K'RUNA

DEL 2: FUNCTIONAL SPECIFICATION

This functional specification document covers in depth how our system will meet the business requirements for our client

The overview of this document covers a full set of use case diagrams and narratives which breakdown how each subsystem will operate, process models to illustrate the different process within the system, along with a full set of Activity Diagrams. Complete Logical Data Model, Input and output descriptions, Validation of the functional specifications against the requirements and a review of the complexity marks then concluding with a sign-off by the client and the team.

CLIENT INFORMATION

NdilaTransfers;
A company based in Gauteng and Western Cape with the aim of supplying affordable, professional, safe and reliable shuttle services to the South African tourism industry and also to the corporate industry. Our services are reliant on the following values Respect, Reliability, Integrity, Flexibility and Professionalism. We have the capacity to transport our esteemed clients safely to any respective destination from either Gauteng or Western Cape.

Ndila Transfers
Moving with Times

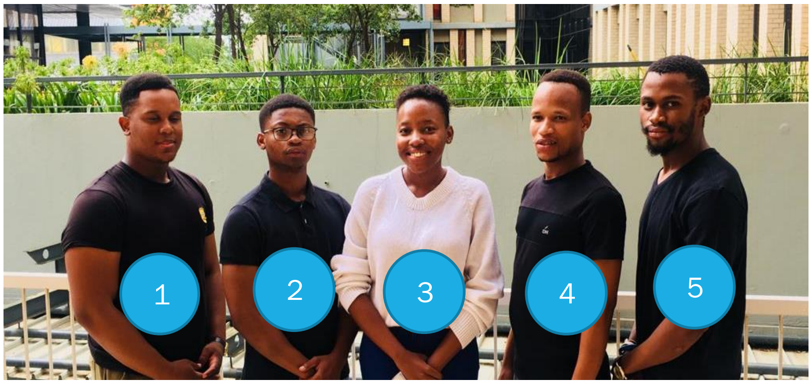
CONTACT

PHONE:
Mr. Thendo
061 402 6227

WEBSITE:
www.ndilatransfers.com

EMAIL:
Thendo@Ndilatransfers.com

THE TEAM : GROUP 20



From left to right:
Kwena Maboka, Mpho Mosotho, Nondumiso Mahlangu (**Team Leader**), Paballo Matabane, Mninikhaya Mavundla

1. u14059867	u14059867@tuks.co.za	071 872 6779
2. u16056762	u16056762@tuks.co.za	082 833 0426
3. u17150312	u17150312@tuks.co.za	076 602 5272
4. u16186835	u16186835@tuks.co.za	076 889 4071
5. u16146060	u16146060@tuks.co.za	076 496 6076

TABLE OF CONTENTS

1. DELIVERABLE 2 INTRODUCTION

INTRODUCTION

The previous deliverable saw the team, K'Runa, present our project proposal. Briefly we will run through what the document contained, as it is the fundamental foundation for Deliverable 2, which is the Functional Specification.

1.1 SYNOPSIS OF THE PREVIOUS DELIVERABLE: DELIVERABLE 1

This is the content that went into the first deliverable

- Client information
- Project Request
- Preliminary Investigation
- Problem Analysis
- Requirement Analysis
- Decision Analysis
- Appendices on client documents, other systems investigated and complexity requirements.
- Sign-off by Client
- Sign-off by Team

CONCLUSION

The following deliverable reveals the Functional Specification of the proposed system. This document will demonstrate what our system will be capable of doing but not focusing on how the system will perform it.
This document is crucial as it serves the purpose of enlightening the stakeholders with an in-depth understanding of the proposed new system, how all the requirements will be met and how the data will be manipulated to assist them to make better business decisions. A description of how the different actors are expected to interact with the system and what will system respond to the actors. This is portrayed through the System Analysis Methods such as a complete set of Use Case Diagrams along with Use Case Narratives to outline the roles of each user and a detailed step by step description of the system requirements will be met. Process Models to breakdown the flow of information such as the data input and output interacting between use cases, actors and entities. UML Modelling to illustrate the activity of the use case, the various decisions and the logical flow to complete the relevant steps towards achieving the requirement. A complete, fully-attributed logical Data Model demonstrating how the data will be structured and which relevant attributes will be stored.

2. USE CASES

INTRODUCTION

This following part of the functional specification document comprises of two main sections. A complete set of UML Use Case Diagrams illustrating the actors and how they play their role in our system. Together with the second section, which is a list of Use Case Narratives detailing each activity and process within each use case. The Use Case Narratives explain each process within the Use Case -step by step detailing how the actors interact with the system and how the system responds also including alternative steps and relevant business rules.

USE CASE DIAGRAM

USE CASE NARRITIVES

Siyaya Travel Assist	
Author (s): Kwena Maboka	Date: 04-22-2019 Version: 2

USE CASE NAME:	Create Access Level	USE CASE TYPE
USE CASE ID:	1.1	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events of an employee creating a new access level into the system database.	
PRE-CONDITION:	Access Level should not already exist in the system database Employee should have authority to create an access level.	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	**Step 1**: Employee would like to create a new access level and selects the option to create access level	Step 2: System checks if user has authority to create an access level
		Step 3: Request Employee to provide details about the new access level such as: “AccessLevel_ID” “AccessLevel_Name”

		"AccessLevel_Description"
	Step 4: Employee enters details about the new access level	Step 5: System verifies details provided by the Employee
		Step 6: System creates new access level into system database
		Step 7: Systems notifies the employee that the new access level has successfully been added.
ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to create an access level. Use case terminates,	
	[Alt Step 5] : System verification of the details provided by the employee fails and return to step 4	
CONCLUSION:	New access level has been created and stored into the system database	
POST-CONDITION:	None.	
BUSINESS RULES	• An employee without the authority may not create an access level	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None.	
ASSUMPTIONS:	• Access level does not exist already in the database	
OPEN ISSUES:	None.	

\

Siyaya Travel Assist	
Author (s): Kwena Maboka	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update Access Level	**USE CASE TYPE** Business Requirements:☐ System Analysis: ☐ System Design: ☐
USE CASE ID:	1.2	
PRIORITY:	High	
SOURCE:		
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events of an employee updating an access level into the system database.	
PRE-CONDITION:	Access Level should not already exist in the system database Employee should have authority to create an access level.	

TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	Step 1: The employee would like to update an existing access level and selects the option to update access level	Step 2: System checks if user has authority to update an access level
		Step 3: System Display all existing access levels And prompts the employee to select the access level they wish to update
	Step 4: Employee selects the access level they'd like to update	Step 5: The system then displays the attributes that the employee can edit AccessLevel_ID" "AccessLevel_Name" "AccessLevel_Description"
	Step 6: Employee updates the details they'd like update: AccessLevel_ID" "AccessLevel_Name" "AccessLevel_Description"	
	Step 7: Employee selects the save option to save the updated details into the system database	**Step 8: System verifies the details updated by the employee**
		Step 9: System then saves the updated access level details into the system database.
		Step 10: System notifies the employee that the update has been successfully.
ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to update an access level. Use case terminates,	
	[Alt Step 8] : System verification of the details provided by the employee fails and return to step 4	
CONCLUSION:	Access level has been updated and stored into the system database	
POST-CONDITION:	None.	
BUSINESS RULES	• An employee without the authority may not update an access level	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None.	
ASSUMPTIONS:	• Access level does exist already in the database	
OPEN ISSUES:	None.	

USE CASE NAME:	Search Access Level	**USE CASE TYPE** Business Requirement:☐ System Analysis: ☐ System Design: ☐
USE CASE ID:	1.3	
PRIORITY:	High	
SOURCE:		
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	- None	
OTHER INTERESTED STAKEHOLDERS:	- None	
DESCRIPTION:	This use case describes the events where an employee is looking for a certain Access level. It involves the employee entering the access level name so that the system can return the access level that has been searched for by the employee.	
PRE-CONDITION:	Access level already exists in the system database	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	**Step 1**:The employee selects the ‘search access level’ option	Step 2: System checks if the employee has authority to search access levels
		Step 3: System requests the employee to enter the details of the access level to search for AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description”
	Step 4: Employee enters the access level details for which they would like to search for: AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description”	Step 5: System validates the details entered by the employee
		Step 6: System then checks system database for a access level that the employee is searching for
		Step 7: System then displays the access level that has been searched for
ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to search an access level. Use case terminates,	
	[Alt Step 5] : System verification of the details provided by the employee fails and return to step 3	
	[Alt Step 7]: System notifies the employee that the search could not find the access level they have searched for	

CONCLUSION:	Access level search has been found in the system database
POST-CONDITION:	None.
BUSINESS RULES	• An employee without the authority may not search for an access level
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None.
ASSUMPTIONS:	• Access level does exist already in the database
OPEN ISSUES:	None.

Siyaya Travel Assist	
Author (s): Kwena Maboka	Date: 04-22-2019 Version: 2

USE CASE NAME:	Register New Employee	USE CASE TYPE Business Requirements:□ System Analysis: □ System Design: □
USE CASE ID:	2.1	
PRIORITY:	High	
SOURCE:		
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	Manager	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events where a manager would like to register a new employee into the system database. This involves the entering the new employee details into the system and verifying the entered details.	
PRE-CONDITION:	Employee does not already exist in the current system database.	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	**Step 1**: Manager selects the register new employee option	Step 2: System checks whether the manager has authority to register a new employee
		Step 3: System then request the manager to enter all employee details into the system: "EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID"

	Step 4: Manager enters all required employee details to register a new employee “EMPID” “EMP_Name” “EMP_Surname” “AuditID” “EMP_TypeID” “EMP_EmailAddress” “EMP_Contact” “EMP_Password” “EMP_IDNum” “TitleID” “GenderID”	
	Step 5: Manager selects the save option to save the entered details	**Step 6: System then verifies all entered details by the manager**
		Step 7: System then adds the entered employee details into the system database.
		Step 8: System notifies the manager that the new employee has successfully been added.
ALTERNATE COURSES:	[Alt Step 2]: [Alt Step 2]: System notifies the employee that they do not have the authority to register a new employee. Use case terminates.	
	[Alt Step 6]: System verification of the details provided by the manager fails and return to step 3	
CONCLUSION:	The manager has successfully register a new employee into the system	
POST-CONDITION:	None	
BUSINESS RULES	• Only the owner and the managers are allowed to register new employees	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

USE CASE NAME:	Update Employee	**USE CASE TYPE**
USE CASE ID:	2.2	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	

OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes the events where a manager would like to update employee’s details in the system database. This involves the manger selecting the employee they wish to update as well as entering the new details of this employee.	
PRE-CONDITION:	An employee should not already exist in the system database The owner and the managers should have authority to update an employee’s details	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	Step 1: The employee would like to update an existing employee and selects the option to update employee	Step 2: System checks if user has authority to update an employee
		Step 3: System Display all existing employees And prompts the employee to select the employee they wish to update
	Step 4: Employee selects the employee they’d like to update	Step 5: The system then displays the attributes that the employee can edit EMP_Name” “EMP_Surname” “EMP_TypeID” “EMP_EmailAddress” “EMP_Contact” “EMP_Password” “EMP_IDNum” “TitleID” “GenderID”
	Step 6: Employee updates the details they’d like update EMP_Name” “EMP_Surname” “EMP_TypeID” “EMP_EmailAddress” “EMP_Contact” “EMP_Password” “EMP_IDNum” “TitleID” “GenderID”	
	Step 7: Employee selects the save option to save the updated details into the system database	**Step 8: System verifies the details updated by the employee** EMP_Name” “EMP_Surname” “EMP_TypeID” “EMP_EmailAddress” “EMP_Contact” “EMP_Password” “EMP_IDNum” “TitleID” “GenderID”

		Step 9: System then saves the updated employee details into the system database.
		Step 10: System notifies the employee that the update has been successfully.
ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to update an employee's details. Use case terminates,	
	[Alt Step 8] : System verification of the details provided by the employee fails and return to step 4	
CONCLUSION:	Employee has been updated and stored into the system database	
POST-CONDITION:	None.	
BUSINESS RULES	• An employee without the authority may not update an employee's details	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None.	
ASSUMPTIONS:	• Employee does exist already in the database	
OPEN ISSUES:	None.	

Siyaya Travel Assist	
Author (s): Kwena Maboka	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Search Employee	**USE CASE TYPE**
USE CASE ID:	2.3	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Manager	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes the events where the manager is looking for a specific employee. It involves the manager entering the employee details so that the system can return the employee that has been searched for by the manager.	
PRE-CONDITION:	Employee already exists in the system database	
TRIGGER:	Manager	
TYPICAL COURSE OF EVENTS:	**Step 1:**The manager selects the 'search employee' option	**Step 2:** System checks if the manager has authority to search employees

		Step 3: System requests the manager to enter the details of the employee to search for EMP_Name" "EMP_Surname" "EMP_TypeID
	Step 4: Manager enters the employee details for which they would like to search for EMP_Name" "EMP_Surname" "EMP_TypeID	**Step 5:** System validates the details entered by the manager
		Step 6: System then checks system database for the employee that the manager **is searching for**
		Step 7: System then displays the employee that has been searched for "EMP_Name" "EMP_Surname" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID"
ALTERNATE COURSES:	[Alt Step 2]: System notifies the manager that they do not have the authority to search an employee. Use case terminates,	
	[Alt Step 5] : System verification of the details provided by the manager fails and return to step 3	
	[Alt Step 7]: System notifies the manager that the search could not find the employee they have searched for	
CONCLUSION:	Employee search has been found in the system database	
POST-CONDITION:	None	
BUSINESS RULES	None.	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	• Employee does exist already in the database	
OPEN ISSUES:	•	
	None.	

USE CASE NAME:	Create New Employee Type	USE CASE TYPE
USE CASE ID:	3.1	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	Manager	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events where a manager would like to register a new type into the system database. This involves the entering the new employee type details into the system and verifying the entered details.	
PRE-CONDITION:	Employee type does not already exist in the current system database.	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	**Step 1**: Manager selects the create new employee type option	Step 2: System checks whether the manager has authority to create a new employee type
		Step 3: System then request the manager to enter all employee type details into the system “EMP_TypeID” “EMP_TypeName” “EMP_TypeDes”
	Step 4: Manager enters all required employee type details to register a new employee “EMP_TypeID” “EMP_TypeName” “EMP_TypeDes”	
	Step 5: Manager selects the save option to save the entered details	**Step 6: System then verifies all entered details by the manager**
		Step 7: System then adds the entered employee type details into the system database.
		Step 8: System notifies the manager that the new employee type has successfully been added.

ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to register a new employee type. Use case terminates.	
	[Alt Step 6]: System verification of the details provided by the manager fails and return to step 3 EMP_TypeID" "EMP_TypeName" "EMP_TypeDes"	
CONCLUSION:	The manager has successfully register a new employee type into the system	
POST-CONDITION:	None	
BUSINESS RULES	• Only the owner and the managers are allowed to register new employee type	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Kwena Maboka	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update Employee Type	USE CASE TYPE Business Requirements:☐ System Analysis: ☐ System Design: ☐
USE CASE ID:	3.2	
PRIORITY:	High	
SOURCE:		
PRIMARY BUSINESS ACTOR	Employee	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes the events where a manager would like to update an employee type details in the system database. This involves the manger selecting the employee type they wish to update as well as entering the new details of this employee type	
PRE-CONDITION:	An employee type should not already exist in the system database The owner and the managers should have authority to update an employee type details	
TRIGGER:	Employee	
TYPICAL COURSE OF EVENTS:	**Step 1**: The employee would like to update an existing employee type and selects	Step 2: System checks if user has authority to update an employee type

	the option to update employee type	
		Step 3: System Display all existing employee types And prompts the employee to select the employee type they wish to update
	Step 4: Employee selects the employee type they’d like to update	Step 5: The system then displays the attributes that the employee can edit EMP_TypeName” “EMP_TypeDes”
	Step 6: Employee updates the details they’d like update “EMP_TypeName” “EMP_TypeDes”	
	Step 7: Employee selects the save option to save the updated details into the system database	**Step 8: System verifies the details updated by the employee**
		Step 9: System then saves the updated employee type details into the system database. EMP_TypeName” “EMP_TypeDes”
		Step 10: System notifies the employee that the update has been successfully.
ALTERNATE COURSES:	[Alt Step 2]: System notifies the employee that they do not have the authority to update an employee type details. Use case terminates,	
	[Alt Step 8] : System verification of the details provided by the employee fails and return to step 4 “EMP_TypeName” “EMP_TypeDes	
CONCLUSION:	Employee type has been updated and stored into the system database	
POST-CONDITION:	None.	
BUSINESS RULES	• An employee without the authority may not update an employee type details	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None.	
ASSUMPTIONS:	• Employee type does exist already in the database	
OPEN ISSUES:	None.	

USE CASE NAME:	Search Employee Type	USE CASE TYPE
USE CASE ID:	3.3	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Manager	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes the events where the manager is looking for a specific employee type. It involves the manager entering the employee type details so that the system can return the employee type that has been searched for by the manager.	
PRE-CONDITION:	Employee type already exists in the system database	
TRIGGER:	Manager	
TYPICAL COURSE OF EVENTS:	**Step 1**:The manager selects the 'search employee type option	Step 2: System checks if the manager has authority to search employee type
		Step 3: System requests the manager to enter the details of the employee type to search for "EMP_TypeID" "EMP_TypeName" "EMP_TypeDes"
	Step 4: Manager enters the employee type details for which they would like to search for "EMP_TypeID" "EMP_TypeName" "EMP_TypeDes"	Step 5: System validates the details entered by the manager
		Step 6: System then checks system database for an employee type **that the** manager **is searching for**
		Step 7: System then displays the employee type that has been searched for "EMP_TypeID" "EMP_TypeName" "EMP_TypeDes"
ALTERNATE COURSES:	[Alt Step 2]: System notifies the manager that they do not have the authority to search an employee type. Use case terminates,	

	[Alt Step 5] : System verification of the details provided by the manager fails and return to step 3 “EMP_TypeID” “EMP_TypeName” “EMP_TypeDes”
	[Alt Step 7]: System notifies the manager that the search could not find the employee type they have searched for
CONCLUSION:	Employee type search has been found in the system database
POST-CONDITION:	None
BUSINESS RULES	None.
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None
ASSUMPTIONS:	• Employee type does exist already in the database
OPEN ISSUES:	• None.

Siyaya Travel Assist	
Author (s): Kwena Maboka	Date: 04-22-2019 Version: 2

USE CASE NAME:	Remove Employee Type	USE CASE TYPE
USE CASE ID:	3.4	Business Requirements:☐
PRIORITY:	High	System Analysis: ☐
SOURCE:		System Design: ☐
PRIMARY BUSINESS ACTOR	Manager	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events where a manager would like to remove an employee type from being used for any new transactions. This involves the manager selecting the employee type to remove and confirming that the employee type should not be accessible any longer.	
PRE-CONDITION:	The employee type should already exist in the system database	
TRIGGER:	Manager/Owner	
TYPICAL COURSE OF EVENTS:	**Step 1**: The manager selects the ‘remove employee type’ option	Step 2: System checks to see if the manager has authority to remove an employee type
		Step 3: System then displays all the employee types in the database “EMP_TypeID” “EMP_TypeName” “EMP_TypeDes”
		Step 4: System prompts the manager to select the employee type they wish to remove

	Step 5: Manager selects the employee type they would like to remove “EMP_TypeID” “EMP_TypeName” “EMP_TypeDes”	**Step 6: System requests confirmation that the selected employee type is the correct one to be removed**
	Step 7: Manager selects the correct option	Step 8: System then disables the employee type to be used again in the database
		Step 9: System then displays a confirmation message to let the manager that the removal has been successful
ALTERNATE COURSES:	[Alt Step 2]: System finds the manager to not have authority to remove an employee type. Terminates use case	
	[Alt Step 8]: System fails to remove the employee type and returns to step 4	
CONCLUSION:	The manager selected employee type has successfully been removed from the system	
POST-CONDITION:	Employee Type still exists In previous records	
BUSINESS RULES	• None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• Employee Type already exists in the system database	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019 Version: 2**

USE CASE NAME:	Search driver	**USE CASE TYPE**
USE CASE ID:	4.1	Business Requirements:☐ □ System Analysis:☐ □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes an event where booking consultant wants to search a driver on the system and enters the Driver name they want to search and retrieve the details of the driver such as the driver name, surname, license number and license type from the Driver table in the database.	

PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant selects the Search Driver option on the system	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant selects the Search Driver option of the driver on the system	**Step 2:** The system displays a page for the booking system to enter the search criteria.
	Step 3: The booking consultant enters the characters of the name of the driver they want to search	**Step 4:** The system verifies the character input
		Step 5: The system reads from the **Driver** table the **Driver_Name**
		Step 6: The system compares the characters received with the characters in the **Driver** table in the **Driver_Name** field until a match is found
		Step 7: The system retrieves the **Driver_ID,** **Driver_Name,** **Driver_Surname,** **Driver_LicenceNumber,** **Driver_LincenceType,** **EMPID** information related to the characters given from the **Driver** Table
		Step 8: The system displays the driver information retrieved
ALTERNATE COURSES:	Alt step 3: The characters given did not match system requirements and an error message is displayed	
	Alt step 6: A match is not found and an alert message is displayed	
CONCLUSION:	The system retrieves the information related to the Driver characters given	
POST-CONDITION:	The driver information searched has to be displayed for the booking consultant	
BUSINESS RULES	● The booking consultant and the team leader work with the driver	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist

Author (s): Nondumiso Mahlangu

Date: 04-22-2019
Version: 2

USE CASE NAME:	Check driver availability	**USE CASE TYPE**
USE CASE ID:	4.2	Business Requirements:□ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● Owner ● Driver supervisor	
DESCRIPTION:	This use case describes an instance where the booking consultant wants to confirm a trip and checks the whether or not a driver is available. The booking consultant can then select the check availability option and the system returns a list of drivers which are available.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant wants to check whether or not a driver is available	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant wants to check whether or not a driver is available and selects the Check Driver option	**Step 2:** The system retrieves the **Trip Date** **TripTime** From the information given through use case
		Step 3: The system reads from the Slot table the: **Slot_Date** **Slot_Time**
		Step 4: The system compares the **Slot_Date** and **Slot_Time** To the **Trip Date** and **Trip Time** and returns a list of the **Driver_ID**'s and **Driver_Name** whose **Slot_Date** and **Slot_Time** did not match the **Trip Date** and **Trip Time**
		Step 5: The system displays the list of drivers that have been returned
ALTERNATE COURSES:	Alt step 4: There are no drivers available on the given date and time and a message is displayed that there are drivers available and the use case ends	
CONCLUSION:	The system displays the drivers who are available	
POST-CONDITION:	The schedule should be up to date at all times	
BUSINESS RULES	● None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

USE CASE NAME:	Assign driver to trip	**USE CASE TYPE**
USE CASE ID:	4.3	Business Requirements:☐ □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● Team leader ● Driver supervisor	
DESCRIPTION:	This use case describes the event when the booking consultant wants to assign a specific driver to a vehicle. They check the availability of a driver and use case 4.2 Check Driver Availability is invoked. The booking consultant then chooses an available driver The name of the driver is then added on the slot.	
PRE-CONDITION:	There has to be a trip to assign a driver to.	
TRIGGER:	The booking consultant wants to assign a specific driver to a trip	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to assign a specific driver to a trip and selects assign driver option	**Step 2:** Invoke use case 4.2
	Step 3: The booking consultant selects **Driver_Name** to be booked	**Step 4**: The system verifies the information received
		Step 5: The system writes the **Driver_Name** to the **Slot.**
		Step 6: The system displays a message confirming the assignment.
ALTERNATE COURSES:	Alt step 2: The Use case returns a negative result, a message is displayed to the booking consultant and the use case ends	
CONCLUSION:	The slot is updated	
POST-CONDITION:	If a driver is unavailable, they cannot be assigned to another trip	
BUSINESS RULES	● A vehicle has to be assigned to a booked trip before a driver can be assigned to it	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist		
Author (s): Nondumiso Mahlangu		Date: 04-22-2019 Version: 2
USE CASE NAME:	Outsource Driver	**USE CASE TYPE**
USE CASE ID:	4.4	Business Requirements:☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Team leader • Driver supervisor	
DESCRIPTION:	This use case describes the event when the booking consultant wants to assign a specific driver to a vehicle. The booking consultant prompts the system to display the outsourced driver who is then added on the slot.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant wants to assign a specific driver to a trip	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to assign a driver to a trip and selects Outsource Driver option	**Step 2:** The system reads and displays information of: **OutSoucreDriver_ID** **OutSource_Driver_Name** **OutSource_Driver_Surname** **LicenceType_ID** from the **OutSourced_Driver** table
	Step 3: The booking consultant selects the **driver** desired to be booked	**Step 4:** The system writes the **OutSoucreDriver_ID** and **OutSource_Driver_Name** to the **Slot.**
		Step 5: The system displays a message confirming the assignment
ALTERNATE COURSES:	None	
CONCLUSION:	The slot is updated	
POST-CONDITION:	If a driver is unavailable, they cannot be assigned to another trip	
BUSINESS RULES	• There has to be a pre-existing booking before a vehicle can be assigned	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

USE CASE NAME:	Add vehicle	**USE CASE TYPE**
USE CASE ID:	5.1	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● None	
DESCRIPTION:	This use case describes when the booking consultant wants to add another vehicle onto the system database. They retrieve all the relevant vehicle information and a new vehicle is saved onto the system database.	
PRE-CONDITION:	The vehicle cannot be an existing vehicle in the system	
TRIGGER:	The booking consultant wants to add another vehicle	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to add another vehicle and selects the Add Vehicle option.	**Step 2**: The system displays a page where the booking consultant can add the required details of the vehicle such as: **Vehicle_Model** **Vehicle_Make_Name** **Vehicle_Make_Description** **Vehicle_Colour** **ServiceProvider**
	Step 3: The booking consultant adds the necessary details of the new vehicle	**Step 4**: The system verifies the information gathered
		Step 5: The system creates a new **Vehicle_ID, VehicleMake_ID, VehicleMaintenance_ID** by retrieving the last line in **Vehicle_ID, VehicleMake_ID, VehicleMaintenance_ID** in the **Vehicle, VehicleMake, VehicleMaintenance** table, increments by 1 and assigns a new **Vehicle_ID VehicleMake_ID, VehicleMaintenance_ID**
		Step 6: The system adds the new **Vehicle_ID VehicleMake_ID, VehicleMaintenance_ID** and

		the given **Vehicle_Model, Vehicle_Colour_Description Vehicle_Make_Name, Vehicle_Make_Description Vehicle_ID ServiceProvider** into the **Vehicle, Vehicle_Make, VehicleMaintenance** tables
		Step 7: The system displays a message that a new vehicle has been added to the system
ALTERNATE COURSES:	Alt step 4: The booking consultant made an error with the entered information and an error message is provided	
CONCLUSION:	A new vehicle is added onto the system database	
POST-CONDITION:	The vehicle should be ready to book and use	
BUSINESS RULES	● None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Search vehicle	**USE CASE TYPE**
USE CASE ID:	5.2	Business Requirements: □
PRIORITY:	High	System Analysis:□
SOURCE:	Ndila Transfers	System Design: □
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● Team Leader ● Driver supervisor	
DESCRIPTION:	This use case describes an event where booking consultant wants to search for a vehicle on the system and enters the vehicle name they want to search and retrieve the details of the vehicle from the Vehicle table in the database.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant selects the Search Vehicle option on the system	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant selects the Search Vehicle option of the driver on the system	**Step 2**: The system displays a page for the booking system to enter the search criteria.

	Step 3: The booking consultant enters the characters of the vehicle they want to search	**Step 4:** The system verifies the character input
		Step 5: The system reads from the **Vehicle** table the **Vehicle_Model**
		Step 6: The system compares the characters received with the characters in the Vehicle table in the **Vehicle_Model** field until a match is found
		Step 7: The system retrieves the **Vehicle_ID** **Vehicle_Model, the VehicleMake_ID,** **VehicleMaintenance_ID,** **VehicleColour_ID,** **VehicleLicencePlates** information related to the characters given from the **Vehicle** Table
		Step 8: The system displays the retrieved information.
ALTERNATE COURSES:	Alt step 4: The characters given did not match system requirements and an error message is displayed	
	Alt step 6: A match is not found and an alert message is displayed	
CONCLUSION:	The system retrieves the information related to the Vehicle characters given	
POST-CONDITION:	The vehicle information searched has to be displayed for the booking consultant	
BUSINESS RULES	● The booking consultant and the team leader work with the driver	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Confirm Trip	**USE CASE TYPE**
USE CASE ID:	5.3	Business Requirements:□ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● Team Leader ● Driver supervisor ● Passenger	

	● Driver	
DESCRIPTION:	This use case describes the event where the trip information such as the driver name, driver surname, departure point and destination point as well as the vehicle information is sent to the relevant people to inform them of the trip details. The system sends an SMS to the passenger and the driver	
PRE-CONDITION:	The trip has to be an existing trip	
TRIGGER:	The booking consultant wants to send out confirmation messages	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to send out a confirmation message and clicks on the Confirm booking option	Step 2: The system retrieves the current date and time and increments the day of the current day by 1
		Step 3: The system reads from the **Slot** table in the database the following fields: **BookingReference_ID** **Passenger_ID,** **Driver_ID,** **DeparturePoint,** **DestinationPoint** **Time** for every line
		Step 4: The system retrieves the **Passenger_Tel** in the **Passenger** table and the **Driver_Contact** **Driver_Name** from the **Driver** table
		Step 5: The system sends a confirmation message to the contact number of the **Passenger** with the: **Booking reference,** **Driver name,** **Departure point,** **Destination point and** **Time** of trip via SMS
		Step 6: The system runs a function that reads the first **Driver_ID** **Driver_Contact** **AddedToLog** line in **Slot** Table, compares it to the rest and stores it in a temporary function
		Step 7: The system sends a message to the **Driver_Contact** with the following: **Passenger_Name,** **Departure point,** **Destination Point** **Time** of trip via SMS

		Step 8: The system displays a message to that the trips have been confirmed
ALTERNATE COURSES:	None	
CONCLUSION:	A confirmation message of the trip is sent to the Passenger and the Driver	
POST-CONDITION:	The driver and the passenger must be aware of the confirmation of the trip	
BUSINESS RULES	● The booking consultant sends out the trips at the end of a business day	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Check Vehicle Availability	**USE CASE TYPE**
USE CASE ID:	5.4	Business Requirements:☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner • Driver supervisor	
DESCRIPTION:	This use case describes an instance where the booking consultant wants to confirm a trip and checks the whether or not a vehicle is available. The booking consultant can then select the check availability option and the system returns a list of Vehicles which are available.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant wants to check whether or not a vehicle is available	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to check whether or not a driver is available and selects the Check availability option	**Step 2:** The system retrieves the **Trip Date** **TripTime** **Number_of_Passengers** From the **Booking** table
		Step 3: The system reads from the **Slot** table the: **Slot_Date**

		Slot_Time and the **Number_of_Passengers** from the **Vehicle Group** table
		Step 4: The system compares the **Slot_Date** and **Slot_Time** To the **Trip Date** and **Trip Time** and returns a list of the **Vehicle_ID**'s and **Vehicle_Model** **Number_of_Passengers** whose **Slot_Date** and **Slot_Time** did not match the **Trip Date** and **Trip Time**
		Step 5: The system displays the list of vehicles that have been returned
ALTERNATE COURSES:	Alt step 4: There are no vehicle available on the given date and time and a message is displayed that there are vehicles available and the use case ends	
CONCLUSION:	The system displays the vehicles that are available	
POST-CONDITION:	The schedule should be up to date at all times	
BUSINESS RULES	• None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Assign vehicle to trip	**USE CASE TYPE**
USE CASE ID:	5.5	Business Requirements:☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Team leader • Driver supervisor	
DESCRIPTION:	This use case describes the event when the booking consultant wants to assign a specific driver to a vehicle. They check the availability of a driver and use case 4.2 Check Driver Availability is invoked. The booking consultant then chooses an available vehicle that is then added on the slot.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant wants to assign a specific driver to a trip	

TYPICAL COURSE OF EVENTS:	Actor Action	System Response
	Step 1: The booking consultant wants to assign a specific vehicle to a trip and selects assign vehicle option	**Step 2:** Invoke use case 4.2
	Step 3: The booking consultant selects **Vehicle_Model** to be booked	**Step 4:** The system writes the **Vehicle_ID and Vehicle_Model** to the **Slot.**
		Step 5: The system displays a message confirming the assignment.
ALTERNATE COURSES:	None	
CONCLUSION:	The slot is updated	
POST-CONDITION:	If a driver is unavailable, they cannot be assigned to another trip	
BUSINESS RULES	• There has to be a pre-existing booking before a driver can be assigned	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	Date: 04-22-2019 Version: 2

USE CASE NAME:	Outsource Vehicle	USE CASE TYPE
USE CASE ID:	5.6	Business Requirements:☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Team leader • Driver supervisor	
DESCRIPTION:	This use case describes the event when the booking consultant wants to assign a specific driver to a vehicle. The booking consultant prompts the system to display the vehicle then chooses an available vehicle that is then added on the slot.	
PRE-CONDITION:	The booking consultant has to be logged onto the system	
TRIGGER:	The booking consultant wants to assign a specific driver to a trip	
TYPICAL COURSE OF EVENTS:	Actor Action	System Response
	Step 1: The booking consultant wants to assign a specific vehicle to a trip and selects assign vehicle option	**Step 2:** The system displays information of: **OutSoucreVehicle_ID** **OutSource_Vehicle_Model**

		OutSource_Make from the **OutSource_Vehicle** table
	Step 3: The booking consultant selects the vehicle model desired to be booked	**Step 4:** The system writes the **Vehicle_ID and Vehicle_Model** to the **Slot.**
		Step 5: The system displays a message confirming the assignment
		.
ALTERNATE COURSES:	None	
CONCLUSION:	The slot is updated	
POST-CONDITION:	If a vehicle is unavailable, they cannot be assigned to another trip	
BUSINESS RULES	• There has to be a pre-existing booking before a vehicle can be assigned	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Create vehicle group	**USE CASE TYPE**
USE CASE ID:	6.1	Business Requirements:☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes an instance when the booking consultant wants to add another vehicle group onto the system database. They retrieve all the relevant vehicle group information and a new vehicle is saved onto the system database.	
PRE-CONDITION:	The vehicle group cannot be an existing vehicle group in the system	
TRIGGER:	The booking consultant wants to add another vehicle group	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1:** The booking consultant wants to add another vehicle group and selects the Add Vehicle Group option.	**Step 2:** The system displays a page where the booking consultant can add the required details of the vehicle group such as: **VehicleGroup_Name** **VehicleGroup_Description**
	Step 3: The booking consultant adds the necessary	**Step 4:** The system verifies the information gathered

	details of the new vehicle group	
		Step 5: The system retrieves the last line in **Vehicle group ID**, increments by 1 and assigns a new **Vehicle group ID**
		Step 6: The system writes the information given into the **Vehicle group** table under the new **Vehicle group ID**
		Step 7: The system sends a message that a new vehicle group is added
ALTERNATE COURSES:	Alt step 4: The booking consultant made an error with the entered information and an error message is provided	
CONCLUSION:	A new vehicle group is added onto the system database	
POST-CONDITION:	The vehicle group information has to be complete	
BUSINESS RULES	● None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Nondumiso Mahlangu	Date: 04-22-2019 Version: 2

USE CASE NAME:	Search vehicle group	**USE CASE TYPE**
USE CASE ID:	6.2	Business Requirements:□ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	● None	
OTHER INTERESTED STAKEHOLDERS:	● None	
DESCRIPTION:	This use case describes the booking consultant wanting to search a vehicle group uses the Vehicle group ID to search and retrieve the details of the vehicle from the Vehicle group table in the database.	
PRE-CONDITION:	The vehicle group has to have a unique ID	
TRIGGER:	The booking consultant enters the Vehicle group ID of the driver onto the system	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant selects the Search Vehicle Group option of the driver on the system	**Step 2**: The system displays a page for the booking system to enter the search criteria.
	Step 3: The booking consultant enters the	**Step 4:** The system verifies the character input

	characters of the vehicle group they want to search	
		Step 5: The system reads the **VehicleGroup_Name** from the table **Vehicle Group**
		Step 6: The system compares the characters received with the characters in the Vehicle table in the **Vehicle_Model** field until a match is found
		Step 7: The system retrieves the **VehicleGroup_ID** **VehicleGroup_Name,** **VehicleGroup_Description,** information related to the characters given from the **Vehicle Group** Table
		Step 8: The system displays the retrieved information.
ALTERNATE COURSES:	Alt step 3: The characters given did not match system requirements and an error message is displayed	
	Alt step 6: A match is not found and an alert message is displayed	
CONCLUSION:	The system retrieves the information related to the Vehicle group ID given	
POST-CONDITION:	The vehicle group information searched has to be displayed for the booking consultant	
BUSINESS RULES	● None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Register client	**USE CASE TYPE**
USE CASE ID:	7.1	Business Requirements:
PRIORITY:	HIGH	System Analysis: □
SOURCE:	Ndila Transfers	System Design: □
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking Consultants	
OTHER PARTICIPATING ACTORS:	● Operational Manager ●	
OTHER INTERESTED STAKEHOLDERS:	●	
DESCRIPTION:	The use case describes an event where a new client is registered on the system. The use case begins when the client wants to start using Ndila services or has interest in using the Ndila services, client details such as	

	name of agency in case of agency, physical address, email, contact number, Client reference can either be client id number (Private clients) or Company registration number(Agencies) will be required to be captured and stored under the **Client Table.** The use case concludes when the client is added to the **Client** table	
PRE-CONDITION:	The client must have not used Ndila services before	
TRIGGER:	Request to be added to the system through phone call	
TYPICAL COURSE OF EVENTS:	**Step 1:** Client will request to use the Ndila services or associate themselves with Ndila	**Step 2:** The booking consultant selects to add client
		Step 3: The system prompts the booking consultant to enter the following client details: **Name of client/agency** **Physical address of the agency** **Email address** **Phone number** **Client type** **Client_Reference**
	Step 4: The client provides the requested details	**Step 5:** The booking consultant captures the details of the new client
		Step 6: The booking consultant confirms with the client if the information is correct
	Step 7: The client confirms the information is correct	**Step 8:** The Booking consultant submits, and the system validates the information provided for in case of incorrect format
		Step 9: The system validates if the client exists in the **CLIENT** table
		Step 10: The system stores the details of the client under **Client** table. **Name ->Client_Name** **PhysAddress-> Client_Address** **EmailAddress->Client_Email** **PhoneNo-> Client_Tel** **ClientRef -> Client_Reference** Select **Client type ID** retrieved from **Client_Type** table
		Step 11: The system confirms that the client is successfully added to the **Client** table and it is communicated to the client by the Booking Consultant
ALTERNATE COURSES:	**Alt Step 7:** The client declines **Step 7a:** The booking consultant requests for the details to be rectified. ***Return to step 5***	
	Step 8: Returns an error message explaining wrong information is provided and the first place where the error is identified it will be highlighted requesting correct details. ***Return to step 3***	

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2
	ALT Step 9: The system retrieves the details of the existing client and the booking consultants communicate with the client that they have already been added to the system. **Step 9a**: The booking consultant requests the client to confirm the details **Step 9b:** The client confirms the details. **Step 9c:** The booking consultant ends the task
	Step 11: The system fails to add the client to the table and returns an error message. The use case end
CONCLUSION:	The client is added to the system
POST-CONDITION:	The client should use our services
BUSINESS RULES	•
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•
ASSUMPTIONS:	•
OPEN ISSUES:	

USE CASE NAME:	SEARCH CLIENT	**USE CASE TYPE**
USE CASE ID:	7.2	Business Requirements:☐
PRIORITY:	HIGH	System Analysis:☐
SOURCE:	Ndila Transfers	System Design: ☐
PRIMARY BUSINESS ACTOR	Booking Consultant	
PRIMARY SYSTEM ACTOR		
OTHER PARTICIPATING ACTORS:	• Manager	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	The use case describes an event where a user searches for the client in the system. The use case begins by requesting the name of the agency or client to search, Information about the client if it exists in the **Client** table, else an error message is displayed informing the booking consultant that the user is not registered on the system. The use case concludes when the details are retrieved from the **Client** table provided it exists	
PRE-CONDITION:		
TRIGGER:	Retrieve a client's details	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Selects the option to search for a client	**Step 2:** The system displays the following information about all the clients in the **Client** table: **Client_Name** **Client_Reference** **Client_PhoneNumber** **Client_Email**
		Step 3: The system prompts the booking consultant to enter: **Client_Name** or **Client_Reference**
	Step 4: The booking consultant enters the client name/agency or Client reference and selects search	**Step 5:** The system validates for correct input formats
		Step 6: System narrows the list of clients based on the client/agency name
	Step 7: The booking consultant selects the specific client	**Step 8:** The system displays the following details of the client: **Client_Name** **Client_Reference** **Client_PhoneNumber** **Client_Email** **Client_Address** **Client_type** Retrieved from the **Client** table.
ALTERNATE COURSES:	**ALT Step 5:** An error message is display by the system informing the booking consultant with input error. The booking consultant can either go back to **Step 3** or terminate the search	

	ALT Step 6: An error message informing the booking consultant that the specific client does not exist in the **Client** table. The booking consultant can either go back to **Step 3** or terminate the search.
CONCLUSION:	Client details are retrieved from the **client s**table
POST-CONDITION:	
BUSINESS RULES	•
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•
ASSUMPTIONS:	•
OPEN ISSUES:	

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update Client	**USE CASE TYPE**
USE CASE ID:	7.3	Business Requirements:X
PRIORITY:	High	System Analysis: □
SOURCE:	Ndila Transfer	System Design: □
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes an event where a user details are to be updated by the booking consultant. The use case begins by the client requesting to update their details such as location, contact details and so forth. The client provides their name and the client is searched on the system. Client provides the specific details that needs to be updated. The booking consultant will then capture the details and update them on the system	
PRE-CONDITION:	The client must exist in the system	
TRIGGER:	The client calls in to update their details	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The client calls in to request to update their details	**Step 2:** The booking consultant will select client Section
		Step 3: The system invokes Use case 7.3 search client
		Step 4: The booking consultant selects update client.
		Step 6: The system allows the following fields to be modified:

		Client_Name Client_Reference Client_PhoneNumber Client_Email Client_Address Client_Fax Client_type
		Step 7: The booking consultant requests for the details of the specific fields that the client would like to update
	Step 8: The client provides the specific details that needs to be updated	**Step 9:** The booking consultant captures the information of the specific details that needs to be updated and selects update
		Step 10: The system validates the information in the input fields
		Step 11: The system saves the updated details in the **Client** table
		Step 12: The system informs the booking consultant of successfully updating the information in the **Client** table
ALTERNATE COURSES:	**Alt Step 10:** The information provided is invalid the booking consultants' requests for correct information and return to step 7	
	ALT Step 12: The system notifies the booking consultant that it failed to update the client details returns to step 7 or terminates	
CONCLUSION:	The client's information is updated in the **Client** table.	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2

USE CASE NAME:	Add Client Type	**USE CASE TYPE**
USE CASE ID:	8.1	Business Requirements:X
PRIORITY:	High	System Analysis: ☐
SOURCE:	Ndila Transfers	System Design: ☐
PRIMARY BUSINESS ACTOR	Operations Manager	
PRIMARY SYSTEM ACTOR		

OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	The use case describes an event where client type is added to the system the client type can either be private or public, public clients are agencies, governments and companies. Private clients are individual people requesting for a personal transportation. The name, description of the client type will be added to the system. The system concludes when the type is added to the database.	
PRE-CONDITION:	Operation Manager should be logged in	
TRIGGER:	Request to add new Client Type	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The operations manager logs on the system and request to add new client type	**Step 2:** The system prompts the Operational Manager to capture following client type details: **Client_Type_Name** **Client_Type_Description**
	Step 3: The operations manager enters the required details of the client type, selects submit	**Step 4:** The system checks if the information provided is in valid format
		Step 5: The system validates if the client type exists in the **Client_Type** table
		Step 6: The system stores the required details in the **Client_Type** table
		Step 7: The system confirms the successful adding of the client type to the **Client_Type** table
ALTERNATE COURSES:	**ALT Step 4:** The system displays an error message informing the operations manager that the information provided is invalid and request the operations to either enter information or terminate.	
	ALT Step 5: The system displays an error message Informing the operations manager that the client type exists.	
	ALT Step 7: The system informs the operational manager that it failed to store the new client_type ,the manager can return to step 3 or terminate	
CONCLUSION:	The client type is added to the database	
POST-CONDITION:		
BUSINESS RULES		
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS		
ASSUMPTIONS:		
OPEN ISSUES:		
	•	

USE CASE NAME:	Search Client Type	USE CASE TYPE Business Requirements: ☐ System Analysis: X System Design: ☐
USE CASE ID:	8.2	
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking Consultants	
PRIMARY SYSTEM ACTOR		
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes an event of searching a client type in the database. The name of the client type is typed and it is search if it exists. The details of the client type are retrieved if it exists	
PRE-CONDITION:		
TRIGGER:	Checking if the client type exists	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The Booking consultant wants to search and selects the Client section on the system	**Step 2:** The system displays operations that the user can perform such as update, search, add new client type
	Step 3: The Booking consultant selects the search operation	**Step 4:** The system prompts the name of the client type to be searched And displays all the client types in the **Client_Type table** **Client_Type_Name** **Client_Type_Description**
	Step 5: The user enters the name of the client type in the search field	**Step 6:** The system validates the name provided in the input box if it's in correct format such as no numbers or the search field is not empty
		Step 7: The system displays the following details retrieved from **Client_Type** table of a client types matching the name entered in the search field: **Client_Type_Name** **Client_Type_Description**
	Step 8: The Booking consultant selects the specific client type	
		Step 9: The details of the client type retrieved from the **Client_Type** table are displayed on the screen:

		Client_Type_Name-> Client_Type_Name Client_Type_Description-> Client_Type_Description
ALTERNATE COURSES:	**ALT Step 6:** The system displays an error message informing the user incorrect input and request for correct input or the user can terminate the search	
	ALT Step 9: The specific client type does not exist in the **ClientType** table	
CONCLUSION:	The details of the client type are retrieved from the **ClientType** table.	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update Client Type	USE CASE TYPE
USE CASE ID:	8.3	Business Requirements:X
PRIORITY:	High	System Analysis: ☐
SOURCE:	Ndila Transfer	System Design:☐
PRIMARY BUSINESS ACTOR	Operations Manager	
PRIMARY SYSTEM ACTOR		
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	The use case describes an event where a client type is to be updated on the system. The use case begins by searching the specific client type and retrieving all the updatable field such as description or the name of the client type. The use case concludes when the Client type is updated in the **ClientType** table.	
PRE-CONDITION:	Operations manager needs to be logged in	
TRIGGER:	To update the specific client type	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The Operations manager requests to update a specific client type by	**Step 2:** The system displays the available operations to be performed under client type

	selecting the Client section on the screen	
	Step 3: The operations manager selects the update client type	**Step 4:** The system requests the user to enter the name of the specific type, the user wants to update.
	Step 5: The operations manager enters the name of the specific client type	**Step 6:** The system invokes use case 8.2(Search Client type).
	Step 7: The operations manager selects the update options	**Step 7:** The system allows the following fields to be updated : **Client_Type_Description** **Client_Type_Name**
	Step 8: The operations manager enters new information on the fields they want to update and clicks the button update	**Step 9:** The system validates the information in the updated fields
		Step 10: The system saves the updated information in the **ClientType** table. And alerts the operations manager that it is successfully added.
ALTERNATE COURSES:	**ALT Step 9:** The system informs the operations manager that the information provided is not in a correct format. The operations manager can either terminate or Go to **Step 5.**	
	ALT Step 10: The system informs the operations manager that it failed to update the client type details and can either terminate or return to step 5	
CONCLUSION:	The client type is updated in the **ClientType** table	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2

USE CASE NAME:	CREATE BOOKING	**USE CASE TYPE**
USE CASE ID:	9.1	Business Requirements: X
PRIORITY:	High	System Analysis: □
SOURCE:	Ndila Transfers	System Design: □
PRIMARY BUSINESS ACTOR	Client (Agency or Private Client)	

PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	- Driver - Driver Supervisor	
OTHER INTERESTED STAKEHOLDERS:	- Operations Manager	
DESCRIPTION:	This use case describes an event where a client request to make a booking for a trip. The use case begins when the client calls in or sends an email to request for services. The name of the client/agency will be required to check if they have used the services before else they are added to the system. The booking information such as date, time, location (Departure and destination), Vehicle group/number of people to be transported and Passenger details. A quote is generated and communicated to the client. The use case concludes when the Booking is saved in the booking table.	
PRE-CONDITION:	Must be registered client	
TRIGGER:	Calls or sends an email to request to book a trip	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Client calls/emails in to request a booking	**Step 2**: The Booking consultant selects to create a Booking
		Step 3: The system prompts the booking consultant to enter the following details **ClientName**
	Step 4: The client provides their agency name	**Step 5**: The booking consultant captures the name of the agency in the input field and select search
		Step 6: The system validates the information entered in the input field
		Step 7: invoke use case search client and retrieves the client details
		Step 8: The booking consultant confirms the details of the client retrieved from the Client table
	Step 9: The client confirms the details	**Step 10:** The System saves the client information retrieved from the Client table in the input fields
		Step 11: The system prompts the booking consultant to enter the following details: ***DateOfPickUp*** ***DateOfArrival*** ***TimeOfPickUp*** ***TimeOfArrival*** ***PickupLocation*** ***DropOffLocation*** ***NumberOfPassengers***
	Step 12: The client provides the requested details	**Step 13:** The booking consultant captures the details provided by the client
		Step 14: The system validates the input if it's in correct format

		Step 15: The system validates if the booking exists in the **Booking** table
		Step 16: invokes use case *Generate quote and retrieves the generated quote*
		Step 17: The system prompts confirmation of the estimated price of the trip
	Step 18: The client agrees to the trip price	**Step 19**: The system prompts the Booking consultant to enter the following details of Passenger **Passenger Name** **Passenger Surname** **Passenger PhoneNo** **Booking Order**
	Step 20: The client provides the details of the person to be collected	**Step 21:** The booking consultant captures the details provided to their respective fields
		Step 22: The system validates the information in the input fields
		Step 23: The system validates if the passenger exists in the Passenger table
		Step 24: System saves the Passenger details in Passenger Table **Name -> Passenger_Name** **Surname -> Passenger_Surname** **PhoneNo-> Passenger_PhoneNo** **BookingOrder-> Booking_Order**
		Step 25: The system prompts the booking consultant to enter pickup Instructions
	Step 26: The client provides instructions about the collection	**Step 27:** The booking consultant captures the instructions
		Step 28: The system prompts for Booking confirmation
	Step 29: The client confirms the booking details	**Step 30:** The system Auto Generate Booking_Reference, Saves the Booking details in **Booking_Trip table** ***DateOfPickup*->Date_of_PickUp** ***DateOfArrival*->Date_of_Arrival** ***PickUpTime*->Time_Of_PickUp** ***ArrivalTime*->Time_Of_Arrival** ***"Awaiting Voucher"* -> Booking_Status** Retrieve ***Pickup_Location_ID*** from ***Location*** table ***PLocationID*->Pickup_Location_ID** ***DLocationID*->DropOff_Location_ID** ***PassNumber*->Number_Of_Passengers**

		saves the following quote details in the Invoice table **InvoiceNo->Invoice_Number** **InvoiceDate->Invoice_Date** **“Outstanding”-> Payment_Status** **BookingRef->Booking_Reference** **Booking_Cost->Booking_Cost**
		Step 31: The System displays the booking reference.
		Step 32: The booking consultant notifies the client that the trip has been captured and it will be approved upon receiving a voucher card
ALTERNATE COURSES:		
	ALT step 6: Incorrect input returns to step 4	
	ALT Step 7: Error message informing that the client doesn’t exist and invokes use 7.1 Add Client	
	ALT Step 9: Declines and request to update their details by invoking use case 7.3 Update client	
	ALT Step 14: The input is incorrect and request valid input returns to step 12	
	ALT Step 15: The booking already exists, and it is communicated to the client terminates.	
	ALT Step 18: The client declines and the price is negotiated, The price gets overwritten	
	ALT Step 22: The input is invalid returns to step 31	
	ALT Step 23: The Passenger already exists in the Passenger table → Retrieve the Passenger_ID from **Passenger table** , return step 25	
	ALT Step 24: The system failed to store Passenger , terminates	
	ALT Step 29: The client declines the booking details and provides the specifics that needs update	
	ALT Step 31: The system failed to save the booking and terminates	
CONCLUSION:	The trip is successfully stored in the booking table	
POST-CONDITION:	A voucher card must be submitted for the trip to be executed	
BUSINESS RULES	• Booking reference will be assigned to every booking	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• Other agencies submit their voucher card late, System must be overwritten to allow such cases	
ASSUMPTIONS:	•	
OPEN ISSUES:		

USE CASE NAME:	Search Booking	USE CASE TYPE Business Requirements: X System Analysis: □ System Design: □
USE CASE ID:	9.2	
PRIORITY:	High	
SOURCE:	Ndila Trasnfers	
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	• Manager	
OTHER INTERESTED STAKEHOLDERS:	• Drivers	
DESCRIPTION:	This use case describes an event where a booking is searched by a booking consultant in a case where they want to update it or retrieve its details. The search can be performed by either using name of client or booking reference. Booking details are retrieved from the following tables; **Booking** table, **Invoice**, **Vehicle_Group**, **Driver**, **Passenger** and are displayed on the screen when the booking consultant types in characters like the booking reference or client name.The use case concludes when the results are found	
PRE-CONDITION:	The booking reference should be valid	
TRIGGER:	Search for the booking	
TYPICAL COURSE OF EVENTS:	Actor Action	System Response
	Step 1: The booking consultant selects search Booking.	**Step 2:** The system displays a list of bookings containing the following details retrieved from the **Booking** table: **Booking_Reference** **Client_Name** **Date_of_PickUp** **Date_of_Arrival** **Time_Of_PickUp** **Time_Of_Arrival** **Pickup_Location** **DropOff_Location**
		Step 3: The system prompts the booking consultant to enter the **Booking_Reference/agency name**
	Step 4: The Booking consultant enters the booking reference /agency name	**Step 5:** The system validates the input for valid format

		Step 6: Bookings that have similar characters as the one typed in the search field are retrieved from the **Booking** table displayed
	Step 7: The booking consultant selects the specific client	**Step 8:** Information about the booking such as **Booking_Reference** **Client_Name** **Date_of_PickUp** **Date_of_Arrival** **Time_Of_PickUp** **Time_Of_Arrival** **Pickup_Location** **DropOff_Location** **Booking_Status** **Vehicle_Assigned** retrieved from the **Vehicle_Group table** **Driver_Assigned** retrieved from the **Driver table** **Number_Of_Passengers** **Passenger_Name** **Passenger_Surname** **Passenger_PhoneNo** **Booking_Order** Retrieved from **Booking** table is displayed on the screen.
ALTERNATE COURSES:	**ALT Step 5:** The system prompts Invalid input return to step 3	
	ALT Step 6: An empty list is displayed	
	ALT step 7: The Booking doesn't exist can either return to step 3 or terminate	
CONCLUSION:	Details of the specific booking are retrieved	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

USE CASE NAME:	Update Booking	**USE CASE TYPE** Business Requirements: X System Analysis: □ System Design: □
USE CASE ID:	9.3	
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes an event where a user wants to update a booking made by a client. The booking consultant will request the booking reference or the name of the client/agency who made the booking. The booking will invoke the search booking and details of the booking are retrieved enabling the booking consultant to modify the details. The use case concludes when the booking is updated	
PRE-CONDITION:	A valid booking reference should be provided	
TRIGGER:	Request to update booking details	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The client will request to update the details of the Booking	**Step 2:** The booking consultant selects the Booking Section
		Step 3: The system prompts the booking consultant to enter the **Booking_Reference**
	Step 4: The client provides the booking reference	**Step 5**: The booking consultant enters the booking reference and invoke use case search booking
		Step 6: The booking consultant selects update booking

		Step 7: The system enables the booking consultant to update the following fields Date_of_PickUp Date_of_Arrival Time_Of_PickUp Time_Of_Arrival Pickup_Location DropOff_Location Number_Of_Passengers Passenger_Name Passenger_Surname Passenger_PhoneNo Booking_Order Booking_status Client_Name
		Step 8: The booking consultants asks the client which details they will like to update.
	Step 9: The client provides the details that they want to update.	**Step 10**: The booking consultant enters the new information and submit
		Step 11: The system verifies if the information provided is in a correct format
		Step 12: The system stores the updated information under Booking table
		Step 13: The System confirms that the update is complete
		Step 14: The booking consultant confirms with the client that the update is complete.
ALTERNATE COURSES:	**ALT Step 11:** The system displays an error message informing the booking consultant that incorrect input return to step 10	
	ALT Step 13: The system informs the booking consultant that the system failed to update the booking details	
CONCLUSION:	This use case concludes when the booking is updated.	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

USE CASE NAME:	Cancel Booking	USE CASE TYPE
USE CASE ID:	9.4	Business Requirements:X
PRIORITY:	High	System Analysis: □
SOURCE:	Ndila Transfers	System Design: □
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes an event where a client will like to cancel a booking. The booking consultant will request the name of the client and a reference provided to the client for the booking. The booking consultant will check if the booking really exists in the **Booking** table and check if the driver has been dispatched already. If driver has been dispatched the client will be requested to pay the amount charged, Else the booking is cancelled, and the driver is informed about the cancelation of the trip. The system concludes when the booking has been cancelled	
PRE-CONDITION:	A valid booking reference should be provided	
TRIGGER:	The client calls in and requests to cancel a trip	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The client calls in to request a cancelation of a booking	**Step 2:** The booking consultant selects the Booking section and select Cancel Booking
		Step 3: The system prompts the booking consultant to enter the Booking reference
	Step 4: The client provides the booking reference	**Step 5:** The booking consultant enters the booking reference and invokes use case Search Booking
		Step 6: The booking consultant checks if the trip is due by comparing departure time with the current time the client is requesting for a cancelation
		Step 7: The booking consultant selects proceed
		Step 8: The system prompts the booking consultant to enter reasons for cancelation
	Step 9: The client provides the specific reasons of trip cancellation	**Step 10:** The booking consultant captures the cancelation reason and submits
		Step 11: The system updates the availability of a vehicle for that

		specific date and time by removing it in **slot table**
		Step 12: The system updates the availability of driver and the driver is informed about the cancellation of the trip**.**
		Step 13: The system updates the schedule to indicate that the trip is canceled
		Step 14: The system updates the **Payment_Status** under **Invoice** table that the trip is cancelled.
		Step 15: The system updates the **Booking_Status** to cancelled
		Step 16: The booking consultant communicates with the client the completion of the booking cancellation
ALTERNATE COURSES:		
	Alt Step 7: The booking consultant informs the client that the trip is due already and they must pay 50% of the total cost of the trip **then proceed**	
	Alt Step 14: The booking consultant will update the payment status under Invoice table to payment outstanding.	
	ALT Step 16: Failed to cancel the booking then terminates	
CONCLUSION:	The use case concludes when the trip is cancelled	
POST-CONDITION:		
BUSINESS RULES	• Client must pay up 50% of the amount if the trip is already due	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Matabane Mathopatona	Date: 04-22-2019 Version: 2

USE CASE NAME:	Confirm Booking	USE CASE TYPE
USE CASE ID:	9.5	Business Requirements:X
PRIORITY:	High	System Analysis: □
SOURCE:	Ndila	System Design: □
PRIMARY BUSINESS ACTOR	Booking Consultant	
PRIMARY SYSTEM ACTOR		
OTHER PARTICIPATING ACTORS:	• Agency	
OTHER INTERESTED STAKEHOLDERS:	• Operational /financial Manager	
DESCRIPTION:	This use case describes an event where a booking is to be confirmed after receiving a voucher card from the agency. The agency sends an email with a voucher card which is used to confirm that the agency is aware of the trip and approves of the costs of the trip. The booking consultants assigns vehicle and driver upon receiving the email from the agency. This use case concludes when the trip is booked, and its status is updated.	
PRE-CONDITION:	The booking should exist and a voucher card	
TRIGGER:	Voucher card received from the agency	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant receives an email from the agency containing the voucher card for a trip	**Step 2:** The system requests the booking reference from the booking consultant
	Step 3: The booking consultant inputs the booking reference	**Step 4**: The system invokes use case search Booking and retrieves the details of the booking and allows modification
	Step 5: The booking consultant checks for available vehicle by invoking use case *check vehicle availability*.	**Step 6:** The available vehicles are displayed based on the criteria given
	Step 7: The booking consultant selects the specific vehicle suitable for the specific trip	**Step 8:** The system invokes use case *assign vehicle*
	Step 9: The booking consultant checks for available driver and invoke use case *check driver availability.*	**Step 10:** The available drivers are displayed based on the criteria given.
	Step 11: The booking consultant selects the specific driver	**Step 12:** The system invokes use case *assign driver*

	Step 13: The booking consultant invokes use case *Create schedule*	**Step 14**: The system updates the booking's status to complete in the **booking** table
	Step 15: The booking consultant calls the agency to confirm that the voucher is received the booking is approved.	
ALTERNATE COURSES:	**ALT step 4:** The booking is not found and use case *create booking* is invoked.	
	ALT Step 7: Outsource a vehicle and a driver and skip steps to step 13	
	ALT step 11: Outsource a driver.	
CONCLUSION:	This use concludes when the trip's is confirmed and updated in the schedule	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

USE CASE NAME:	Add to Schedule	USE CASE TYPE Business Requirements: □ System Analysis: √ System Design: □
USE CASE ID:	10.1	
PRIORITY:	High	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Booking Consultant	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where a booking is added to the schedule. The Consultant will add a booking that is being made to the schedule by providing all the necessary details, allowing the system to validate the input information and save it to a database.	
PRE-CONDITION:	Consultant has captured all booking details	
TRIGGER:	Use Case Confirm Booking	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: A Consultant wants to add a booking to the Main schedule	**Step 2**: The system retrieves the relevant information from the booking Use Case that has been made: • **BookingReference** • **Client_ID** • **Driver_ID** • **Date** • **Time** • **Destination Time** • **DestinationLocation** • **TripDuration (Which is calculated using Time and Destination Time)**
		Step 3: System validates the format for the retrieved details
		Step 4: System retrieves the last Schedule_ID from the Schedule Table and increments it by one.
		Step 5: System will store the booking using the retrieved information (**Date, Time, TripDuration**) on the day chosen using the **Date** details alongside the new Schedule_ID
		Step 6: System displays a confirmation message to the manager

ALTERNATE COURSES:	**ALT Step 3**: The entered information is invalid. The system displays an error message stating that the formatting is incorrect and terminates the use case and returns to the **Create Booking** Use Case.	
CONCLUSION:	A new booking has been added to the schedule and has been saved	
POST-CONDITION:	The Consultant will be able to view the entered booking on the schedule	
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Mpho Mosotho	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update Schedule	USE CASE TYPE
USE CASE ID:	10.2	Business Requirements:☐
PRIORITY:	Medium	System Analysis:√
SOURCE:	Requirement List	System Design: ☐
PRIMARY BUSINESS ACTOR	Client	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where a Manager wants to update the details of the bookings that are already slotted and saved onto the system as well as in the Schedule table in the database, this may be due to outdated or incorrect information that was entered. The use case concludes with the system updating the required details on the system.	
PRE-CONDITION:	Booking has already been stored in the schedule	
TRIGGER:	Consultant wants to update the scheduled booking	

TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Consultant wants to update the scheduled booking	**Step 2**: System displays the schedule options
	Step 3: Consultant selects the View Schedule option	**Step 4:** System invokes the View Schedule Use case
		Step 5: System displays the booking details which are retrieved from the **Schedule** table. The following details can be edited: • Date • Time • TripDuration
	Step 6: Consultant updates the relevant information based upon the given changes	**Step 7:** The system reads the updated details entered and validates them
		Step 8: The system displays a validation message indicating the correct format for the details
		Step 9: System then saves the updated information in the **schedule** table
		Step 10: System then displays the updated schedule.
ALTERNATE COURSES:	**ALT Step 8**: The entered information is invalid. The system displays an error message stating that the format is incorrect and returns to **step 6**	
CONCLUSION:	The schedule details have been entered and stored on the system	
POST-CONDITION:	The schedule details are now correctly updated and saved	
BUSINESS RULES	• n/a	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• n/a	
ASSUMPTIONS:	• n/a	
OPEN ISSUES:	n/a	

USE CASE NAME:	View Schedule	**USE CASE TYPE** Business Requirements:☐ System Analysis: √ System Design: ☐
USE CASE ID:	10.3	
PRIORITY:	High	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Booking Consultant	
PRIMARY SYSTEM ACTOR	Booking Consultant	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where the consultant selects an option that makes the system displays a schedule of all the bookings that have been made. Allowing the consultant to view bookings that have been made on a specific day, month and/or year.	
PRE-CONDITION:	Consultant is logged onto the system	
TRIGGER:	Consultant wants to view bookings that have been made	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Consultant wants to view bookings that have been made	**Step 2**: The system displays the schedule allowing the consultant to choose a date or search for a specific date.
	Step 3: Consultant clicks on the chosen date he/she wants to view	**Step 4:** System displays all bookings that have been made on the specific date chosen. The following information will be displayed from the **Schedule** table: • BookingRefernce • Client_ID • Driver_ID • Date • Time • ContactPerson_ID • DestinationLocation • TripDuration
	Step 5: The consultant chooses the booking in which he/she wants to view	**Step 6:** System displays the bookings chosen by the consultant. The following information will be displayed from the **Schedule** table: • BookingRefernce • Client_ID • Driver_ID • Date • Time • ContactPerson_ID

		- DestinationLocation - TripDuration
ALTERNATE COURSES:	**ALT Step 3:** Consultant enters the date which he/she wants to view. The following details are entered - Day - Month - Year	
CONCLUSION:	A booking has been displayed	
POST-CONDITION:		
BUSINESS RULES	- n/a	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	- n/a	
ASSUMPTIONS:	- n/a	
OPEN ISSUES:	n/a	

USE CASE NAME:	Generate Bookings Report	**USE CASE TYPE**
USE CASE ID:	11.1	Business Requirements:☐
PRIORITY:	High	System Analysis: x
SOURCE:	Requirement List	System Design:☐
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	n/a	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This Use case describes an event where a Booking Report is generated based on two criteria. One being the list of all bookings made by a certain client. Second being A list generated to list all bookings made in a certain month of a certain year	
PRE-CONDITION:	Operational Manager must be logged on to the system	
TRIGGER:	Manager requests for a report	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1:Operational manager Requests to Create a Booking	**Step 2:** System shows all the reports the manager will be able to generate: • Generate Bookings • Generate Vehicle Report
	Step 3: Manager chooses to choose the booking report generator.	**Step 4:** System shows him the two options of the kind of reports he can choose from. Whether he will search by client or search by date.
		Step 5: System requires manger to enter a **Surname** or **Date**
	Step 6: Operational manager searches client by **Surname** or **Date**	**Step 7:** System uses the information entered by client and reads the **Booking Table**
		Step 8: Once client is found system returns list of all bookings made by client searched from the **Booking Table.** The system will return the following details: **Client_ID** **Client_Surname** **Booking_Date** **Pickup-Location** **DropOff-location**

	Step 9: After report is generated Operational Manger can now choose to download the returned results into a pdf document.	
ALTERNATE COURSES:	**ALT Step 8:** System could not find a booking and displays the error message to the Operational Manager	
CONCLUSION:	A booking report is generated	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• The "**To Date**" must not be a date after the current date.	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Mpho Mosotho	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Generate Vehicle Report	**USE CASE TYPE** Business Requirements:☐ System Analysis: x System Design: ☐
USE CASE ID:	11.2	
PRIORITY:	High	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	n/a	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This Use case describes an event where a Vehicle Report is generated based on two criteria. One being the list of all bookings made by a certain client. Second being A list generated to list all bookings made in a certain month of a certain year	
PRE-CONDITION:	Operational Manager must be logged on to the system	
TRIGGER:	Manager requests for a report	

TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Operational manager Requests to Create a Booking	**Step 2:** System shows all the reports the manager will be able to generate: • Generate Bookings • Generate Vehicle Report
	Step 3: Manager chooses to choose the Vehicle report generator.	**Step 4:** System shows him the two options of the kind of reports he can choose from. Whether he will search by client or search by date.
		Step 5: System requires manger to enter a **Surname** or **Date**
	Step 6: Operational manager searches client by **Surname** or **Date**	**Step 7:** System uses the information entered by client and reads the **Booking Table**
		Step 8: Once client is found system returns list of all bookings made by client searched from the **Booking Table.** The system will return the following details: **Client_ID** **Client_Surname** **Booking_Date** **Pickup-Location** **DropOff-location**
	Step 9: After report is generated Operational Manger can now choose to download the returned results into a pdf document.	
ALTERNATE COURSES:	**ALT Step 8:** System could not find a booking and displays the error message to the Operational Manager	
CONCLUSION:	A Vehicle report is generated	
POST-CONDITION:		
BUSINESS RULES	•	

IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•
ASSUMPTIONS:	•
OPEN ISSUES:	

Siyaya Travel Assist	
Author (s): Mpho Mosotho	Date: 04-22-2019 Version: 2

USE CASE NAME:	Generate Mileage Report	USE CASE TYPE
USE CASE ID:	11.3	Business Requirements:□
PRIORITY:	Medium	System Analysis √
SOURCE:	Requirement List	System Design: □
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	Operational Manager	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where a manager requests the mileage report from the system. The system generates the report by retrieving information matching certain criteria. The report is then saved to a database. This use case concludes with the report being saved to the system.	
PRE-CONDITION:	The manager must be logged in.	
TRIGGER:	The manager wants to generate the mileage report	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The manager selects the generate the mileage report.	**Step 2**: The system displays the mileage generation report screen.
	Step 3: The manager enters the requested report criteria based on the following options: • Vehicle_ID • DistanceTravelled	**Step 4:** The system reads the entered report criteria information used for the report.
		Step 4: System retrieves the information and returns of all vehicles that match the input criteria From the **Vehicle Maintenance** Table
		Step 5: System the generates the Mileage report
		Step 6: System stores the report in the documents table
		Step 7: System displays a confirmation message to the manager

ALTERNATE COURSES:	ALT Step 5: Once the report generation is completed, no information is displayed as no data matched the report criteria. Error notification is sent to the manager stating there were no matches and requests the manager to re-enter the report criteria. • Return to step 3.	
CONCLUSION:	The report is generated	
POST-CONDITION:	The complete report is generated and stored	
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Mpho Mosotho	Date: 04-22-2019 Version: 2

USE CASE NAME:	Schedule Vehicle Maintenance	USE CASE TYPE Business Requirements:☐ System Analysis: x System Design: ☐
USE CASE ID:	12.1	
PRIORITY:	High	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	n/a	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where vehicle maintenance is added to the schedule. The manager will add to the schedule by providing all the necessary details, allowing the system to validate the input information and save it to a database. The use case concludes with the new maintenance or maintenance appointment being saved to the system.	

PRE-CONDITION:	The vehicle maintenance must not currently be in the schedule. The manager must be logged onto the system.	
TRIGGER:	Manager requests to add a vehicle on the Maintenance schedule	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1:Operational manager Requests to add a vehicle to the maintenance schedule	**Step 2:** The system requests that the manager inputs the required information for a new vehicle maintenance • Vehicle_ID • Vehicle_LicseneceNumber • DateTime • ServiceProvider_ID • DistanceTraveled
	Step 3: Manager inputs the maintenance details: • Vehicle_ID • Vehicle_LicseneceNumber • DateTime • ServiceProvider_ID • DistanceTraveled	**Step 4:** System reads the entered maintenance details
		Step 5: System validates the format for the entered details
		Step 6: System retrieves the last schedule number and increments it by one
		Step 7:The system stores the maintenance details in the **Vehicle_Maintenance** Table with its new unique number
		Step 9: The System sends the manager a confirmation message
ALTERNATE COURSES:	**ALT Step 6:** The entered information is invalid. The system displays an error message stating that the formatting is incorrect and asks the user to re-enter the information in a valid format. • Return to step 3.	
CONCLUSION:	The maintenance has been added to the maintenance schedule	
POST-CONDITION:		
BUSINESS RULES	•	

IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•
ASSUMPTIONS:	•
OPEN ISSUES:	

Siyaya Travel Assist	
Author (s): Mpho Mosotho	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Update Vehicle Maintenance Schedule	**USE CASE TYPE** Business Requirements:☐ **System Analysis: x** System Design: ☐
USE CASE ID:	12.2	
PRIORITY:	Medium	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	n/a	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where a Manager wants to update the details of maintenance that are already registered and entered onto the system as well as in the **Maintenance Schedule** table in the database, this may be due to outdated or incorrect information that was entered. The manager may also remove maintenance / maintenance appointment if necessary. The use case concludes with the system updating the required details on the system.	
PRE-CONDITION:	The manager must be logged onto the system.	
TRIGGER:	Schedule details need to be updated or removed	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: Operational manager Requests to update or remove the details on the schedule using the system	**Step 2:** The System displays the update schedule menus
	Step 3: Manager selects the search schedule option	**Step 4:** ***System invokes use case 12.3 - "View Vehicle Maintenance schedule"***
		Step 5: Step 5: The system displays all the maintenance details that is currently on the schedule, with the following details that can be edited: • Vehicle_ID • Vehicle_LicseneceNumber • DateTime • ServiceProvider_ID

		• DistanceTraveled
	Step 6: Manager enters all the relevant details that need to be updated and saves them	**Step 7:** The system reads the updated details entered by the user
		Step 8: The System validates the format of the updated information entered by the Manager
		Step 9: The System saves the edited schedule details in the **Maintenance schedule** table
		Step 10: The system displays the updated schedule details and notifies the Manager of the changes
ALTERNATE COURSES:	ALT Step 6: The user selects the remove vehicle procedure to be processed. The system prompts The system prompts the user with a confirmation message asking if the user really wants to remove the maintenance / maintenance appointment from the schedule. ALT ALT Step 6: The manager does not confirm removal. • Return to step 5. The system removes the maintenance / maintenance appointment from the schedule. • Continue to step 11	
	ALT Step 9: The details are invalid; an error message is displayed telling the user that the format is invalid. • Return to step 6.	
CONCLUSION:	The schedule details have been entered and stored on the system or removed where necessary.	
POST-CONDITION:	The schedule details are now correctly stored on the system ensuring all functions relating to the schedule are handled correctly.	
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

USE CASE NAME:	View Vehicle Maintenance Schedule	**USE CASE TYPE** Business Requirements:☐ System Analysis's : X System Design: ☐
USE CASE ID:	12.3	
PRIORITY:	High	
SOURCE:	Requirement List	
PRIMARY BUSINESS ACTOR	Operational Manager	
PRIMARY SYSTEM ACTOR	n/a	
OTHER PARTICIPATING ACTORS:	•	
OTHER INTERESTED STAKEHOLDERS:	•	
DESCRIPTION:	This use case describes the event where the Operational Manager searches for a vehicle that has been scheduled for maintenance on the system. This will allow the manager to view all details associated with the schedule. The manager enters based on a search criteria where the system will then search for the corresponding vehicle on the schedule	
PRE-CONDITION:	The manager must be logged onto the system.	
TRIGGER:	Manager requests to view Maintenance schedule	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1:Operational manager Requests to search the schedule for a specific vehicle on the schedule	**Step 2:** The system displays the schedule search menu and requests the search criteria
	Step 3: Manager searches based on the following criteria • Vehicle_ID • Vehicle_LicseneceNumber • DateTime • ServiceProvider_ID • DistanceTraveled	**Step 4:** System reads the entered maintenance details
		Step 5: System validates the format for the entered details and searches using criteria provided in the **Maintenance schedule** table
		Step 6: System displays a list of all scheduled maintenances for that specific vehicle
	Step 7: The manager then selects the specific	**Step 8:** The system the displays all the information based on the manager's selection:

	scheduled maintenance he/she wishes to see	• Vehicle_ID • Vehicle_LicseneceNumber • DateTime • ServiceProvider_ID • DistanceTraveled
ALTERNATE COURSES:	**ALT Step 6:** Once the search is completed, no information is displayed as no data matched the entered search criteria. An error notification is sent to the manager stating that they need to re-enter the search criteria. • Return to step 3	
CONCLUSION:	The vehicle maintenance has been searched	
POST-CONDITION:		
BUSINESS RULES	•	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	•	
ASSUMPTIONS:	•	
OPEN ISSUES:		

Siyaya Travel Assist	
Author (s): Mninikhaya Mavundla	Date: 04-22-2019 Version: 2

USE CASE NAME:	Add New Location	USE CASE TYPE
USE CASE ID:	13.1	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	• None	

OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	The booking consultant wants to add a new location to the system. The system prompts the Consultant to enter all relevant information. The system must allow user to add a new location with details of the location such as Country, Province, City, town/Suburb, Street Number and name. The consultant enters all requested details, the system then captures, validates and stores all information regarding the new location.	
PRE-CONDITION:	• The consultant must be logged into the system. • The added new location must not already exist on the system	
TRIGGER:	The booking consultant wants to add a new location to the system	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1**: The booking consultant wants to add a new location to the system	• **Step 2**: The system requests details of the location such as Location name [**Location_name**] • Province name [**Province_name**] • City [**City_Name**] • Suburb [**Suburb_name**] • Street [**Street_name**] • LocationType [**Locationtype]**
	Step 3: The booking consultant enters requested details of the all new relevant details requested in **Step 2.**	
	Step 4: The Consultant selects the option save details.	**Step 5:** The system prompts the Consultant to confirm details.
	Step 6: The Consultant selects to confirm details.	**Step 7:** The system captures and validates the input data. Validation includes: • Location Name – cannot be null • Province Name – cannot be null • City Name – cannot be null • Suburb Name – cannot be null • street Name – cannot be null
		Step 8: The system does not detect duplicates of Location being added.

		Step 9: The system generates a unique Land ID **[LocationID]** by adding 1 to the last **[LocationID]** found in the **Location** table.
		Step 10: The system creates and stores the validated information for the Location in the **Location** table using the captured details in **Step 7** and generated Unique ID in **Step 9.**
		Step 11: The system creates and stores a new audit entry with the following details: ● Audit ID (Generated) **[AuditID]** ● Audit Type ID **[AuditTypeID]** (Retrieved from the **AuditType** Table) ● Audit Reference Table **[AuditRefTable]** (Table where transaction was performed) ● Employee ID **[EmployeeID]** (Person initiating the transaction) – Retrieved from the **Employee** Table
		Step 12: The system displays a confirmation message stating that the new location has successfully been added to the system.
ALTERNATE COURSES:	**[ALT] Step 6:** The consultant revokes the decision to confirm details and selects the option to cancel the confirmation. ☐ Return to **Step 2.**	
	[ALT] Step 7: The system fails to pass validation checks for input data. The system displays an error message stating which fields need attention. ☐ Return to **Step 2.**	
	[ALT] Step 8: The system detects a duplicate Land already on the system with the same **[locationName]** ☐ The system displays an error message. ☐ Return to **Step 2.**	
	[ALT] Step 9: The system fails to find a **[LocationID]** in the **Location** Table. System initializes the **[LocationID]** with 1. ☐ Continue to **Step 10.**	
CONCLUSION:	The system captures the details of the new location and saves it in a database	
POST-CONDITION:	● The new location should be added to the systems database for later use. ● The transaction details are saved in the **Audit** Table.	
BUSINESS RULES	● None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	● None	
ASSUMPTIONS:	● None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Mavundla Mninikhaya	Date: 04-22-2019 Version: 2

USE CASE NAME:	Update location	USE CASE TYPE
USE CASE ID:	13.2	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	The booking consultant wants to update a location. They select the Update location option and selects location they want to update. The system updates the location information.	
PRE-CONDITION:	Location must exsit in the systems database	
TRIGGER:	The booking consultant wants to update a location	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant wants to update a location and selects the location in given options	**Step 2**: The system retrieves the information of the location and displays it for the booking consultant
	Step 3: The booking consultant enters the information they want to change on the existing location	**Step 4**: The system verifies the new information entered
		Step 5: The system saves the changes made to the location table and the status of the location is updated.
ALTERNATE COURSES:	None	
CONCLUSION:	The system updates the information of the selected location	
POST-CONDITION:	The location information has to be correct	
BUSINESS RULES	• None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Mavundla Mninikhaya	**Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Search Location	USE CASE TYPE
USE CASE ID:	13.2	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	The booking consultant uses the location name to search and retrieve the details of the location from the location table in the database.	
PRE-CONDITION:	The location has to be in the systems database	
TRIGGER:	The booking consultant enters the location name to search and retrieve the details on the system	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant enters the location name to search on the system	**Step 2**: The system verifies the location name received.
		Step 3: The system compares the location name received with the location name hat are in the location table until a match is found
		Step 4: The system retrieves the location details such as Country, Province, City, town/Suburb, Street Number and name given and displays it.
ALTERNATE COURSES:	Alt step 2: The Location name given does not meet the system requirements. The booking consultant has to type the correct Location name.	
	Alt step 3: The system cannot find a match for Location name given and sends an error message. “no results found”	
CONCLUSION:	The system retrieves the information related to the location name given	
POST-CONDITION:	The location information searched has to be displayed for the booking consultant	
BUSINESS RULES	• None	

IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	- None
ASSUMPTIONS:	- None
OPEN ISSUES:	None

Siyaya Travel Assist	
Author (s): Mavundla Mninikhaya	Date: 04-22-2019 Version: 2

USE CASE NAME:	Add Zone	USE CASE TYPE
USE CASE ID:	14.1	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	- None	
OTHER INTERESTED STAKEHOLDERS:	- None	
DESCRIPTION:	The booking consultant wants to add a zone onto the system database. They retrieve all the relevant information and a new zone and it is saved onto the system database.	
PRE-CONDITION:	The Zone rate cannot be an existing vehicle group in the system	
TRIGGER:	The booking consultant wants to add another vehicle group	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant wants to add a zone rate and selects the Add Zone option.	**Step 2**: Thesystemdisplays the zone details from the Zone table: • Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**
	Step 3: The booking consultant adds the necessary details of the new Zone rate such as name and	**Step 4**: The system verifies the information gathered

	range in kilometres **Step 2**	
	Step 4: The Consultant selects the option save details.	**Step 5:** The system prompts the Consultant to confirm details.
	Step 6: The Consultant selects to confirm details. [ALT]	**Step 7:** The system captures updated details for Zone and validates input data. Validation includes. • **Zone _name**--cannot be null, must be in string • **Zone _range**–cannot be null, must be in number**[ALT]**
		Step 8: The system does not detect duplicates of Zone being added. [ALT]
		Step 9: The system generates a unique Zone ID **[ZoneID]** by adding 1 to the last **[ZoneID]** found in the **Zone** table. [ALT]
		Step 10: The system creates and stores the validated information for the Zone in the **zone** table using the captured details in **Step 7** and generated UniqueID in **Step 9.**
		Step 11: The system creates and stores a new audit entry with the following details: • Audit ID (Generated) **[AuditID]** • Audit Type ID **[AuditTypeID]** (Retrieved from the **AuditType** Table) • Audit Reference Table **[AuditRefTable]** (Table where transaction was performed) • Employee ID **[EmployeeD]** (Person initiating the transaction) – Retrieved from the **Employee** Table
		Step 12: The system displays a confirmation message stating that the new Zone has successfully been added to the system.
ALTERNATE COURSES:	**[ALT] Step 6:** The consultant revokes the decision to confirm details and selects the option to cancel the confirmation. ☐ Return to **Step 2.**	
	[ALT] Step 7: The system fails to pass validation checks for input data. The system displays an error message stating which fields need attention. ☐ Return to **Step 2.**	
	[ALT] Step 8: The system detects a duplicate Zone already on the system with the same **[ZoneName]** ☐ The system displays an error message. ☐ Return to **Step 2.**	
	[ALT] Step 9: The system fails to find a **[ZoneID]** in the **Zone** Table. System initializes the **[ZoneID]** with 1.	

	☐ Continue to **Step10.**
CONCLUSION:	A zone rate is added onto the system database
POST-CONDITION:	• The zone rate information has to be complete • The transaction details are saved in the **Audit** Table.
BUSINESS RULES	• None
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None
ASSUMPTIONS:	• None
OPEN ISSUES:	None

Siyaya Travel Assist	
Author (s): Mavundla Mninikhaya	Date: 04-22-2019 Version: 2

USE CASE NAME:	Search Zone	USE CASE TYPE
USE CASE ID:	14.2	Business Requirements: ☐ System Analysis: ☐ System Design: ☐
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	The booking consultant uses the Zone name to search and retrieve the details of the Zone from the Zone table in the database.	
PRE-CONDITION:	The Zone has to be in the systems database	
TRIGGER:	The booking consultant enters the Zone name to search and retrieve the details on the system	
TYPICAL COURSE OF EVENTS:	**Actor Action**	**System Response**
	Step 1: The booking consultant enters the Zone name to search on the system	**Step 2:** The system verifies the Zone name received.
		Step 3: The system compares the Zone name received with the Zone name that are in the Zone table until a match is found
		Step 4: The system retrieves the Zone. details such as price and range in kilometres
ALTERNATE COURSES:	**Alt step 2:** The Zone name given does not meet the system requirements. The booking consultant has to type the correct Zone name.	
	Alt step 3: The system cannot find a match for Zone name given and sends an error message. “no results found”	

CONCLUSION:	The system retrieves the information related to the Zone name given
POST-CONDITION:	The Zone information searched has to be displayed for the booking consultant
BUSINESS RULES	• None
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None
ASSUMPTIONS:	• None
OPEN ISSUES:	None

Siyaya Travel Assist
Author (s): Mavundla Mninikhaya **Date: 04-22-2019** **Version: 2**

USE CASE NAME:	Update Zone	USE CASE TYPE
USE CASE ID:	14.3	Business Requirements: □ System Analysis: □ System Design: □
PRIORITY:	High	
SOURCE:	Ndila Transfers	
PRIMARY BUSINESS ACTOR	Booking consultant	
PRIMARY SYSTEM ACTOR	Booking consultant	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• None	
DESCRIPTION:	This use case describes the event The booking consultant wants to update a zone rate. They select the Update Zone rate option and selects the zone they want to update. The system prompts the Consultant to enter updated details for the Zone. The system then captures, validates and stores updated information The system updates the Zone information.	
PRE-CONDITION:	• Zone must exist in the systems database • The booking consultant must be logged into the system.	
TRIGGER:	The booking consultant wants to update a zone	
TYPICAL COURSE	**Actor Action**	**System Response**
	Step 1: The booking consultant wants to update a zone	**Step 2.** The system Invokes Use Case **13.3 "Search zone"**
OF EVENTS:	**Step 3**: The booking consultant enters the information they want to change on the existing zone in the search box.	**Step 4**: The system retrieves the information of the Zone and displays it for the booking consultant from the zone table to be updated: • Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**

	Step 5: The booking consultant enters the information they want to change on the existing Zone	**Step 6**: The system verifies the new information entered and confirms with the booking consultant.
	Step 7: The Consultant selects to confirm changes to be updated for the Zone. [ALT]	**Step 8**: The system captures updated details for location and validates input data. Validation includes. • **Zone _name**–cannot be null, must be in string • **Zone _range**–cannot be null, must be in number[ALT]
		Step 9: The system creates and stores a new audit entry with the following details: • Audit ID(Generated) **[AuditID]** • Audit Type ID **[AuditTypeID]** (Retrieved from the **AuditType** Table) • Audit Reference Table **[AuditRefTable]** (Table where transaction was performed) • Employee ID **[EmployeeD]** (Person initiating the transaction) – Retrieved from the **Employee** Table .
		Step 10: The system saves the changes made to the Zone table and the status of the zone is updated.
ALTERNATE COURSES:	[ALT]Step 7: The Consultant selects to cancel changes made to the location details. ➢ Return to **Step 3**. [ALT] Step 8: The system is unable to validate the details entered and displays an invalid details provided error message. Return to **Step 3.**	
CONCLUSION:	• The system updates the information of the selected Zone • The transaction details are saved in the **Audit** Table.	
POST-CONDITION:	The Zone information has to be correct	
BUSINESS RULES	• None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• None	
OPEN ISSUES:	None	

USE CASE NAME:	Remove Zone	USE CASE TYPE
USE CASE ID:	14.4	Business Requirements: □ System Analysis: □ System Design:□
PRIORITY:	High	
SOURCE:		
PRIMARY BUSINESS ACTOR	Manager	
PRIMARY SYSTEM ACTOR	None	
OTHER PARTICIPATING ACTORS:	• None	
OTHER INTERESTED STAKEHOLDERS:	• Owner	
DESCRIPTION:	This use case describes the events where a manager would like to remove a Zone from being used for any new transactions. This involves the manager selecting the zone to remove and confirming that the zone should not be accessible any longer.	
PRE-CONDITION:	The Zone should already exist in the system database The Consultant Owner must be logged into the system	
TRIGGER:	Manager/Owner wants to remove a zone	
TYPICAL COURSE	**Actor Action**	**System Response**
	Step 1: The manager Owner requests to remove a Location from the system selects the ‘remove Zone option	**Step 2:** System checks to see if the manager has authority to remove a Zone**[ALT]**
OF EVENTS:		**Step 3:** The system invokes U.C 14.3 “Search Zone”.
	Step 4: The booking consultant enters the location name to search on the system	**Step 5**: System then displays all the Zones matches in the database[ALT]
		Step 6: System prompts the manager to select the Zone they wish to remove
	Step 7: Manager selects the Zone they would like to remove	**Step 8**: System requests confirmation that the selected Zone is the correct one to be removed
	Step 9: Manager confirms that they want to remove the selected Zone**[ALT]**	**Step 10:** System then disables and removes the selected the Zone to be used again in the database.
		Step 11: The system creates and stores a new audit entry with the following details:

		• Audit ID (Generated) **[AuditID]** • Audit Type ID **[AuditTypeID**] (Retrieved from the Audit Type Table) • Audit Reference Table [**AuditRefTable]** (Table where transaction was performed) • Employee ID **[EmployeeID]** (Person initiating the transaction) – Retrieved from the Employee Table
		Step 12: System then displays a confirmation message to let the manager that the removal has been successful
ALTERNATE COURSES:	**[Alt Step 2**]: System finds the manager to not have authority to remove a Zone. Terminates use case	
	[Alt] step 5: The Zone name given does not meet the system requirements or any matches ➢ Return to **Step 4.**	
	[ALT] Step 9: The Consultant revokes the decision to confirm details and selects the option to cancel the confirmation. ➢ Return to **Step 4.**	
CONCLUSION:	The manager selected Zone has successfully been removed from the system	
POST-CONDITION:	Zone has been removed from the system.	
BUSINESS RULES	• None	
IMPLEMENTATION CONTRAINTS AND SPECIFICATIONS	• None	
ASSUMPTIONS:	• Zone already exists in the system database	
OPEN ISSUES:	None	

Siyaya Travel Assist	
Author (s): Mavundla Mninikhaya	**Date: 04-22-2019** **Version: 2**

USE CASENAME:	Generate Quote	USE CASE TYPE
USE CASE ID:	**15.1**	BusinessRequirements:☐ System Analysis:☐ SystemDesign:☐
PRIORITY:	High	
SOURCE:	Requirements Description & Detail	
PRIMARY BUSINESS ACTOR:	The booking Consultant	
PRIMARY SYSTEM ACTOR:	None	
OTHER PARTICIPATING ACTORS:	None	
OTHER INTERESTED STAKEHOLDERS:	None	

DESCRIPTION:	Thisusecasedescribestheprocessof a Consultant generating a Quote.Theusecase startswiththeConsultant selectingtogenerateaQuote,it continueswiththe Consultant entering the dates for the Quote, and the system generatingtheQuote.The usecaseconcludeswhentheQuote is generated and displayed.	
PRE-CONDITION:	• The Consultant needs to be logged in to the system.	
TRIGGER:	The Consultant selects to generate an Quote .	
TYPICAL COURSE	**Actor Action**	**System Response**
OF EVENTS:	**Step 1.** The Consultant selects to generate a Quote .	Step 2. The system prompts the Consultant to enter the start and end date for the Quotation .
	Step 3. The Consultant provides the start and end date.	Step 4. The system requests details of booking such as, • Booking_Reference [Booking_Reference]
	Step 5. The Consultant enters details and selects to continue with generating the Quote .	Step 6. The system validates that the dates and details are provided arebefore the current date. [alt]
		Step 7: The system retrieves the following informationfortheQuotefromthe Quote and Booking table: • The quotereference, [Quote_Reference_No] [Quote] • Booking_Reference [Booking_Reference] • Quote [Quote _ID] [Payment_status] [Date]
		Step 8. The system notifies the Consultant that the Quote has beengenerated.
		Step 9. The system displays the generated Quote .
ALTERNATE COURSES:	ALT 6: The entered dates are not before the current date. The system displays a notificationandreturnsto Step2.	
CONCLUSION:	The use case concludes when an Quote has been created.	
POST-CONDITION:	• No changes have been made to the system after the Quote has been generated.	

