SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

Domain

TRAVEL AND HOSPITALITY (BEYOND BOOKING)

Problem Statement

TRAVEL ITINERARY AND COORDINATION PLATFORM

---

1. Preface

1.1 Expected Readership

This Software Requirements Specification (SRS) document is intended for faculty evaluators, project reviewers, system analysts, designers, and developers involved in the academic development of the **Travel Itinerary and Coordination Platform (TICP)**.

The document is written in clear and simple English so that readers with varying technical backgrounds can easily understand the system scope, actors, and requirements.

1.2 Version History

- **Version 1.0 – Initial version of the Requirements Analysis Document**

This version documents the finalized problem statement, actors, features, and system requirements identified during the requirements analysis phase. It establishes a baseline before moving to design-level artifacts such as UML diagrams.

1.3 Summary

This is the first version of the document; therefore, no changes from previous versions exist. Future revisions may include refinements based on faculty feedback, expanded system scope, or changes in project understanding.

---

2. Introduction

2.1 Need for the System

Planning a trip involves multiple complex tasks such as:

- Selecting destinations
- Creating travel schedules
- Managing bookings
- Coordinating with travel guides
- Handling payments
- Tracking expenses

Currently, travelers rely on multiple disconnected platforms for these activities, which leads to confusion, poor coordination, and inefficiency.

A **Travel Itinerary and Coordination Platform** is required to provide a centralized system where travelers can plan trips, coordinate with travel guides, manage bookings, and receive assistance in a structured and organized manner.

2.2 System Overview and Functions

The TICP acts as a centralized platform that supports multiple roles and manages travel planning activities.

The system provides functionality such as:

- Traveler registration and authentication
- Creation and management of travel itineraries
- Destination and activity planning
- Integration with external booking services
- Communication with travel guides
- Payment processing
- Customer support and issue resolution

The system ensures smooth coordination between travelers, guides, administrators, and external services.

2.3 Interaction with Other Systems

The platform interacts with supporting external systems such as:

- Payment gateways
- Hotel and transport booking APIs
- Map and location services
- Notification services

These interactions support bookings, payments, navigation, and communication within the platform.

2.4 Business and Strategic Objectives

The platform aims to:

- Simplify trip planning

- Improve coordination between travelers and guides
- Provide secure payment processing
- Centralize travel-related services
- Reduce dependency on multiple applications
- Enhance user travel experience

---

3. Glossary

Term	Definition
Travel Itinerary and Coordination Platform (TICP)	A centralized system for planning and managing travel activities
Traveler	End user who plans and manages trips
Travel Guide	Professional assisting travelers during trip planning
Customer Care	Support team assisting users
Administrator	Person managing platform operations
External Services	Third-party systems like payment, maps, booking APIs
Itinerary	A planned schedule of travel activities
Booking	Reservation of hotel or transport
Notification	Alerts and updates sent to users

---

4. User Requirements Definition

4.1 User Services (Functional Requirements)

4.1.1 Traveler

- The system shall allow travelers to register and login.
- The system shall allow travelers to create new trips.
- The system shall allow travelers to add destinations.
- The system shall allow travelers to plan daily itineraries.
- The system shall allow travelers to book hotels and transport.
- The system shall allow travelers to make payments.
- The system shall allow travelers to track expenses.
- The system shall allow travelers to communicate with travel guides.
- The system shall allow travelers to view maps and directions.
- The system shall allow travelers to upload travel documents.

4.1.2 Travel Guide

- The system shall allow guides to view assigned travelers.
- The system shall allow guides to suggest itinerary changes.
- The system shall allow guides to add activities.
- The system shall allow guides to communicate with travelers.
- The system shall allow guides to update travel plans.

4.1.3 Customer Care

- The system shall allow customer care to handle user queries.

- The system shall allow customer care to create support tickets.
- The system shall allow customer care to resolve booking issues.
- The system shall allow customer care to process refunds.

4.1.4 Administrator

- The system shall allow administrators to manage users.
- The system shall allow administrators to manage guides.
- The system shall allow administrators to monitor bookings.
- The system shall allow administrators to generate reports.
- The system shall allow administrators to manage external services.

4.2 Non-Functional Requirements

- The system shall be easy to use and user-friendly.
- The system shall respond quickly to user actions.
- The system shall ensure secure handling of user data.
- The system shall be available 24/7 except during maintenance.
- The system shall protect payment information.

4.3 Product and Process Standards

- The system shall follow UML modeling standards.

- The system shall follow secure software engineering practices.
- The system shall maintain data privacy principles.

---

5. System Architecture

5.1 High-Level Architectural Overview

The platform follows a modular architecture consisting of:

- User Management Module
- Itinerary Planning Module
- Booking Management Module
- Guide Coordination Module
- Payment Processing Module
- Support and Notification Module

5.2 Distribution of Functions Across Modules

- Travelers interact with planning and booking modules.
- Guides interact with coordination modules.
- Customer care interacts with support modules.
- Administrators manage all modules.

---

6. System Requirements Specification

6.1 Functional Requirements

- The system shall support user authentication.
- The system shall allow itinerary creation.
- The system shall integrate external booking services.
- The system shall support online payments.
- The system shall enable traveler–guide communication.

