POLITECNICO
MILANO 1863

Integration Testing Document

Lo Bianco Riccardo - Manzoni Mirco - Mascellaro Giuseppe

December 29, 2016
v1.0

Contents

1 Introduction

1.1 Purpose

This document contains the complete description of the tests to be performed on the components of the PowerEnJoy system. Along the discussion, we refer to the components and the architecture described in the design document, so a deep knowledge of the previous document is required to fully understand the content of the current one.

1.2 Scope

We plan on having a software that fulfills both the functional and non-functional requirements defined in the RASD. Keeping this objective in mind, we will perform a complete testing of the designed components, starting from the testing of the single components and moving on with a exhaustive integration test.
In particular, we must consider the fact that many of the functionalities can be considered "correct" not only with respect to the code, but first of all with respect to the desired output, that is influenced by factors that are not strictly code-related (for example, the right allocation of weights through the zones). Because of this kind of issues, it is fundamental not only that this document is read carefully by the team members, but also that it is discussed with the customer.

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

- **Bottom-up approach:** approach widely used for integration of big informatic systems which is based on a first integration of the single components into subsystems to be integrated in a second time to obtain the final system.

1.3.2 Acronyms

- **API:** Application Programming Interface, a set of routine, protocols and tools used for building software and generally provided by external actors
- **DBMS:** DataBase Management System
- **JEE:** Java Enterprise Edition
- **N&M:** Navigation & Maps

1.4 Reference documents

1. **Analysis document:** the document which focuses on both functional and non-functional requirements for the PowerEnJoy system.

2. **Design document:** the document which focuses on the architecture of the PowerEnJoy system. The DD is to be intended as the starting point for the current document.

3. **IEEE Standard for Information Technology - Systems Design - Software Design Descriptions:** the standard for the development of informatic systems.

2 Integration strategy

2.1 Entry criteria

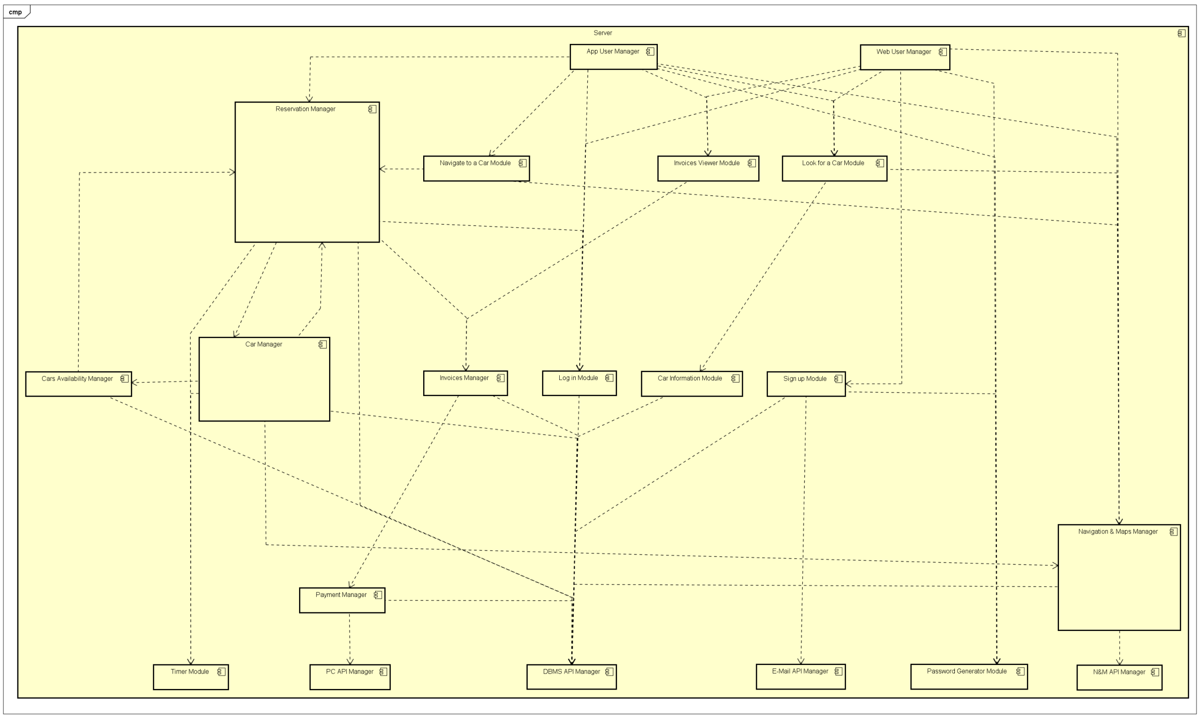
The integration test is intended to verify the completeness and realiability of the code produced for the PowerEnJoy system. The integration testing phase can start only if a number of conditions are obtained from previous project phases:

- Single units must have been previosly tested. In particular, every function must have been tested exhaustively.
- All the constants defined in the code must be inspected, if necessary with the aid of the field expert who defined them or of the customer himself.
- A consistent set of test data is available in order to test functions that require interaction with the database.
- Reservation manager, car manager, signup module, navigation maps module are represented as single module because they must be intended as already completely tested at the start of the integration process.

