Project Development Phase Model
Performance Test

Project Development Phase Model Performance Test

Date	10 February 2025
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	

Model Performance Testing:

Objective: Evaluate the performance of a machine learning model in various scenarios to ensure it meets the required standards.

In a cosmetic store management project in Salesforce, the project development phase includes multiple stages such as planning, design, development, testing, deployment, and maintenance. A model performance test evaluates how well the system functions in terms of efficiency, scalability, and accuracy.

Model Performance Testing in Cosmetic Store Management Project

This phase ensures that Salesforce solutions perform optimally under different conditions. Performance testing in Salesforce can include:

1. **Load Testing –** Measures how the system handles multiple users accessing features like product inventory, order management, and customer interactions.
2. **Stress Testing –** Checks system stability under extreme conditions, such as high traffic on promotional sale days.
3. **Scalability Testing –** Assesses how well the system can expand when new users, products, or stores are added.
4. **Data Processing Speed –** Evaluates how quickly transactions are completed, such as order placement, checkout, and customer queries.
5. **Integration Performance –** Tests how Salesforce integrates with payment gateways, inventory systems, and marketing platforms.
6. **Security Testing –** Ensures data protection for customers and store operations.

Test Cases Table for Salesforce-Based Cosmetic Store Management System

Project development phase model performance test in cosmetic store management project in Salesforce and test cases tables explain briefly and additional test cases

Project Development Phase & Model Performance Test in Cosmetic Store Management (Salesforce)

A Cosmetic Store Management Project in Salesforce goes through various project development phases, ensuring smooth implementation and performance. Here’s how model performance testing fits into the development process:

Project Development Phases

1. **Requirement Gathering & Planning**
    - Define business needs (inventory, orders, customers, payments, reports).

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

- Identify Salesforce features to implement.

2. **Design Phase**
   - Create data models (products, customers, sales transactions).
   - Plan Salesforce UI, workflows, and integrations.
3. **Development Phase**
   - Configure Salesforce objects (custom fields, relationships).
   - Develop automation (flows, process builders, Apex triggers).
   - Integrate third-party systems (payment gateways, marketing tools).
4. **Testing Phase (Includes Model Performance Test)**
   - Functional Testing: Validates features like order placement, inventory updates.
   - Model Performance Testing: Ensures system efficiency, scalability, and reliability.
   - Integration Testing: Verifies external system connectivity.
   - Security Testing: Checks data protection measures.
5. **Deployment Phase**
   - Move configurations from sandbox to production.
   - Train store staff on Salesforce usage.
6. **Maintenance & Optimization**
   - Monitor system performance and fix issues.
   - Improve features based on feedback.

Model Performance Test in Cosmetic Store Management (Salesforce)

Performance Testing Parameters

1. **Load Testing:** Ensures Salesforce can handle multiple users simultaneously.
2. **Stress Testing:** Tests system behavior under peak usage (e.g., festival season).
3. **Scalability Testing:** Evaluates how the system performs when new products/stores are added.
4. **Data Processing Speed:** Measures order processing, checkout time, and report generation.
5. **Integration Performance:** Tests how external systems (payment gateways, email marketing tools) function.
6. **Security Testing:** Ensures role-based access control and data protection.

Test Cases Table for Salesforce-Based Cosmetic Store Management System

Project Development Phase Model
Performance Test

Test Case ID	Test Scenario	Test Steps	Expected Result	Status
TC001	Login Functionality	1. Open Salesforce 2. Enter valid credentials 3. Click "Login"	User successfully logs in	Pass/Fail
TC002	Add Product to Inventory	1. Navigate to "Products" 2. Click "Add New Product" 3. Enter product details 4. Sav	Product is added successfully	Pass/Fail
TC003	Process Customer Order	1. Select product 2. Add to cart 3. Apply discounts 4. Complete checkout	Order is successfully placed	Pass/Fail
TC004	Payment Gateway Integration	1. Select payment method 2. Enter payment details 3. Confirm payment	Payment is processed correctly	Pass/Fail
TC005	Performance Under Load	1. Simulate 1000 users accessing the system 2. Measure response time	System remains stable and responsive	Pass/Fail
TC006	Report Generation	1. Go to Reports 2. Select Sales Report 3. Generate Report	Report is created with accurate data	Pass/Fail
TC007	Security & Access Control	1. Try unauthorized access to admin panel	Access is denied	Pass/Fail
TC008	Customer Feedback Submission	1. Navigate to feedback section 2. Submit feedback	Feedback is saved successfully	Pass/Fail