6.2 Non-Functional Requirements

- Performance: Handle multiple users simultaneously
- Security: Protect personal and payment data
- Reliability: Maintain accurate booking records
- Usability: Simple and intuitive interface

6.3 External Interfaces

- Web-based user interface
- Payment gateway interface
- Map service interface

- Booking API interface

---

7. System Models

This section contains:

- Use Case Diagram
- Activity Diagrams
- Sequence Diagrams

(These diagrams will be designed based on the following actors and use cases)

Actors in System Models

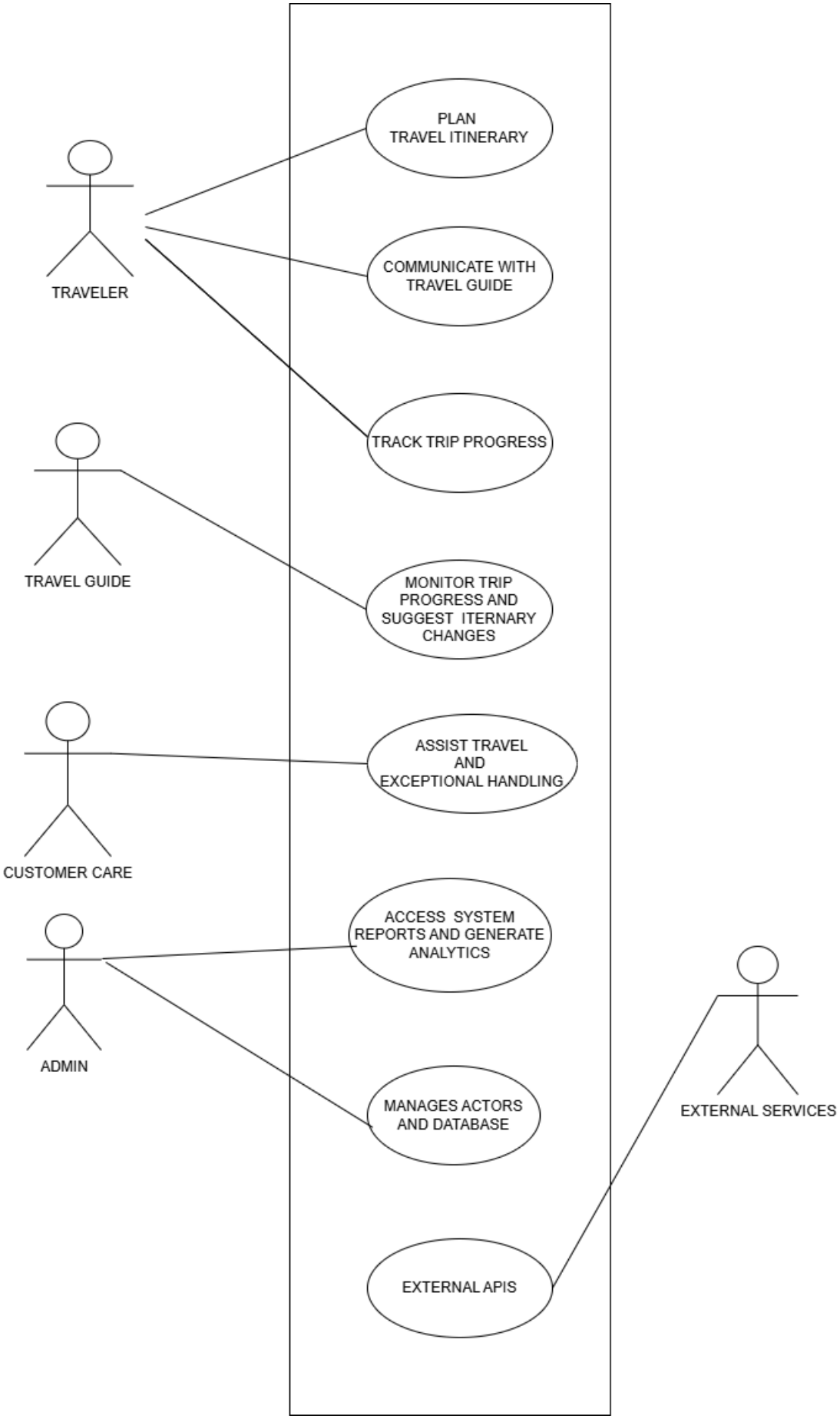
- Traveler
- Travel Guide
- Customer Care
- Administrator
- External Services