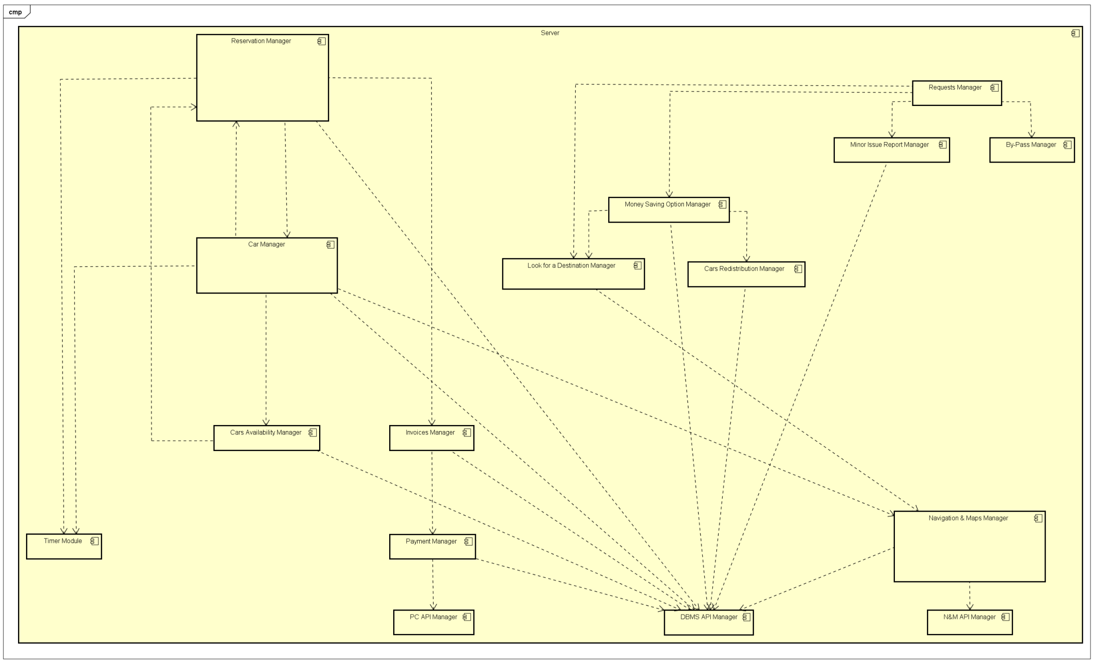
2.2 Elements to be integrated

The items to be tested are all the components previously designed for the PowerEnJoy system, except for those that are related to the GUI. For a complete description of these components we refer to the Design Document delivered in the complete documentation of the project.
In the diagrams below, all the software components are listed and they are linked by "depends on" relation, represented by the dashed arcs. In this way it's easy to reconize which components are the most important to tests because of their relation between lots of modules of our system. The "depends on" relation has been built starting from the Component Diagrams of Design Document, replacing all the interfaces between components with dashed arcs.