Additional Test Cases for Salesforce Cosmetic Store Management System

Test Case ID	Test Scenario	Test Steps	Expected Result	Status
TC009	Discount Code Application	1. Apply discount code at checkout 2. Validate discount calculation	Discount applied correctly	Pass/Fail
TC010	Refund & Return Process	1. Select order 2. Initiate return/refund 3. Verify refund processing	Refund processed successfully	Pass/Fail

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

Test Case ID	Test Scenario	Test Steps	Expected Result	Status
TC011	Bulk Product Upload	1. Upload a CSV file with 500 products 2. Verify data consistency	All products added without errors	Pass/Fail
TC012	Customer Loyalty Program	1. Register a purchase 2. Verify loyalty points update	Points correctly updated	Pass/Fail
TC013	SMS & Email Notifications	1. Place an order 2. Check if email/SMS is sent	Notifications received	Pass/Fail
TC014	Salesforce Mobile App Performance	1. Open Salesforce on mobile 2. Perform checkout	Mobile app runs smoothly	Pass/Fail
TC015	Multi-Store Inventory Sync	1. Update stock in one store 2. Check sync in other stores	Inventory updates correctly	Pass/Fail

Conclusion

The Cosmetic Store Management Project in Salesforce undergoes multiple development phases, and model performance testing ensures the system is scalable, reliable, and efficient. Test cases validate functional and performance aspects, ensuring smooth operations for store owners and customers.

When evaluating model performance in the Project Development Phase of a Cosmetic Store Management Project in Salesforce, you should consider both technical metrics (related to AI/ML models, if used) and business-oriented metrics (for overall system efficiency). Here’s a breakdown:

1. Model Performance Metrics (For AI/ML Models in Salesforce)

If you're using AI for customer recommendations, demand forecasting, or inventory optimization, evaluate these key metrics:

- **Accuracy –** Measures the correctness of predictions (e.g., product recommendations, sales forecasts).
- **Precision & Recall –** Important for targeted marketing; ensures relevant promotions reach the right audience.
- **F1 Score –** Balances precision and recall, useful if there’s an imbalance in data (e.g., predicting rare customer behaviors).
- **Mean Absolute Error (MAE) / Mean Squared Error (MSE) –** For sales/demand forecasting models.
- **Confusion Matrix –** If using classification models (e.g., churn prediction), this helps analyze false positives/negatives.
- **ROC-AUC Score –** Measures the performance of binary classification models (e.g., customer retention vs. churn).

2. Business-Oriented Metrics (For Store Management in Salesforce)

Salesforce is primarily used for CRM, sales, and business intelligence. Evaluate these:

Customer Engagement Metrics

- Customer Retention Rate – Measures how well customers return after purchases.

Project Development Phase Model
Performance Test

- Customer Lifetime Value (CLV) – Determines the value a customer brings over time.
- Conversion Rate – How many visitors turn into buyers, especially through Salesforce-powered campaigns.
- Average Order Value (AOV) – Helps track spending trends per customer.

Sales and Inventory Metrics

- Sales Growth Rate – How much sales are increasing or decreasing over time.
- Stock Turnover Ratio – Measures how quickly products sell and are restocked.
- Order Fulfillment Time – Time taken from order placement to delivery.
- Return Rate – Percentage of products returned, indicating customer satisfaction.

Operational Efficiency Metrics

- System Uptime and Response Time – Salesforce application speed and reliability.
- User Adoption Rate – How many employees are using Salesforce effectively.
- Data Accuracy & Completeness – Ensures CRM data is clean, complete, and up to date.