Core Use Cases

- Create Itinerary
- Book Services

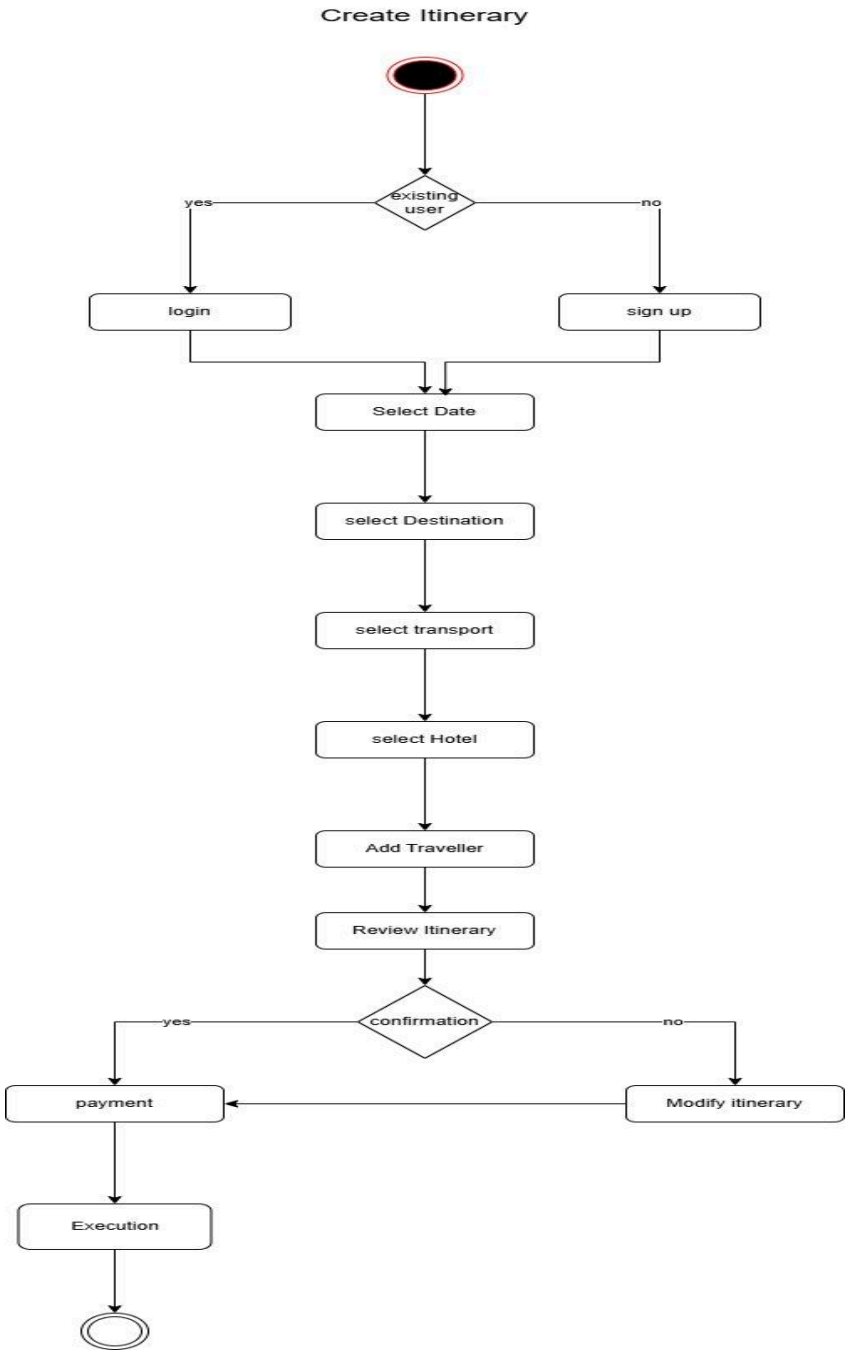
- Make Payment
- Coordinate with Guide
- Handle Support Tickets
- Manage Use