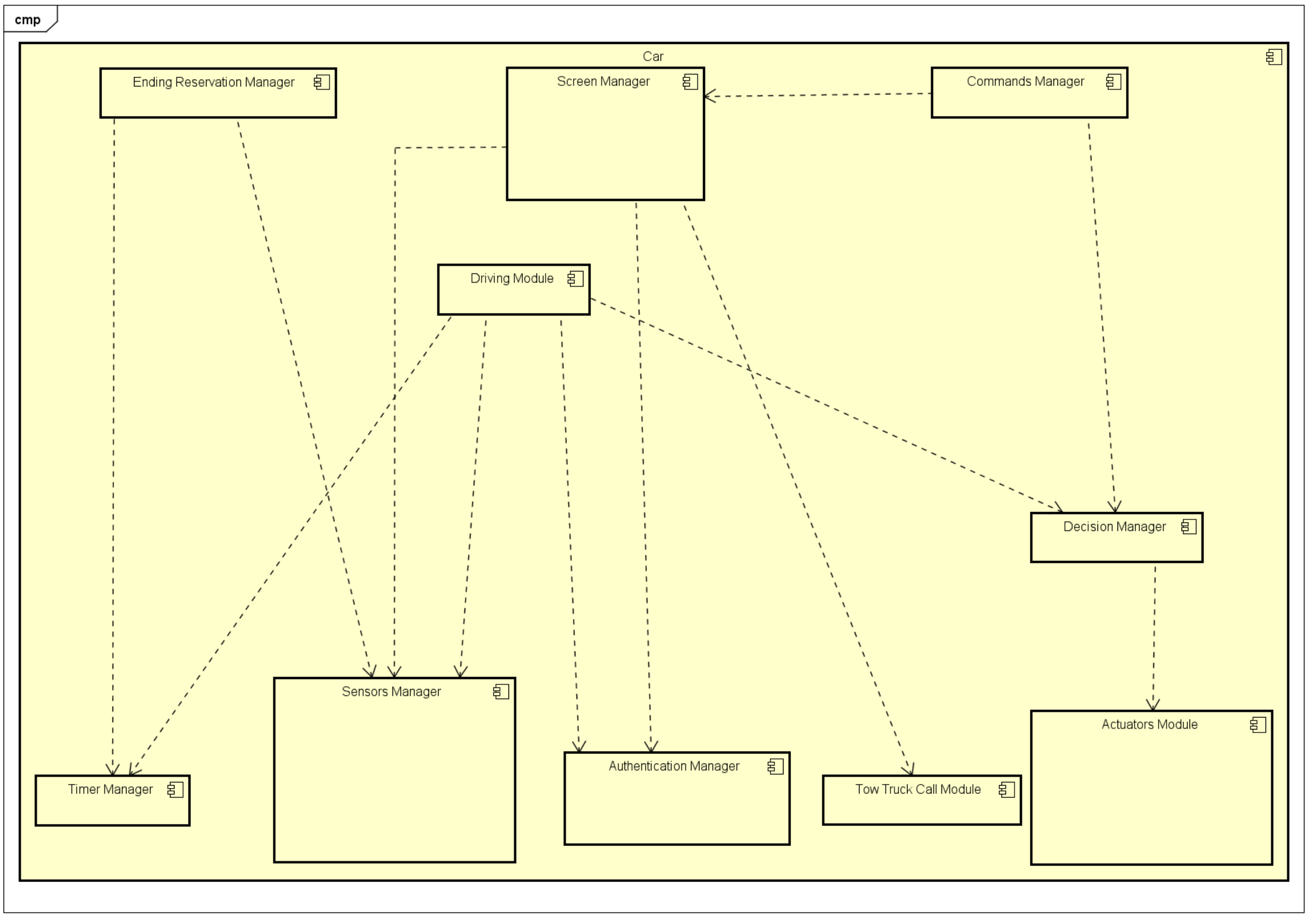
2.2.1 User App Browser



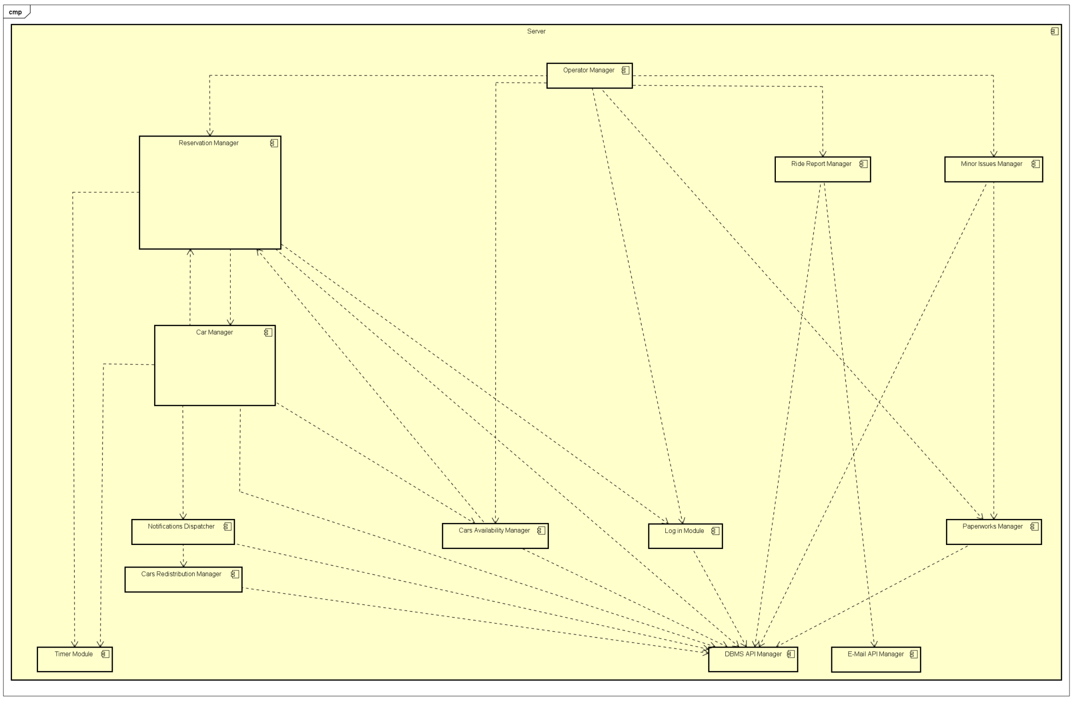
2.2.2 Car Server Side


cmp
Server
Reservation Manager
Requests Manager
Minor Issue Report Manager
By-Pass Manager
Money Saving Option Manager
Car Manager
Look for a Destination Manager
Cars Redistribution Manager
Cars Availability Manager
Invoices Manager
Navigation & Maps Manager
Timer Module
Payment Manager
PC API Manager
DBMS API Manager
N&M API Manager

