**📄 Sample PRD Template (Product Requirements Document)**

**📌 1. Document Metadata**

Title: Cancel Order Feature – PRD

Author: Raman (Product owner)

Version: 1.0

Date: July 30, 2025

Reviewers: Product Lead, Engineering Manager, Customer Ops Lead

**🎯 2. Problem Statement**

Users currently cannot cancel orders after placing them. This causes increased support tickets and user dissatisfaction.

**✅ 3. Goals and Success Criteria**

* Allow users to cancel an order within 10 minutes of placing it **if it hasn't shipped**
* Reduce "order cancelation" related support tickets by **30% in 6 weeks**
* Ensure < 500ms cancel response time in 95th percentile

**📦 4. In-Scope and Out-of-Scope**

| **✅ In-Scope** | **❌ Out-of-Scope** |
| --- | --- |
| - Cancel button on Web and App UI | - Refunds after cancellation |
| - Cancel API via Order Service | - Return process post-cancellation |
| - Integration with Fulfillment API | - Email/SMS notification templates |

**🧩 5. User Stories / Use Cases**

* *As a user*, I should be able to cancel an unshipped order within 10 minutes.
* *As a support agent*, I should see cancelation logs in the dashboard.

**🧱 6. UX Flows (link wireframes or attach screenshots)**

**🔁 7. System Flow Summary (brief text + reference diagram)**

When a user clicks “Cancel,” the frontend calls the Order API. If the status is processing, it changes to cancelled, and a message is sent to the warehouse system.

**📌 8. Assumptions**

* Order status is updated in near real-time by Fulfillment system
* Cancelation window is fixed to 10 mins post-order time

**⚠️ 9. Open Questions**

* What if a user tries canceling after fulfillment begins but before shipping?
* Should we notify 3rd party delivery partners too?

**🔗 10. Dependencies**

* Fulfillment API
* Email Notification System
* Warehouse Event Bus /pub-sub /sqs …

**📅 11. Timeline & Milestones**

| **Phase** | **Target Date** |
| --- | --- |
| Spec Finalization | Aug 2, 2025 |
| Dev Start | Aug 5, 2025 |
| UAT Completion | Aug 25, 2025 |
| Release to Prod | Sep 2, 2025 |

**⚙️ Sample Technical Spec Template**

**📌 1. Document Metadata**

Title: Cancel Order – Technical Design Spec

Author: Engineering Lead

Reviewed by: TPM, Product Lead, Architect

Version: 1.1

Last Updated: July 30, 2025

**🧠 2. Objective**

Enable cancelation of unshipped orders within 10 minutes through API, and notify downstream systems (e.g., warehouse).

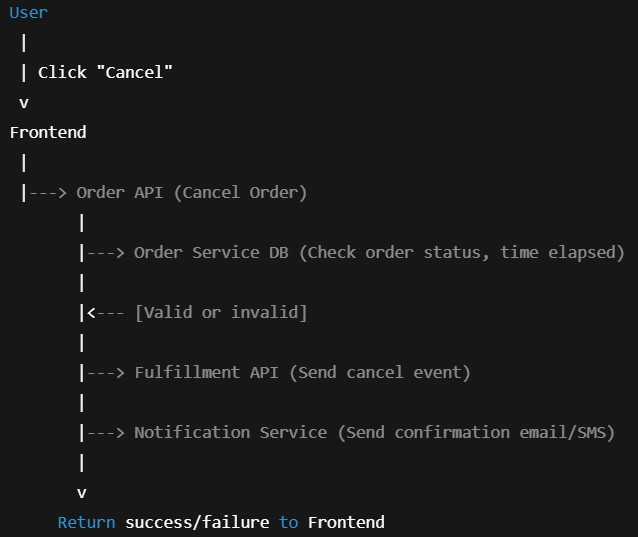
**🧱 3. Architecture Overview (include or link diagrams)**

See attached component and sequence diagrams for flow between Frontend → Order Service → Fulfillment API.