BUSINESS RULES:	•
IMPLEMENTATION CONSTRAINTS AND SPECIFICATIONS:	• None
ASSUMPTIONS:	• None
OPEN ISSUES:	• There are currently no open issues

CONCLUSION

This section serves an imperative aspect for the Functional Specification document as it gives us an outline of how the various actors interact with the system and how each requirement will be met. The Use Case narratives will be a guide throughout this document as it serves as one of the foundations, allowing stakeholders to understand any business rules or details about any use cases that may seem ambiguous.

3. PROCESS MODELLING

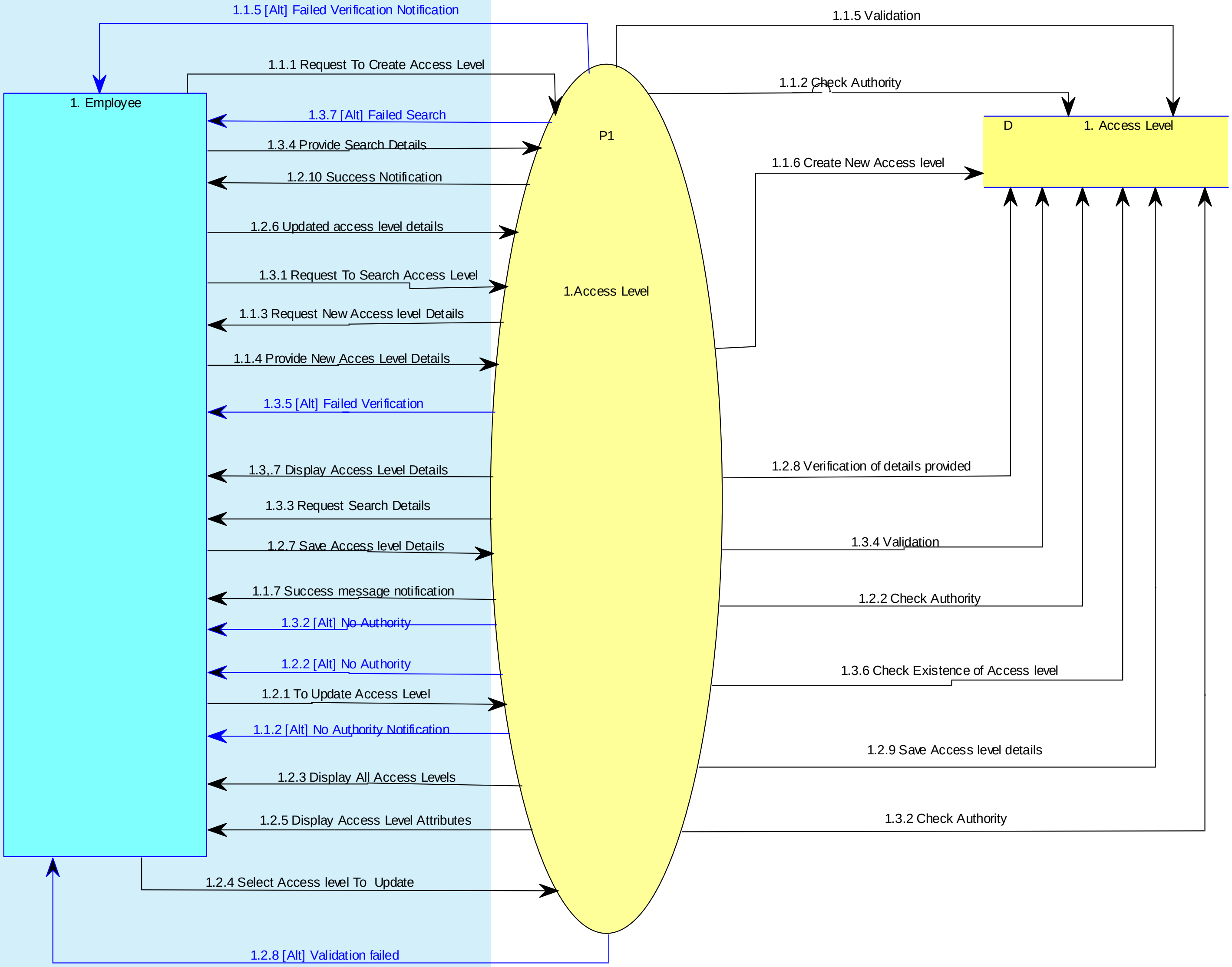
INTRODUCTION

This section is composed of a complete context diagram, functional decomposition diagram, data flow diagrams (high, middle and primitive level) as well as a complete Data Dictionary for the data flow diagrams. These diagrams are laid out to support the logical view of the system.

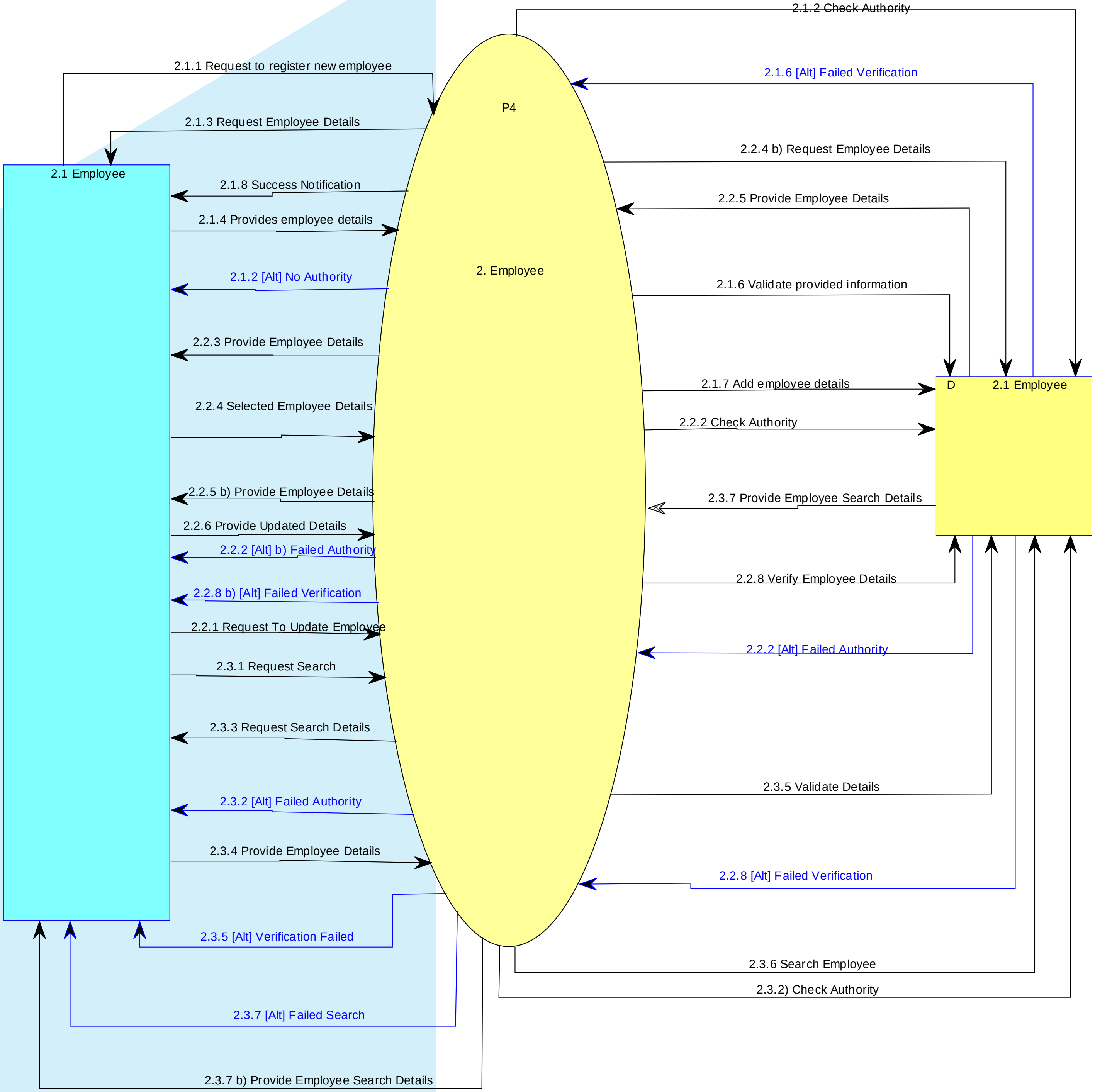
Context Appendix

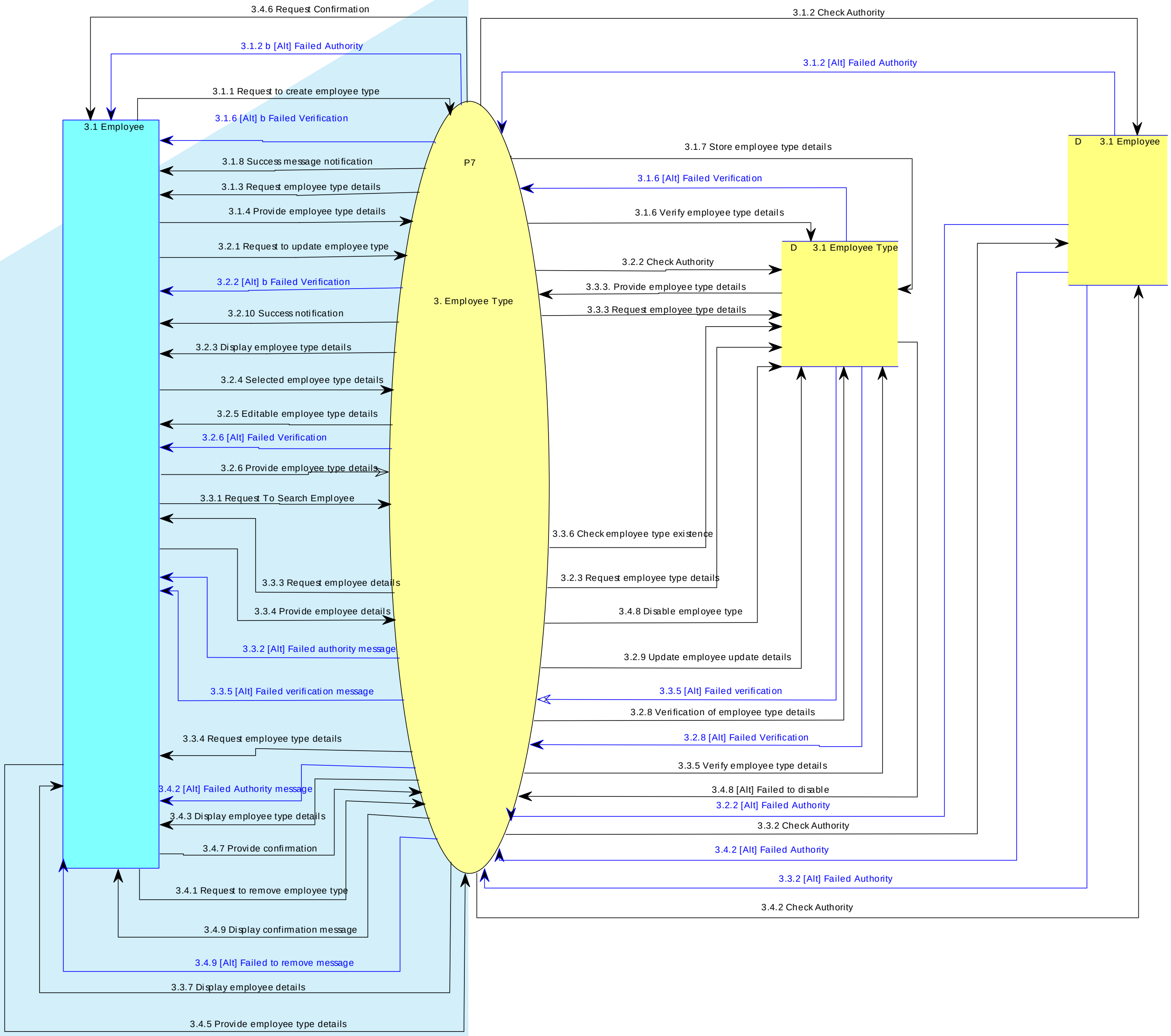
Functional Appendix

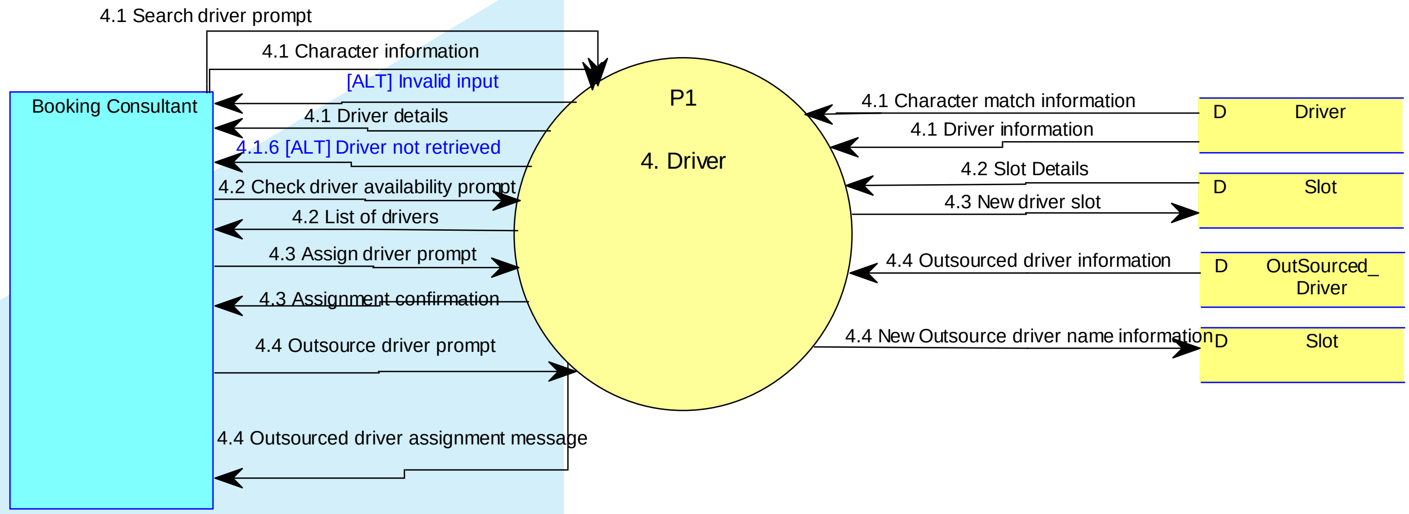
High Levels

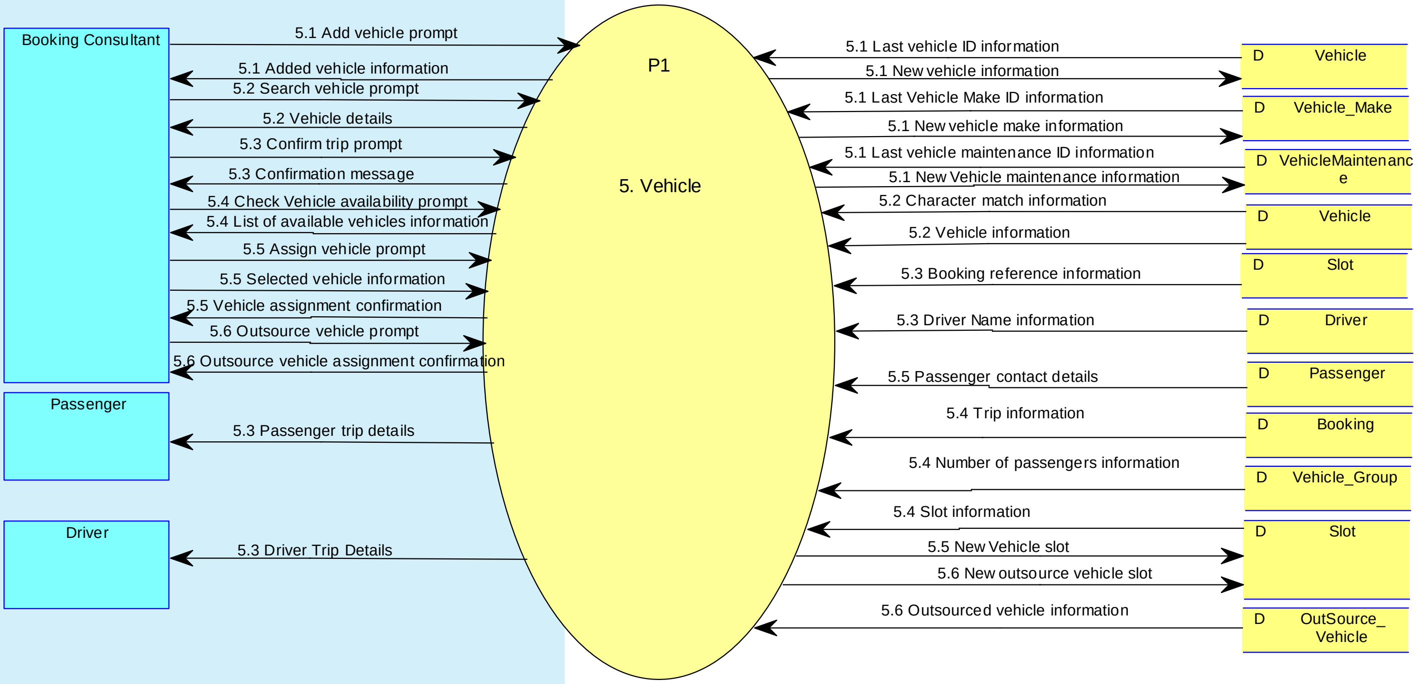


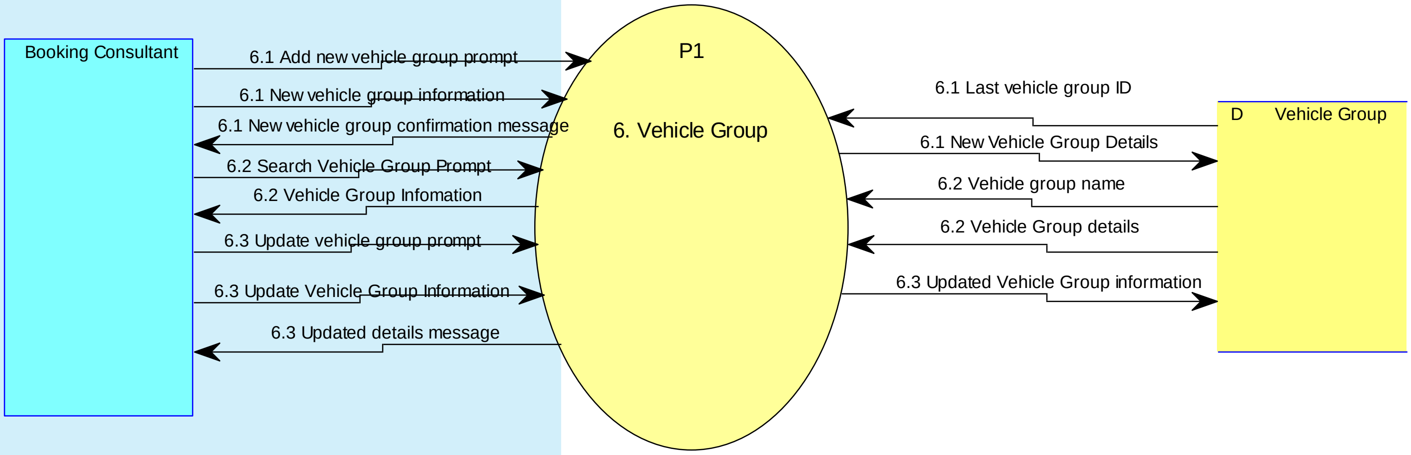
1.1.5 [Alt] Failed Verification Notification
1.1.5 Validation
1.1.1 Request To Create Access Level
1.1.2 Check Authority
1. Employee
1.3.7 [Alt] Failed Search
P1
D
1. Access Level
1.3.4 Provide Search Details
1.1.6 Create New Access level
1.2.10 Success Notification
1.2.6 Updated access level details
1.3.1 Request To Search Access Level
1.Access Level
1.1.3 Request New Access level Details
1.1.4 Provide New Acces Level Details
1.3.5 [Alt] Failed Verification
1.2.8 Verification of details provided
1.3,.7 Display Access Level Details
1.3.3 Request Search Details
1.3.4 Validation
1.2.7 Save Access level Details
1.1.7 Success message notification
1.2.2 Check Authority
1.3.2 [Alt] No Authority
1.2.2 [Alt] No Authority
1.3.6 Check Existence of Access level
1.2.1 To Update Access Level
1.1.2 [Alt] No Authority Notification
1.2.9 Save Access level details
1.2.3 Display All Access Levels
1.2.5 Display Access Level Attributes
1.3.2 Check Authority
1.2.4 Select Access level To Update
1.2.8 [Alt] Validation failed

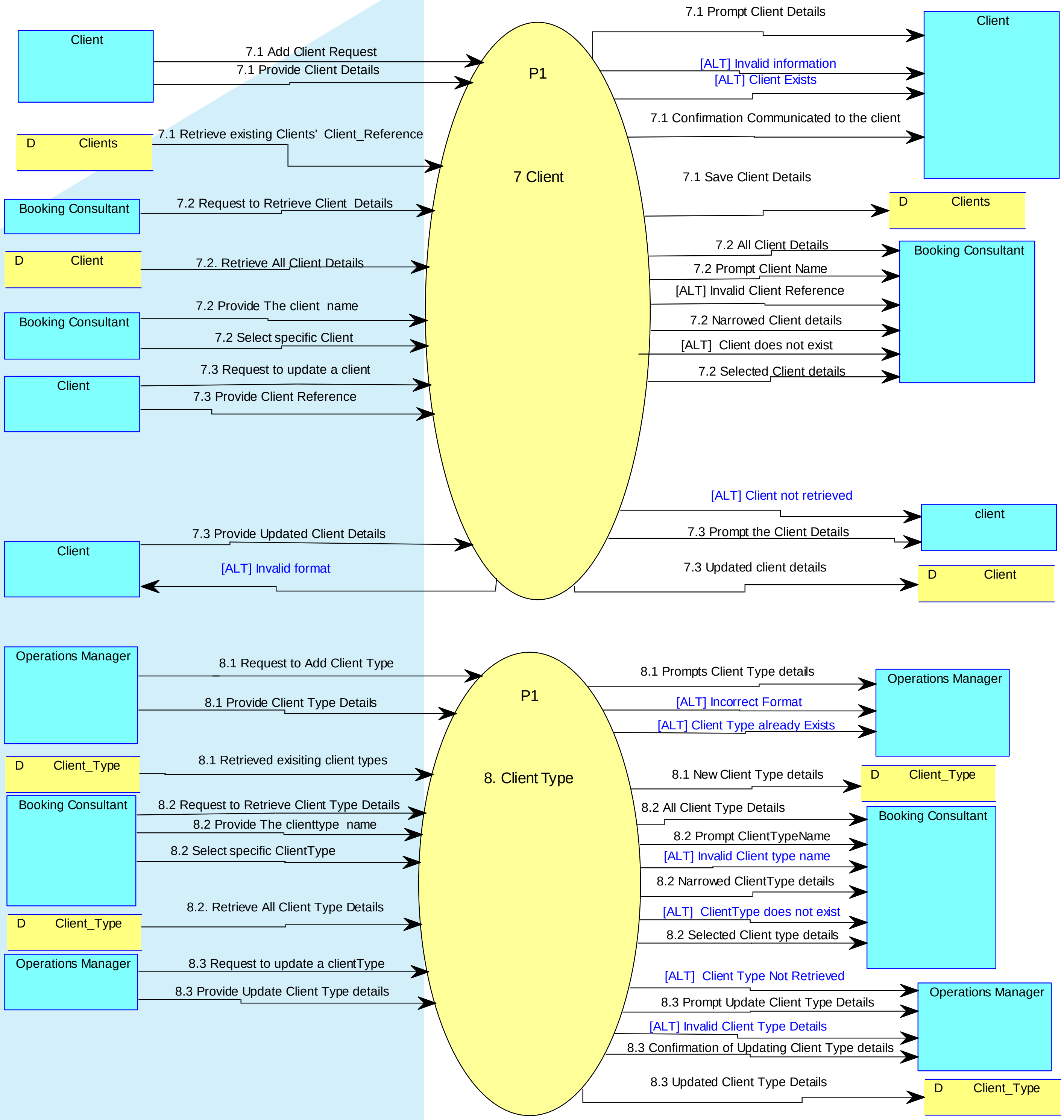

2.1.2 Check Authority
2.1.1 Request to register new employee
2.1.6 [Alt] Failed Verification
P4
2.1.3 Request Employee Details
2.2.4 b) Request Employee Details
2.1 Employee
2.1.8 Success Notification
2.2.5 Provide Employee Details
2.1.4 Provides employee details
2. Employee
2.1.2 [Alt] No Authority
2.1.6 Validate provided information
2.2.3 Provide Employee Details
D
2.1 Employee
2.1.7 Add employee details
2.2.4 Selected Employee Details
2.2.2 Check Authority
2.2.5 b) Provide Employee Details
2.3.7 Provide Employee Search Details
2.2.6 Provide Updated Details
2.2.2 [Alt] b) Failed Authority
2.2.8 Verify Employee Details
2.2.8 b) [Alt] Failed Verification
2.2.1 Request To Update Employee
2.2.2 [Alt] Failed Authority
2.3.1 Request Search
2.3.3 Request Search Details
2.3.5 Validate Details
2.3.2 [Alt] Failed Authority
2.3.4 Provide Employee Details
2.2.8 [Alt] Failed Verification
2.3.5 [Alt] Verification Failed
2.3.6 Search Employee
2.3.2) Check Authority
2.3.7 [Alt] Failed Search
2.3.7 b) Provide Employee Search Details


3.4.6 Request Confirmation
3.1.2 Check Authority
3.1.2 b [Alt] Failed Authority
3.1.2 [Alt] Failed Authority
3.1.1 Request to create employee type
3.1.6 [Alt] b Failed Verification
3.1 Employee
D 3.1 Employee
P7
3.1.7 Store employee type details
3.1.8 Success message notification
3.1.6 [Alt] Failed Verification
3.1.3 Request employee type details
3.1.4 Provide employee type details
3.1.6 Verify employee type details
3.2.1 Request to update employee type
D 3.1 Employee Type
3.2.2 Check Authority
3.2.2 [Alt] b Failed Verification
3. Employee Type
3.3.3. Provide employee type details
3.3.3 Request employee type details
3.2.10 Success notification
3.2.3 Display employee type details
3.2.4 Selected employee type details
3.2.5 Editable employee type details
3.2.6 [Alt] Failed Verification
3.2.6 Provide employee type details
3.3.1 Request To Search Employee
3.3.6 Check employee type existence
3.3.3 Request employee details
3.2.3 Request employee type details
3.3.4 Provide employee details
3.4.8 Disable employee type
3.3.2 [Alt] Failed authority message
3.2.9 Update employee update details
3.3.5 [Alt] Failed verification message
3.3.5 [Alt] Failed verification
3.2.8 Verification of employee type details
3.3.4 Request employee type details
3.2.8 [Alt] Failed Verification
3.3.5 Verify employee type details
3.4.2 [Alt] Failed Authority message
3.4.8 [Alt] Failed to disable
3.2.2 [Alt] Failed Authority
3.4.3 Display employee type details
3.3.2 Check Authority
3.4.7 Provide confirmation
3.4.2 [Alt] Failed Authority
3.3.2 [Alt] Failed Authority
3.4.1 Request to remove employee type
3.4.2 Check Authority
3.4.9 Display confirmation message
3.4.9 [Alt] Failed to remove message
3.3.7 Display employee details
3.4.5 Provide employee type details

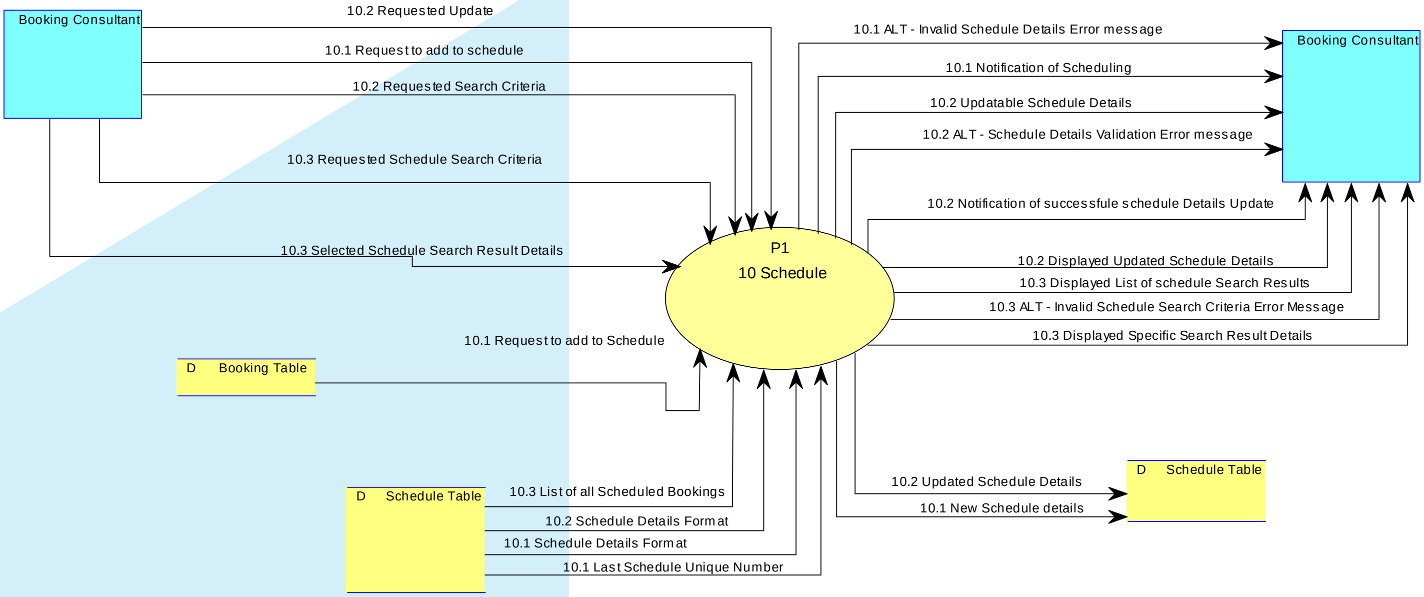

4.1 Search driver prompt
4.1 Character information
[ALT] Invalid input
Booking Consultant
4.1 Driver details
4.1.6 [ALT] Driver not retrieved
4.2 Check driver availability prompt
4.2 List of drivers
4.3 Assign driver prompt
4.3 Assignment confirmation
4.4 Outsource driver prompt
4.4 Outsourced driver assignment message
P1
4. Driver
4.1 Character match information
4.1 Driver information
4.2 Slot Details
4.3 New driver slot
4.4 Outsourced driver information
4.4 New Outsource driver name information
D Driver
D Slot
D OutSourced_ Driver
D Slot

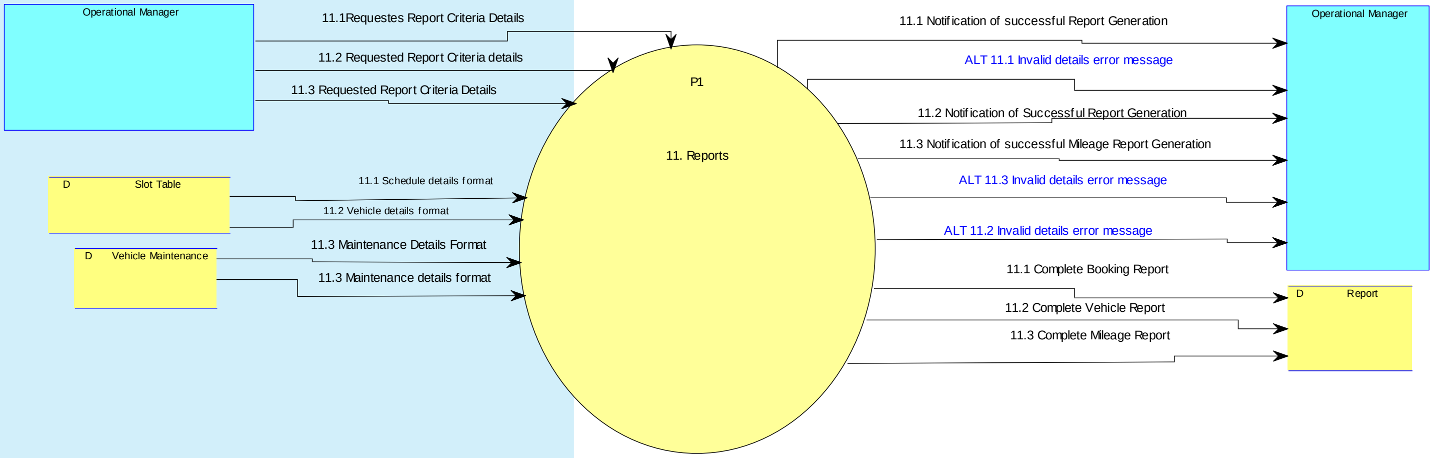

Booking Consultant
5.1 Add vehicle prompt
5.1 Added vehicle information
5.2 Search vehicle prompt
5.2 Vehicle details
5.3 Confirm trip prompt
5.3 Confirmation message
5.4 Check Vehicle availability prompt
5.4 List of available vehicles information
5.5 Assign vehicle prompt
5.5 Selected vehicle information
5.5 Vehicle assignment confirmation
5.6 Outsource vehicle prompt
5.6 Outsource vehicle assignment confirmation
Passenger
5.3 Passenger trip details
Driver
5.3 Driver Trip Details
P1
5. Vehicle
5.1 Last vehicle ID information
5.1 New vehicle information
5.1 Last Vehicle Make ID information
5.1 New vehicle make information
5.1 Last vehicle maintenance ID information
5.1 New Vehicle maintenance information
5.2 Character match information
5.2 Vehicle information
5.3 Booking reference information
5.3 Driver Name information
5.5 Passenger contact details
5.4 Trip information
5.4 Number of passengers information
5.4 Slot information
5.5 New Vehicle slot
5.6 New outsource vehicle slot
5.6 Outsourced vehicle information
D Vehicle
D Vehicle_Make
D VehicleMaintenanc e
D Vehicle
D Slot
D Driver
D Passenger
D Booking
D Vehicle_Group
D Slot
D OutSource_ Vehicle

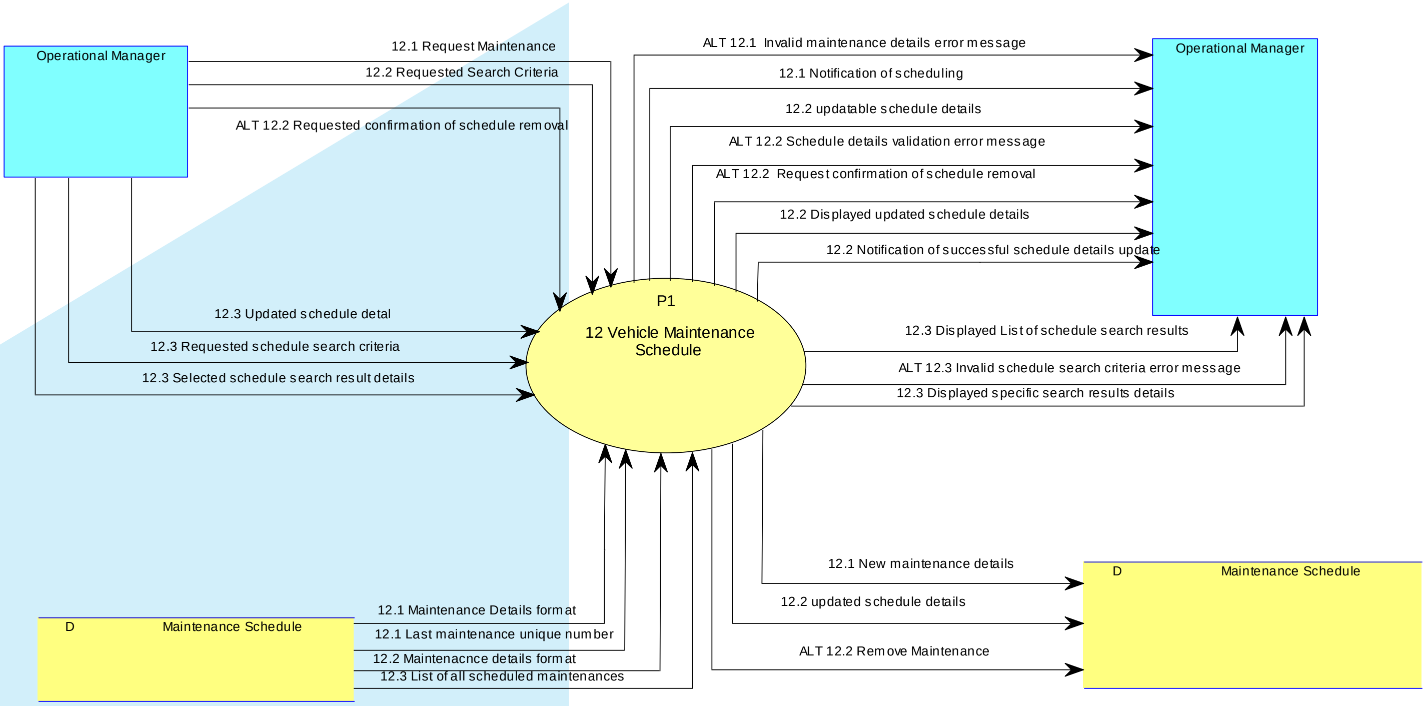

Booking Consultant
6.1 Add new vehicle group prompt
6.1 New vehicle group information
6.1 New vehicle group confirmation message
6.2 Search Vehicle Group Prompt
6.2 Vehicle Group Infomation
6.3 Update vehicle group prompt
6.3 Update Vehicle Group Information
6.3 Updated details message
P1
6. Vehicle Group
6.1 Last vehicle group ID
6.1 New Vehicle Group Details
6.2 Vehicle group name
6.2 Vehicle Group details
6.3 Updated Vehicle Group information
D Vehicle Group

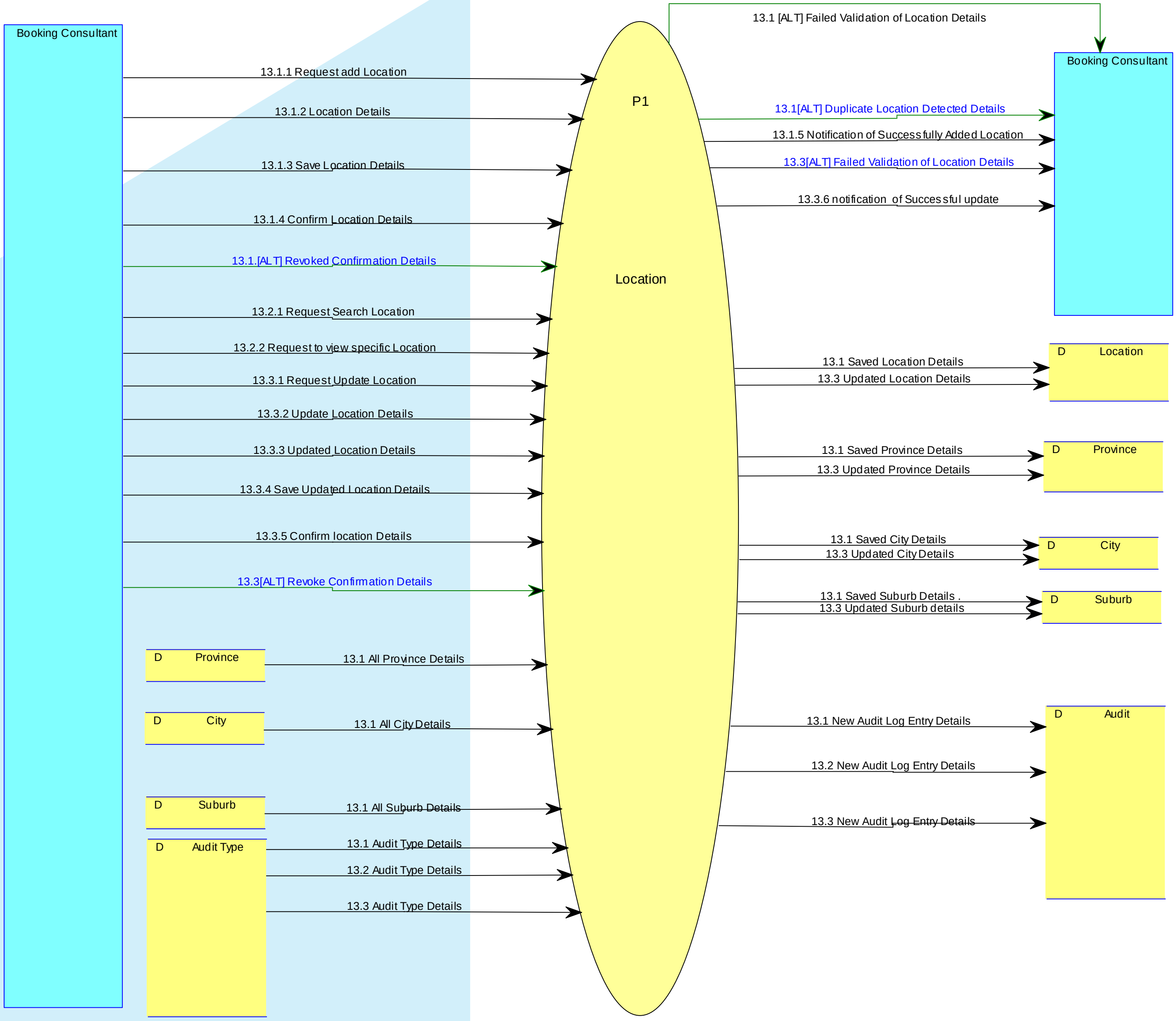

Client
7.1 Add Client Request
7.1 Provide Client Details
D Clients
7.1 Retrieve existing Clients' Client_Reference
Booking Consultant
7.2 Request to Retrieve Client Details
D Client
7.2. Retrieve All Client Details
Booking Consultant
7.2 Provide The client name
7.2 Select specific Client
Client
7.3 Request to update a client
7.3 Provide Client Reference
Client
7.3 Provide Updated Client Details
[ALT] Invalid format
P1
7 Client
7.1 Prompt Client Details
[ALT] Invalid information
[ALT] Client Exists
7.1 Confirmation Communicated to the client
Client
7.1 Save Client Details
D Clients
7.2 All Client Details
7.2 Prompt Client Name
[ALT] Invalid Client Reference
7.2 Narrowed Client details
[ALT] Client does not exist
7.2 Selected Client details
Booking Consultant
[ALT] Client not retrieved
7.3 Prompt the Client Details
client
7.3 Updated client details
D Client
Operations Manager
8.1 Request to Add Client Type
8.1 Provide Client Type Details
D Client_Type
8.1 Retrieved exisiting client types
Booking Consultant
8.2 Request to Retrieve Client Type Details
8.2 Provide The clienttype name
8.2 Select specific ClientType
D Client_Type
8.2. Retrieve All Client Type Details
Operations Manager
8.3 Request to update a clientType
8.3 Provide Update Client Type details
P1
8. Client Type
8.1 Prompts Client Type details
[ALT] Incorrect Format
[ALT] Client Type already Exists
Operations Manager
8.1 New Client Type details
D Client_Type
8.2 All Client Type Details
8.2 Prompt ClientTypeName
[ALT] Invalid Client type name
8.2 Narrowed ClientType details
[ALT] ClientType does not exist
8.2 Selected Client type details
Booking Consultant
[ALT] Client Type Not Retrieved
8.3 Prompt Update Client Type Details
[ALT] Invalid Client Type Details
8.3 Confirmation of Updating Client Type details
Operations Manager
8.3 Updated Client Type Details
D Client_Type

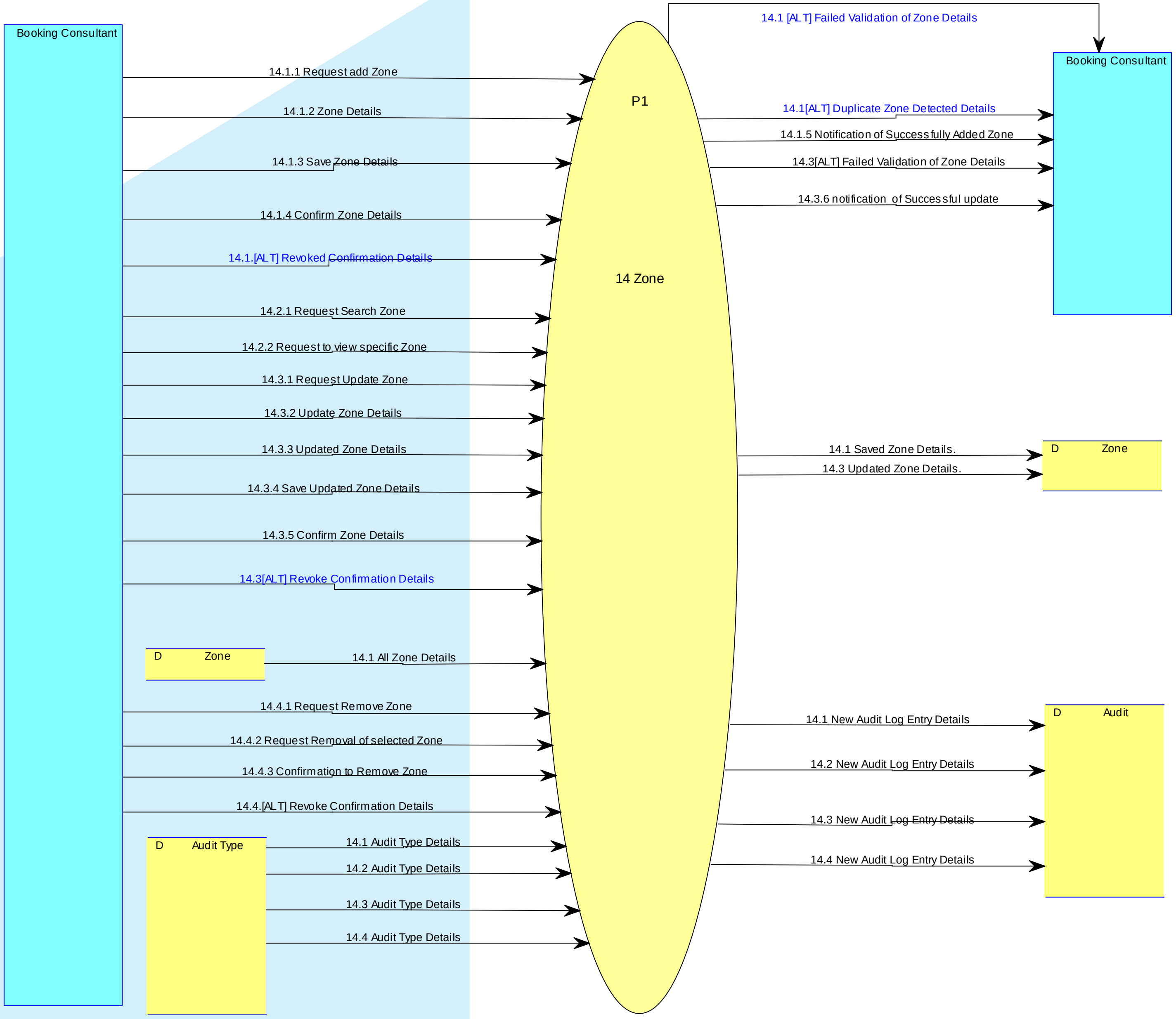
Book Trip Appendix

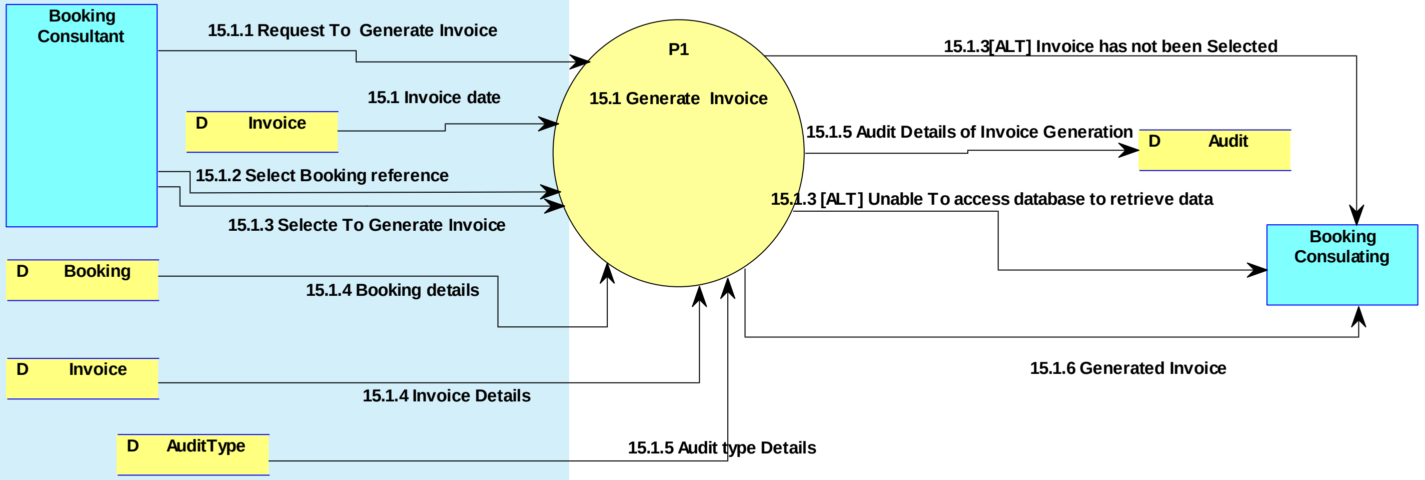

Booking Consultant
10.2 Requested Update
10.1 Request to add to schedule
10.2 Requested Search Criteria
10.3 Requested Schedule Search Criteria
10.3 Selected Schedule Search Result Details
10.1 ALT - Invalid Schedule Details Error message
10.1 Notification of Scheduling
10.2 Updatable Schedule Details
10.2 ALT - Schedule Details Validation Error message
10.2 Notification of successfule schedule Details Update
Booking Consultant
P1
10 Schedule
10.2 Displayed Updated Schedule Details
10.3 Displayed List of schedule Search Results
10.3 ALT - Invalid Schedule Search Criteria Error Message
10.3 Displayed Specific Search Result Details
10.1 Request to add to Schedule
D Booking Table
D Schedule Table
10.3 List of all Scheduled Bookings
10.2 Schedule Details Format
10.1 Schedule Details Format
10.1 Last Schedule Unique Number
10.2 Updated Schedule Details
10.1 New Schedule details
D Schedule Table


Operational Manager
11.1Requestes Report Criteria Details
11.2 Requested Report Criteria details
11.3 Requested Report Criteria Details
D Slot Table
11.1 Schedule details format
11.2 Vehicle details format
D Vehicle Maintenance
11.3 Maintenance Details Format
11.3 Maintenance details format
P1
11. Reports
11.1 Notification of successful Report Generation
ALT 11.1 Invalid details error message
11.2 Notification of Successful Report Generation
11.3 Notification of successful Mileage Report Generation
ALT 11.3 Invalid details error message
ALT 11.2 Invalid details error message
11.1 Complete Booking Report
11.2 Complete Vehicle Report
11.3 Complete Mileage Report
Operational Manager
D Report

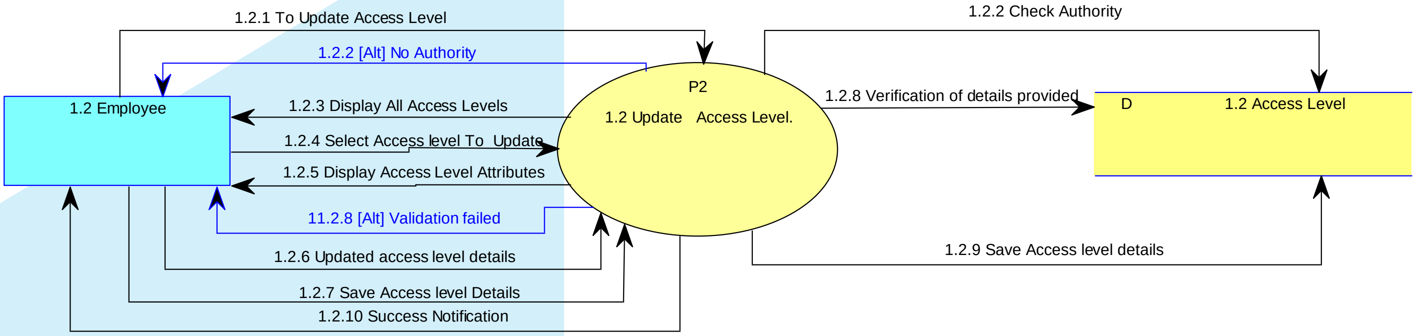

Operational Manager
12.1 Request Maintenance
12.2 Requested Search Criteria
ALT 12.2 Requested confirmation of schedule removal
12.3 Updated schedule detal
12.3 Requested schedule search criteria
12.3 Selected schedule search result details
ALT 12.1 Invalid maintenance details error message
12.1 Notification of scheduling
12.2 updatable schedule details
ALT 12.2 Schedule details validation error message
ALT 12.2 Request confirmation of schedule removal
12.2 Displayed updated schedule details
12.2 Notification of successful schedule details update
Operational Manager
P1
12 Vehicle Maintenance Schedule
12.3 Displayed List of schedule search results
ALT 12.3 Invalid schedule search criteria error message
12.3 Displayed specific search results details
12.1 New maintenance details
12.2 updated schedule details
ALT 12.2 Remove Maintenance
D
Maintenance Schedule
D
Maintenance Schedule
12.1 Maintenance Details format
12.1 Last maintenance unique number
12.2 Maintenacnce details format
12.3 List of all scheduled maintenances

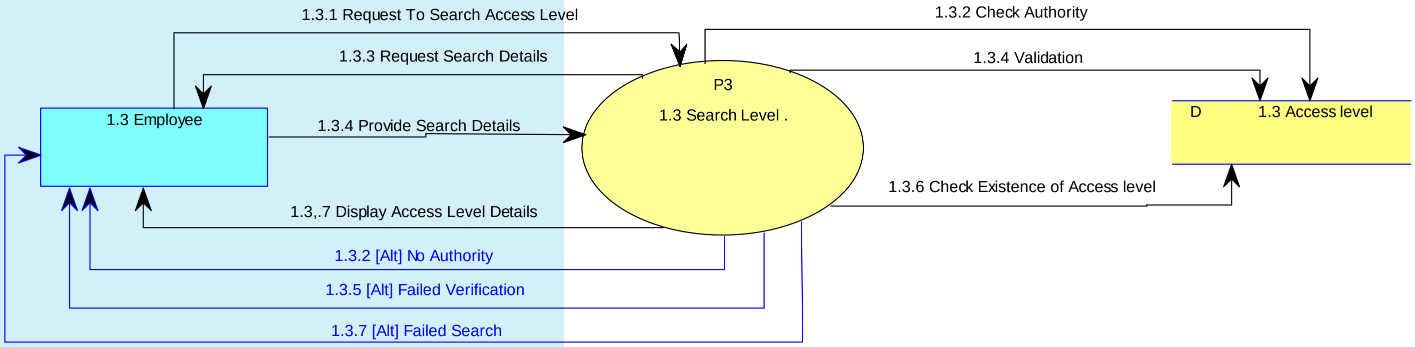

Booking Consultant
13.1.1 Request add Location
13.1.2 Location Details
13.1.3 Save Location Details
13.1.4 Confirm Location Details
13.1.[ALT] Revoked Confirmation Details
13.2.1 Request Search Location
13.2.2 Request to view specific Location
13.3.1 Request Update Location
13.3.2 Update Location Details
13.3.3 Updated Location Details
13.3.4 Save Updated Location Details
13.3.5 Confirm location Details
13.3[ALT] Revoke Confirmation Details
D Province
13.1 All Province Details
D City
13.1 All City Details
D Suburb
13.1 All Suburb Details
D Audit Type
13.1 Audit Type Details
13.2 Audit Type Details
13.3 Audit Type Details
P1
Location
13.1 [ALT] Failed Validation of Location Details
Booking Consultant
13.1[ALT] Duplicate Location Detected Details
13.1.5 Notification of Successfully Added Location
13.3[ALT] Failed Validation of Location Details
13.3.6 notification of Successful update
D Location
13.1 Saved Location Details
13.3 Updated Location Details
D Province
13.1 Saved Province Details
13.3 Updated Province Details
D City
13.1 Saved City Details
13.3 Updated City Details
D Suburb
13.1 Saved Suburb Details .
13.3 Updated Suburb details
D Audit
13.1 New Audit Log Entry Details
13.2 New Audit Log Entry Details
13.3 New Audit Log Entry Details

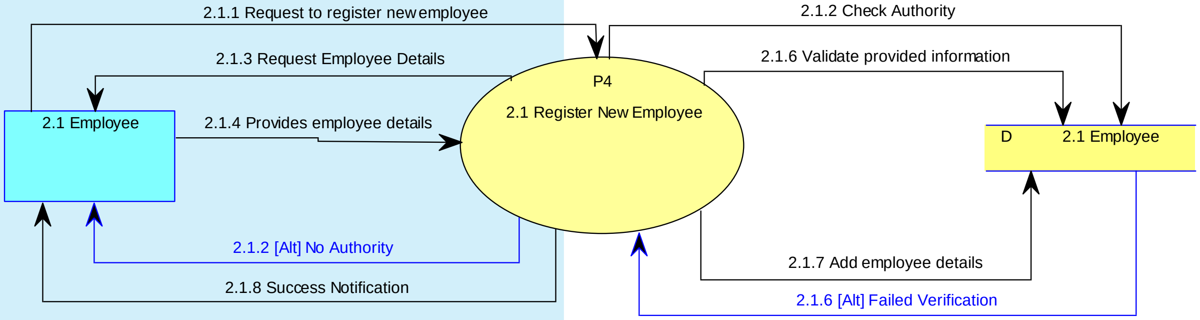

Booking Consultant
14.1.1 Request add Zone
14.1.2 Zone Details
14.1.3 Save Zone Details
14.1.4 Confirm Zone Details
14.1.[ALT] Revoked Confirmation Details
14.2.1 Request Search Zone
14.2.2 Request to view specific Zone
14.3.1 Request Update Zone
14.3.2 Update Zone Details
14.3.3 Updated Zone Details
14.3.4 Save Updated Zone Details
14.3.5 Confirm Zone Details
14.3[ALT] Revoke Confirmation Details
D Zone
14.1 All Zone Details
14.4.1 Request Remove Zone
14.4.2 Request Removal of selected Zone
14.4.3 Confirmation to Remove Zone
14.4.[ALT] Revoke Confirmation Details
D Audit Type
14.1 Audit Type Details
14.2 Audit Type Details
14.3 Audit Type Details
14.4 Audit Type Details
P1
14 Zone
14.1 [ALT] Failed Validation of Zone Details
Booking Consultant
14.1[ALT] Duplicate Zone Detected Details
14.1.5 Notification of Successfully Added Zone
14.3[ALT] Failed Validation of Zone Details
14.3.6 notification of Successful update
14.1 Saved Zone Details.
14.3 Updated Zone Details.
D Zone
14.1 New Audit Log Entry Details
14.2 New Audit Log Entry Details
14.3 New Audit Log Entry Details
14.4 New Audit Log Entry Details
D Audit

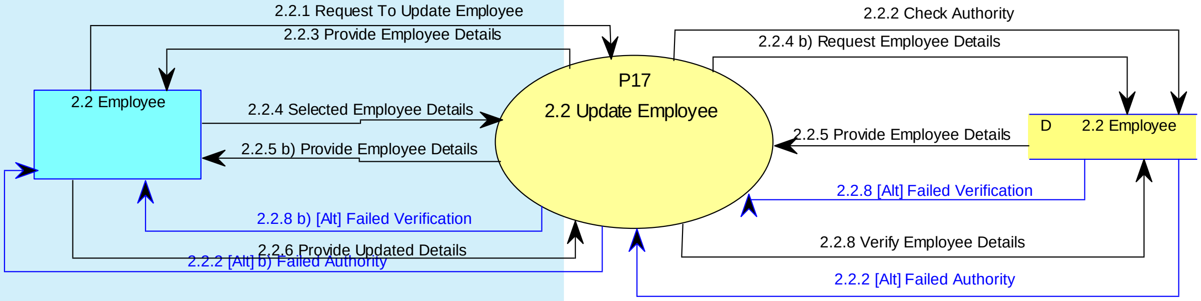

Booking Consultant
15.1.1 Request To Generate Invoice
D Invoice
15.1 Invoice date
15.1.2 Select Booking reference
15.1.3 Selecte To Generate Invoice
D Booking
15.1.4 Booking details
D Invoice
15.1.4 Invoice Details
D AuditType
15.1.5 Audit type Details
P1
15.1 Generate Invoice
15.1.3[ALT] Invoice has not been Selected
15.1.5 Audit Details of Invoice Generation
D Audit
15.1.3 [ALT] Unable To access database to retrieve data
Booking Consulating
15.1.6 Generated Invoice

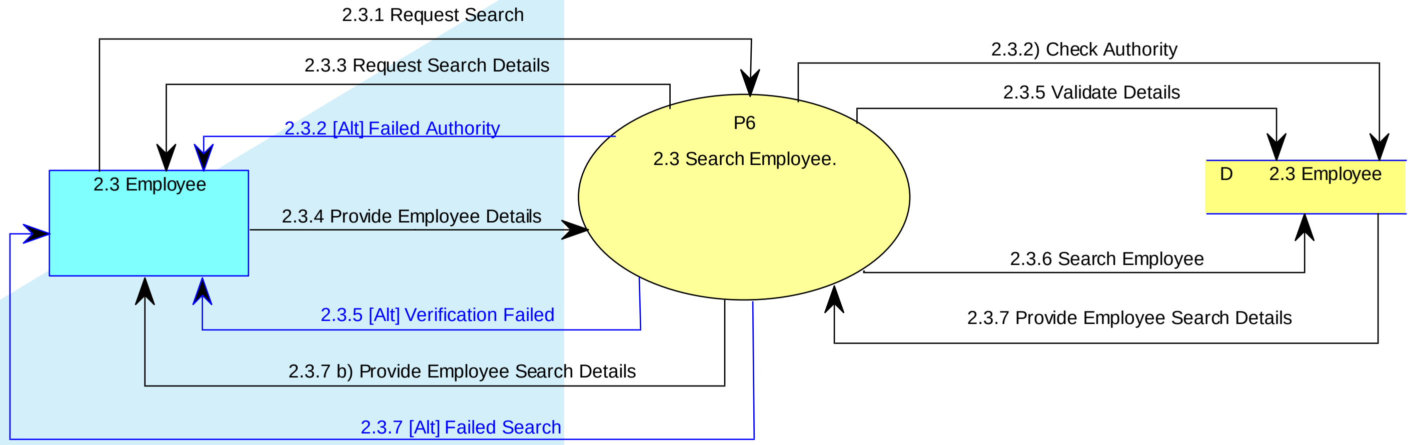
Mid Level

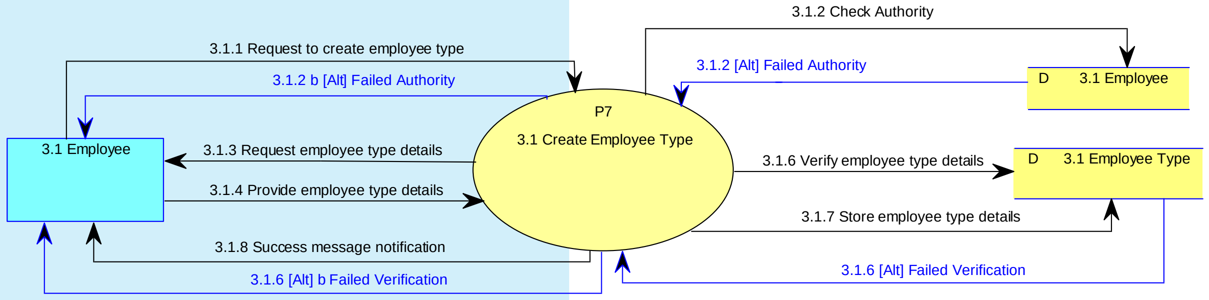

1.2.1 To Update Access Level
1.2.2 [Alt] No Authority
1.2.2 Check Authority
1.2 Employee
1.2.3 Display All Access Levels
1.2.4 Select Access level To Update
1.2.5 Display Access Level Attributes
11.2.8 [Alt] Validation failed
1.2.6 Updated access level details
1.2.7 Save Access level Details
1.2.10 Success Notification
P2
1.2 Update Access Level.
1.2.8 Verification of details provided
D
1.2 Access Level
1.2.9 Save Access level details

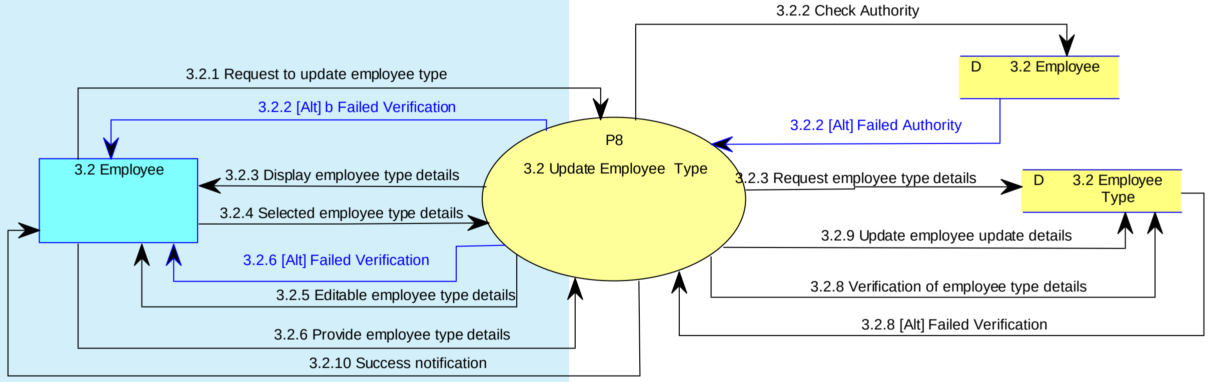

1.3.1 Request To Search Access Level
1.3.3 Request Search Details
1.3 Employee
1.3.4 Provide Search Details
1.3,.7 Display Access Level Details
1.3.2 [Alt] No Authority
1.3.5 [Alt] Failed Verification
1.3.7 [Alt] Failed Search
P3
1.3 Search Level .
1.3.2 Check Authority
1.3.4 Validation
D
1.3 Access level
1.3.6 Check Existence of Access level

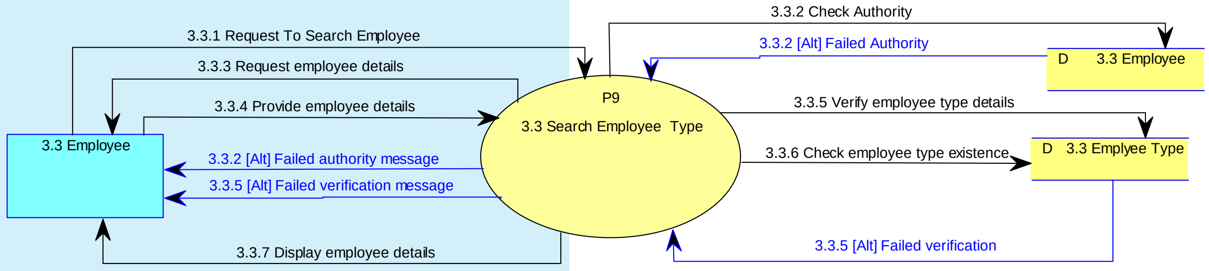

2.1.1 Request to register new employee
2.1.3 Request Employee Details
2.1 Employee
2.1.4 Provides employee details
2.1.2 [Alt] No Authority
2.1.8 Success Notification
P4
2.1 Register New Employee
2.1.2 Check Authority
2.1.6 Validate provided information
D
2.1 Employee
2.1.7 Add employee details
2.1.6 [Alt] Failed Verification

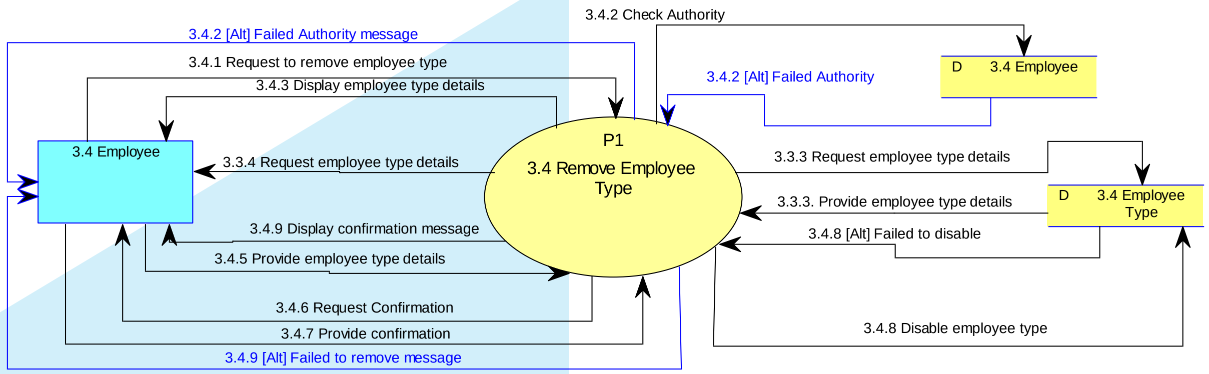

2.2.1 Request To Update Employee
2.2.3 Provide Employee Details
2.2 Employee
2.2.4 Selected Employee Details
2.2.5 b) Provide Employee Details
2.2.8 b) [Alt] Failed Verification
2.2.6 Provide Updated Details
2.2.2 [Alt] b) Failed Authority
P17
2.2 Update Employee
2.2.2 Check Authority
2.2.4 b) Request Employee Details
2.2.5 Provide Employee Details
D
2.2 Employee
2.2.8 [Alt] Failed Verification
2.2.8 Verify Employee Details
2.2.2 [Alt] Failed Authority

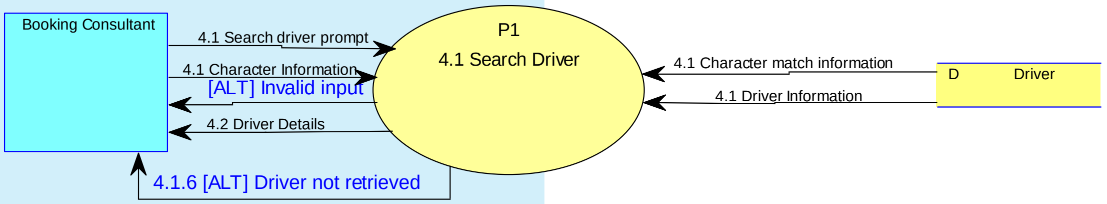

2.3.1 Request Search
2.3.3 Request Search Details
2.3.2 [Alt] Failed Authority
2.3.2) Check Authority
2.3.5 Validate Details
P6
2.3 Search Employee.
2.3 Employee
2.3.4 Provide Employee Details
D
2.3 Employee
2.3.6 Search Employee
2.3.5 [Alt] Verification Failed
2.3.7 Provide Employee Search Details
2.3.7 b) Provide Employee Search Details
2.3.7 [Alt] Failed Search

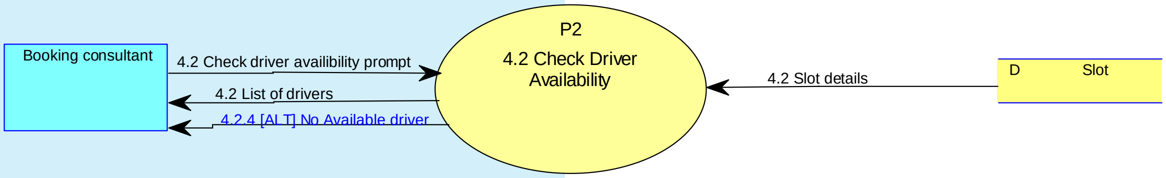

3.1.2 Check Authority
3.1.1 Request to create employee type
3.1.2 [Alt] Failed Authority
3.1.2 b [Alt] Failed Authority
D
3.1 Employee
P7
3.1 Create Employee Type
3.1 Employee
3.1.3 Request employee type details
3.1.6 Verify employee type details
D
3.1 Employee Type
3.1.4 Provide employee type details
3.1.7 Store employee type details
3.1.8 Success message notification
3.1.6 [Alt] Failed Verification
3.1.6 [Alt] b Failed Verification

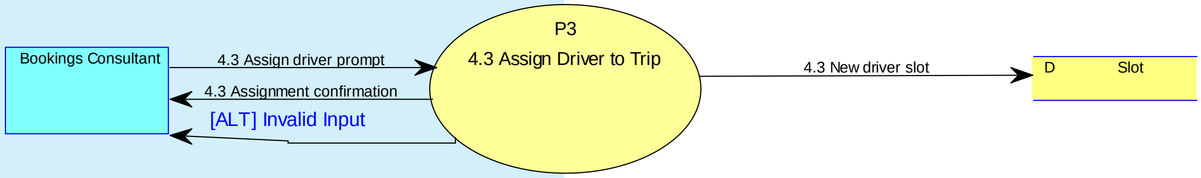

3.2.2 Check Authority
D
3.2 Employee
3.2.1 Request to update employee type
3.2.2 [Alt] b Failed Verification
3.2.2 [Alt] Failed Authority
P8
3.2 Update Employee Type
3.2 Employee
3.2.3 Display employee type details
3.2.3 Request employee type details
D
3.2 Employee Type
3.2.4 Selected employee type details
3.2.9 Update employee update details
3.2.6 [Alt] Failed Verification
3.2.5 Editable employee type details
3.2.8 Verification of employee type details
3.2.8 [Alt] Failed Verification
3.2.6 Provide employee type details
3.2.10 Success notification

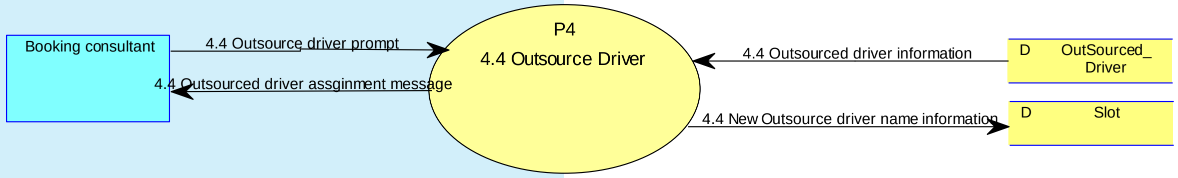

3.3.2 Check Authority
3.3.1 Request To Search Employee
3.3.2 [Alt] Failed Authority
D
3.3 Employee
3.3.3 Request employee details
P9
3.3 Search Employee Type
3.3.4 Provide employee details
3.3.5 Verify employee type details
3.3 Employee
3.3.6 Check employee type existence
D
3.3 Emplyee Type
3.3.2 [Alt] Failed authority message
3.3.5 [Alt] Failed verification message
3.3.5 [Alt] Failed verification
3.3.7 Display employee details

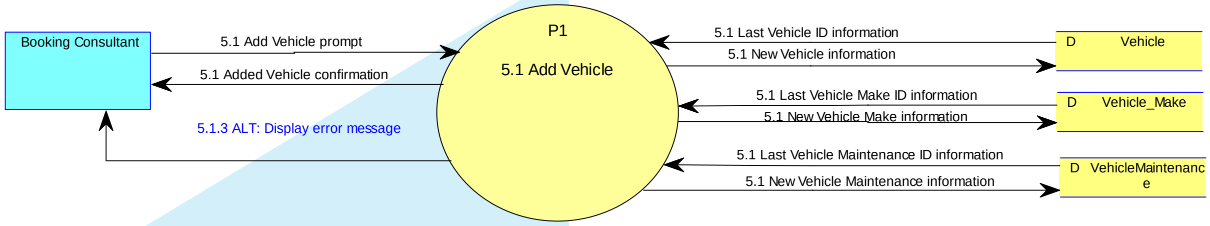

3.4.2 Check Authority
3.4.2 [Alt] Failed Authority message
3.4.1 Request to remove employee type
3.4.3 Display employee type details
3.4.2 [Alt] Failed Authority
D 3.4 Employee
3.4 Employee
P1
3.4 Remove Employee Type
3.3.4 Request employee type details
3.3.3 Request employee type details
3.3.3. Provide employee type details
D 3.4 Employee Type
3.4.9 Display confirmation message
3.4.8 [Alt] Failed to disable
3.4.5 Provide employee type details
3.4.6 Request Confirmation
3.4.7 Provide confirmation
3.4.8 Disable employee type
3.4.9 [Alt] Failed to remove message

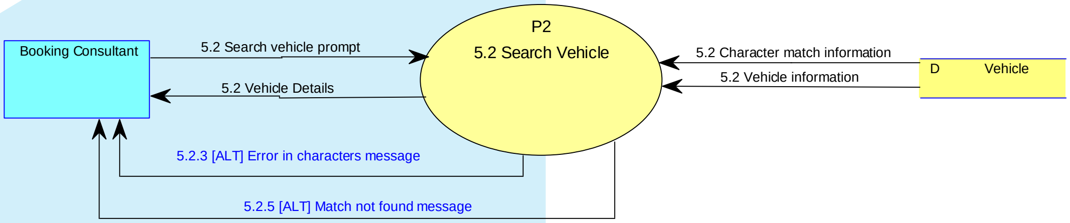

Booking Consultant
4.1 Search driver prompt
4.1 Character Information
[ALT] Invalid input
4.2 Driver Details
4.1.6 [ALT] Driver not retrieved
P1
4.1 Search Driver
4.1 Character match information
4.1 Driver Information
D Driver

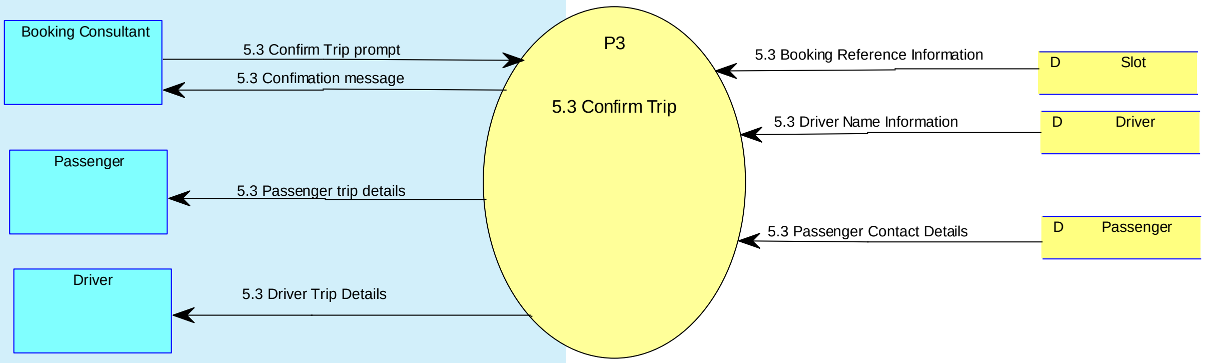

Booking consultant
4.2 Check driver availibility prompt
4.2 List of drivers
4.2.4 [ALT] No Available driver
P2
4.2 Check Driver Availability
4.2 Slot details
D Slot

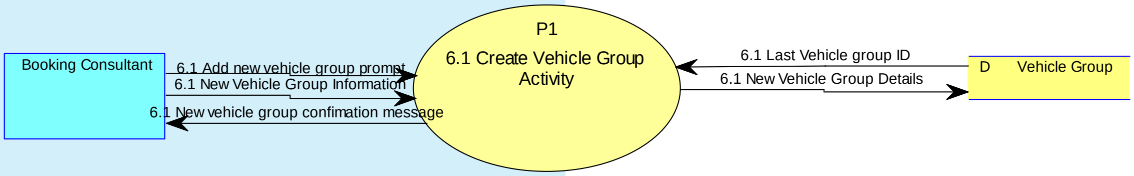

Bookings Consultant
4.3 Assign driver prompt
4.3 Assignment confirmation
[ALT] Invalid Input
P3
4.3 Assign Driver to Trip
4.3 New driver slot
D Slot

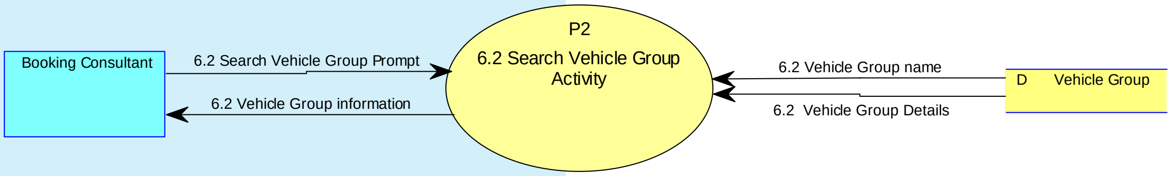

Booking consultant
4.4 Outsource driver prompt
4.4 Outsourced driver assginment message
P4
4.4 Outsource Driver
4.4 Outsourced driver information
D OutSourced_ Driver
4.4 New Outsource driver name information
D Slot

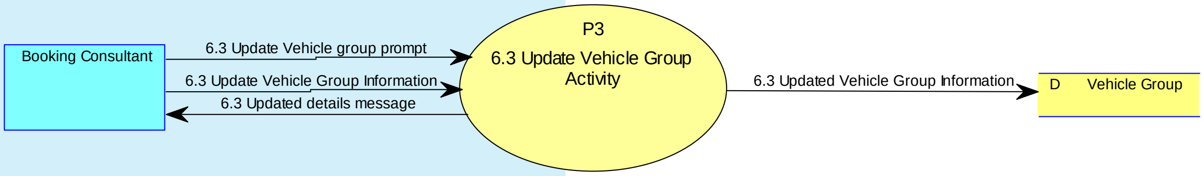

Booking Consultant
5.1 Add Vehicle prompt
5.1 Added Vehicle confirmation
5.1.3 ALT: Display error message
P1
5.1 Add Vehicle
5.1 Last Vehicle ID information
5.1 New Vehicle information
D Vehicle
5.1 Last Vehicle Make ID information
5.1 New Vehicle Make information
D Vehicle_Make
5.1 Last Vehicle Maintenance ID information
5.1 New Vehicle Maintenance information
D VehicleMaintenanc e

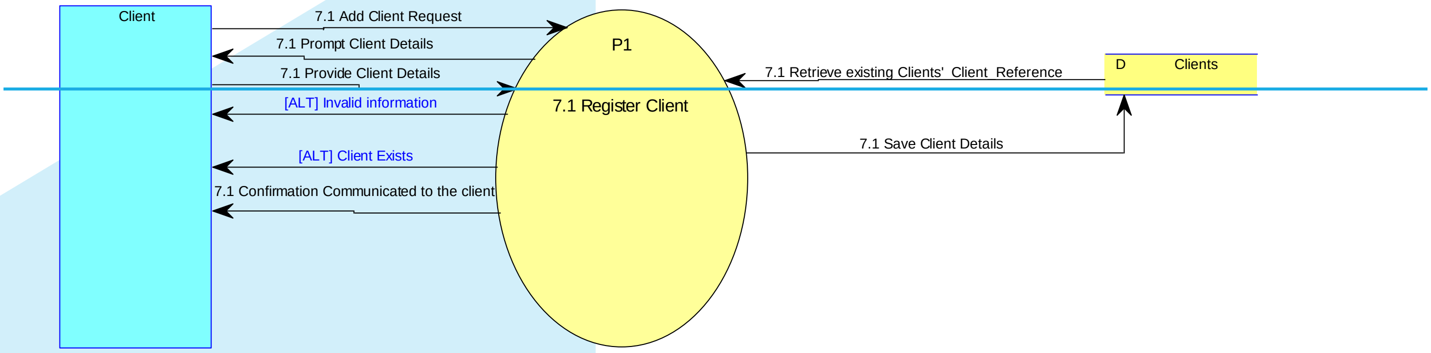

Booking Consultant
5.2 Search vehicle prompt
5.2 Vehicle Details
5.2.3 [ALT] Error in characters message
5.2.5 [ALT] Match not found message
P2
5.2 Search Vehicle
5.2 Character match information
5.2 Vehicle information
D Vehicle