2.2.3 Car Client Side



2.2.4 Operator



2.3 Integration testing strategy

The complete system was divided into four subsystems. The approach we chose for the testing phase inside the subsystems is of type bottom-up: we start from integrating pairs of singularly tested modules and we go up until we reach the integration of the entire subsystem. Once we finish testing every subsystem in its entirety, we proceed with the integration of all of them, threated as black boxes, to obtain the complete system. It has to be pointed out that this last phase is strictly related to the previous one, and its complexity should be very low assuming that the first integration phase was completely successful.
We chose not to use a big bang approach, where clusters of components are aggregated by logic field and then integrated together, because of the risk of increased complexity in the integration of the complete system: this kind of approach requires extreme attention in recording every step of the integration already performed and, considered the complexity of our system that reflects on the subsystems, we preferred to follow the classic and tidy bottom up approach.

2.4 Sequence of component integration

We have identified four different subsystems in the organization of our tests:

- **Car Server:** the most critical part for our system. It concerns all functionalities and configurations of the cars and their business logic. It takes care for creating communication channels between the system server side and the cars and vice versa. Reservations, cars' requests, decisions and timers are handled by this part of the system.

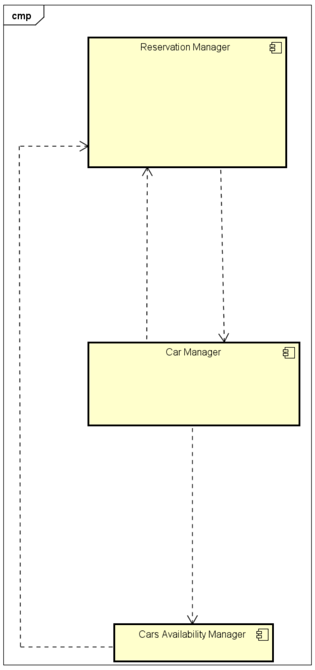
- **Car Client:** we have decided to test both server and client side in depth because the communication between them is a fundamental aspect for the operation mode of the entire system. Indeed, we want to be sure (as much as possible) that the communication works properly. This subsystem is responsible for driving decisions, taken following server instructions via actuators, and for handling sensors and user interfaces.

- **Operator Server:** the part of the system concerning business logic of operators on the server side. We do not spent time on the client side because it is basically a thin client and simple unit tests should be sufficient at this level also because we assume that the business logic is correct. Moreover, operator clients are inside the local private network of our system and we rely on simple local communication protocols. The operator server side developed in this document mainly concerns the notifications dispatcher, paperworks handling, ride report managing and minor issues report managing.

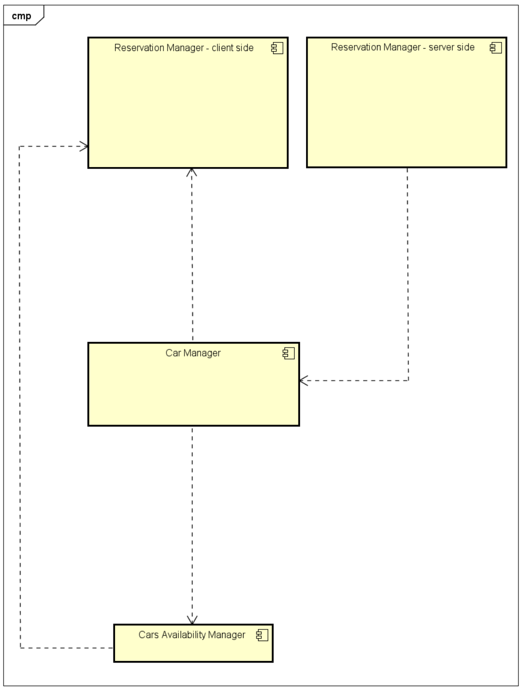
- **App User – Web User Server:** the subsystem concerning business logic parts present in the server that communicate with the final user. They can be accessed via the user mobile app or via web browsing pages. They allow to exploit functionalities of the system client side. We decided to test these parts in depth because some of clients' decisions may cost them extra expenses and this is why they are to be considered critical tasks. The client side of this software part is not covered by integration tests because they are basically thin clients and simple unit tests should be sufficient at this level also because we assume that the business logic is correct.

2.4.1 Loop dependencies management

The planning of the integration testing phase required us to manage the existence of loop dependencies in the design of the components. In particular, we decideed to distiguish between client and server sides when any loop dependency showed up.
So, for example, in the following image we can see how a loop dependency shows itself in section 2.2



and here we represented how we treated the very same dependency in section 3


cmp
Reservation Manager - client side
Reservation Manager - server side
Car Manager
Cars Availability Manager

This dependency exists because both components need to send asynchronous messages in order to request services to the other one. Indeed, the taxi driver client both asks for a service but it also provides services. At the testing level, we've solved the dependency problem splitting the component "Reservation Manager" into two different ones. Each sub-component exports and provides functionalities according to the side with which it interfaces. In that way the component is tested following a functionalities approach. In section 3, sometimes we used the name “Reservation Manager” to refer to both server and client sides.