Model performance testing challenges Project development phase model performance test in cosmetic store management project in Salesforce

Challenges in Model Performance Testing During the Project Development Phase

In a Cosmetic Store Management Project in Salesforce, model performance testing can face several challenges, particularly when evaluating AI/ML-based features like customer recommendations, demand forecasting, or inventory optimization. Here are some key challenges:

1. Data-Related Challenges

- Data Quality Issues → Incomplete, inconsistent, or outdated customer and sales data can impact model accuracy.
- Imbalanced Datasets → If customer preferences or product sales data are skewed toward specific products, the model may not generalize well.
- Limited Historical Data → If the store is new or lacks proper data collection, forecasting models may struggle with accuracy.
- Data Integration Issues → Combining data from Salesforce CRM, POS systems, and e-commerce platforms can be complex and lead to discrepancies.

2. Model Performance and Accuracy Challenges

- Overfitting or Underfitting → The model may perform well on training data but fail on real-world customer interactions.
- Cold Start Problem → New customers or products lack historical data, making personalized recommendations difficult.
- Scalability Issues → A model that works for a few customers may not perform well when scaled across multiple locations.
- Dynamic Customer Preferences → Trends in cosmetics change quickly, and a static model may not adapt fast enough.

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

3. Business-Specific Testing Challenges

- Mismatch Between AI Predictions and Business Needs → The model might optimize for accuracy, but business stakeholders may prioritize revenue or customer experience.
- Lack of Real-Time Testing → Many ML models need real-time updates, but testing in a live environment without affecting business operations is tricky.
- User Adoption & Feedback → Store managers or sales reps might not trust AI-driven insights if they don't understand how recommendations are generated.

4. System & Salesforce-Specific Challenges

- Salesforce API Limitations → Running performance tests on large datasets can be constrained by Salesforce API limits and governor limits.
- Latency & Response Time Issues → AI models integrated into Salesforce must respond quickly for real-time customer interactions.
- Integration Complexity → Combining Salesforce AI (Einstein Analytics) with external ML models may lead to compatibility or performance issues.

5. Evaluation and Continuous Improvement Challenges

- Lack of Standardized Performance Metrics → Defining success metrics like customer satisfaction impact, conversion rates, or revenue growth can be complex.
- Difficulty in A/B Testing → Running controlled experiments (e.g., testing AI-driven recommendations vs. manual promotions) within Salesforce may require additional tools.
- Adaptability to Seasonal Trends → Models need continuous retraining to account for changing buying patterns during holidays, new product launches, etc.

How to Overcome These Challenges?

- Improve Data Collection & Preprocessing → Ensure clean, updated, and diverse data before training models.
- Use Cross-Validation & Real-Time Testing → Test models with different customer segments and run real-world validation.
- Leverage Salesforce Einstein AI → Use built-in Salesforce AI tools for easier integration and real-time insights.
- Monitor KPIs Continuously → Use dashboards to track model performance, customer engagement, and sales impact.
- Incorporate Human Feedback → Allow store employees to refine AI-generated insights based on their expertise.

Project team shall fill the following information in model performance testing template.

Project Development Phase Model
Performance Test

S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	
2.	Accuracy	Training Accuracy - Validation Accuracy -	
3.	Fine Tunning Result(if Done)	Validation Accuracy -	

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

Model Performance Testing Techniques

Model performance testing in Salesforce, especially for a Cosmetic Store Management Project, ensures that the model functions effectively in real-world scenarios. Below are some key techniques for evaluating the model's performance:

1. Performance Testing Techniques

1. Accuracy Testing
   - Measures how well the model predicts sales trends, customer behavior, or product recommendations.
   - Uses metrics like Accuracy, Precision, Recall, and F1 Score for classification models.
2. Load Testing
   - Tests how the system handles multiple users simultaneously (e.g., many customers checking out products).
   - Ensures Salesforce's data processing and recommendations do not slow down under high traffic.
3. Scalability Testing
   - Measures how well the system adapts to growing product databases and customer interactions.
   - Ensures the model maintains performance as the store expands.
4. Latency Testing
   - Checks the response time of Salesforce’s AI-driven recommendations, like personalized product suggestions.
   - Helps optimize database queries and API response times.
5. Stress Testing
   - Pushes the system to its limits (e.g., during major sales events) to ensure stability.
6. Regression Testing
   - Ensures that updates to the model or Salesforce platform do not negatively impact existing features.