USE CASE DIAGRAM

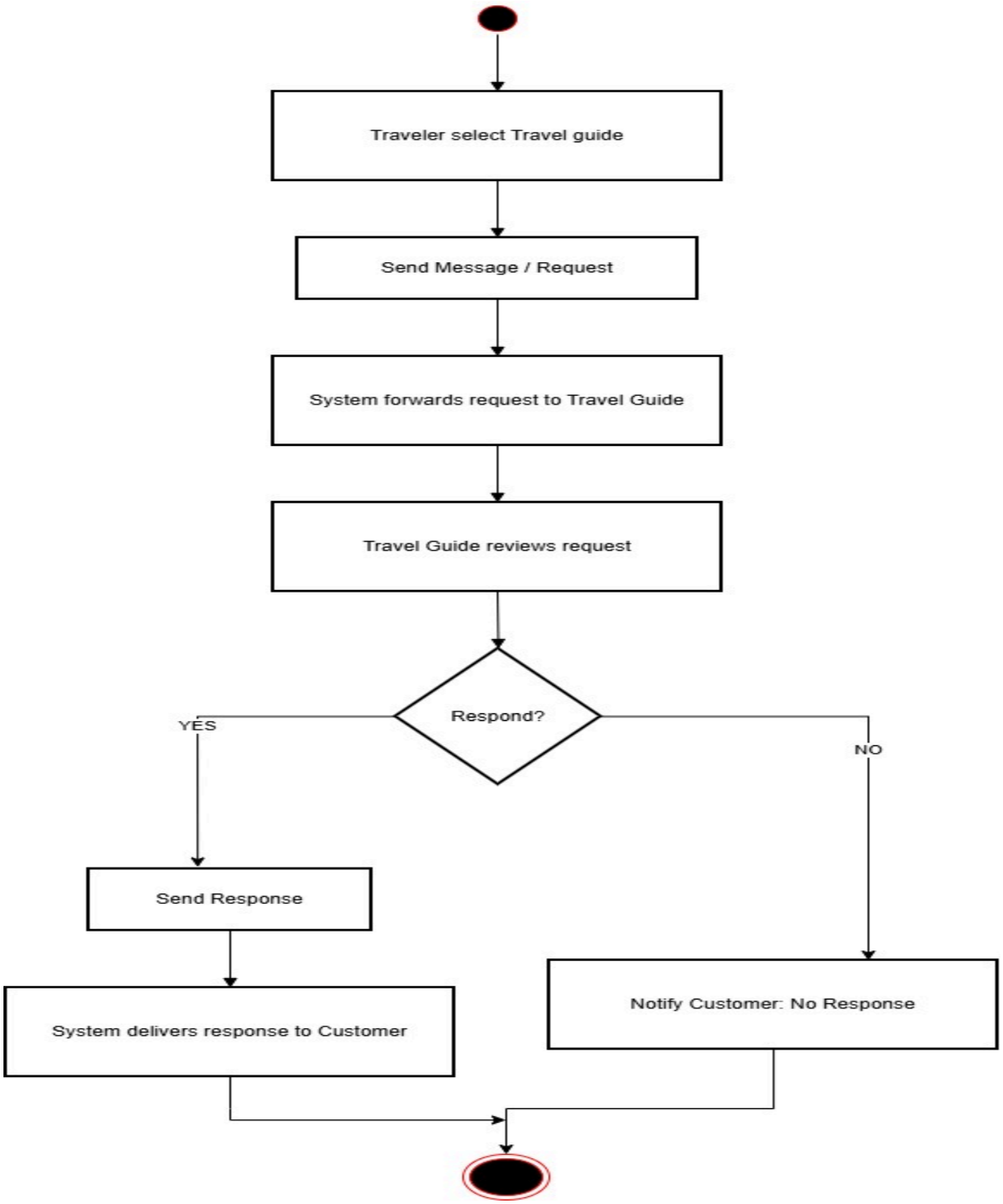

PLAN
TRAVEL ITINERARY
COMMUNICATE WITH
TRAVEL GUIDE
TRACK TRIP PROGRESS
MONITOR TRIP
PROGRESS AND
SUGGEST ITERNARY
CHANGES
ASSIST TRAVEL
AND
EXCEPTIONAL HANDLING
ACCESS SYSTEM
REPORTS AND GENERATE
ANALYTICS
MANAGES ACTORS
AND DATABASE
EXTERNAL APIS
TRAVELER
TRAVEL GUIDE
CUSTOMER CARE
ADMIN
EXTERNAL SERVICES