2.4.2 Parallelization approach

As stated in section 2.3, we used a bottom-up approach for integration tests and, as it can be seen in dependences diagrams in section 2.2, we started testing starting from the bottom of diagrams and then we go above integrating focusing on dependences among components. In this way we can easily highlight how integration tests may be parallelized in order to make this task more efficient using different integration tester teams and speeding up the entire testing task.

3 Individual steps and test description

In the following paragraph we represent the flow of the integration between the various components of the PowerEnJoy system. Every subsection is meant to describe the integration process of a single element, brought on through one or multiple tests. The meaning of the fields in the tables below can be summarized as follows:

- **Test Case ID:** unique identifier for the analyzed integration test.
- **Test Item:** couple of components involved in the test. The right arrow represents the ”used by” relationship.
- **Input Specification:** high level description of the input data provided by the second element of the test items to the first one.
- **Output Specification:** high level description of the desired output.
- **Environmental needs:** conditions necessary for the execution of the test case.

3.1 User App and Browser Server Core Subsystem

3.1.1 Timer Module Integration Test

Test Case Identifier	I1T1
Test Item(s)	Timer Module → Reservation Manager
Input Specification	Simulate Timer Module component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

3.1.2 PC API Manager Integration Test

Test Case Identifier	I2T1
Test Item(s)	PC API Manager → Payment Manager
Input Specification	Simulate PC API Manager component typical input coming from External Payment Manager, paying attention to cover the exceptional and edge cases related to the PCInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Payment Manager driver must have been already implemented.

3.1.3 DBMS API Manager Integration Test

Test Case Identifier	I3T1
Test Item(s)	DBMS API Manager → Payment Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Payment Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Payment Manager driver must have been already implemented.

Test Case Identifier	I3T2
Test Item(s)	DBMS API Manager → Car Availability Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Car Availability Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Availability Manager driver must have been already implemented.

Test Case Identifier	I3T3
Test Item(s)	DBMS API Manager → Reservation Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

Test Case Identifier	I3T4
Test Item(s)	DBMS API Manager → Car Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Car Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented.

Test Case Identifier	I3T5
Test Item(s)	DBMS API Manager → Invoice Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Invoice Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Invoice Manager driver must have been already implemented.

Test Case Identifier	I3T6
Test Item(s)	DBMS API Manager → Log In Module
Input Specification	Simulate DBMS API Manager component typical input coming from External Log In Module, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Log In Module driver must have been already implemented.

Test Case Identifier	I3T7
Test Item(s)	DBMS API Manager → Car Information Module
Input Specification	Simulate DBMS API Manager component typical input coming from External Car Information Module, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Information Module driver must have been already implemented.

Test Case Identifier	I3T8
Test Item(s)	DBMS API Manager → Sign Up Module
Input Specification	Simulate DBMS API Manager component typical input coming from External Sign Up Module, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Sign Up Module driver must have been already implemented.

Test Case Identifier	I3T9
Test Item(s)	DBMS API Manager → Navigation & Maps Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Navigation Maps Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigation & Maps Manager driver must have been already implemented.

3.1.4 E-Mail API Manager Integration Test

Test Case Identifier	I4T1
Test Item(s)	E-Mail API Manager → Sign Up Module
Input Specification	Simulate E-Mail API Manager component typical input coming from External Sign Up Module, paying attention to cover the exceptional and edge cases related to the E-MailInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Sign Up Module driver must have been already implemented.

3.1.5 Password Generator Module Integration Test

Test Case Identifier	I5T1
Test Item(s)	Password Generator Module → Sign Up Module
Input Specification	Simulate Password Generator Module component typical input coming from External Sign Up Module, paying attention to cover the exceptional and edge cases related to the PswInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Sign Up Module driver must have been already implemented.

Test Case Identifier	I5T2
Test Item(s)	Password Generator Module → Web User Manager
Input Specification	Simulate Password Generator Module component typical input coming from External Web User Manager, paying attention to cover the exceptional and edge cases related to the PswInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Manager driver must have been already implemented.

Test Case Identifier	I5T3
Test Item(s)	Password Generator Module → Sign up Module
Input Specification	Simulate Password Generator Module component typical input coming from External Sign up Module, paying attention to cover the exceptional and edge cases related to the PswInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Sign up Module driver must have been already implemented.

3.1.6 NM API Manager Integration Test

Test Case Identifier	I6T1
Test Item(s)	N&M API Manager → Navigation & Maps Manager
Input Specification	Simulate N&M API Manager component typical input coming from External Navigation & Maps Manager, paying attention to cover the exceptional and edge cases related to the N&MInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigation & Maps Manager driver must have been already implemented.