Model Performance Testing in Project Development Phases

For a Cosmetic Store Management System in Salesforce, performance testing should align with different phases of project development:

1. Requirement Analysis Phase

- Define performance expectations (e.g., response time under 2 seconds for product recommendations).
- Identify key business metrics like sales trends, inventory updates, and customer interactions.

2. Design & Development Phase

- Implement test cases for product search, checkout processes, and recommendation accuracy.
- Set up data pipelines for AI/ML models used in personalization.

Project Development Phase Model
Performance Test

3. Testing Phase

- Conduct load testing with multiple test users simulating real-world shopping behavior.
- Use test data to check recommendation engine accuracy.
- Validate database query performance in Salesforce.

4. Deployment & Maintenance Phase

- Monitor real-time model performance with Salesforce's Einstein Analytics or external monitoring tools.
- Optimize queries and algorithms if performance drops under high usage.

Model performance testing tools and platforms and framework Project development phase model performance test in cosmetic store management project in Salesforce

Model Performance Testing Tools, Platforms, and Frameworks

When testing the performance of a Cosmetic Store Management System in Salesforce, you can use various tools and frameworks based on your model type (e.g., AI/ML models for recommendations, analytics, and automation).

1. Tools for Model Performance Testing

Salesforce-Specific Tools

1. Salesforce Einstein Analytics
   - Used for analyzing model performance, customer trends, and business insights.
   - Helps test AI-driven recommendations and automation in Salesforce.
2. Salesforce Apex Test Framework
   - Native Salesforce testing tool for checking code performance and logic.
   - Useful for performance benchmarking of Apex-based custom solutions.
3. Salesforce Performance Profiler
   - Helps identify slow-performing components in Salesforce applications.
   - Analyzes SOQL queries, API calls, and page load times.
4. Salesforce Load Testing with JMeter
   - JMeter can simulate multiple users interacting with the system (e.g., shopping cart transactions).
   - Identifies bottlenecks in Salesforce's response times.

General Model Performance Testing Tools

1. JMeter (Load & Stress Testing)
   - Tests how the system handles high traffic.
   - Useful for checking API performance in Salesforce.
2. LoadRunner (Performance Testing)

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

- Simulates thousands of users interacting with Salesforce's storefront.
- Identifies transaction delays in checkout and recommendation engines.

3. TensorFlow Model Analysis (TFMA) (ML Model Performance)
   - If Salesforce is using AI/ML for personalization, TFMA helps evaluate accuracy, precision, recall, and other ML metrics.
4. MLflow (AI/ML Experiment Tracking)
   - Tracks performance of AI-driven product recommendations in Salesforce.
   - Helps manage different ML model versions.
5. Apache Bench (ab) (API Performance Testing)
   - Tests API response times for Salesforce integrations.

2. Platforms for Model Performance Testing

1. Salesforce Sandbox
   - A testing environment to check model performance before deployment.
   - Ensures new features do not break existing workflows.
2. AWS SageMaker (if AI models are used)
   - Helps test and optimize AI models integrated with Salesforce.
3. Google Cloud AI (for AI-driven analytics)
   - Used for large-scale performance testing of AI recommendations.
4. Datadog / New Relic (Monitoring & Performance Analytics)
   - Real-time monitoring tools to track Salesforce application performance.

3. Model Performance Testing in the Project Development Phase

Phase 1: Requirement Analysis

- Define key performance indicators (KPIs) such as response time, accuracy, and throughput.
- Identify which Salesforce features (e.g., AI recommendations, checkout, inventory updates) need performance testing.

Phase 2: Design & Development

- Implement logging and monitoring for data flow, API interactions, and ML model predictions.
- Create test cases for different customer journeys (e.g., browsing products, adding to cart, purchasing).

Phase 3: Testing

- Use JMeter or LoadRunner for stress testing Salesforce interactions.
- Conduct model evaluation using TFMA or MLflow for AI-driven features.