ACTIVITY DIAGRAMS

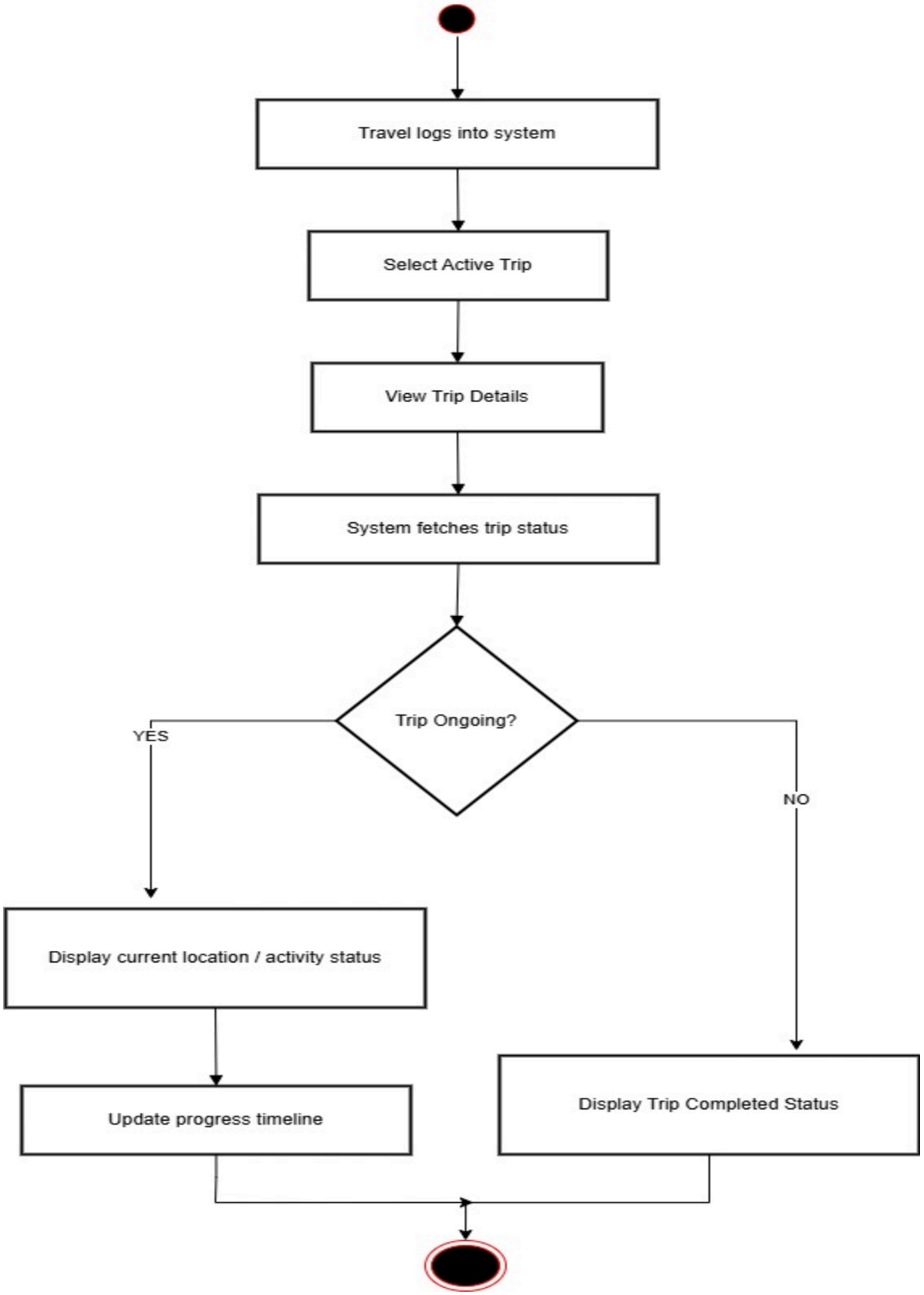
Create Itinerary



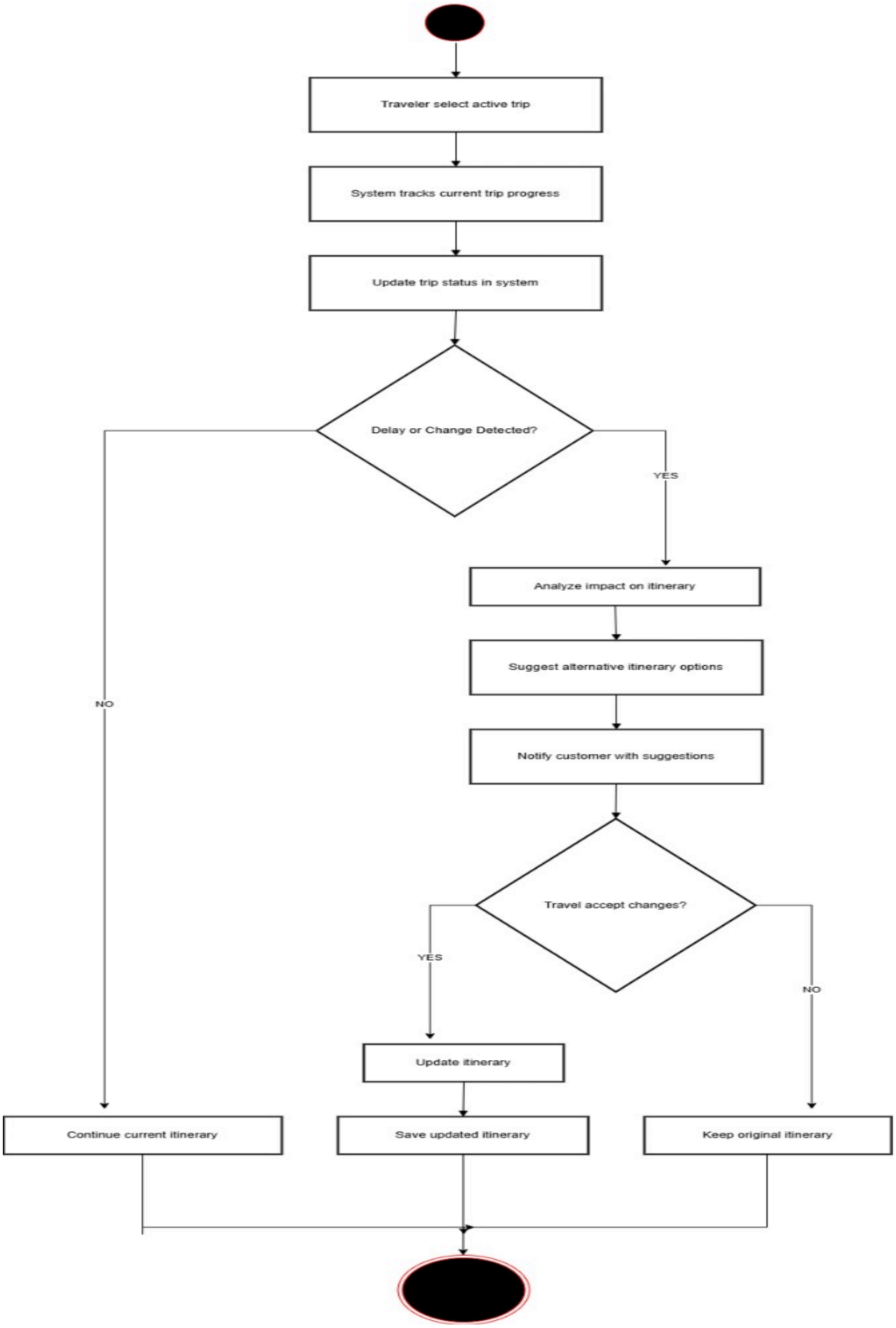
Communicate with travel guide



Track Trip Progress

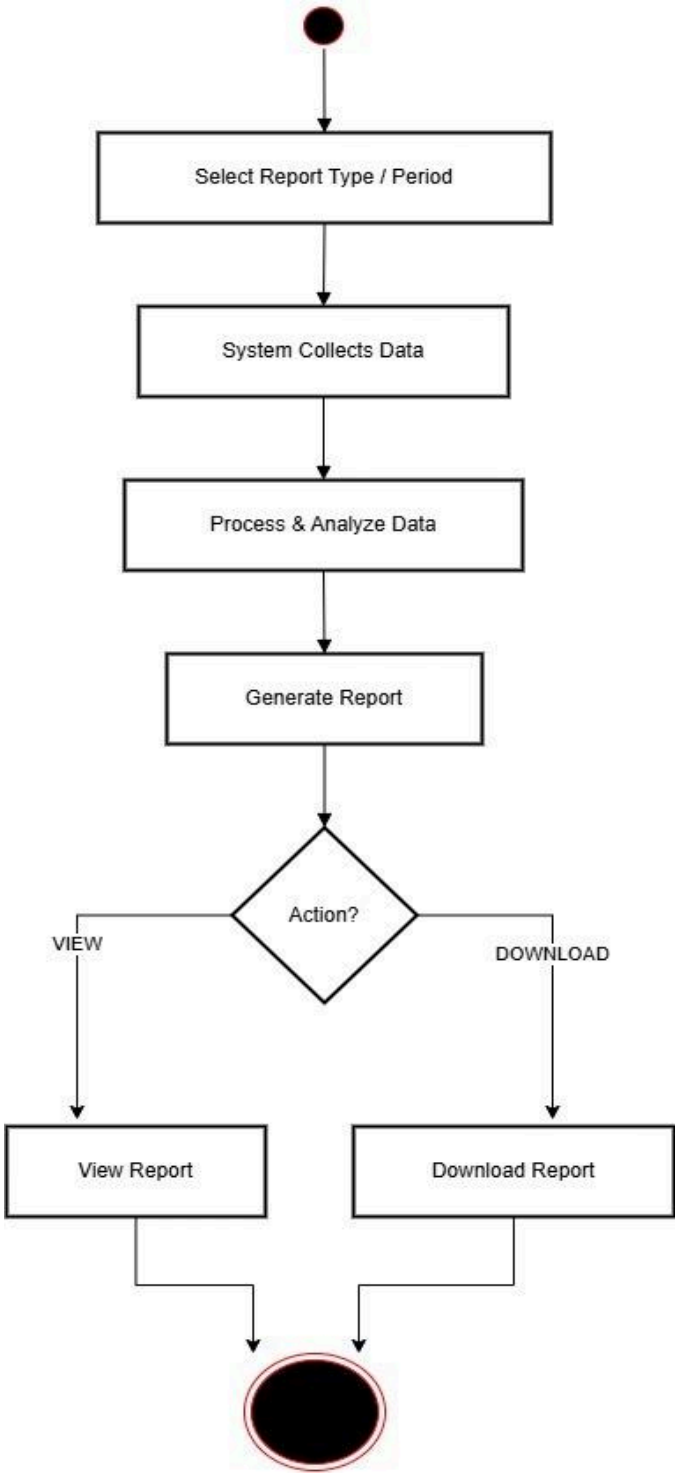


Monitor Trip Progress And Suggest Itinerary changes