3.1.7 Payment Manager Integration Test

Test Case Identifier	I7T1
Test Item(s)	Payment Manager → Invoices Manager
Input Specification	Simulate Payment Manager component typical input coming from External Invoices Manager, paying attention to cover the exceptional and edge cases related to the PaymentInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values
Environmental Needs	The Invoices Manager driver must have been already implemented and I2T1 must have already been performed.

3.1.8 Navigation & Maps Integration Test

Test Case Identifier	I8T1
Test Item(s)	Navigation & Maps Manager → Navigate to a Car Module
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External Navigate to a Car Module, paying attention to cover the exceptional and edge cases related to the NavigationInt and MapsInt interfaces.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigate to a Car Module driver must have been already implemented and I6T1 must have already been performed.

Test Case Identifier	I8T2
Test Item(s)	Navigation & Maps Manager → Look for a Car Module
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External Look for a Car Module, paying attention to cover the exceptional and edge cases related to the MapsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Look for a Car Module driver must have been already implemented and I6T1 must have already been performed.

Test Case Identifier	I8T3
Test Item(s)	Navigation & Maps Manager → App User Manager
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the NavigationInt and MapsInt interfaces.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented and I6T1 must have already been performed.

Test Case Identifier	I8T4
Test Item(s)	Navigation & Maps Manager → Navigate to a Car Module
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External Navigate to a Car Module, paying attention to cover the exceptional and edge cases related to the MapsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigate to a Car Module driver must have been already implemented and I6T1 must have already been performed.

Test Case Identifier	I8T5
Test Item(s)	Navigation & Maps Manager → Web User Manager
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External Web User Manager, paying attention to cover the exceptional and edge cases related to the MapsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Manager driver must have been already implemented and I6T1 must have already been performed.

Test Case Identifier	I8T6
Test Item(s)	Navigation & Maps Manager → Car Manager
Input Specification	Simulate Navigation & Maps Manager component typical input coming from External Car Manager, paying attention to cover the exceptional and edge cases related to the MapsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I6T1 must have already been performed.

3.1.9 Cars Availability Manager Integration Test

Test Case Identifier	I9T1
Test Item(s)	Cars Availability Manager → Car Manager
Input Specification	Simulate Cars Availability Manager component typical input coming from External Car Manager, paying attention to cover the exceptional and edge cases related to the AvailabilityInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I3T2 must have already been performed.

3.1.10 Car Manager Integration Test

Test Case Identifier	I10T1
Test Item(s)	Car Manager → Reservation Manager
Input Specification	Simulate Car Manager component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented, I3T4, I8T7 and I9T1 must have already been performed.

3.1.11 Invoice Manager Integration Test

Test Case Identifier	I11T1
Test Item(s)	Invoice Manager → Reservation Manager
Input Specification	Simulate Invoice Manager component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented, I7T1 and I3T5 must have already been performed.

Test Case Identifier	I11T2
Test Item(s)	Invoice Manager → Invoice Viewer Module
Input Specification	Simulate Invoice Manager component typical input coming from External Invoice Viewer Module, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Invoice Viewer Module driver must have been already implemented, I7T1 and I3T5 must have already been performed.

3.1.12 Log in Module Integration Test

Test Case Identifier	I12T1
Test Item(s)	Log in Module → App User Manager
Input Specification	Simulate Log in Module component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented and I3T6 must have already been performed.

Test Case Identifier	I12T2
Test Item(s)	Log in Module → Web User Manager
Input Specification	Simulate Log in Module component typical input coming from External Web User Manager, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Manager driver must have been already implemented and I3T6 must have already been performed.

Test Case Identifier	I12T3
Test Item(s)	Log in Module → Reservation Manager
Input Specification	Simulate Log in Module component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented and I3T6 must have already been performed.

3.1.13 Car Information Module Integration Test

Test Case Identifier	I13T1
Test Item(s)	Car Information Module → Look for a Car Module
Input Specification	Simulate Car Information Module component typical input coming from External Look for a Car Module, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Look for a Car Module driver must have been already implemented and I3T7 must have already been performed.

3.1.14 Sign up Module Integration Test

Test Case Identifier	I14T1
Test Item(s)	Sign up Module → Web User Module
Input Specification	Simulate Sign up Module component typical input coming from External Web User Module, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Module driver must have been already implemented, I3T8, I4T1 and I5T3 must have already been performed.

3.1.15 Reservation Manager Integration Test

Test Case Identifier	I15T1
Test Item(s)	Reservation Manager → App User Manager
Input Specification	Simulate Reservation Manager component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented, I10T1, I12T3, I3T3 and I1T1 must have already been performed.

Test Case Identifier	I15T2
Test Item(s)	Reservation Manager → Navigate to a Car Module
Input Specification	Simulate Reservation Manager component typical input coming from External Navigate to a Car Module, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigate to a Car Module driver must have been already implemented, I10T1, I12T3, I3T3 and I1T1 must have already been performed.

Test Case Identifier	I15T3
Test Item(s)	Reservation Manager → Cars Availability Manager
Input Specification	Simulate Reservation Manager component typical input coming from External Cars Availability Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Availability Manager driver must have been already implemented, I10T1, I12T3, I3T3 and I1T1 must have already been performed.