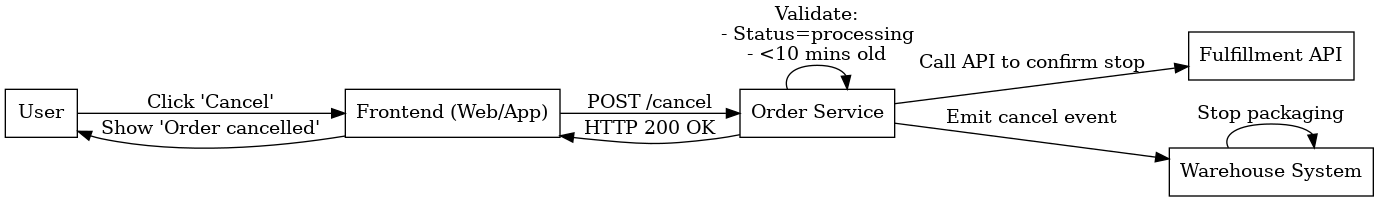
* SEQUENTIAL DIAGRAM :

Show how the feature behaves during real usage

Example 1 :

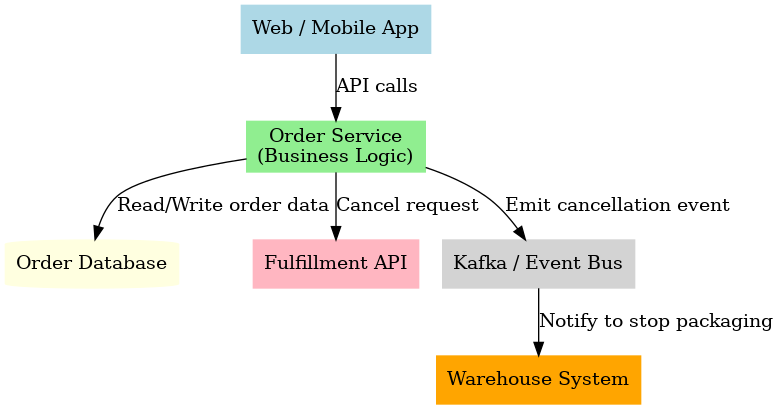


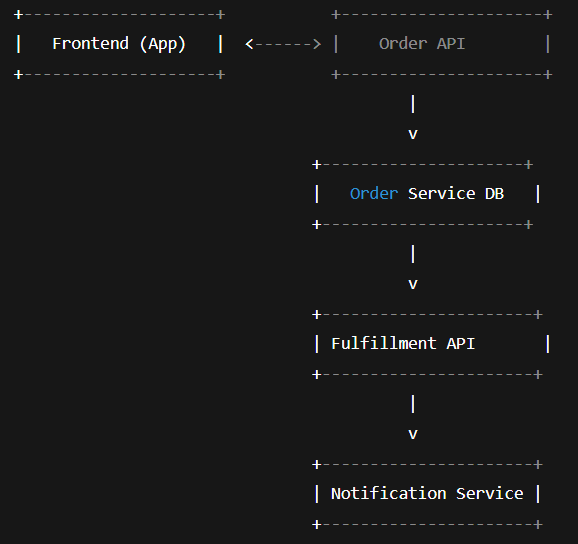
Example 2 :



* COMPONENT DIAGRAM

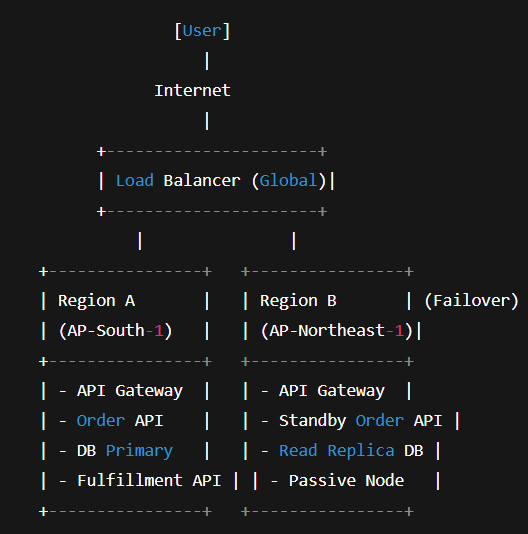
Help non-engineers understand what parts of the system are touched

