Maintenance, Monitoring, and Troubleshooting

1. Introduction

This section outlines the strategies and activities undertaken to maintain, monitor, and troubleshoot the developed Salesforce-based Customer Relationship Management (CRM) system for a Tours and Travels application. These activities are essential to ensure system reliability, data accuracy, and adherence to business workflows, even though the project was implemented within a Salesforce Developer Org for academic and learning purposes. The applied practices simulate real-world CRM maintenance and quality assurance protocols suitable for global deployment by travel agencies.

2. Objectives

- Ensure data integrity and consistency across custom objects.
- Validate functional and user interface elements.
- Identify and address performance or configuration issues.
- Verify automation logic using real-time scenarios.
- Ensure secure access and error-handling mechanisms.
- Confirm analytical accuracy through dashboards and reports.

3. Testing and Validation

3.1 UI Validation of Features

All core features were tested through the Salesforce Lightning Interface. The validated components include:

- Booking creation and editing
- Travel Package configuration
- Customer and Guest onboarding

- Booking Payments and Feedback submission
- Dynamic layout and field-level visibility

3.2 Functional Testing of Automation

Automated processes such as Flows, Workflow Rules, Process Builders, and Approval Processes were tested using real-time input data. A sample scenario includes:

- When a booking is created with a "Confirmed" status, the system automatically sends a confirmation email to the customer.
  The test was conducted using mock input values such as customer details, booking dates, payment references, and travel package data.

3.3 End-to-End Business Flow Verification

Complete business process workflows were validated from start to finish. This includes:

- Customer creation
- Travel Package selection
- Booking lifecycle
- Payment processing
- Guest management
- Feedback collection

3.4 Data Accuracy Checks

Relationships between custom objects were verified to maintain data consistency. Bookings were tested to ensure proper linkage to Customers, Payments, Travel Packages, and Guests. Data validation rules were applied to prevent illogical or invalid entries, such as booking past dates.

4. Monitoring and Troubleshooting

4.1 Monitoring Scheduled Jobs and Flows

Although no custom Batch Apex or Queueable jobs were created, scheduled Flows and Approval Processes were reviewed for proper execution. Standard Salesforce monitoring tools were used to verify that automation logic executed as intended.

4.2 Debug Logs Review

Standard Salesforce Debug Logs were used to trace Apex Triggers and Process Builders. Error analysis led to the identification of a bug allowing past booking dates, which was resolved by implementing a validation rule.

4.3 Error Handling and Notifications

User-friendly error messages were created to guide users when invalid inputs were submitted. For example, attempting to enter a past booking date triggered a custom validation message. Email alerts were configured to notify users of key booking or payment updates, in alignment with project instructions.

5. Security and Access Management

Security configurations were applied to ensure appropriate data access:

- Role hierarchies and sharing rules were implemented to enforce data visibility.
- Profiles and Permission Sets were created for distinct roles, including Travel Agents and Admins.
- Field-level security was tested to prevent unauthorized data exposure.

6. Validation Rules and Dynamic Forms

6.1 Validation Rules

A key validation rule was created to prevent users from entering booking dates that are earlier than the current date. This rule helped maintain logical accuracy and data consistency.

6.2 Duplicate Rules

Basic duplicate rules were applied to prevent the creation of records with matching key fields, such as customer email addresses or booking reference numbers.

6.3 Dynamic Form Behavior

Dynamic Forms were configured so that certain sections, such as additional booking options or follow-up tasks, became visible only when specific field values were selected, such as a confirmed booking status.

7. Reporting and Dashboards

Custom reports and dashboards were built to reflect system activity and key performance indicators. These visual analytics tools were tested using actual record data to ensure:

- Accurate grouping (e.g., bookings per month)
- Filter configurations (e.g., only confirmed bookings)
- Field-level formulas (e.g., total booking cost, number of guests)

Reports and dashboards included:

- Monthly Booking Trends
- Top Travel Packages
- Revenue Breakdown
- Customer Retention Metrics

8. Cross-Browser and Device Testing

The CRM system was primarily tested using Google Chrome. Compatibility issues were encountered in alternative browsers such as Arc, where objects failed to load properly. Refreshing resolved minor loading issues in Chrome. No testing was performed on mobile devices; however, the design was checked for responsiveness on varying screen sizes within Chrome.

9. Summary and Conclusion

The Maintenance, Monitoring, and Troubleshooting phase played a critical role in ensuring the stability and effectiveness of the Tours and Travels CRM system. Comprehensive testing across the UI, business processes, automation flows, and reports confirmed that the system behaves as expected under real-time use cases. Security configurations, validation logic, and user experience enhancements were implemented based on industry best practices and internship

guidelines. These efforts contributed to delivering a reliable, scalable CRM prototype suitable for modern travel agencies.