3.1.16 Navigate to a Car Module Integration Test

Test Case Identifier	I16T1
Test Item(s)	Navigate to a Car Module → App User Manager
Input Specification	Simulate Navigate to a Car Module component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the ToCarInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented, I15T2 and I8T2 must have already been performed.

3.1.17 Invoices Viewer Module Integration Test

Test Case Identifier	I17T1
Test Item(s)	Invoices Viewer Module → App User Manager
Input Specification	Simulate Invoices Viewer Module component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the InvoicesViewerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented and I11T2 must have already been performed.

Test Case Identifier	I17T2
Test Item(s)	Invoices Viewer Module → Web User Manager
Input Specification	Simulate Invoices Viewer Module component typical input coming from External Web User Manager, paying attention to cover the exceptional and edge cases related to the InvoicesViewInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Manager driver must have been already implemented and I11T2 must have already been performed.

3.1.18 Look for a Car Module Integration Test

Test Case Identifier	I18T1
Test Item(s)	Look for a Car Module → App User Manager
Input Specification	Simulate Look for a Car Module component typical input coming from External App User Manager, paying attention to cover the exceptional and edge cases related to the CarSearchInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The App User Manager driver must have been already implemented, I13T1 and I8T2must have already been performed.

Test Case Identifier	I18T2
Test Item(s)	Look for a Car Module → Web User Manager
Input Specification	Simulate Look for a Car Module component typical input coming from External Web User Manager, paying attention to cover the exceptional and edge cases related to the CarSearchInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web User Manager driver must have been already implemented, I13T1 and I8T2 must have already been performed.

3.1.19 App User Manager Integration Test

Test Case Identifier	I19T1
Test Item(s)	App User Manager → User Mobile App
Input Specification	Simulate App User Manager component typical input coming from External User Mobile App, paying attention to cover the exceptional and edge cases related to the UserInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The User Mobile App driver must have been already implemented, I12T1, I15T1, I16T1, I17T1, I18T1, I5T1 and I8T3 must have already been performed.

3.1.20 Web User Manager Integration Test

Test Case Identifier	I20T1
Test Item(s)	Web User Manager → Web Browsing Pages
Input Specification	Simulate Web User Manager component typical input coming from External Web Browsing Pages, paying attention to cover the exceptional and edge cases related to the WebInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Web Browsing Pages driver must have been already implemented, I12T2, I14T1, I17T2, I18T2, I5T2 and I8T5 must have already been performed.

3.1.21 Reservation Manager - Server Side Integration Test

Test Case Identifier	I21T1
Test Item(s)	Reservation Manager → Car Manager
Input Specification	Simulate Reservation Manager component typical input coming from External Car Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented, I10T1, I12T3, I3T3 and I1T1 must have already been performed.

Test Case Identifier	I21T2
Test Item(s)	User Mobile Manager → Reservation Manager
Input Specification	Simulate User Mobile App component typical input coming from External Reservation Manager, paying attention to cover the exceptional and edge cases related to the EmergencyInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented, I10T1, I12T3, I3T3 and I1T1 must have already been performed.

3.2 Car Server Core Subsystem

3.2.1 PC API Manager Integration Test

Test Case Identifier	I1T1
Test Item(s)	PC API Manager → Payment Manager
Input Specification	Simulate PC API Manager component typical input coming from External Payment Manager, paying attention to cover the exceptional and edge cases related to the PCInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Payment Manager driver must have been already implemented.

3.2.2 DBMS API MAnager Integration Test

Test Case Identifier	I2T1
Test Item(s)	DBMS API Manager → Payment Manager
Input Specification	Simulate DBMS API Manager component typical input coming from External Payment Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Payment Manager driver must have been already implemented.

Test Case Identifier	I2T2
Test Item(s)	DBMS API Manager → Cars Availability Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Cars Availability Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Availability Manager driver must have been already implemented.

Test Case Identifier	I2T3
Test Item(s)	DBMS API Manager → Invoices Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Invoices Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Invoices Manager driver must have been already implemented.

Test Case Identifier	I2T4
Test Item(s)	DBMS API Manager → Car Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented.

Test Case Identifier	I2T5
Test Item(s)	DBMS API Manager → Reservation Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Reservation Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

Test Case Identifier	I2T6
Test Item(s)	DBMS API Manager → Money Saving Option Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Money Saving Option Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Money Saving Option Manager driver must have been already implemented.

Test Case Identifier	I2T7
Test Item(s)	DBMS API Manager → Cars Redistribution Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Cars Redistribution Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Redistribution Manager driver must have been already implemented.

Test Case Identifier	I2T8
Test Item(s)	DBMS API Manager → Minor Issue Report Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Minor Issue Report Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Minor Issue Report Manager driver must have been already implemented.

Test Case Identifier	I2T9
Test Item(s)	DBMS API Manager → Navigation & Maps Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Navigation & Maps Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigation & Maps Manager driver must have been already implemented.