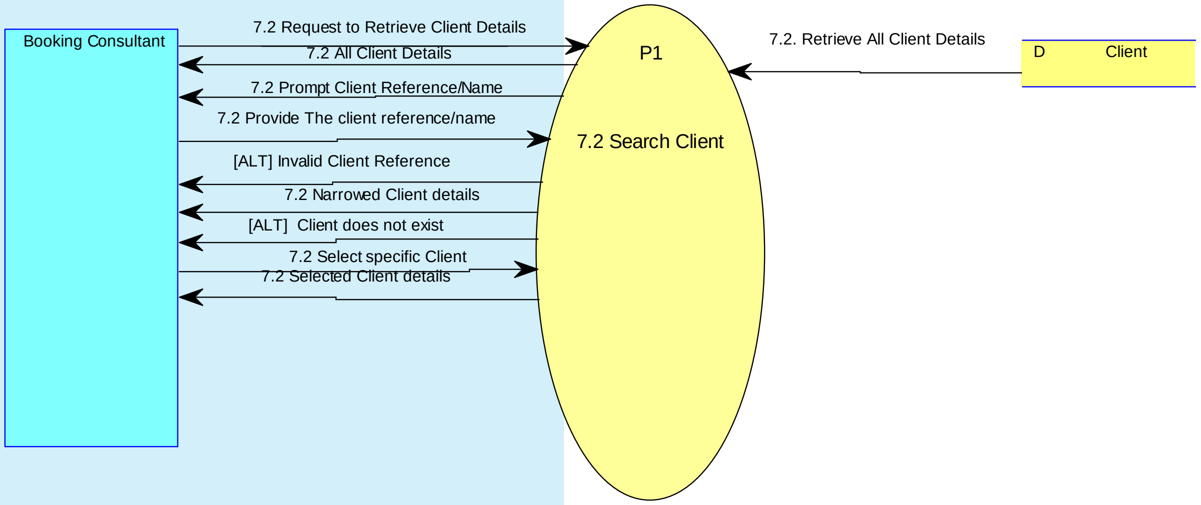

Booking Consultant
5.3 Confirm Trip prompt
5.3 Confimation message
Passenger
5.3 Passenger trip details
Driver
5.3 Driver Trip Details
P3
5.3 Confirm Trip
5.3 Booking Reference Information
D Slot
5.3 Driver Name Information
D Driver
5.3 Passenger Contact Details
D Passenger

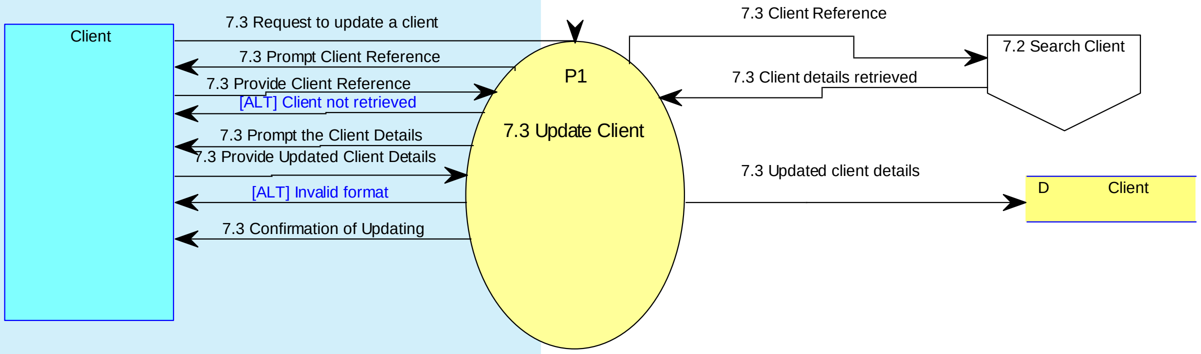

Booking Consultant
6.1 Add new vehicle group prompt
6.1 New Vehicle Group Information
6.1 New vehicle group confimation message
P1
6.1 Create Vehicle Group Activity
6.1 Last Vehicle group ID
6.1 New Vehicle Group Details
D Vehicle Group

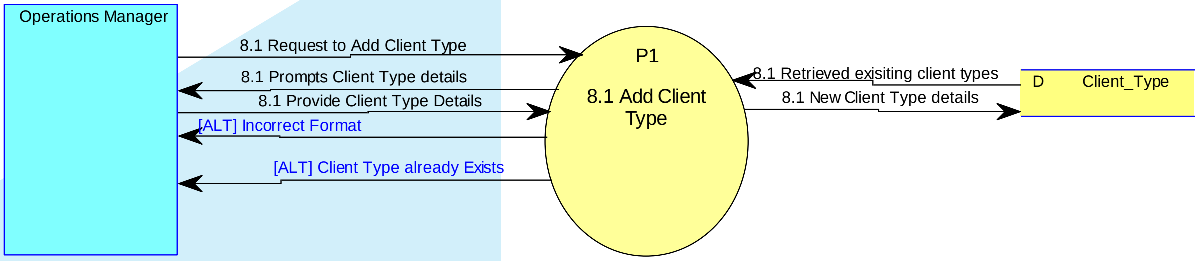

Booking Consultant
6.2 Search Vehicle Group Prompt
6.2 Vehicle Group information
P2
6.2 Search Vehicle Group Activity
6.2 Vehicle Group name
6.2 Vehicle Group Details
D Vehicle Group

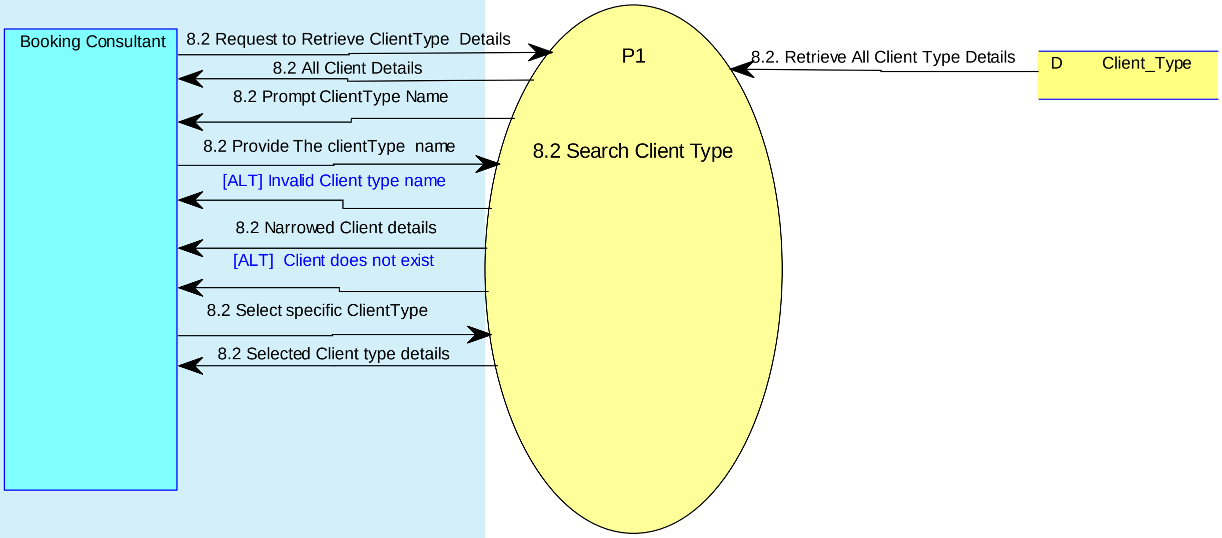

Booking Consultant
6.3 Update Vehicle group prompt
6.3 Update Vehicle Group Information
6.3 Updated details message
P3
6.3 Update Vehicle Group Activity
6.3 Updated Vehicle Group Information
D Vehicle Group

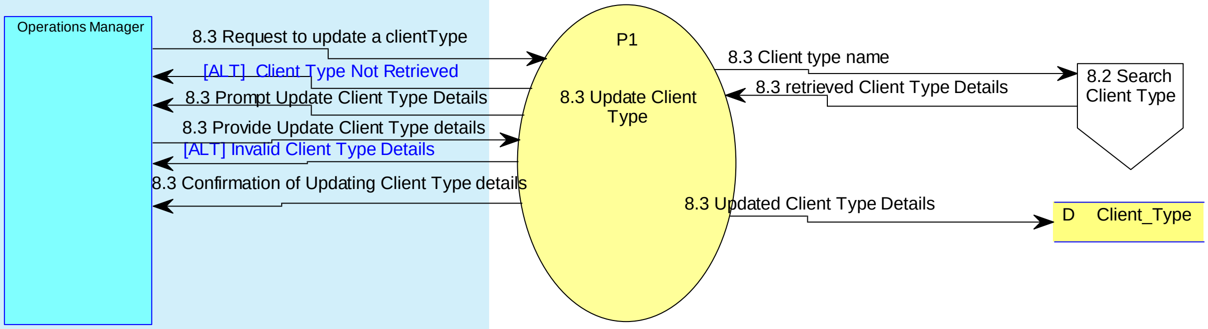

Client
7.1 Add Client Request
7.1 Prompt Client Details
7.1 Provide Client Details
[ALT] Invalid information
[ALT] Client Exists
7.1 Confirmation Communicated to the client
P1
7.1 Register Client
7.1 Retrieve existing Clients' Client Reference
D
Clients
7.1 Save Client Details

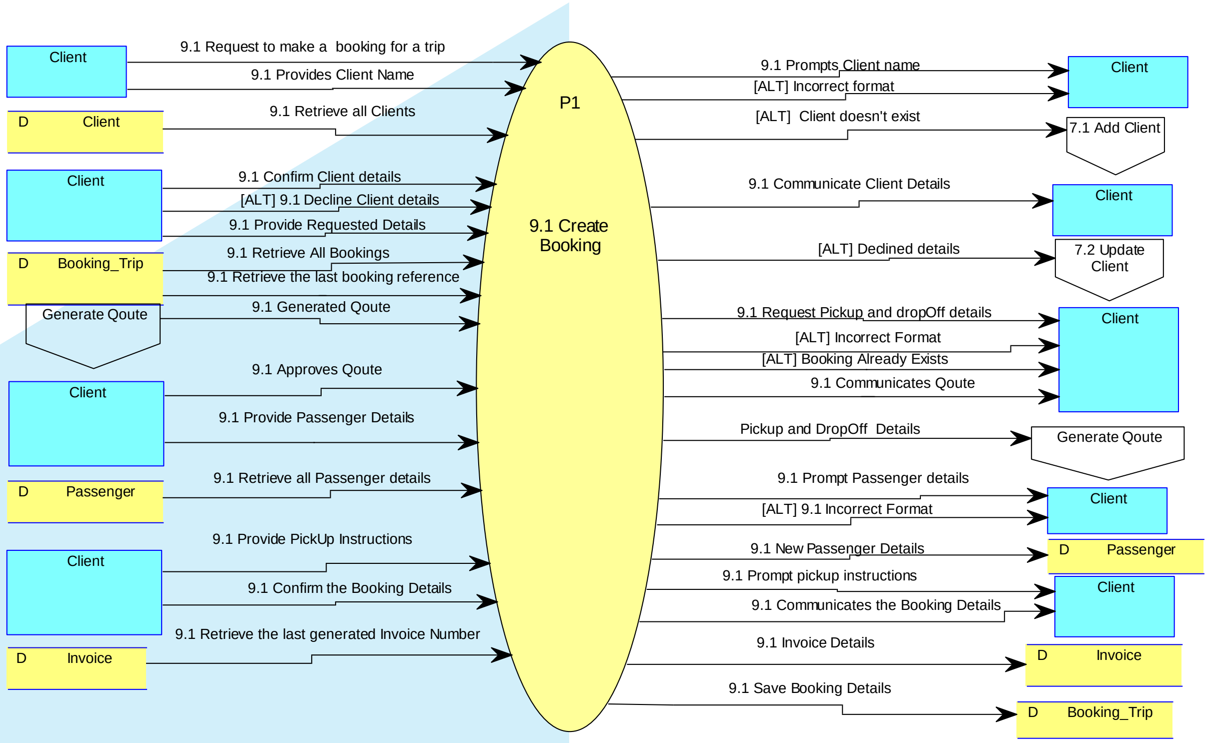

Booking Consultant
7.2 Request to Retrieve Client Details
7.2 All Client Details
7.2 Prompt Client Reference/Name
7.2 Provide The client reference/name
[ALT] Invalid Client Reference
7.2 Narrowed Client details
[ALT] Client does not exist
7.2 Select specific Client
7.2 Selected Client details
P1
7.2 Search Client
7.2. Retrieve All Client Details
D
Client

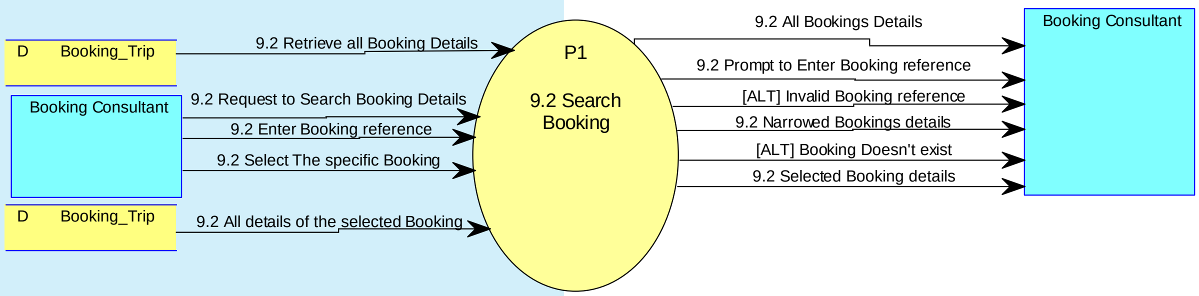

Client
7.3 Request to update a client
7.3 Prompt Client Reference
7.3 Provide Client Reference
[ALT] Client not retrieved
7.3 Prompt the Client Details
7.3 Provide Updated Client Details
[ALT] Invalid format
7.3 Confirmation of Updating
P1
7.3 Update Client
7.3 Client Reference
7.2 Search Client
7.3 Client details retrieved
7.3 Updated client details
D
Client

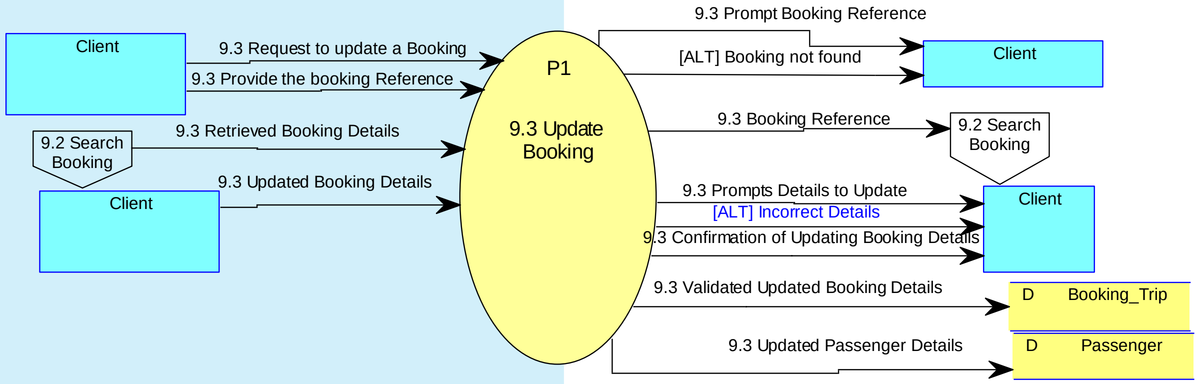

Operations Manager
8.1 Request to Add Client Type
8.1 Prompts Client Type details
8.1 Provide Client Type Details
[ALT] Incorrect Format
[ALT] Client Type already Exists
P1
8.1 Add Client Type
8.1 Retrieved exisiting client types
8.1 New Client Type details
D
Client_Type

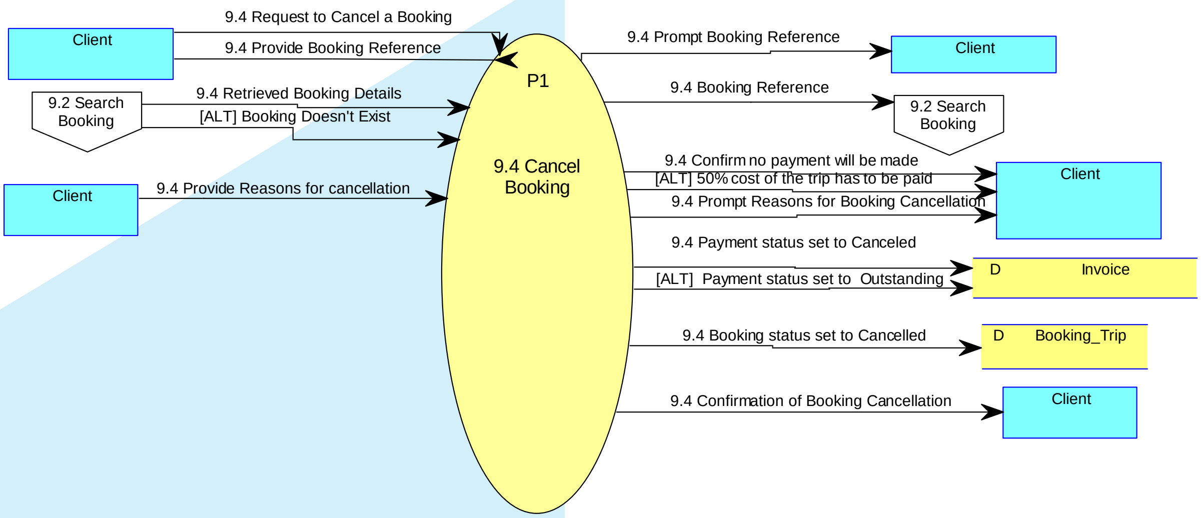

Booking Consultant
8.2 Request to Retrieve ClientType Details
8.2 All Client Details
8.2 Prompt ClientType Name
8.2 Provide The clientType name
[ALT] Invalid Client type name
8.2 Narrowed Client details
[ALT] Client does not exist
8.2 Select specific ClientType
8.2 Selected Client type details
P1
8.2 Search Client Type
8.2. Retrieve All Client Type Details
D
Client_Type

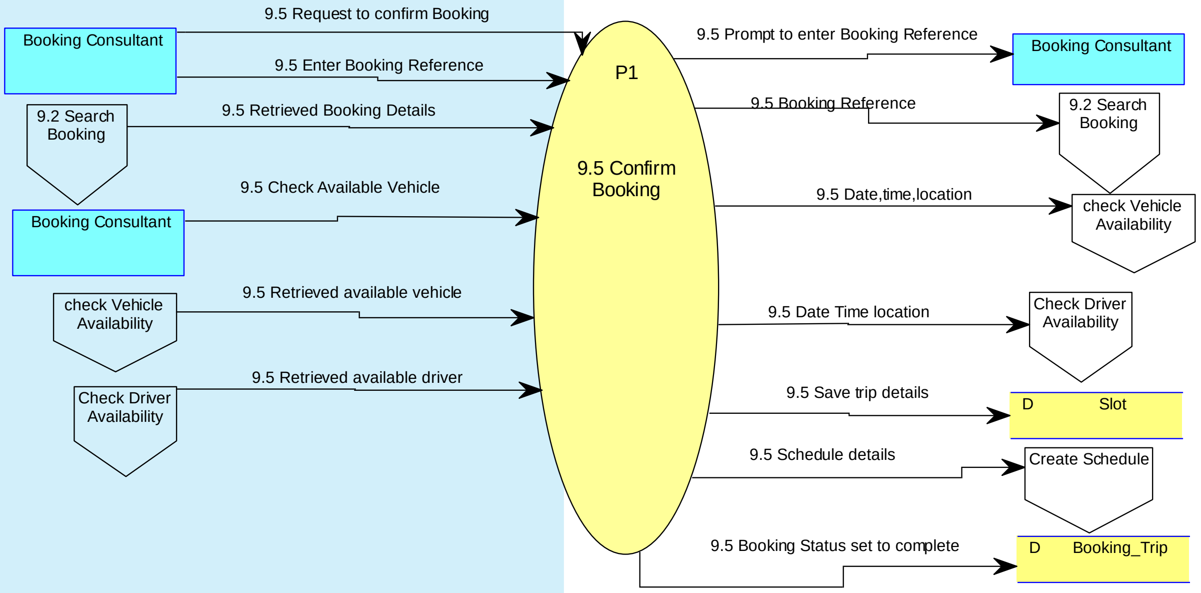

Operations Manager
8.3 Request to update a clientType
[ALT] Client Type Not Retrieved
8.3 Prompt Update Client Type Details
8.3 Provide Update Client Type details
[ALT] Invalid Client Type Details
8.3 Confirmation of Updating Client Type details
P1
8.3 Update Client Type
8.3 Client type name
8.3 retrieved Client Type Details
8.2 Search Client Type
8.3 Updated Client Type Details
D
Client_Type

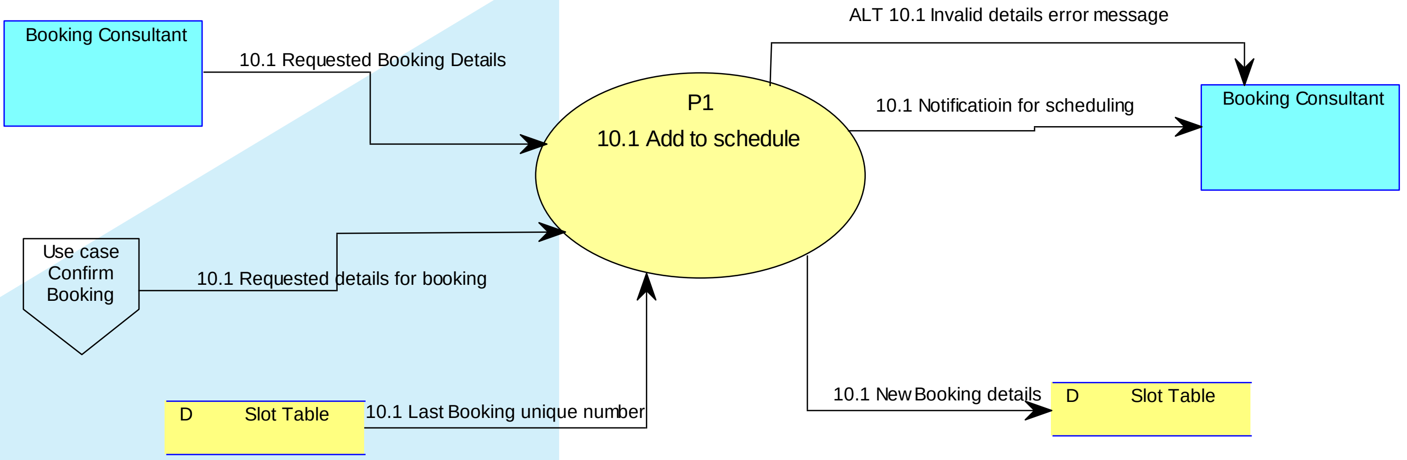

P1
9.1 Create Booking
Client
9.1 Request to make a booking for a trip
9.1 Provides Client Name
D Client
9.1 Retrieve all Clients
Client
9.1 Confirm Client details
[ALT] 9.1 Decline Client details
9.1 Provide Requested Details
D Booking_Trip
9.1 Retrieve All Bookings
9.1 Retrieve the last booking reference
Generate Qoute
9.1 Generated Qoute
Client
9.1 Approves Qoute
9.1 Provide Passenger Details
D Passenger
9.1 Retrieve all Passenger details
Client
9.1 Provide PickUp Instructions
9.1 Confirm the Booking Details
D Invoice
9.1 Retrieve the last generated Invoice Number
9.1 Prompts Client name
[ALT] Incorrect format
Client
[ALT] Client doesn't exist
7.1 Add Client
9.1 Communicate Client Details
Client
[ALT] Declined details
7.2 Update Client
9.1 Request Pickup and dropOff details
[ALT] Incorrect Format
[ALT] Booking Already Exists
9.1 Communicates Qoute
Client
Pickup and DropOff Details
Generate Qoute
9.1 Prompt Passenger details
[ALT] 9.1 Incorrect Format
Client
9.1 New Passenger Details
D Passenger
9.1 Prompt pickup instructions
9.1 Communicates the Booking Details
Client
9.1 Invoice Details
D Invoice
9.1 Save Booking Details
D Booking_Trip

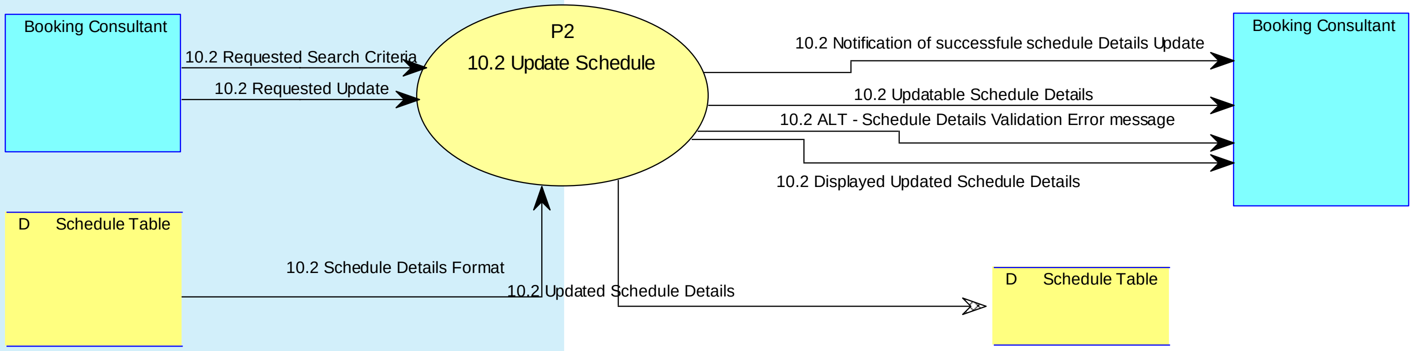

P1
9.2 Search Booking
D Booking_Trip
9.2 Retrieve all Booking Details
Booking Consultant
9.2 Request to Search Booking Details
9.2 Enter Booking reference
9.2 Select The specific Booking
D Booking_Trip
9.2 All details of the selected Booking
9.2 All Bookings Details
9.2 Prompt to Enter Booking reference
[ALT] Invalid Booking reference
9.2 Narrowed Bookings details
[ALT] Booking Doesn't exist
9.2 Selected Booking details
Booking Consultant

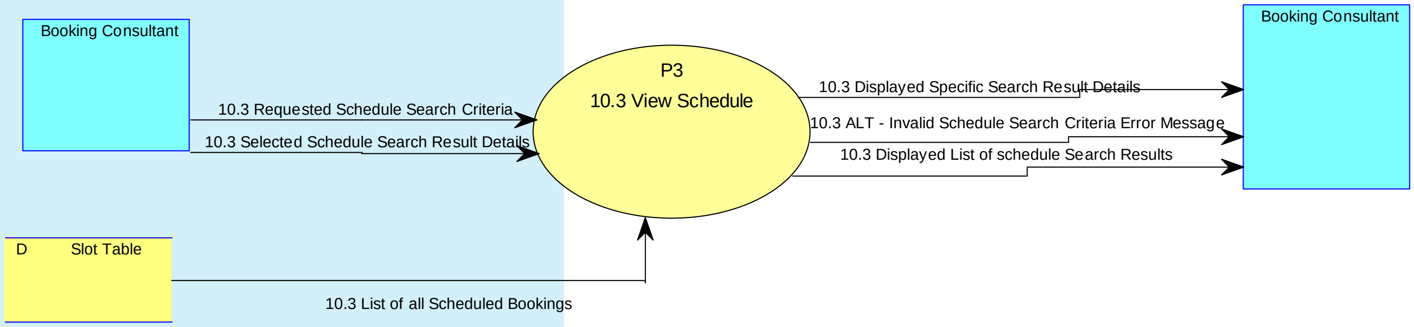

P1
9.3 Update Booking
Client
9.3 Request to update a Booking
9.3 Provide the booking Reference
9.2 Search Booking
9.3 Retrieved Booking Details
Client
9.3 Updated Booking Details
9.3 Prompt Booking Reference
[ALT] Booking not found
Client
9.3 Booking Reference
9.2 Search Booking
9.3 Prompts Details to Update
[ALT] Incorrect Details
9.3 Confirmation of Updating Booking Details
Client
9.3 Validated Updated Booking Details
D Booking_Trip
9.3 Updated Passenger Details
D Passenger

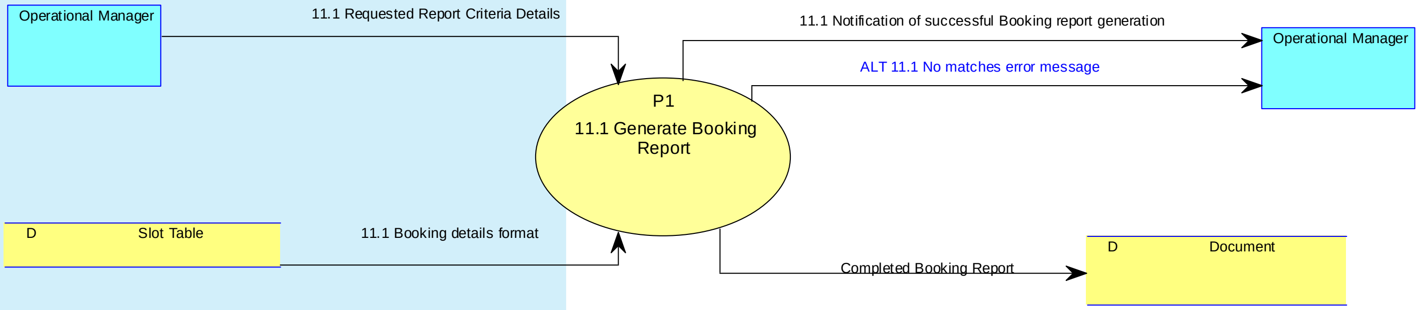

Client
9.4 Request to Cancel a Booking
9.4 Provide Booking Reference
9.2 Search Booking
9.4 Retrieved Booking Details
[ALT] Booking Doesn't Exist
Client
9.4 Provide Reasons for cancellation
P1
9.4 Cancel Booking
9.4 Prompt Booking Reference
Client
9.4 Booking Reference
9.2 Search Booking
9.4 Confirm no payment will be made
[ALT] 50% cost of the trip has to be paid
9.4 Prompt Reasons for Booking Cancellation
Client
9.4 Payment status set to Canceled
[ALT] Payment status set to Outstanding
D
Invoice
9.4 Booking status set to Cancelled
D
Booking_Trip
9.4 Confirmation of Booking Cancellation
Client

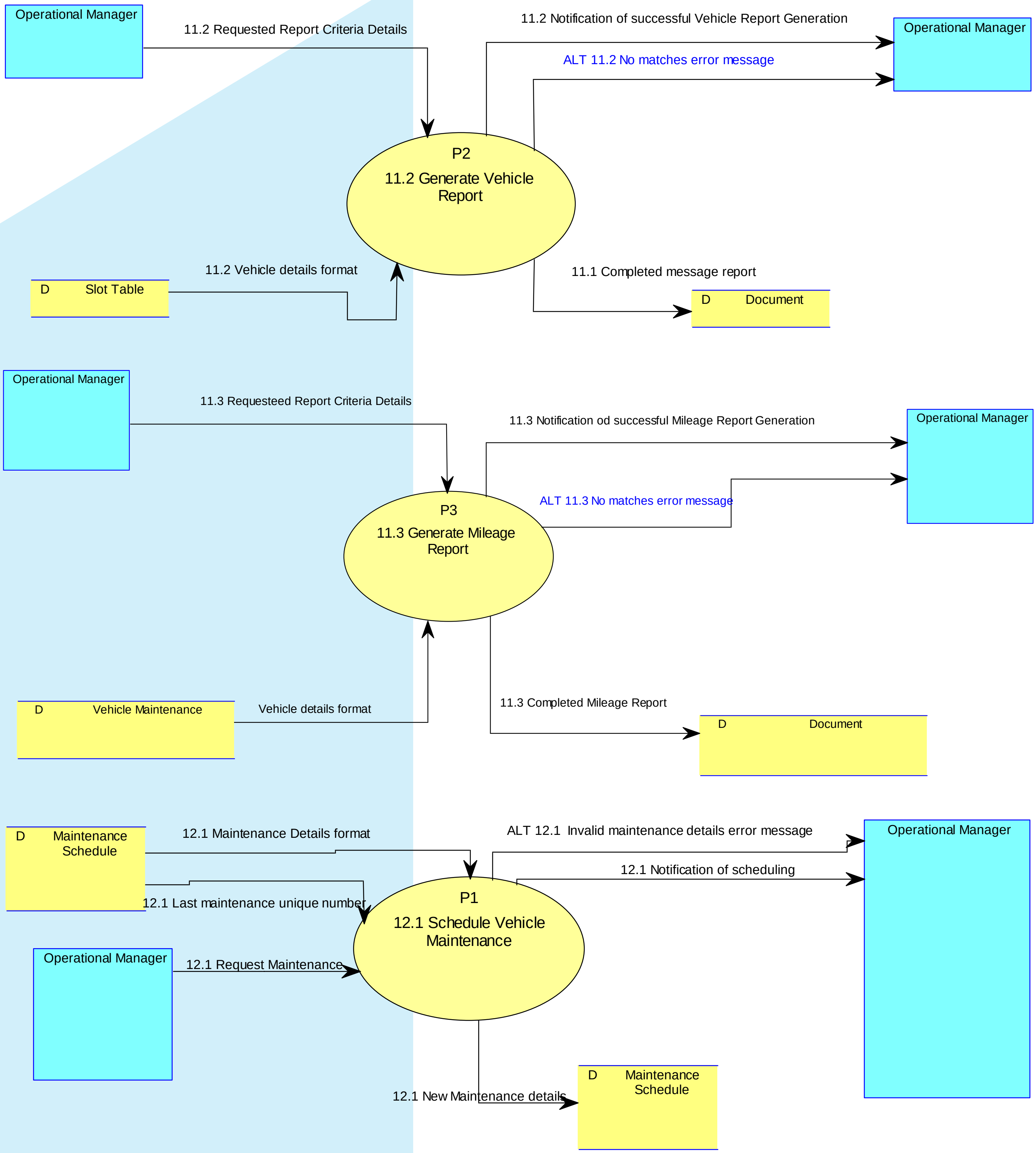

Booking Consultant
9.5 Request to confirm Booking
9.5 Enter Booking Reference
9.2 Search Booking
9.5 Retrieved Booking Details
9.5 Check Available Vehicle
Booking Consultant
check Vehicle Availability
9.5 Retrieved available vehicle
Check Driver Availability
9.5 Retrieved available driver
P1
9.5 Confirm Booking
9.5 Prompt to enter Booking Reference
Booking Consultant
9.5 Booking Reference
9.2 Search Booking
9.5 Date,time,location
check Vehicle Availability
9.5 Date Time location
Check Driver Availability
9.5 Save trip details
D
Slot
9.5 Schedule details
Create Schedule
9.5 Booking Status set to complete
D
Booking_Trip

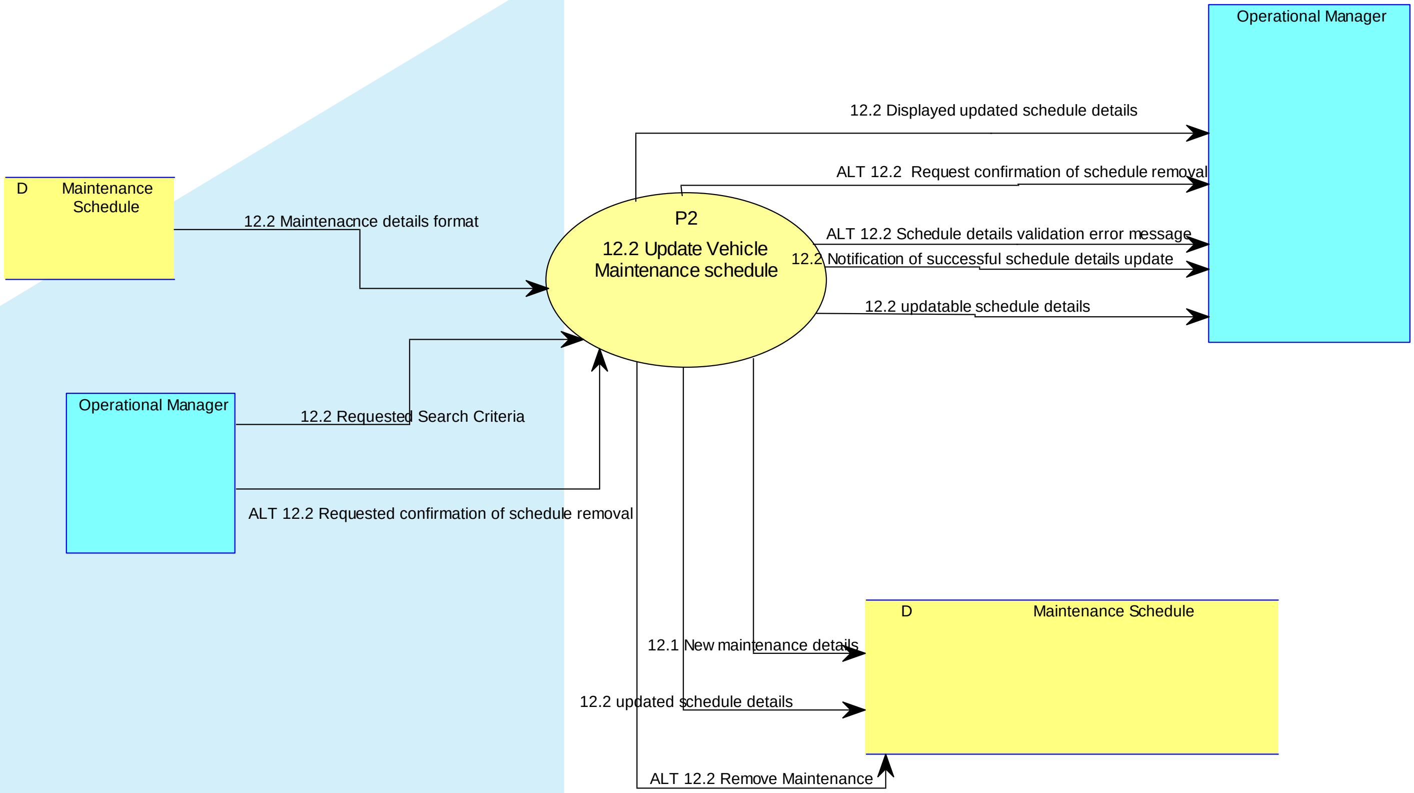

Booking Consultant
10.1 Requested Booking Details
ALT 10.1 Invalid details error message
P1
10.1 Add to schedule
10.1 Notificatioin for scheduling
Booking Consultant
Use case Confirm Booking
10.1 Requested details for booking
D Slot Table
10.1 Last Booking unique number
10.1 New Booking details
D Slot Table

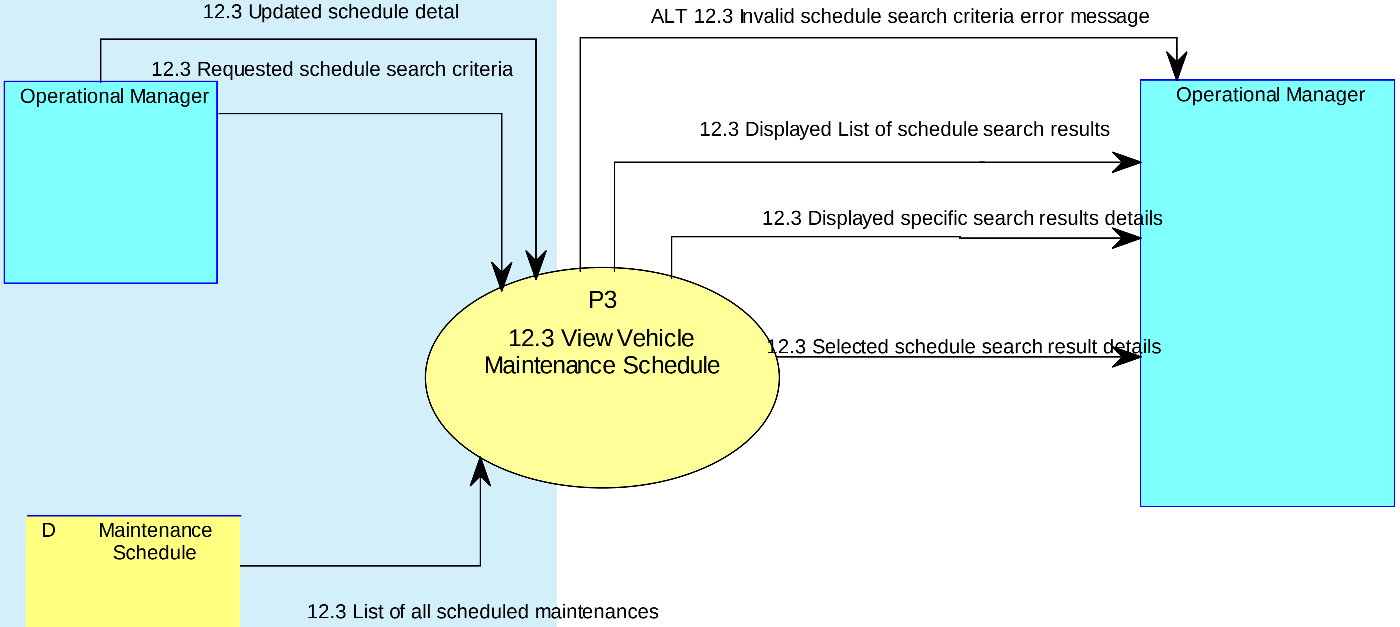

Booking Consultant
10.2 Requested Search Criteria
10.2 Requested Update
P2
10.2 Update Schedule
10.2 Notification of successfule schedule Details Update
10.2 Updatable Schedule Details
10.2 ALT - Schedule Details Validation Error message
10.2 Displayed Updated Schedule Details
Booking Consultant
D Schedule Table
10.2 Schedule Details Format
10.2 Updated Schedule Details
D Schedule Table

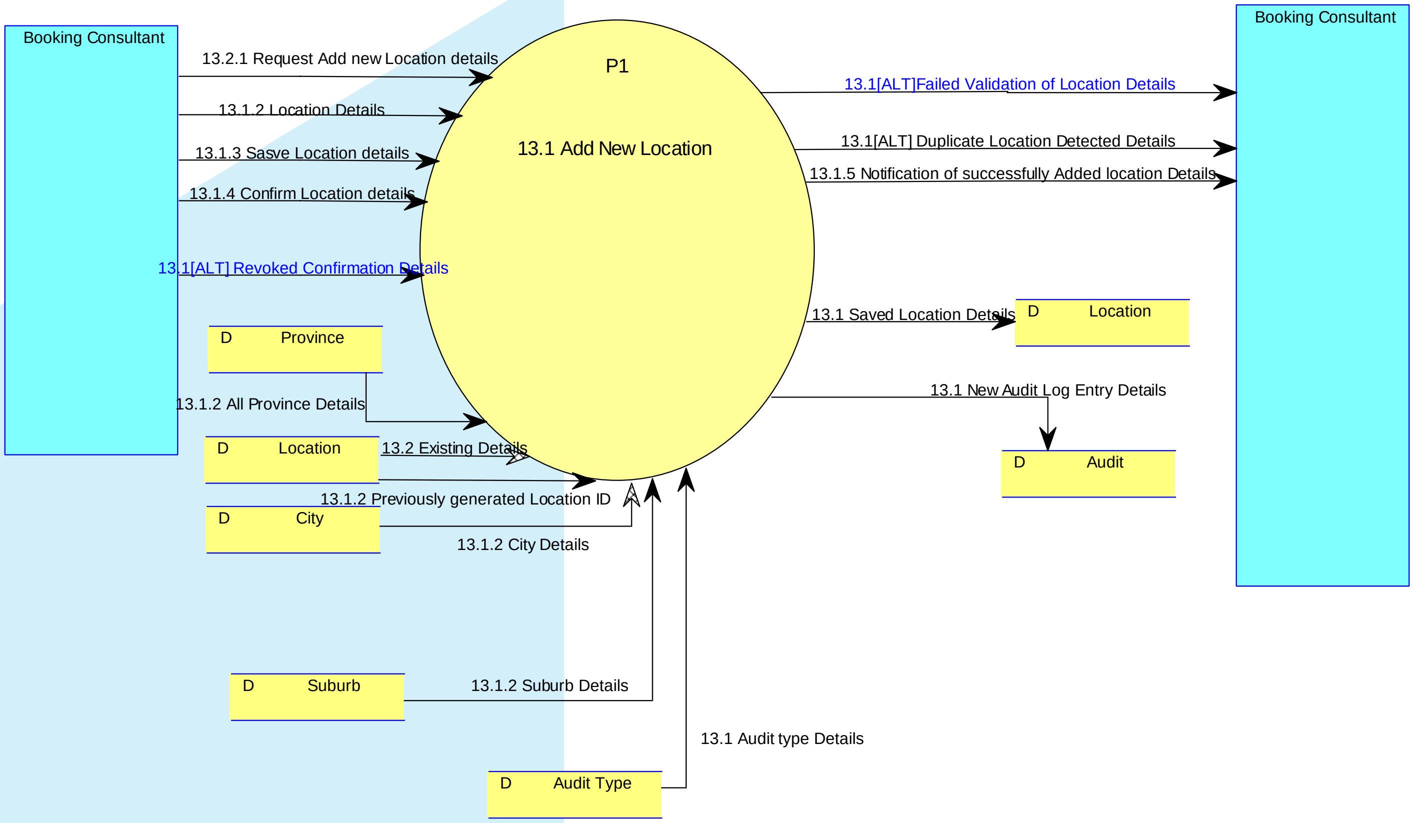

Booking Consultant
10.3 Requested Schedule Search Criteria
10.3 Selected Schedule Search Result Details
P3
10.3 View Schedule
10.3 Displayed Specific Search Result Details
10.3 ALT - Invalid Schedule Search Criteria Error Message
10.3 Displayed List of schedule Search Results
Booking Consultant
D Slot Table
10.3 List of all Scheduled Bookings

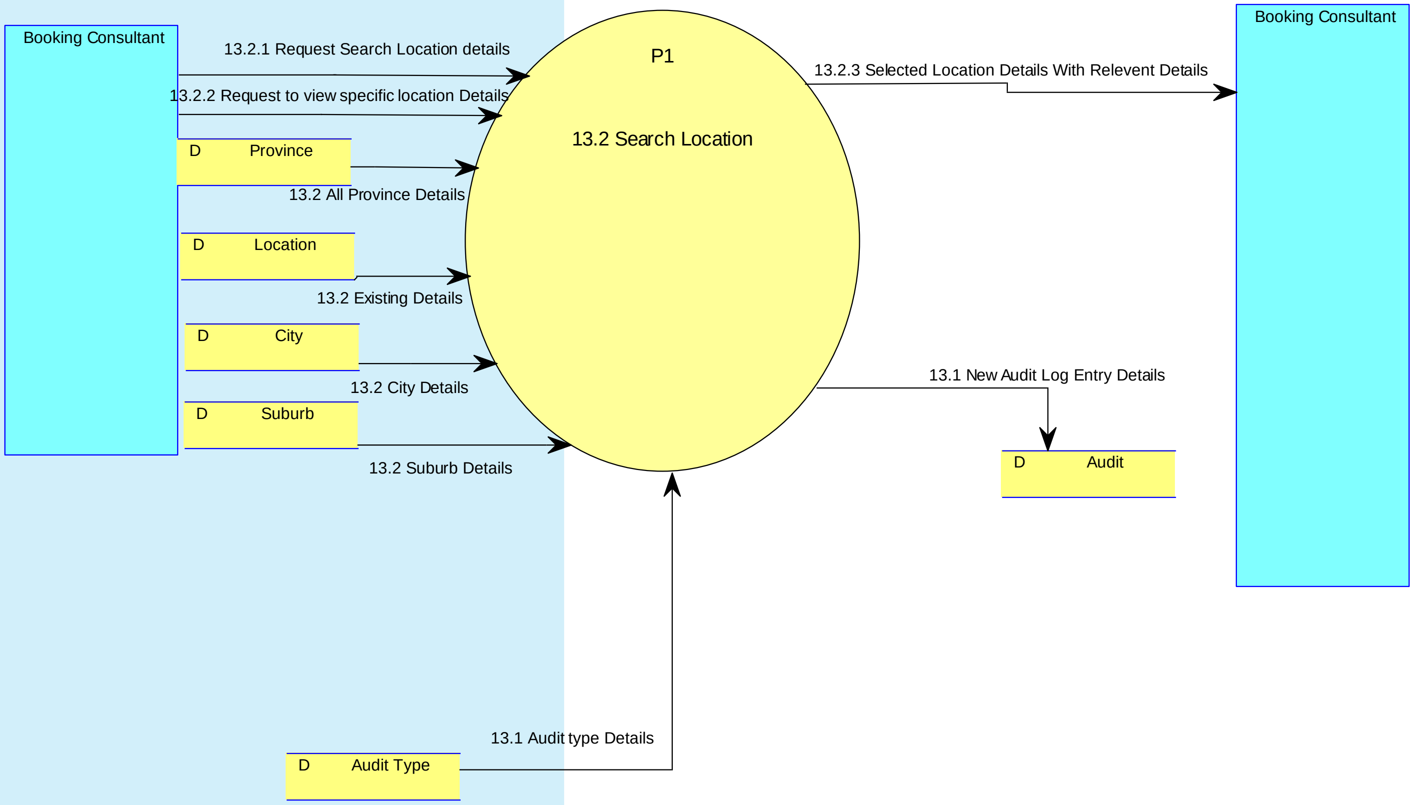

Operational Manager
11.1 Requested Report Criteria Details
11.1 Notification of successful Booking report generation
ALT 11.1 No matches error message
Operational Manager
P1
11.1 Generate Booking Report
D Slot Table
11.1 Booking details format
Completed Booking Report
D Document

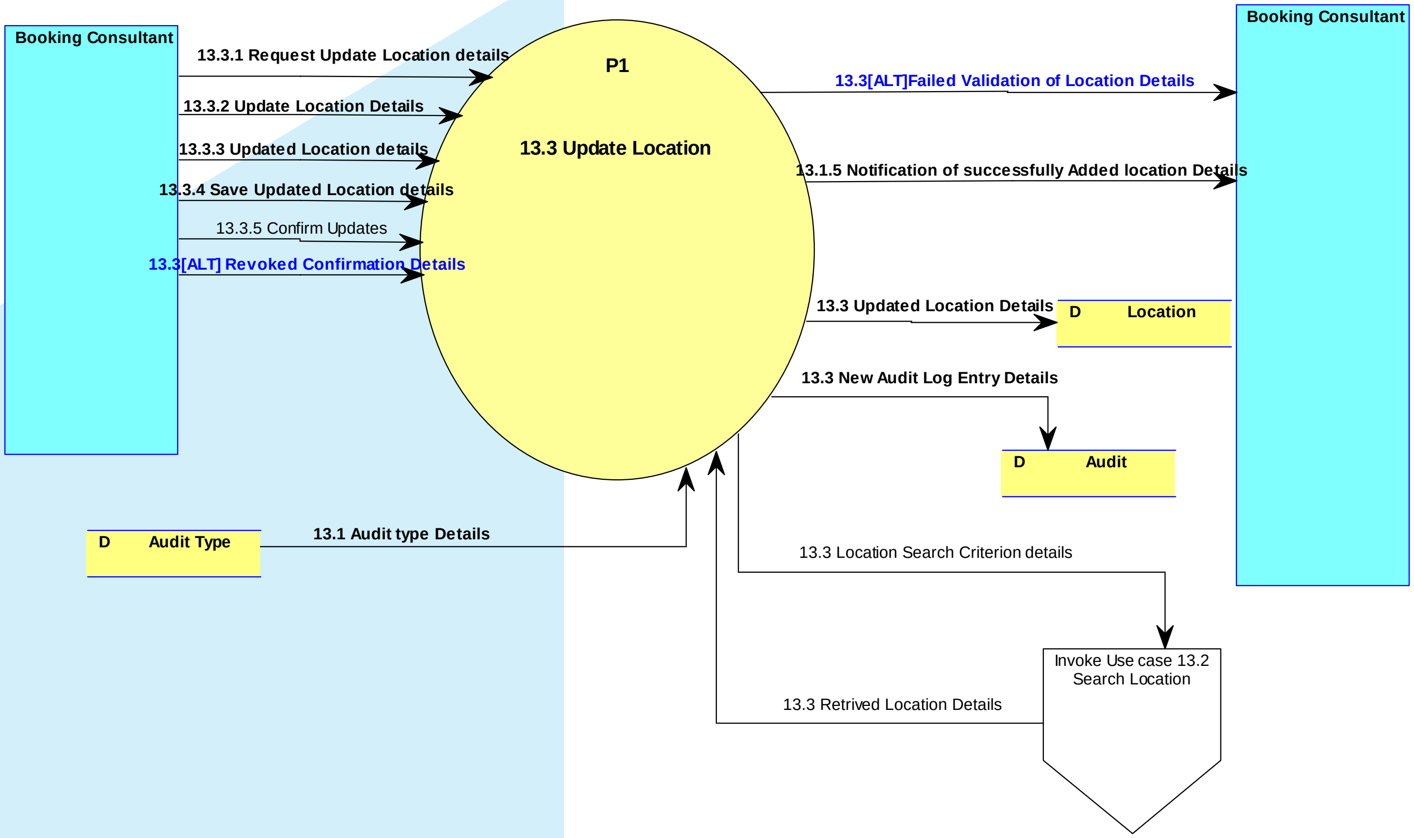

Operational Manager
11.2 Requested Report Criteria Details
11.2 Notification of successful Vehicle Report Generation
Operational Manager
ALT 11.2 No matches error message
P2
11.2 Generate Vehicle Report
11.2 Vehicle details format
D
Slot Table
11.1 Completed message report
D
Document
Operational Manager
11.3 Requesteed Report Criteria Details
11.3 Notification od successful Mileage Report Generation
Operational Manager
ALT 11.3 No matches error message
P3
11.3 Generate Mileage Report
D
Vehicle Maintenance
Vehicle details format
11.3 Completed Mileage Report
D
Document
D
Maintenance Schedule
12.1 Maintenance Details format
ALT 12.1 Invalid maintenance details error message
Operational Manager
12.1 Notification of scheduling
12.1 Last maintenance unique number
P1
12.1 Schedule Vehicle Maintenance
Operational Manager
12.1 Request Maintenance
12.1 New Maintenance details
D
Maintenance Schedule

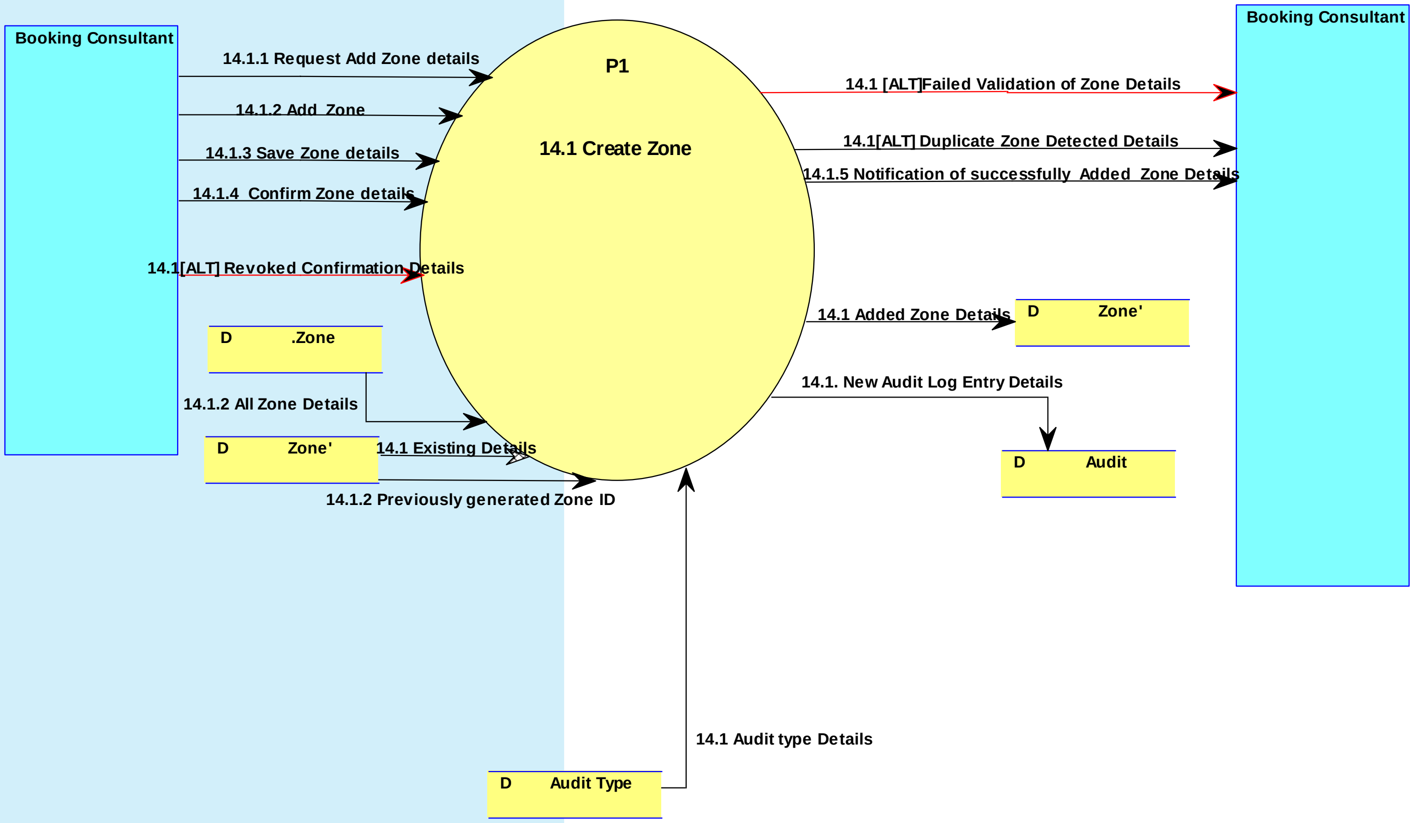

D
Maintenance Schedule
12.2 Maintenacnce details format
Operational Manager
12.2 Requested Search Criteria
ALT 12.2 Requested confirmation of schedule removal
P2
12.2 Update Vehicle Maintenance schedule
12.2 Displayed updated schedule details
ALT 12.2 Request confirmation of schedule removal
ALT 12.2 Schedule details validation error message
12.2 Notification of successful schedule details update
12.2 updatable schedule details
Operational Manager
12.1 New maintenance details
12.2 updated schedule details
ALT 12.2 Remove Maintenance
D
Maintenance Schedule

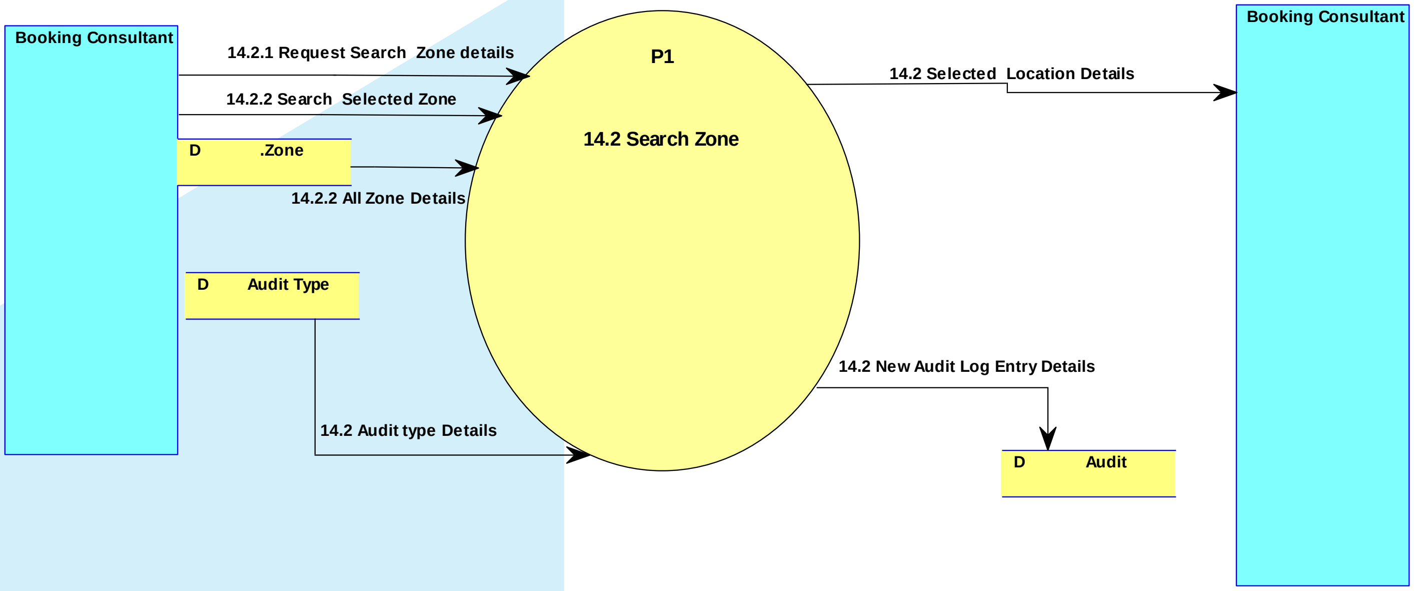

12.3 Updated schedule detal
12.3 Requested schedule search criteria
Operational Manager
ALT 12.3 Invalid schedule search criteria error message
12.3 Displayed List of schedule search results
12.3 Displayed specific search results details
P3
12.3 View Vehicle Maintenance Schedule
12.3 Selected schedule search result details
Operational Manager
D
Maintenance Schedule
12.3 List of all scheduled maintenances

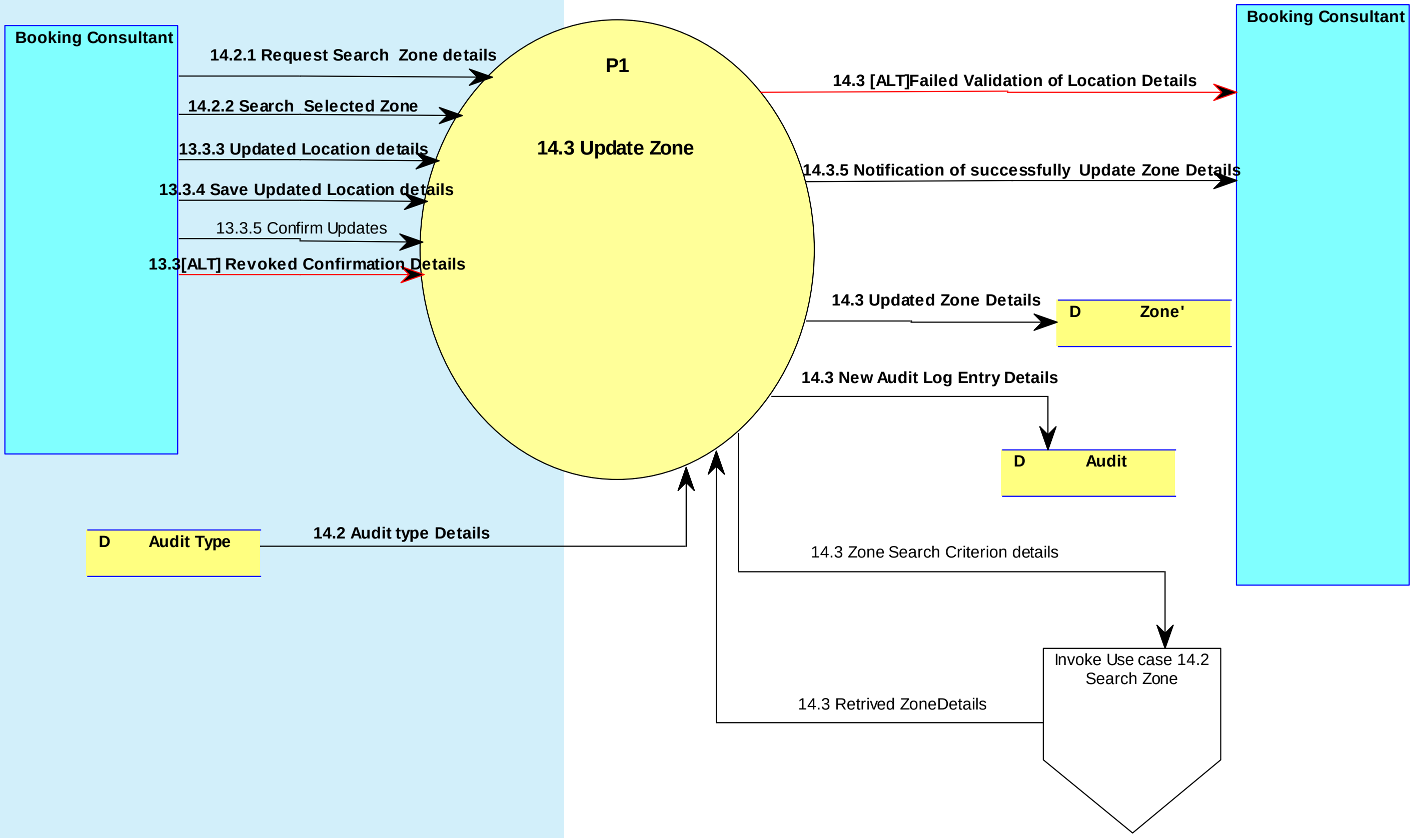

Booking Consultant
13.2.1 Request Add new Location details
13.1.2 Location Details
13.1.3 Sasve Location details
13.1.4 Confirm Location details
13.1[ALT] Revoked Confirmation Details
P1
13.1 Add New Location
13.1[ALT]Failed Validation of Location Details
13.1[ALT] Duplicate Location Detected Details
13.1.5 Notification of successfully Added location Details
Booking Consultant
D Province
13.1.2 All Province Details
D Location
13.2 Existing Details
13.1.2 Previously generated Location ID
D City
13.1.2 City Details
13.1 Saved Location Details
D Location
13.1 New Audit Log Entry Details
D Audit
D Suburb
13.1.2 Suburb Details
13.1 Audit type Details
D Audit Type

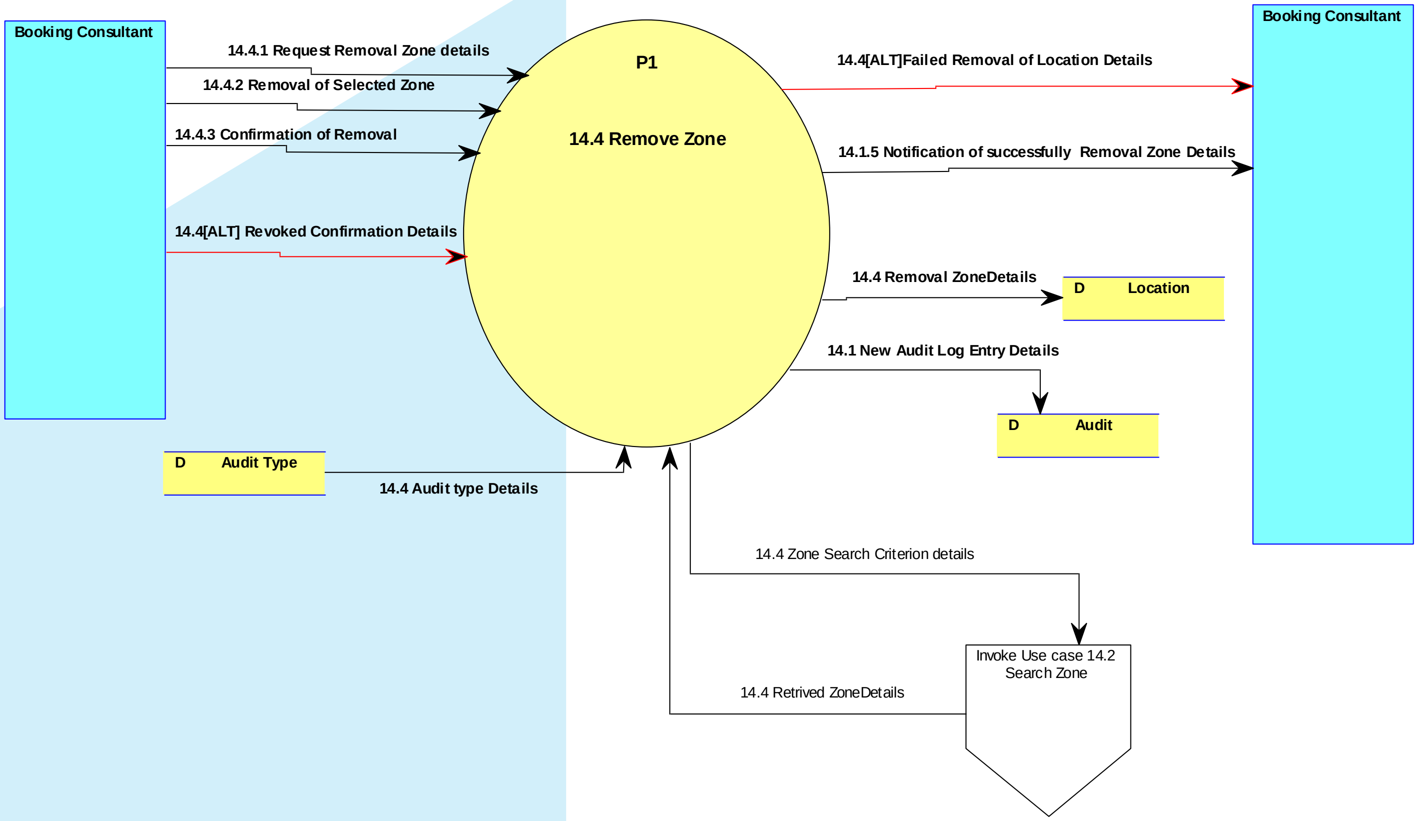

Booking Consultant
13.2.1 Request Search Location details
13.2.2 Request to view specific location Details
D Province
13.2 All Province Details
D Location
13.2 Existing Details
D City
13.2 City Details
D Suburb
13.2 Suburb Details
P1
13.2 Search Location
13.2.3 Selected Location Details With Relevent Details
Booking Consultant
13.1 New Audit Log Entry Details
D Audit
13.1 Audit type Details
D Audit Type

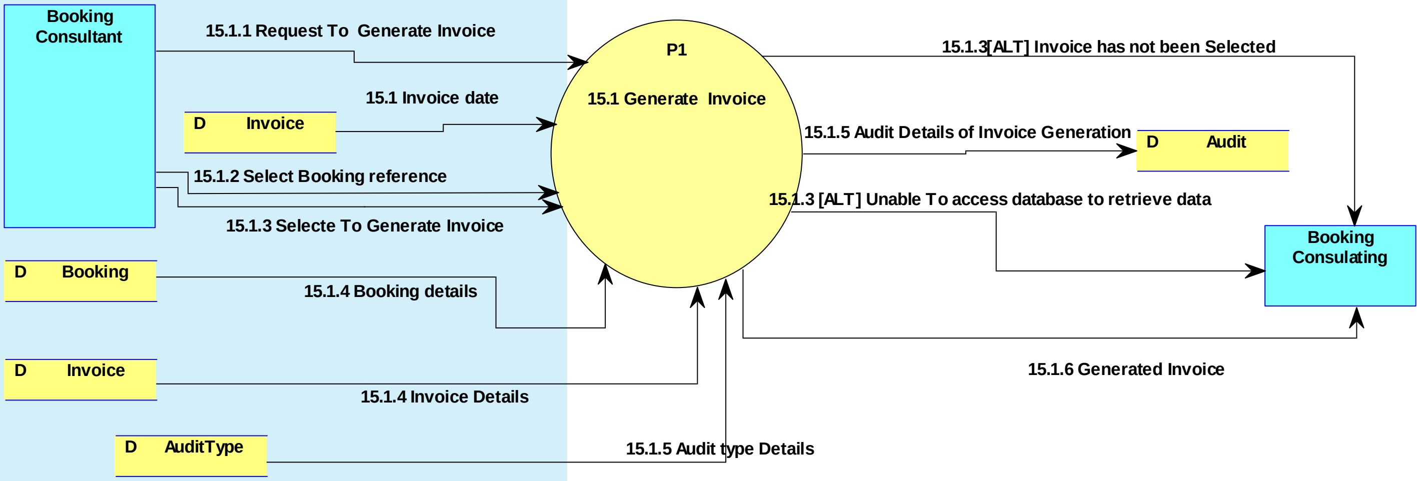

Booking Consultant
13.3.1 Request Update Location details
13.3.2 Update Location Details
13.3.3 Updated Location details
13.3.4 Save Updated Location details
13.3.5 Confirm Updates
13.3[ALT] Revoked Confirmation Details
P1
13.3 Update Location
13.3[ALT]Failed Validation of Location Details
13.1.5 Notification of successfully Added location Details
Booking Consultant
13.3 Updated Location Details
D Location
13.3 New Audit Log Entry Details
D Audit
D Audit Type
13.1 Audit type Details
13.3 Location Search Criterion details
Invoke Use case 13.2 Search Location
13.3 Retrived Location Details


Booking Consultant
14.1.1 Request Add Zone details
14.1.2 Add Zone
14.1.3 Save Zone details
14.1.4 Confirm Zone details
14.1[ALT] Revoked Confirmation Details
D .Zone
14.1.2 All Zone Details
D Zone'
14.1 Existing Details
14.1.2 Previously generated Zone ID
P1
14.1 Create Zone
14.1 [ALT]Failed Validation of Zone Details
14.1[ALT] Duplicate Zone Detected Details
14.1.5 Notification of successfully Added Zone Details
Booking Consultant
14.1 Added Zone Details
D Zone'
14.1. New Audit Log Entry Details
D Audit
14.1 Audit type Details
D Audit Type

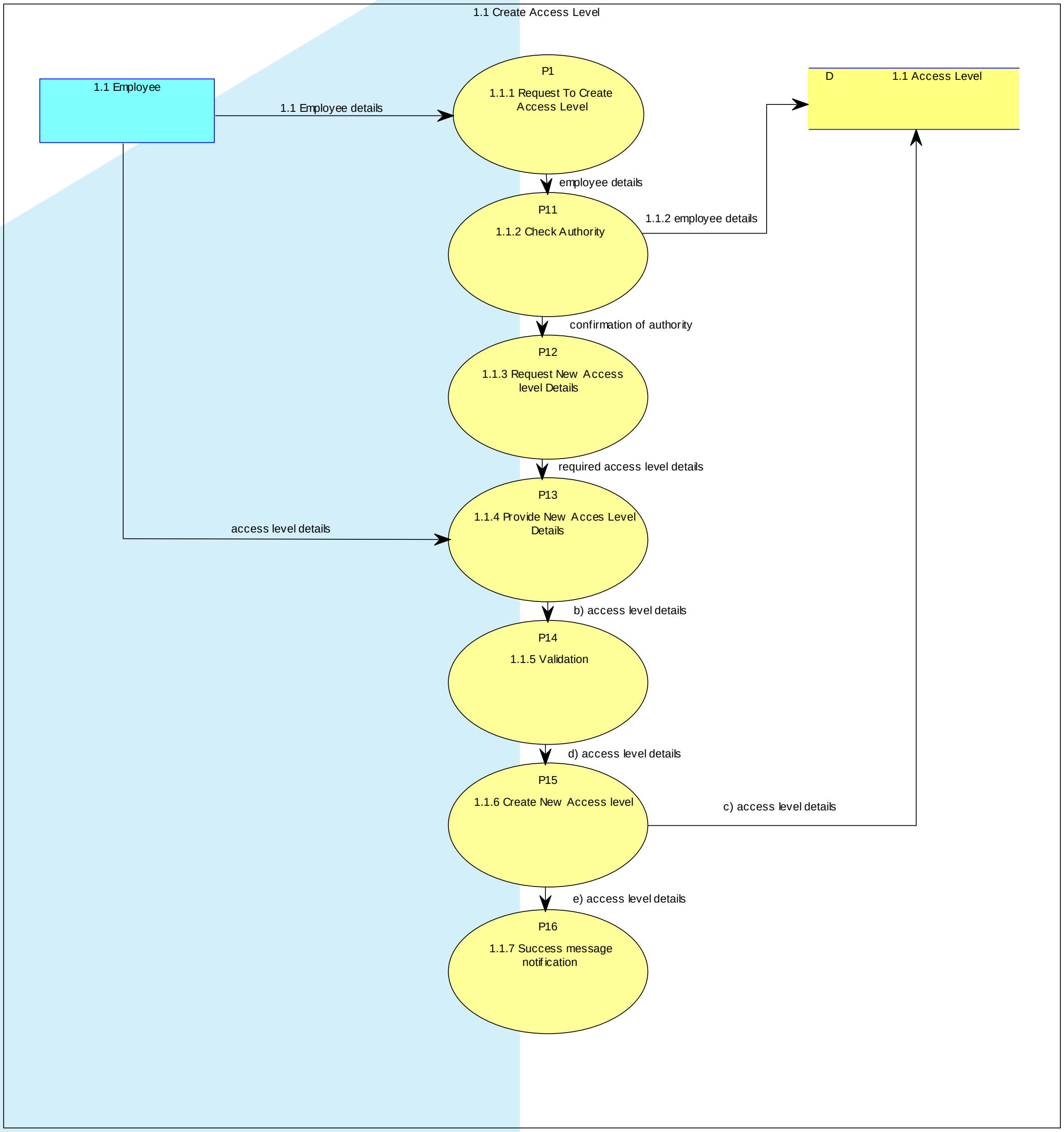

Booking Consultant
14.2.1 Request Search Zone details
14.2.2 Search Selected Zone
D .Zone
14.2.2 All Zone Details
D Audit Type
14.2 Audit type Details
P1
14.2 Search Zone
14.2 Selected Location Details
Booking Consultant
14.2 New Audit Log Entry Details
D Audit

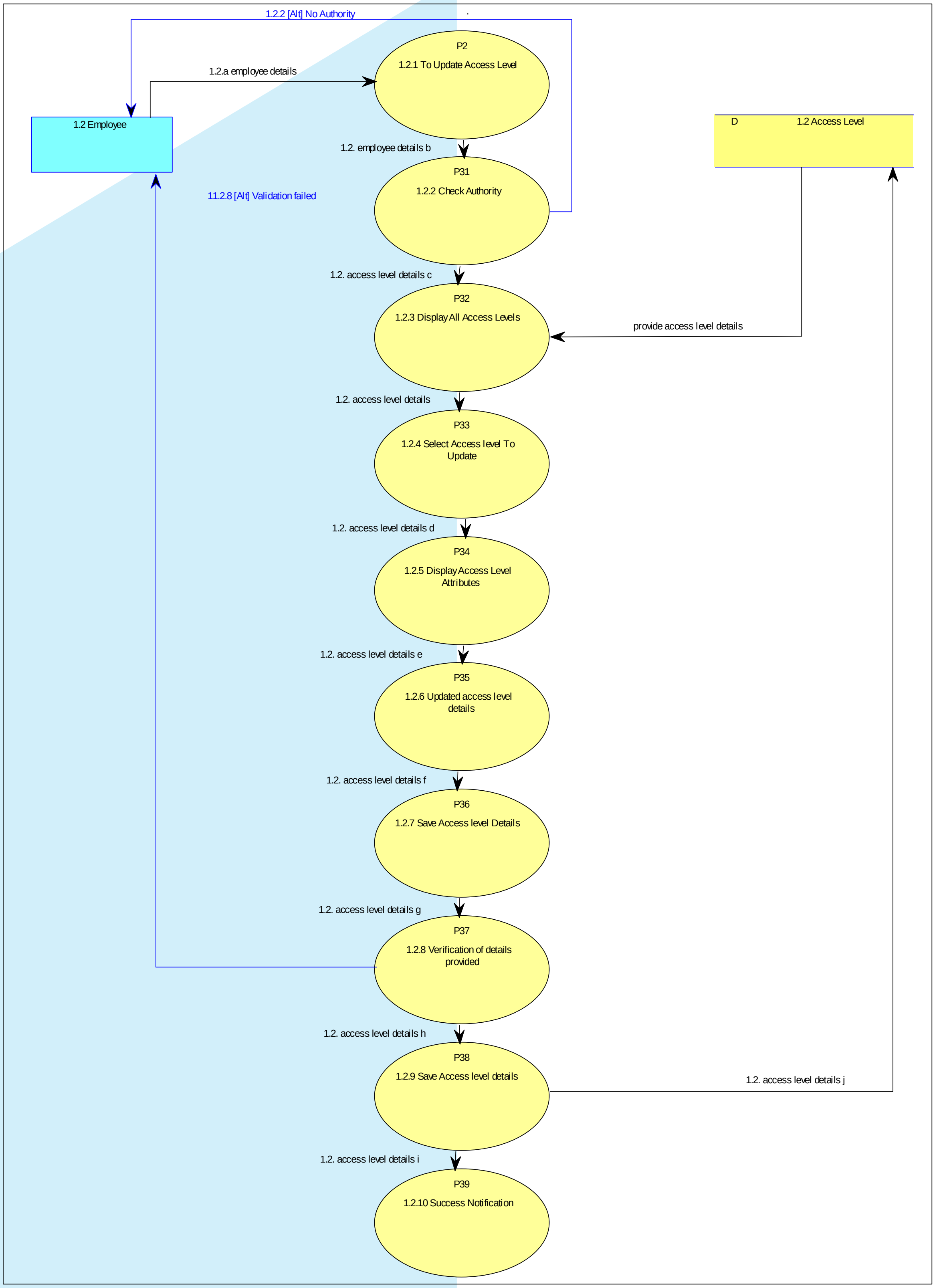

Booking Consultant
14.2.1 Request Search Zone details
14.2.2 Search Selected Zone
13.3.3 Updated Location details
13.3.4 Save Updated Location details
13.3.5 Confirm Updates
13.3[ALT] Revoked Confirmation Details
P1
14.3 Update Zone
14.3 [ALT]Failed Validation of Location Details
14.3.5 Notification of successfully Update Zone Details
Booking Consultant
14.3 Updated Zone Details
D Zone'
14.3 New Audit Log Entry Details
D Audit
D Audit Type
14.2 Audit type Details
14.3 Zone Search Criterion details
Invoke Use case 14.2 Search Zone
14.3 Retrived ZoneDetails

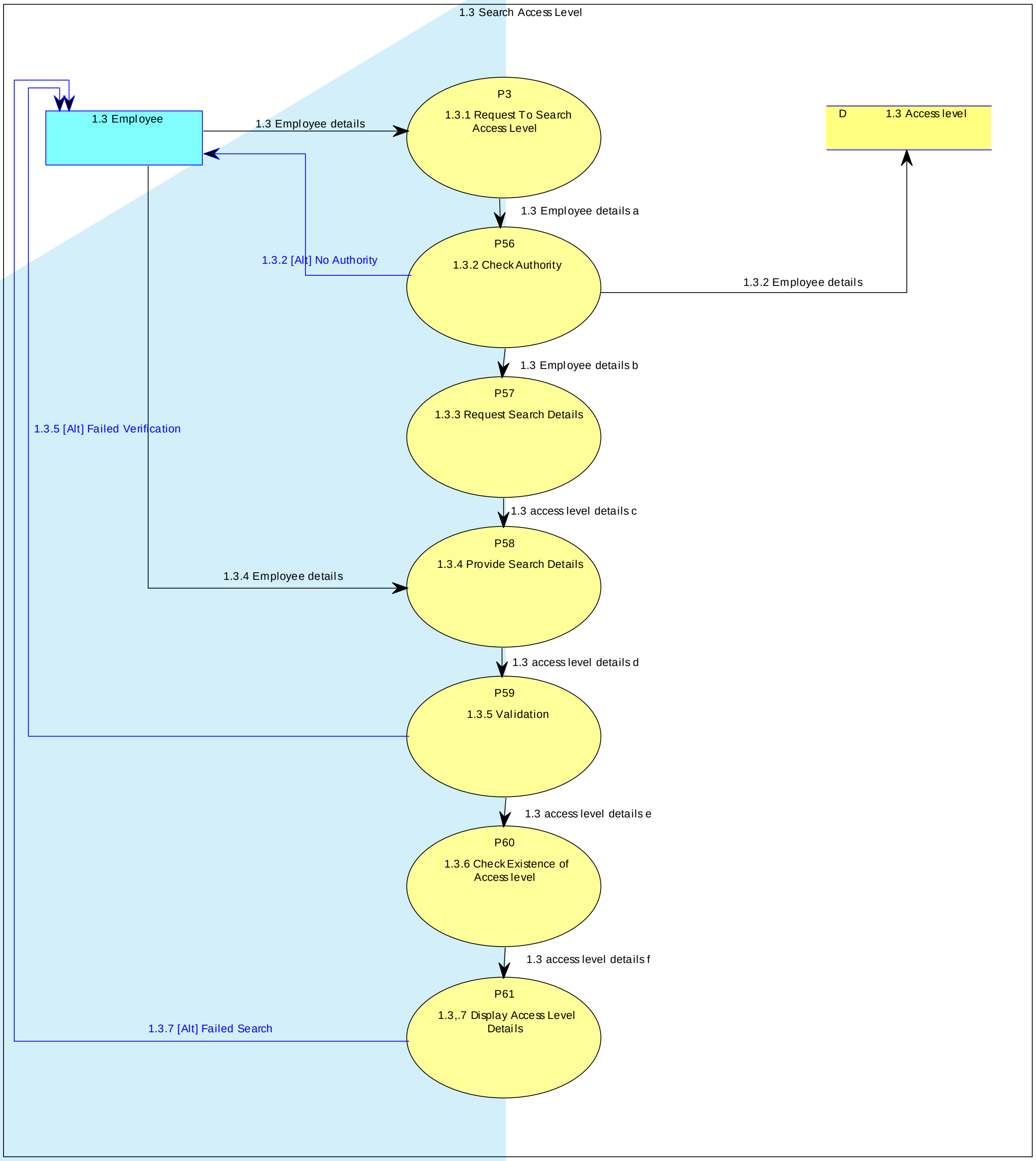

Booking Consultant
14.4.1 Request Removal Zone details
14.4.2 Removal of Selected Zone
14.4.3 Confirmation of Removal
14.4[ALT] Revoked Confirmation Details
P1
14.4 Remove Zone
14.4[ALT]Failed Removal of Location Details
Booking Consultant
14.1.5 Notification of successfully Removal Zone Details
14.4 Removal ZoneDetails
D Location
14.1 New Audit Log Entry Details
D Audit
D Audit Type
14.4 Audit type Details
14.4 Zone Search Criterion details
Invoke Use case 14.2 Search Zone
14.4 Retrived ZoneDetails

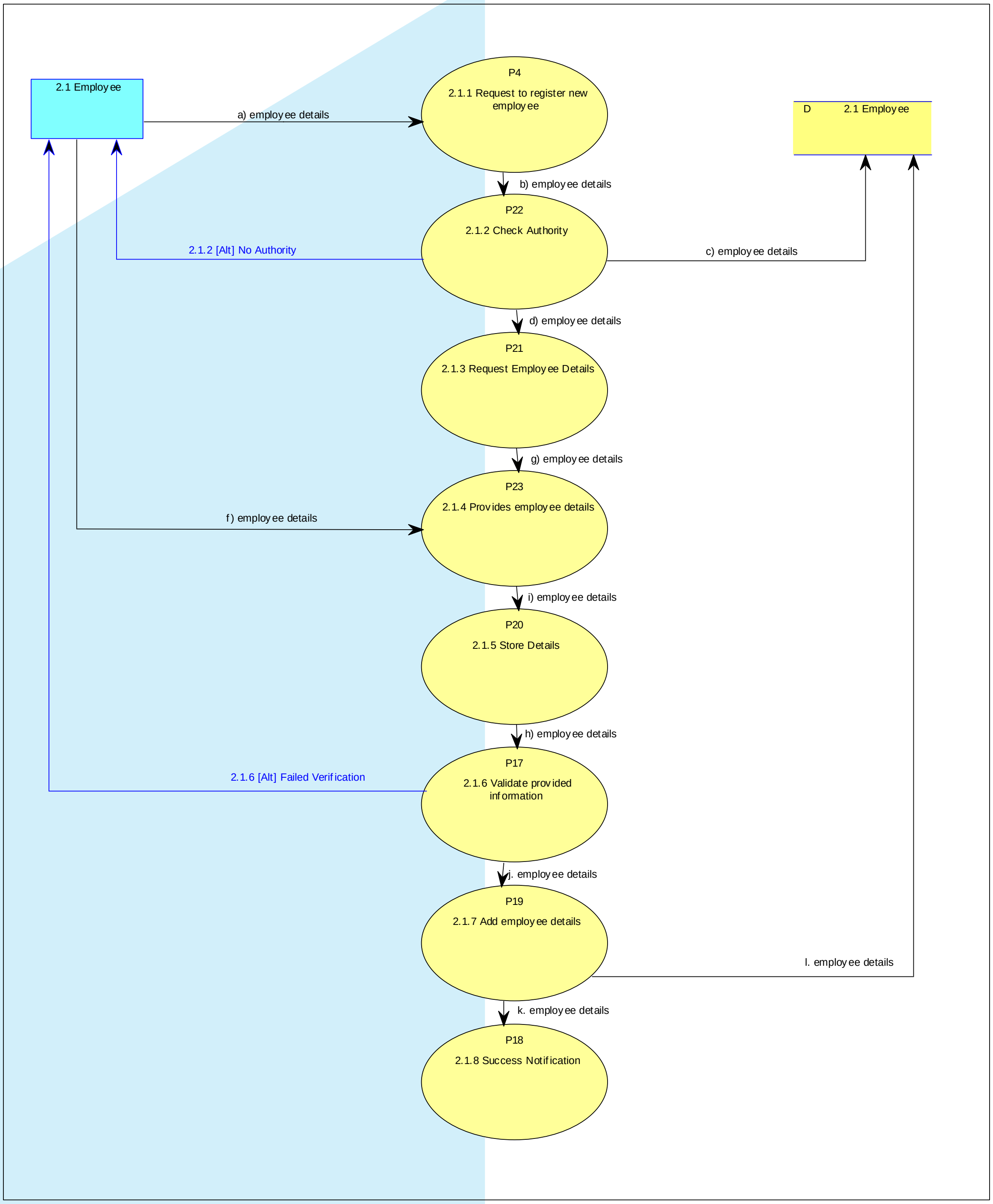

Booking Consultant
15.1.1 Request To Generate Invoice
P1
15.1 Generate Invoice
15.1.3[ALT] Invoice has not been Selected
D Invoice
15.1 Invoice date
15.1.5 Audit Details of Invoice Generation
D Audit
15.1.2 Select Booking reference
15.1.3 [ALT] Unable To access database to retrieve data
15.1.3 Selecte To Generate Invoice
Booking Consulating
D Booking
15.1.4 Booking details
D Invoice
15.1.4 Invoice Details
15.1.6 Generated Invoice
D AuditType
15.1.5 Audit type Details