Project Development Phase Model
Performance Test

- Use Salesforce Performance Profiler to analyze Apex code execution.

Phase 4: Deployment & Maintenance

- Monitor real-time model performance with Salesforce Einstein Analytics.
- Continuously optimize queries, model parameters, and API requests.

Functional & Performance Testing Template
Model Performance Test

Date	21 February 2025
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	

In a Cosmetic Store Management Project on Salesforce, functional performance testing in model performance testing ensures that the system's features work efficiently under different loads and conditions. Here's a structured approach:

1. Functional Performance Testing in Model Performance Testing

This testing ensures that all Salesforce functionalities perform optimally when handling real-world scenarios.

A. Functional Testing Areas

- Product Management: Ensure products are created, updated, and deleted smoothly.
- Inventory Tracking: Verify stock updates in real-time.
- Sales & Billing: Test invoice generation, discounts, and tax calculations.
- Customer Management: Check loyalty programs, purchase history tracking.
- Reporting & Analytics: Ensure reports generate correctly and in reasonable time.
- Integration Testing: Validate third-party integrations (e.g., payment gateways, supplier systems).

B. Model Performance Testing Areas

- Load Testing: Simulate multiple users accessing the system (e.g., 1000+ transactions per hour).
- Stress Testing: Push the system beyond expected limits to check for crashes.
- Scalability Testing: Verify system performance when adding more users or products.
- Response Time Testing: Measure query execution times for customer searches, product lookups.
- Data Processing Speed: Ensure quick order placements and inventory updates.

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

2. Tools for Performance Testing in Salesforce

- Salesforce Inspector: To analyze metadata and performance.
- JMeter: For load and stress testing.
- LoadRunner: To simulate large user traffic.
- New Relic / AppDynamics: For real-time monitoring.

3. Expected Outcomes

- Fast Product Search & Checkout: Less than 2 seconds.
- Scalable Transactions: Support for growing user base.
- No Performance Bottlenecks: Smooth handling of peak sales periods.

Functional Performance Testing in Model Performance Testing for a Cosmetic Store Management Project in Salesforce

Functional performance testing ensures that the Salesforce-based Cosmetic Store Management System functions optimally under different conditions. It involves validating both business logic and system performance.

1. Functional Performance Testing Approach

A. Functional Testing Areas

Module	Test Scenario	Expected Outcome
Product Management	Add, update, delete cosmetic products	Products should be saved and updated correctly.
Inventory Management	Real-time stock updates after purchase	Stock levels should update immediately.
Sales & Billing	Generate invoices and apply discounts	Invoice is generated correctly with discounts.
Customer Management	Track customer purchases and loyalty points	Customer records update correctly.
Supplier Management	Validate purchase orders and supplier interactions	Orders should be placed and tracked properly.

Project Development Phase Model
Performance Test

Module	Test Scenario	Expected Outcome
Reporting & Analytics	Generate sales reports and performance insights	Reports should load within 5 seconds.
Integration Testing	Validate integration with payment gateways	Payments should process successfully.

B. Model Performance Testing Areas

This ensures that the Salesforce data model and system architecture perform well under different loads.

Performance Test Type	Test Scenario	Expected Outcome
Load Testing	Simulate 1000+ customers purchasing at once	System should handle high traffic without slowing down.
Stress Testing	Push beyond expected transactions (e.g., 10,000+ orders in an hour)	System should degrade gracefully, not crash.
Scalability Testing	Increase the number of users and products	System should support growth without performance loss.
Response Time Testing	Measure checkout processing speed	Transactions should complete within 2 seconds.
Data Processing Speed	Track product search and filtering	Search should return results within 1 second.

