Performance Testing

Date	06 November 2025
Team ID	F529277D47452DB9E7447BD087E08E5E
Project name	Medical inventory management
Maximum marks	4 marks

Goals & high-level objectives :

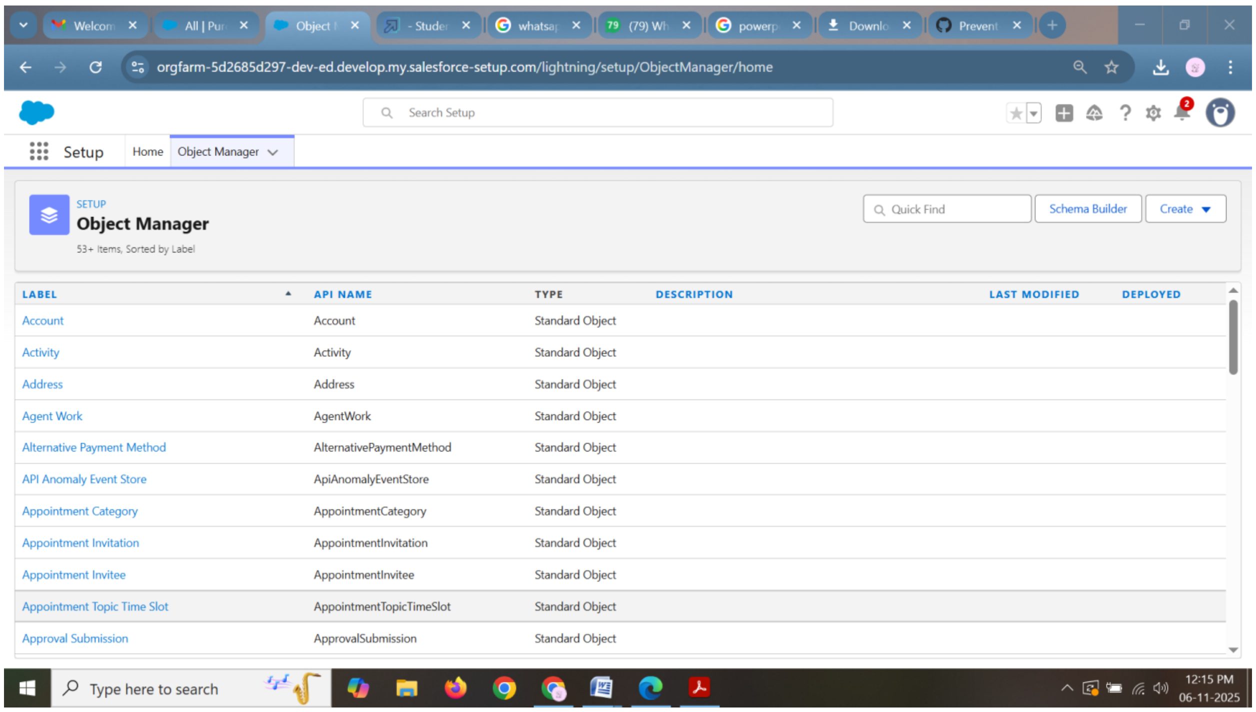
Verify system can handle expected and peak loads for core flows (search, read inventory, place/confirm orders, add/update stock, bulk imports).

Validate response-time SLAs, throughput, and resource usage under load.

Confirm data integrity and concurrency safety during high contention (e.g., two users decrement same stock).

Find bottlenecks (DB, API, caches) and gather actionable metrics for tuning and capacity planning.

Authentication / token refresh — visible at session star



SETUP
Object Manager
53+ Items, Sorted by Label

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Account	Account	Standard Object			
Activity	Activity	Standard Object			
Address	Address	Standard Object			
Agent Work	AgentWork	Standard Object			
Alternative Payment Method	AlternativePaymentMethod	Standard Object			
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object			
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			
Appointment Topic Time Slot	AppointmentTopicTimeSlot	Standard Object			
Approval Submission	ApprovalSubmission	Standard Object			

Performance metrics & SLAs (example):