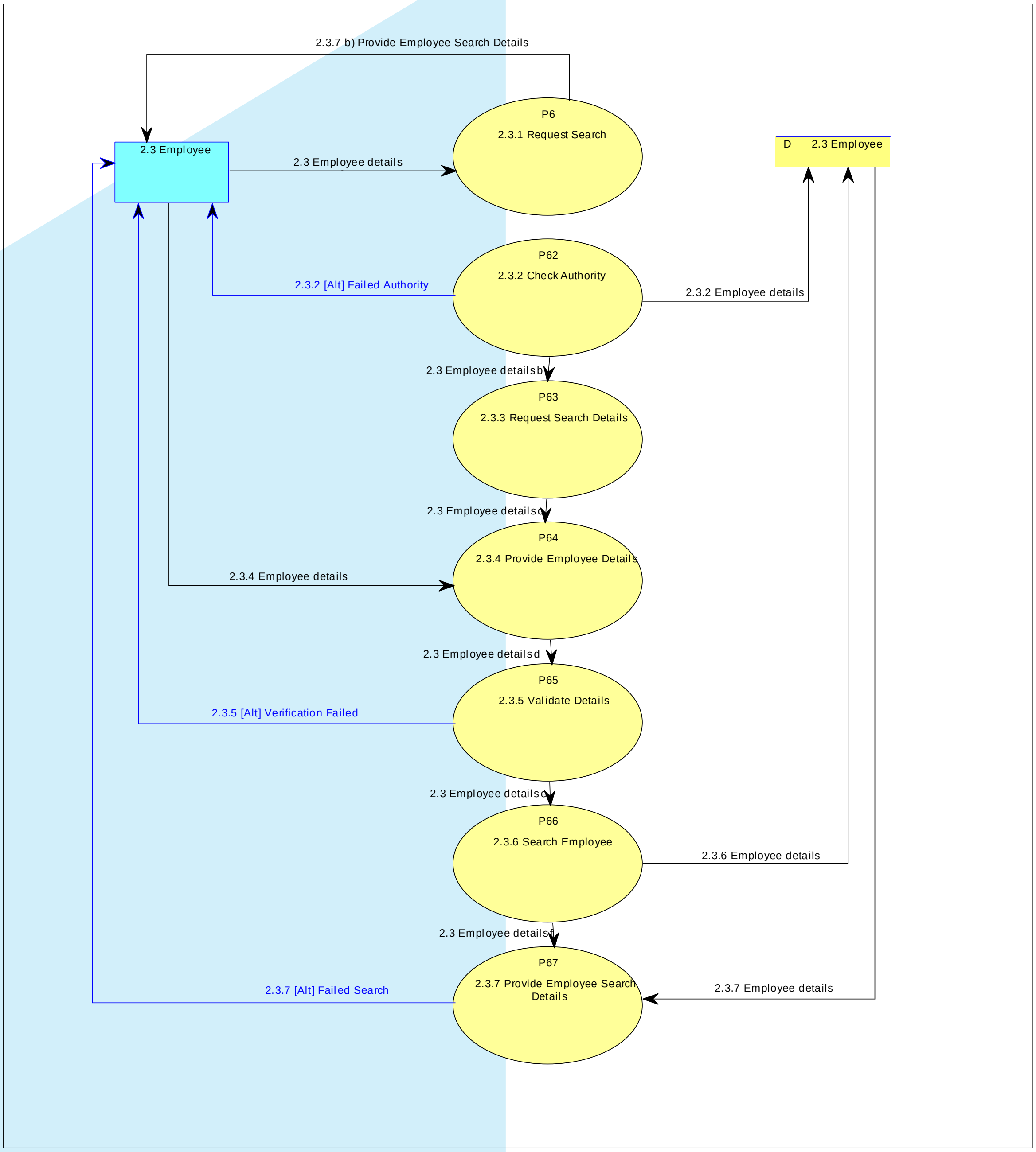
Primitives

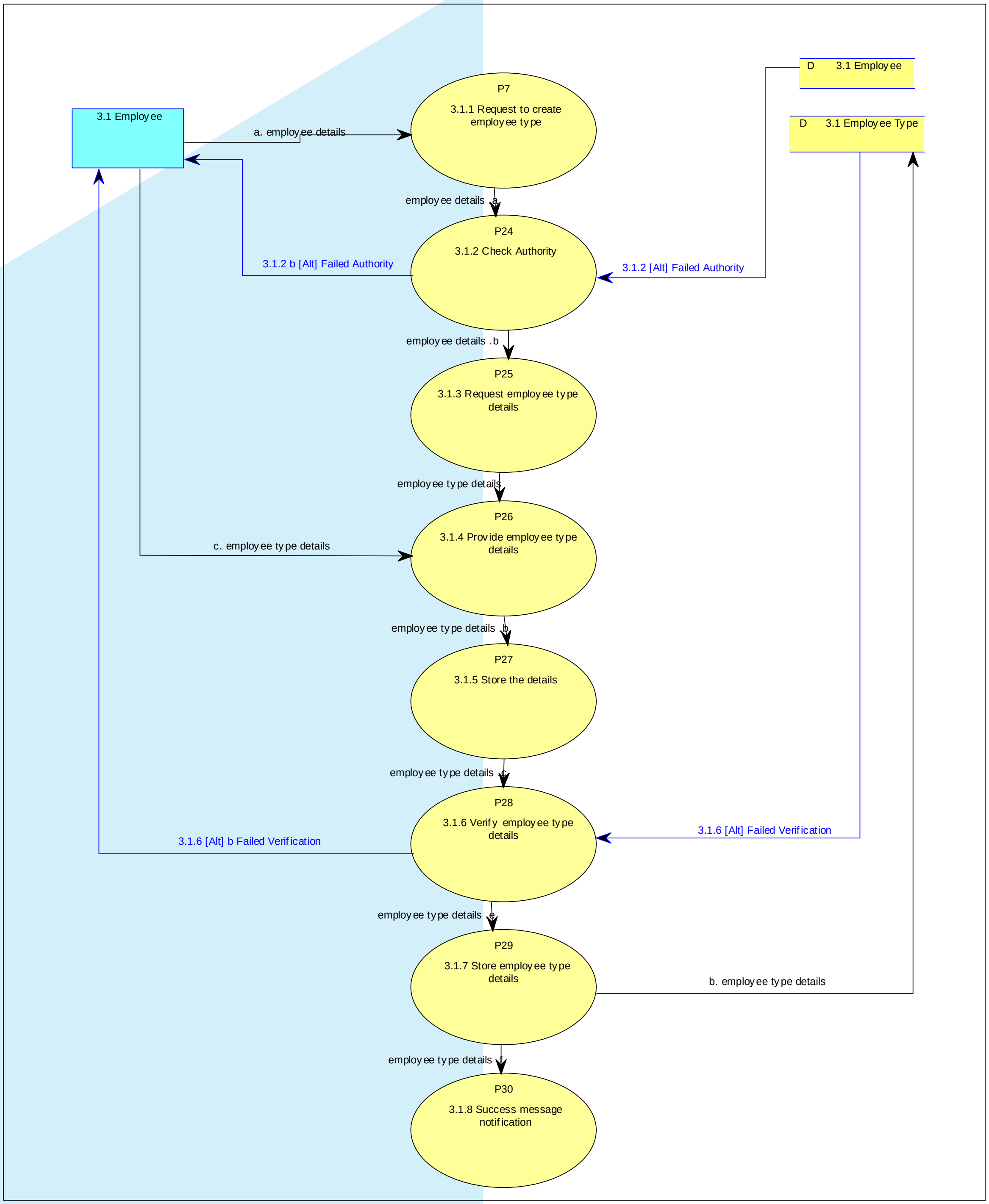

1.1 Create Access Level
1.1 Employee
1.1 Employee details
P1
1.1.1 Request To Create Access Level
D
1.1 Access Level
employee details
P11
1.1.2 Check Authority
1.1.2 employee details
confirmation of authority
P12
1.1.3 Request New Access level Details
required access level details
access level details
P13
1.1.4 Provide New Acces Level Details
b) access level details
P14
1.1.5 Validation
d) access level details
P15
1.1.6 Create New Access level
c) access level details
e) access level details
P16
1.1.7 Success message notification

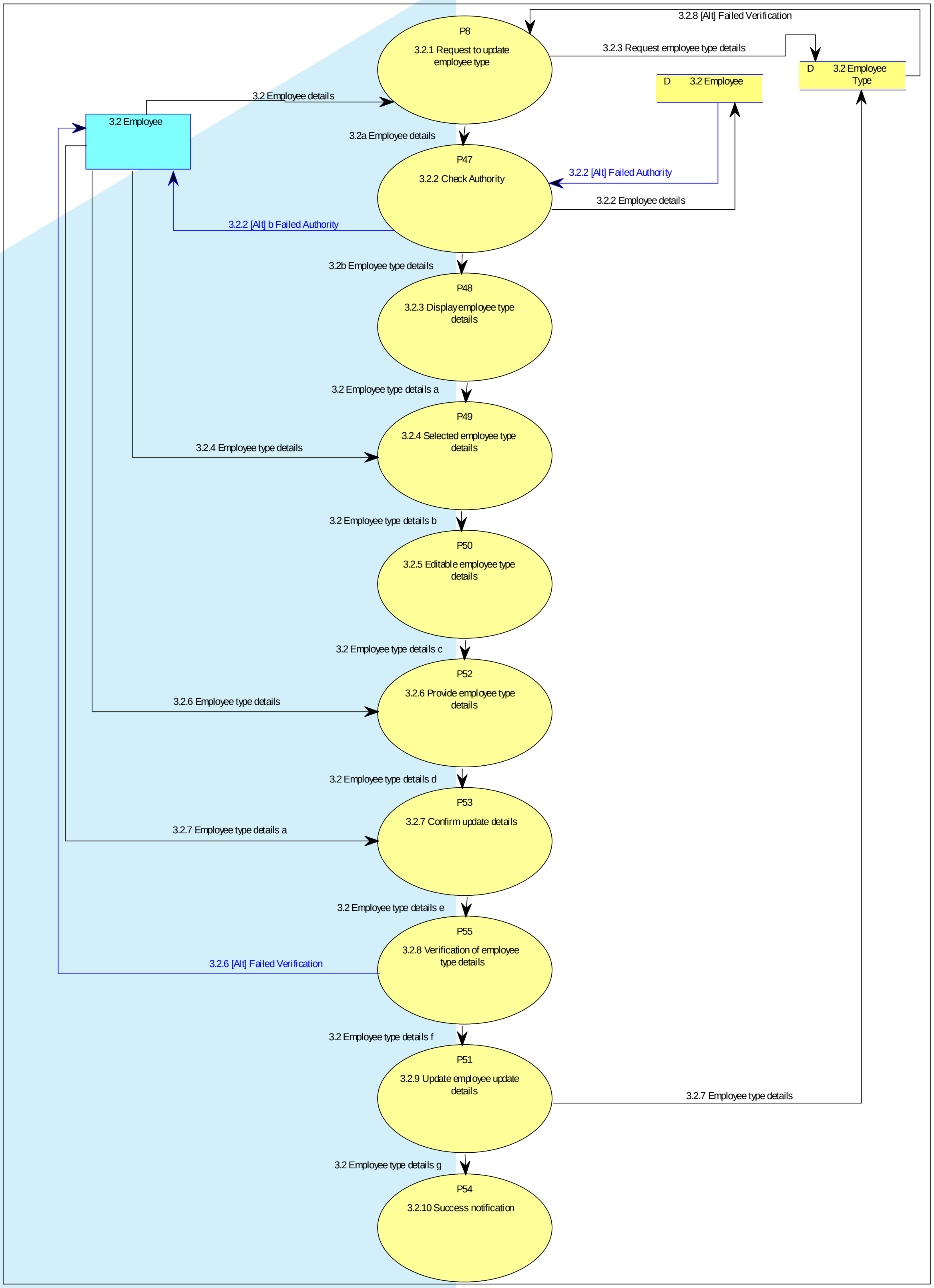

1.2.2 [Alt] No Authority
1.2.a employee details
P2
1.2.1 To Update Access Level
1.2 Employee
D
1.2 Access Level
1.2. employee details b
P31
1.2.2 Check Authority
11.2.8 [Alt] Validation failed
1.2. access level details c
P32
1.2.3 Display All Access Levels
provide access level details
1.2. access level details
P33
1.2.4 Select Access level To Update
1.2. access level details d
P34
1.2.5 Display Access Level Attributes
1.2. access level details e
P35
1.2.6 Updated access level details
1.2. access level details f
P36
1.2.7 Save Access level Details
1.2. access level details g
P37
1.2.8 Verification of details provided
1.2. access level details h
P38
1.2.9 Save Access level details
1.2. access level details j
1.2. access level details i
P39
1.2.10 Success Notification

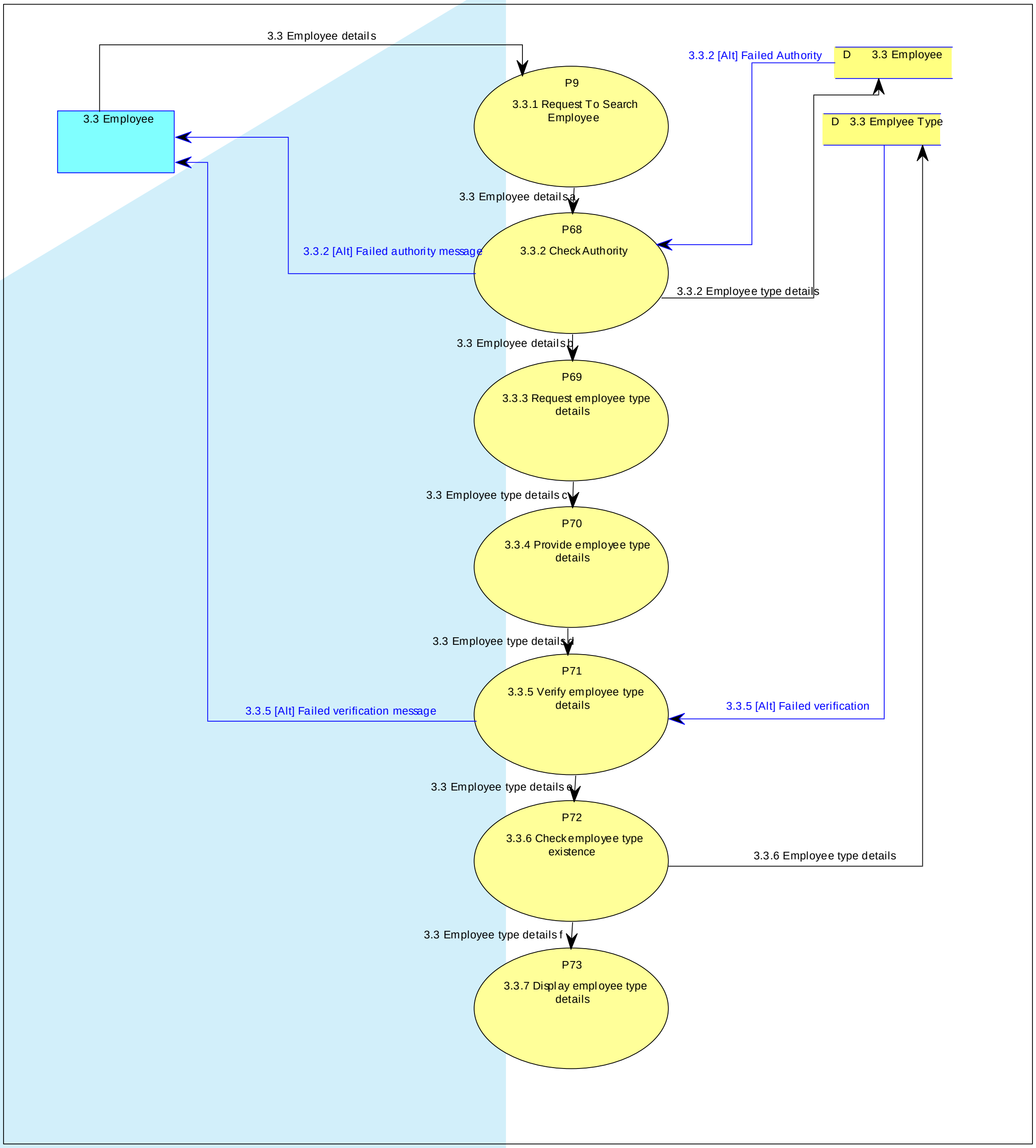

1.3 Search Access Level
1.3 Employee
1.3 Employee details
P3
1.3.1 Request To Search Access Level
D
1.3 Access level
1.3 Employee details a
P56
1.3.2 Check Authority
1.3.2 [Alt] No Authority
1.3.2 Employee details
1.3 Employee details b
P57
1.3.3 Request Search Details
1.3.5 [Alt] Failed Verification
1.3 access level details c
P58
1.3.4 Provide Search Details
1.3.4 Employee details
1.3 access level details d
P59
1.3.5 Validation
1.3 access level details e
P60
1.3.6 Check Existence of Access level
1.3 access level details f
P61
1.3,.7 Display Access Level Details
1.3.7 [Alt] Failed Search

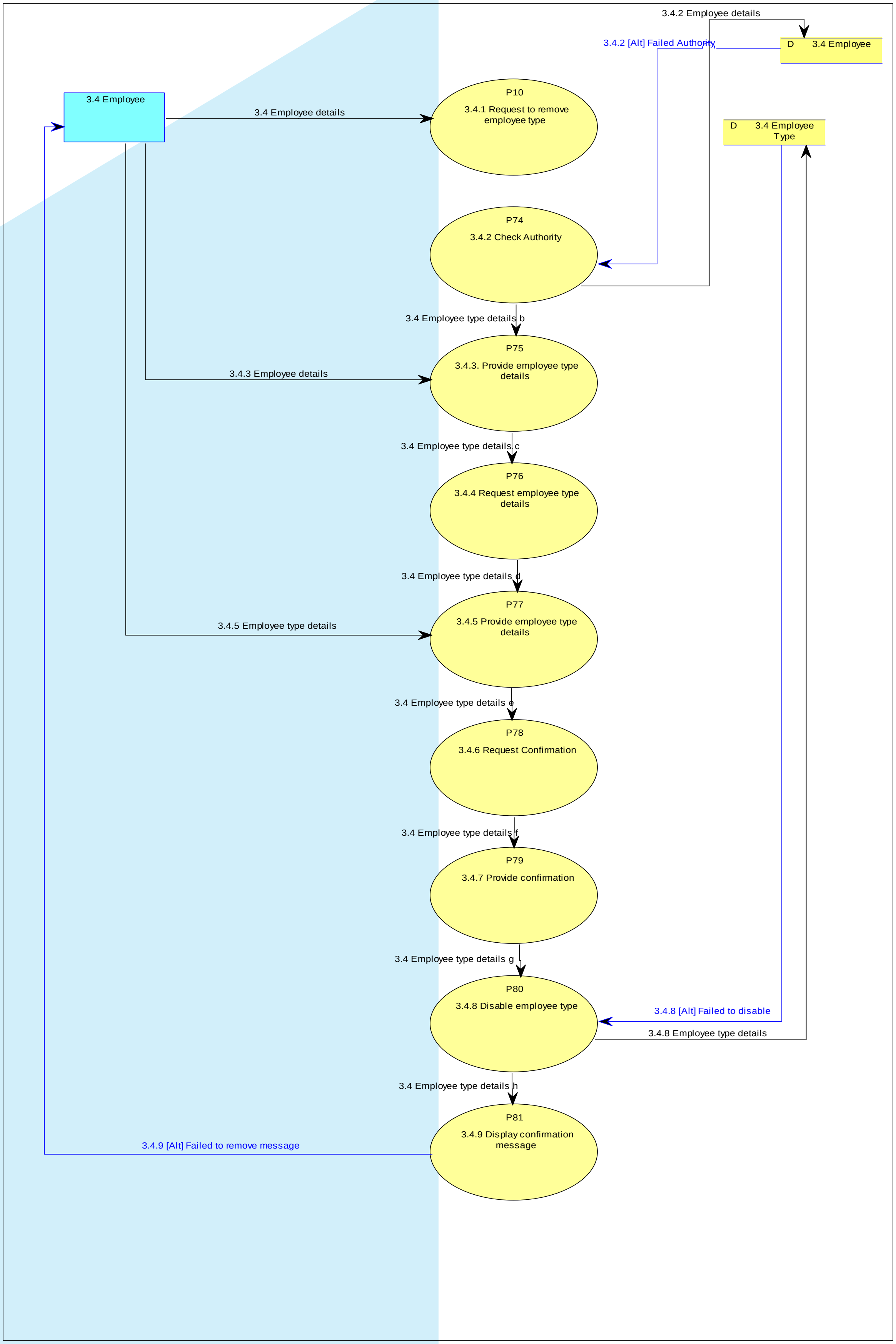

2.1 Employee
a) employee details
P4
2.1.1 Request to register new employee
D
2.1 Employee
b) employee details
P22
2.1.2 Check Authority
2.1.2 [Alt] No Authority
c) employee details
d) employee details
P21
2.1.3 Request Employee Details
g) employee details
P23
2.1.4 Provides employee details
f) employee details
i) employee details
P20
2.1.5 Store Details
h) employee details
P17
2.1.6 Validate provided information
2.1.6 [Alt] Failed Verification
j. employee details
P19
2.1.7 Add employee details
l. employee details
k. employee details
P18
2.1.8 Success Notification

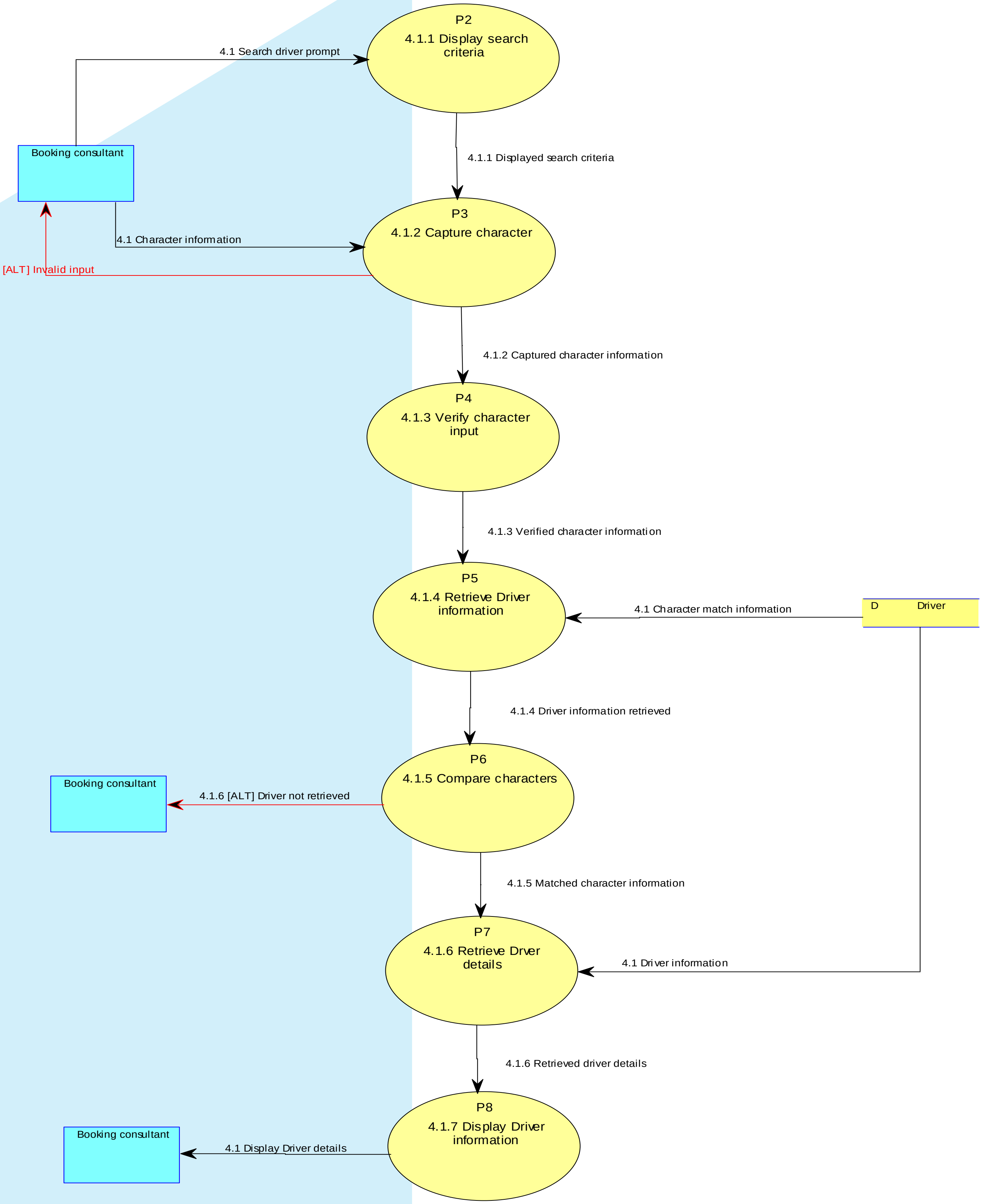

2.3.7 b) Provide Employee Search Details
2.3 Employee
2.3 Employee details
P6
2.3.1 Request Search
D
2.3 Employee
P62
2.3.2 Check Authority
2.3.2 [Alt] Failed Authority
2.3.2 Employee details
2.3 Employee detailsb
P63
2.3.3 Request Search Details
2.3 Employee detailsc
P64
2.3.4 Provide Employee Details
2.3.4 Employee details
2.3 Employee detailsd
P65
2.3.5 Validate Details
2.3.5 [Alt] Verification Failed
2.3 Employee detailse
P66
2.3.6 Search Employee
2.3.6 Employee details
2.3 Employee detailsf
P67
2.3.7 Provide Employee Search Details
2.3.7 [Alt] Failed Search
2.3.7 Employee details

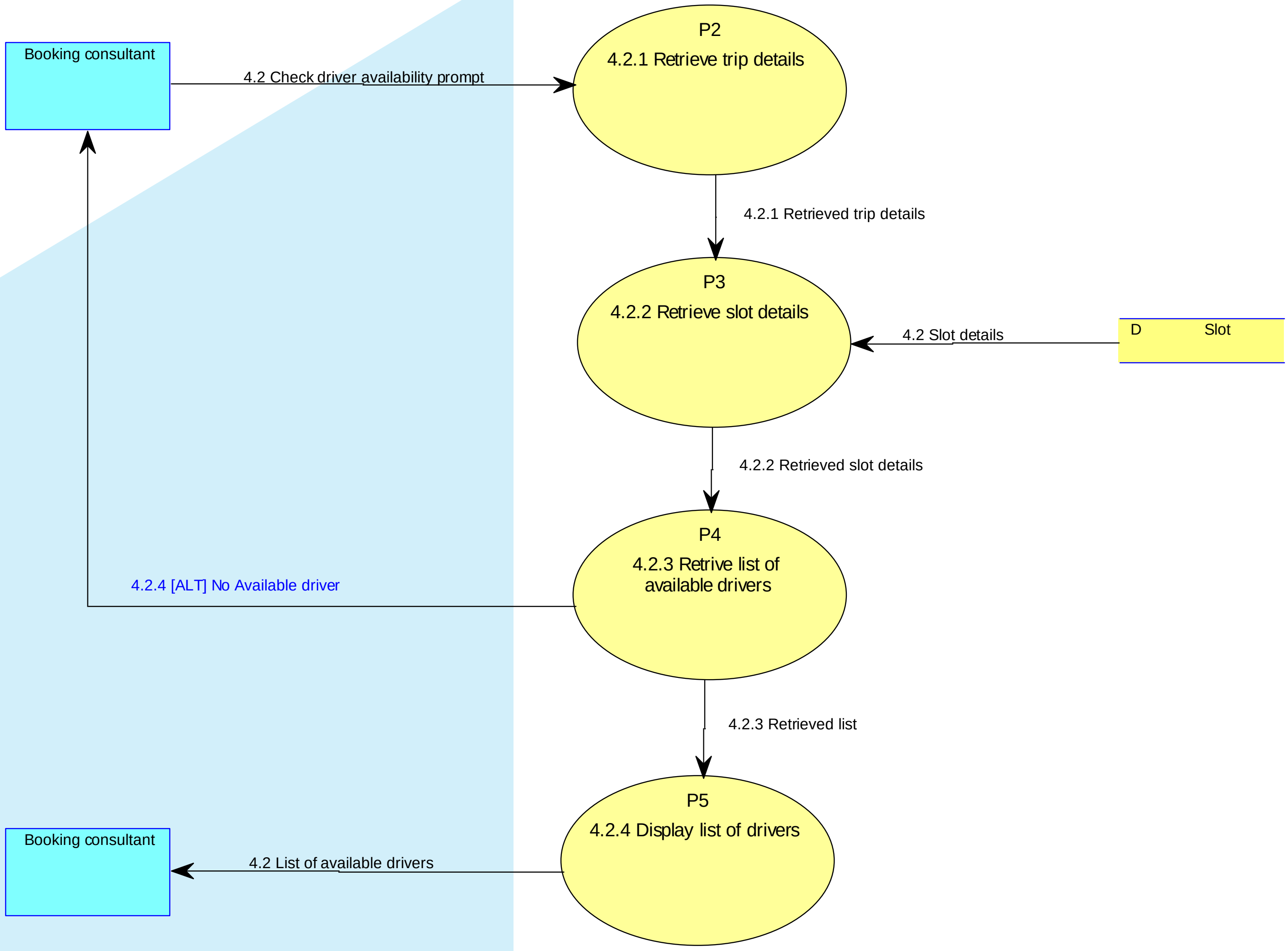

3.1 Employee
a. employee details
P7
3.1.1 Request to create employee type
D 3.1 Employee
D 3.1 Employee Type
employee details .a
P24
3.1.2 Check Authority
3.1.2 b [Alt] Failed Authority
3.1.2 [Alt] Failed Authority
employee details .b
P25
3.1.3 Request employee type details
employee type details
P26
3.1.4 Provide employee type details
c. employee type details
employee type details .b
P27
3.1.5 Store the details
employee type details .c
P28
3.1.6 Verify employee type details
3.1.6 [Alt] b Failed Verification
3.1.6 [Alt] Failed Verification
employee type details .e
P29
3.1.7 Store employee type details
b. employee type details
employee type details .f
P30
3.1.8 Success message notification

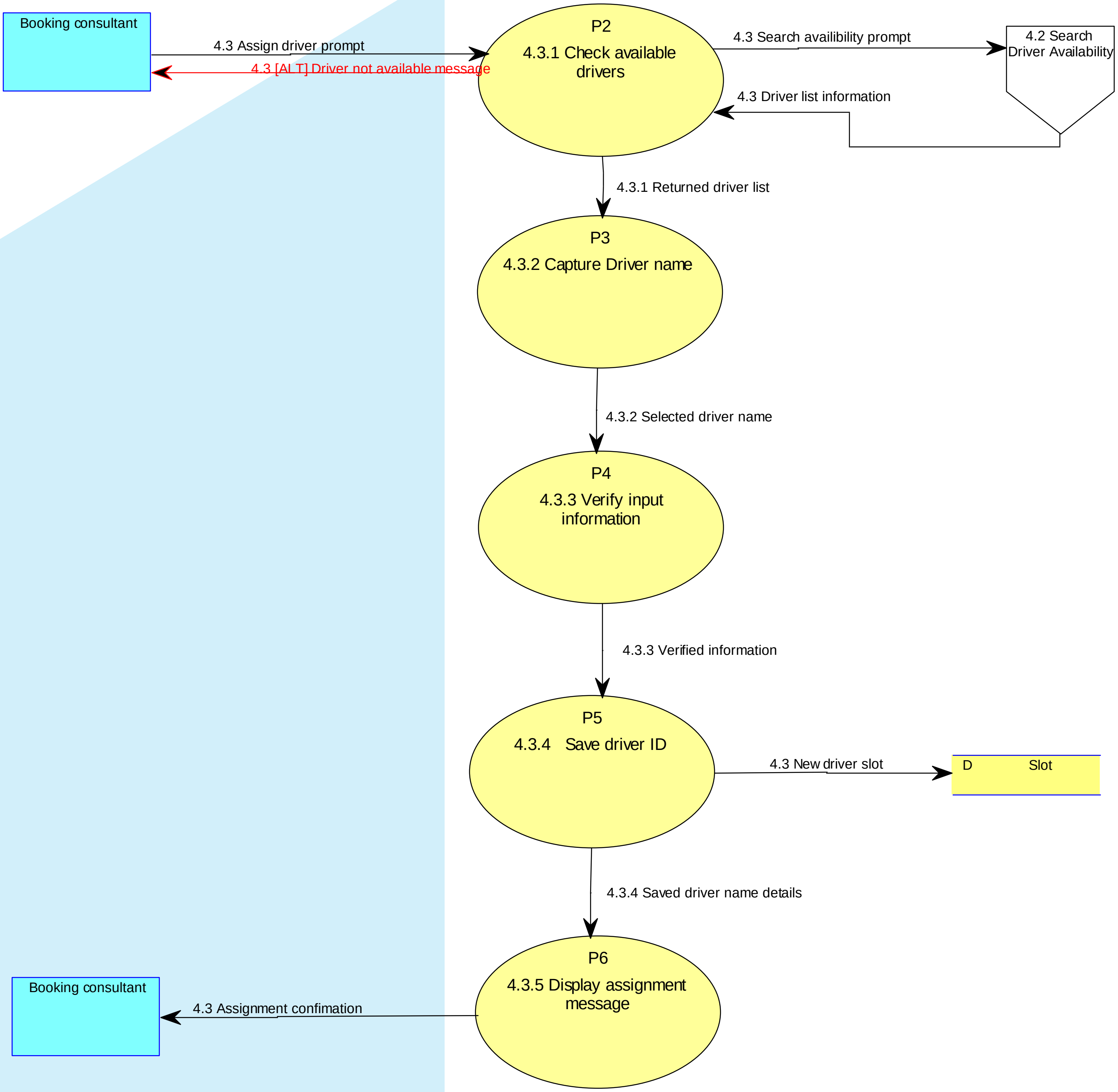

3.2.8 [Alt] Failed Verification
P8
3.2.1 Request to update employee type
3.2.3 Request employee type details
D 3.2 Employee
D 3.2 Employee Type
3.2 Employee details
3.2 Employee
3.2a Employee details
P47
3.2.2 Check Authority
3.2.2 [Alt] Failed Authority
3.2.2 Employee details
3.2.2 [Alt] b Failed Authority
3.2b Employee type details
P48
3.2.3 Display employee type details
3.2 Employee type details a
P49
3.2.4 Selected employee type details
3.2.4 Employee type details
3.2 Employee type details b
P50
3.2.5 Editable employee type details
3.2 Employee type details c
P52
3.2.6 Provide employee type details
3.2.6 Employee type details
3.2 Employee type details d
P53
3.2.7 Confirm update details
3.2.7 Employee type details a
3.2 Employee type details e
P55
3.2.8 Verification of employee type details
3.2.6 [Alt] Failed Verification
3.2 Employee type details f
P51
3.2.9 Update employee update details
3.2.7 Employee type details
3.2 Employee type details g
P54
3.2.10 Success notification

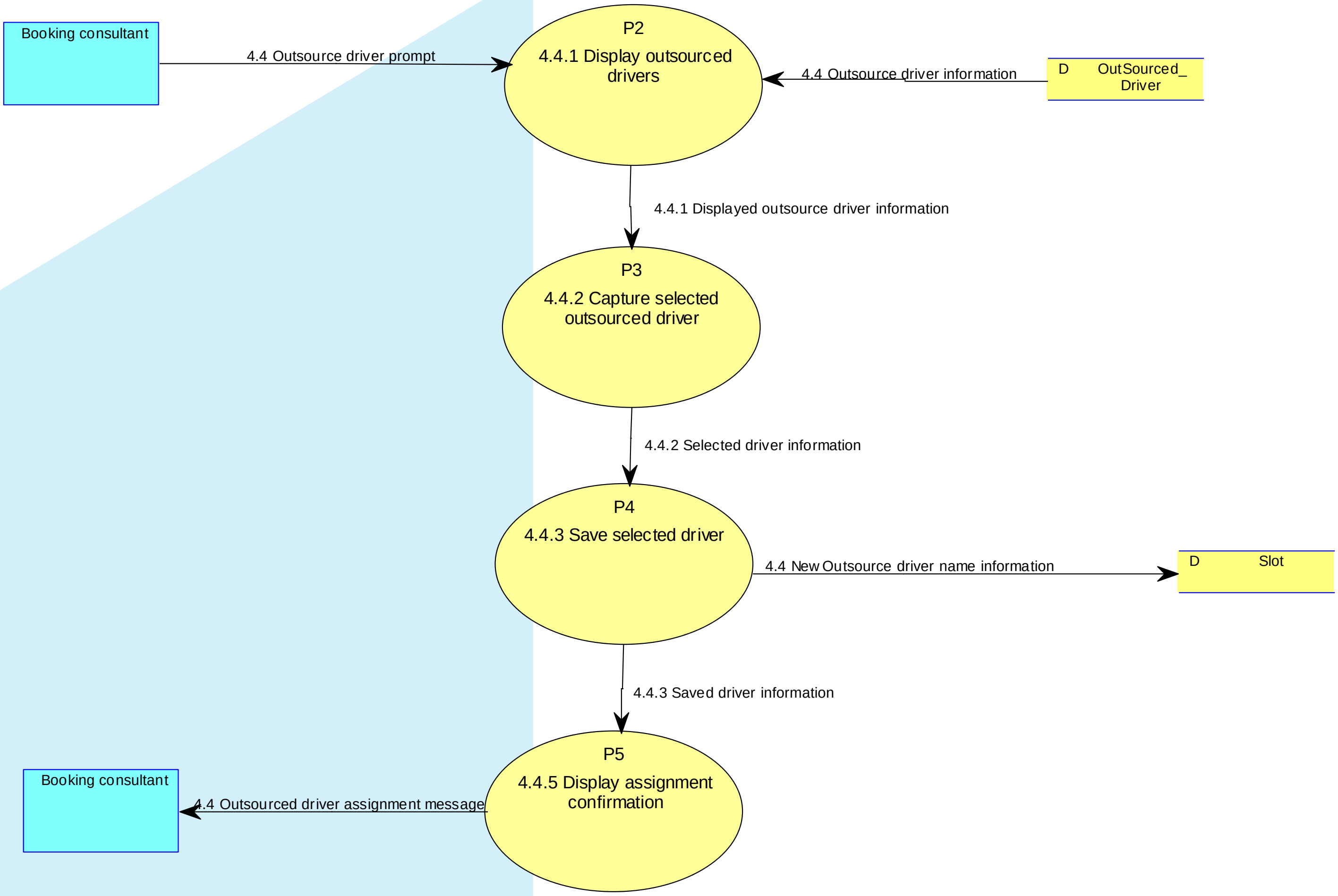

3.3 Employee details
3.3 Employee
P9
3.3.1 Request To Search Employee
3.3.2 [Alt] Failed Authority
D 3.3 Employee
D 3.3 Emplyee Type
3.3 Employee details a
P68
3.3.2 Check Authority
3.3.2 [Alt] Failed authority message
3.3.2 Employee type details
3.3 Employee details b
P69
3.3.3 Request employee type details
3.3 Employee type details c
P70
3.3.4 Provide employee type details
3.3 Employee type details d
P71
3.3.5 Verify employee type details
3.3.5 [Alt] Failed verification message
3.3.5 [Alt] Failed verification
3.3 Employee type details e
P72
3.3.6 Check employee type existence
3.3.6 Employee type details
3.3 Employee type details f
P73
3.3.7 Display employee type details

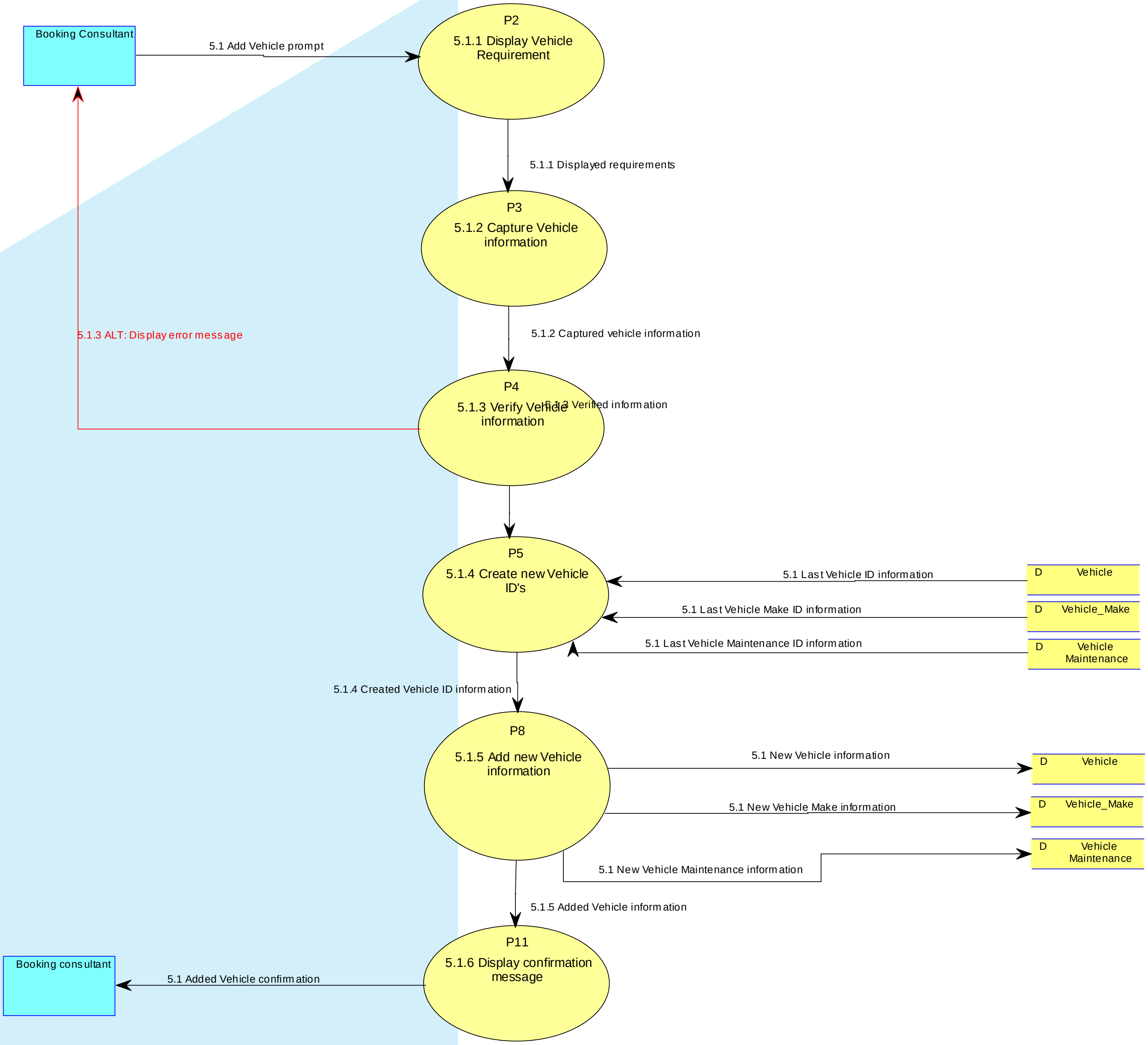

3.4.2 Employee details
3.4.2 [Alt] Failed Authority
D 3.4 Employee
3.4 Employee
3.4 Employee details
P10
3.4.1 Request to remove employee type
D 3.4 Employee Type
P74
3.4.2 Check Authority
3.4 Employee type details b
P75
3.4.3. Provide employee type details
3.4.3 Employee details
3.4 Employee type details c
P76
3.4.4 Request employee type details
3.4 Employee type details d
P77
3.4.5 Provide employee type details
3.4.5 Employee type details
3.4 Employee type details e
P78
3.4.6 Request Confirmation
3.4 Employee type details f
P79
3.4.7 Provide confirmation
3.4 Employee type details g
P80
3.4.8 Disable employee type
3.4.8 [Alt] Failed to disable
3.4.8 Employee type details
3.4 Employee type details h
P81
3.4.9 Display confirmation message
3.4.9 [Alt] Failed to remove message

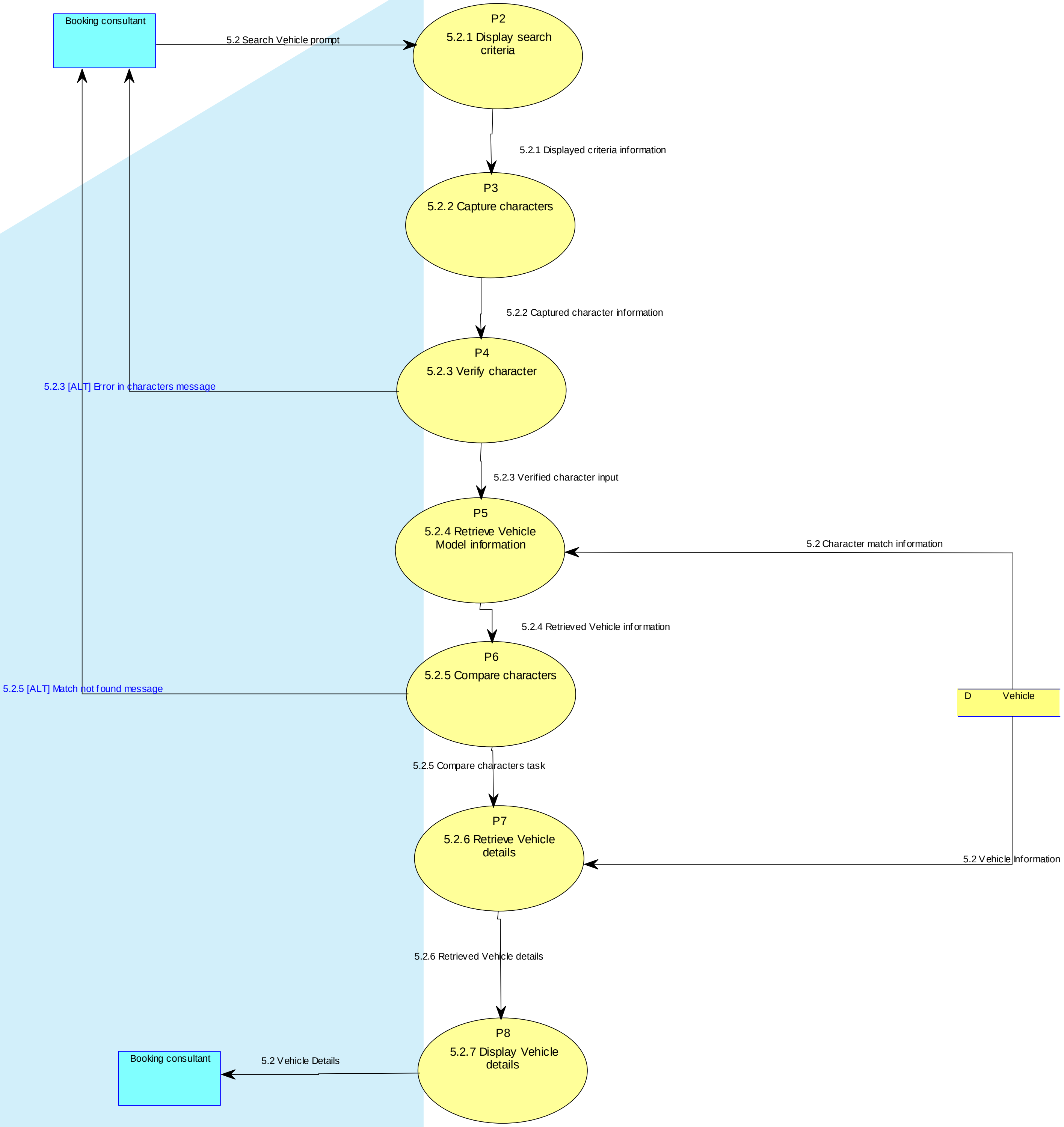

P2
4.1.1 Display search criteria
4.1 Search driver prompt
Booking consultant
4.1.1 Displayed search criteria
P3
4.1.2 Capture character
4.1 Character information
[ALT] Invalid input
4.1.2 Captured character information
P4
4.1.3 Verify character input
4.1.3 Verified character information
P5
4.1.4 Retrieve Driver information
4.1 Character match information
D
Driver
4.1.4 Driver information retrieved
P6
4.1.5 Compare characters
Booking consultant
4.1.6 [ALT] Driver not retrieved
4.1.5 Matched character information
P7
4.1.6 Retrieve Drver details
4.1 Driver information
4.1.6 Retrieved driver details
P8
4.1.7 Display Driver information
Booking consultant
4.1 Display Driver details

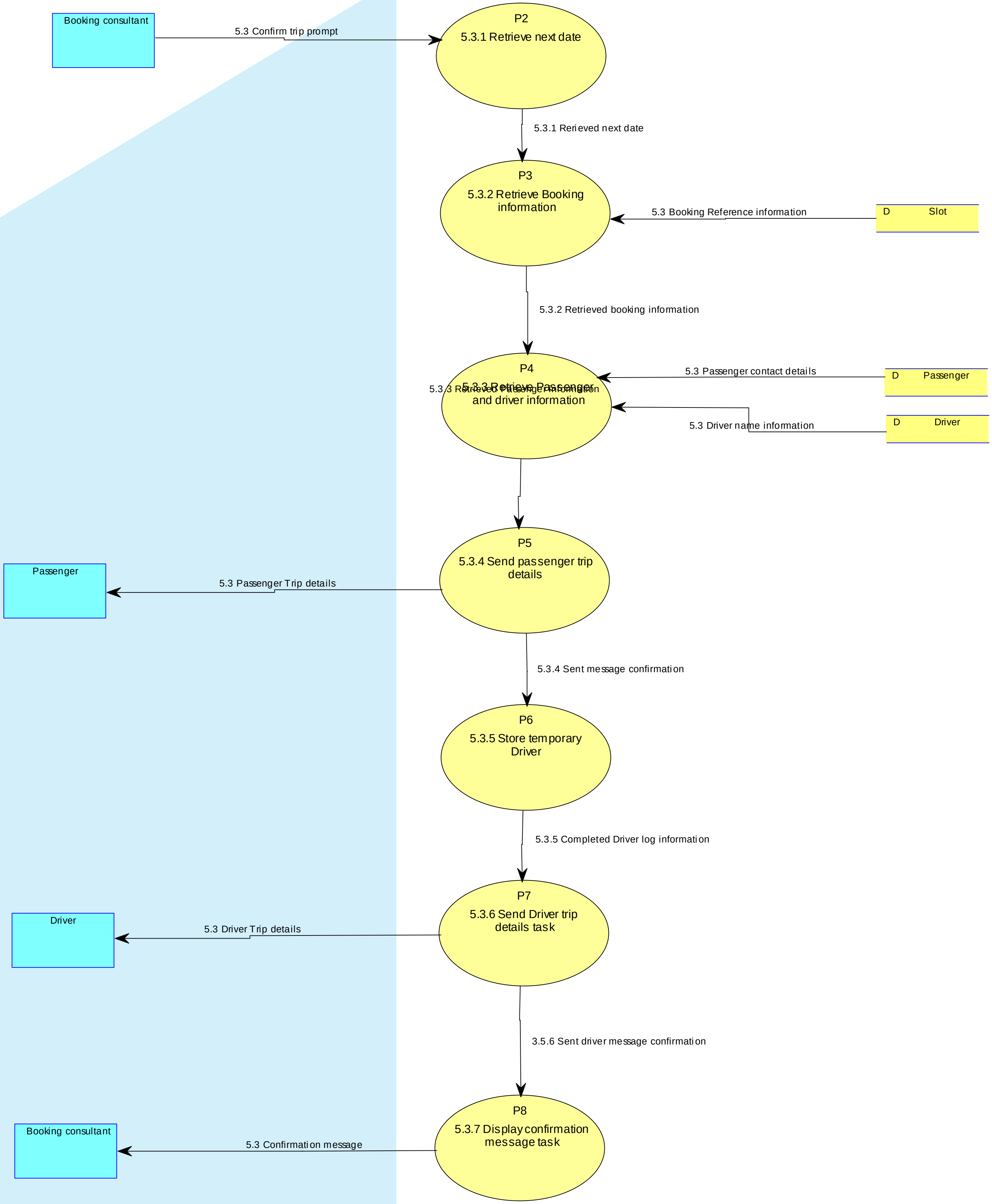

Booking consultant
4.2 Check driver availability prompt
P2
4.2.1 Retrieve trip details
4.2.1 Retrieved trip details
P3
4.2.2 Retrieve slot details
4.2 Slot details
D
Slot
4.2.2 Retrieved slot details
P4
4.2.3 Retrive list of available drivers
4.2.4 [ALT] No Available driver
4.2.3 Retrieved list
P5
4.2.4 Display list of drivers
Booking consultant
4.2 List of available drivers

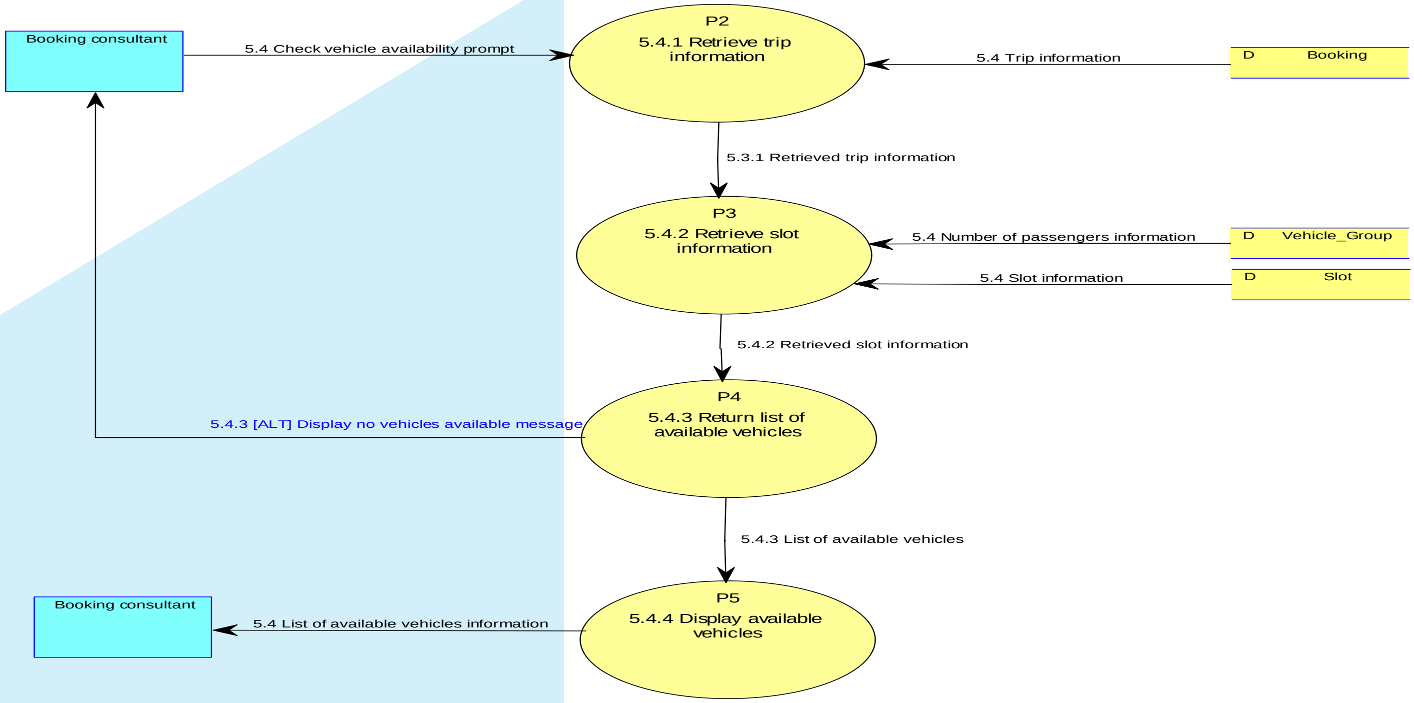

Booking consultant
4.3 Assign driver prompt
4.3 [ALT] Driver not available message
P2
4.3.1 Check available drivers
4.3 Search availibility prompt
4.2 Search Driver Availability
4.3 Driver list information
4.3.1 Returned driver list
P3
4.3.2 Capture Driver name
4.3.2 Selected driver name
P4
4.3.3 Verify input information
4.3.3 Verified information
P5
4.3.4 Save driver ID
4.3 New driver slot
D
Slot
4.3.4 Saved driver name details
P6
4.3.5 Display assignment message
Booking consultant
4.3 Assignment confimation

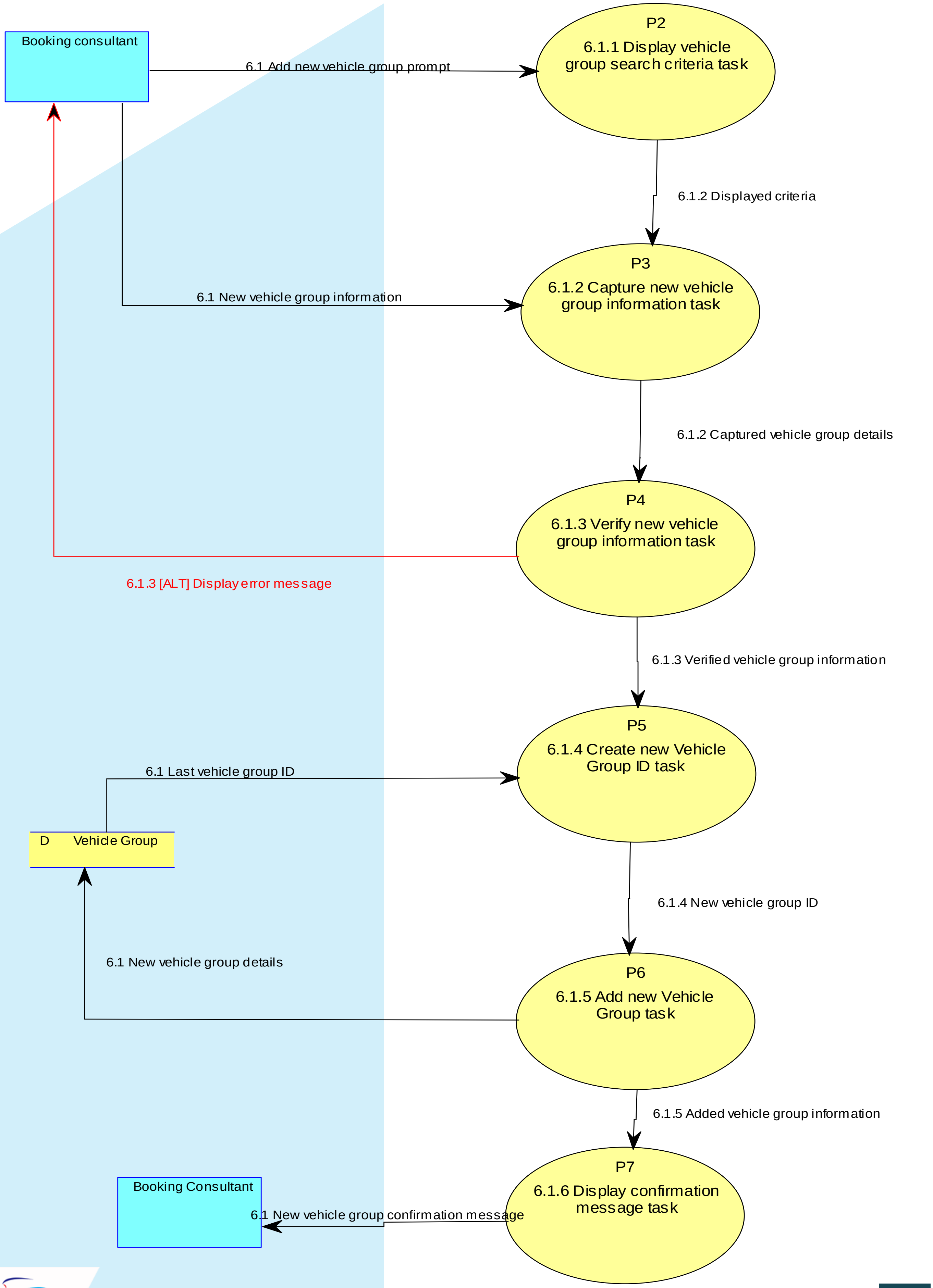

Booking consultant
4.4 Outsource driver prompt
P2
4.4.1 Display outsourced drivers
4.4 Outsource driver information
D OutSourced_ Driver
4.4.1 Displayed outsource driver information
P3
4.4.2 Capture selected outsourced driver
4.4.2 Selected driver information
P4
4.4.3 Save selected driver
4.4 New Outsource driver name information
D Slot
4.4.3 Saved driver information
P5
4.4.5 Display assignment confirmation
4.4 Outsourced driver assignment message
Booking consultant

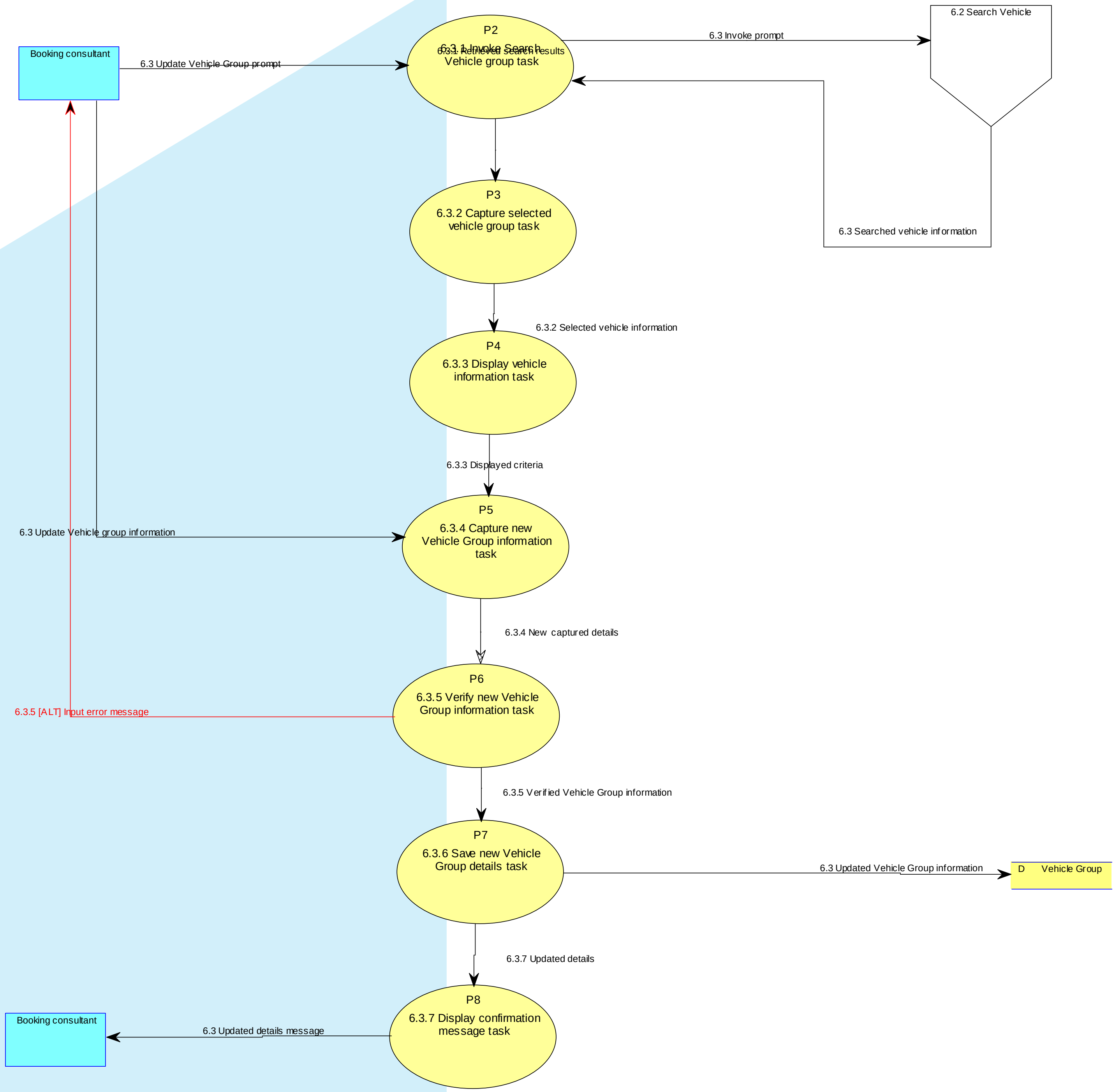

Booking Consultant
5.1 Add Vehicle prompt
P2
5.1.1 Display Vehicle Requirement
5.1.1 Displayed requirements
P3
5.1.2 Capture Vehicle information
5.1.2 Captured vehicle information
5.1.3 ALT: Display error message
P4
5.1.3 Verify Vehicle information
5.1.3 Verified information
P5
5.1.4 Create new Vehicle ID's
5.1 Last Vehicle ID information
D Vehicle
5.1 Last Vehicle Make ID information
D Vehicle_Make
5.1 Last Vehicle Maintenance ID information
D Vehicle Maintenance
5.1.4 Created Vehicle ID information
P8
5.1.5 Add new Vehicle information
5.1 New Vehicle information
D Vehicle
5.1 New Vehicle Make information
D Vehicle_Make
5.1 New Vehicle Maintenance information
D Vehicle Maintenance
5.1.5 Added Vehicle information
P11
5.1.6 Display confirmation message
5.1 Added Vehicle confirmation
Booking consultant

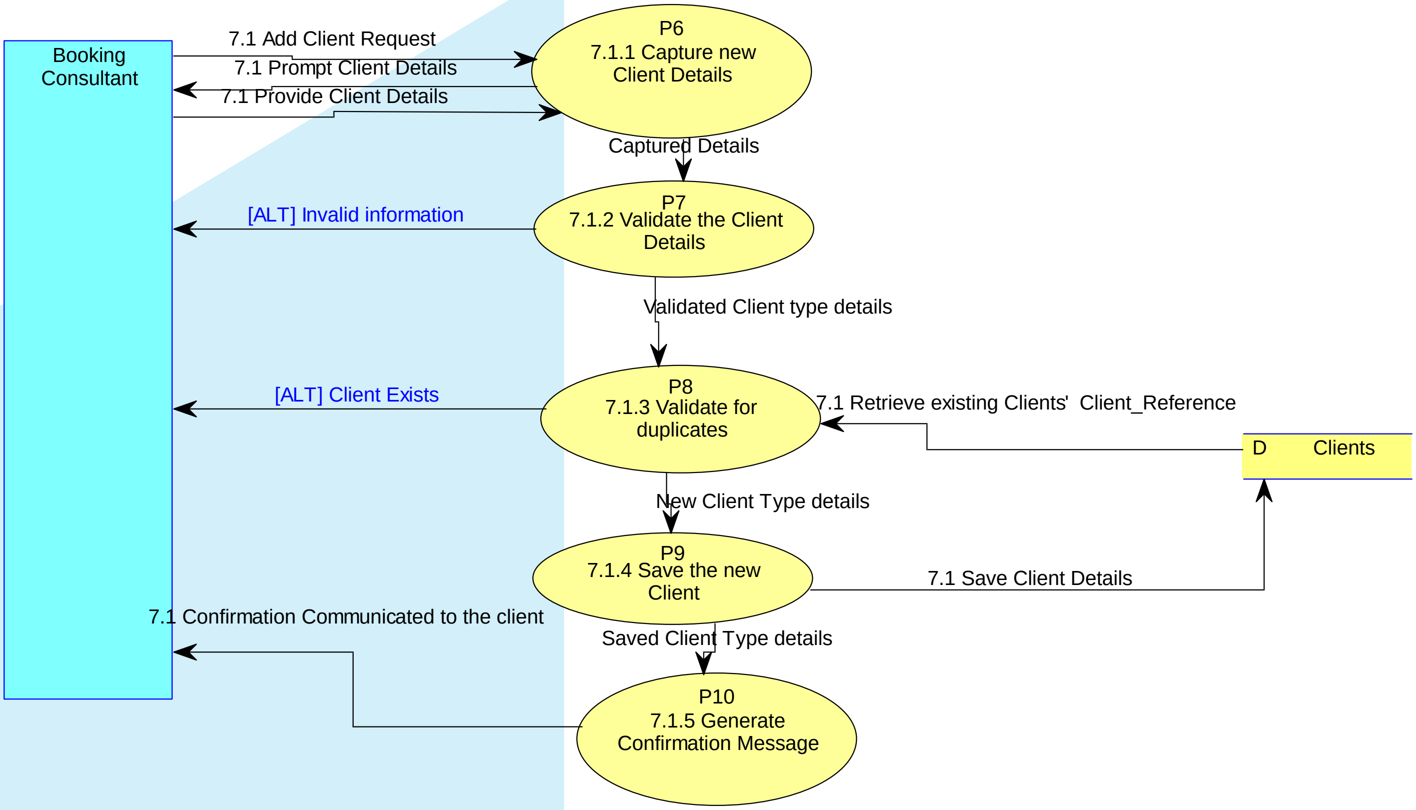

Booking consultant
5.2 Search Vehicle prompt
P2
5.2.1 Display search criteria
5.2.1 Displayed criteria information
P3
5.2.2 Capture characters
5.2.2 Captured character information
P4
5.2.3 Verify character
5.2.3 [ALT] Error in characters message
5.2.3 Verified character input
P5
5.2.4 Retrieve Vehicle Model information
5.2 Character match information
5.2.4 Retrieved Vehicle information
P6
5.2.5 Compare characters
5.2.5 [ALT] Match not found message
D
Vehicle
5.2.5 Compare characters task
P7
5.2.6 Retrieve Vehicle details
5.2 Vehicle Information
5.2.6 Retrieved Vehicle details
P8
5.2.7 Display Vehicle details
5.2 Vehicle Details
Booking consultant

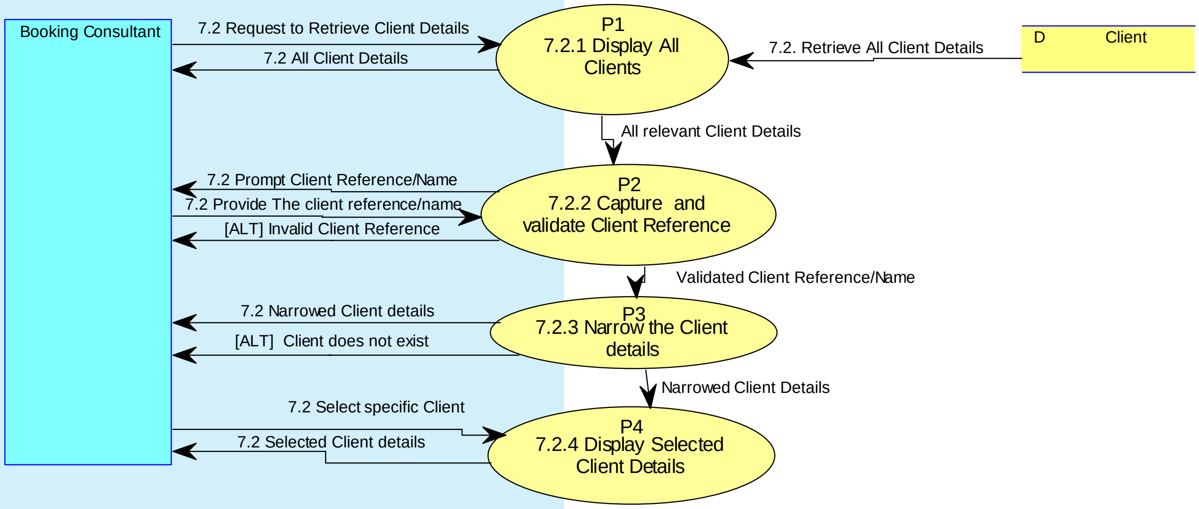

Booking consultant
5.3 Confirm trip prompt
P2
5.3.1 Retrieve next date
5.3.1 Rerieved next date
P3
5.3.2 Retrieve Booking information
5.3 Booking Reference information
D Slot
5.3.2 Retrieved booking information
P4
5.3.3 Retrieve Passenger and driver information
5.3.3 Retrieved Passenger information
5.3 Passenger contact details
D Passenger
5.3 Driver name information
D Driver
P5
5.3.4 Send passenger trip details
Passenger
5.3 Passenger Trip details
5.3.4 Sent message confirmation
P6
5.3.5 Store temporary Driver
5.3.5 Completed Driver log information
P7
5.3.6 Send Driver trip details task
Driver
5.3 Driver Trip details
3.5.6 Sent driver message confirmation
P8
5.3.7 Display confirmation message task
Booking consultant
5.3 Confirmation message

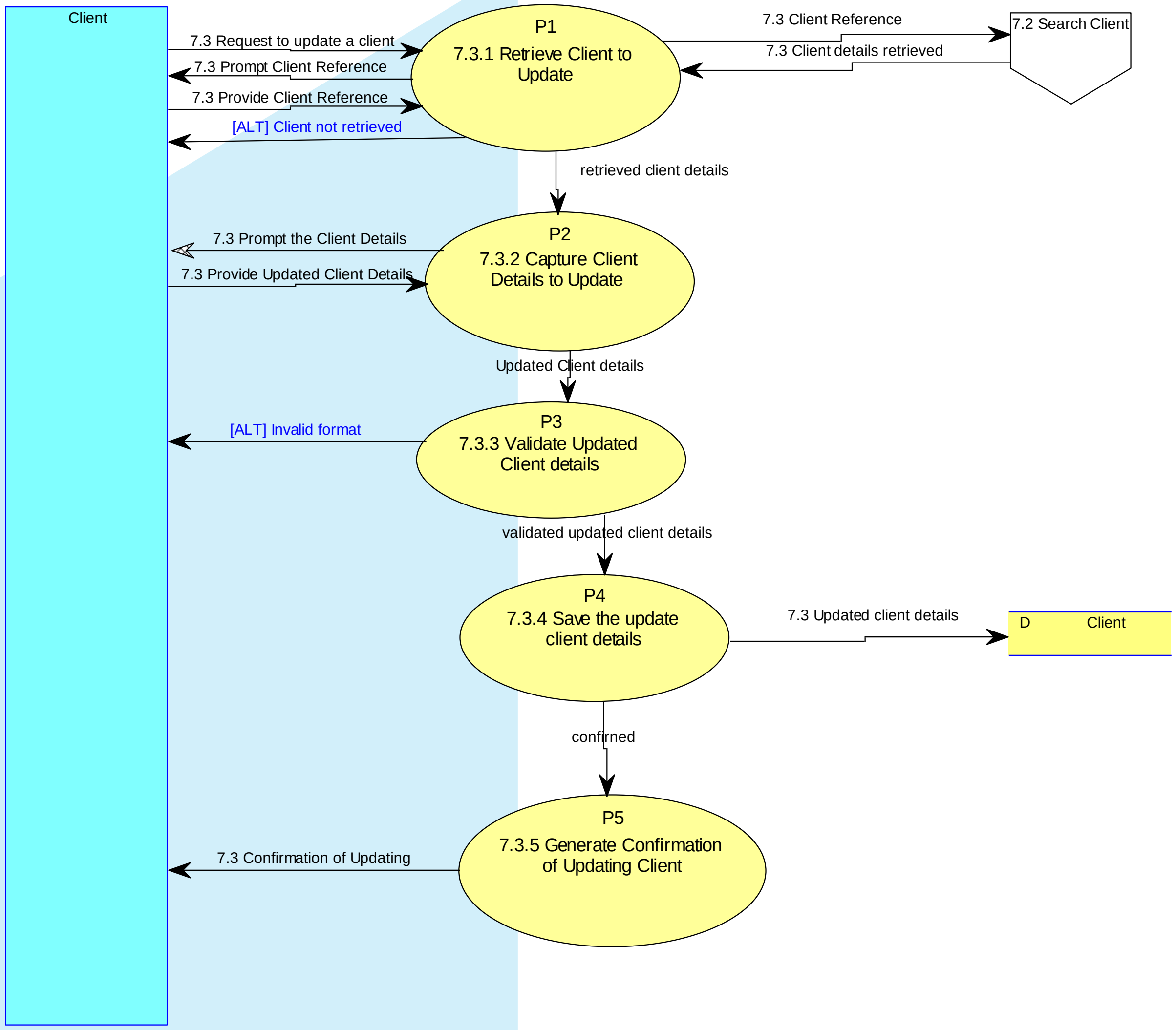

Booking consultant
5.4 Check vehicle availability prompt
P2
5.4.1 Retrieve trip information
5.4 Trip information
D
Booking
5.3.1 Retrieved trip information
P3
5.4.2 Retrieve slot information
5.4 Number of passengers information
D
Vehicle_Group
5.4 Slot information
D
Slot
5.4.2 Retrieved slot information
P4
5.4.3 Return list of available vehicles
5.4.3 [ALT] Display no vehicles available message
5.4.3 List of available vehicles
P5
5.4.4 Display available vehicles
Booking consultant
5.4 List of available vehicles information

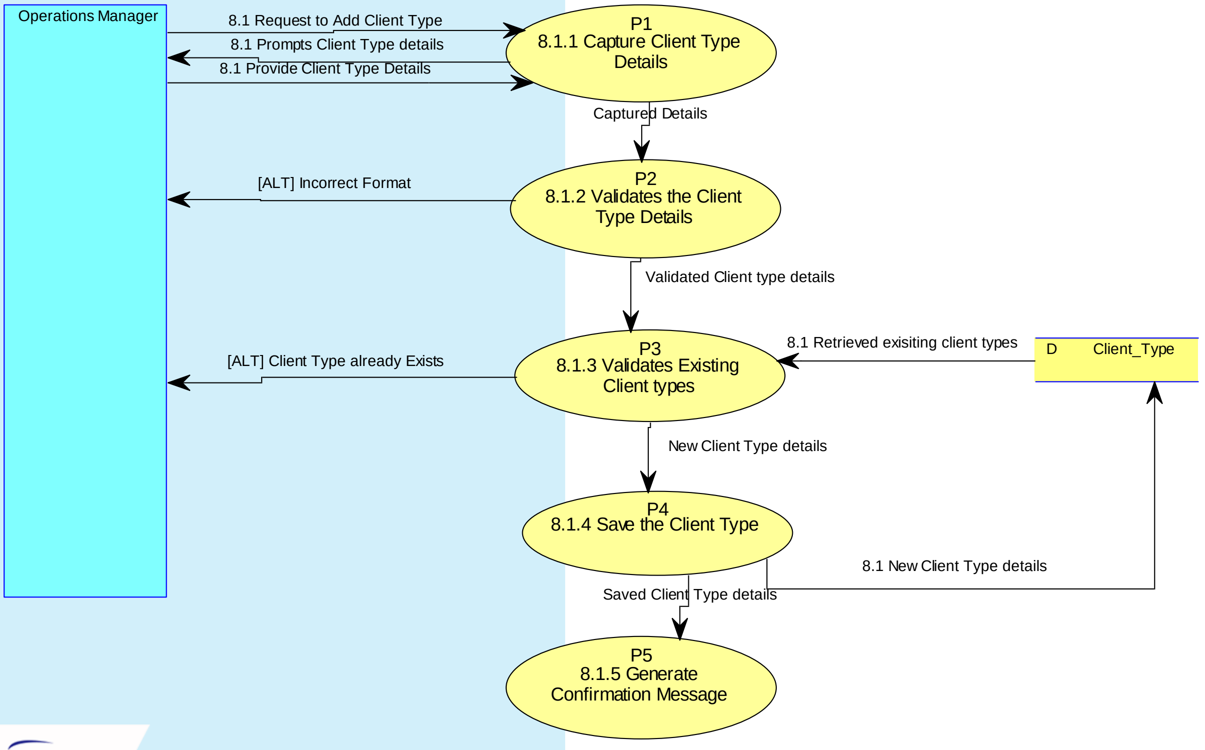

Booking consultant
6.1 Add new vehicle group prompt
P2
6.1.1 Display vehicle group search criteria task
6.1.2 Displayed criteria
6.1 New vehicle group information
P3
6.1.2 Capture new vehicle group information task
6.1.2 Captured vehicle group details
P4
6.1.3 Verify new vehicle group information task
6.1.3 [ALT] Display error message
6.1.3 Verified vehicle group information
P5
6.1.4 Create new Vehicle Group ID task
6.1 Last vehicle group ID
D Vehicle Group
6.1.4 New vehicle group ID
6.1 New vehicle group details
P6
6.1.5 Add new Vehicle Group task
6.1.5 Added vehicle group information
P7
6.1.6 Display confirmation message task
Booking Consultant
6.1 New vehicle group confirmation message

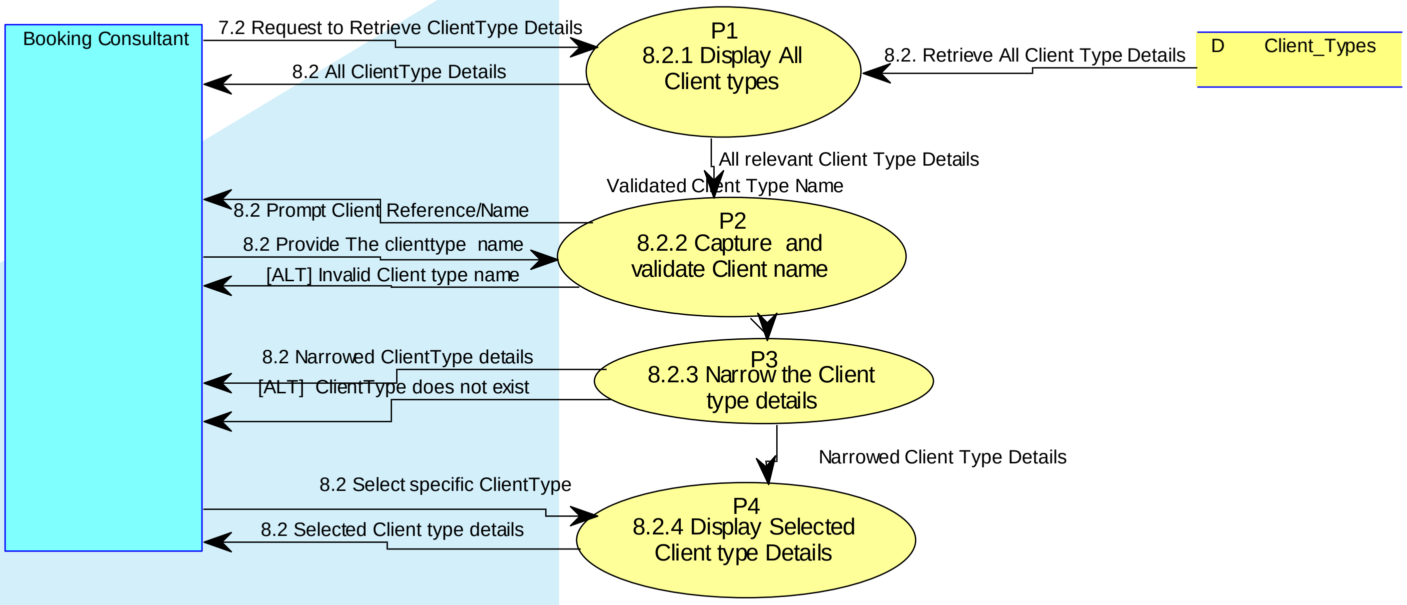

Booking consultant
6.3 Update Vehicle Group prompt
P2
6.3.1 Invoke Search Vehicle group task
6.3.1 Retrieved search results
6.3 Invoke prompt
6.2 Search Vehicle
6.3 Searched vehicle information
P3
6.3.2 Capture selected vehicle group task
6.3.2 Selected vehicle information
P4
6.3.3 Display vehicle information task
6.3.3 Displayed criteria
6.3 Update Vehicle group information
P5
6.3.4 Capture new Vehicle Group information task
6.3.4 New captured details
P6
6.3.5 Verify new Vehicle Group information task
6.3.5 [ALT] Input error message
6.3.5 Verified Vehicle Group information
P7
6.3.6 Save new Vehicle Group details task
6.3 Updated Vehicle Group information
D
Vehicle Group
6.3.7 Updated details
P8
6.3.7 Display confirmation message task
6.3 Updated details message
Booking consultant

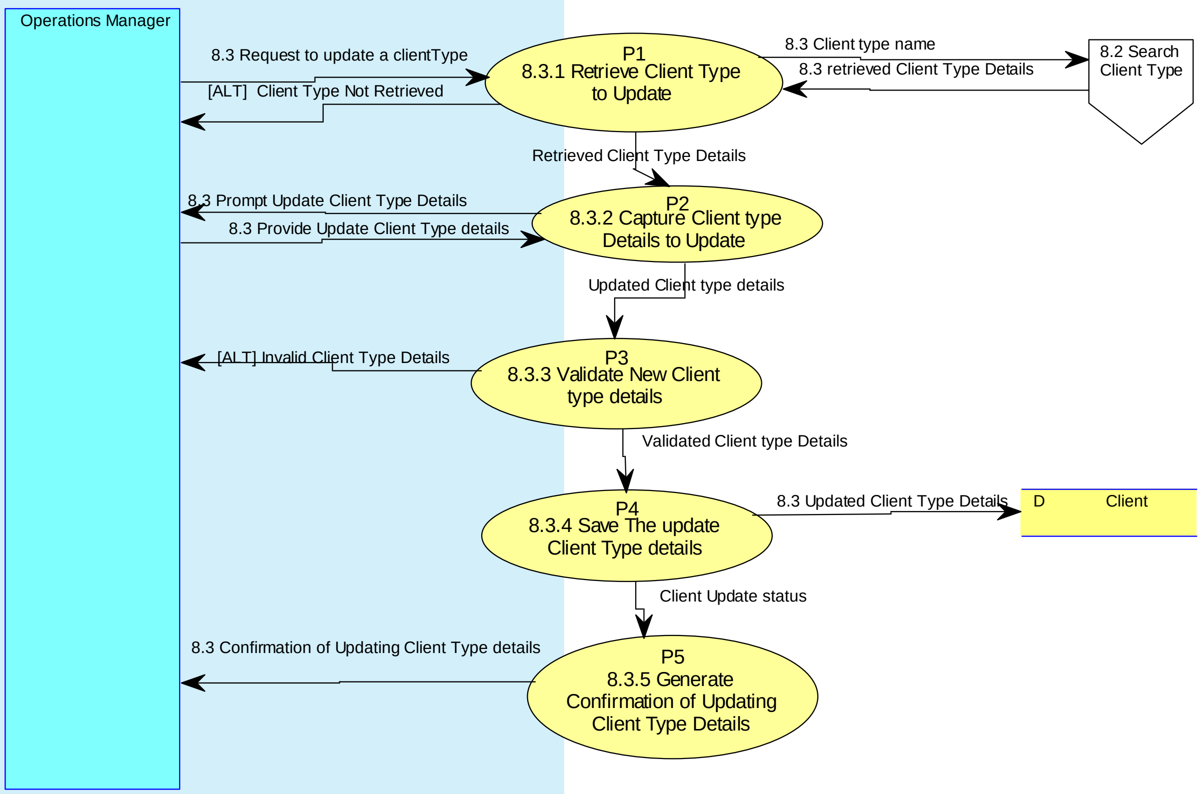

Booking Consultant
7.1 Add Client Request
7.1 Prompt Client Details
7.1 Provide Client Details
P6
7.1.1 Capture new Client Details
Captured Details
P7
7.1.2 Validate the Client Details
[ALT] Invalid information
Validated Client type details
P8
7.1.3 Validate for duplicates
[ALT] Client Exists
7.1 Retrieve existing Clients' Client_Reference
D
Clients
New Client Type details
P9
7.1.4 Save the new Client
7.1 Save Client Details
7.1 Confirmation Communicated to the client
Saved Client Type details
P10
7.1.5 Generate Confirmation Message