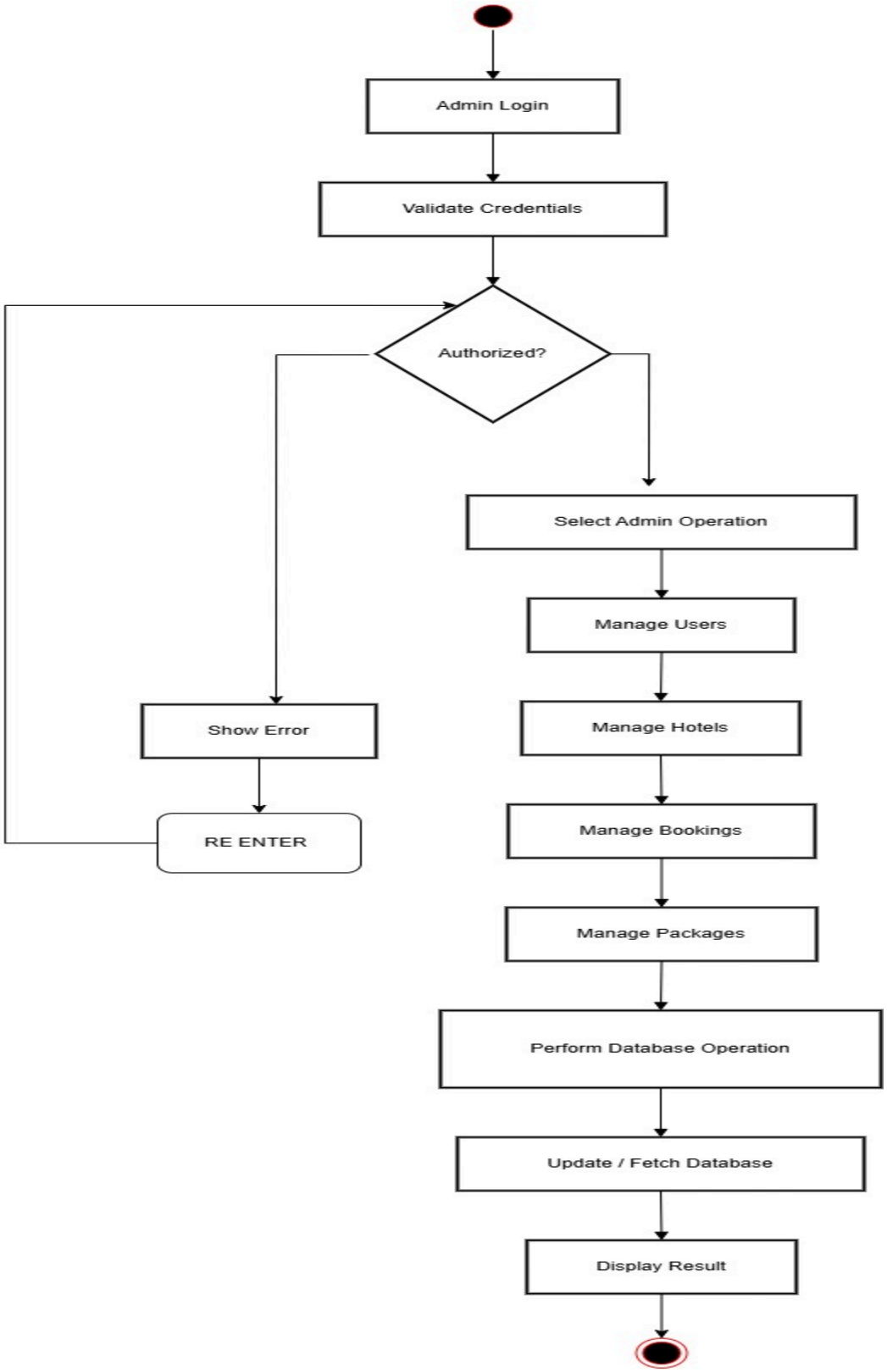

Traveler select active trip
System tracks current trip progress
Update trip status in system
Delay or Change Detected?
YES
NO
Analyze impact on itinerary
Suggest alternative itinerary options
Notify customer with suggestions