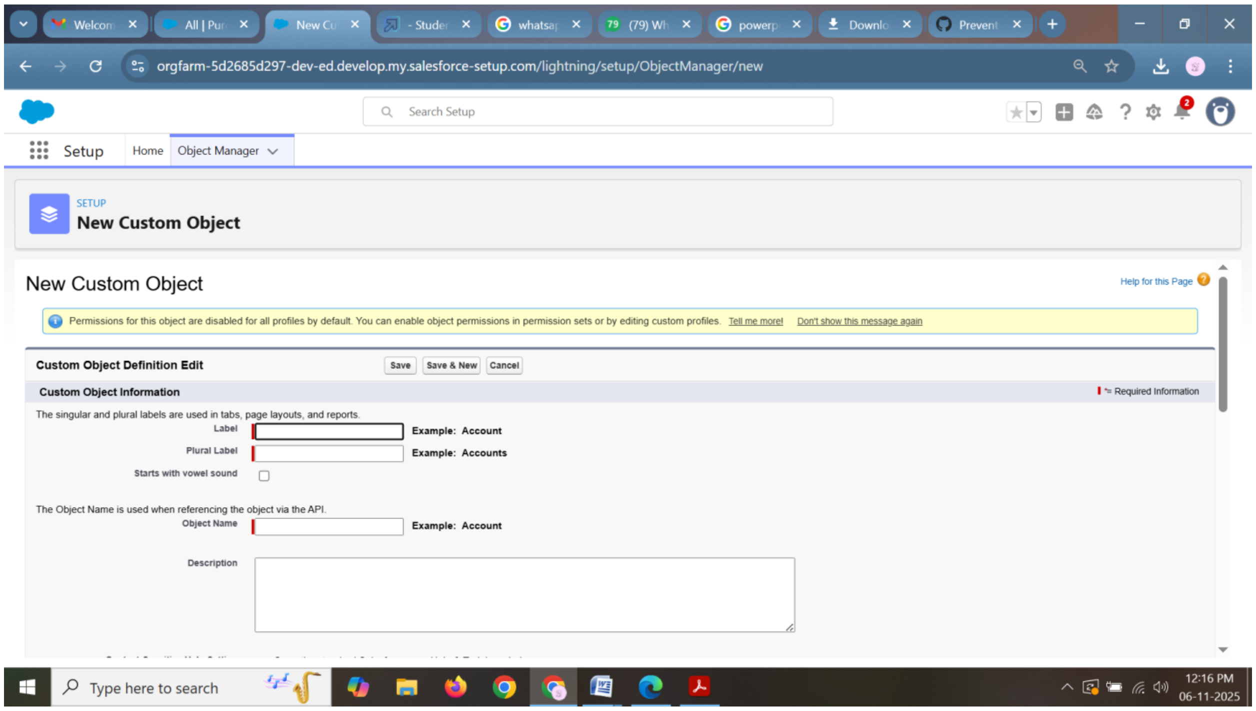
Inventory read (GET): 99% ≤ 500 ms, 99.9% ≤ 1.5 s

Inventory search (list) paginated: 95% ≤ 1.0 s, 99% ≤ 2.0 s

Place order (write flow): 95% ≤ 2.0 s, 99% ≤ 4.0 s

Bulk import: completes within X minutes depending on size (e.g., 10k rows < 5 min) — measure per-case

Error rate: ≤ 0.1% under normal load



New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. Tell me more! Don't show this message again

Custom Object Definition Edit

Custom Object Information | = Required Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label		**Example: Account**
Plural Label		**Example: Accounts**
Starts with vowel sound	☐	

The Object Name is used when referencing the object via the API.

Object Name		**Example: Account**
Description		

Test scenarios & user journeys (with mixes):

Read-heavy scenario (70% reads, 20% searches, 10% writes) — representative of normal operations.