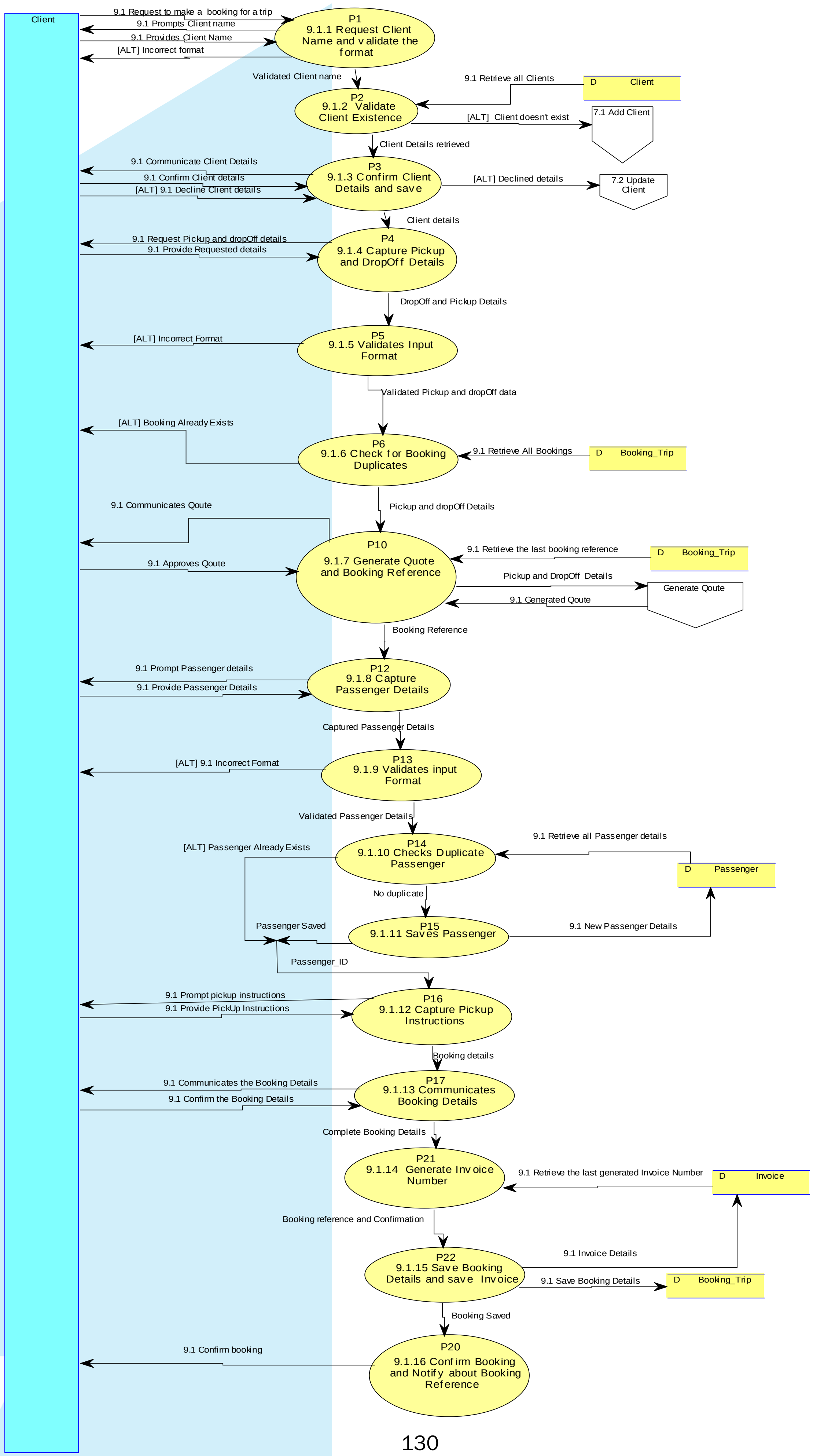

Booking Consultant
7.2 Request to Retrieve Client Details
7.2 All Client Details
P1
7.2.1 Display All Clients
7.2. Retrieve All Client Details
D
Client
All relevant Client Details
7.2 Prompt Client Reference/Name
7.2 Provide The client reference/name
[ALT] Invalid Client Reference
P2
7.2.2 Capture and validate Client Reference
Validated Client Reference/Name
7.2 Narrowed Client details
[ALT] Client does not exist
P3
7.2.3 Narrow the Client details
Narrowed Client Details
7.2 Select specific Client
7.2 Selected Client details
P4
7.2.4 Display Selected Client Details

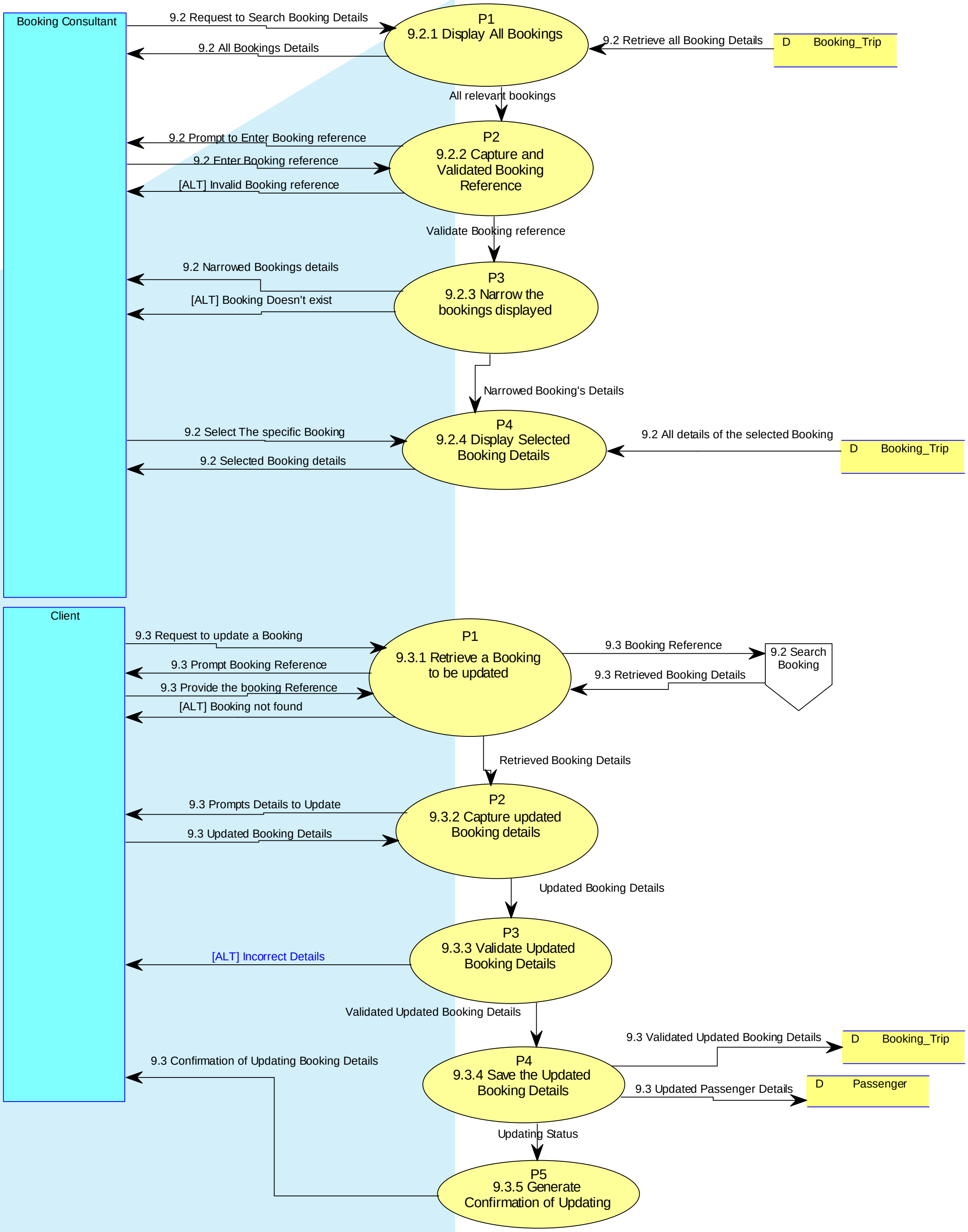

Client
7.3 Request to update a client
7.3 Prompt Client Reference
7.3 Provide Client Reference
[ALT] Client not retrieved
P1
7.3.1 Retrieve Client to Update
7.3 Client Reference
7.3 Client details retrieved
7.2 Search Client
retrieved client details
7.3 Prompt the Client Details
7.3 Provide Updated Client Details
P2
7.3.2 Capture Client Details to Update
Updated Client details
[ALT] Invalid format
P3
7.3.3 Validate Updated Client details
validated updated client details
P4
7.3.4 Save the update client details
7.3 Updated client details
D
Client
confirned
P5
7.3.5 Generate Confirmation of Updating Client
7.3 Confirmation of Updating

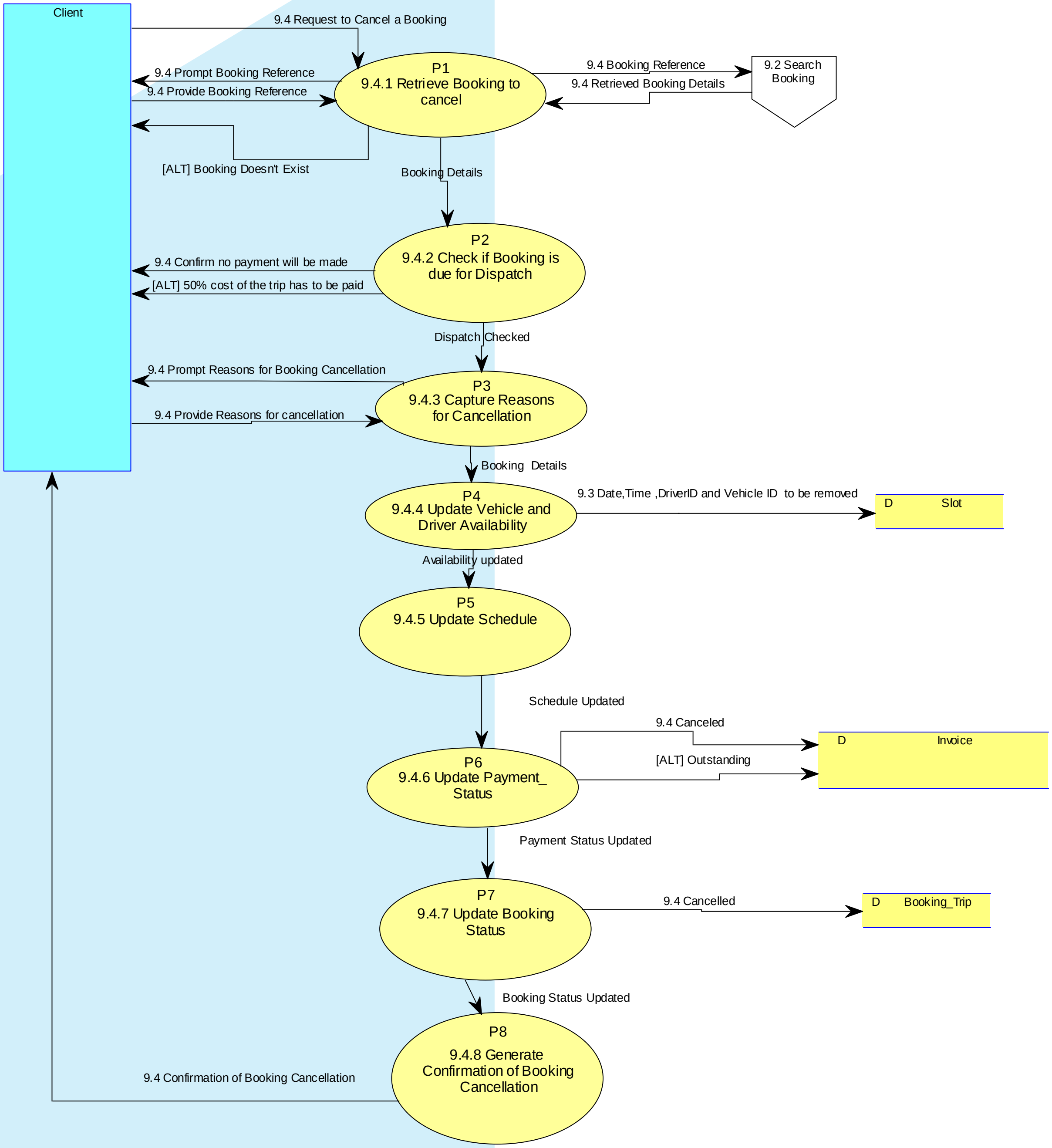

Operations Manager
8.1 Request to Add Client Type
8.1 Prompts Client Type details
8.1 Provide Client Type Details
P1
8.1.1 Capture Client Type Details
Captured Details
[ALT] Incorrect Format
P2
8.1.2 Validates the Client Type Details
Validated Client type details
[ALT] Client Type already Exists
P3
8.1.3 Validates Existing Client types
8.1 Retrieved exisiting client types
D
Client_Type
New Client Type details
P4
8.1.4 Save the Client Type
8.1 New Client Type details
Saved Client Type details
P5
8.1.5 Generate Confirmation Message

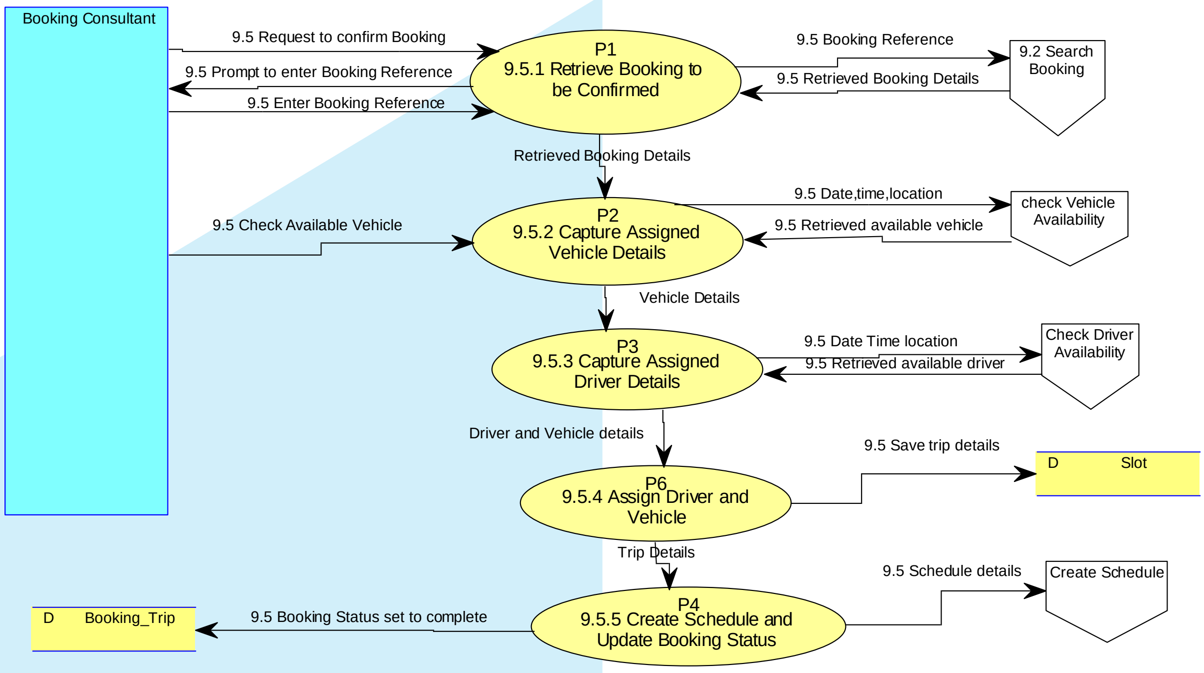

Booking Consultant
7.2 Request to Retrieve ClientType Details
8.2 All ClientType Details
P1
8.2.1 Display All Client types
8.2. Retrieve All Client Type Details
D Client_Types
All relevant Client Type Details
Validated Client Type Name
8.2 Prompt Client Reference/Name
8.2 Provide The clienttype name
[ALT] Invalid Client type name
P2
8.2.2 Capture and validate Client name
8.2 Narrowed ClientType details
[ALT] ClientType does not exist
P3
8.2.3 Narrow the Client type details
Narrowed Client Type Details
8.2 Select specific ClientType
8.2 Selected Client type details
P4
8.2.4 Display Selected Client type Details

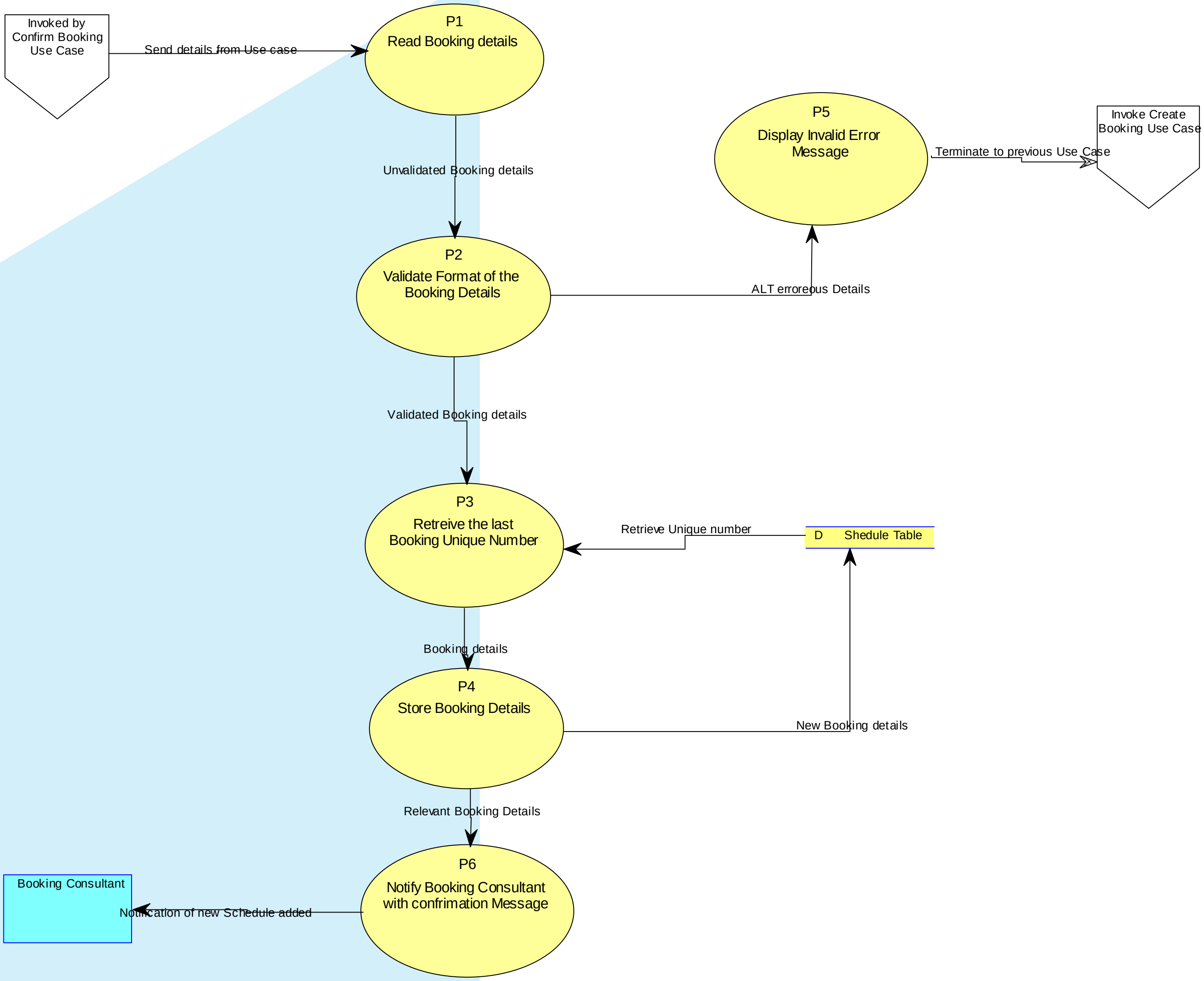

Operations Manager
8.3 Request to update a clientType
[ALT] Client Type Not Retrieved
P1
8.3.1 Retrieve Client Type to Update
8.3 Client type name
8.3 retrieved Client Type Details
8.2 Search Client Type
Retrieved Client Type Details
8.3 Prompt Update Client Type Details
8.3 Provide Update Client Type details
P2
8.3.2 Capture Client type Details to Update
Updated Client type details
[ALT] Invalid Client Type Details
P3
8.3.3 Validate New Client type details
Validated Client type Details
P4
8.3.4 Save The update Client Type details
8.3 Updated Client Type Details
D Client
Client Update status
8.3 Confirmation of Updating Client Type details
P5
8.3.5 Generate Confirmation of Updating Client Type Details

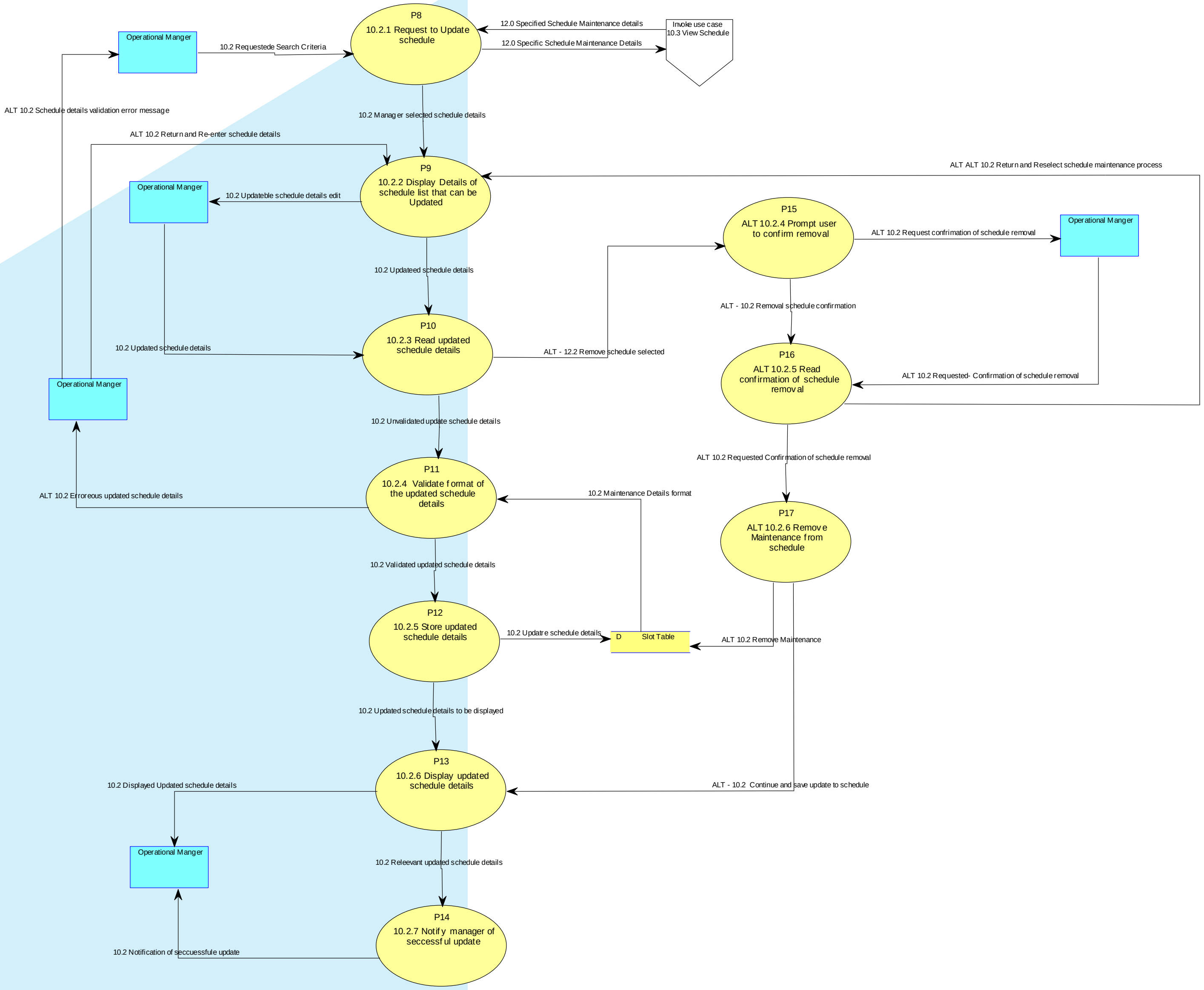

Client
9.1 Request to make a booking for a trip
9.1 Prompts Client name
9.1 Provides Client Name
[ALT] Incorrect format
P1
9.1.1 Request Client Name and validate the format
Validated Client name
9.1 Retrieve all Clients
D Client
P2
9.1.2 Validate Client Existence
[ALT] Client doesn't exist
7.1 Add Client
Client Details retrieved
9.1 Communicate Client Details
9.1 Confirm Client details
[ALT] 9.1 Decline Client details
P3
9.1.3 Confirm Client Details and save
[ALT] Declined details
7.2 Update Client
Client details
9.1 Request Pickup and dropOff details
9.1 Provide Requested details
P4
9.1.4 Capture Pickup and DropOff Details
DropOff and Pickup Details
[ALT] Incorrect Format
P5
9.1.5 Validates Input Format
Validated Pickup and dropOff data
[ALT] Booking Already Exists
P6
9.1.6 Check for Booking Duplicates
9.1 Retrieve All Bookings
D Booking_Trip
9.1 Communicates Qoute
Pickup and dropOff Details
9.1 Approves Qoute
P10
9.1.7 Generate Quote and Booking Reference
9.1 Retrieve the last booking reference
D Booking_Trip
Pickup and DropOff Details
Generate Qoute
9.1 Generated Qoute
Booking Reference
9.1 Prompt Passenger details
9.1 Provide Passenger Details
P12
9.1.8 Capture Passenger Details
Captured Passenger Details
[ALT] 9.1 Incorrect Format
P13
9.1.9 Validates input Format
Validated Passenger Details
[ALT] Passenger Already Exists
P14
9.1.10 Checks Duplicate Passenger
9.1 Retrieve all Passenger details
D Passenger
No duplicate
Passenger Saved
P15
9.1.11 Saves Passenger
9.1 New Passenger Details
Passenger_ID
9.1 Prompt pickup instructions
9.1 Provide PickUp Instructions
P16
9.1.12 Capture Pickup Instructions
Booking details
9.1 Communicates the Booking Details
9.1 Confirm the Booking Details
P17
9.1.13 Communicates Booking Details
Complete Booking Details
P21
9.1.14 Generate Invoice Number
9.1 Retrieve the last generated Invoice Number
D Invoice
Booking reference and Confirmation
P22
9.1.15 Save Booking Details and save Invoice
9.1 Invoice Details
9.1 Save Booking Details
D Booking_Trip
Booking Saved
9.1 Confirm booking
P20
9.1.16 Confirm Booking and Notify about Booking Reference

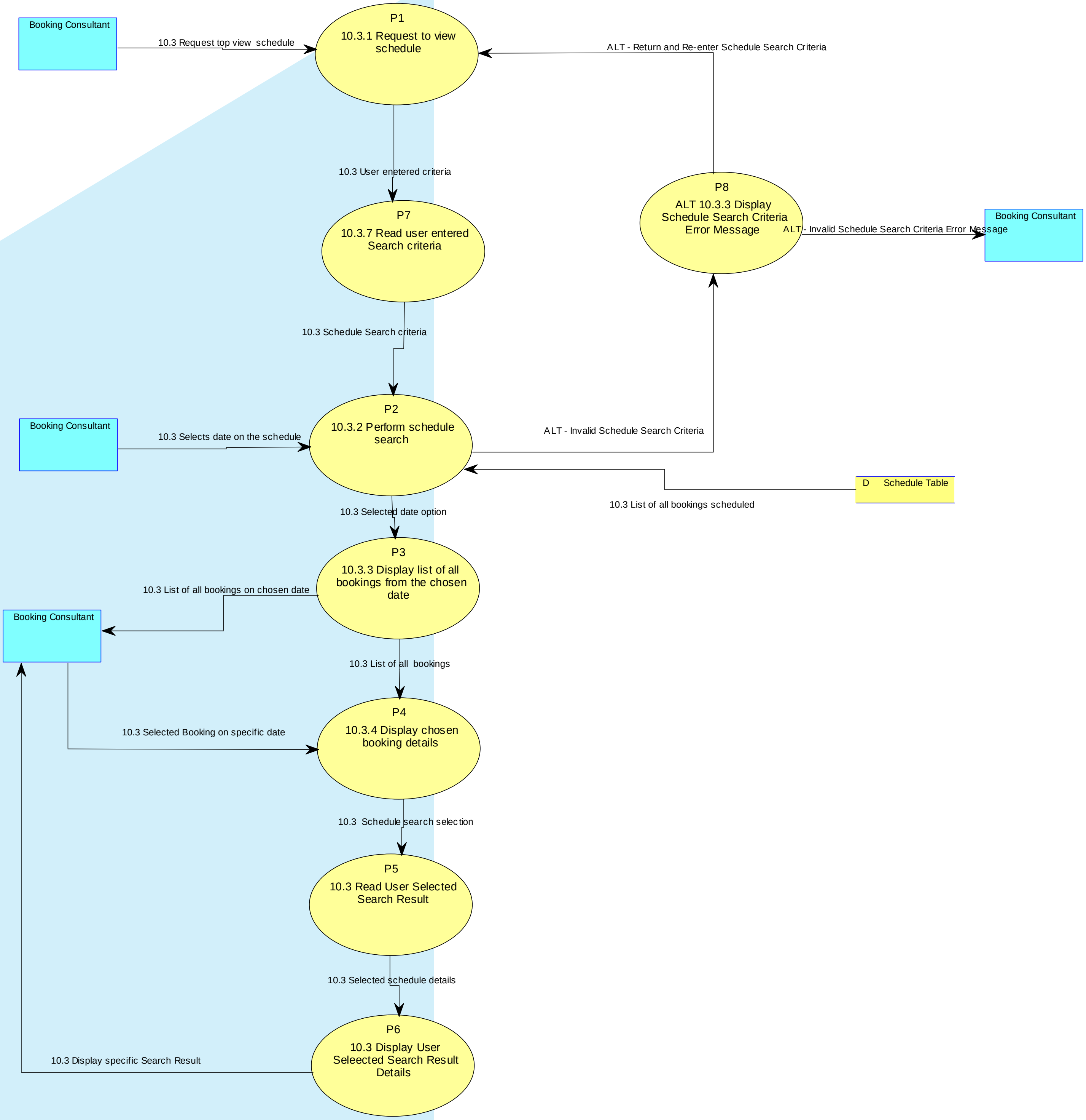

Booking Consultant
9.2 Request to Search Booking Details
9.2 All Bookings Details
P1
9.2.1 Display All Bookings
9.2 Retrieve all Booking Details
D
Booking_Trip
All relevant bookings
9.2 Prompt to Enter Booking reference
9.2 Enter Booking reference
[ALT] Invalid Booking reference
P2
9.2.2 Capture and Validated Booking Reference
Validate Booking reference
9.2 Narrowed Bookings details
[ALT] Booking Doesn't exist
P3
9.2.3 Narrow the bookings displayed
Narrowed Booking's Details
9.2 Select The specific Booking
9.2 Selected Booking details
P4
9.2.4 Display Selected Booking Details
9.2 All details of the selected Booking
D
Booking_Trip
Client
9.3 Request to update a Booking
9.3 Prompt Booking Reference
9.3 Provide the booking Reference
[ALT] Booking not found
P1
9.3.1 Retrieve a Booking to be updated
9.3 Booking Reference
9.3 Retrieved Booking Details
9.2 Search Booking
Retrieved Booking Details
9.3 Prompts Details to Update
9.3 Updated Booking Details
P2
9.3.2 Capture updated Booking details
Updated Booking Details
[ALT] Incorrect Details
P3
9.3.3 Validate Updated Booking Details
Validated Updated Booking Details
9.3 Confirmation of Updating Booking Details
P4
9.3.4 Save the Updated Booking Details
9.3 Validated Updated Booking Details
D
Booking_Trip
9.3 Updated Passenger Details
D
Passenger
Updating Status
P5
9.3.5 Generate Confirmation of Updating

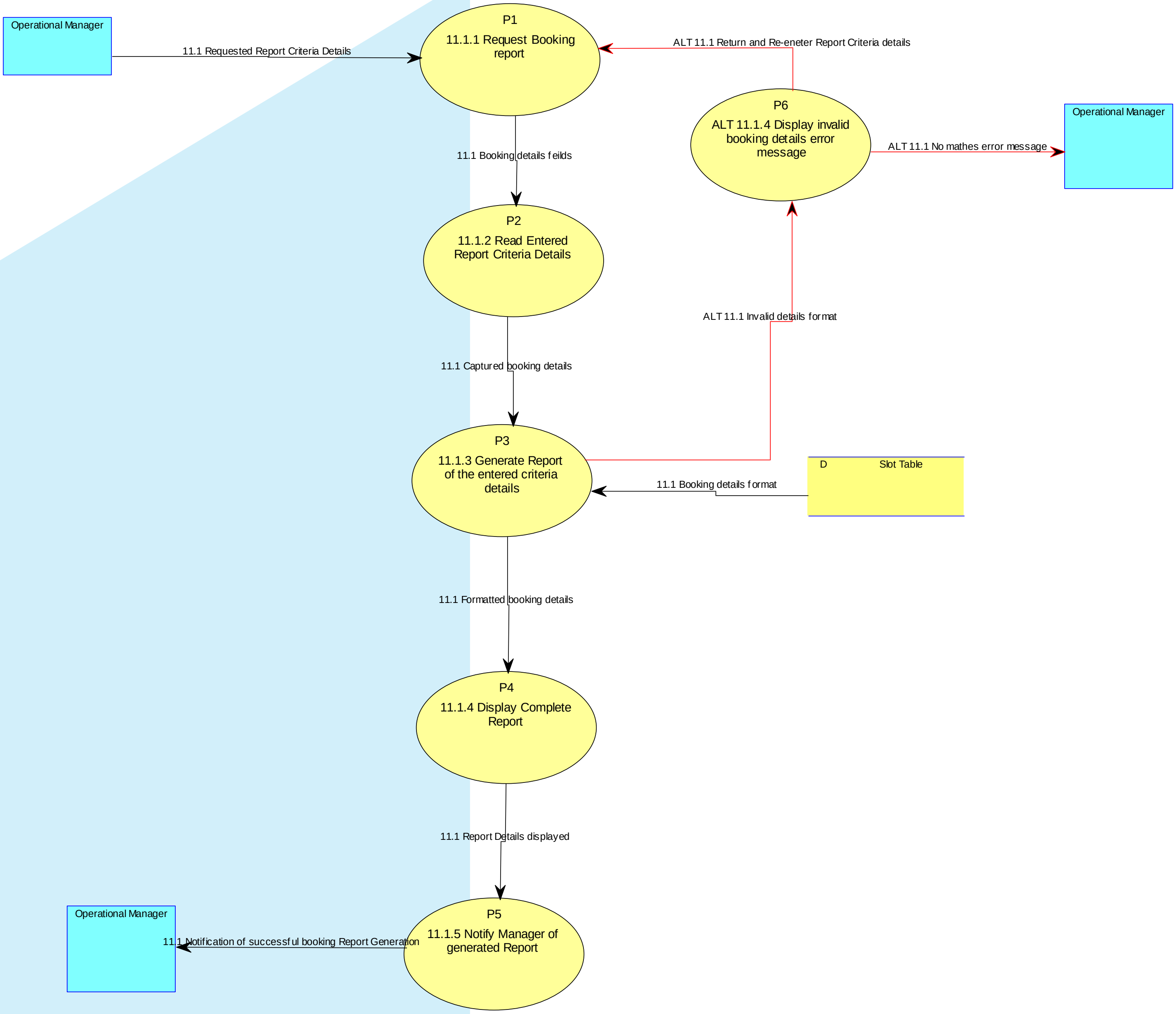

Client
9.4 Request to Cancel a Booking
P1
9.4.1 Retrieve Booking to cancel
9.4 Prompt Booking Reference
9.4 Provide Booking Reference
[ALT] Booking Doesn't Exist
9.4 Booking Reference
9.4 Retrieved Booking Details
9.2 Search Booking
Booking Details
P2
9.4.2 Check if Booking is due for Dispatch
9.4 Confirm no payment will be made
[ALT] 50% cost of the trip has to be paid
Dispatch Checked
P3
9.4.3 Capture Reasons for Cancellation
9.4 Prompt Reasons for Booking Cancellation
9.4 Provide Reasons for cancellation
Booking Details
P4
9.4.4 Update Vehicle and Driver Availability
9.3 Date,Time ,DriverID and Vehicle ID to be removed
D Slot
Availability updated
P5
9.4.5 Update Schedule
Schedule Updated
P6
9.4.6 Update Payment_ Status
9.4 Canceled
[ALT] Outstanding
D Invoice
Payment Status Updated
P7
9.4.7 Update Booking Status
9.4 Cancelled
D Booking_Trip
Booking Status Updated
P8
9.4.8 Generate Confirmation of Booking Cancellation
9.4 Confirmation of Booking Cancellation

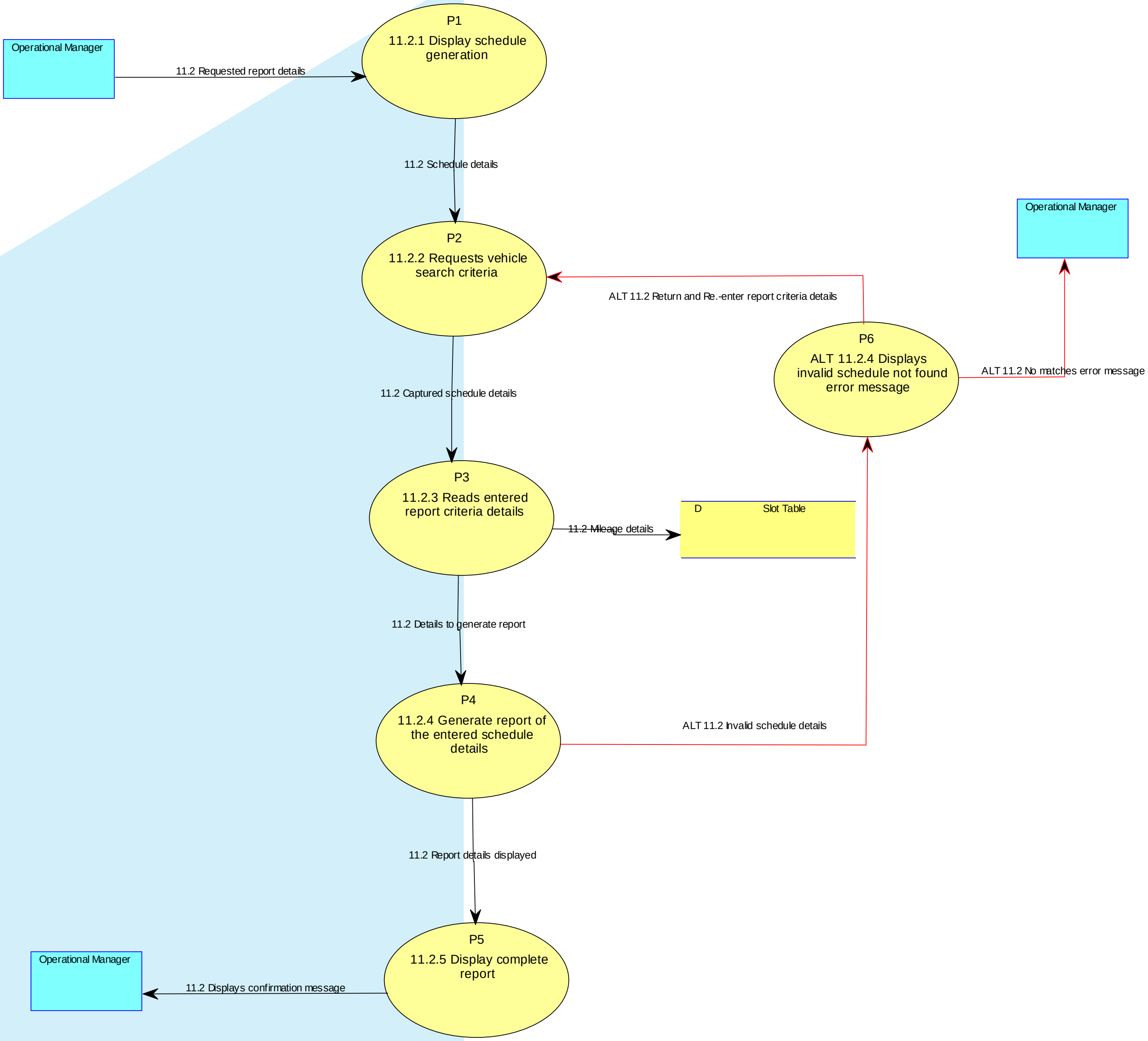

Booking Consultant
9.5 Request to confirm Booking
9.5 Prompt to enter Booking Reference
9.5 Enter Booking Reference
P1
9.5.1 Retrieve Booking to be Confirmed
9.5 Booking Reference
9.5 Retrieved Booking Details
9.2 Search Booking
Retrieved Booking Details
9.5 Check Available Vehicle
P2
9.5.2 Capture Assigned Vehicle Details
9.5 Date,time,location
9.5 Retrieved available vehicle
check Vehicle Availability
Vehicle Details
P3
9.5.3 Capture Assigned Driver Details
9.5 Date Time location
9.5 Retrieved available driver
Check Driver Availability
Driver and Vehicle details
P6
9.5.4 Assign Driver and Vehicle
9.5 Save trip details
D Slot
Trip Details
P4
9.5.5 Create Schedule and Update Booking Status
9.5 Schedule details
Create Schedule
9.5 Booking Status set to complete
D Booking_Trip

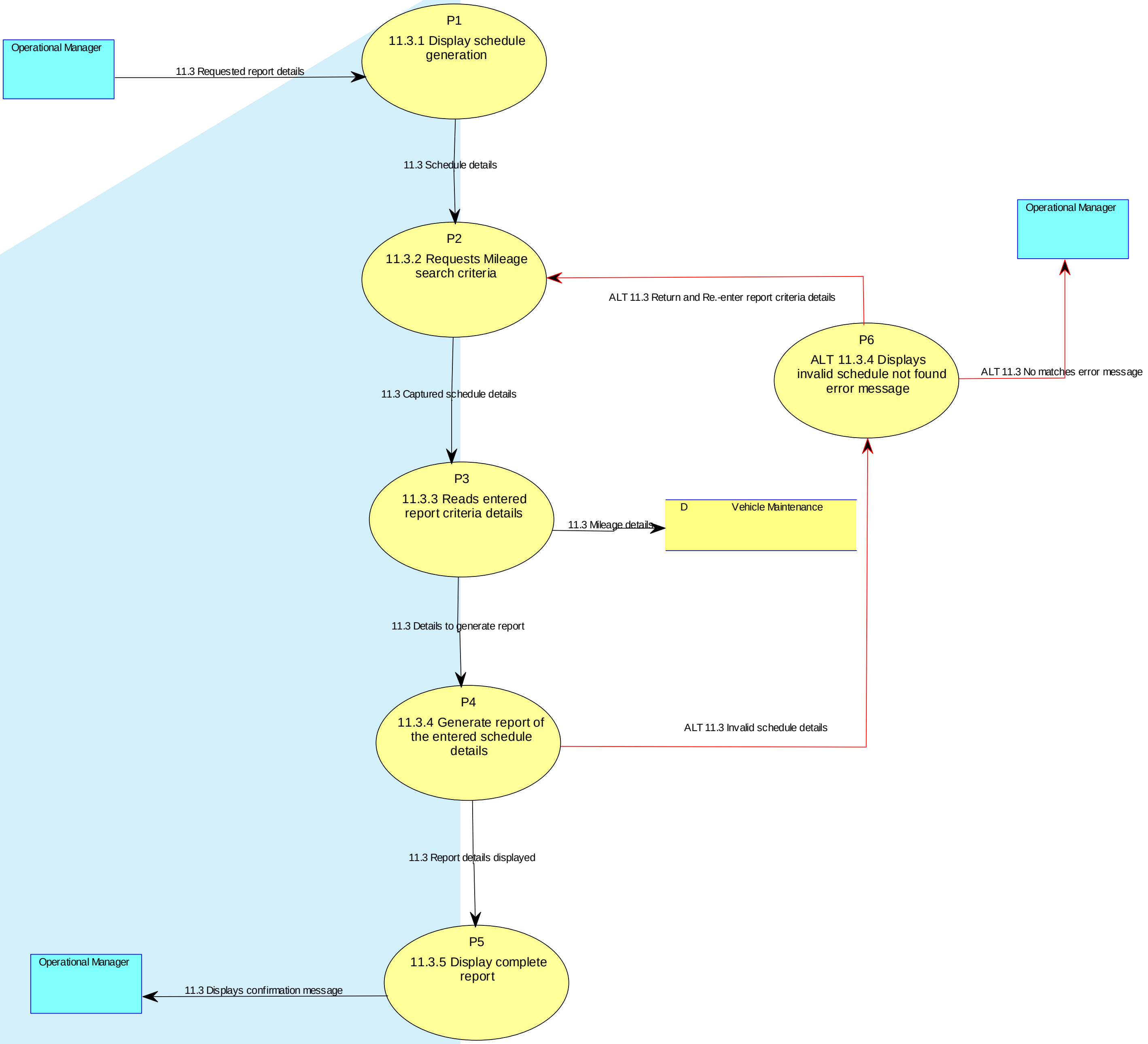

Invoked by Confirm Booking Use Case
Send details from Use case
P1
Read Booking details
Unvalidated Booking details
P5
Display Invalid Error Message
Terminate to previous Use Case
Invoke Create Booking Use Case
P2
Validate Format of the Booking Details
ALT erroreous Details
Validated Booking details
P3
Retreive the last Booking Unique Number
Retrieve Unique number
D
Shedule Table
Booking details
P4
Store Booking Details
New Booking details
Relevant Booking Details
P6
Notify Booking Consultant with confrimation Message
Booking Consultant
Notification of new Schedule added

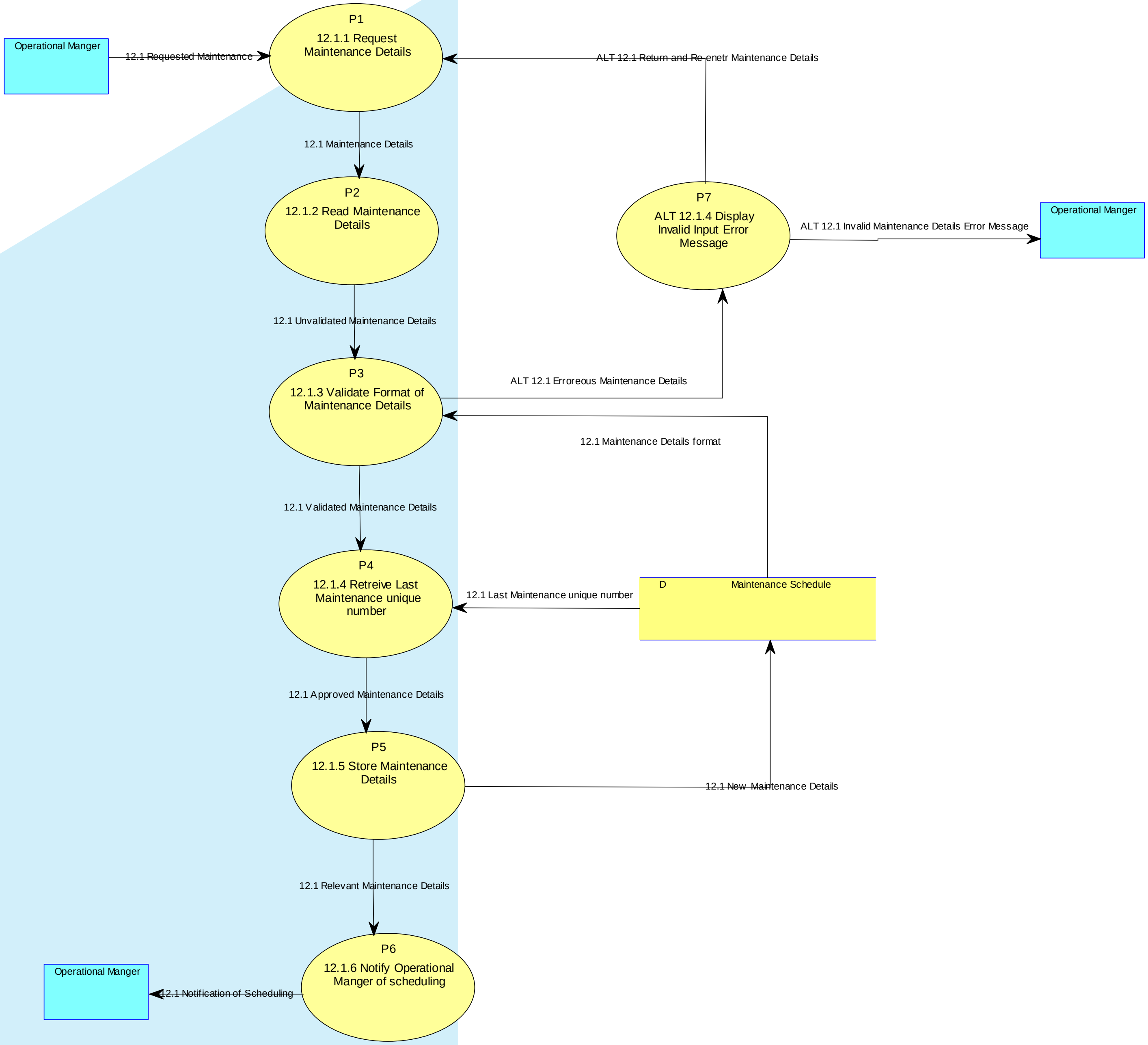

Operational Manger
10.2 Requestede Search Criteria
P8
10.2.1 Request to Update schedule
12.0 Specified Schedule Maintenance details
12.0 Specific Schedule Maintenance Details
Invoke use case
10.3 View Schedule
ALT 10.2 Schedule details validation error message
10.2 Manager selected schedule details
ALT 10.2 Return and Re-enter schedule details
ALT ALT 10.2 Return and Reselect schedule maintenance process
Operational Manger
10.2 Updateble schedule details edit
P9
10.2.2 Display Details of schedule list that can be Updated
P15
ALT 10.2.4 Prompt user to confirm removal
ALT 10.2 Request confrimation of schedule removal
Operational Manger
10.2 Updateed schedule details
ALT - 10.2 Removal schedule confirmation
P10
10.2.3 Read updated schedule details
10.2 Updated schedule details
ALT - 12.2 Remove schedule selected
P16
ALT 10.2.5 Read confirmation of schedule removal
ALT 10.2 Requested- Confirmation of schedule removal
Operational Manger
10.2 Unvalidated update schedule details
ALT 10.2 Requested Confirmation of schedule removal
P11
10.2.4 Validate format of the updated schedule details
ALT 10.2 Erroreous updated schedule details
10.2 Maintenance Details format
P17
ALT 10.2.6 Remove Maintenance from schedule
10.2 Validated updated schedule details
P12
10.2.5 Store updated schedule details
10.2 Updatre schedule details
D
Slot Table
ALT 10.2 Remove Maintenance
10.2 Updated schedule details to be displayed
P13
10.2.6 Display updated schedule details
10.2 Displayed Updated schedule details
ALT - 10.2 Continue and save update to schedule
Operational Manger
10.2 Releevant updated schedule details
P14
10.2.7 Notify manager of seccessful update
10.2 Notification of seccuessfule update

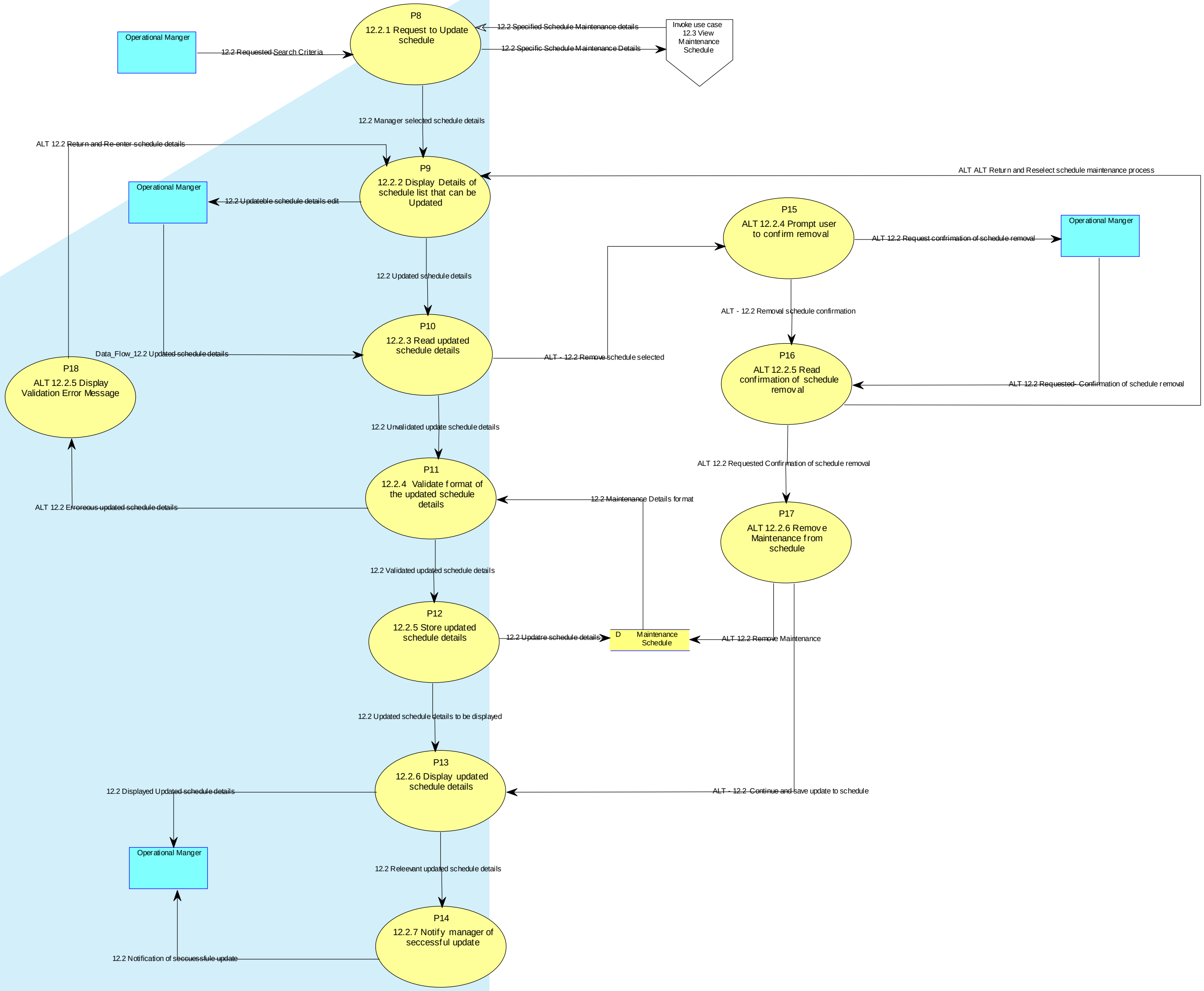

Booking Consultant
10.3 Request top view schedule
P1
10.3.1 Request to view schedule
ALT - Return and Re-enter Schedule Search Criteria
10.3 User enetered criteria
P7
10.3.7 Read user entered Search criteria
P8
ALT 10.3.3 Display Schedule Search Criteria Error Message
ALT - Invalid Schedule Search Criteria Error Message
Booking Consultant
10.3 Schedule Search criteria
Booking Consultant
10.3 Selects date on the schedule
P2
10.3.2 Perform schedule search
ALT - Invalid Schedule Search Criteria
D
Schedule Table
10.3 List of all bookings scheduled
10.3 Selected date option
P3
10.3.3 Display list of all bookings from the chosen date
10.3 List of all bookings on chosen date
Booking Consultant
10.3 List of all bookings
10.3 Selected Booking on specific date
P4
10.3.4 Display chosen booking details
10.3 Schedule search selection
P5
10.3 Read User Selected Search Result
10.3 Selected schedule details
P6
10.3 Display User Seleected Search Result Details
10.3 Display specific Search Result

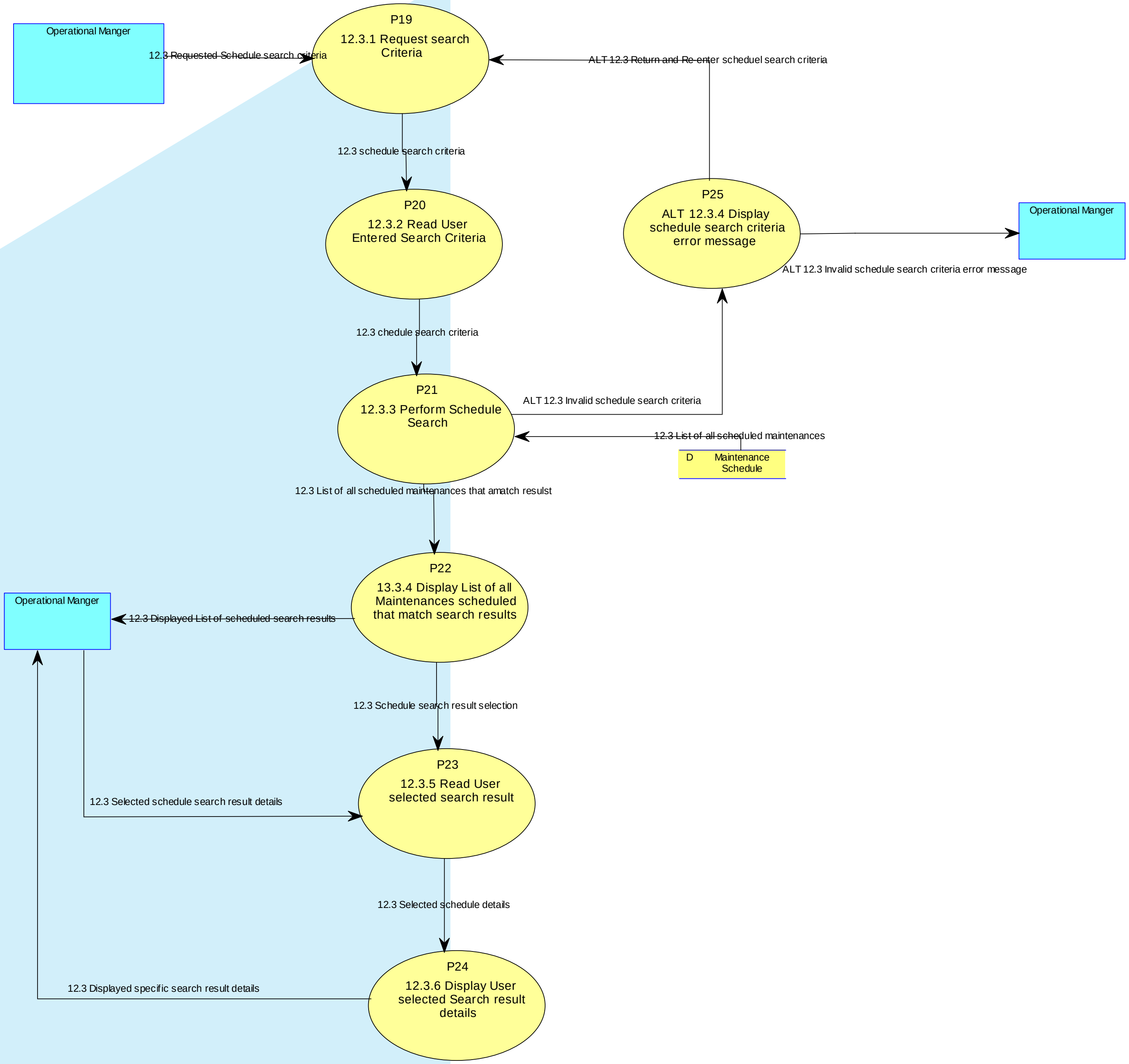

Operational Manager
11.1 Requested Report Criteria Details
P1
11.1.1 Request Booking report
ALT 11.1 Return and Re-eneter Report Criteria details
P6
ALT 11.1.4 Display invalid booking details error message
ALT 11.1 No mathes error message
Operational Manager
11.1 Booking details feilds
P2
11.1.2 Read Entered Report Criteria Details
ALT 11.1 Invalid details format
11.1 Captured booking details
P3
11.1.3 Generate Report of the entered criteria details
D
Slot Table
11.1 Booking details format
11.1 Formatted booking details
P4
11.1.4 Display Complete Report
11.1 Report Details displayed
Operational Manager
11.1 Notification of successful booking Report Generation
P5
11.1.5 Notify Manager of generated Report

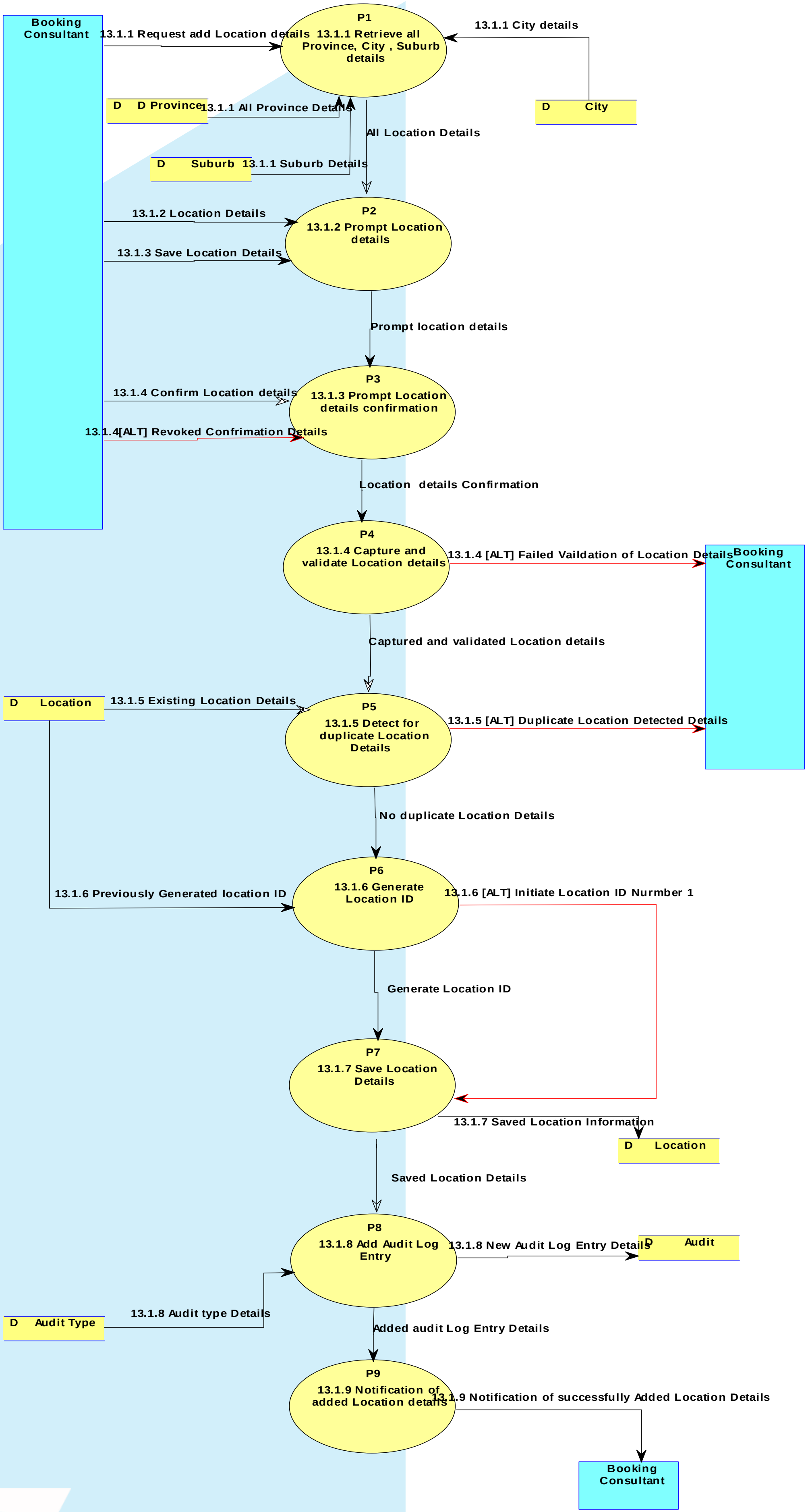

Operational Manager
11.2 Requested report details
P1
11.2.1 Display schedule generation
11.2 Schedule details
P2
11.2.2 Requests vehicle search criteria
Operational Manager
ALT 11.2 Return and Re.-enter report criteria details
P6
ALT 11.2.4 Displays invalid schedule not found error message
ALT 11.2 No matches error message
11.2 Captured schedule details
P3
11.2.3 Reads entered report criteria details
11.2 Mileage details
D
Slot Table
11.2 Details to generate report
P4
11.2.4 Generate report of the entered schedule details
ALT 11.2 Invalid schedule details
11.2 Report details displayed
P5
11.2.5 Display complete report
Operational Manager
11.2 Displays confirmation message

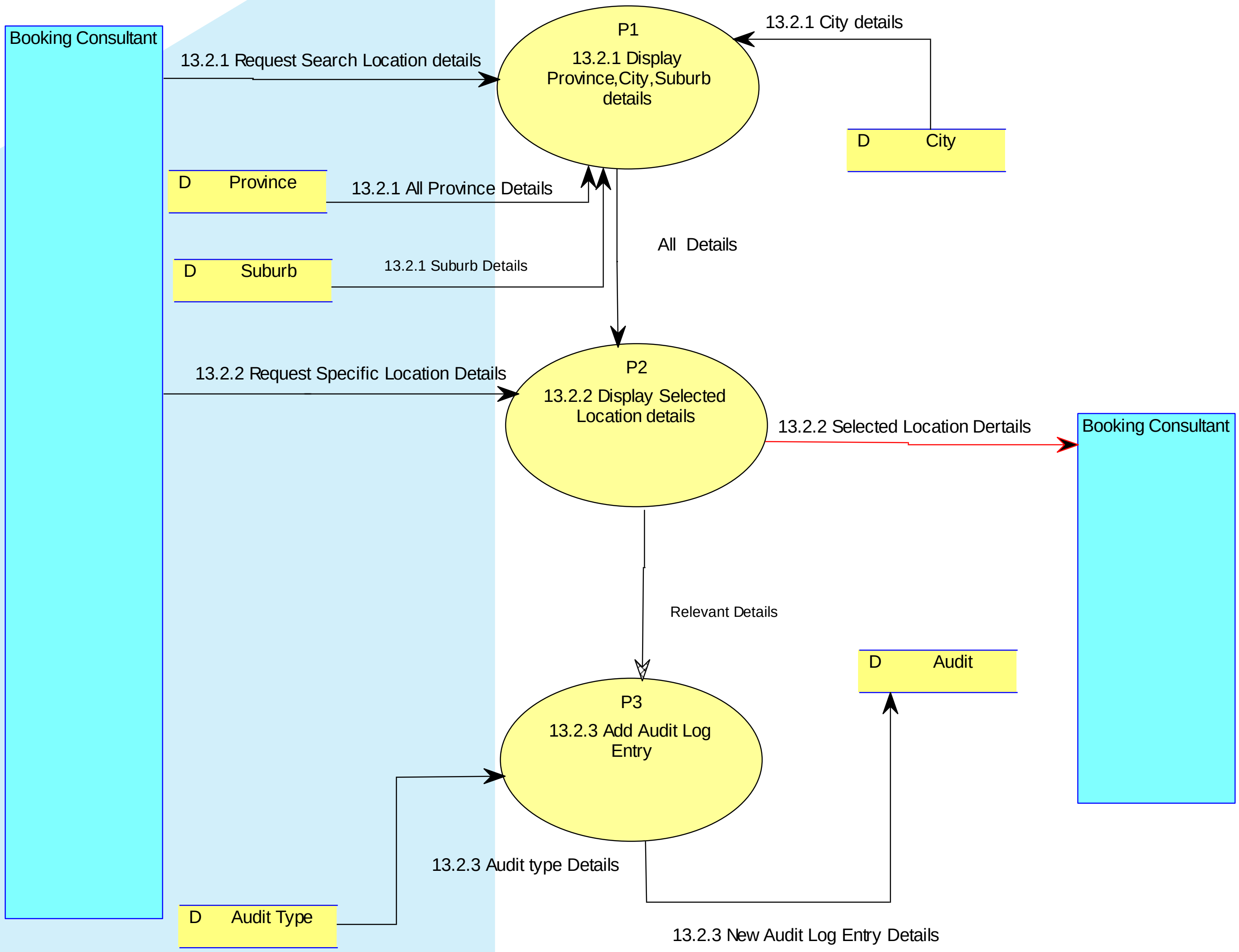

Operational Manager
11.3 Requested report details
P1
11.3.1 Display schedule generation
11.3 Schedule details
P2
11.3.2 Requests Mileage search criteria
Operational Manager
ALT 11.3 Return and Re.-enter report criteria details
P6
ALT 11.3.4 Displays invalid schedule not found error message
ALT 11.3 No matches error message
11.3 Captured schedule details
P3
11.3.3 Reads entered report criteria details
11.3 Mileage details
D
Vehicle Maintenance
11.3 Details to generate report
P4
11.3.4 Generate report of the entered schedule details
ALT 11.3 Invalid schedule details
11.3 Report details displayed
P5
11.3.5 Display complete report
Operational Manager
11.3 Displays confirmation message

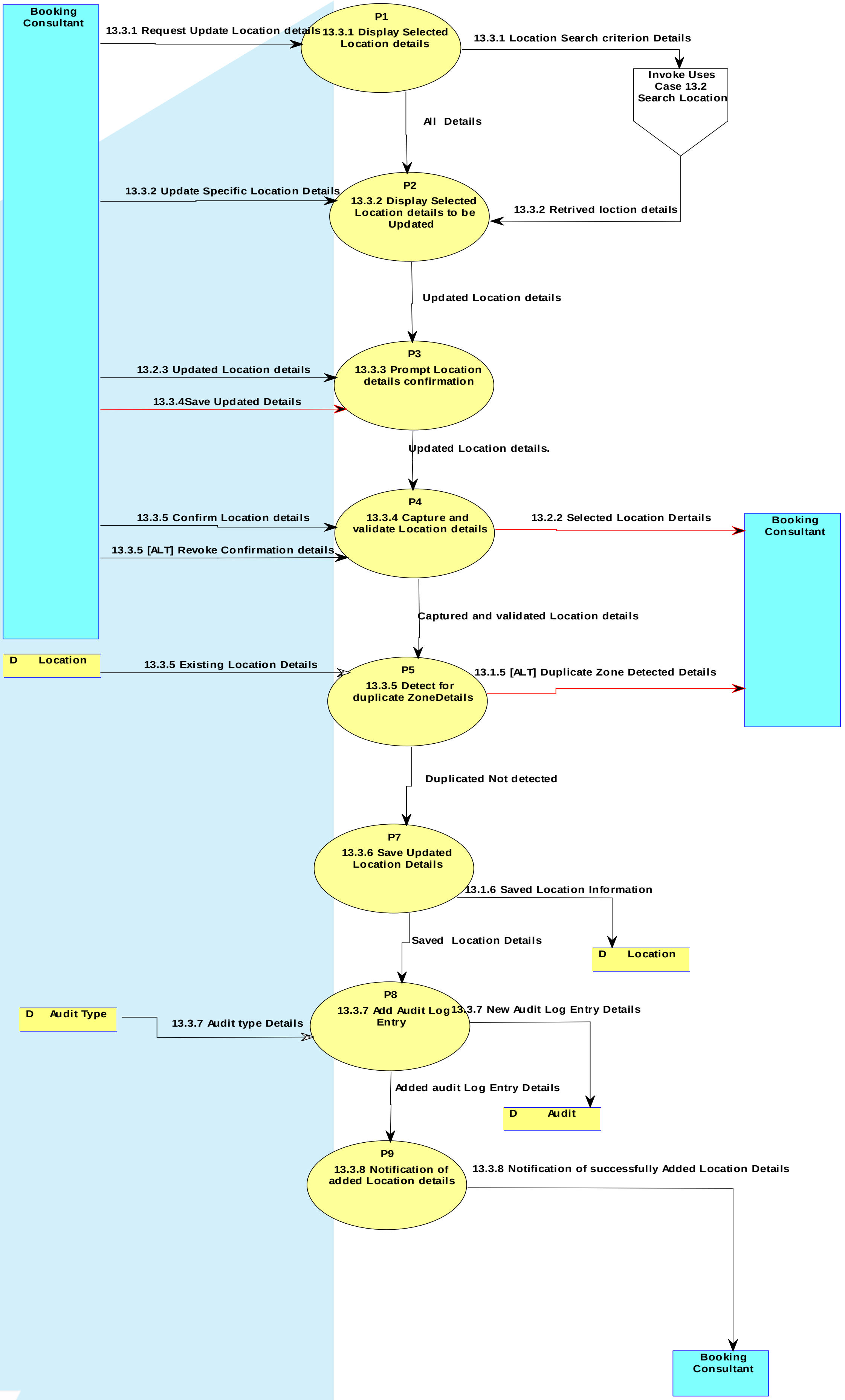

Operational Manger
12.1 Requested Maintenance
P1
12.1.1 Request Maintenance Details
ALT 12.1 Return and Re-enetr Maintenance Details
12.1 Maintenance Details
P2
12.1.2 Read Maintenance Details
P7
ALT 12.1.4 Display Invalid Input Error Message
ALT 12.1 Invalid Maintenance Details Error Message
Operational Manger
12.1 Unvalidated Maintenance Details
P3
12.1.3 Validate Format of Maintenance Details
ALT 12.1 Erroreous Maintenance Details
12.1 Maintenance Details format
12.1 Validated Maintenance Details
P4
12.1.4 Retreive Last Maintenance unique number
12.1 Last Maintenance unique number
D
Maintenance Schedule
12.1 Approved Maintenance Details
P5
12.1.5 Store Maintenance Details
12.1 New Maintenance Details
12.1 Relevant Maintenance Details
P6
12.1.6 Notify Operational Manger of scheduling
12.1 Notification of Scheduling
Operational Manger

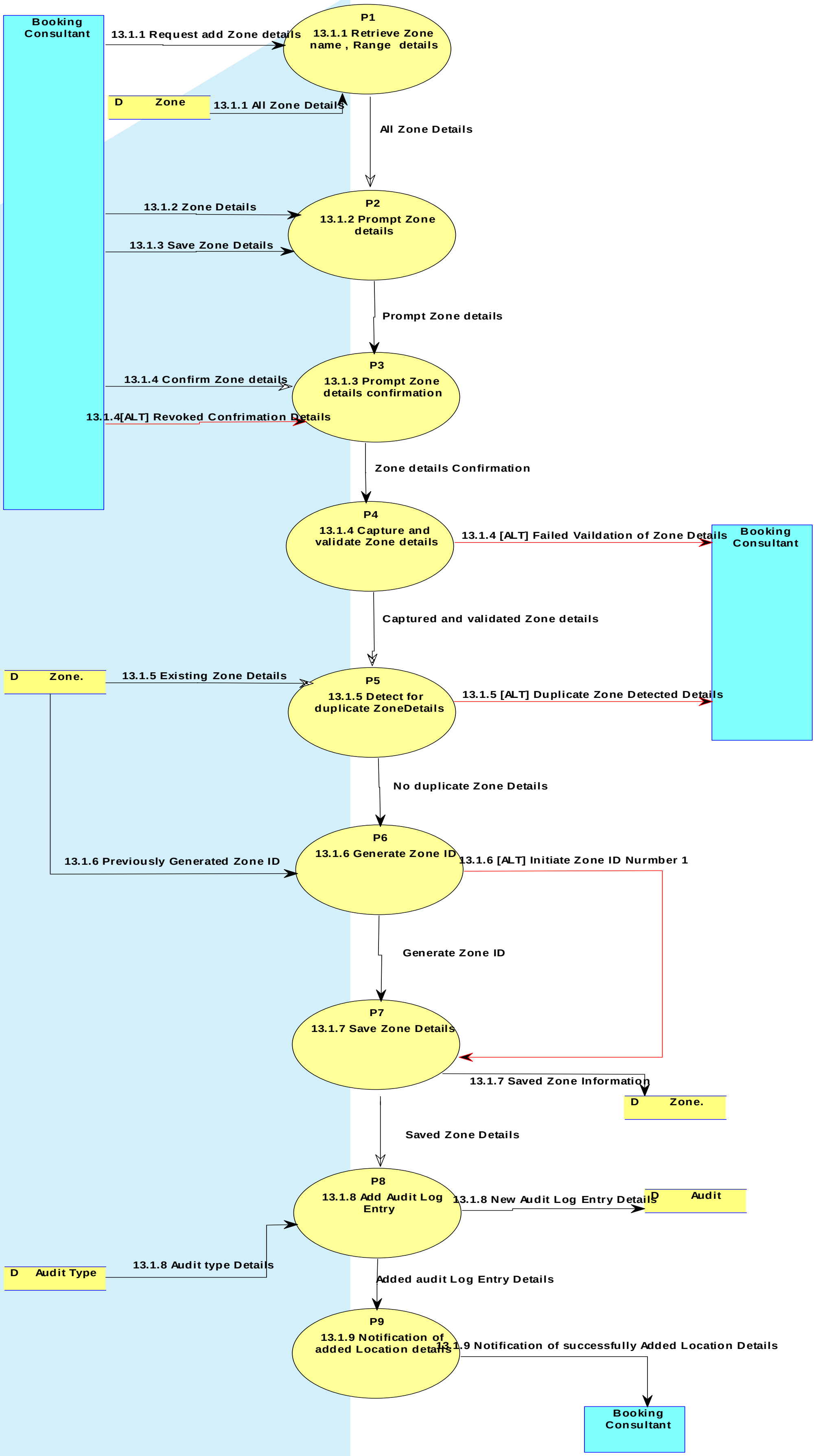

Operational Manger
12.2 Requested Search Criteria
P8
12.2.1 Request to Update schedule
12.2 Specified Schedule Maintenance details
12.2 Specific Schedule Maintenance Details
Invoke use case 12.3 View Maintenance Schedule
12.2 Manager selected schedule details
ALT 12.2 Return and Re-enter schedule details
P9
12.2.2 Display Details of schedule list that can be Updated
Operational Manger
12.2 Updateble schedule details edit
ALT ALT Return and Reselect schedule maintenance process
P15
ALT 12.2.4 Prompt user to confirm removal
ALT 12.2 Request confrimation of schedule removal
Operational Manger
12.2 Updated schedule details
ALT - 12.2 Removal schedule confirmation
P10
12.2.3 Read updated schedule details
Data_Flow_12.2 Updated schedule details
ALT - 12.2 Remove schedule selected
P16
ALT 12.2.5 Read confirmation of schedule removal
ALT 12.2 Requested- Confirmation of schedule removal
P18
ALT 12.2.5 Display Validation Error Message
12.2 Unvalidated update schedule details
ALT 12.2 Requested Confirmation of schedule removal
P11
12.2.4 Validate format of the updated schedule details
12.2 Maintenance Details format
ALT 12.2 Erroreous updated schedule details
P17
ALT 12.2.6 Remove Maintenance from schedule
12.2 Validated updated schedule details
P12
12.2.5 Store updated schedule details
12.2 Updatre schedule details
D
Maintenance Schedule
ALT 12.2 Remove Maintenance
12.2 Updated schedule details to be displayed
P13
12.2.6 Display updated schedule details
12.2 Displayed Updated schedule details
ALT - 12.2 Continue and save update to schedule
Operational Manger
12.2 Releevant updated schedule details
P14
12.2.7 Notify manager of seccessful update
12.2 Notification of seccuessfule update

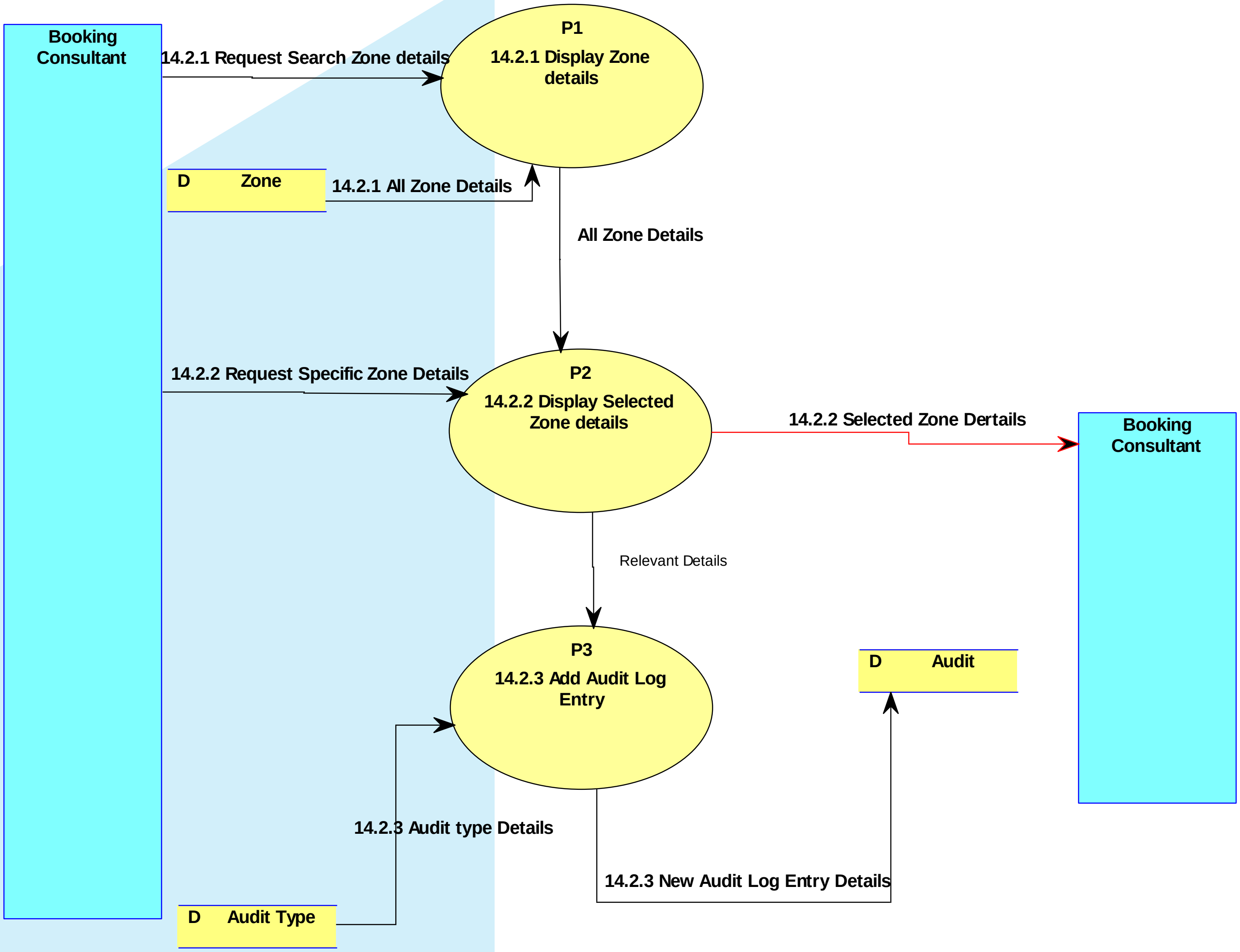

Operational Manger
12.3 Requested Schedule search criteria
P19
12.3.1 Request search Criteria
ALT 12.3 Return and Re-enter scheduel search criteria
12.3 schedule search criteria
P20
12.3.2 Read User Entered Search Criteria
P25
ALT 12.3.4 Display schedule search criteria error message
Operational Manger
ALT 12.3 Invalid schedule search criteria error message
12.3 chedule search criteria
P21
12.3.3 Perform Schedule Search
ALT 12.3 Invalid schedule search criteria
12.3 List of all scheduled maintenances
D
Maintenance Schedule
12.3 List of all scheduled maintenances that amatch resulst
P22
13.3.4 Display List of all Maintenances scheduled that match search results
Operational Manger
12.3 Displayed List of scheduled search results
12.3 Schedule search result selection
P23
12.3.5 Read User selected search result
12.3 Selected schedule search result details
12.3 Selected schedule details
P24
12.3.6 Display User selected Search result details
12.3 Displayed specific search result details

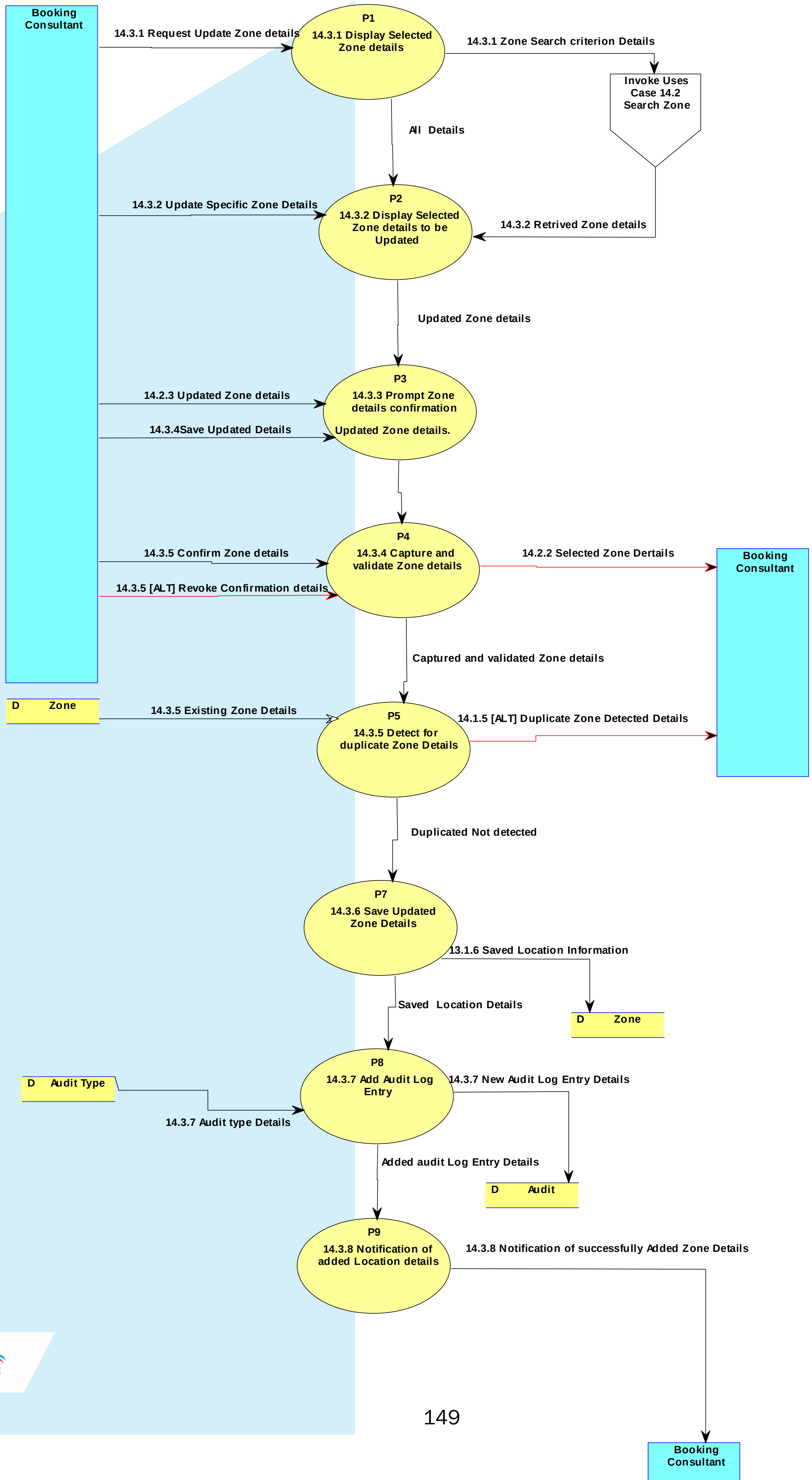

Booking Consultant
13.1.1 Request add Location details
P1
13.1.1 Retrieve all Province, City , Suburb details
13.1.1 City details
D City
D D Province
13.1.1 All Province Details
D Suburb
13.1.1 Suburb Details
All Location Details
13.1.2 Location Details
13.1.3 Save Location Details
P2
13.1.2 Prompt Location details
Prompt location details
13.1.4 Confirm Location details
13.1.4[ALT] Revoked Confrimation Details
P3
13.1.3 Prompt Location details confirmation
Location details Confirmation
P4
13.1.4 Capture and validate Location details
13.1.4 [ALT] Failed Vaildation of Location Details
Booking Consultant
Captured and validated Location details
D Location
13.1.5 Existing Location Details
P5
13.1.5 Detect for duplicate Location Details
13.1.5 [ALT] Duplicate Location Detected Details
No duplicate Location Details
13.1.6 Previously Generated location ID
P6
13.1.6 Generate Location ID
13.1.6 [ALT] Initiate Location ID Nurmber 1
Generate Location ID
P7
13.1.7 Save Location Details
13.1.7 Saved Location Information
D Location
Saved Location Details
P8
13.1.8 Add Audit Log Entry
13.1.8 New Audit Log Entry Details
D Audit
D Audit Type
13.1.8 Audit type Details
Added audit Log Entry Details
P9
13.1.9 Notification of added Location details
13.1.9 Notification of successfully Added Location Details
Booking Consultant