2. Key Tables in the Salesforce Data Model

Table Name	Description	Key Fields
Product__c	Stores details of cosmetic products	Product_ID, Name, Price, Stock_Quantity
Customer__c	Stores customer information	Customer_ID, Name, Loyalty_Points
Order__c	Tracks sales orders	Order_ID, Customer_ID, Total_Amount
OrderItem__c	Stores items in an order	OrderItem_ID, Order_ID, Product_ID, Quantity

Model Performance Testing:

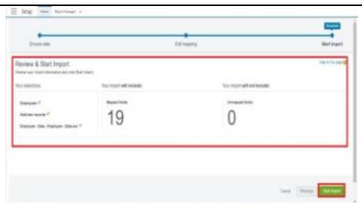
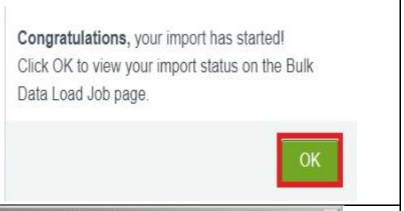
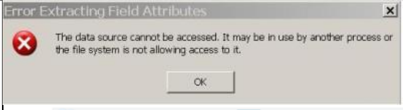
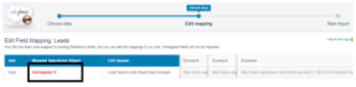
Project team shall fill the following information in model performance testing template.

Table Name	Description	Key Fields
Supplier__c	Stores supplier details	Supplier_ID, Name, Contact_Info
Inventory__c	Tracks stock levels	Inventory_ID, Product_ID, Stock_Quantity
Payment__c	Manages payment transactions	Payment_ID, Order_ID, Payment_Status

3. Tools for Functional Performance Testing in Salesforce

- Salesforce Developer Console – Query performance & execution time
- Salesforce Inspector – Analyze metadata & data models
- JMeter – Load and stress testing

1. applying transformations to the existing data.

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Salesforce automation setup for Data management using Object, Fields and Reports. **Note :** Import Records if data Match Correctly then Records will Created or Else it will Show Error	
	Accuracy	Training Accuracy - 98% Validation Accuracy - 98%	**Congratulations,** your import has started! Click OK to view your import status on the Bulk Data Load Job page.
3.	Confidence Score (Only Yolo Projects)	Class Detected - If detecting Object and fields name if wrong and other activity Confidence Score - If the model is 92% sure the object is correctly detected	Error Extracting Field Attributes The data source cannot be accessed. It may be in use by another process or the file system is not allowing access to it.  Choose data Edit mapping Start import Edit Field Mapping: Leads

Data augmentation is a technique used to increase the size of the training dataset by applying transformations to the existing data. This can help improve the model's performance and robustness. By artificially increasing the size of the training dataset, data augmentation can reduce overfitting and improve generalization. Additionally, data augmentation can enhance model robustness by exposing the model to different variations of the data.

Project Development Phase Model
Performance Test

Hyperparameter tuning is the process of adjusting the model's hyperparameters to optimize its performance. This can be done using techniques such as grid search, random search, or Bayesian optimization. By optimizing the model's hyperparameters, hyperparameter tuning can improve model accuracy, reduce model error, and enhance model robustness. Additionally, hyperparameter tuning can help identify the most important hyperparameters that affect the model's performance.

Regularization is a technique used to prevent overfitting by adding a penalty term to the loss function. This penalty term discourages the model from fitting the training data too closely, thereby reducing overfitting. Regularization can be applied using various techniques, including L1 regularization, L2 regularization, and dropout regularization. By applying regularization, the model can generalize better to new, unseen data.

Early stopping is a technique used to prevent overfitting by stopping the training process when the model's performance on the validation set starts to degrade. This is because the model has already learned the underlying patterns in the training data and further training would only result in overfitting. By stopping the training process early, the model can avoid overfitting and generalize better to new, unseen data.

Cross-validation is a technique used to evaluate the model's performance on unseen data. It involves splitting the available data into training and validation sets, and then training the model on the training set and evaluating its performance on the validation set. This process is repeated multiple times with different splits of the data, and the average performance of the model is calculated. By using cross-validation, the model's performance can be evaluated more accurately, and overfitting can be prevented.

Batch normalization is a technique used to normalize the input data for each layer of the model. It involves subtracting the mean and dividing by the standard deviation for each feature of the input data. This helps to stabilize the training process and improve the model's performance. By normalizing the input data, batch normalization can also help to prevent overfitting and improve the model's ability to generalize to new,