Travel accept changes?
YES
NO
Update itinerary
Continue current itinerary
Save updated itinerary
Keep original itinerary

Assist Travel and EXCEPTIONAL HANDLING

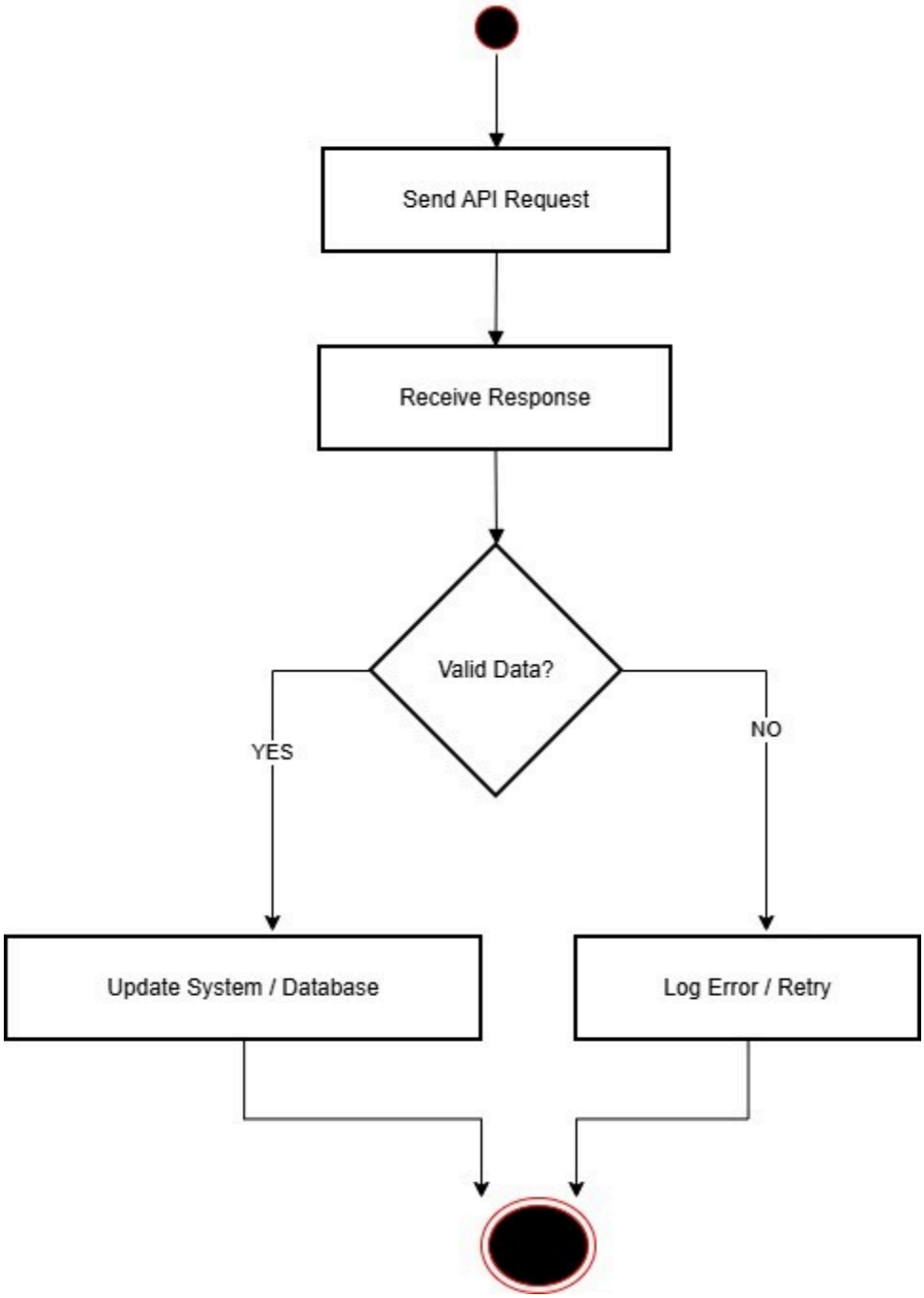
Access System Reports And Generate Analytics