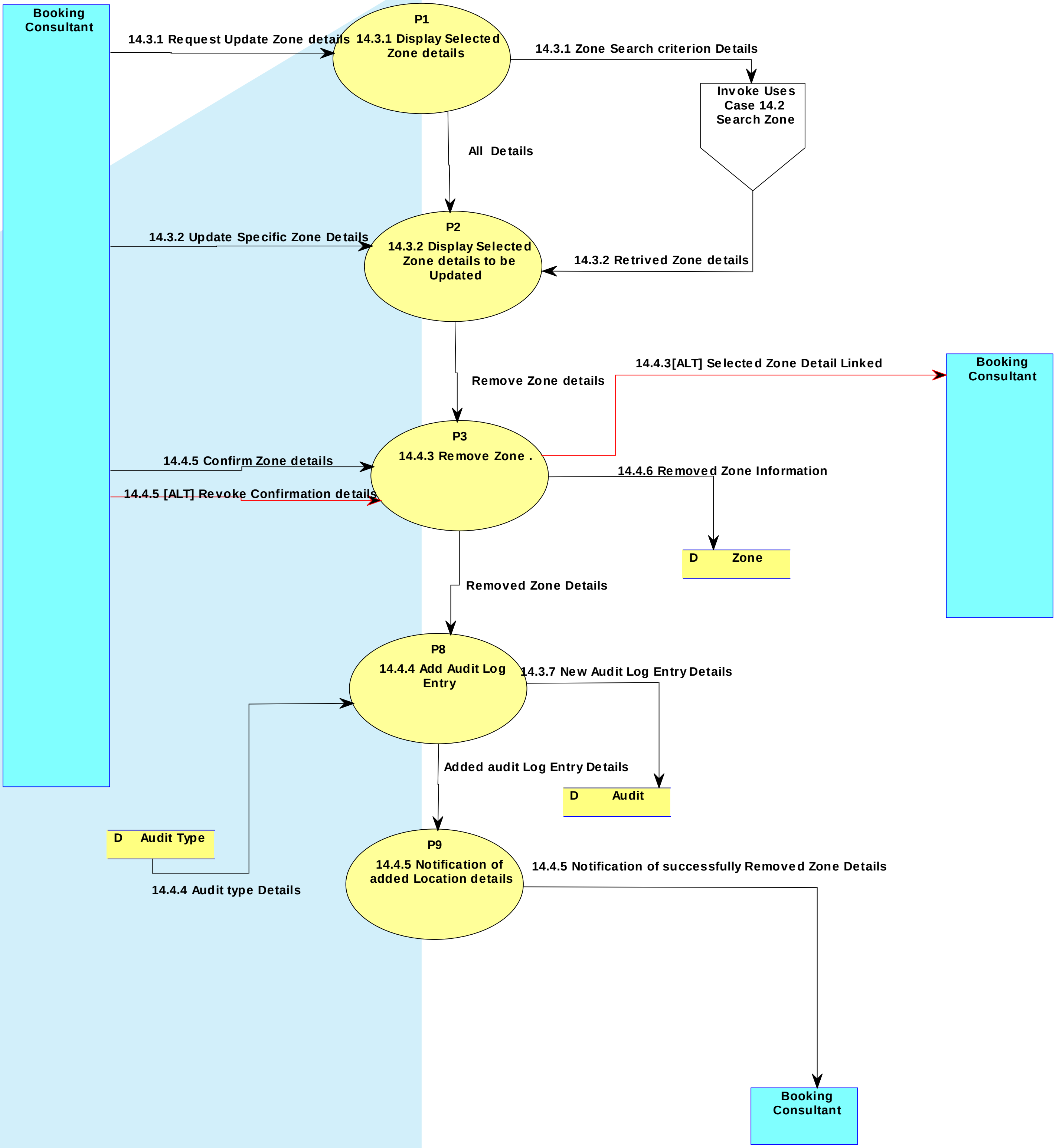

Booking Consultant
13.2.1 Request Search Location details
P1
13.2.1 Display Province,City,Suburb details
13.2.1 City details
D City
D Province
13.2.1 All Province Details
D Suburb
13.2.1 Suburb Details
All Details
13.2.2 Request Specific Location Details
P2
13.2.2 Display Selected Location details
13.2.2 Selected Location Dertails
Booking Consultant
Relevant Details
D Audit
P3
13.2.3 Add Audit Log Entry
13.2.3 Audit type Details
D Audit Type
13.2.3 New Audit Log Entry Details

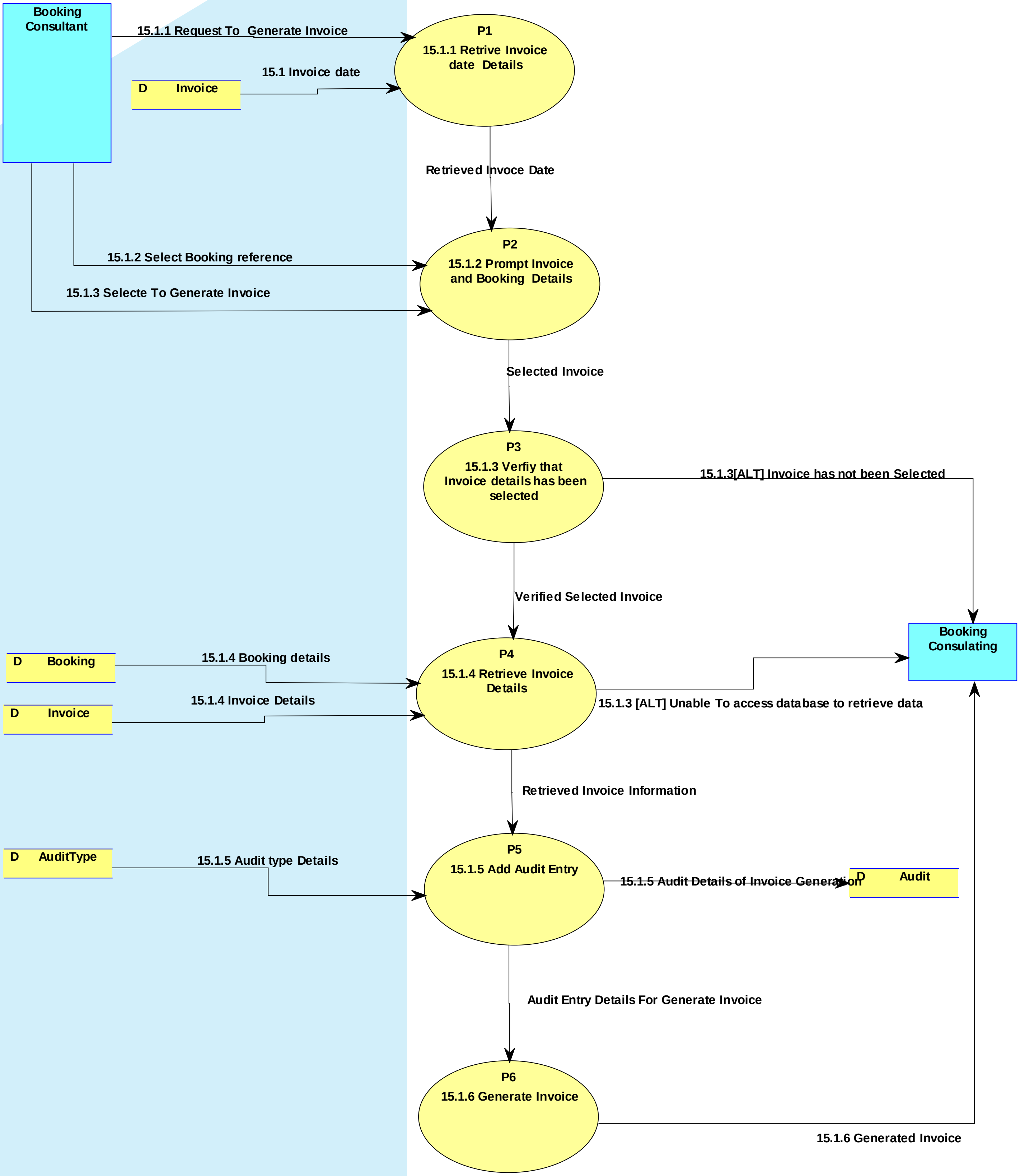

Booking Consultant
13.3.1 Request Update Location details
P1
13.3.1 Display Selected Location details
13.3.1 Location Search criterion Details
Invoke Uses Case 13.2 Search Location
All Details
13.3.2 Update Specific Location Details
P2
13.3.2 Display Selected Location details to be Updated
13.3.2 Retrived loction details
Updated Location details
13.2.3 Updated Location details
13.3.4Save Updated Details
P3
13.3.3 Prompt Location details confirmation
Updated Location details.
13.3.5 Confirm Location details
13.3.5 [ALT] Revoke Confirmation details
P4
13.3.4 Capture and validate Location details
13.2.2 Selected Location Dertails
Booking Consultant
Captured and validated Location details
D Location
13.3.5 Existing Location Details
P5
13.3.5 Detect for duplicate ZoneDetails
13.1.5 [ALT] Duplicate Zone Detected Details
Duplicated Not detected
P7
13.3.6 Save Updated Location Details
13.1.6 Saved Location Information
Saved Location Details
D Location
D Audit Type
13.3.7 Audit type Details
P8
13.3.7 Add Audit Log Entry
13.3.7 New Audit Log Entry Details
Added audit Log Entry Details
D Audit
P9
13.3.8 Notification of added Location details
13.3.8 Notification of successfully Added Location Details
Booking Consultant


Booking Consultant
13.1.1 Request add Zone details
P1
13.1.1 Retrieve Zone name , Range details
D Zone
13.1.1 All Zone Details
All Zone Details
13.1.2 Zone Details
P2
13.1.2 Prompt Zone details
13.1.3 Save Zone Details
Prompt Zone details
13.1.4 Confirm Zone details
P3
13.1.3 Prompt Zone details confirmation
13.1.4[ALT] Revoked Confrimation Details
Zone details Confirmation
P4
13.1.4 Capture and validate Zone details
13.1.4 [ALT] Failed Vaildation of Zone Details
Booking Consultant
Captured and validated Zone details
D Zone.
13.1.5 Existing Zone Details
P5
13.1.5 Detect for duplicate ZoneDetails
13.1.5 [ALT] Duplicate Zone Detected Details
No duplicate Zone Details
13.1.6 Previously Generated Zone ID
P6
13.1.6 Generate Zone ID
13.1.6 [ALT] Initiate Zone ID Nurmber 1
Generate Zone ID
P7
13.1.7 Save Zone Details
13.1.7 Saved Zone Information
D Zone.
Saved Zone Details
P8
13.1.8 Add Audit Log Entry
13.1.8 New Audit Log Entry Details
D Audit
D Audit Type
13.1.8 Audit type Details
Added audit Log Entry Details
P9
13.1.9 Notification of added Location details
13.1.9 Notification of successfully Added Location Details
Booking Consultant


Booking Consultant
14.2.1 Request Search Zone details
P1
14.2.1 Display Zone details
D Zone
14.2.1 All Zone Details
All Zone Details
14.2.2 Request Specific Zone Details
P2
14.2.2 Display Selected Zone details
14.2.2 Selected Zone Dertails
Booking Consultant
Relevant Details
P3
14.2.3 Add Audit Log Entry
D Audit
14.2.3 Audit type Details
14.2.3 New Audit Log Entry Details
D Audit Type


Booking Consultant
14.3.1 Request Update Zone details
P1
14.3.1 Display Selected Zone details
14.3.1 Zone Search criterion Details
Invoke Uses Case 14.2 Search Zone
All Details
14.3.2 Update Specific Zone Details
P2
14.3.2 Display Selected Zone details to be Updated
14.3.2 Retrived Zone details
Updated Zone details
14.2.3 Updated Zone details
P3
14.3.3 Prompt Zone details confirmation
14.3.4Save Updated Details
Updated Zone details.
14.3.5 Confirm Zone details
P4
14.3.4 Capture and validate Zone details
14.2.2 Selected Zone Dertails
Booking Consultant
14.3.5 [ALT] Revoke Confirmation details
Captured and validated Zone details
D Zone
14.3.5 Existing Zone Details
P5
14.3.5 Detect for duplicate Zone Details
14.1.5 [ALT] Duplicate Zone Detected Details
Duplicated Not detected
P7
14.3.6 Save Updated Zone Details
13.1.6 Saved Location Information
Saved Location Details
D Zone
D Audit Type
P8
14.3.7 Add Audit Log Entry
14.3.7 New Audit Log Entry Details
14.3.7 Audit type Details
Added audit Log Entry Details
D Audit
P9
14.3.8 Notification of added Location details
14.3.8 Notification of successfully Added Zone Details
Booking Consultant


Booking Consultant
14.3.1 Request Update Zone details
P1
14.3.1 Display Selected Zone details
14.3.1 Zone Search criterion Details
Invoke Uses Case 14.2 Search Zone
All Details
14.3.2 Update Specific Zone Details
P2
14.3.2 Display Selected Zone details to be Updated
14.3.2 Retrived Zone details
Remove Zone details
14.4.3[ALT] Selected Zone Detail Linked
Booking Consultant
14.4.5 Confirm Zone details
14.4.5 [ALT] Revoke Confirmation details
P3
14.4.3 Remove Zone .
14.4.6 Removed Zone Information
D Zone
Removed Zone Details
P8
14.4.4 Add Audit Log Entry
14.3.7 New Audit Log Entry Details
Added audit Log Entry Details
D Audit
D Audit Type
14.4.4 Audit type Details
P9
14.4.5 Notification of added Location details
14.4.5 Notification of successfully Removed Zone Details
Booking Consultant


Booking Consultant
15.1.1 Request To Generate Invoice
P1
15.1.1 Retrive Invoice date Details
D
Invoice
15.1 Invoice date
Retrieved Invoce Date
P2
15.1.2 Prompt Invoice and Booking Details
15.1.2 Select Booking reference
15.1.3 Selecte To Generate Invoice
Selected Invoice
P3
15.1.3 Verfiy that Invoice details has been selected
15.1.3[ALT] Invoice has not been Selected
Verified Selected Invoice
Booking Consulating
D
Booking
15.1.4 Booking details
D
Invoice
15.1.4 Invoice Details
P4
15.1.4 Retrieve Invoice Details
15.1.3 [ALT] Unable To access database to retrieve data
Retrieved Invoice Information
D
AuditType
15.1.5 Audit type Details
P5
15.1.5 Add Audit Entry
15.1.5 Audit Details of Invoice Generation
D
Audit
Audit Entry Details For Generate Invoice
P6
15.1.6 Generate Invoice
15.1.6 Generated Invoice

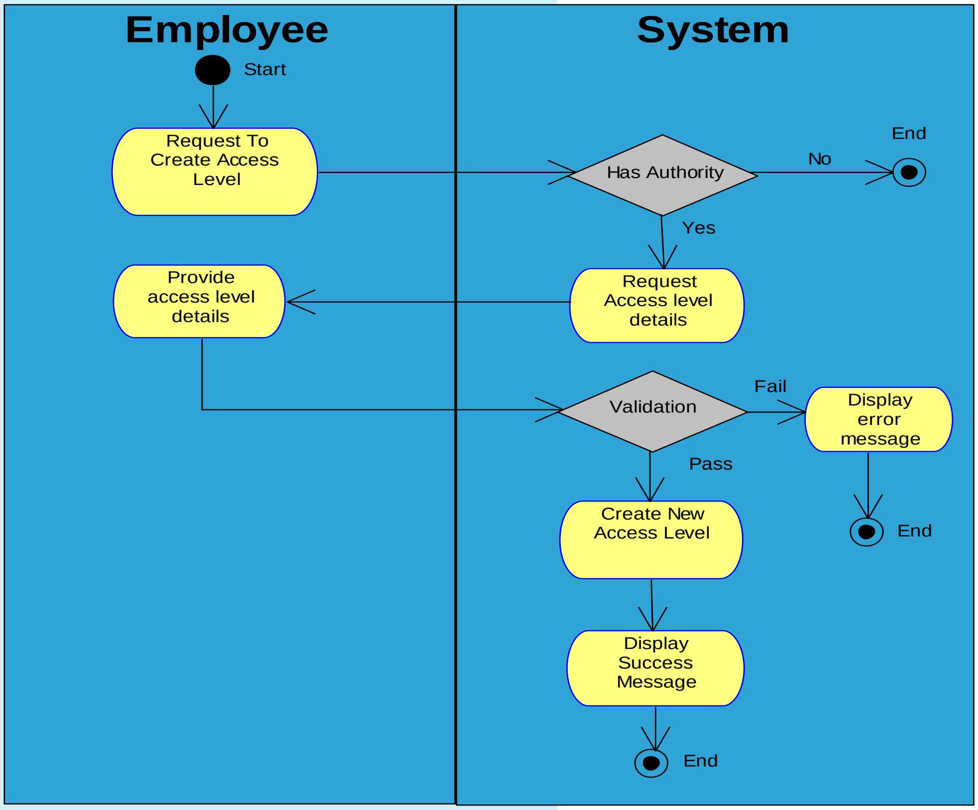
CONCLUSION

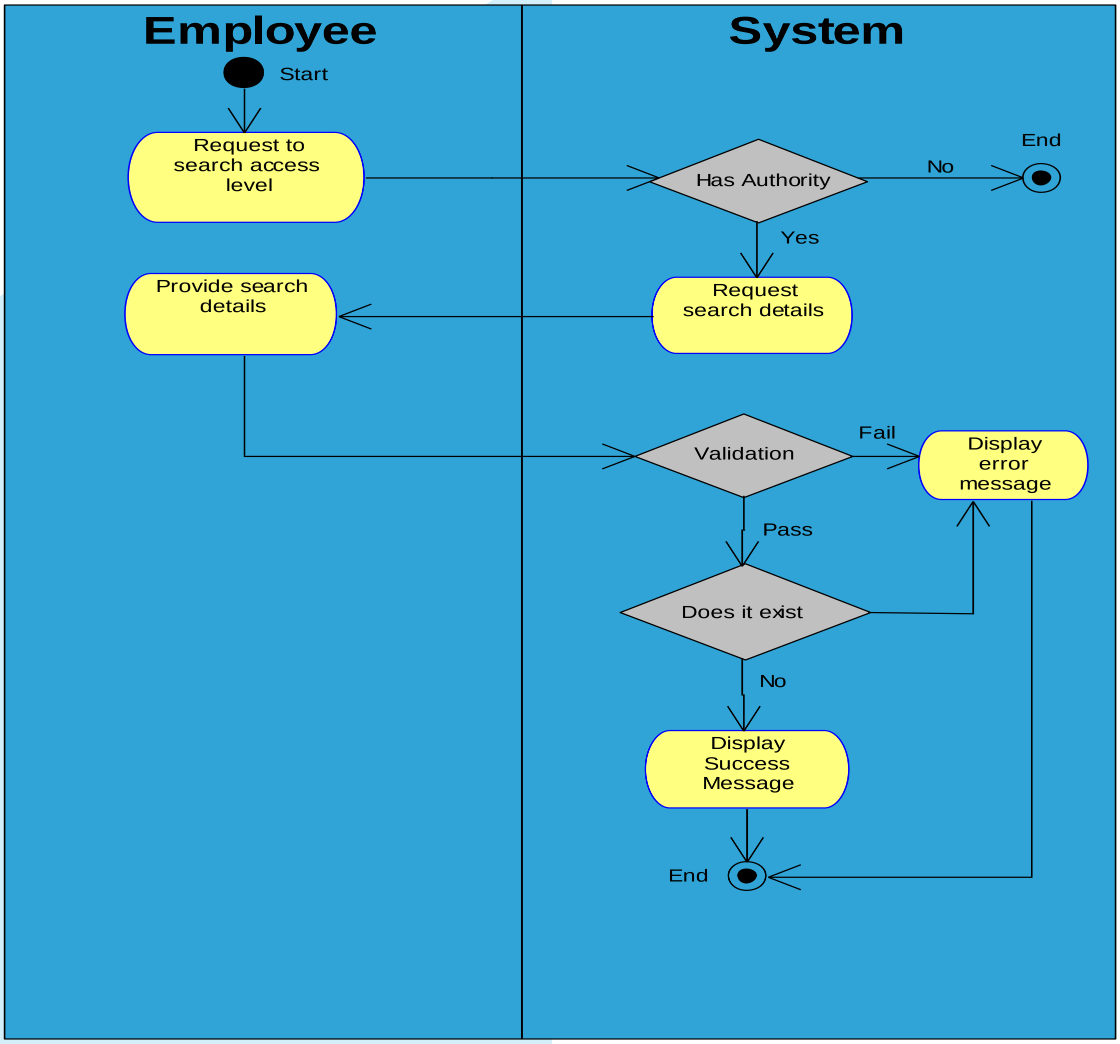
The section seen was a combination of the complete context diagram, functional decomposition diagram, data flow diagrams (high, middle and primitive level) as well as a complete Data Dictionary for the data flow diagrams. These diagrams are laid out to support the logical view of the system.

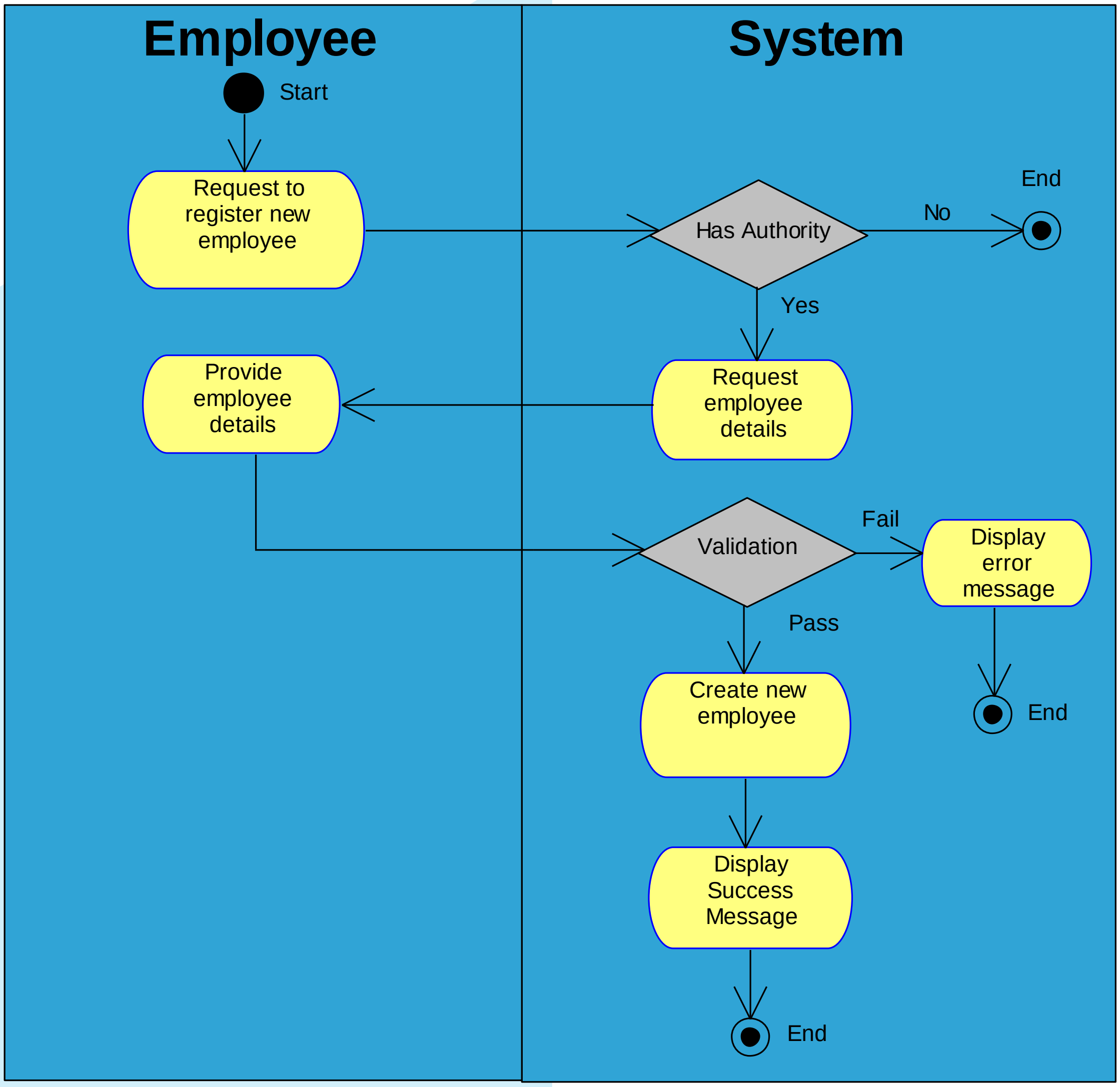
4. UML MODELLING

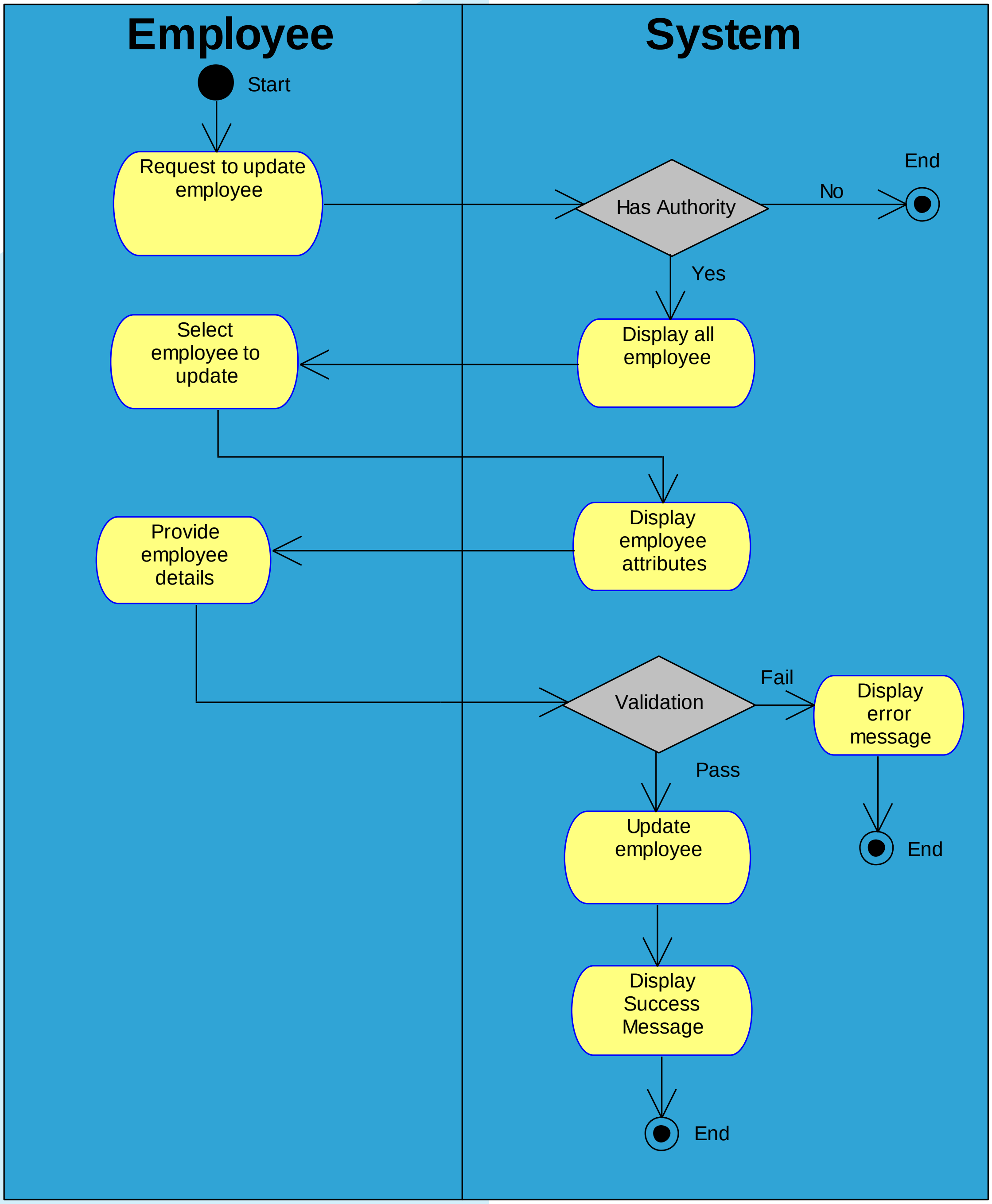
INTRODUCTION

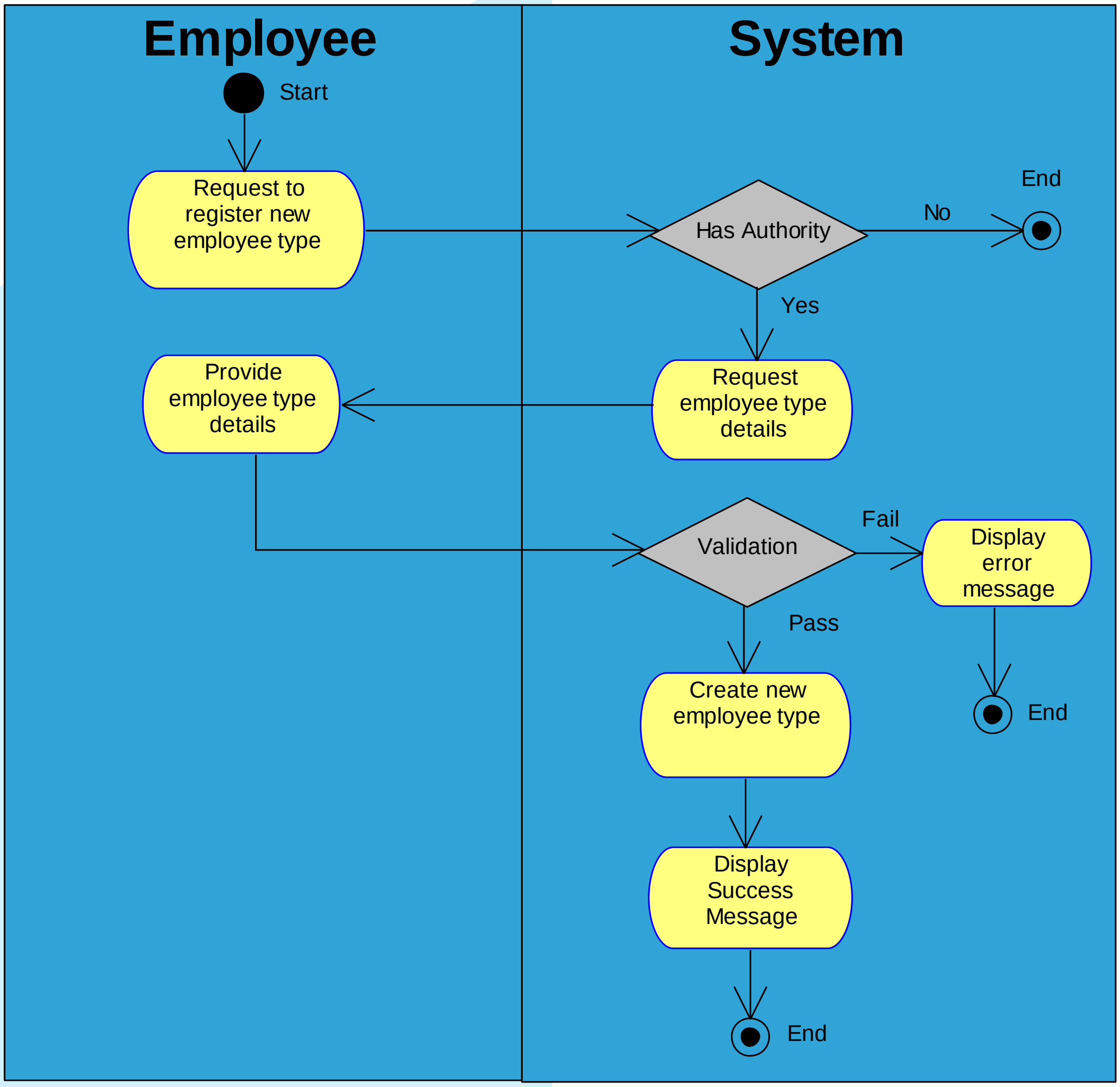
This section of the document contains UML Activity diagrams for each of our functional requirements. The activity diagrams will show the interaction between the user and the system by means of swim-lanes in each diagram.

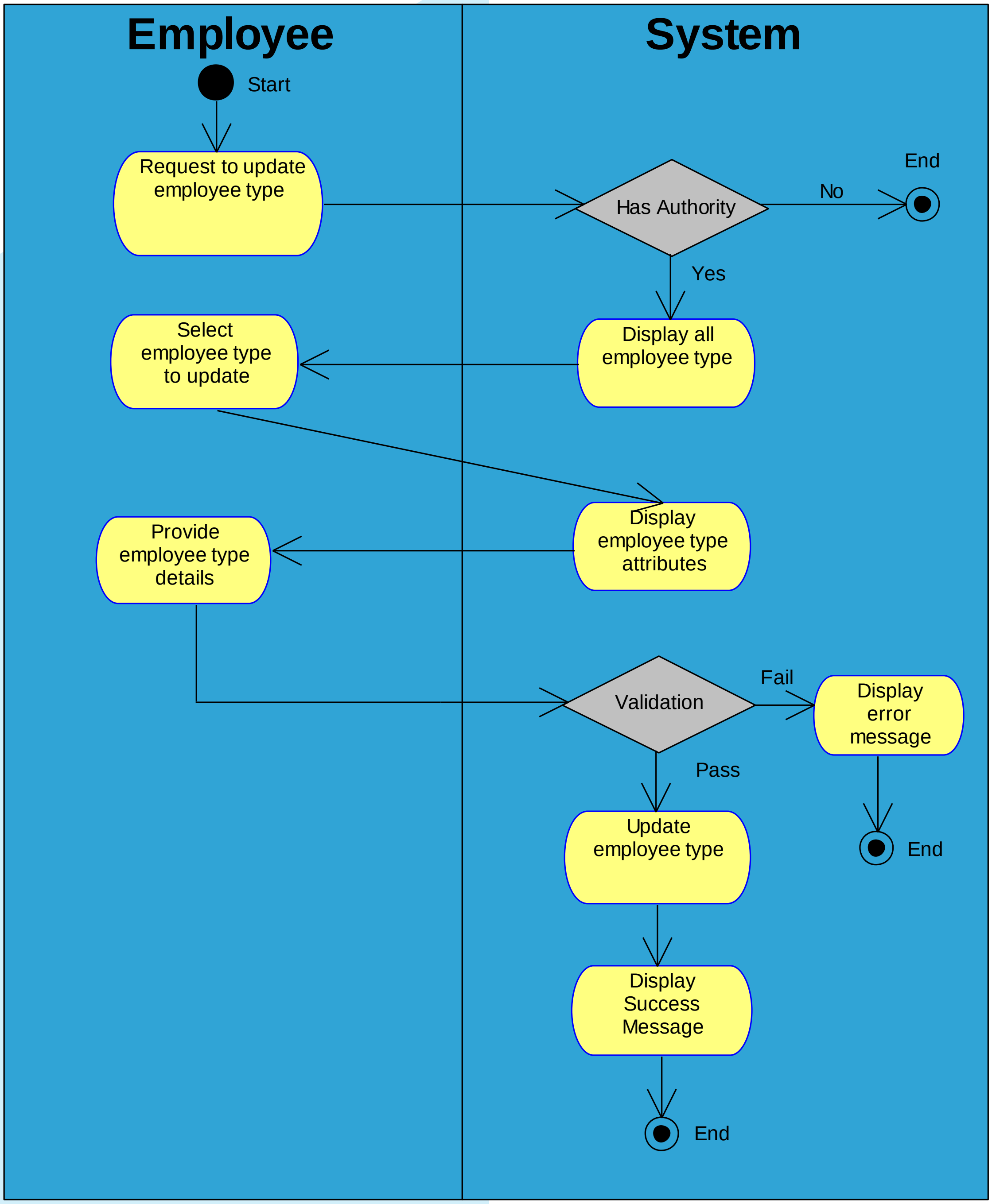


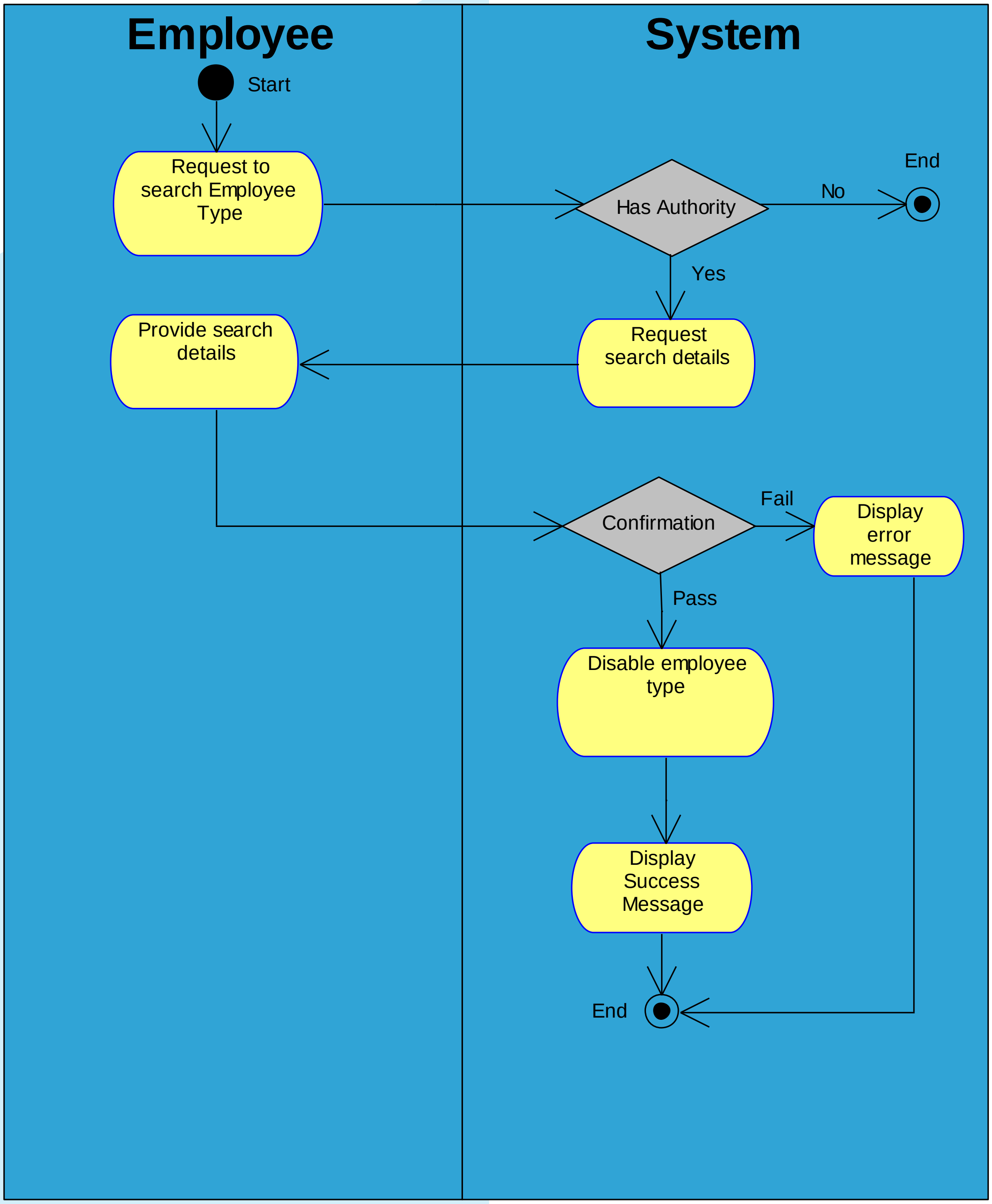

Employee
System
Start
Request to search access level
Has Authority
No
End
Yes
Request search details
Provide search details
Validation
Fail
Display error message
Pass
Does it exist
No
Display Success Message
End

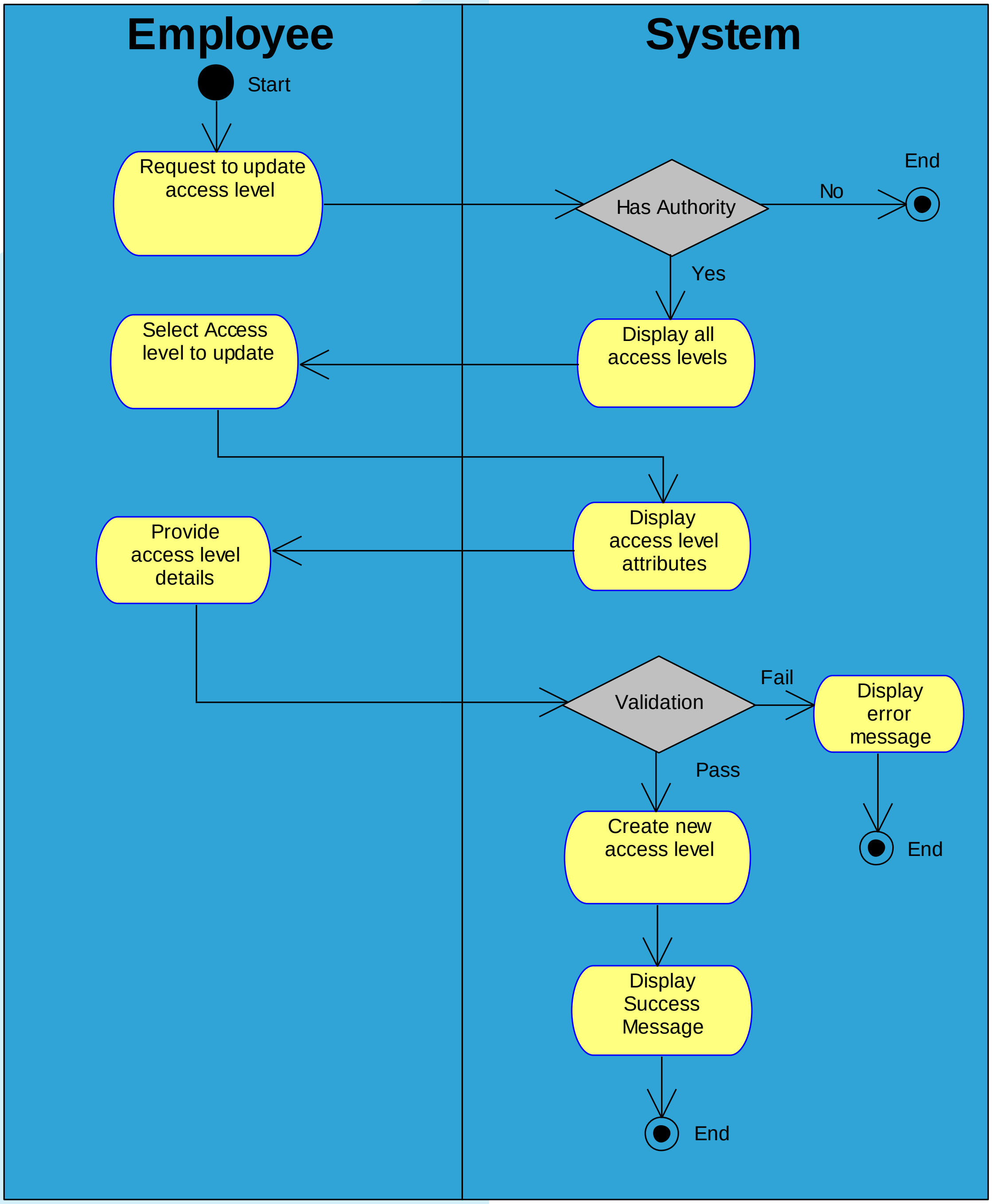

Employee
System
Start
Request to register new employee
Has Authority
No
End
Yes
Request employee details
Provide employee details
Validation
Fail
Display error message
End
Pass
Create new employee
Display Success Message
End

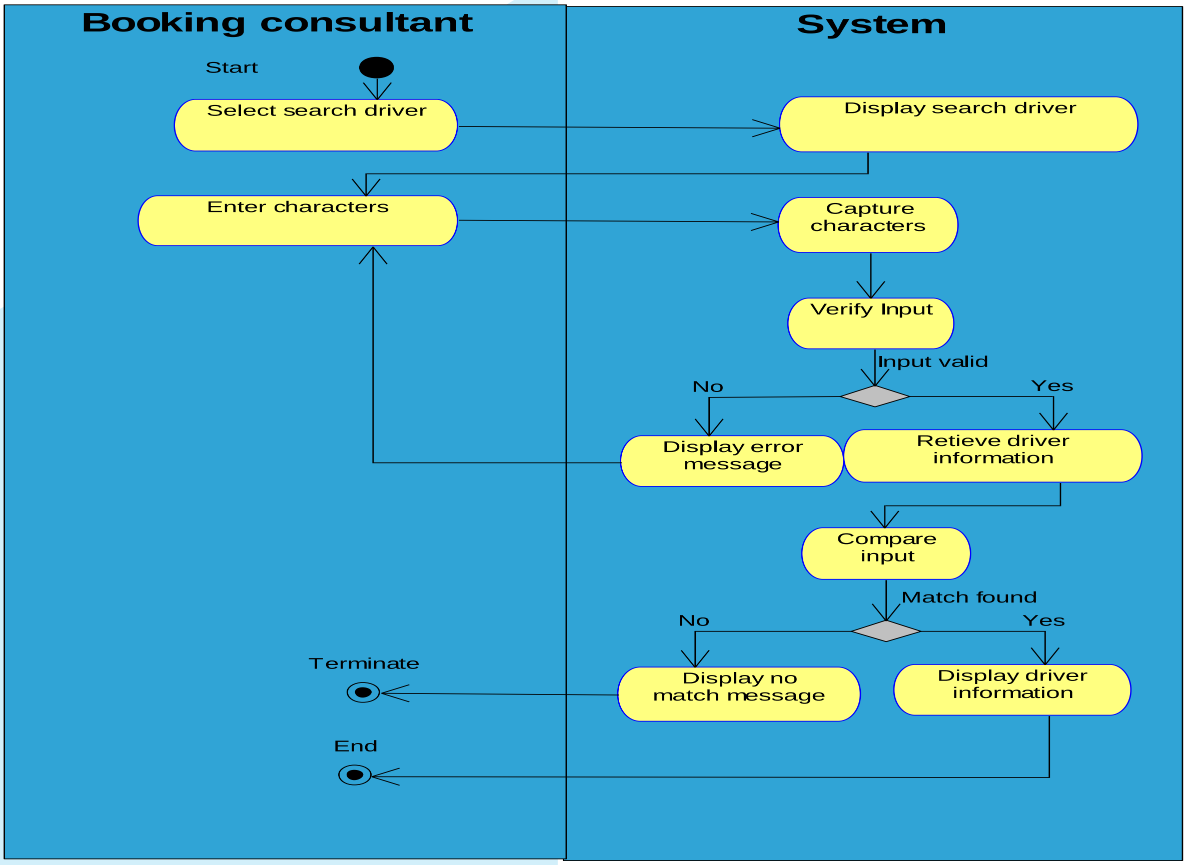

Employee
System
Start
Request to update employee
Has Authority
No
End
Yes
Select employee to update
Display all employee
Display employee attributes
Provide employee details
Validation
Fail
Display error message
Pass
Update employee
End
Display Success Message
End

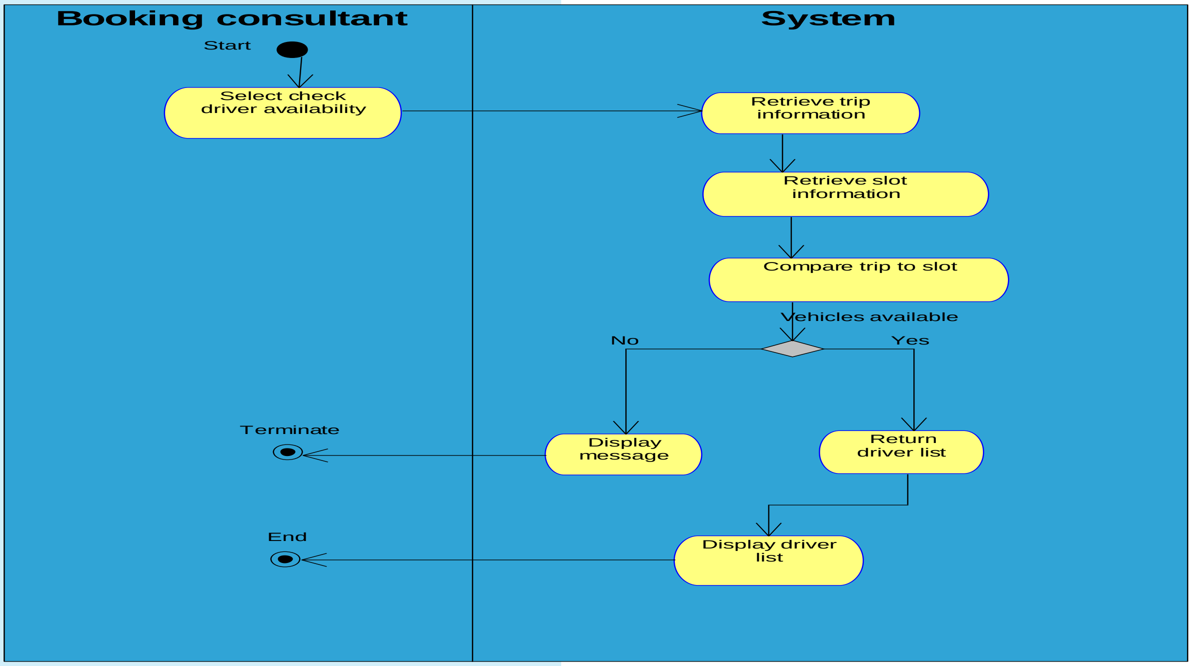

Employee
System
Start
Request to register new employee type
Has Authority
No
End
Yes
Request employee type details
Provide employee type details
Validation
Fail
Display error message
End
Pass
Create new employee type
Display Success Message
End

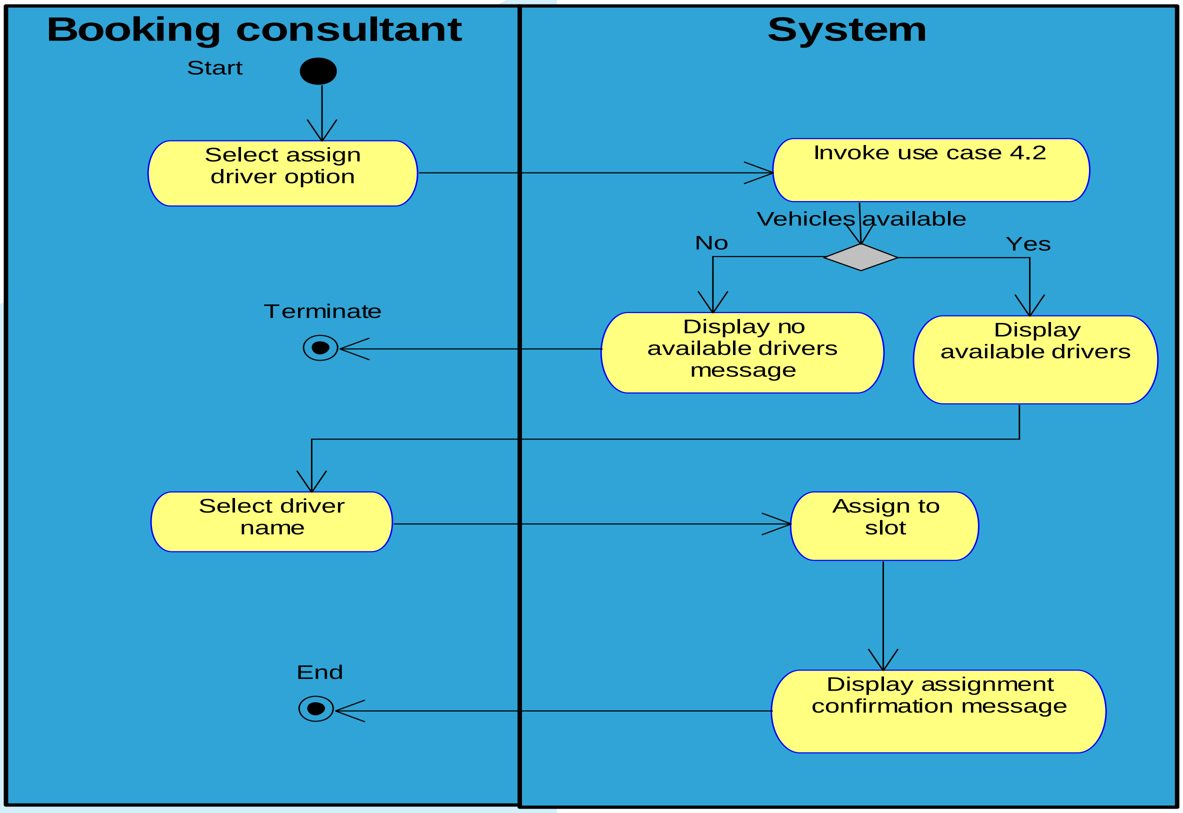

Employee
System
Start
Request to update employee type
Has Authority
No
End
Yes
Select employee type to update
Display all employee type
Display employee type attributes
Provide employee type details
Validation
Fail
Display error message
Pass
Update employee type
End
Display Success Message
End

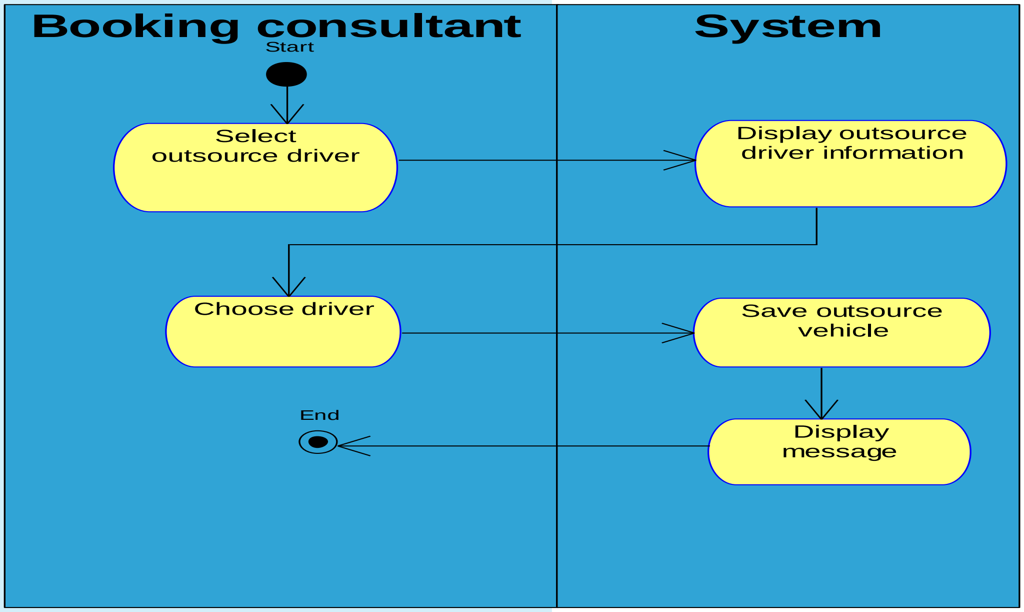

Employee
System
Start
Request to search Employee Type
Has Authority
No
End
Yes
Request search details
Provide search details
Confirmation
Fail
Display error message
Pass
Disable employee type
Display Success Message
End

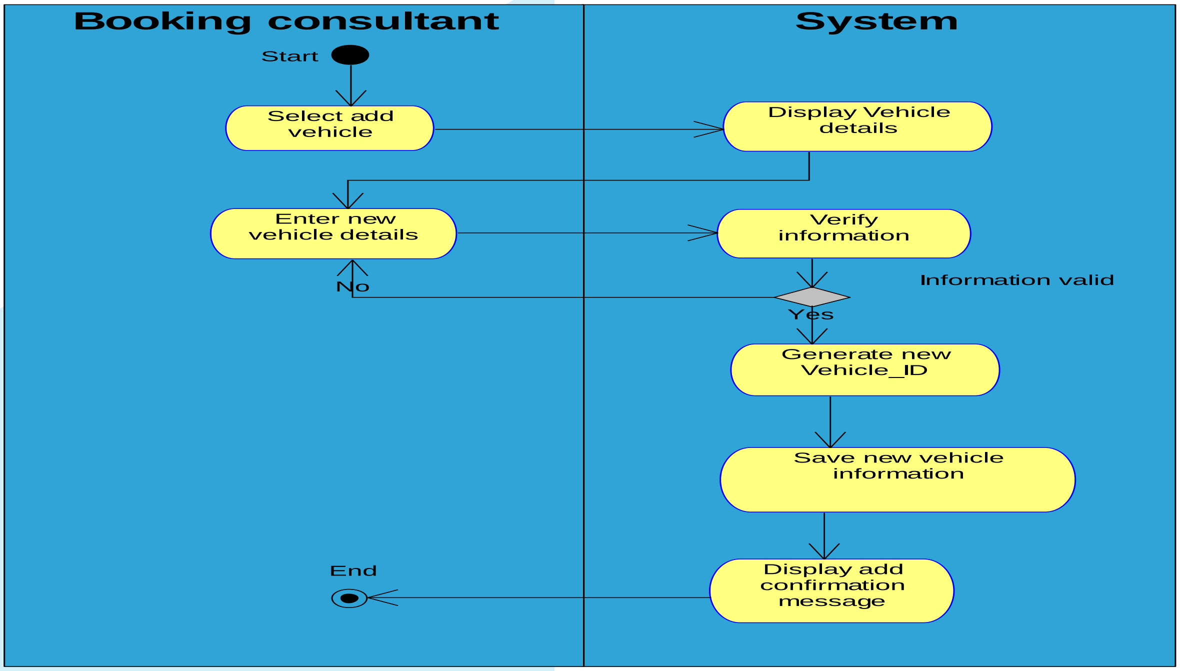

Employee
System
Start
Request to update access level
Has Authority
No
End
Yes
Display all access levels
Select Access level to update
Display access level attributes
Provide access level details
Validation
Fail
Display error message
End
Pass
Create new access level
Display Success Message
End

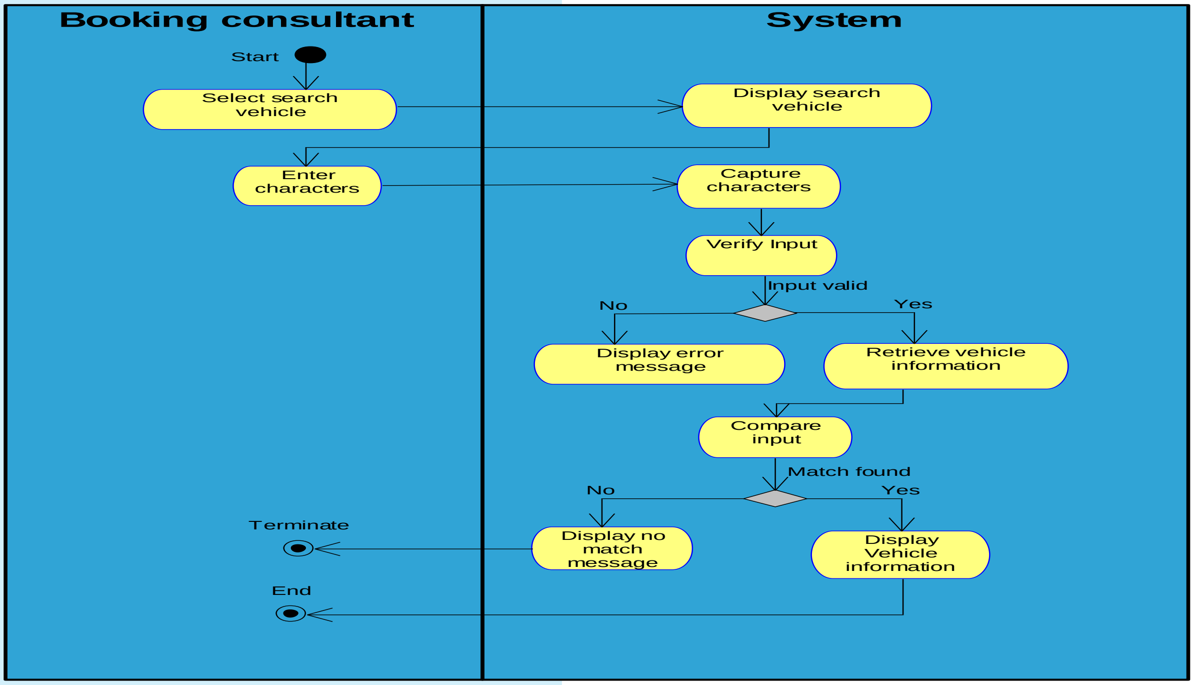

Booking consultant
System
Start
Select search driver
Display search driver
Enter characters
Capture characters
Verify Input
Input valid
No
Yes
Display error message
Retieve driver information
Compare input
Match found
No
Yes
Terminate
Display no match message
Display driver information
End

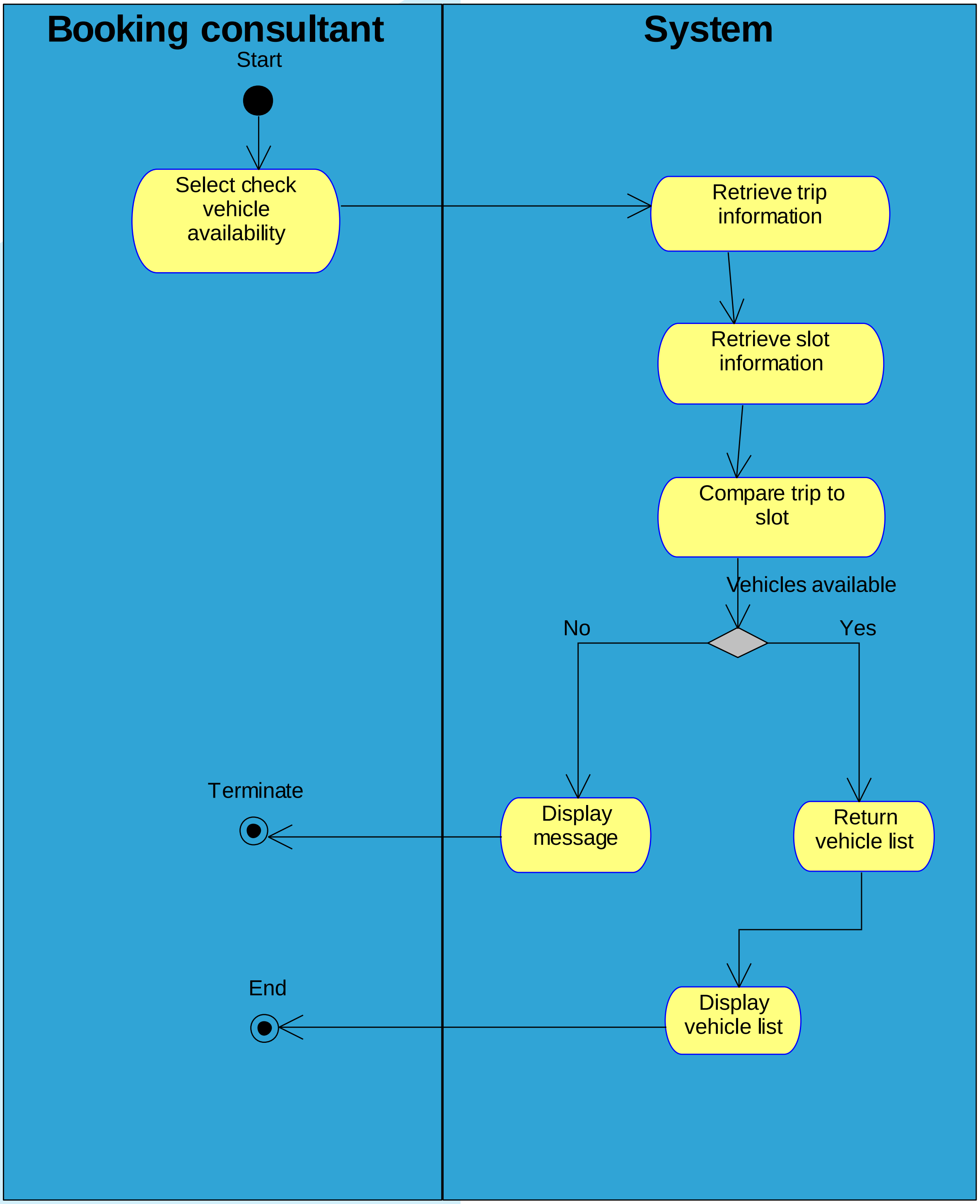

Booking consultant
System
Start
Select check driver availability
Retrieve trip information
Retrieve slot information
Compare trip to slot
Vehicles available
No
Yes
Terminate
Display message
Return driver list
End
Display driver list

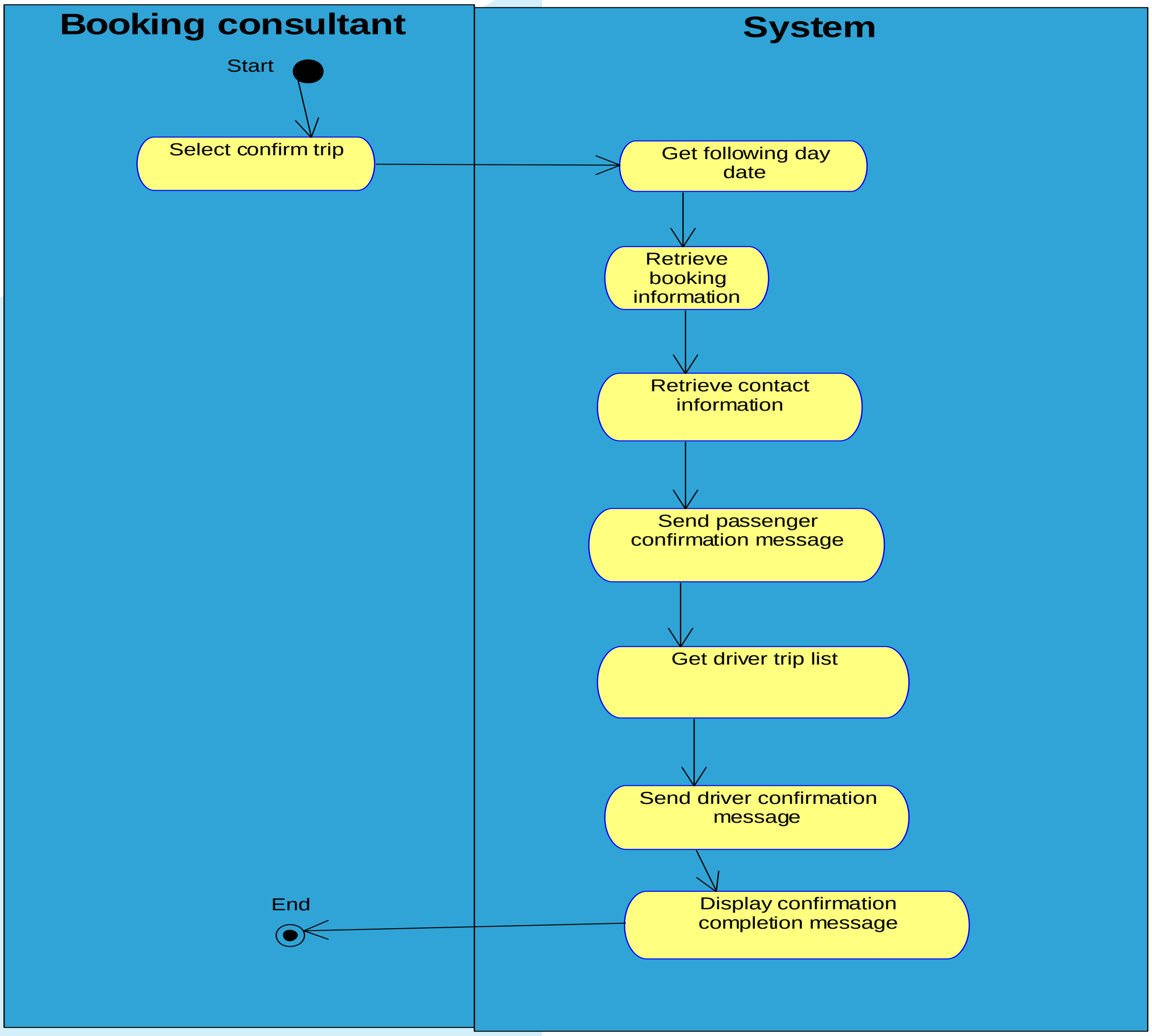

Booking consultant
System
Start
Select assign driver option
Invoke use case 4.2
Vehicles available
No
Yes
Terminate
Display no available drivers message
Display available drivers
Select driver name
Assign to slot
End
Display assignment confirmation message

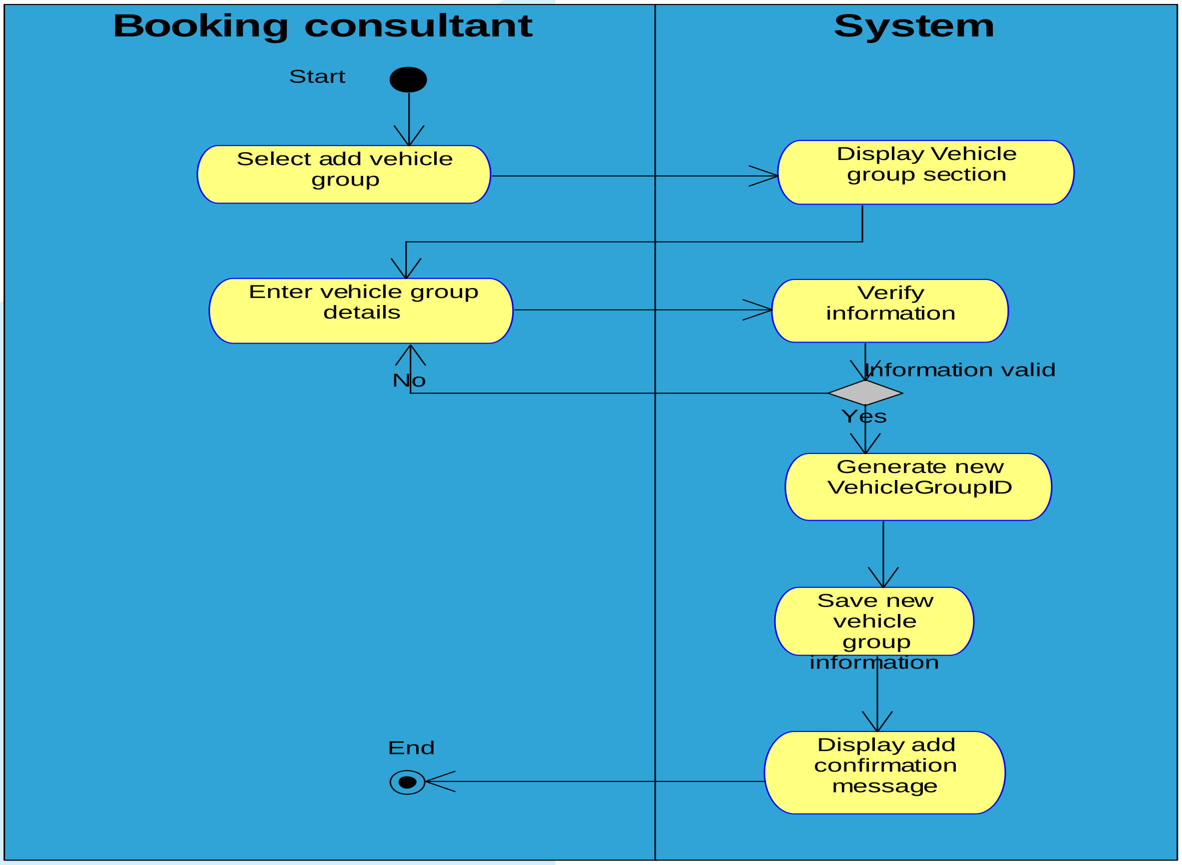

Booking consultant
System
Start
Select outsource driver
Display outsource driver information
Choose driver
Save outsource vehicle
End
Display message

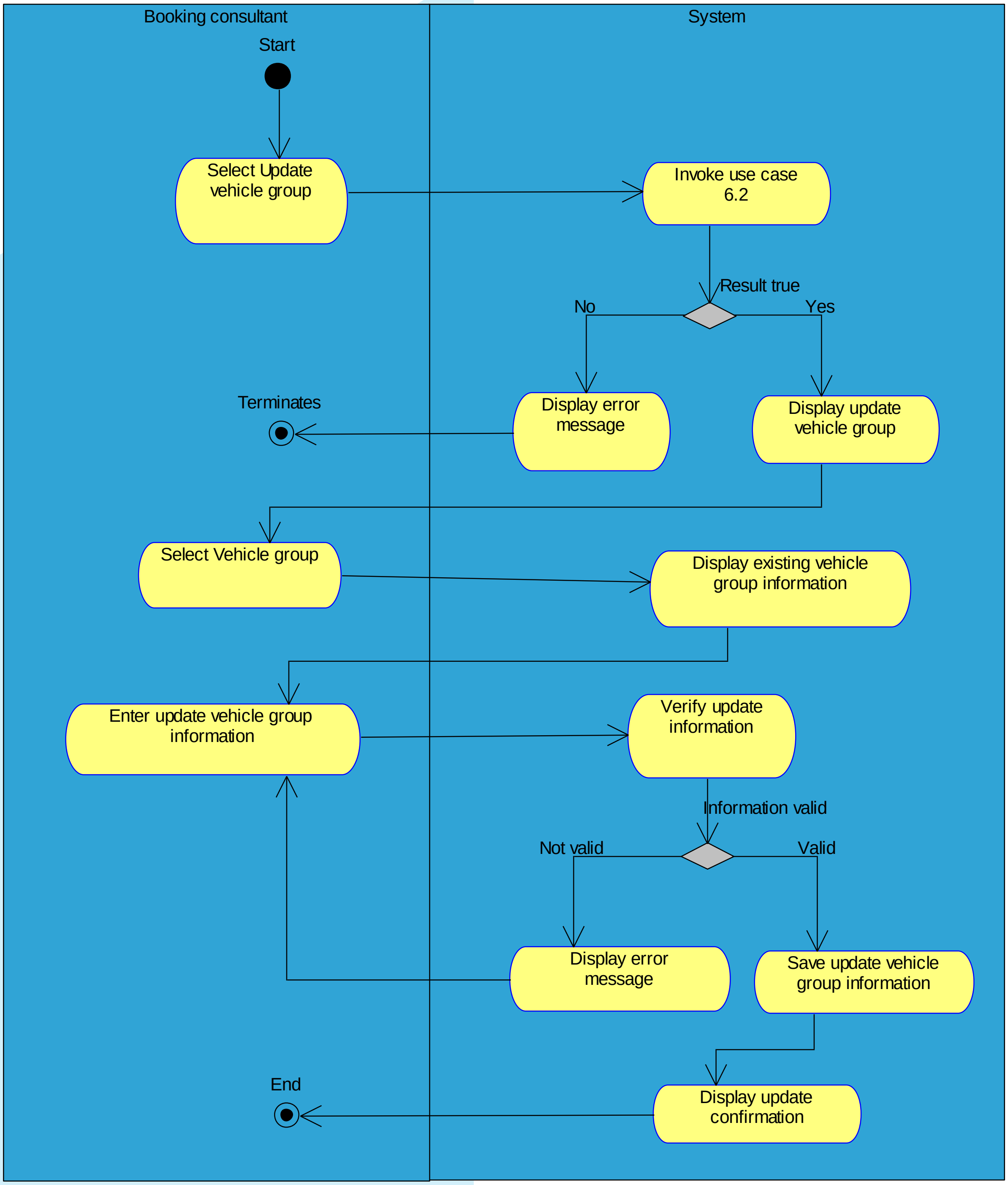

Booking consultant
System
Start
Select add vehicle
Display Vehicle details
Enter new vehicle details
Verify information
Information valid
No
Yes
Generate new Vehicle_ID
Save new vehicle information
Display add confirmation message
End

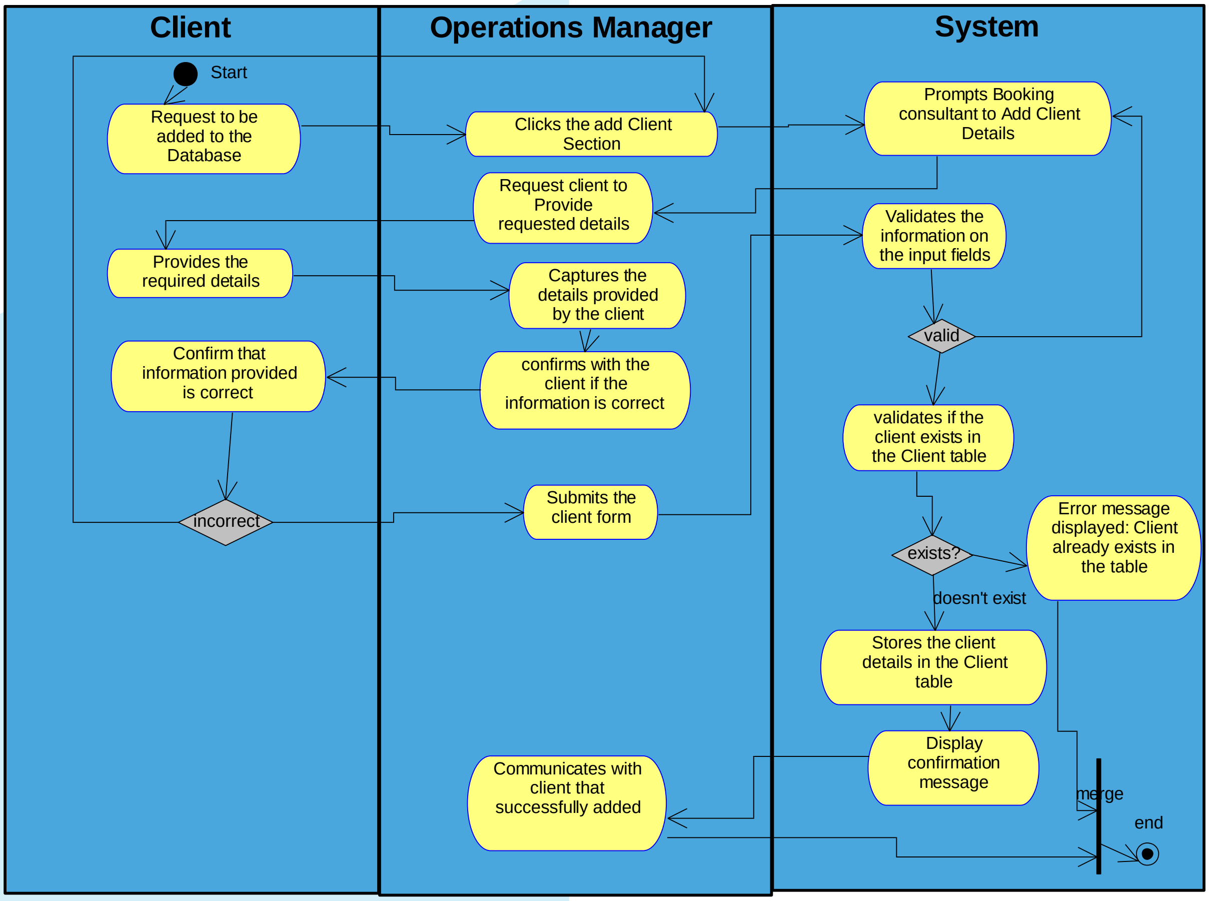

Booking consultant
System
Start
Select search vehicle
Display search vehicle
Enter characters
Capture characters
Verify Input
Input valid
No
Yes
Display error message
Retrieve vehicle information
Compare input
Match found
No
Yes
Display no match message
Display Vehicle information
Terminate
End

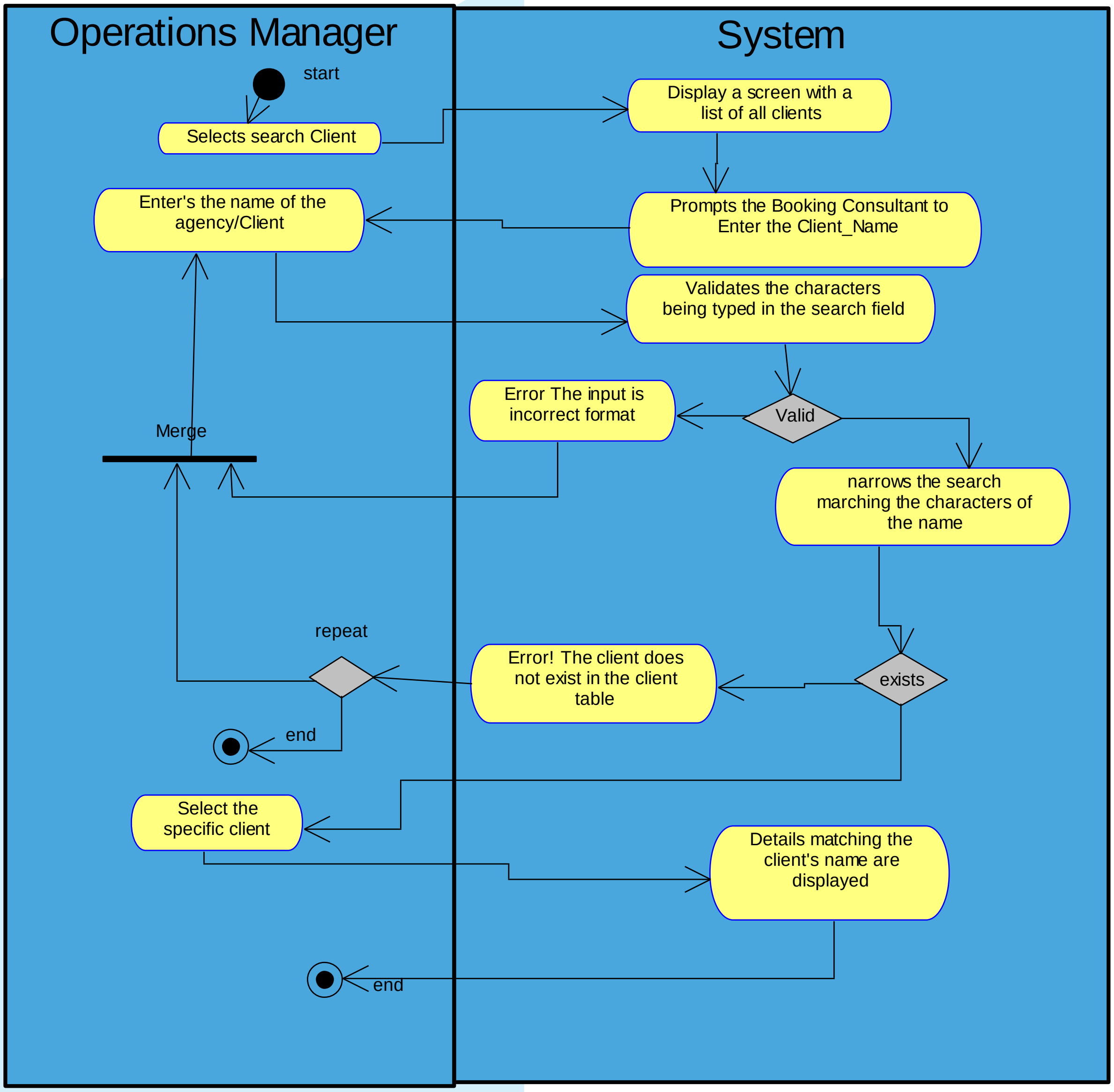

Booking consultant
System
Start
Select check vehicle availability
Retrieve trip information
Retrieve slot information
Compare trip to slot
Vehicles available
No
Yes
Terminate
Display message
Return vehicle list
End
Display vehicle list

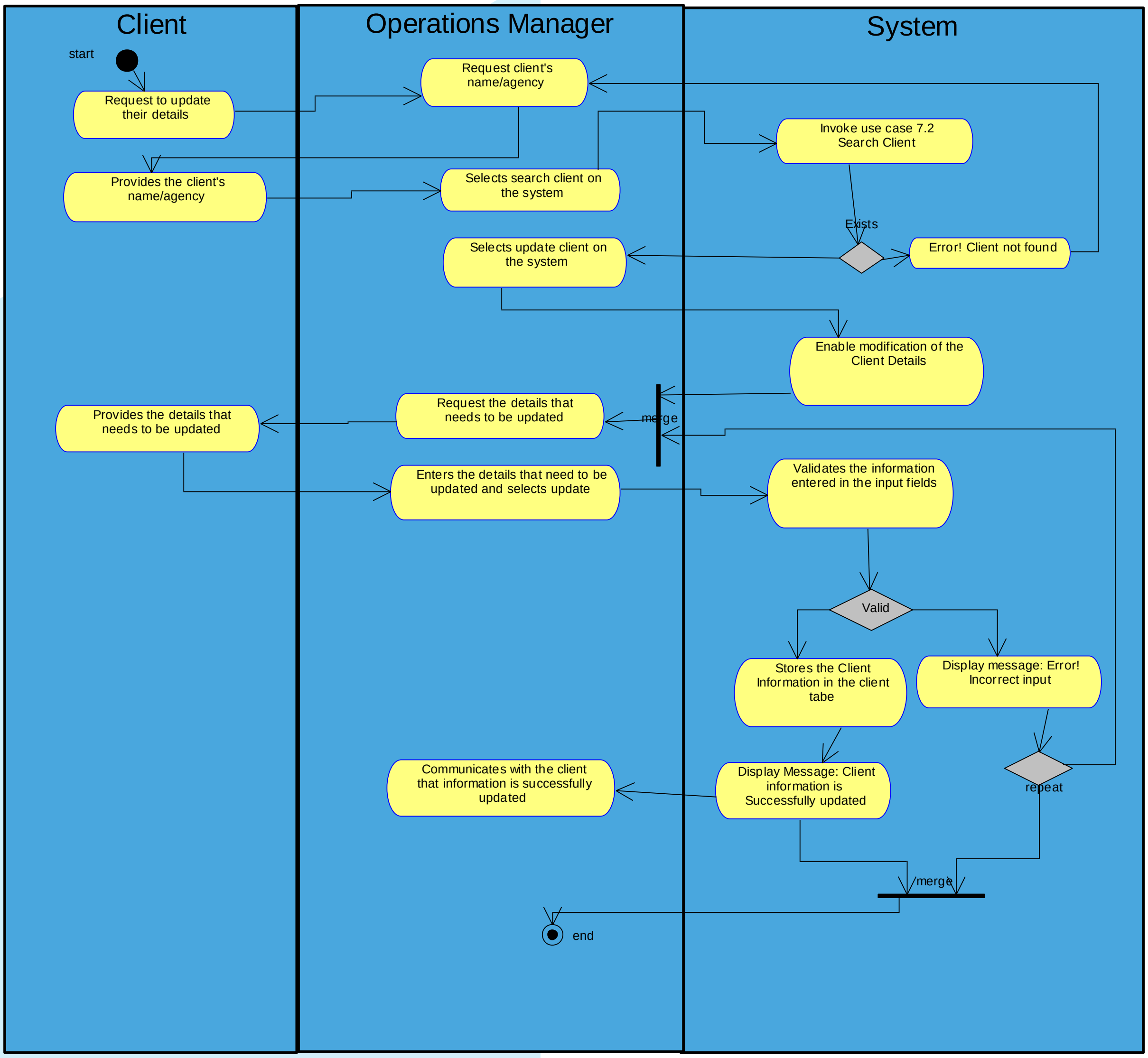

Booking consultant
System
Start
Select confirm trip
Get following day date
Retrieve booking information
Retrieve contact information
Send passenger confirmation message
Get driver trip list
Send driver confirmation message
Display confirmation completion message
End