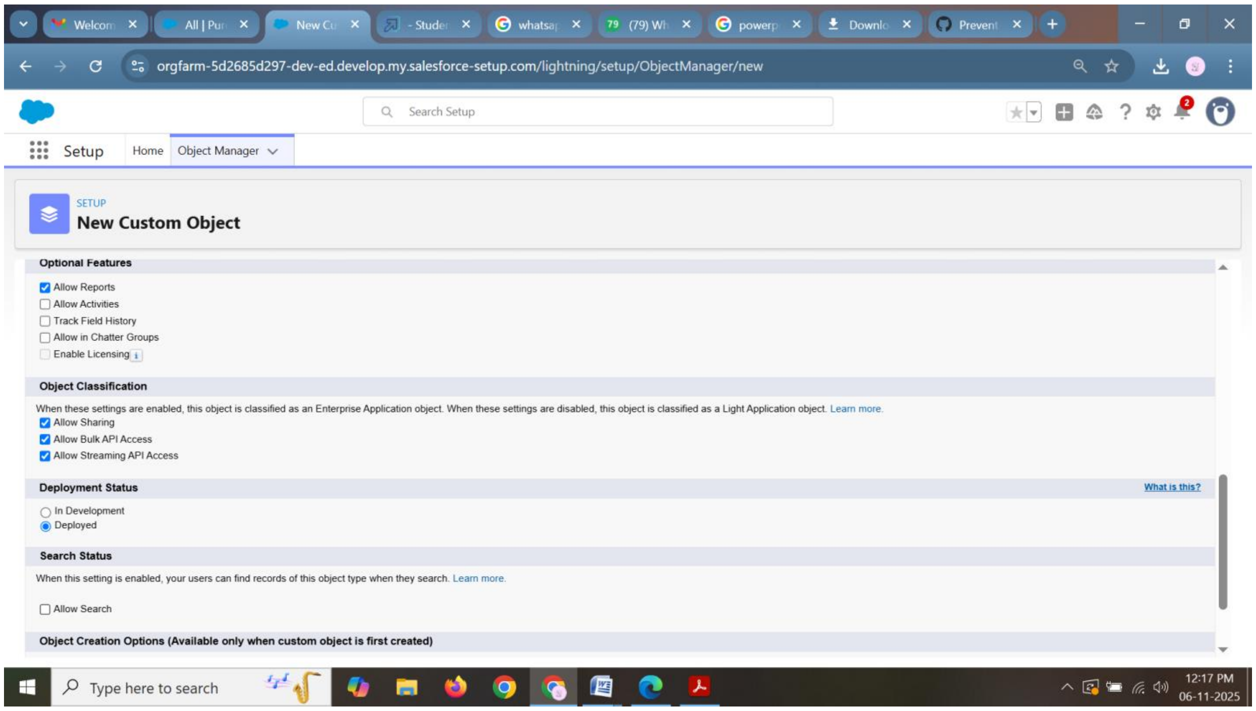
Write-heavy scenario (50% writes — orders/stock updates, 30% reads, 20% searches) — simulate stock reconciliation or inventory audit day.

Concurrency contention: many users attempt to decrement same SKU concurrently (simulate limited stock) — test optimistic/pessimistic locking.

Bulk upload scenario: run import job while system under normal load (observe effect).

Failover/ degraded mode: take one app node down mid-test, verify graceful degradation and performance of remaining nodes.

Long-run endurance: sustained peak for 6–8 hours.


orgfarm-5d2685d297-dev-ed.develop.my.salesforce-setup.com/lightning/setup/ObjectManager/new
Search Setup
Setup
Home
Object Manager

SETUP
New Custom Object

Optional Features

- [✓] Allow Reports
- [ ] Allow Activities
- [ ] Track Field History
- [ ] Allow in Chatter Groups
- [ ] Enable Licensing

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. Learn more.

- [✓] Allow Sharing
- [✓] Allow Bulk API Access
- [✓] Allow Streaming API Access

Deployment Status What is this?

In Development [✓ Deployed]

Search Status

When this setting is enabled, your users can find records of this object type when they search. Learn more.

- [ ] Allow Search

Object Creation Options (Available only when custom object is first created)

Test data & state management:

Use a dedicated test environment with production-like dataset sizes (number of SKUs, categories, orders).

Prepare multiple user accounts with roles and tokens; reuse session cookies to simulate session behavior.

Include realistic SKU distributions: many low-activity SKUs + a smaller set of "hot" SKUs that get frequent reads/updates.

Data seeding scripts: create N SKUs (e.g., 100k), M locations (warehouses), and historic orders to produce realistic DB sizes.

Reset/rollback strategy between runs (DB snapshots, container reset, or test DB clones).