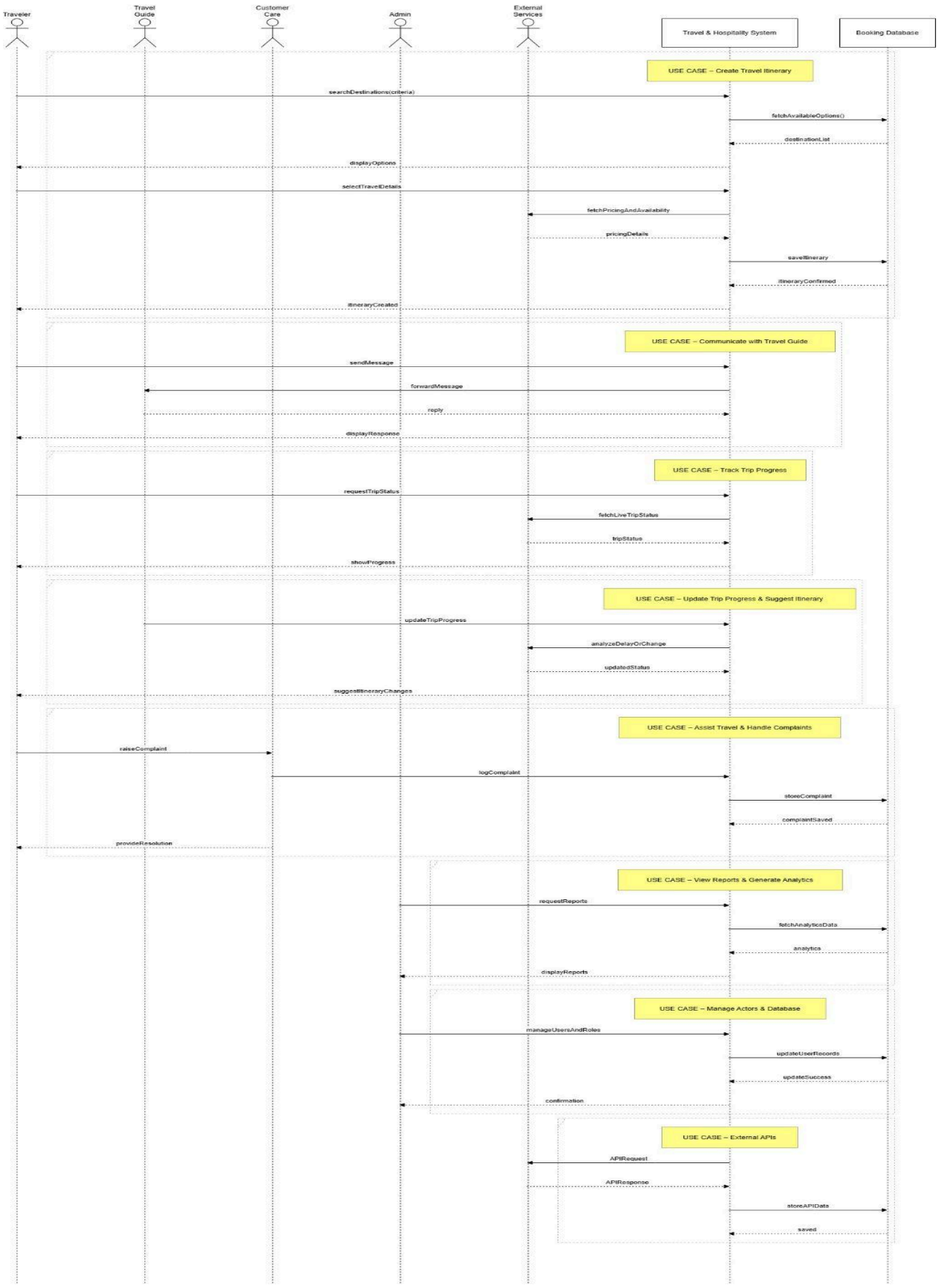
Manages Actors And Database



External API’S



SEQUENCE DIAGRAM


Traveler
Travel Guide
Customer Care
Admin
External Services
Travel & Hospitality System
Booking Database
USE CASE – Create Travel Itinerary
searchDestinations(criteria)
fetchAvailableOptions()
destinationList
displayOptions
selectTravelDetails
fetchPricingAndAvailability
pricingDetails
saveItinerary
itineraryConfirmed
itineraryCreated
USE CASE – Communicate with Travel Guide
sendMessage
forwardMessage
reply
displayResponse
USE CASE – Track Trip Progress
requestTripStatus
fetchLiveTripStatus
tripStatus
showProgress
USE CASE – Update Trip Progress & Suggest Itinerary
updateTripProgress
analyzeDelayOrChange
updatedStatus
suggestItineraryChanges
USE CASE – Assist Travel & Handle Complaints
raiseComplaint
logComplaint
storeComplaint
complaintSaved
provideResolution
USE CASE – View Reports & Generate Analytics
requestReports
fetchAnalyticsData
analytics
displayReports
USE CASE – Manage Actors & Database
manageUsersAndRoles
updateUserRecords
updateSuccess
confirmation
USE CASE – External APIs
APIRequest
APIResponse
storeAPIData
saved

8. System Evolution

8.1 Fundamental Assumptions

- The system is accessed via a web interface.
- The number of users will grow over time.
- New services may be integrated later.

8.2 Anticipated Future Changes

- Mobile application support
- AI-based trip recommendations
- Advanced analytics
- Integration with more travel APIs

---

9. Appendices

9.1 Hardware Requirements

- Basic server infrastructure
- Internet connectivity

9.2 Database Requirements

- Centralized database for users and bookings
- Secure data storage
- Backup and recovery support

---

10. Index

- Actor Index
- Function Index
- Diagram Index