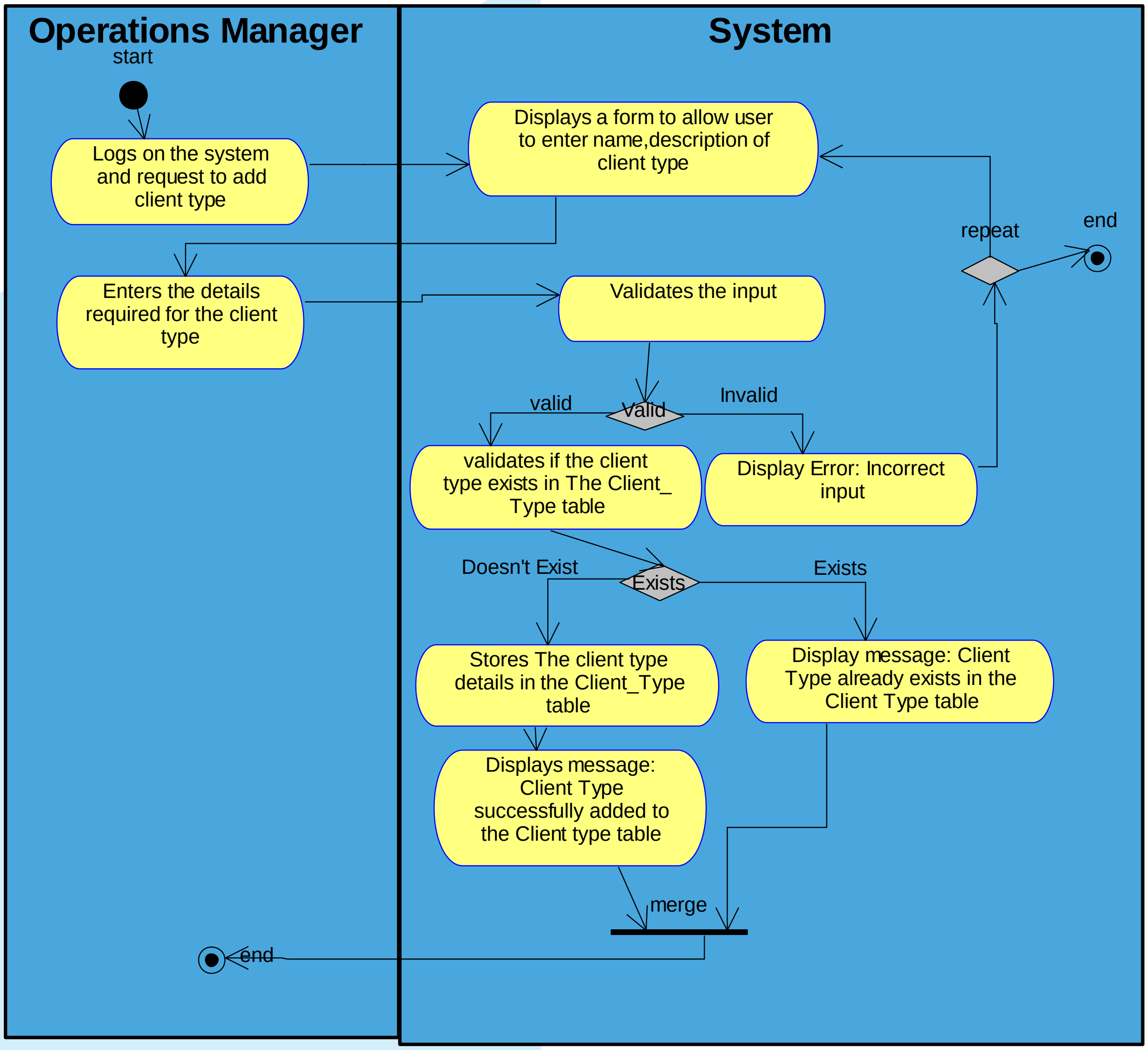

Booking consultant
System
Start
Select add vehicle group
Display Vehicle group section
Enter vehicle group details
Verify information
Information valid
No
Yes
Generate new VehicleGroupID
Save new vehicle group information
End
Display add confirmation message

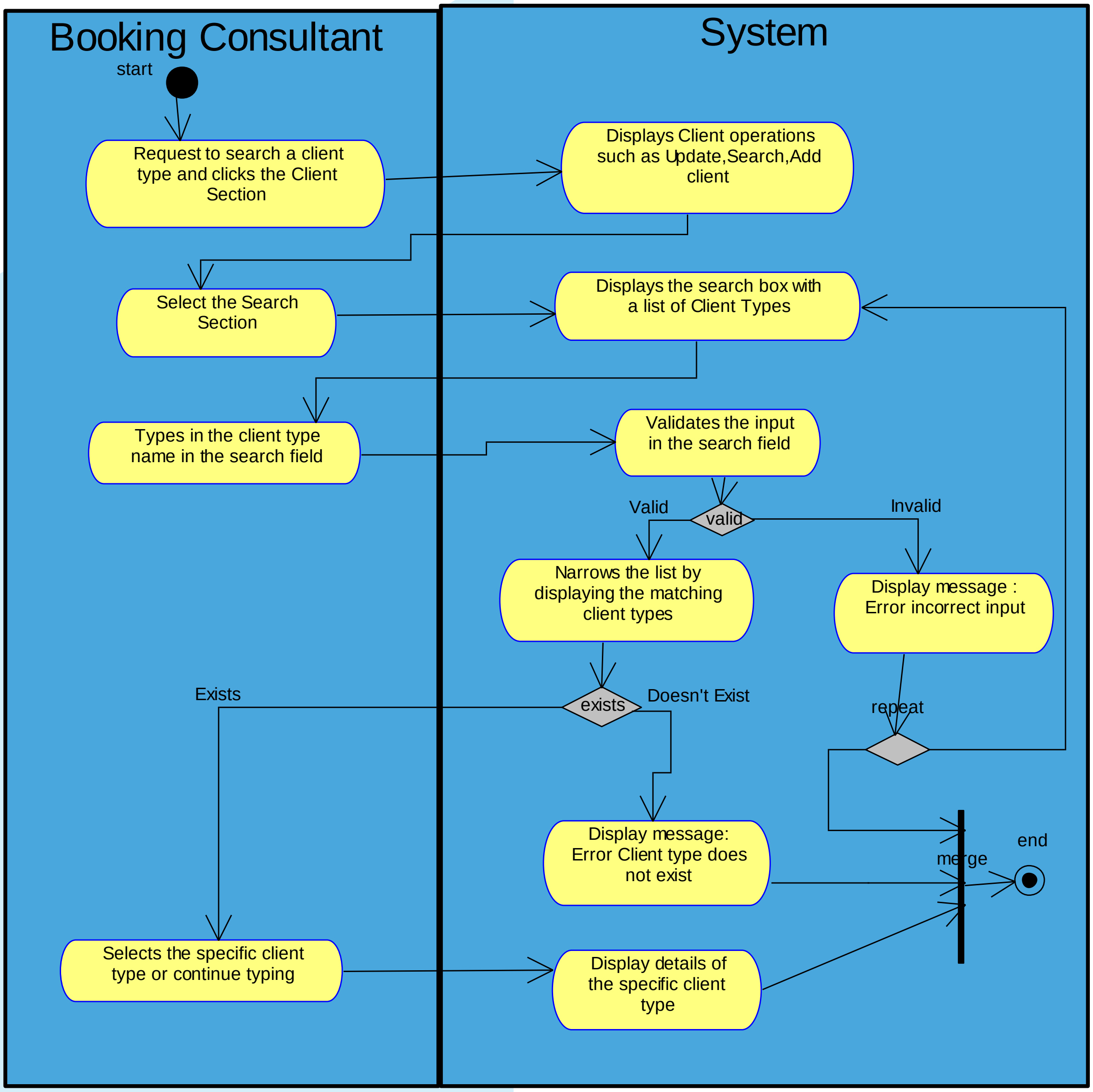

Booking consultant
System
Start
Select Update vehicle group
Invoke use case 6.2
Result true
No
Yes
Terminates
Display error message
Display update vehicle group
Select Vehicle group
Display existing vehicle group information
Enter update vehicle group information
Verify update information
Information valid
Not valid
Valid
Display error message
Save update vehicle group information
End
Display update confirmation

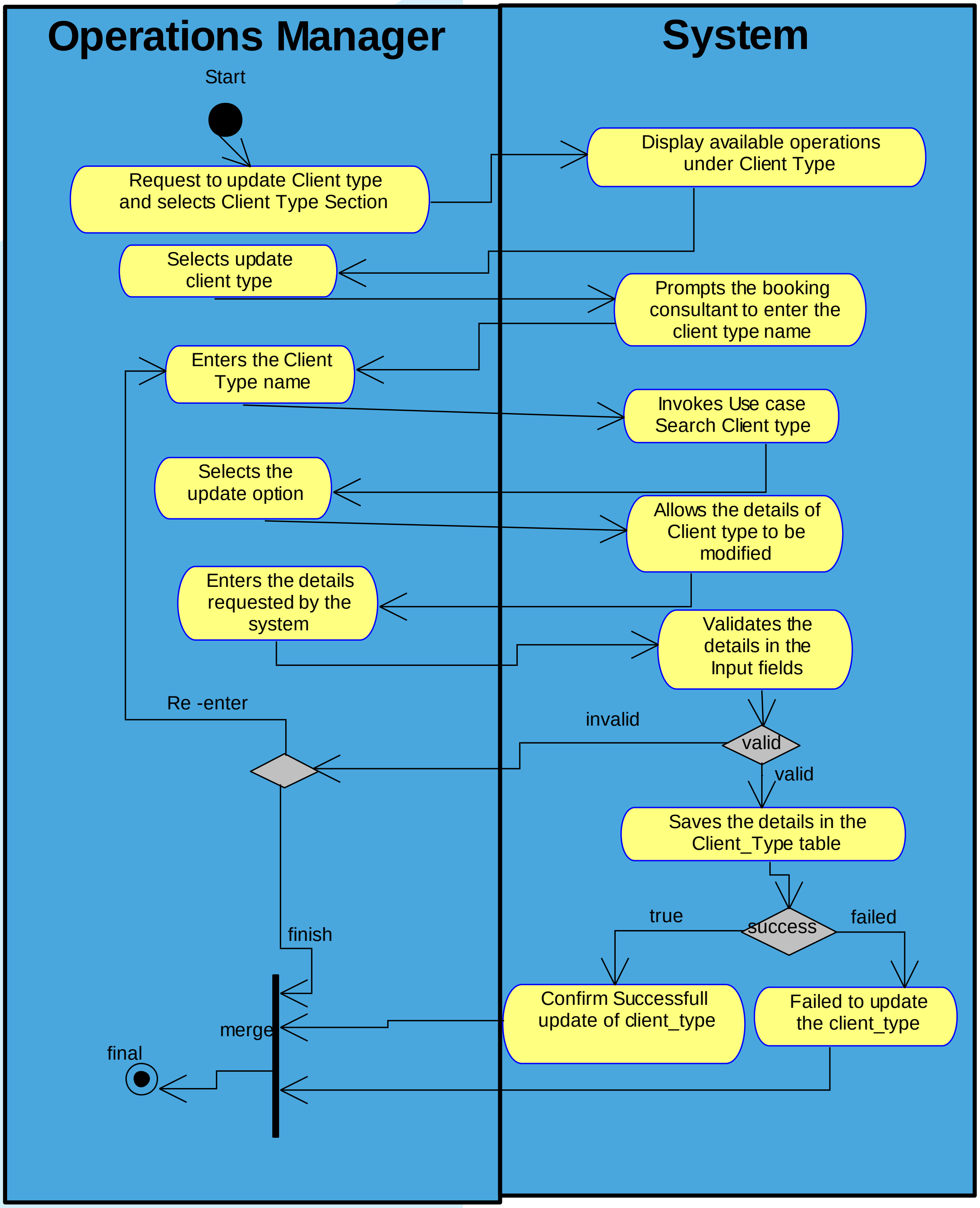

Client
Operations Manager
System
Start
Request to be added to the Database
Clicks the add Client Section
Prompts Booking consultant to Add Client Details
Request client to Provide requested details
Validates the information on the input fields
Provides the required details
Captures the details provided by the client
valid
Confirm that information provided is correct
confirms with the client if the information is correct
validates if the client exists in the Client table
incorrect
Submits the client form
Error message displayed: Client already exists in the table
exists?
doesn't exist
Stores the client details in the Client table
Display confirmation message
Communicates with client that successfully added
merge
end

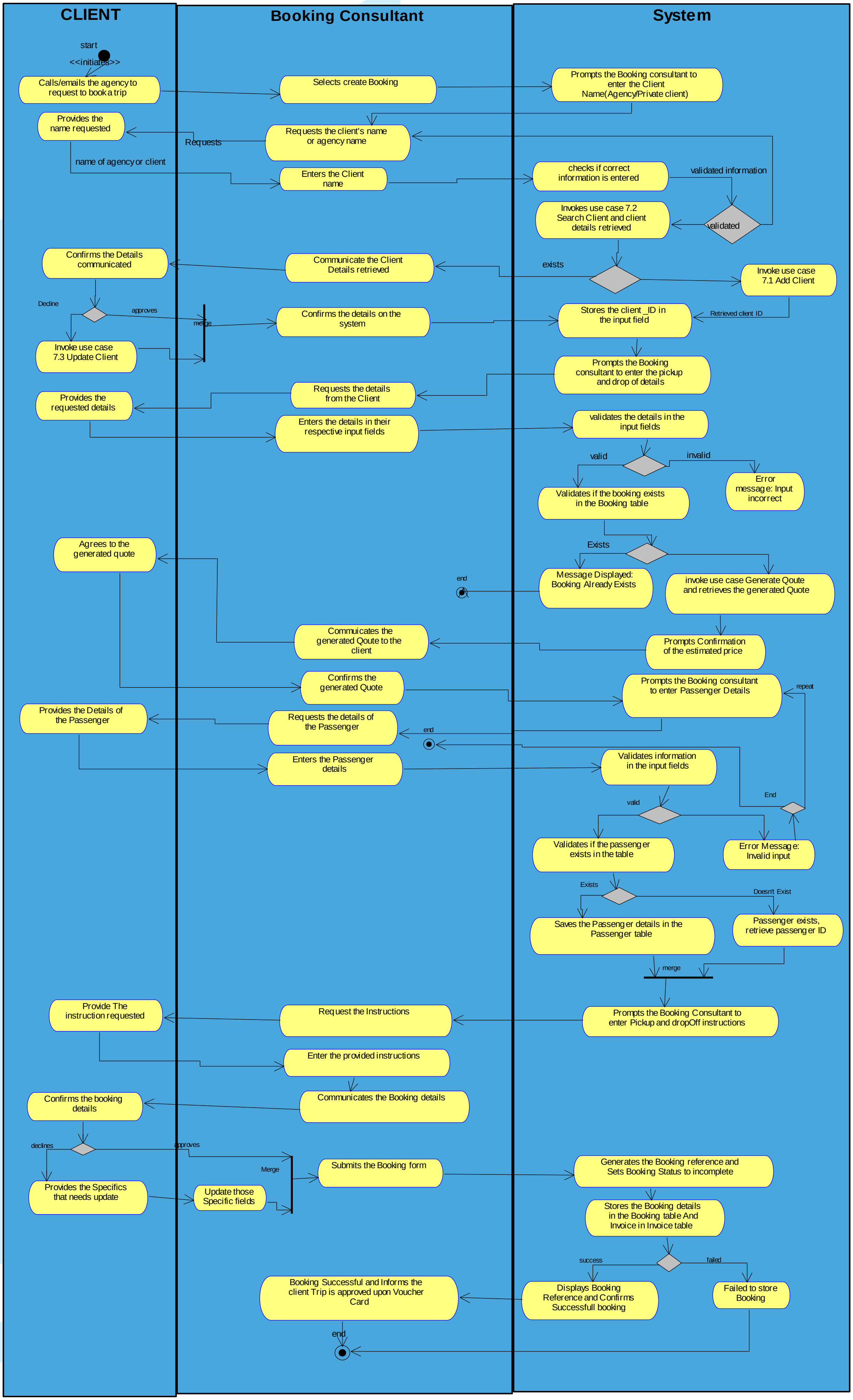

Operations Manager
System
start
Selects search Client
Display a screen with a list of all clients
Prompts the Booking Consultant to Enter the Client_Name
Enter's the name of the agency/Client
Validates the characters being typed in the search field
Error The input is incorrect format
Valid
Merge
narrows the search marching the characters of the name
repeat
Error! The client does not exist in the client table
exists
end
Select the specific client
Details matching the client's name are displayed
end

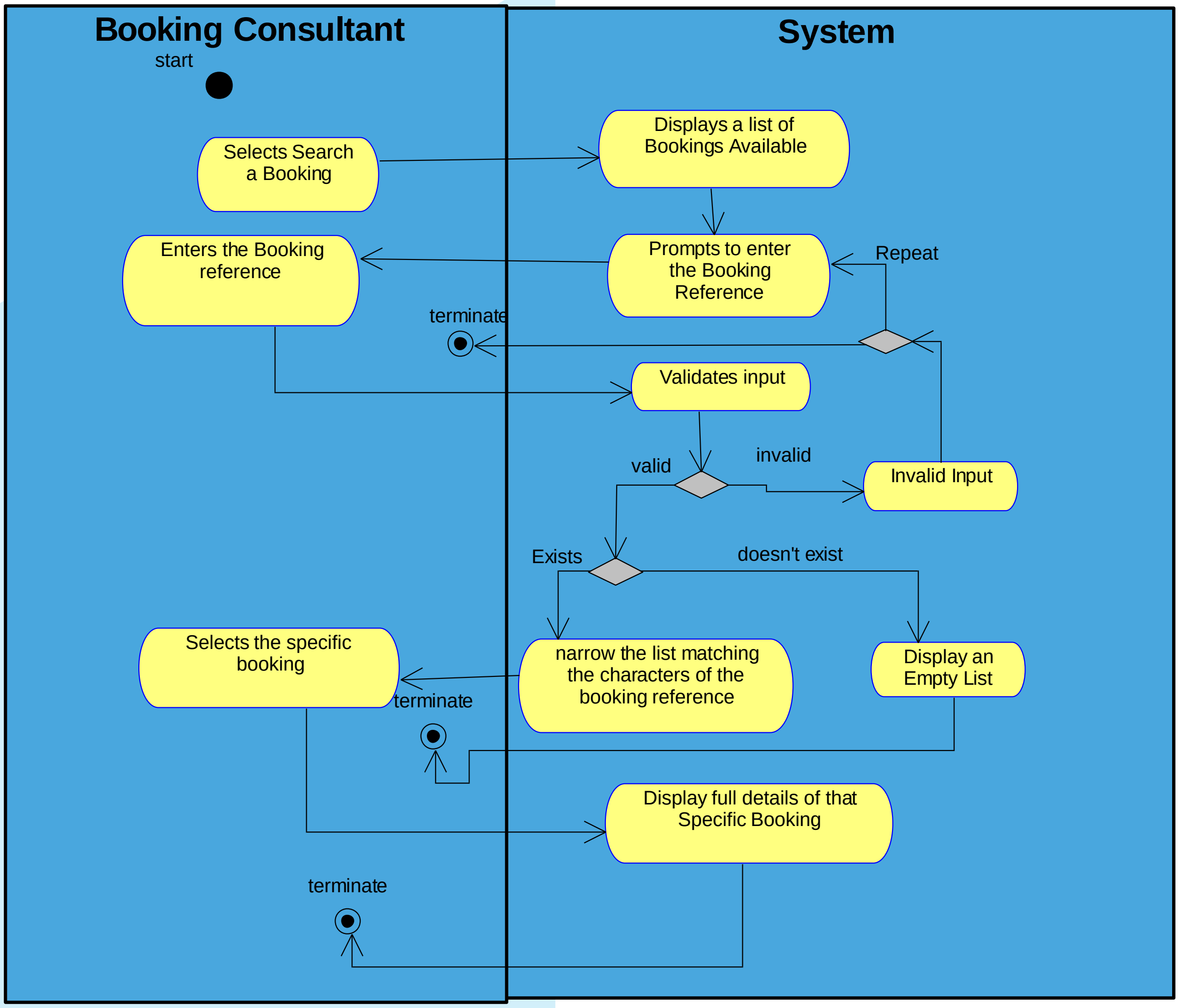

Client
Operations Manager
System
start
Request to update their details
Request client's name/agency
Invoke use case 7.2 Search Client
Provides the client's name/agency
Selects search client on the system
Exists
Selects update client on the system
Error! Client not found
Enable modification of the Client Details
Provides the details that needs to be updated
Request the details that needs to be updated
merge
Enters the details that need to be updated and selects update
Validates the information entered in the input fields
Valid
Stores the Client Information in the client tabe
Display message: Error! Incorrect input
Communicates with the client that information is successfully updated
Display Message: Client information is Successfully updated
repeat
merge
end

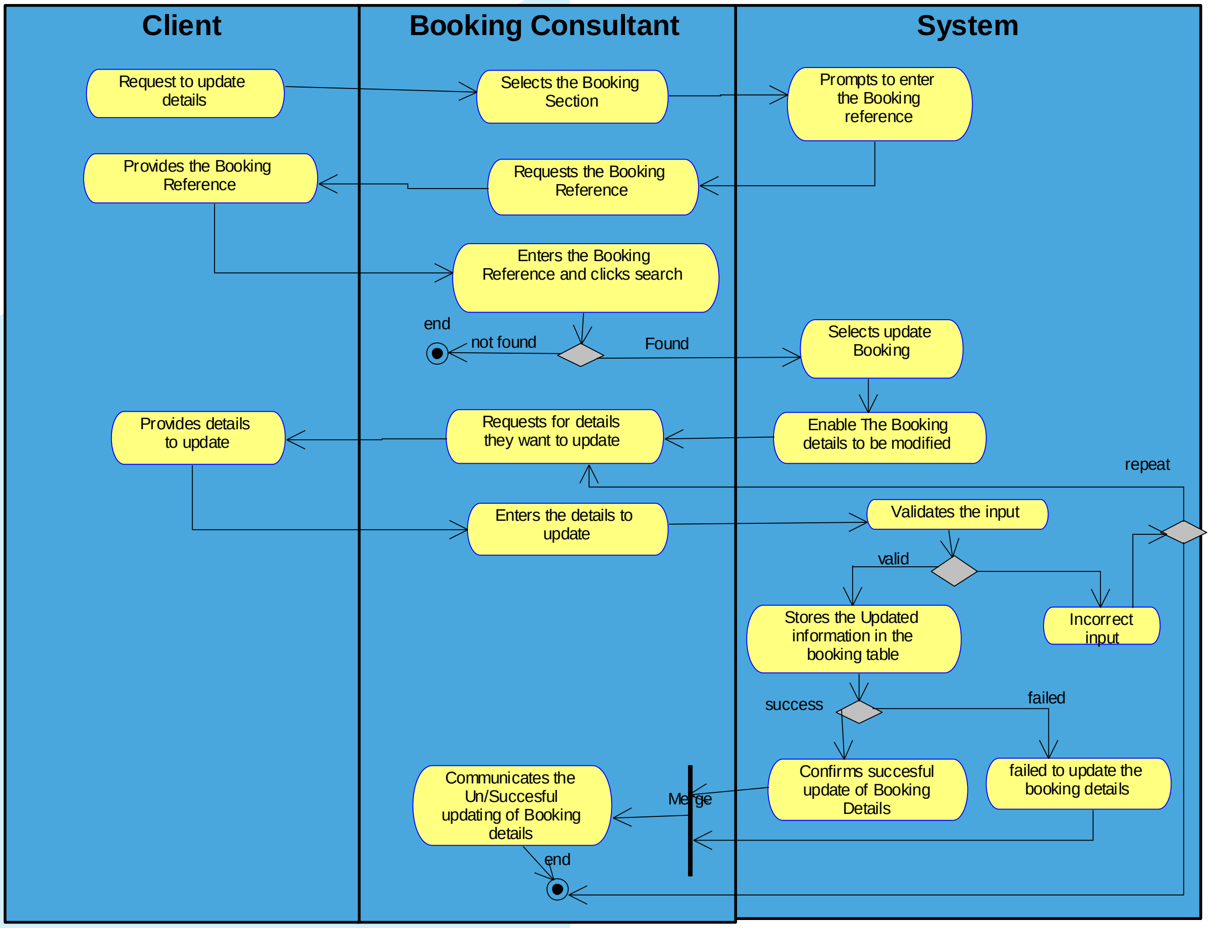

Operations Manager
System
start
Logs on the system and request to add client type
Displays a form to allow user to enter name,description of client type
Enters the details required for the client type
Validates the input
repeat
end
valid
Valid
Invalid
validates if the client type exists in The Client_ Type table
Display Error: Incorrect input
Doesn't Exist
Exists
Exists
Stores The client type details in the Client_Type table
Display message: Client Type already exists in the Client Type table
Displays message: Client Type successfully added to the Client type table
merge
end

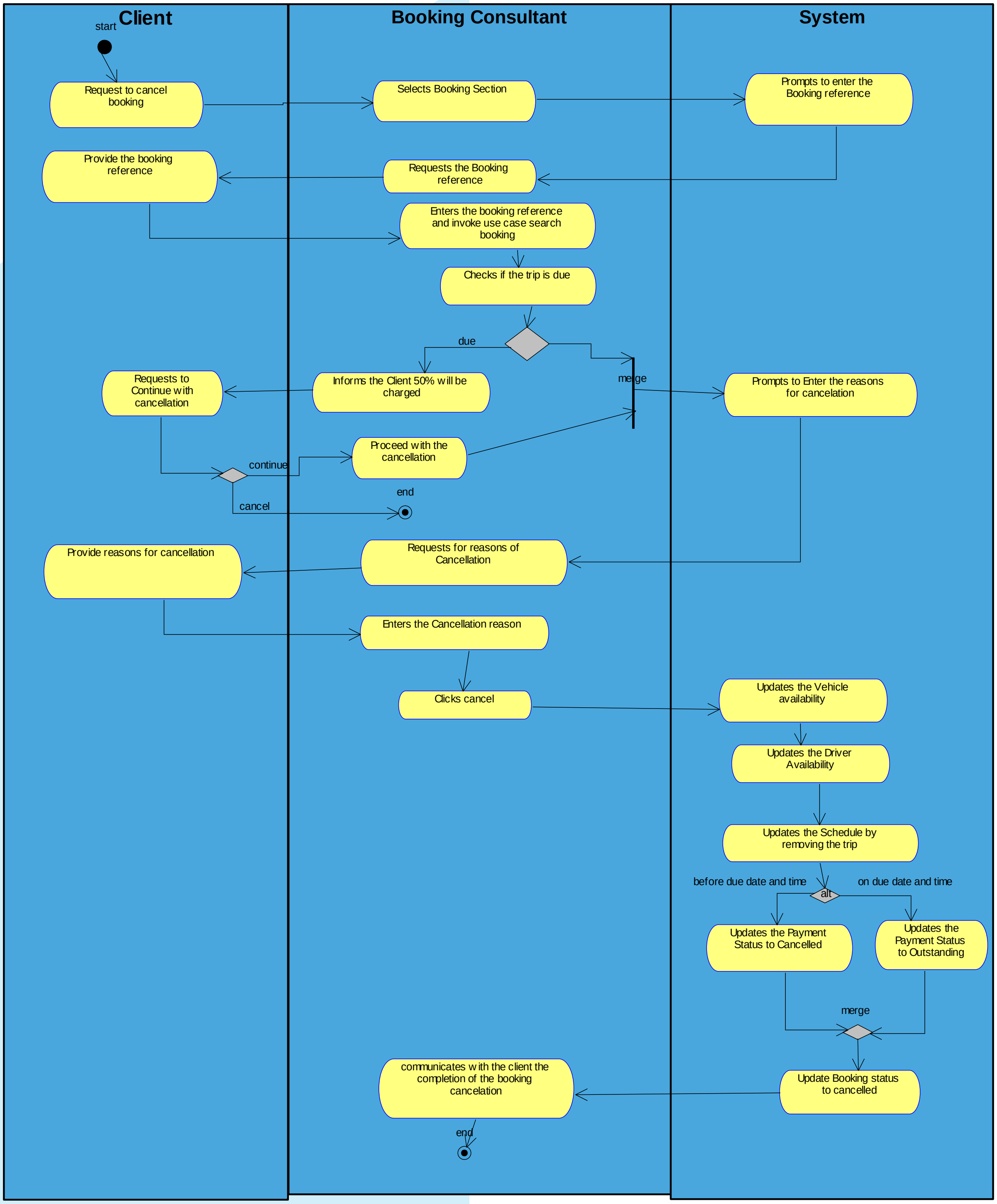

Booking Consultant
System
start
Request to search a client type and clicks the Client Section
Displays Client operations such as Update,Search,Add client
Select the Search Section
Displays the search box with a list of Client Types
Types in the client type name in the search field
Validates the input in the search field
Valid
valid
Invalid
Narrows the list by displaying the matching client types
Display message : Error incorrect input
Exists
exists
Doesn't Exist
repeat
Display message: Error Client type does not exist
end
merge
Selects the specific client type or continue typing
Display details of the specific client type

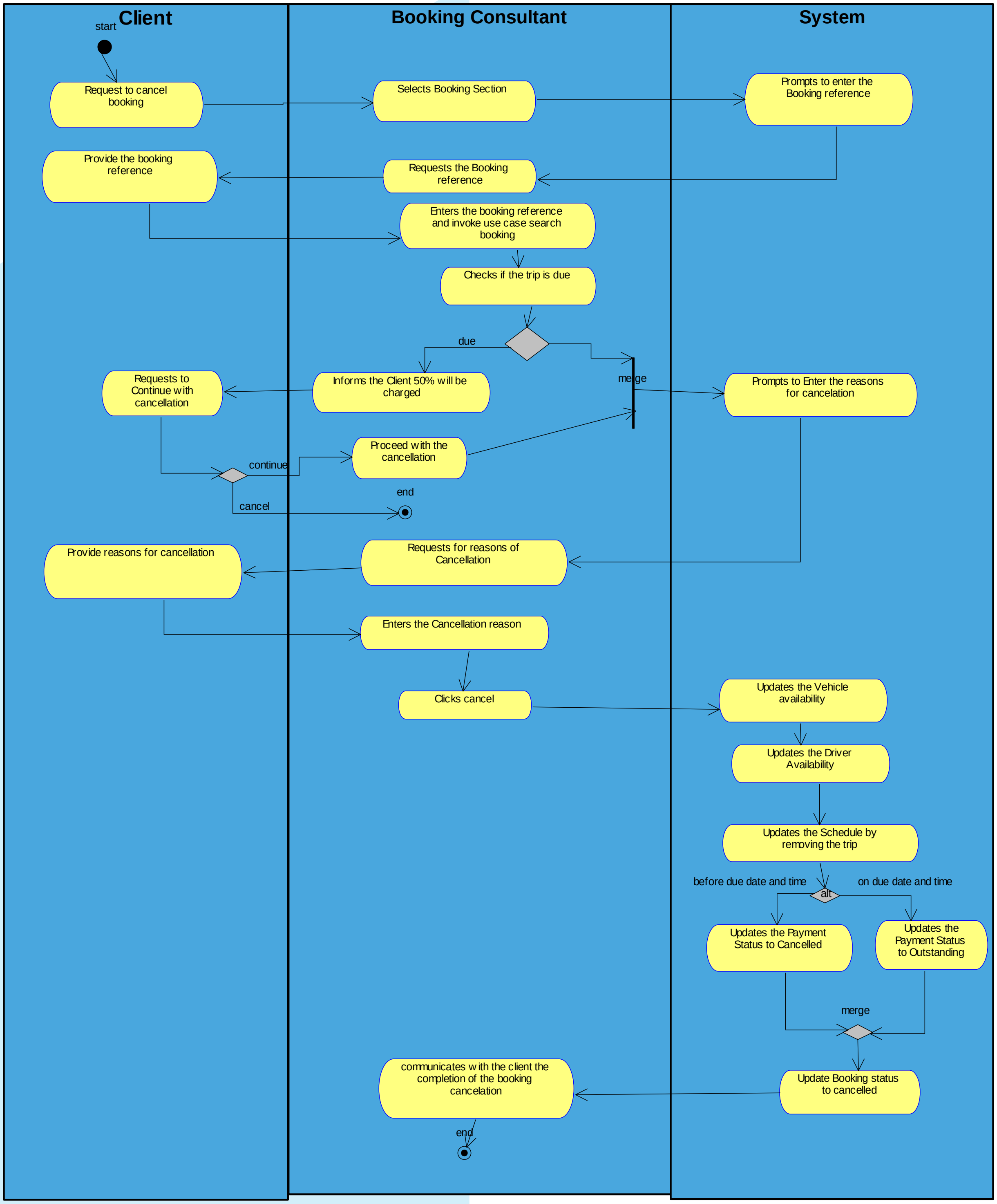

Operations Manager
System
Start
Request to update Client type and selects Client Type Section
Display available operations under Client Type
Selects update client type
Prompts the booking consultant to enter the client type name
Enters the Client Type name
Invokes Use case Search Client type
Selects the update option
Allows the details of Client type to be modified
Enters the details requested by the system
Validates the details in the Input fields
Re -enter
invalid
valid
valid
Saves the details in the Client_Type table
true
success
failed
finish
Confirm Successfull update of client_type
Failed to update the client_type
merge
final

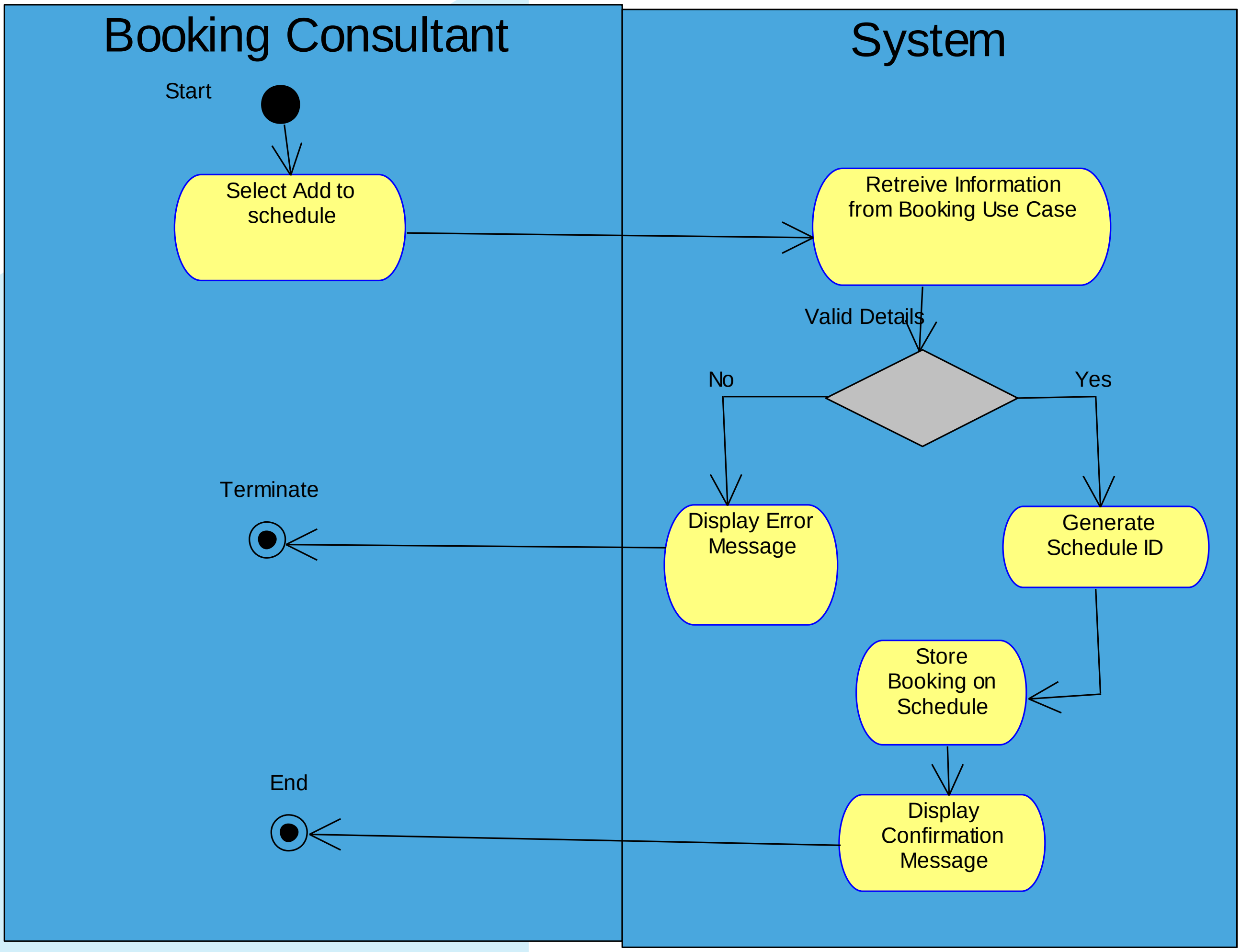

CLIENT
Booking Consultant
System
start
<<initiates>>
Calls/emails the agency to request to book a trip
Selects create Booking
Prompts the Booking consultant to enter the Client Name(Agency/Private client)
Provides the name requested
Requests
Requests the client's name or agency name
name of agency or client
Enters the Client name
checks if correct information is entered
validated information
Invokes use case 7.2 Search Client and client details retrieved
validated
Confirms the Details communicated
Communicate the Client Details retrieved
exists
Invoke use case 7.1 Add Client
Decline
approves
merge
Confirms the details on the system
Stores the client _ID in the input field
Retrieved client ID
Invoke use case 7.3 Update Client
Prompts the Booking consultant to enter the pickup and drop of details
Provides the requested details
Requests the details from the Client
Enters the details in their respective input fields
validates the details in the input fields
valid
invalid
Validates if the booking exists in the Booking table
Error message: Input incorrect
Exists
Agrees to the generated quote
end
Message Displayed: Booking Already Exists
invoke use case Generate Qoute and retrieves the generated Quote
Commuicates the generated Qoute to the client
Prompts Confirmation of the estimated price
Confirms the generated Quote
Prompts the Booking consultant to enter Passenger Details
repeat
Provides the Details of the Passenger
Requests the details of the Passenger
end
Enters the Passenger details
Validates information in the input fields
End
valid
Validates if the passenger exists in the table
Error Message: Invalid input
Exists
Doesn't Exist
Saves the Passenger details in the Passenger table
Passenger exists, retrieve passenger ID
merge
Provide The instruction requested
Request the Instructions
Prompts the Booking Consultant to enter Pickup and dropOff instructions
Enter the provided instructions
Confirms the booking details
Communicates the Booking details
declines
approves
Merge
Submits the Booking form
Generates the Booking reference and Sets Booking Status to incomplete
Provides the Specifics that needs update
Update those Specific fields
Stores the Booking details in the Booking table And Invoice in Invoice table
success
failed
Booking Successful and Informs the client Trip is approved upon Voucher Card
Displays Booking Reference and Confirms Successfull booking
Failed to store Booking
end

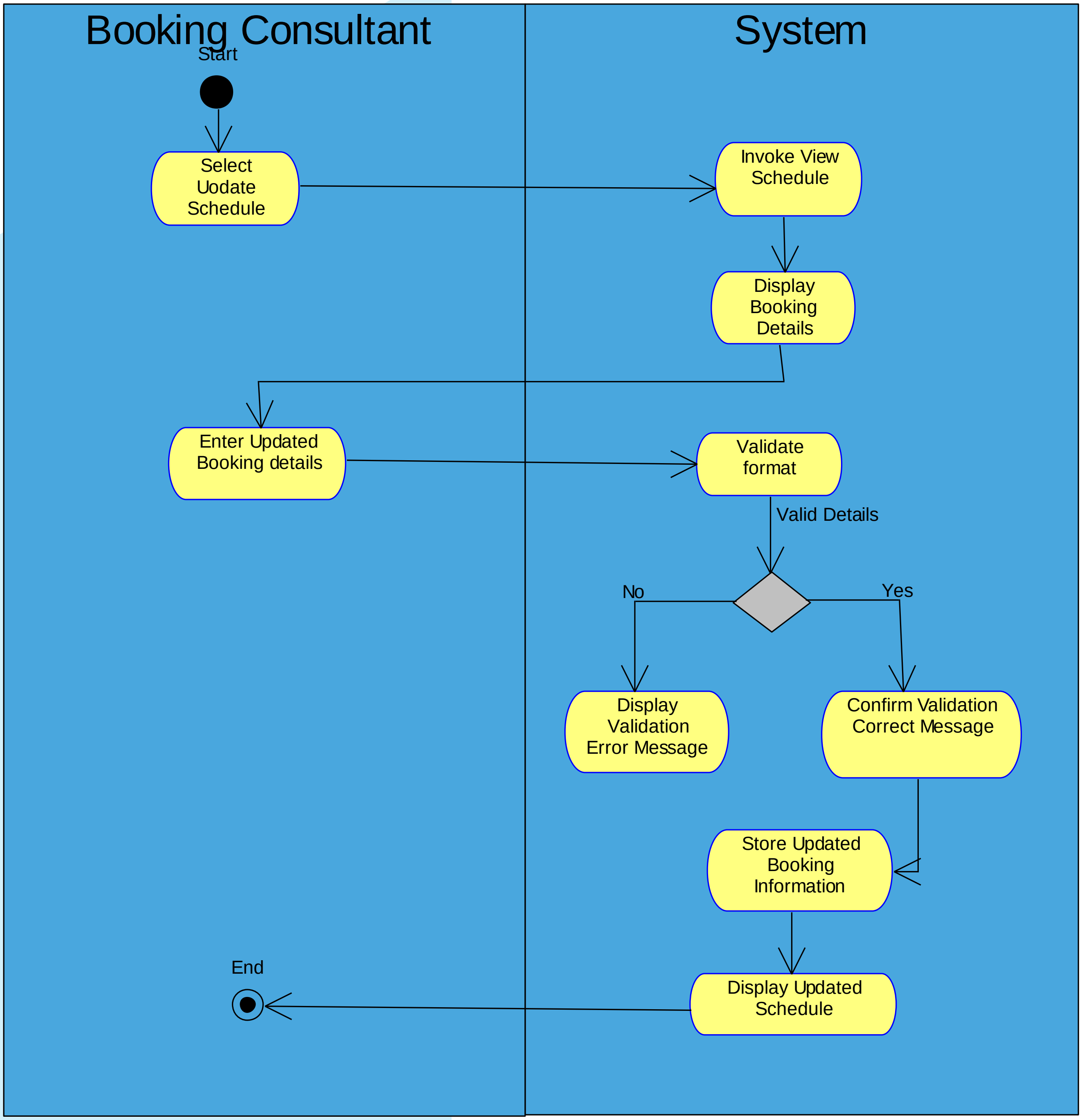

Booking Consultant
start
Selects Search a Booking
Enters the Booking reference
terminate
Selects the specific booking
terminate
terminate
System
Displays a list of Bookings Available
Prompts to enter the Booking Reference
Repeat
Validates input
valid
invalid
Invalid Input
Exists
doesn't exist
narrow the list matching the characters of the booking reference
Display an Empty List
Display full details of that Specific Booking

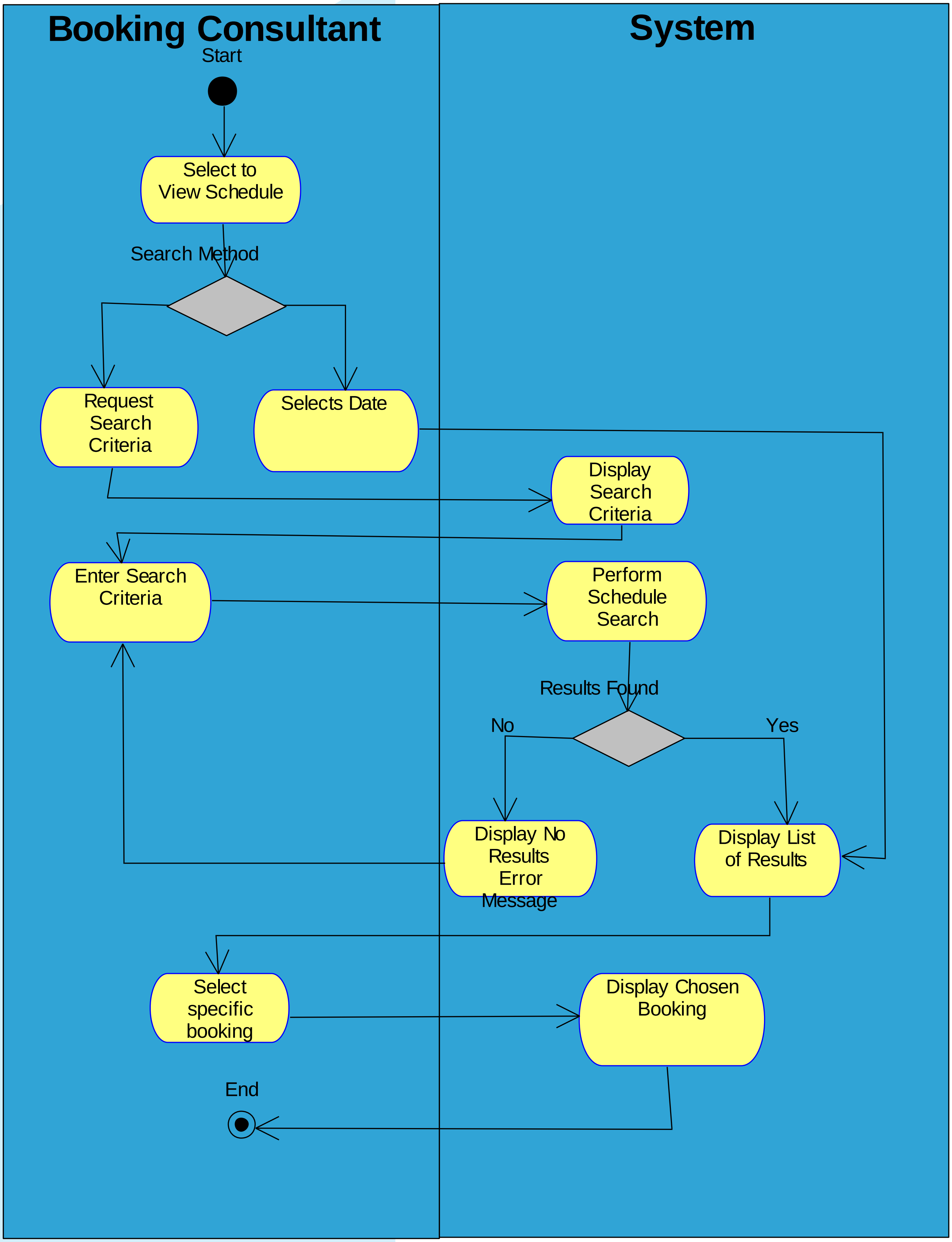

Client
Booking Consultant
System
Request to update details
Selects the Booking Section
Prompts to enter the Booking reference
Provides the Booking Reference
Requests the Booking Reference
Enters the Booking Reference and clicks search
end
not found
Found
Selects update Booking
Provides details to update
Requests for details they want to update
Enable The Booking details to be modified
repeat
Enters the details to update
Validates the input
valid
Stores the Updated information in the booking table
Incorrect input
success
failed
Confirms succesful update of Booking Details
failed to update the booking details
Communicates the Un/Succesful updating of Booking details
Merge
end

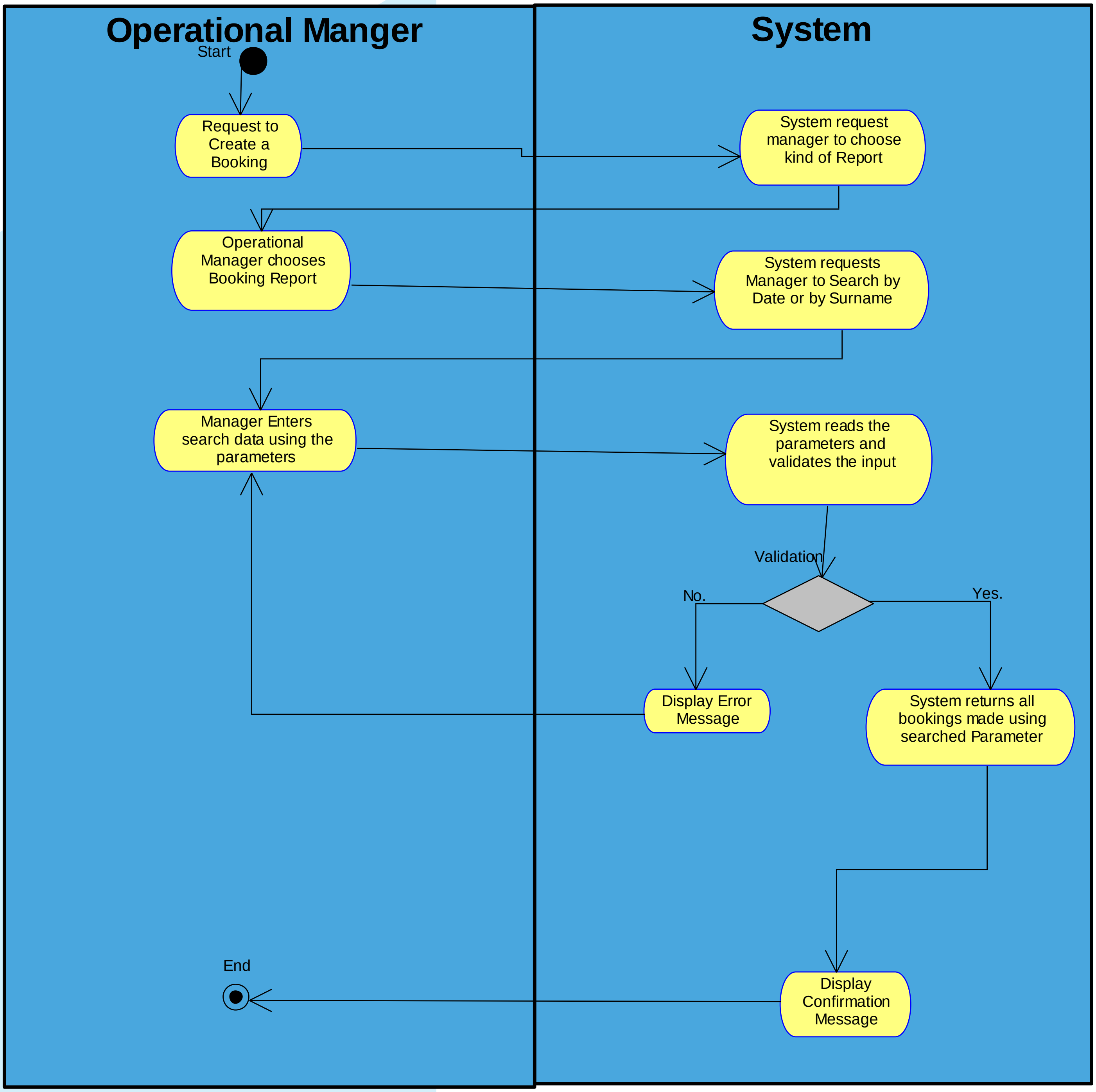

Client
Booking Consultant
System
start
Request to cancel booking
Selects Booking Section
Prompts to enter the Booking reference
Provide the booking reference
Requests the Booking reference
Enters the booking reference and invoke use case search booking
Checks if the trip is due
due
merge
Requests to Continue with cancellation
Informs the Client 50% will be charged
Prompts to Enter the reasons for cancelation
continue
Proceed with the cancellation
end
cancel
Provide reasons for cancellation
Requests for reasons of Cancellation
Enters the Cancellation reason
Clicks cancel
Updates the Vehicle availability
Updates the Driver Availability
Updates the Schedule by removing the trip
before due date and time
alt
on due date and time
Updates the Payment Status to Cancelled
Updates the Payment Status to Outstanding
merge
communicates with the client the completion of the booking cancelation
Update Booking status to cancelled
end

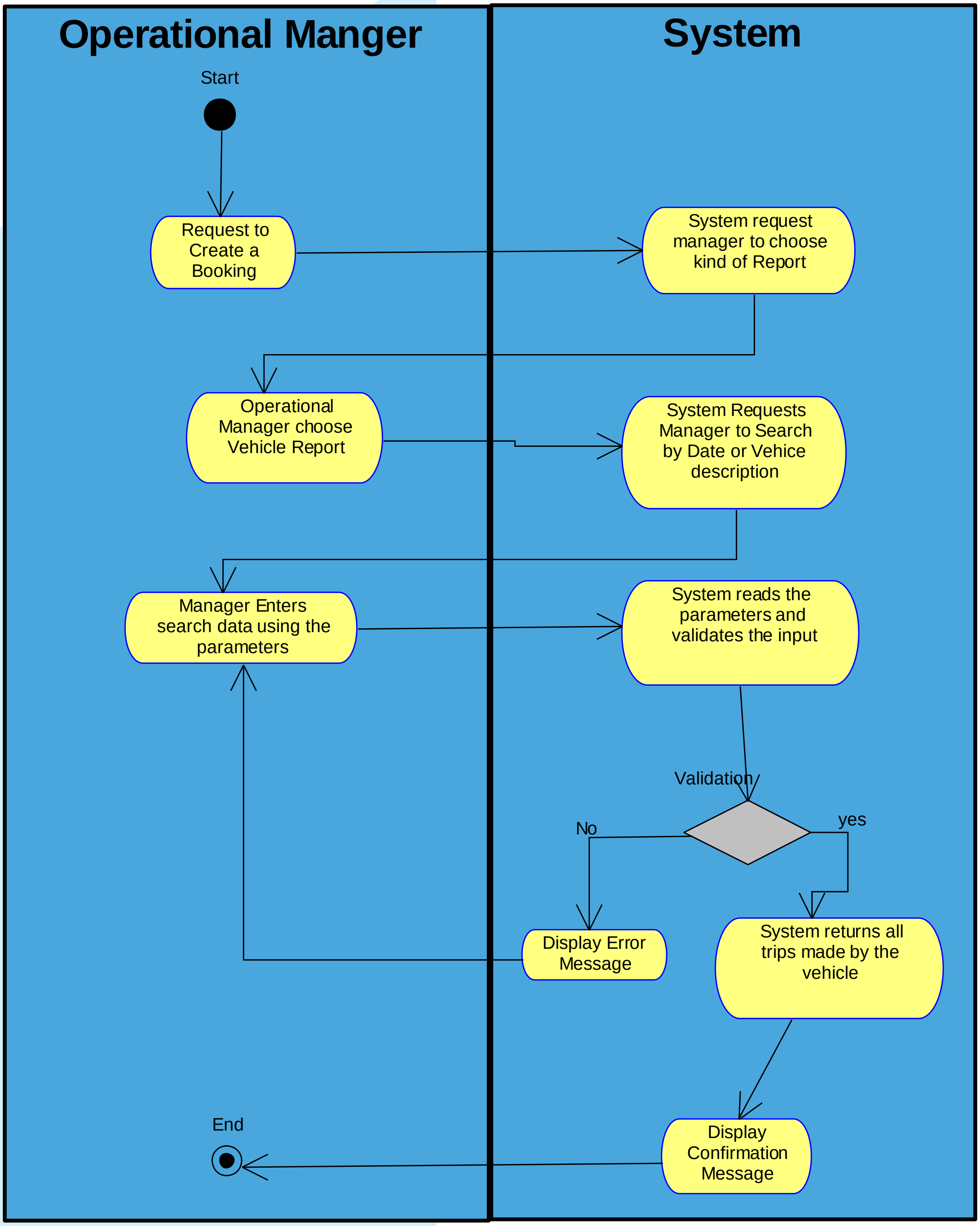

Client
Booking Consultant
System
start
Request to cancel booking
Selects Booking Section
Prompts to enter the Booking reference
Provide the booking reference
Requests the Booking reference
Enters the booking reference and invoke use case search booking
Checks if the trip is due
due
merge
Requests to Continue with cancellation
Informs the Client 50% will be charged
Prompts to Enter the reasons for cancelation
continue
Proceed with the cancellation
end
cancel
Provide reasons for cancellation
Requests for reasons of Cancellation
Enters the Cancellation reason
Clicks cancel
Updates the Vehicle availability
Updates the Driver Availability
Updates the Schedule by removing the trip
before due date and time
alt
on due date and time
Updates the Payment Status to Cancelled
Updates the Payment Status to Outstanding
merge
communicates with the client the completion of the booking cancelation
Update Booking status to cancelled
end

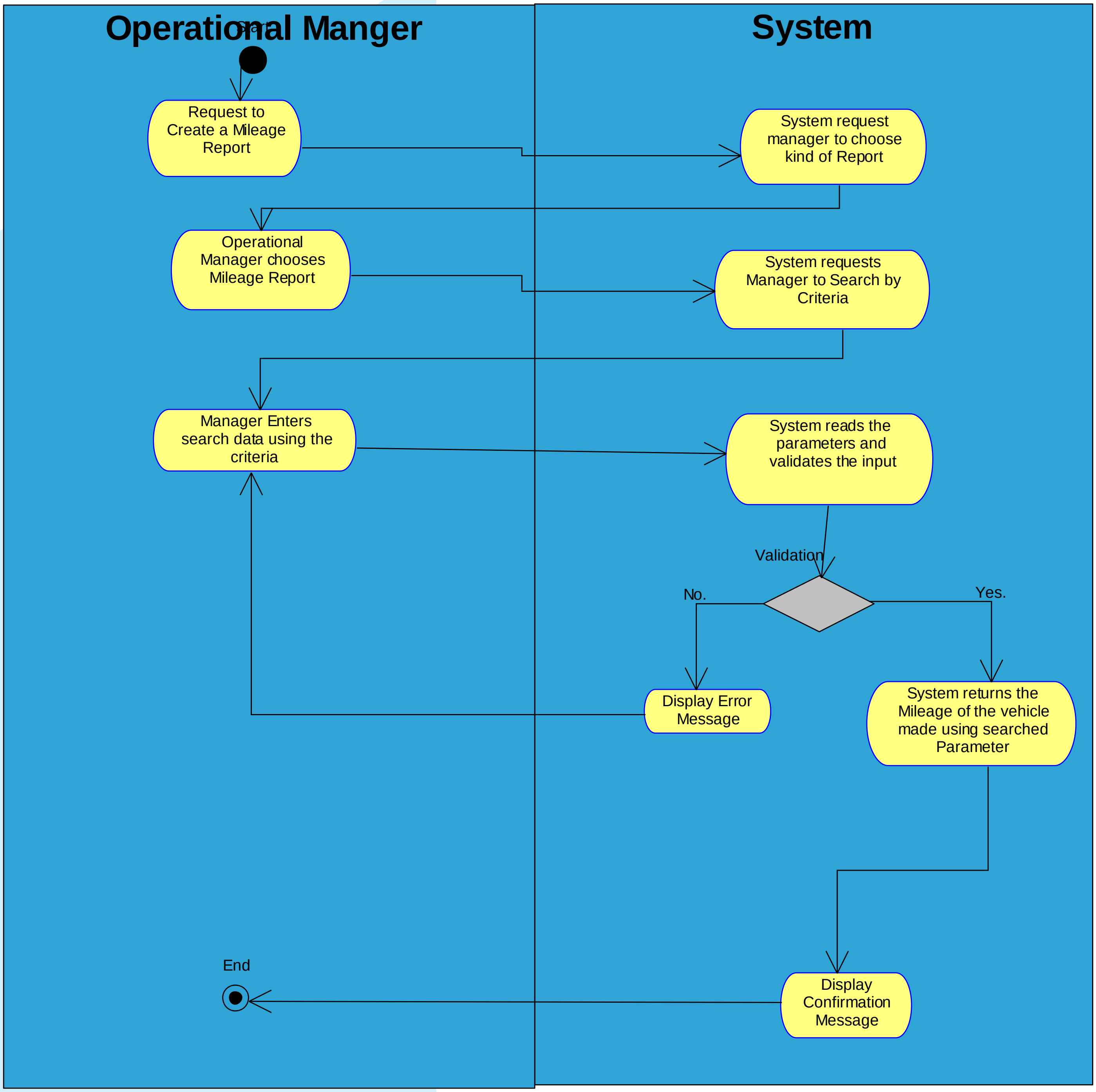

Booking Consultant
System
Start
Select Add to schedule
Retreive Information from Booking Use Case
Valid Details
No
Yes
Terminate
Display Error Message
Generate Schedule ID
Store Booking on Schedule
End
Display Confirmation Message

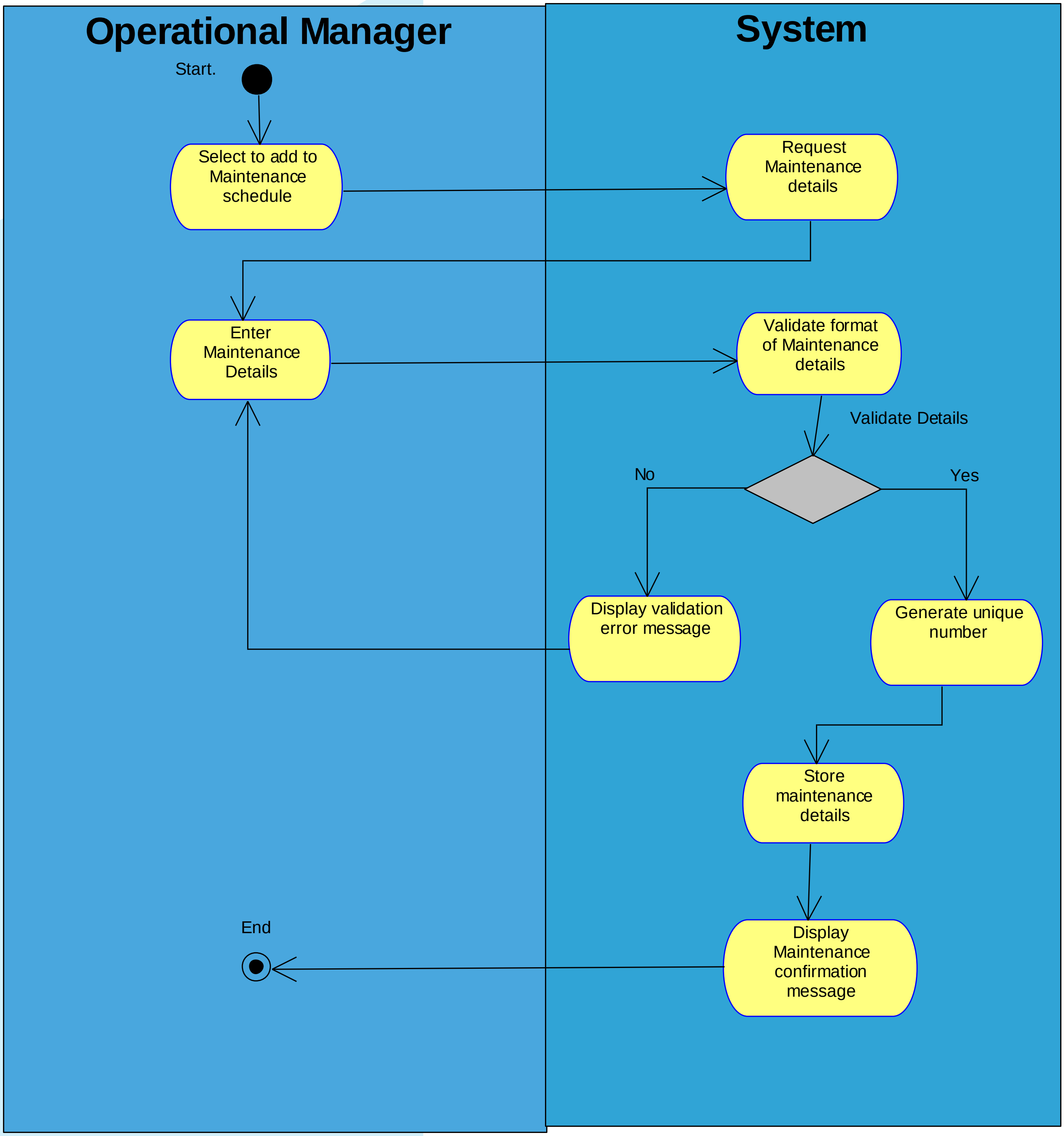

Booking Consultant
System
Start
Select
Uodate
Schedule
Invoke View
Schedule
Display
Booking
Details
Enter Updated
Booking details
Validate
format
Valid Details
No
Yes
Display
Validation
Error Message
Confirm Validation
Correct Message
Store Updated
Booking
Information
End
Display Updated
Schedule

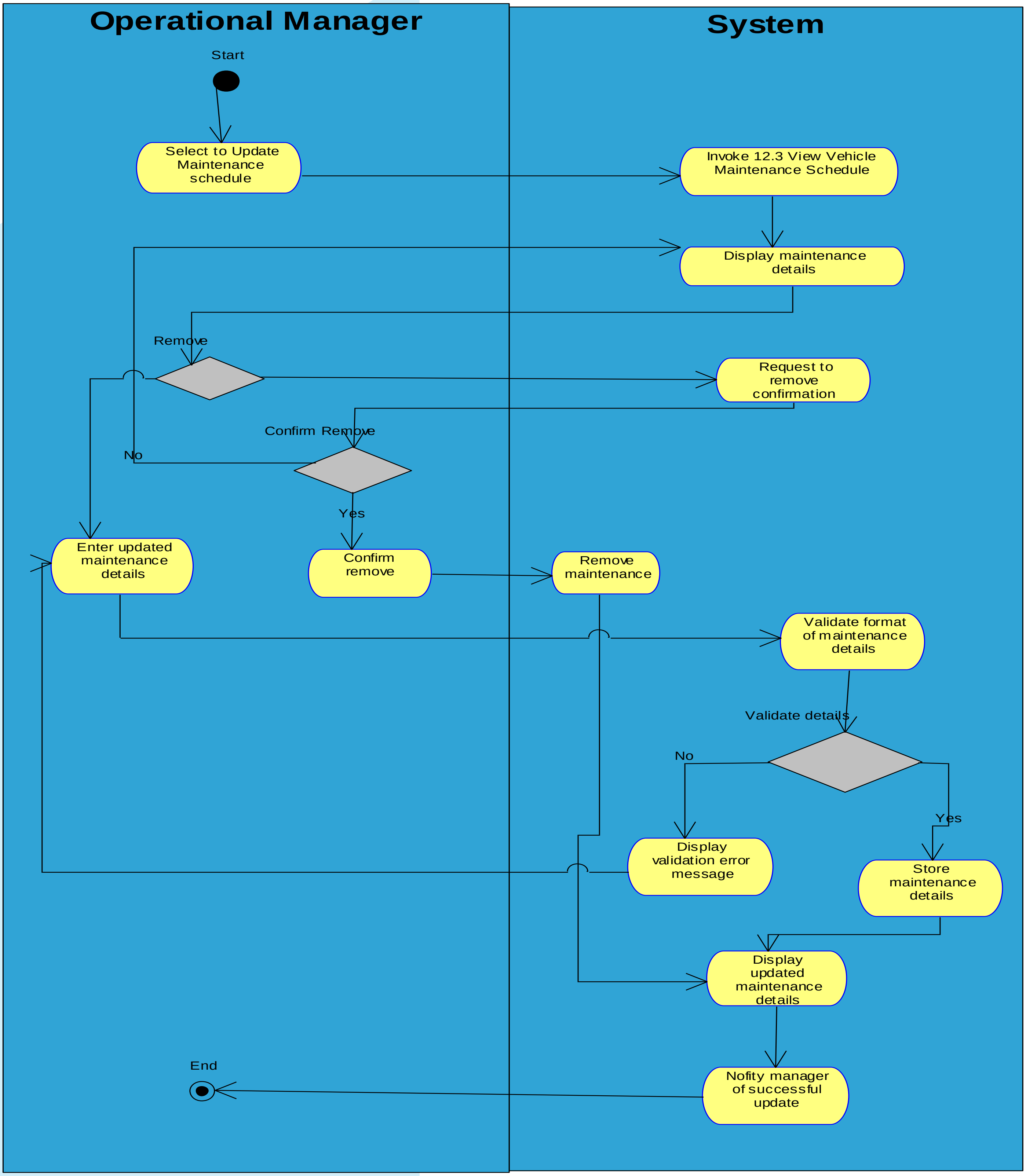

Booking Consultant
System
Start
Select to View Schedule
Search Method
Request Search Criteria
Selects Date
Display Search Criteria
Enter Search Criteria
Perform Schedule Search
Results Found
No
Yes
Display No Results Error Message
Display List of Results
Select specific booking
Display Chosen Booking
End

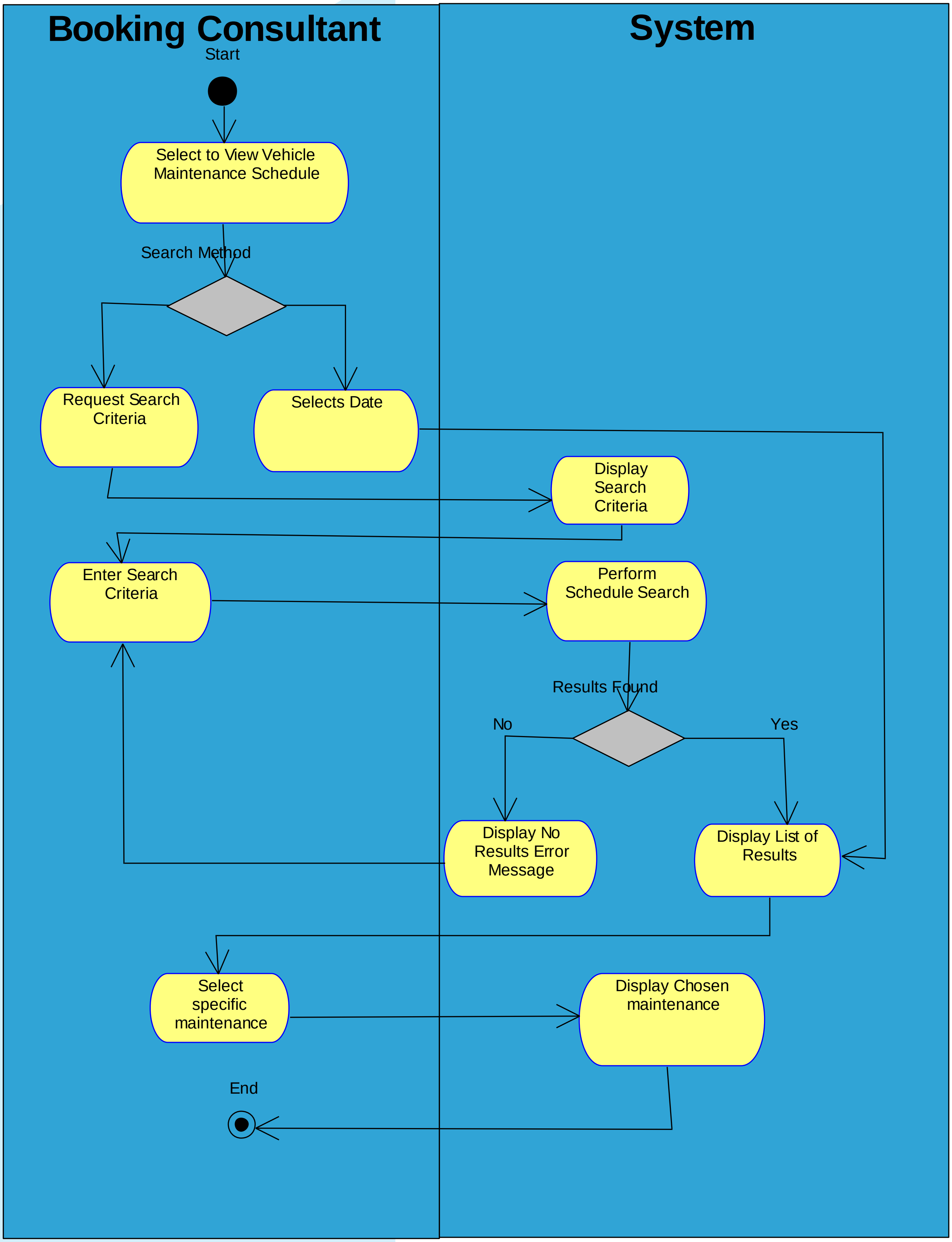

Operational Manger
System
Start
Request to Create a Booking
System request manager to choose kind of Report
Operational Manager chooses Booking Report
System requests Manager to Search by Date or by Surname
Manager Enters search data using the parameters
System reads the parameters and validates the input
Validation
No.
Yes.
Display Error Message
System returns all bookings made using searched Parameter
Display Confirmation Message
End

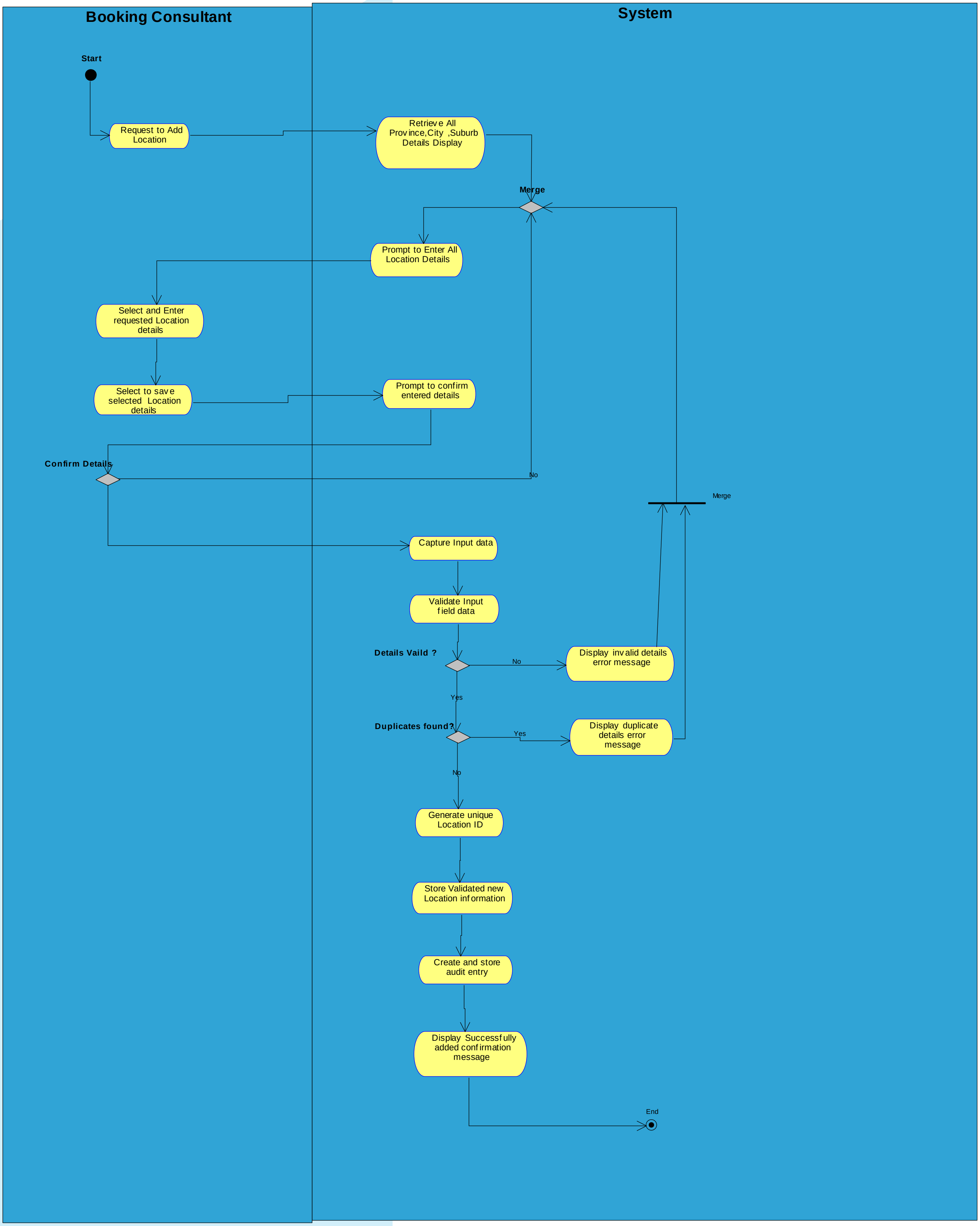

Operational Manger
System
Start
Request to Create a Booking
System request manager to choose kind of Report
Operational Manager choose Vehicle Report
System Requests Manager to Search by Date or Vehice description
Manager Enters search data using the parameters
System reads the parameters and validates the input
Validation
No
yes
Display Error Message
System returns all trips made by the vehicle
End
Display Confirmation Message

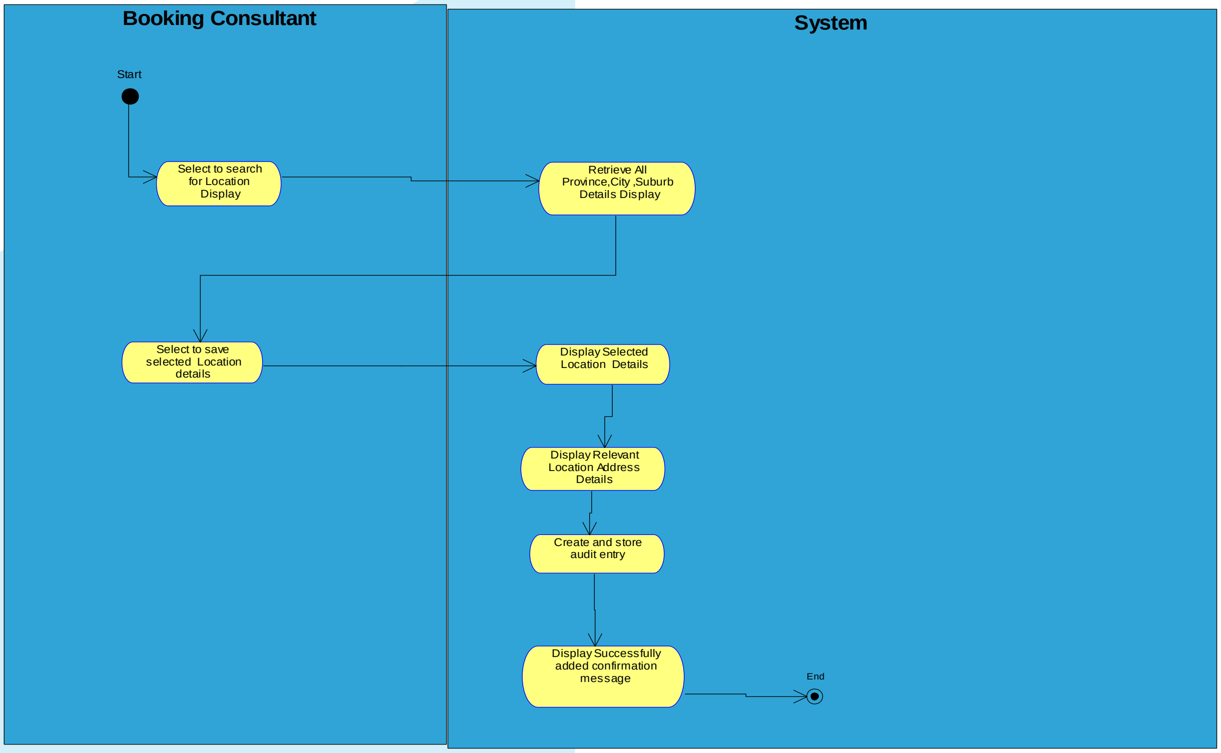

Operational Manger
Start
System
Request to Create a Mileage Report
System request manager to choose kind of Report
Operational Manager chooses Mileage Report
System requests Manager to Search by Criteria
Manager Enters search data using the criteria
System reads the parameters and validates the input
Validation
No.
Yes.
Display Error Message
System returns the Mileage of the vehicle made using searched Parameter
End
Display Confirmation Message

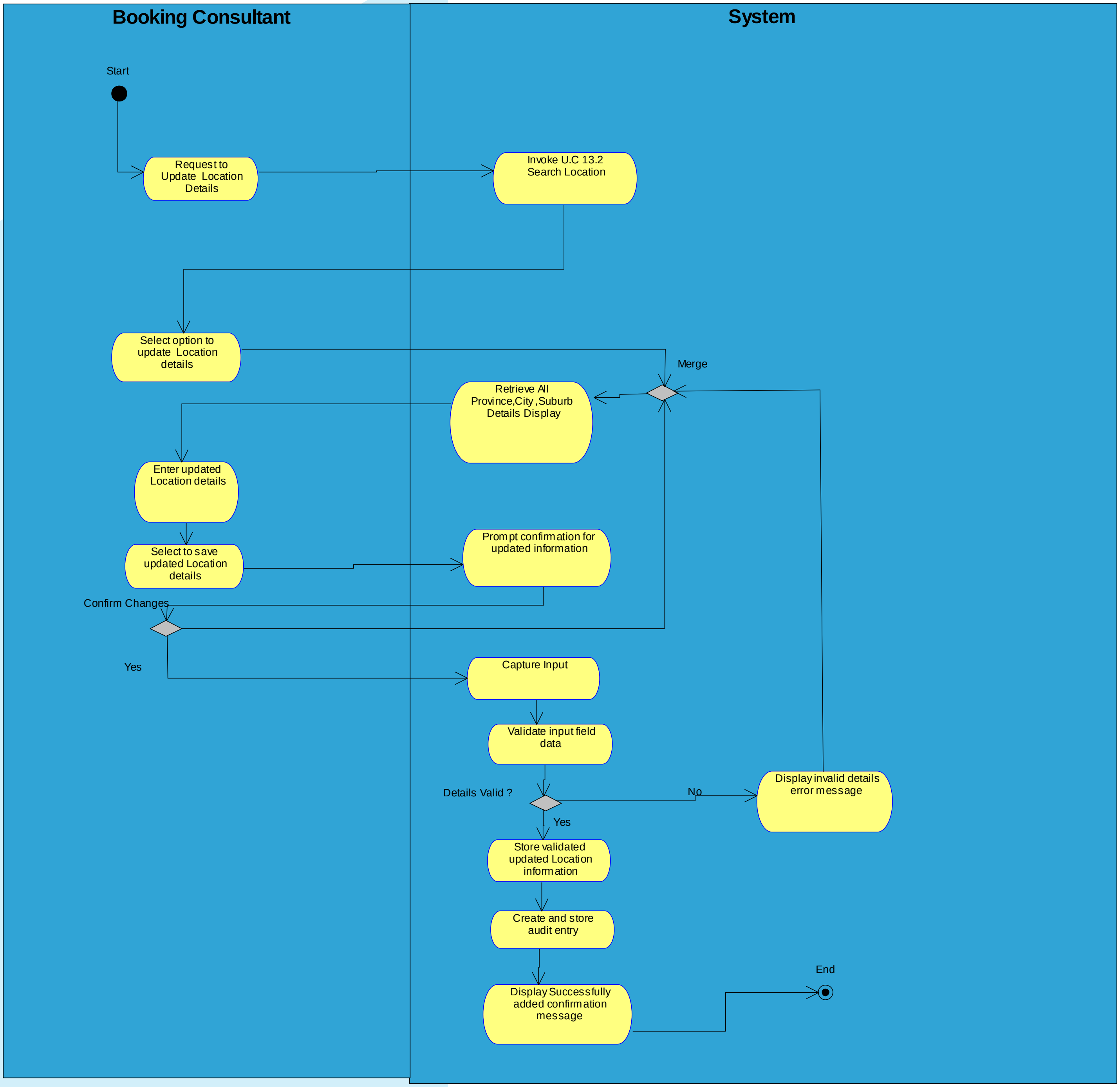

Operational Manager
System
Start.
Select to add to Maintenance schedule
Request Maintenance details
Enter Maintenance Details
Validate format of Maintenance details
Validate Details
No
Yes
Display validation error message
Generate unique number
Store maintenance details
Display Maintenance confirmation message
End

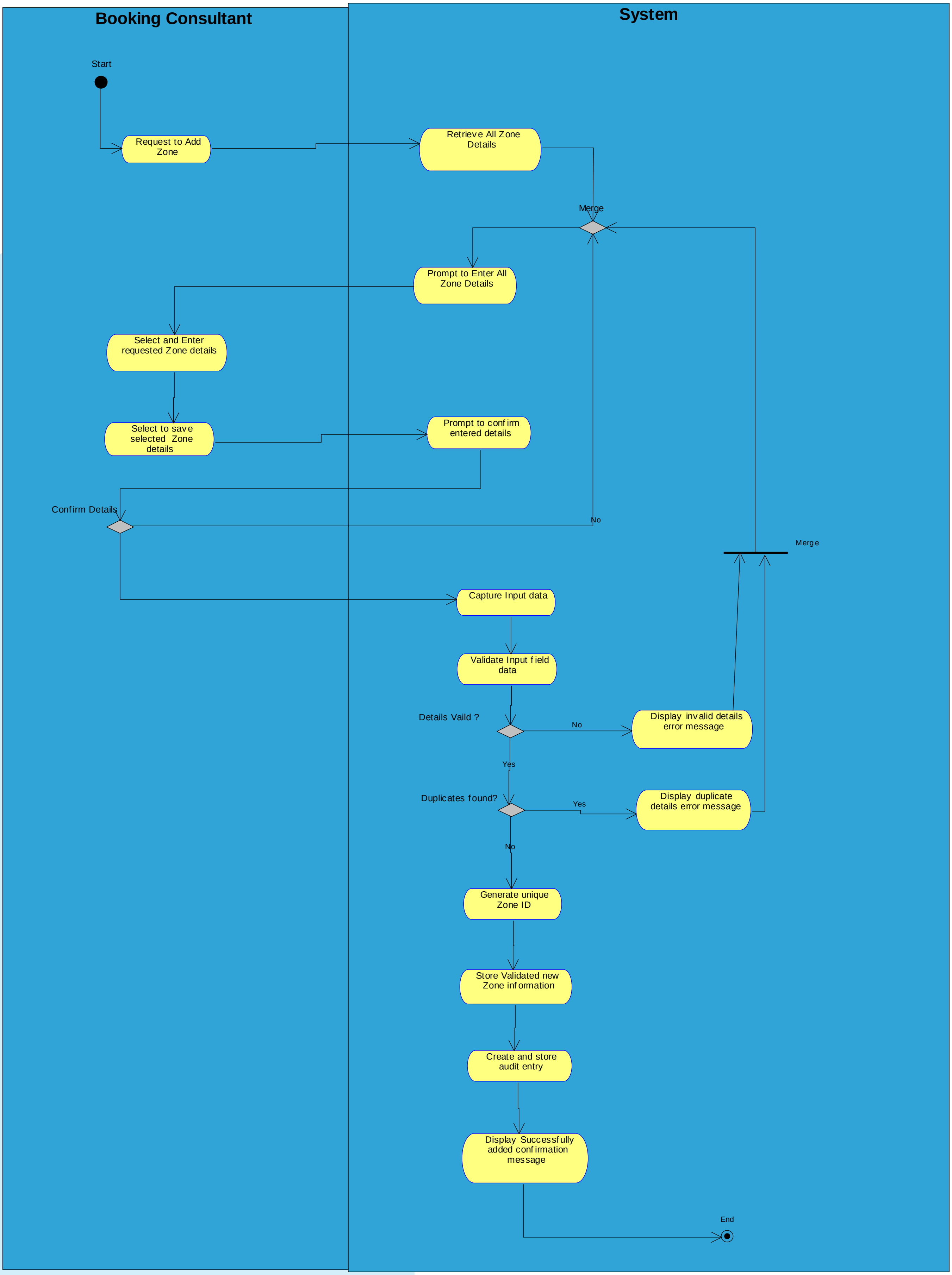

Operational Manager
System
Start
Select to Update Maintenance schedule
Invoke 12.3 View Vehicle Maintenance Schedule
Display maintenance details
Remove
Request to remove confirmation
Confirm Remove
No
Yes
Enter updated maintenance details
Confirm remove
Remove maintenance
Validate format of maintenance details
Validate details
No
Yes
Display validation error message
Store maintenance details
Display updated maintenance details
Nofity manager of successful update
End