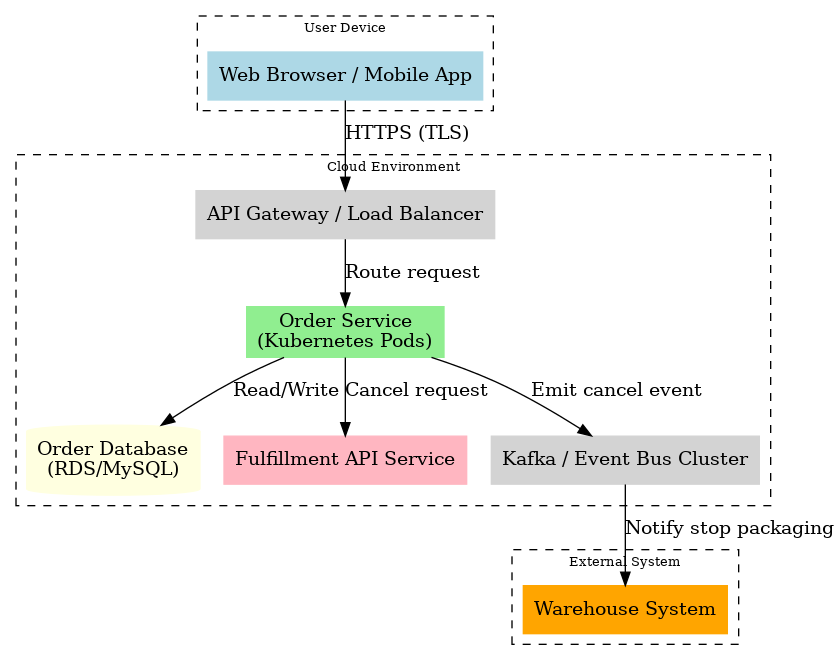




* DEPLOYMENT DIAGRAM :

Show external/internal systems and infra reliance





**⚙️ 4. Design Details**

**a. New API Endpoint**

POST /orders/{order\_id}/cancel

**b. Validation Logic**

* Only cancel if:
  + order.status ∈ [‘processing’]
  + current\_time < order.created\_time + 10 mins
* Return HTTP 400 if invalid

**c. State Transition**

processing → cancelled

**d. Event Notification**

* Push event to Kafka topic order.cancellation.requested
* Fulfillment listens and stops packaging if not started

**📚 5. Data Model Changes**

* Add cancelled\_by\_user\_at timestamp to Order table
* Add cancel\_reason (optional field)

**💥 6. Failure Handling**

| **Scenario** | **Handling Strategy** |
| --- | --- |
| Fulfillment API is down | Retry w/ exponential backoff |
| Warehouse event missed | Periodic audit job every hour |
| Duplicate cancel request | Return 409 Conflict |

**📊 7. Metrics & Observability**

* Metric: order\_cancellations\_attempted
* Metric: order\_cancellations\_failed
* Alert: ≥10 failed cancellations in 5 min window

**🧪 8. Test Plan**

| **Test Case** | **Type** |
| --- | --- |
| Cancel order within 10 mins | Unit, API |
| Cancel shipped order | API |
| Cancel with downstream failure | Integration |
| Retry logic success after outage | E2E |

**📎 9. Non-Functional Requirements**

* Response time < 500ms (P95)
* Availability SLA: 99.9%
* Secure: Authenticated API with role check

**🔀 10. Alternatives Considered**

| **Option** | **Pros** | **Cons** |
| --- | --- | --- |
| Event-based cancelation model | Decoupled, scalable | Requires async handling |
| Synchronous cancel + DB update | Simple to test | Tightly coupled with warehouse |

**📌 11. Risks & Mitigations**

| **Risk** | **Mitigation** |
| --- | --- |
| Warehouse system slow | Use retry queue with TTL |
| User cancels twice rapidly | Deduplicate using unique event ID |

**✅ 12. Approval Checklist**

* Reviewed by Fulfillment Owner
* Load tested in staging
* All tests pass
* Diagrams included