.
- **Checkout:** The final step in the shopping process is where customers confirm their order and complete the payment.
- **Customer:** Any individual who visits the e-commerce site to browse and purchase event cards.
- **Inventory:** The total number of event cards available for sale at any time.
- **Order:** A confirmed request from a customer to purchase one or more products from the website.
- **Payment Gateway:** A secure online service (like PayPal or Stripe) that processes customers' electronic payments.
- **Product Listing:** A dedicated page showcasing available event cards, complete with images, descriptions, and pricing.
- **SKU (Stock Keeping Unit):** A unique code assigned to each product in the inventory, making it easy to track and manage.
- **User Account:** A registered profile that allows customers to log in securely, store their personal information, and track their order history.

## 7. SUPPLEMENTARY DIAGRAMS



DFD LEVEL 0



DFD LEVEL 1

## ERD DIAGRAM