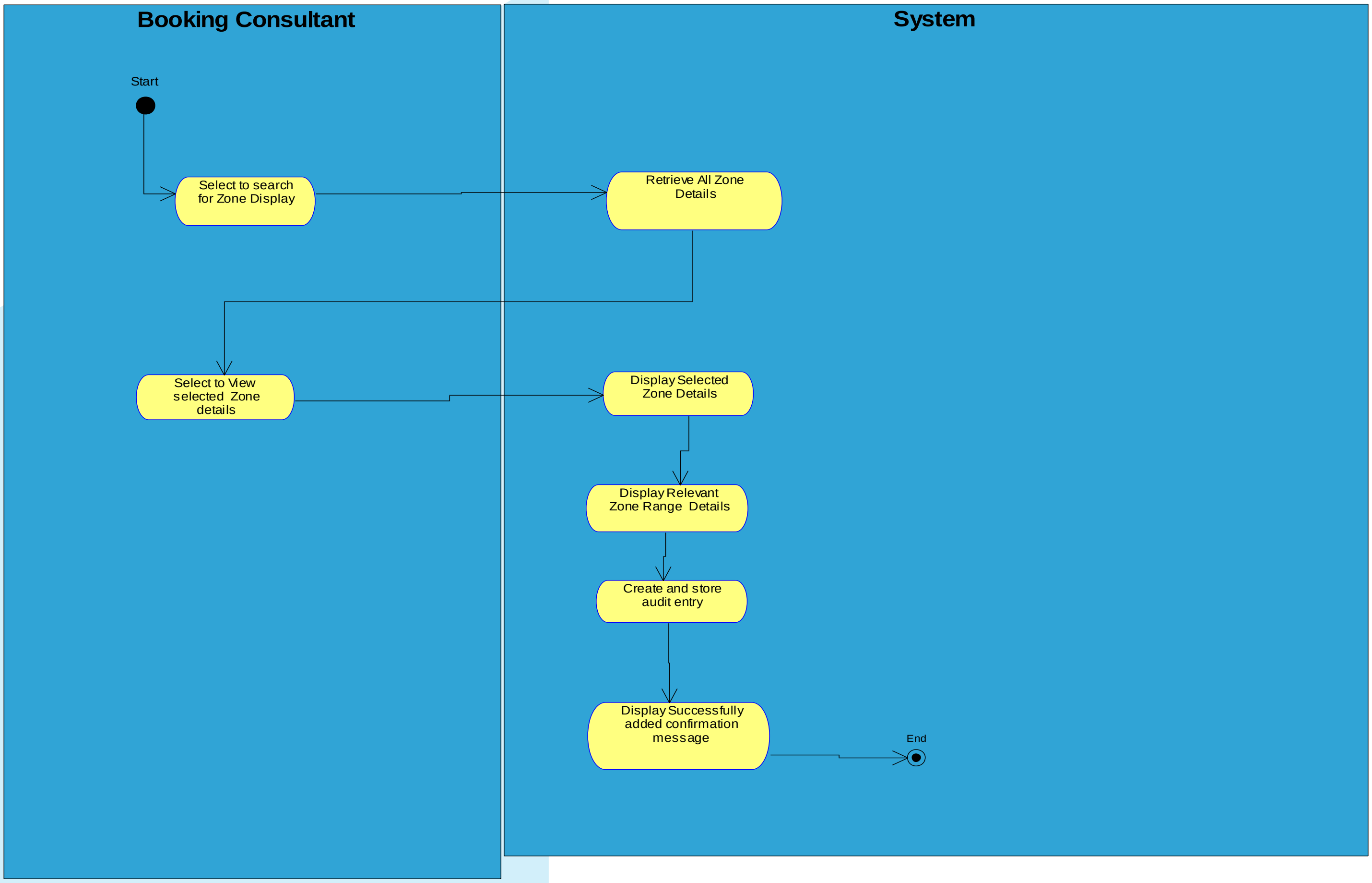

Booking Consultant
System
Start
Select to View Vehicle Maintenance Schedule
Search Method
Request Search Criteria
Selects Date
Display Search Criteria
Enter Search Criteria
Perform Schedule Search
Results Found
No
Yes
Display No Results Error Message
Display List of Results
Select specific maintenance
Display Chosen maintenance
End

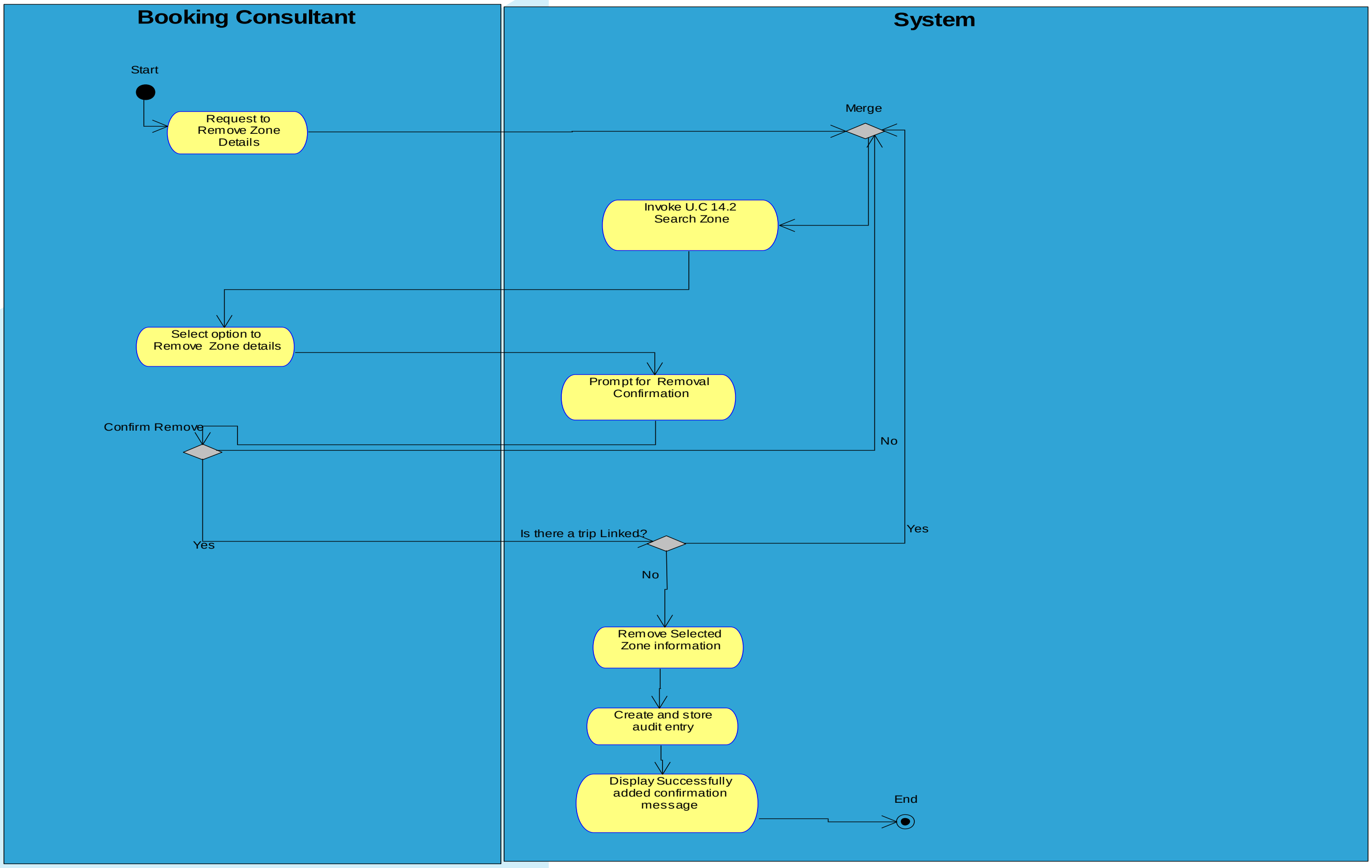

Booking Consultant
System
Start
Request to Add Location
Retrieve All Province,City ,Suburb Details Display
Merge
Prompt to Enter All Location Details
Select and Enter requested Location details
Select to save selected Location details
Prompt to confirm entered details
Confirm Details
No
Merge
Capture Input data
Validate Input field data
Details Vaild ?
No
Display invalid details error message
Yes
Duplicates found?
Yes
Display duplicate details error message
No
Generate unique Location ID
Store Validated new Location information
Create and store audit entry
Display Successfully added confirmation message
End

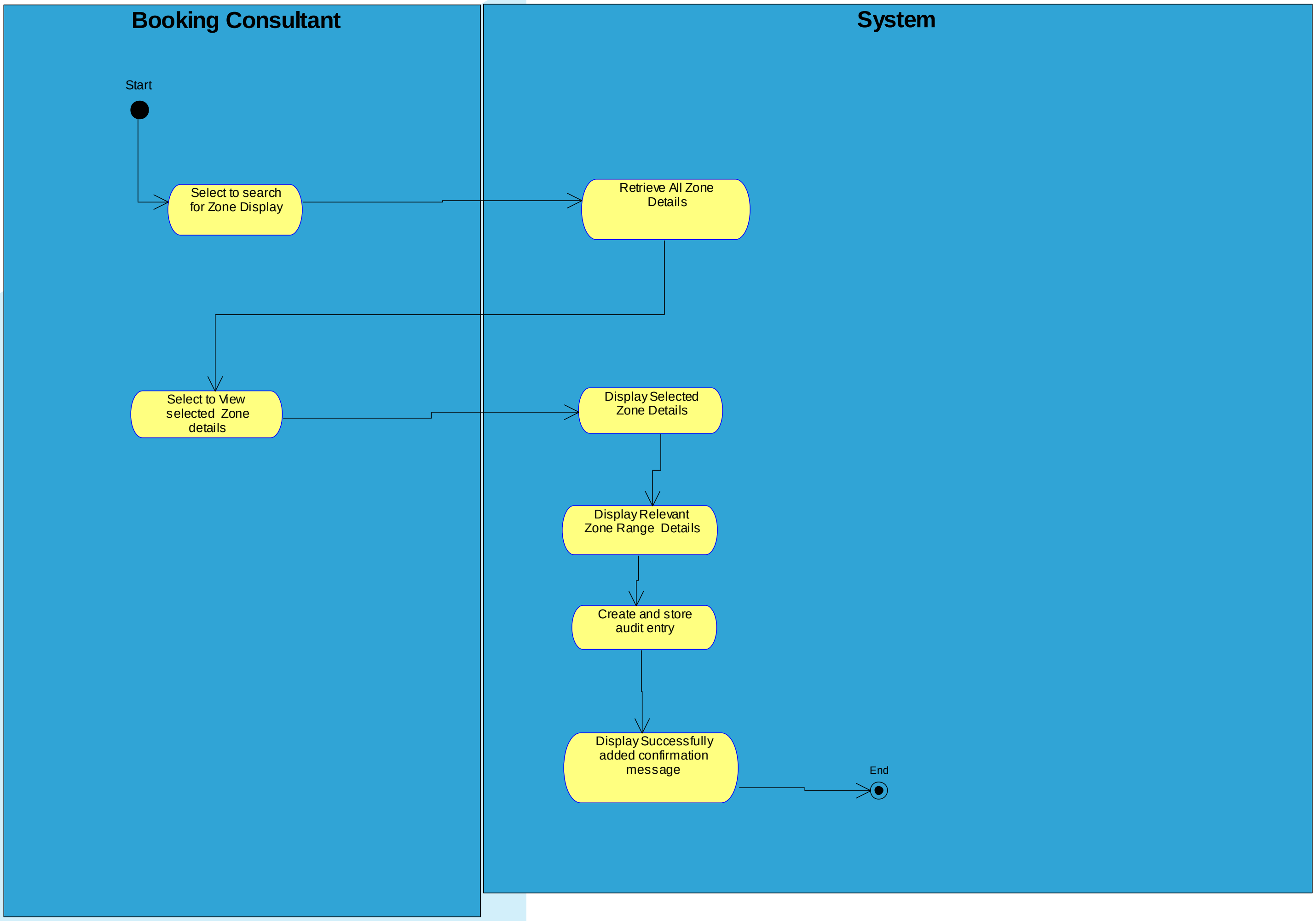

Booking Consultant
System
Start
Select to search for Location Display
Retrieve All Province,City ,Suburb Details Display
Select to save selected Location details
Display Selected Location Details
Display Relevant Location Address Details
Create and store audit entry
Display Successfully added confirmation message
End

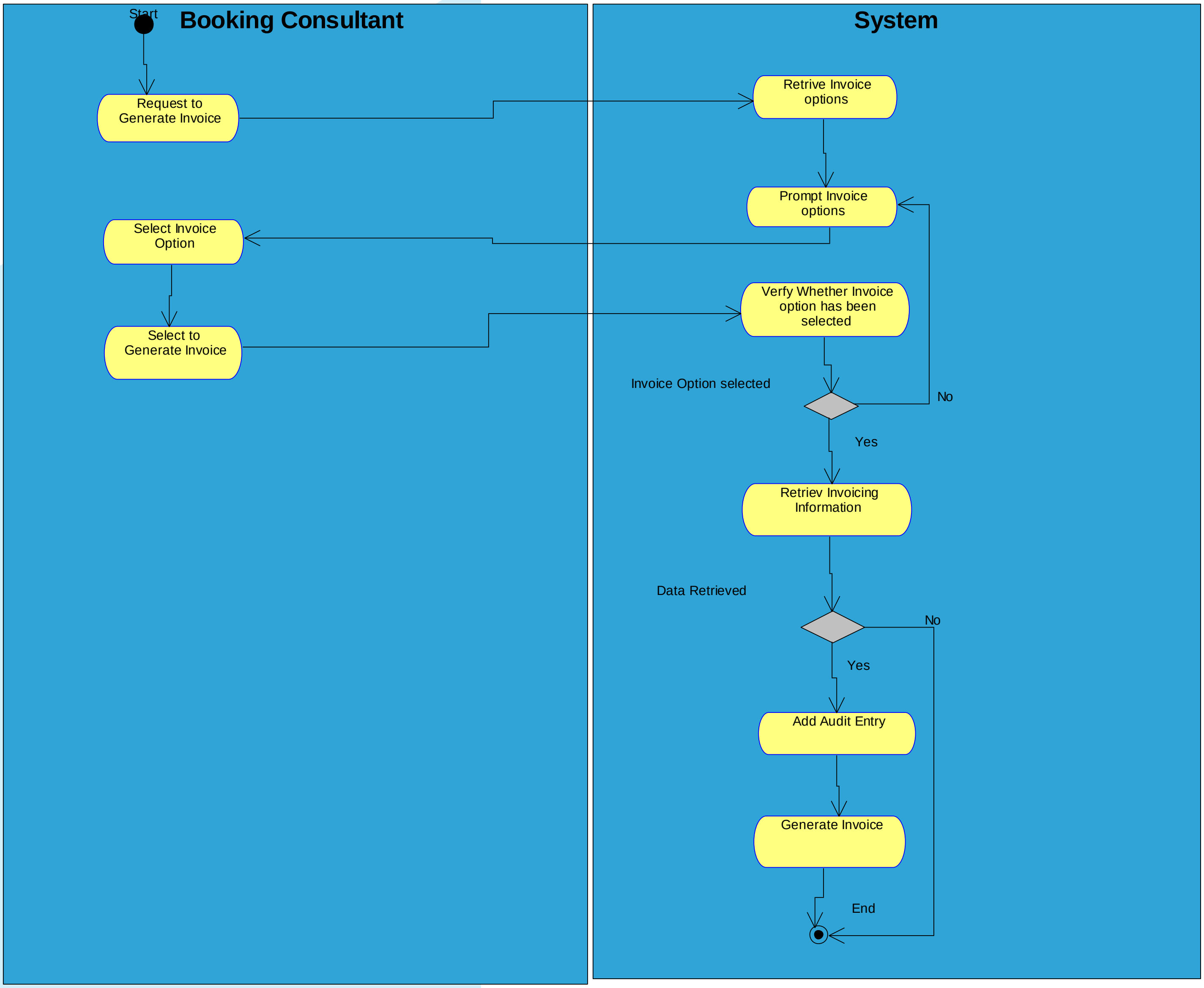

Booking Consultant
System
Start
Request to Update Location Details
Invoke U.C 13.2 Search Location
Select option to update Location details
Merge
Retrieve All Province,City ,Suburb Details Display
Enter updated Location details
Select to save updated Location details
Prompt confirmation for updated information
Confirm Changes
Yes
Capture Input
Validate input field data
Details Valid ?
No
Display invalid details error message
Yes
Store validated updated Location information
Create and store audit entry
End
Display Successfully added confirmation message


Booking Consultant
System
Start
Request to Add Zone
Retrieve All Zone Details
Merge
Prompt to Enter All Zone Details
Select and Enter requested Zone details
Select to save selected Zone details
Prompt to confirm entered details
Confirm Details
No
Merge
Capture Input data
Validate Input field data
Details Vaild ?
No
Display invalid details error message
Yes
Duplicates found?
Yes
Display duplicate details error message
No
Generate unique Zone ID
Store Validated new Zone information
Create and store audit entry
Display Successfully added confirmation message
End


Booking Consultant
System
Start
Select to search for Zone Display
Retrieve All Zone Details
Select to View selected Zone details
Display Selected Zone Details
Display Relevant Zone Range Details
Create and store audit entry
Display Successfully added confirmation message
End


Booking Consultant
System
Start
Request to Remove Zone Details
Merge
Invoke U.C 14.2 Search Zone
Select option to Remove Zone details
Prompt for Removal Confirmation
Confirm Remove
No
Yes
Is there a trip Linked?
Yes
No
Remove Selected Zone information
Create and store audit entry
Display Successfully added confirmation message
End


Booking Consultant
System
Start
Select to search for Zone Display
Retrieve All Zone Details
Select to View selected Zone details
Display Selected Zone Details
Display Relevant Zone Range Details
Create and store audit entry
Display Successfully added confirmation message
End



CONCLUSION

The section above provided an overview of how each user interacts with the system and how the system will respond when certain tasks are completed by the user.

5. DATA MODELLING

INTRODUCTION

This section contains the data model, showing all data that will be stored in the information system and the structure that the data will be placed into. The logical Entity Relationship Diagram illustrates the attributes used to describe each entity and the cardinality of the relationships between entities.

CONCLUSION

This section contained the data model, modeled using the logical Entity Relationship Diagram, representing the structure of the data in Third Normal Form.

6. INTERFACES AND OTHER INPUTS

INTRODUCTION

This section outlines the detailed interfaces and input depicted in the context diagram (See Section 3.1). This section will go into further detail showing the data used in processes from start to finish and also the description, purpose of input data, when the data will be used, the entities and attributes associated with entered data as well as the logical layout.

UseCase	Flow Line Description	Purpose	When Used	Entities and Attributes	Logical Layout
1.1.1	Add new access level	To trigger the process of adding an access level	Adding a new access level to the system		Provided by the employee
1.1.4	Enter access level details	To keep details about the new access level	Adding a new access level to the system	“AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” AccessLevel Entity	Provided by the employee
1.1.6	Store access level details	To keep the new access level in the system	Adding a new access level to the system	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” **AccessLevel Entity**	Selected by the employee
1.2.1	Update access level	To trigger the process of updating an access level	When an edit is required to an existing access level		Selected by the employee
1.2.6	Enter new details to access level	To update the access level with the new details entered	When an edit is required to an existing access level	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” AccessLevel Entity	Provided by the employee
1.2.9	Store updated details	To update the access level with the new details entered	When an edit is required to an existing access level	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” **AccessLevel Entity**	Selected by the employee

1.3.1	Request to search for an access level	To trigger the process of searching for an access level	When certain details are needed about an access level		Selected by the employee
1.3.4	Enter search details	To Know which details are to be searched for	When certain details are needed about an access level	AccessLevel_ID” “AccessLevel_Name” **AccessLevel Entity**	Provided by the employee
1.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an access level	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” AccessLevel Entity	Provided by the employee

UseCase	Flow Line Description	Purpose	When Used	Entities and Attributes	Logical Layout
2.1.1	Request to register an employee	To trigger the process of adding an employee	Adding a new employee to the system		Provided by the employee
2.1.4	Enter employee details	To keep details about the new employee	Adding a new employee to the system	"EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID" **Employee Entity**	Provided by the employee
2.1.7	Store employee details	To keep the new employee in the system	Adding a new employee to the system	"EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID" **Employee Entity**	Selected by the employee
2.2.1	Request to update employee	To trigger the process of updating an employee	When an edit is required to an existing employee		Selected by the employee
2.2.6	Enter new details to employee	To update the employee with the new details entered	When an edit is required to an existing employee	"EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID" **Employee Entity**	Selected by the employee
2.2.9	Store updated employee details	To update the employee with the new details entered	When an edit is required to an existing employee	"EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID" **Employee Entity**	Selected by the employee

2.3.1	Request to search for an employee	To trigger the process of searching for an employee	When certain details are needed about an employee		Selected by the employee
2.3.4	Enter employee search details	To Know which details are to be searched for	When certain details are needed about an employee	“EMP_Name” “EMP_Surname” “EMP_TypeID” **Employee entity**	
2.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an employee	“EMP_Name” “EMP_Surname” “EMP_TypeID” **Employee entity**	Selected by the employee

UseCase	Flow Line Description	Purpose	When Used	Entities and Attributes	Logical Layout
3.1.1	Request to register an employee type	To trigger the process of adding an employee type	Adding a new employee type to the system		Selected by the employee
3.1.4	Enter employee type details	To keep details about the new employee type	Adding a new employee type to the system	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Provided by the employee
3.1.7	Store employee type details	To keep the new employee type in the system	Adding a new employee type to the system	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Provided by the employee
3.2.1	Request to update employee type	To trigger the process of updating an employee type	When an edit is required to an existing employee type		Provided by the employee
3.2.6	Enter new details to employee type	To update the employee type with the new	When an edit is required to an existing	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Provided by the employee

UseCase	Flow Line Description	Purpose	When Used	Entities and Attributes	Logical Layout
		details entered	employee type		
3.2.9	Store updated employee type details	To update the employee type with the new details entered	When an edit is required to an existing employee type	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.3.1	Request to search for an employee type	To trigger the process of searching for an employee type	When certain details are needed about an employee type		Provided by the employee
3.3.4	Enter employee type search details	To Know which details are to be searched for	When certain details are needed about an employee type	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Provided by the employee
3.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an employee type	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.3.1	Request to remove employee type	To trigger the process of removing an employee type	So that the employee type can no longer be used anymore		Provided by the employee
3.3.5	Enter the employee type details	To know which employee type is to be removed	So that the employee type can no longer be used anymore	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Provided by the employee
3.3.8	Disable the employee type	To prevent further use of the employee type	So that the employee type can no longer be used anymore	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee

Use Case No.	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
4.1 Search driver	4.1a	Search driver prompt	Triggering input to begin the search driver process	When the booking consultant wants to search a driver	-	Requested to search by booking consultant
	4.1b	Character information	To enter character information	When the booking consultant wants to enter characters of the driver name they are searching for	DRIVER - Driver_ID - Driver_Name	Entered by booking consultant
4.2 Check driver availability	4.2a	Check driver availability prompt	To trigger the system to start the check availability process	When the booking consultant wants to check available drivers	SLOT - Slot_ID - Slot_Date - Slot_Time	Requested to check by booking consultant
4.3 Assign driver to trip	4.3a	Assign driver availability prompt	To trigger the process that assigns a vehicle to a trip	When the booking wants to confirm a booked trip	SLOT - Slot_ID - Slot_Date - Slot_Time DRIVER - Driver_ID - Driver_Name	Selected by booking consultant

4.4 Outsource driver	4.4a	Outsource driver output	To trigger the outsource a driver process	When there are no internal drivers available and the booking consultant wants to assign a driver to a trip	SLOT - Slot_ID - Slot_Date - Slot_Time OutsourceDriver - OutsourceDriver_ID - OutsourceDriver_Name	Selected by the booking consultant
5.1 Add vehicle	5.1 a	Add vehicle prompt	To begin the process that adds another vehicle to the system	When there is a new vehicle to be added to the system	VEHICLE - Vehicle_ID - Vehicle_Model - Vehicle_Colour_Description VEHICLE_MAKE - Vehicle_Make_Name, - Vehicle_Make_Description - VehicleMake_ID VEHICLE MAINTENANCE - VehicleMaintenance_ID - ServiceProvider - Vehicle_ID	Requested by the booking consultant
5.2 Search vehicle	5.2a	Search vehicle prompt	Triggers the system to begin the process that searches a vehicle	When the booking consultant wants to search for a vehicle	VEHICLE - Vehicle_ID - Vehicle_Model - VehicleMake_ID - VehicleMaintenance_ID - VehicleColour_ID - VehicleLicencePlates	Requested by the booking consultant
5.3 Confirm trip	5.3a	Confirm trip prompt	This starts the system process which confirms a booking	When the booking consultant wants to send confirmation messages of trip to	SLOT - BookingReference_ID - DeparturePoint - DestinationPoint - Slot_Time PASSENGER - Passenger_ID - Passenger_Name	Requested by the booking consultant

				the driver and the passenger	- Passenger_Contact **DRIVER** - Driver_ID - Driver_Name - Driver_Contact	
5.4 Check vehicle availability	5.4a	Check driver availability prompt	This triggers the process that checks the drivers that are available	When a list of available vehicles	**BOOKING** - BookingReference_ID - Passenger_ID - Driver_ID - DeparturePoint - DestinationPoint - Time **VEHICLE GROUP** - Number_of_Passengers **SLOT** - Slot_Date - Slot_Time	Selected by the bookng consultant
5.5 Assign vehicle to trip	5.5a	Assign vehicle prompt	This begins the process that assigns a vehicle to a trip	When the booking consultant wants to confirm a booking and needs to assign a vehicle to a trip	**SLOT** - Slot_ID - Slot_Date - Slot_Time	Requested by the booking consultant
	5.5b	Selected vehicle information	This triggers the process the captures the details of a chosen vehicle	When the booking consultant selects a vehicle they want to assign to a trip	**SLOT** - Slot_ID - Slot_Date - Slot_Time	Requested by the booking consultant
5.6 Outsource vehicle	5.6a	Outsource vehicle prompt	Triggers the process that assigns an	When the booking consultant wants to	**SLOT** - Slot_ID - Slot_Date - Slot_Time	Requested by the booking

			outsourced driver to a trip	confirm a booking and assigns an outsourced driver	OUTSOURCE_VEHICLE - OutsourceVehicle_ID - OutSource_Vehicle_Model - OutSource_Make	consultant
6.1 Create vehicle group activity	6.1a	Add new vehicle group prompt	This begins the process of adding a new vehicle group in the system	When the booking consultant wants to add a new vehicle group into the system	VEHICLE GROUP - Vehicle group ID, - VehicleGroup_Name - VehicleGroup_Description	Requested by the booking consultant
	6.1b	New vehicle group information	This begins the process that captures the new vehicle group information	When a new vehicle group is to be added by the booking consultant	VEHICLE GROUP - Vehicle group ID, - VehicleGroup_Name - VehicleGroup_Description	Requested by the booking consultant
6.2 Search vehicle group	6.2a	Search vehicle group prompt	This triggers the process that searches for a group activity	When the booking consultant wants to search for a vehicle group	VEHICLE GROUP - Vehicle group ID, - VehicleGroup_Name - VehicleGroup_Description	Requested by the booking consultant

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
7.1 Add Client	Request to add new client	Contains the details used to initiate the Add	When the booking consultant adds the new Client to		Provided by the client

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
		Client process	the system		
	New Client Details	Contains new client details	When the Booking consultant captures the details of the new client	Entity:Client Attributes Client_Name Client_Email Client_Tel Client_Reference Client_Address Client_Type	Provided by the client
7.2 Search Client	Request to Search a client	Contains details used to initiate client search	When the Booking consultant wants to search a client	None	Selected by the Booking consultant
	Client Name	To retrieve a client being searched	The booking consultant enters the name of the client	Entity: Client Client_Name Client_Email Client_Tel Client_Reference Client_Address Client_Type	Provided by the Booking consultant
	Select Specific client	The option select the specific client they want	When the booking consultant wants to select the specific client and view details of the client		Selected by the Booking consultant
7.3 Update Client	Request to update client details	Contains all the details needed to initiate the update client details	When the Booking consultant wants to update the client details		Provided by the client
	The Updated Client details	Contains all the updated client details	When the Booking consultant enters the Updated client details		Selected by the booking consultant

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
	Confirm Updated Details	The Booking consultant selects confirm the changes made	When the Booking Consultant wants to confirm the changes made	Entity: Client Client_Name Client_Email Client_Tel Client_Reference Client_Address Client_Type	Selected by the booking consultant
8.1 Add ClientType	Request to add new clientType	Contains the details used to initiate the Add ClientType process	When the Operation Manager adds the new ClientType to the system		Operation Manager Provides the details
	New ClientType Details	Contains new clientType details	When the Operations Manager captures the details of the new clientType	Entity:Client_Type Attributes **Client_Type_Name** **Client_Type_Description**	Operations Manager Provides the details
8.2 Search ClientType	Request to Search a clientType	Contains details used to initiate clientType search	When the Booking consultant wants to search a clientType	None	Provided by the Booking consultant
	ClientType Name	To retrieve clientType being searched	The booking consultant enters the name of the clientType	Entity:Client_Type Attributes **Client_Type_Name** **Client_Type_Description**	Provided by the Booking consultant
	Select Specific clientType	The option selects the specific clientType they want	When the booking consultant wants to select the specific clientType and		Selected by the Booking consultant

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
			view details of the clientType		

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
8.3 Update ClientType	Request to update clientType details	Contains all the details needed to initiate the update clientType details	When the Booking consultant wants to update the clientType details		Selected by Booking consultant
	The Updated ClientType details	Contains all the updated clientType details	When the Booking consultant enters the Updated clientType details	Entity:Client_Type Attributes **Client_Type_Name** **Client_Type_Description**	Provided by the Booking consultant
	Confirm Updated Details	The Booking consultant selects confirm the changes made	When the Booking Consultant wants to confirm the changes made	Entity:Client_Type Attributes **Client_Type_Name** **Client_Type_Description**	Selected by the Booking consultant

Use case	Description Flow Line	Purpose	When will it be used	Entities and Attributes	Logical Layout
9.1 Create Booking	Request to Create a booking	Contains all the details needed to initiate	When the Booking Consultant wants to		Provided by client

Use case	Description Flow Line	Purpose	When will it be used	Entities and Attributes	Logical Layout
		Create Booking Process	create a Booking		
	Client Name/Reference	Contains details of the Client making a booking	When the consultant wants to retrieve details of the client		Provided by client
	[ALT] Add new Client	Contains all the details of the new Client	When the booking consultant wants to create new Client		Selected by the Booking consultant
	The Pickup and dropOff details Details	Contains all the required Pickup and dropOff details	When the Booking consultant wants to generate a quote for booking	Entity:Client_Type Attributes **Client_Type_Name** **Client_Type_Description**	Provided by client
	Passenger Details	Contains all the required to create a new Passenger	When the Booking consultant wants to enter details of the Passenger	**Entity: Passenger** **Passenger Name** **Passenger Surname** **Passenger PhoneNo**	Provided by client
	PickUp Instruction	Contains all the required details needed for instruction	When the manager wants to enter the instructions of PickUp		Provided by client
	Compiled Booking Details	Contains all the details required to create a new Booking	When the manager wants to enter the Booking details	**Entity: Booking_Trip** **Date_of_PickUp** **Date_of_Arrival** **Time_Of_PickUp** **Time_Of_Arrival** **Booking_Status** **Pickup_Location_ID** **DropOff_Location_ID** **Number_Of_Passengers** **PickUp_Instruction** **Entity : Invoice** **Invoice_Number**	System generated a compiled booking details

Use case	Description Flow Line	Purpose	When will it be used	Entities and Attributes	Logical Layout
				Invoice_Date **Payment_Status** **Booking_Reference** **Booking_Cost**	
9.2 Search Booking	Request to Search a Booking	Contains details used to initiate Search Booking	When the Booking consultant wants to search a Booking	None	Selected by Booking Consultant
	Booking Reference	To retrieve Booking being searched	The booking consultant enters the Booking reference	Entity:Booking_Trip Attributes **Booking_Reference**	Provided by Booking consultant
	Select Specific Booking	The option selects the specific Booking they want	When the booking consultant wants to select the specific Booking and view details		Selected by Booking consultant
9.3 Update Booking	Request to update Booking details	Contains all the details needed to initiate the update Booking details	When the Booking consultant wants to update the Booking details		Provided by the client
	The Updated Booking details	Contains all the updated Booking details	When the Booking consultant enters the Updated Booking details		Provided by the client
	Confirm Updated Details	The Booking consultant selects confirm the changes made	When the Booking Consultant wants to confirm the changes made	**Entity: Booking_Trip** **Date_of_PickUp** **Date_of_Arrival** **Time_Of_PickUp** **Time_Of_Arrival** **Booking_Status** **Pickup_Location_ID** **DropOff_Location_ID** **Number_Of_Passengers**	System confirms the updated details

Use case	Description Flow Line	Purpose	When will it be used	Entities and Attributes	Logical Layout
				PickUp_Instruction **Entity : Invoice** **Invoice_Number** **Invoice_Date** **Payment_Status** **Booking_Reference** **Booking_Cost**	
9.4 Cancel Booking	Request to Cancel a Booking	The Booking consultant selects the option to cancel the Booking	When the Booking consultant wants to cancel a Booking		Provided by the client
	Booking Reference	Contains the details of the Booking that needs to be canceled	When the Booking consultant wants to retrieve the Booking to cancel	Entity: Booking_Trip Attributes: Booking_Reference	Provided by the client
	Cancellation Reasons	Contains the details of cancelling the Booking	When the Booking consultant want to enter the Cancellation reasons		Provided by the client
9.5 Confirm Booking	Request to Confirm a Booking	The Booking consultant selects to confirm a Booking	When the Booking consultant wants to confirm a Booking		Email
	Booking Reference	Contains the details of the Booking that needs to be Updated	When the Booking consultant wants to retrieve the Booking to Updated		Retrieved from email
	Select a Vehicle	Contains details needed to	When the Booking consultant wants to	Entity:Vehicle Attributes Vehicle_ID	Selected by Booking

Use case	Description Flow Line	Purpose	When will it be used	Entities and Attributes	Logical Layout
		select a Vehicle	assign a vehicle to a Booking	Entity: Slot Slot_Date Slote_Time	consultant
	Select a Driver	Contains details needed to select a Driver	When the Booking consultant wants to assign a Driver to a Booking	Entity:Driver Attributes Driver Entity: Slot Slot_Date Slote_Time	Selected by Booking consultant

UseCase	Flow Line	Description	Purpose	When Used	Entities and Attributes	Logical Layout
10.1	Add to Schedule	Notification of Scheduling	Notifies the manager of any maintenances or maintenance appointments that are to the schedule.	When the booking consultant wants to add a booking to the schedule	N/A	Notification the system
		ALT - Invalid Appointment/Maintenance Details Error Message	Notifies the manager that invalid appointment or maintenance details are inserted when adding to the schedule.	When invalid booking details are inserted when adding to the schedule.	N/A	A message that notifies the manager that invalid Booking details were received

UseCase	Flow Line	Description	Purpose	When Used	Entities and Attributes	Logical Layout
10.2	Update Schedule	Updatable Schedule Details Edit Fields	Displays the updatable schedule details edit fields.	When a manager updates the schedule.	N/A	Displayed on the system.
		ALT - Schedule Details Validation Error Message	Notifies the manager that there is a validation error with the schedule details.	When invalid details are inserted when maintaining a schedule.	N/A	A message that notifies the consultant that invalid details were inserted when maintaining the schedule.
		ALT - Request Confirmation of Schedule Removal	Prompts the consultanat for confirmation on removing a schedule.	When the consultant removes a schedule.	N/A	Prompts displayed on the system.
		Notification of Successful Schedule Details Update or Removal	Notifies the consultant that schedule details were successfully updated or removed.	When schedule details are successfully updated or removed.	N/A	Notification on the system.
10.3	View Schedule	Displayed List of Schedule Search Results	Displays a list of schedules that were	When the booking	**Slot** table: • **BookingRefernce** • **Client_ID**	Displayed on the system.

UseCase	Flow Line	Description	Purpose	When Used	Entities and Attributes	Logical Layout
			searched for by the consultant.	consultant wants to search for a booking on the schedule	- Driver_ID - Date - Time - ContactPerson_ID - DestinationLocation - TripDuration	
		Displayed Specific Search Result Details	Displays specific search result details that the consultanat searched for.	When the consultant wants to view the details of a specific Booking in the schedule	**Slot** table: - BookingRefernce - Client_ID - Driver_ID - Date - Time - ContactPerson_ID - DestinationLocation - TripDuration	Displayed on the system.
		ALT - Invalid Schedule Search Criteria Error Message	Notifies the consultant that invalid search criteria were entered when viewing the schedule.	When the consultant enters invalid data when viewing a schedule.	**N/A**	A message that notifies the consultant that invalid details were inserted viewing the schedule.
11.1	Booking Report					
11.2	Driver-Trip Report					
11.3	Mileage Report					

UseCase	Flow Line	Description	Purpose	When Used	Entities and Attributes	Logical Layout
12.1	Schedule Vehicle Maintenance Schedule	Notification of Scheduling	Notifies the manager of any maintenances that are to the schedule.	When a maintenance is added to the schedule.	None	Notification on the system.
		ALT – Invalid Maintenance Details Error Message	Notifies the manager that invalid maintenance details are inserted when adding to the schedule.	When invalid maintenance details are inserted when adding to the schedule.	None	A message that notifies the manager that invalid maintenance details were received.
12.2	Update Vehicle Maintenance Schedule	Updatable Schedule Details Edit Fields	Displays the updatable schedule details edit fields.	When an operational manager maintains the schedule.	None	Displayed on the system.
		ALT - Schedule Details Validation Error Message	Notifies the operational manager that there is a validation error with the schedule details.	When invalid details are inserted when maintaining a schedule.	None	A message that notifies the operational manager that invalid details were inserted when maintaining the schedule.
		ALT - Request Confirmation of Schedule Removal	Prompts the operational manager for	When the operational manager removes a schedule.	None	Prompts displayed on the system.

UseCase	Flow Line	Description	Purpose	When Used	Entities and Attributes	Logical Layout
			confirmation on removing a schedule.			
		Notification of Successful Schedule Details Update or Removal	Notifies the operational manager that schedule details were successfully updated or removed.	When schedule details are successfully updated or removed.	None	Notification on the system.
12.3	View Vehicle Maintenance Schedule	Displayed List of Schedule Search Results	Displays a list of schedules that were searched for by the operational manager.	When the operational manager searches for schedules.		Displayed on the system.
		Displayed Specific Search Result Details	Displays specific search result details that the operational manager searched for.	When the operational manager searches for a specific schedule.		Displayed on the system.
		ALT - Invalid Schedule Search Criteria Error Message	Notifies the operational manager that invalid search	When the operational manager enters invalid data when	N/A	A message that notifies the operational manager that invalid details were

UseCase	Flow Line	Description	Purpo se	When Used	Entities and Attributes	Logical Layout
			criteria were entered when viewing the schedule.	viewing a schedule.		inserted viewing the schedule.

UseCas e Numb er	Flow Line	Description	Purpos e	When it will be Used	Entities and Attributes	Logical Layout
13.1 Add Location	**13.1.1**	Request Add Location Details	Triggering input to start the adding Location process.	When a new Locati on isto be added onthe system.	Booking Consultant (Location) • Location name [**Location_name**] • Province name [**Province_ name**] • City [**City_Nam e**] • Suburb [**Suburb_n ame**] • Street [**Street_na me**] • LocationType [**Locationtype]**	Requested by Booking Consultant .
	13.1.2	Location Details	Details of new Location to be used to add new Location .	When a new Locati on isto be added on the syste m.	• Location name [**Location_name**] • Province name [**Province_ name**] • City [**City_Nam e**] • Suburb	Entered by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
					[**Suburb_name**] • Street [**Street_name**] • LocationType [**Locationtype**]	
	13.1.3	Save Location Details	The Booking Consultant indicates selection to save details for new Location .	When a new Location isto be added on the system.	Booking Consultant (Location) • Location name [**Location_name**] • Province name [**Province_name**] • City [**City_Name**] • Suburb [**Suburb_name**] • Street [**Street_name**] • LocationType [**Locationtype**]	Selected by Booking Consultant .
	13.1.4	Confirm Location Details	The Booking Consultant indicates confirmation of addingnew Location .	When the system prompts the Booking Consultant to confirmnew Location details.	Booking Consultant (Location)	Selected by Booking Consultant .
	13.1 [ALT]	Revoked Confirmation Details	The Booking Consultant revokes confirmation of adding new Location .	When the system prompts the Booking Consultant to confirmnew Location	Booking Consultant (Location)	Selected by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
				details.		
13.2 Search Location	**13.2.1**	Request Search Location Details.	Triggering input to startthe Location Viewing process.	When a specific Location is to be viewed on the system.	Booking Consultant (Location) • Location name [**Location_name**] • Province name [**Province_name**] • City [**City_Name**] • Suburb [**Suburb_name**] • Street [**Street_name**] • LocationType **[Locationtype]**	Requested by Booking Consultant .
	13.2.2	Request to Search Specific Location Details.	The Booking Consultant selects the Location to be viewed on the system.	When a specific Location is to be viewed on the system.	Booking Consultant (Location) • Location name [**Location_name**] • Province name [**Province_name**] • City [**City_Name**] • Suburb [**Suburb_name**] • Street [**Street_name**]	Selected by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
					• LocationType **[Locationtype]**	
13.13 Update Location	**13.3.1**	Request Update Location Details.	Triggering input to start the updating Location process.	When the Location details need to change on the system.	Booking Consultant (Location)	Requested by Booking Consultant .
	13.3.2	Update Location Details.	The Booking Consultant selects which Location to update details	When the Location details need to change on the system.	Booking Consultant (Location) • Location name [**Location_name**] • Province name [**Province_name**] • City [**City_Name**] • Suburb [**Suburb_name**] • Street [**Street_name**] • LocationType **[Locationtype]**	Selected by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
	13.3.3	Updated Location Details.	Details of Location to be used for updating details of Location .	When the Location details need to change on the system.	• Location name **[Location_name]** • Province name **[Province_name]** • City **[City_Name]** • Suburb **[Suburb_name]** • Street **[Street_name]** • LocationType **[Locationtype]**	Entered by Booking Consultant .
	13.3.4	Save Updated Location Details.	The Booking Consultant indicates selection to save details to be updated on the system.	When the Location details need to change on the system.	Booking Consultant(Location). • Location name [Location_name] • Province name [Province_name] • City [City_Name] • Suburb [Suburb_name] • Street [Street_name] • LocationType [Locationtype]	Selected by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
14.1 Add Zone	**14.1.1**	Request Add Zone Details	Triggering input to start the adding Zone process.	When a new Zone is to be added onthe system.	Booking Consultant (Zone) • Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Requested by Booking Consultant .
	14.1.2	Enter Zone Details	Details of new Zone to be used to add new Zone .	When a new Zone is to be added on the system.	• Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Entered by Booking Consultant .
	14.1.3	Save Zone Details	The Booking Consultant indicates selection to save details fornew Zone .	When a new Zone is to be added on the system.	Booking Consultant (Zone) • Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Selected by Booking Consultant .
	14.1.4	Confirm Zone Details	The Booking Consultant indicates confirmation of addingnew Zone .	When the system prompts the Booking Consultant to confirmnew Zone details.	Booking Consultant (Zone)	Selected by Booking Consultant .
	14.1 [ALT]	Revoked Confirmation Details	The Booking Consultant revokes confirmation of adding new Zone .	When the system prompts the Booking Consultant to confirmnew Zone details.	Booking Consultant (Zone)	Selected by Booking Consultant .
14.2 Search Zone	**14.2.1**	Request Search Zone Details.	Triggering input tostart theZone Viewing process.	When a specific Zone is to be viewed onthe system.	Booking Consultant (Zone) • Zone Type **[Zone type]**	Requested by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
	14.2.2	Request to Search Specific Zone Details.	The Booking Consultant selects the Zone to be viewed on the system.	When a specific Zone is to be viewed on the system.	Booking Consultant (Zone) - Zone Id **[Zone_ID]** - Zone name **[Zone_name]** - Zone range **[zone_range]**	Selected by Booking Consultant .
13.3 Update Zone	**14.3.1**	Request Update Zone Details.	Triggering input to start the updating Zone process.	When the Zone details need to change on the system.	Booking Consultant (Zone)	Requested by Booking Consultant .
	14.3.2	Update Zone Details.	The Booking Consultant selects which Zone to update details	When the Zone details need to change on the system.	Booking Consultant (Zone) - Zone Id **[Zone_ID]** - Zone name **[Zone_name]** - Zone range **[zone_range]**	Selected by Booking Consultant .
	14.3.3	Updated Zone Details.	Details of Zone to be used for updating details of Zone .	When the Zone details need to change on the system.	- Zone Id **[Zone_ID]** - Zone name **[Zone_name]** - Zone range **[zone_range]**	Entered by Booking Consultant .
	14.3.4	Save Updated Zone Details.	The Booking Consultant indicates selection to save details to be updated on the system.	When the Zone details need to change on the system.	Booking Consultant **(Zone).** - Zone Id **[Zone_ID]** - Zone name **[Zone_name]** - Zone range **[zone_range]**	Selected by Booking Consultant .

UseCase Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
	14.3.5	Confirm Zone Details.	The Booking Consultant indicates confirmation of updating Zone .	When the system prompts the Booking Consultant b confirm updated details.	Zone Booking Consultant	Selected by Zone Booking Consultant .
	14.3 [ALT]	Revoke Confirmation Details.	The Booking Consultant revokes confirmation of updating Zone .	When the system prompts the Booking Consultant b confirm updated details.	Zone Booking Consultant	Selected by Zone Booking Consultant .
13.4Remove Zone	**14.4.1**	Request Removal of Zone Details	Triggering input to start the Zone Removal process.	When a needs to be removed from the system.	Zone Booking Consultant	Requested by Zone Booking Consultant .
	14.4.2	Request Removal of Selected Zone .	TheBooking Consultant selects which Zone to be removedfrom the system.	When a Zone needs to be removed from the system.	Zone Booking Consultant • Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Selected by Zone Booking Consultant .
	14.4.3	Confirmation to Remove Zone Details.	TheBooking Consultant indicates confirmation of removal of Zone .	When the system prompts the Booking Consultant b confirm removal of Zone .	Zone Booking Consultant	Selected by Zone Booking Consultant .
	14.4 [ALT]	Revoke Confirmation of Removal Details.	The Booking Consultant revokes confirmation of removal of Zone .	When the system prompts the Booking Consultant b confirm removal of Zone .	Zone Booking Consultant	Selected by Zone Booking Consultant .

Use Case Number	Flow Line	Description	Purpose	When it will be Used	Entities and Attributes	Logical Layout
15.1.1Generate Invoice	**15.1.1**	Request to Generate Invoice .	Triggering input to start the Invoiceing process.	When the Booking Consultant wants the Invoice to be generated.	-	Requested by Booking Consultant.
	15.1.2	Selected Booking .	Booking Consultant selects the Booking for the Invoice to be based on.	When the Booking for the Invoice needs to chosen.	Booking • Booking Name	Selected by booking consultant
	15.1.3	Select to Generate Invoice.	Booking Consultant selects the option to generate the Invoice.	When the Booking Consultant wants the Invoice to be generated.	-Invoice • Invoice _ID • Booking_Refference • Invoice _PaymentStatus • Invoice_date • Invoice_Quote	Selected by Booking Consultant.

CONCLUSION

This section outlined in detail the system interfaces and input data depicted in the context diagram. The section went into detail discussing the data used in processes, description of data, purpose of input data, when the data will be used, the entities and attributes associated with entered data as well as the logical layout.

7. REPORTS AND OTHER OUTPUTS

INTRODUCTION

The following section construes each and every single output generated by our system. A table is created describe the purpose of every output, when will outputs will be used, entities and attributes involved and the logical layout of the output. Outputs to external applications are also included in this section.

OUTPUTS

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
1.1.3	Requests access level details	To trigger the process of adding an access level	Adding a new access level to the system		Selected by the employee
1.1.2 [Alt]	Notification failed message	To give notification message to the employee	Adding a new access level to the system		Selected by the employee
1.1.5 [Alt]	Notification failed verification message	To give notification message to the employee	Adding a new access level to the system		Selected by the employee
1.1.7	Notification success message	To give notification message to the employee	Adding a new access level to the system	AccessLevel_ID	Selected by the employee
1.2.3	Display access levels	To all the employee to select an access level	When an edit is required to an existing access level	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” AccessLevel Entity	Selected by the employee
1.2.8 [Alt]	Notification failed verification message	To give notification message to the employee	When an edit is required to an existing access level		Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
1.2.10	Notification success message	To give notification message to the employee	When an edit is required to an existing access level	AccessLevel_ID” “	Selected by the employee
1.2.8 [Alt]	Notification failed verification message	To give notification message to the employee	When an edit is required to an existing access level		Selected by the employee
1.3.3	Request search details	To Know which details are to be searched for	When certain details are needed about an access level	“AccessLevel_Name” **AccessLevel Entity**	Selected by the employee
1.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an access level	AccessLevel_ID” “AccessLevel_Name” “AccessLevel_Description” AccessLevel Entity	Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
2.1.4	Request employee details	To keep details about the new employee	Adding a new employee to the system	“EMPID” “EMP_Name” “EMP_Surname” “AuditID” “EMP_TypeID” “EMP_EmailAddress” “EMP_Contact” “EMP_Password” “EMP_IDNum” “TitleID” “GenderID” **Employee Entity**	Selected by the employee
2.1.8	Display success notification	To notify the employee of the success	Adding a new employee to the system	“EMPID” “EMP_Name” **Employee Entity**	Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
2.1.2 [Alt}	Notification message of no authority	To notify the employee that they don't have the authority	Adding a new employee to the system		Selected by the employee
2.1.6 [Alt]	Notification of failure to verify message	To notify the employee that the details entered aren't correct	Adding a new employee to the system		Selected by the employee
2.2.5	Display update employee details	To show the employee the details that can be updated	When an edit is required to an existing employee	"EMPID" "EMP_Name" "EMP_Surname" "AuditID" "EMP_TypeID" "EMP_EmailAddress" "EMP_Contact" "EMP_Password" "EMP_IDNum" "TitleID" "GenderID" **Employee Entity**	Selected by the employee
2.2.10	Display success message of updated employee	To notify the employee of the success	When an edit is required to an existing employee	"EMPID" "EMP_Name" "EMP_Surname" **Employee Entity**	Selected by the employee
2.2.2 [Alt]	Notification message of no authority	To notify the employee that they don't have the authority	When an edit is required to an existing employee		Selected by the employee
2.2.8 [Alt]	Notification of failure to verify message	To notify the employee that the details entered aren't correct	When an edit is required to an existing employee		Selected by the employee
2.3.3	Request the employee search details	To Know which details are to be searched for	When certain details are needed about an employee	"EMP_Name" "EMP_Surname" "EMP_TypeID" **Employee entity**	Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
2.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an employee	"EMP_Name" "EMP_Surname" "EMP_TypeID" **Employee entity**	Selected by the employee
2.3.2 [Alt]	Notification message of no authority	To notify the employee that they don't have the authority	When an edit is required to an existing employee		Selected by the employee
2.3.5 [Alt[	Notification of failure to verify message	To notify the employee that the details entered aren't correct	When an edit is required to an existing employee		Selected by the employee
2.3.7 [Alt]	Notification of failure to find a matching employee	To notify the employee that the details of the search weren't found	When an edit is required to an existing employee		Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
3.1.3	Request employee type details	To keep details about the new employee type	Adding a new employee type to the system	"EMP_TypeID" "EMP_TypeName" "EMP_TypeDes" **Employee type Entity**	Selected by the employee
3.1.8	Display success message	To let the employee that the employee type has been added	Adding a new employee type to the system	"EMP_TypeID" "EMP_TypeName" "EMP_TypeDes" **Employee type Entity**	Selected by the employee
3.1.2 [Alt]	Notification message of no authority	To notify the employee that they don't have the authority	Adding a new employee type to the system		Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
3.2.5	Request new details to employee type	To update the employee type with the new details entered	When an edit is required to an existing employee type	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.2.10	Display success message	To let the employee that the employee type has been updated	When an edit is required to an existing employee type	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.2.2	Notification message of no authority	To notify the employee that they don’t have the authority	When certain details are needed about an employee type		Selected by the employee
3.3.3	Request employee type search details	To Know which details are to be searched for	When certain details are needed about an employee type	“EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.3.7	Display search results	So that the employee can see the details they were searching for	When certain details are needed about an employee type	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.3.2 [Alt]	Notification message of no authority	To notify the employee that they don’t have the authority	When certain details are needed about an employee type		Selected by the employee
3.4.4	Request the employee type details	To know which employee type is to be removed	So that the employee type can no longer be used anymore	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee

UseCase	Description	Purpose	When Used	Entities and Attributes	Logical Layout
3.4.6	Request confirmation of removal	To ensure that the correct employee type is disabled	So that the employee type can no longer be used anymore		Selected by the employee
3.4.9	Display success notification message	So that the employee can see the details they are removing	So that the employee type can no longer be used anymore	“EMP_TypeID” “EMP_TypeName” “EMP_TypeDes” **Employee type Entity**	Selected by the employee
3.4.2 [Alt]	Notification message of no authority	To notify the employee that they don’t have the authority	So that the employee type can no longer be used anymore		Selected by the employee
3.4.8 [Alt]	Notification message of failure removal	To notify the employee that removal has failed	So that the employee can know the results		Selected by the employee

Use Case Number	Flow Line	Description	Purpose	When it will be used	Entities and attributes	Logical Layout
4. Driver						
4.1 Search driver	4.1.2[ALT]	Display error message	This is to trigger the process that displays an error message to the booking consultant	When the character information entered is not valid	-	Information being displayed to booking consultant
	4.1.6[ALT]	Alert message	This triggers the process that displays an alert message to the booking consultant	When the system cannot find any characters that match the input	-	An alert is being sent to the booking consultant
	4.1	Display driver details	This is the output which displays the details of a driver	When the search result returns a match	DRIVER - Driver_ID - Driver_Name, - Driver_Surname, - Driver_LicenceNumber, - Driver_LinceenceType, - EMPID	The information of a driver is displayed to the booking consultant
4.2 Check driver availability	4.2	List of available drivers	This is the available drivers output that is displayed	When checking driver availability	**SLOT** - Slot_ID - Slot_Date - Slot_Time	The list of drivers that is displayed to the booking consultant
	4.2.4[ALT]	Display message	This is the error message to error displayed to the booking consultant	When the driver list cannot be retrieved	-	An error message is displayed to the booking consultant
4.3 Assign driver to trip	4.3	Assignment confirmation	This is to display a message of successful assignment	When the system successfully assign a driver to a trip	-	Confirmation of assignment displayed to the booking consultant

	4.3.1 [ALT]	Driver not available message	This displays a message to the booking consultant that a driver is not available	When the system does not return a list of the available drivers	-	A message is displayed to the booking consultant
4.4 Outsource driver	4.4	Outsource driver assignment message	This is to display a message of a successful outsource driver assignment	When the system successfully assigns an outsourced driver to a trip	-	Confirmation of assignment displayed to the booking consultant
5. Vehicle						
5.1 Add vehicle	5.1	Added vehicle confirmation	This is the process that displays an a confirmation of a vehicle addition	When the system has added a new vehicle to the system	-	Alerts booking consultant of successful addition of vehicle
	5.1.3 [ALT]	Display error message	This is to inform the booking consultant of an error in the input	When information provided fails the system verification	-	A message is displayed to the booking consultant
5.2 Search vehicle	5.2	Vehicle details	This is the output resulting from a vehicle being searched	When the system retrieves the information of a vehicle	**VEHICLE** - Vehicle_ID - Vehicle_Model - VehicleMake_ID - VehicleMaintenance_ID - VehicleColour_ID - VehicleLicencePlates	Vehicle information displayed to the booking consultant
	5.2.3 [ALT]	Error in characters message	To inform the booking consultant of an error in the input	When the characters entered do not fit the system	-	An error message is displayed to the booking consultant

				requirements		
	5.2.5 [ALT]	Match not found message	Display a message to the booking consultant	When the system cannot find a match for entered characters	-	An error message is displayed to the booking consultant
5.3 Confirm trip	5.3a	Confirmation message	This is the process that displays a confirmation message to the booking consultant	When the system has successfully sent trip confirmation details	-	Confirmation message displayed to booking consultant
	5.3b	Passenger trip details	This is the output that is sent to the passenger	When the system retrieves trip information	**SLOT** - BookingReference_ID - DeparturePoint - DestinationPoint - Slot_Time **PASSENGER** - Passenger_ID - Passenger_Name - Passenger_Contact	Trip details sent to the passenger
	5.3c	Driver Trip Details	This is the output that is sent to the driver	When the system retrieves trip information and accumulates the driver trips of the day	**SLOT** - BookingReference_ID - DeparturePoint - DestinationPoint - Slot_Time **DRIVER** - Driver_ID - Driver_Name - Driver_Contact	Trip details sent to the driver
5.4 Check vehicle availability	5.4	List of available vehicles information	This is the results from the process that searches for available drivers	When the system retrieves a list of the vehicle that are	**BOOKING** - BookingReference_ID - Passenger_ID - Driver_ID - DeparturePoint	List of available vehicles is displayed to the booking

				available	- DestinationPoint - Time **VEHICLE GROUP** - Number_of_Passengers **SLOT** - Slot_Date - Slot_Time	consultant
	5.4.3 [ALT]	Display no vehicles available message	Inform the booking consultant that there are no cars available	When the system cannot find available vehicles on the system	**-**	
5.5 Assign vehicle to trip	5.5	Vehicle assignment confirmation	This process displays the information of a vehicle being assigned	When the system has assigned a driver to the trip	**SLOT** - Slot_ID - Slot_Date - Slot_Time	Cofirmation of assignment message displayed to the booking consultant
5.6 Outsource vehicle	5.6	Outsource vehicle assignment confirmation	This is the output of a process that assigns an outsourced vehicle to a trip	When the system successfully add the outsource vehicle information	**SLOT** - Slot_ID - Slot_Date - Slot_Time **OUTSOURCE_ VEHICLE** - OutsourceVehicle_ID - OutSource_ Vehicle_Model - OutSource_ Make	A message of confirmation is displayed to the booking consultant
6. Vehicle group						
6.1 Create vehicle group	6.1	New vehicle confirmation message	This is the result of a vehicle group being added on the system	When they system adds a new vehicle group	**VEHICLE GROUP** - Vehicle group ID, - VehicleGroup_Name - VehicleGroup_Description	A confirmation message is displayed to the booking consultant

	6.1.3 [ALT]	Display error message	To inform the booking consultant of error in input	When there is an error in input	-	A message is displayed to the booking consultant
6.2 Search vehicle group	6.2	Vehicle group information	This is the process that displays the vehicle group	When a match as been found for the search criteria by the system	**<u>VEHICLE GROUP</u>** - Vehicle group ID, - VehicleGroup_Name - VehicleGroup_Description	Vehicle group information is displayed to the booking consultant
	6.2.2[ALT]	Error in character input information	This displays a message to the booking consultant that there is an error in the input	When the input does not meet system criteria	**<u>-</u>**	A message is displayed to the booking consultant
	6.2.5[ALT]	Match not found information	This displays a message to the booking consultant that a match has not been found	When the system cannot find a match to the characters entered	**<u>-</u>**	A match not found message is displayed to the booking consultant
6.3 Update vehicle group	6.3	Updated details message	This is the process that displays a message confirming update	When the system has updated the vehicle group information	**<u>VEHICLE GROUP</u>** - VehicleGroup_Name - VehicleGroup_Description	Confirmation message is sent to the booking consultant
	6.3.5[ALT]	Input error message	This displays a message to the booking consultant about an error in the input	When the input does not meet the system requirements	-	A message is displayed to the booking consultant
Use case	Description	Purpose	When will it be used	Entities and Attributes		Output

7.1 Add Client	[ALT] Check if the client details are in the correct format	Inform the Booking consultant that data is incorrect format	When the system validates if the input is in the correct format		Message to notify that incorrect input was received in the input fields
	Confirmation of Successfully adding a new client	Inform the Booking consultant that Client is successfully added	Adding a new client to the Client table	Client table Client_Name Client_Email Client_Tel Client_Reference Client_Address Client_Type	Notification on the system
7.2 Search Client	list of all available clients in the Client table	Booking consultant could select the specific client	When searching for a client	Entity : Client Attributes: Client_Name Client_Reference Client_PhoneNumber Client_Email	
	selected Client	To view the details of the client that is being searched for	When the specific client is found	Entity : Client Attributes: Client_Name Client_Reference Client_PhoneNumber Client_Email Client_Address Client_type	

	[ALT] Invalid format details	Booking consultant being informed of incorrect input format	When client’s reference is in an incorrect format		Message to notify that incorrect input was received in the input fields
	[ALT] Client not found	Booking consultant being notified that the Client could not be retrieved in the Client table	When the system finds no match for the searched client		Message that the client could not be retrieved
7.3 Update Client	[ALT]Client to update not found	When the client the Booking consultant wants to update is not found	When the Booking consultant searches for the client to update		Message that the client could not be found
	[ALT] Incorrect input format	When the Booking consultant is informed of the incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
	Updated Client Details	When the Booking consultant is notified of the updating of Client details	Client details has been updated		Notification on the system

8.1 Add Client Type	Confirmation of successfully adding a new Client Type	When the booking consultant is notified that the Client type is added	Client type is added to the system	Entity :Client_Type Attributes ClientType_Name ClientType_Description	Notification on the system
	Duplicate Client types	When the booking consultant is notified that the client type exists	Adding a similar client type	Entity:Client_Type Attributes ClientType_Name ClientType_Description	Message to notify that the clientType already exists
	Invalid input format	When the booking consultant is notified of incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
8.2 Search Client Type	Client type retrieved from client table	When the Booking consultant is notified with the client type details	When the client type details are retrieved from the Client table	Entity : Client_Type Attributes ClientType_Name ClientType_Description	Displayed on the screen
	Client type not retrieved	When the Booking consultant is notified that the Client type not retrieved	The client type name does not match any Client type		Message to notify that the client type could not be retrieved

	Invalid Input format	When the booking consultant is notified of incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
8.3 Update Client Type	[ALT]ClientType to update not found	When the clientType the Booking consultant wants to update is not found	When the Booking consultant searches for the clientType to update		Message to notify that the client type to update could not be found
	[ALT] Incorrect input format	When the Booking consultant is informed of the incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
	Updated ClientType Details	When the Booking consultant is notified of the updating of ClientType details	ClientType details has been updated	Entity : Client_Type Attributes ClientType_Name ClientType_Description	Notificaiton on the system

Use case	Description	Purpose	When will it be used	Entities and Attributes	Output
7.1 Add Client	[ALT] Check if the client details are in the correct format	Inform the Booking consultant that data is incorrect format	When the system validates if the input is in the correct format		Message to notify that incorrect input was received in the input fields

Use case	Description	Purpose	When will it be used	Entities and Attributes	Output
	Confirmation of Successfully adding a new client	Inform the Booking consultant that Client is successfully added	Adding a new client to the Client table	Client table Client_Name Client_Email Client_Tel Client_Reference Client_Address Client_Type	Notification on the system
7.2 Search Client	list of all available clients in the Client table	Booking consultant could select the specific client	When searching for a client	Entity : Client Attributes: Client_Name Client_Reference Client_PhoneNumber Client_Email	
	selected Client	To view the details of the client that is being searched for	When the specific client is found	Entity : Client Attributes: Client_Name Client_Reference Client_PhoneNumber Client_Email Client_Address Client_type	
	[ALT] Invalid format details	Booking consultant being informed of incorrect input format	When client's reference is in an incorrect format		Message to notify that incorrect input was received in the input fields
	[ALT] Client not found	Booking consultant being notified that the Client could not be retrieved in the Client table	When the system finds no match for the searched client		Message that the client could not be retrieved
7.3 Update Client	[ALT]Client to update not found	When the client the Booking consultant wants to update is not found	When the Booking consultant searches for the client to update		Message that the client could not be found
	[ALT] Incorrect input format	When the Booking consultant is informed	When the input is not in a		Message to notify that there is

Use case	Description	Purpose	When will it be used	Entities and Attributes	Output
		of the incorrect input	correct format		incorrect input
	Updated Client Details	When the Booking consultant is notified of the updating of Client details	Client details has been updated		Notification on the system
8.1 Add Client Type	Confirmation of successfully adding a new Client Type	When the booking consultant is notified that the Client type is added	Client type is added to the system	Entity :Client_Type Attributes ClientType_Name ClientType_Description	Notification on the system
	Duplicate Client types	When the booking consultant is notified that the client type exists	Adding a similar client type	Entity:Client_Type Attributes ClientType_Name ClientType_Description	Message to notify that the clientType already exists
	Invalid input format	When the booking consultant is notified of incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
8.2 Search ClientType	Client type retrieved from client table	When the Booking consultant is notified with the client type details	When the client type details are retrieved from the Client table	Entity : Client_Type Attributes ClientType_Name ClientType_Description	Displayed on the screen
	Client type not retrieved	When the Booking consultant is notified that the Client type not retrieved	The client type name does not match any Client type		Message to notify that the client type could not be retrieved
	Invalid Input format	When the booking consultant is notified	When the input is not in a correct format		Message to notify that there is incorrect input

Use case	Description	Purpose	When will it be used	Entities and Attributes	Output
		of incorrect input			
8.3 Update ClientType	[ALT]ClientType to update not found	When the clientType the Booking consultant wants to update is not found	When the Booking consultant searches for the clientType to update		Message to notify that the client type to update could not be found
	[ALT] Incorrect input format	When the Booking consultant is informed of the incorrect input	When the input is not in a correct format		Message to notify that there is incorrect input
	Updated ClientType Details	When the Booking consultant is notified of the updating of ClientType details	ClientType details has been updated	Entity : Client_Type Attributes ClientType_Name ClientType_Description	Notificaiton on the system

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
9.1 Create Booking	Generated Booking Reference	Notify the Booking consultant of the generated Booking reference	When a booking is made successfully	Entity: Booking_Trip Attributes Booking_Reference	Notification on the system
	[ALT] Incorrect input format	When the Booking consultant is informed of the incorrect input	When the input is not in a correct format		Message to notify the Booking consultant that incorrect input
	[ALT] Duplicate Booking	When the Booking consultant is notified that the booking already exists	Adding a Booking with the same details		Message to notify the Booking consultant that the Booking already exists

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
	Generated Trip Quote	When the Booking consultant wants to retrieve the estimated cost of the Trip	When retrieving the total Cost of the trip		Message to Notify of the Cost of the Trip
9.3 Update Booking	[ALT]Booking to update not found	When the Booking to be updated is not found	When the Booking consultant searches for the Booking to update	Booking_Trip	Message to notify that the booking doesn’t exists
	[ALT] Incorrect input format	When the Booking consultant is informed of the incorrect input	When the input is not in a correct format		Message to notify the Booking consultant that incorrect input
	Updated Booking Details	When the Booking consultant is notified of the updating of Booking details	Booking details has been updated		Message to notify that the booking is successful
9.2 Search Booking	list of all available Bookings in the Booking_Trip table	Booking consultant could select the specific Booking	When searching for a Booking	Booking_Trip	System displays a list of the Available Bookings
	selected Booking	To view the details of the Booking that is being searched for	When the specific Booking is found	Booking_Trip Passenger Invoice	Displayed on the screen
	[ALT] Invalid format details	Booking consultant being informed of incorrect input format	When client’s reference is in an incorrect format		Message to notify the Booking consultant that there is incorrect input
	Booking Not Found	When the booking	Booking reference		Message to notify that

Use case	Description	Purpose	When will it be used	Entities and Attributes	Logical Layout
		searched for is not found	does not match any Booking		the booking is not found
9.4 Cancel Booking	[ALT] Booking not found	When the booking searched for is not retrieved	The Booking reference does not match any Booking in the Booking_Trip	Entity: Booking_Trip Attributes Booking_Reference	Message to notify that the booking is not found
	Charge on the Payment status	When the trip is already due for dispatch	The Booking consultant canceling trip		Message that 50% will be charged due to trip already due for dispatch
	Successful Cancellation of Booking	When the Booking cancellation succeeded	When the Booking is removed from the Booking table		Notification on the system
9.5 Confirm Booking	[ALT] The booking could not be retrieved	When searching for the booking to confirm	The booking reference does not match any booking in the Booking_Trip		Notification on the system
	Available Vehicles	Inform the Booking consultant of the available vehicles	When checking which vehicle is available for a trip		Notification on the system
	Available Drivers	Inform the Booking consultant of the available Drivers	When checking which Driver is available for a trip		Notification on the system
	Booking Successfully Confirmed	To inform the booking consultant that the Booking is successfully confirmed	When the Booking consultant finalizes a trip		Notification on the system

Use Case Number	Flow Line	Description	Purpose	When it will be Produced	Entities and Attributes	Logical Layout
13.1 Add Location	**13.1 [ALT]**	Failed Validation of Location Details.	To inform the Booking Consultant that details for Location has been entered incorrectly.	When the Booking Consultant entered details for Location in the wrong format.	(Location) - Location name **[Location_name]** - Province name **[Province_name]** - City **[City_Name]** - Suburb **[Suburb_name]** - Street **[Street_name]** - LocationType **[Locationtype]**	Error message displayed to Booking Consultant .
	13.1 [ALT]	Duplicate Location Detected Details.	To inform the Booking Consultant that duplicate Location has been detected on the system.	When the Location being added has the details as existing Location on the system.	(Location) - Location name **[Location_name]** - Province name **[Province_name]** - City **[City_Name]** - Suburb **[Suburb_name]** - Street **[Street_name]** -	Notification displayed to Booking Consultant .
	13.1.5	Notification of Successfully added Location information.	To inform the Booking Consultant of successful addition of Location to system.	When the Booking Consultant has successfully added Location details to the system.	(Location) - Location name **[Location_name]** - Province name **[Province_name]** - City **[City_Name]** - Suburb **[Suburb_name]**	Notification displayed to the Booking Consultant .

Use Case Number	Flow Line	Description	Purpose	When it will be Produced	Entities and Attributes	Logical Layout
					- Street [**Street_name**]	
13.2 Search Location	**13.2.3**	Selected Location with Relevant Location Land and Location Land Field information.	To display all the relevant information about a Location with its lands and fields.	When the User has requested to view Location s on the system.	(Location) - Location name [**Location_name**] - Province name [**Province_name**] - City [**City_Name**] - Suburb [**Suburb_name**] - Street [**Street_name**]	Information displayed on the screen.
13.3 Update Location	**13.3 [ALT]**	Failed Validation of Location Details.	To inform the Booking Consultant that details have been entered incorrectly .	When the Booking Consultant has entered details in the wrong format.	-	Error message displayed to Booking Consultant .
	13.3.6	Notification of Successful Update information.	To inform the successful update of Location details on the system.	When the Booking Consultant has successfully updated details of Location .	(Location) - Location name [Location_name] - Province name [Province_name] - City [City_Name] - Suburb [Suburb_name] - Street [Street_name]	Notification message displayed to Booking Consultant .

Use Case Number	Flow Line	Description	Purpose	When it will be Produced	Entities and Attributes	Logical Layout
14.1 Add Zone	**14.1 [ALT]**	Failed Validation of Zone Details.	To inform the Booking Consultant that details for Zone has been entered incorrectly.	When the Booking Consultant entered details for Zone in the wrong format.	• Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Error message displayed to Booking Consultant .
	14.1.5	Notification of Successfully added Zone information.	To inform the Booking Consultant of successful addition of Zone to system.	When the Booking Consultant has successfully added Zone details to the system.	• Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Notification displayed to the Booking Consultant .
14.2 Search Zone	**14.2.3**	Selected Zone with Relevant Zone Range and Zone information.	To display all the relevant information about a Zone with Range	When the User has requested to view Zone s on the system.	• Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]** [**Suburb_name**]	Information displayed on the screen.
14.3 Update Zone	14.3 [ALT]	Failed Validation of Zone Details.	To inform the Booking Consultant that details have been entered incorrectly.	When the Booking Consultant has entered details in the wrong format.	-	Error message displayed to Booking Consultant .

Use Case	Flow Line	Description	Purpose	When it will be Produced	Entities and Attributes	Logical Layout
14.4 Remove Zone	**14.4 [ALT]**	There is a Trip linked to Zone .	To inform Booking Consultant that the selected Zone is linked to a Trip	When the system detects Zone is linked to a Trip and cannot proceed to remove.	• Zone Id **[Zone_ID]** • Zone name **[Zone_name]** • Zone range **[zone_range]**	Notification message displayed to User.
	14.4.4	Notification of Successfully Removed Zone information.	To inform the Booking Consultant that Zone has successfully been removed.	When the Booking Consultant has selected to remove a specific Zone .	• Zone Id [Zone_ID] • Zone name [Zone_name] • Zone range [zone_range]	Notification message displayed to Booking Consultant .

Use Case Number	**Flow Line**	**Description**	**Purpose**	**When it will be Produced**	**Entities and Attributes**	**Logical Layout**
15.5Generate Invoice .	**15.5[ALT]**	Booking Has Not Been Selected.	To inform the Booking Consultant that Booking details have not been specified.	When the system tries to generate Invoice but detects that Booking details have not been chosen.	-	Error message displayed to Booking Consultant .
	15.5[ALT]	Unable to Access Database to Retrieve Data	To inform the Booking Consultant that Invoice cannot be generated because data required cannot be retrieved.	When the system tries to retrieve data required for report, it encounters problems.	-	Error Notification displayed to Booking Consultant .

Use Case Number	Flow Line	Description	Purpose	When it will be Produced	Entities and Attributes	Logical Layout
	15.6	Generated and End Dates Failed Validation Checks	To generate an Invoice	When the Consultant requests to generate a Invoice	Invoice • Invoice _ID • Booking_Refference • Invoice _PaymentStatus • Invoice_date • Invoice_Quote	Invoice (List): List of all completed trips on An invoice with all its associated details.

CONCLUSION

This section assisted us in grasping all the system outputs in more detail. It helped us to understand the outputs purpose, when it gets produced, entities and attributes involved as well as the logical layout all contained in a tabular document.

8. VALIDATION

INTRODUCTION

In this section, the validation of each functional specification, process and entities is provided to check if they correspond to the business requirements.

VALIDATION

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
4. Driver	4.1 Search driver	4.1 Search driver	4.1.1 4.1.2 4.1.3 4.1.4 4.1.5 4.1.6 4.1.7	- - - Driver - Driver -
	4.2 Check driver availability	4.2 Check driver availability	4.2.1 4.2.2 4.2.3 4.2.4	- Slot - -
	4.3 Assign driver to trip	4.3 Assign driver to trip	4.3.1 4.3.2 4.3.3 4.3.4 4.3.5	- - - Slot -
	4.4 Outsource driver	4.4 Outsource driver	4.5.1 4.5.2 4.5.3 4.5.4	Outsourced_Driver - Slot -
5. Vehicle	5.1 Add vehicle	5.1 Add vehicle	5.1.1 5.1.2 5.1.3 5.1.4 5.1.5 5.1.6	- - - Vehicle, VehicleMake, VehicleMaintenance Vehicle, VehicleMake, VehicleMaintenance -
	5.2 Search vehicle	5.2 Search vehicle	5.2.1 5.2.2 5.2.3 5.2.4 5.2.5 5.2.6 5.2.7	- - - Vehicle - Vehicle -
	5.3 Confirm trip	5.3 Confirm trip	5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7	- Slot Passenger, Driver - - - -

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
	5.4 Check vehicle availability	5.4 Check vehicle availability	5.4.1 5.4.2 5.4.3 5.4.4	Booking Vehicle_Group, Slot - -
	5.5 Assign vehicle to trip	5.5 Assign vehicle to trip	5.5.1 5.5.2 5.5.3 5.5.4	- - Slot -
	5.6 Outsource vehicle	5.6 Outsource vehicle	5.6.1 5.6.2 5.6.3 5.6.4	Outsource_Vehicle - Slot -
6. Vehicle Group	6.1 Create vehicle group	6.1 Create vehicle group	6.1.1 6.1.2 6.1.3 6.1.4 6.1.5 6.1.6	- - - Vehicle_Group Vehicle_Group -
	6.2 Search vehicle group	6.2 Search vehicle group	6.2.1 6.2.2 6.2.3 6.2.4 6.2.5 6.2.6 6.2.7	- - - Vehicle_Group - - Vehicle_Group
	6.3 Update vehicle group	6.3 Update vehicle group	6.3.1 6.3.2 6.3.3 6.3.4 6.3.5 6.3.6 6.3.7	- - - - - Vehicle_Group -

SubSystem	Requirement	User Case	Process(DFD)	Entities(ERD)
CLIENT	7.1 Add Client	7.1 Add Client	7.1.1 Capture new Client Details	
			7.1.2 Validate the Client Details	
			7.1.3 Validate for duplicates	Client
			7.1.4 Save the new Client	Client
			7.1.5 Generate Confirmation Message	
	7.2 Search Client		7.2.1 Display All Clients	Client
			7.2.2 Capture and validate Client Reference	

			7.2.3 Narrow the Client details	
			7.2.4 Display Selected Client Details	Client
	7.3 Update Client		7.3.1 Retrieve Client to Update	Client
			7.3.2 Capture Client Details to Update	
			7.3.3 Validate Updated Client details	
			7.3.4 Save the update client details	Client
			7.3.5 Generate Confirmation of Updating Client	
CLIENT_TYPE	8.1 Add Client_Type		8.1.1 Capture Client Type Details	
			8.1.2 Validates the Client Type Details	
			8.1.3 Validates Existing Client types	Client_Type
			8.1.4 Save the Client Type	Client_Type
			8.1.5 Generate Confirmation Message	
	8.2 Search Client_Type		8.2.1 Display All Clients_Type	Client_Type
			8.2.2 Capture and validate Client Type name	
			8.2.3 Narrow the Client Type details	
			8.2.4 Display Selected Client Type Details	Client_Type
	8.3 Update Client Type		8.3.1 Retrieve Client Type to Update	Client_Type

			8.3.2 Capture Client type Details to Update	
			8.3.3 Validate New Client type details	
			8.3.4 Save The update Client Type details	Client_Type
			8.3.5 Generate Confirmation of Updating Client Type Details	
9 BOOKING	9.1 Create Booking		9.1.1 Request Client Name and validate the format	
			9.1.2 Validate Client Existence	Client
			9.1.3 Confirm Client Details and save	
			9.1.4 Capture Pickup and DropOff Details	
			9.1.5 Validates Input Format	
			9.1.6 Check for Booking Duplicates	Booking_Trip
			9.1.7 Generate Quote and Booking Reference	
			9.1.8 Capture Passenger Details	
			9.1.9 Validates input Format	
			9.1.10 Checks Duplicate Passenger	Passenger
			9.1.11 Saves Passenger	Passenger
			9.1.12 Capture Pickup Instructions	
			9.1.13 Communicates Booking Details	

			9.1.14 Generate Invoice Number	Invoice
BOOKING			9.1.15 Save Booking Details and save Invoice	Booking_Trip Invoice
			9.1.16 Confirm Booking and Notify about Booking Reference	
	9.2 Search Booking		9.2.1 Display All Bookings	Booking_Trip
			9.2.2 Capture and Validated Booking Reference	
			9.2.3 Narrow the bookings displayed	
			9.2.4 Display Selected Booking Details	Booking_Trip Passenger Invoice
	9.3 Update Booking		9.3.1 Retrieve a Booking to be updated	Booking_Trip Passenger Invoice
			9.3.2 Capture updated Booking details	
			9.3.3 Validate Updated Booking Details	
			9.3.4 Save the Updated Booking Details	Booking_Trip Passenger Invoice
			9.3.5 Generate Confirmation of Updating	
BOOKING	Cancel Booking	9.4 Cancel Booking	9.4.1 Retrieve Booking to cancel	Booking_Reference
			9.4.2 Check if Booking is due for Dispatch	

			9.4.3 Capture Reasons for Cancellation	
			9.4.4 Update Vehicle and Driver Availability	Slot
			9.4.5 Update Schedule	
			9.4.6 Update Payment_Status	Invoice
			9.4.7 Update Booking Status	Booking_Trip
			9.4.8 Generate Confirmation of Booking Cancellation	
Booking	Confirm Booking	9.5 Confirm Booking	9.5.1 Retrieve Booking to be Confirmed	Booking_Trip
			9.5.2 Capture Assigned Vehicle Details	Slot Vehicle_Group
			9.5.3 Capture Assigned Driver Details	Slot Driver
			9.5.4 Assign Driver and Vehicle	Slot
			9.5.5 Create Schedule and Update Booking Status	Booking_Trip

Subsystem	**Requirement**	**Use Case**	**Process(DFD)**	**Entities(ERD)**
10 Schedule	10.1 Create Schedule	Create Schedule	10.1.1 Read Booking details	N/A
			10.1.2 Validate Format of the Booking Details	**Schedule Table** • **BookingReference** • **Client_ID** • **Driver_ID** • **Date** • **Time** • **Destination Time**

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
				• DestinationLocation • TripDuration (Which is calculated using Time and Destination Time)
			10.1.3 Retrieve the last Booking Unique Number	N/A
			10.1.4 Store Booking Details	**Schedule Table** • BookingReference • Client_ID • Driver_ID • Date • Time • Destination Time • DestinationLocation • TripDuration (Which is calculated using Time and Destination Time)
			10.1.5 Notify Booking Consultant with confirmation Message	N/A
			ALT 10.1.2 Display Invalid Error Message	N/A
	10.2 Update schedule	Update schedule	10.2.1Request to Update Scheduled booking	N/A
			10.2.2 Display Details of the booking to be updated	N/A
			10.2.3 Read Updated Booking Details	N/A
			10.2.4 Validate Format of the updated booking details	N/A
			10.2.5Stored updated Booking details	N/A

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
			10.2.6 Display updated booking details	N/A
			10.2.7 Notify of successful Update	N/A
			ALT 10.2.4 Display Error message for incorrect format	N/A
	10.3 View schedule	View schedule	10.3.1 Request to view schedule	N/A
			10.3.2 Read user entered Search criteria	**Schedule** table: • **BookingRefernce** • **Client_ID** • **Driver_ID** • **Date** • **Time** • **DestinationLocation** • **TripDuration**
			10.3.3 Perform schedule search	N/A
			10.3.4 Display list of all bookings from the chosen date	**Schedule** table: • **BookingRefernce** • **Client_ID** • **Driver_ID** • **Date** • **Time** • **DestinationLocation** • **TripDuration**
			10.3.5 Display chosen booking details	**Schedule** table: • **BookingRefernce** • **Client_ID** • **Driver_ID** • **Date** • **Time** • **DestinationLocation** • **TripDuration**
			10.3.6 Read User Selected Search Result	N/A
			10.3.7 Display User Selected Search Result Details	N/A

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
			ALT 10.3.3 Display Schedule Search Criteria Error Message	N/A
11 Reporting	11.1 Booking Report	Booking Report	11.1.1 Display schedule generation	N/A
			11.1.2 Requests booking search criteria	N/A
			11.1.3 Reads entered report criteria details	Schedule Table: • **Vehicle_ID** • **Client_ID** • **Driver_ID** • **Booking_Date** • **Pickup-Location** • **DropOff-location** • **Date** • **Time**
			11.1.4 Generate report of the entered schedule details	Schedule Table: • **Vehicle_ID** • **Client_ID** • **Driver_ID** • **Booking_Date** • **Pickup-Location** • **DropOff-location** • **Date** • **Time**
			11.1.5 Display complete report	Schedule Table: • **Vehicle_ID** • **Client_ID** • **Driver_ID** • **Booking_Date** • **Pickup-Location** • **DropOff-location** • **Date** • **Time**
			ALT 11.1.4 Displays invalid schedule not found error message	N/A
	11.2 Driver-Trip Report	Driver-Trip Report	11.2.1 Display schedule generation	N/A
			11.2.2 Requests	N/A

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
			booking search criteria	
			11.2.3 Reads entered report criteria details	Schedule Table: - **Vehicle_ID** - **Client_ID** - **Driver_ID** - **Booking_Date** - **Pickup-Location** - **DropOff-location** - **Date** - **Time**
			11.2.4 Generate report of the entered schedule details	Schedule Table: - **Vehicle_ID** - **Client_ID** - **Driver_ID** - **Booking_Date** - **Pickup-Location** - **DropOff-location** - **Date** - **Time**
			11.2.5 Display complete report	Schedule Table: - **Vehicle_ID** - **Client_ID** - **Driver_ID** - **Booking_Date** - **Pickup-Location** - **DropOff-location** - **Date** - **Time**
			ALT 11.2.4 Displays invalid schedule not found error message	N/A
12 Vehicle Maintenance	12.1 Schedule Vehicle Maintenance schedule	Schedule Vehicle Maintenance schedule	12.1.1 Request Maintenance Details	N/A
			12.1.2 Read Maintenance Details	**Vehicle Table** - Vehicle_ID - Vehicle_Description - Vehicle_Name - Vehicle_LicseneceNumber - Duration - DateTime
			12.1.3 Validate Format of	N/A

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
			Maintenance Details	
			12.1.4 Retrieve Last Maintenance unique number	**Vehicle maintenance table** • **Vehicle_MaintenanceID**
			12.1.5 Store Maintenance Details	**Vehicle maintenance table** • Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			12.1.6 Notify Operational Manger of scheduling	N/A
	12.2 Update Vehicle Maintenance schedule	Update Vehicle Maintenance schedule	12.2.1 Request to Update schedule	N/A
			12.2.2 Display Details of schedule list that can be Updated	**Vehicle maintenance table** • Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			12.2.3 Read updated schedule details	N/A
			12.2.4 Validate format of the updated schedule details	N/A
			12.2.5 Store updated schedule details	**Vehicle maintenance table** • Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			12.2.6 Display updated	**Vehicle maintenance table**

Subsystem	Requirement	Use Case	Process(DFD)	Entities(ERD)
			schedule details	• Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			12.2.7 Notify manager of successful update	N/A
	12.3View Vehicle Maintenance schedule	View Vehicle Maintenance schedule	12.3.1 Request search Criteria	N/A
			12.3.2 Read User Entered Search Criteria	N/A
			12.3.3 Perform Schedule Search	N/A
			13.3.4 Display List of all Maintenances scheduled that match search results	**Vehicle maintenance table** • Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			12.3.5 Read User selected search result	N/A
			12.3.6 Display User selected Search result details	**Vehicle maintenance table** • Vehicle_ID • Vehicle_Description • Vehicle_Name • Vehicle_LicseneceNumber • Duration • DateTime
			ALT 12.3.4 Display schedule search criteria error message	N/A

13. Location	13.1 Add Location	13.1 Add Location	13.1.1	• Location • City • Suburb • Location Type
			13.1.2	-
			13.1.3	-
			13.1.4	• Location • City • Suburb • Location Type
			13.1.5	• Location • City • Suburb • Location Type
			13.1.6	• Location • City • Suburb • Location Type
			13.1.7	• Audit Type • Audit
			13.1.8	-
	13.2 View Location	13.2 View Location	13.1.9	• Location • City • Suburb • Location Type
			13.2.1	-
			13.2.2	• Location • City • Suburb • Location Type
			13.2.3	• Location • City • Suburb • Location Type
			13.2.4	• Audit Type • Audit
	13.3 Update Location	13.3 Update Location	13.2.5	-
			13.3.1	-
			13.3.2	-
			13.3.3	-
			13.3.4	• Location • City • Suburb

				• Location Type
			13.3.5	Audit Type, Audit
			13.3.6	-

14. Zone	14.1 Add Zone	14.1 Add Zone	14.1.1	• Zone
			14.1.2	-
			14.1.3	-
			14.1.4	• Zone
			14.1.5	• Zone
			14.1.6	• Zone
			14.1.7	• Audit Type • Audit
			14.1.8	-
	14.2 View Zone	14.2 View Zone	14.1.9	• Zone
			14.2.1	-
			14.2.2	• Zone
			14.2.3	• Zone
			14.2.4	• Audit Type • Audit
	14.3 Update Zone	14.3 Update Zone	14.2.5	-
			14.3.1	-
			14.3.2	-
			14.3.3	-
			14.3.4	• Zone
			14.3.5	Audit Type, Audit
			14.3.6	-

Subsystem	Requirement	Use Case	Processes (DFD)	Entities (ERD)
	15.1Generate Invoice	15.1Generate Invoice	15.1.1	Booking
			15.1.2	-
			15.1.3	-

Subsystem	Requirement	Use Case	Processes (DFD)	Entities (ERD)
			15.1.4	• Invoice _ID • Booking_Refference • Invoice_Payment Stats • Invoice_date • Invoice_Quote
			15.1.5	AuditType, Audit
			15.1.6	-

CONCLUSION

The section above contains the validation of each functional specifications and provided an outline, which showed the team which requirements have been dealt with and which one still needs to be addressed.

9. SIGN-OFF BY CLIENT

INTRODUCTION

The development team has presented the final functional specification document to the client for purposes of reviewing the contents therein. It is ensured that all specifications are understood and agreed to, and in this section the client provides the sign off to conclude the document.

CONCLUSION

By signing this document, the client declares that she has read and understood the contents of the document therein, and gives approval of the functional specification.

10. GENERAL

INTRODUCTION

In this section, we will analyses the complexity requirements and indicate the requirements we will meet and the total marks we will achieve.

CONCLUSION

In this section, we analyzed the complexity requirements and indicated the marks we are currently meeting based on content of the functional specification document and also indicated the total marks we will achieve.

11. SIGN OFF

INTRODUCTION

See

CONCLUSION

See