3.2.3 N&M API Manager Integration Test

Test Case Identifier	I3T1
Test Item(s)	N&M API Manager → Navigation & Maps Manager
Input Specification	Simulate N&M API Manager component typical input coming from Navigation & Maps Manager, paying attention to cover the exceptional and edge cases related to the N&MInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Navigation & Maps Manager driver must have been already implemented.

3.2.4 Timer Module Integration Test

Test Case Identifier	I4T1
Test Item(s)	Timer Module → Car Manager
Input Specification	Simulate Timer Module component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented.

Test Case Identifier	I4T2
Test Item(s)	Timer Module → Reservation Manager
Input Specification	Simulate Timer Module component typical input coming from Reservation Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

3.2.5 Payment Manager Integration Test

Test Case Identifier	I5T1
Test Item(s)	Payment Manager → Invoices Manager
Input Specification	Simulate Payment Manager component typical input coming from Invoices Manager, paying attention to cover the exceptional and edge cases related to the PaymentInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Invoices Manager driver must have been already implemented and I1T1, I2T1 must have already been performed.

3.2.6 Navigation & Maps Manager Integration Test

Test Case Identifier	I6T1
Test Item(s)	Navigation & Maps Manager → Car Manager
Input Specification	Simulate Navigation & Maps Manager component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the MapsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I3T1 must have already been performed.

Test Case Identifier	I6T2
Test Item(s)	Navigation & Maps Manager → Look for a Destination Manager
Input Specification	Simulate Navigation & Maps Manager component typical input coming from Look for a Destination Manager, paying attention to cover the exceptional and edge cases related to the MapsInt and NavigationInt interfaces.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Look for a Destination Manager driver must have been already implemented and I3T1 must have already been performed.

3.2.7 Reservation Manager - Client Side Integration Test

Test Case Identifier	I7T1
Test Item(s)	Reservation Manager - Client Side → Cars Availability Manager
Input Specification	Simulate Reservation Manager – Client Side component typical input coming from Cars Availability Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Availability Manager driver must have been already implemented and I2T5, I4T2 must have already been performed.

Test Case Identifier	I7T2
Test Item(s)	Reservation Manager - Client Side → Car Manager
Input Specification	Simulate Reservation Manager – Client Side component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I2T5, I4T2 must have already been performed.

Test Case Identifier	I7T3
Test Item(s)	Reservation Manager - Client Side → Car
Input Specification	Simulate Reservation Manager – Client Side component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarRequestsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car driver must have been already implemented and I2T5, I4T2 must have already been performed.

3.2.8 Cars Availability Manager Integration Test

Test Case Identifier	I8T1
Test Item(s)	Cars Availability Manager → Car Manager
Input Specification	Simulate Cars Availability Manager component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the AvailabilityInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I2T2, I7T1 must have already been performed.

3.2.9 Invoices Manager Integration test

Test Case Identifier	I9T1
Test Item(s)	Invoices Manager → Reservation Manager - Server Side
Input Specification	Simulate Invoices Manager component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the InvoicesInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I2T3, I5T1 must have already been performed.

3.2.10 Car Manager Integration Test

Test Case Identifier	I10T1
Test Item(s)	Car Manager → Reservation Manager - Server Side
Input Specification	Simulate Car Manager component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the CarInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I2T4, I4T1, I6T1, I8T1 must have already been performed.

Test Case Identifier	I10T2
Test Item(s)	Car Manager → Car
Input Specification	Simulate Car Manager component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarRequestsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car driver must have been already implemented and I2T4, I4T1, I6T1, I8T1 must have already been performed.

3.2.11 Reservation Manager - Server Side Integration Test

Test Case Identifier	I11T1
Test Item(s)	Car → Reservation Manager - Server Side
Input Specification	Simulate Car component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the CarCommandsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I2T5, I4T2, I9T1, I10T1 must have already been performed.

3.2.12 Look for a Destination Integration Test

Test Case Identifier	I12T1
Test Item(s)	Look for a Destination Manager → Requests Manager
Input Specification	Simulate Look for a Destination Manager component typical input coming from Requests Manager, paying attention to cover the exceptional and edge cases related to the NavigateInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Requests Manager driver must have been already implemented and I6T2 must have already been performed.

Test Case Identifier	I12T2
Test Item(s)	Look for a Destination Manager → Money Saving Option Manager
Input Specification	Simulate Look for a Destination Manager component typical input coming from Money Saving Option Manager, paying attention to cover the exceptional and edge cases related to the NavigateInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Money Saving Option Manager driver must have been already implemented and I6T2 must have already been performed.

Test Case Identifier	I12T3
Test Item(s)	Car → Look for a Destination Manager
Input Specification	Simulate Car component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarCommandsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Look for a Destination Manager driver must have been already implemented.

3.2.13 Cars Redistribution Manager Integration Test

Test Case Identifier	I13T1
Test Item(s)	Cars Redistribution Manager → Money Saving Option Manager
Input Specification	Simulate Cars Redistribution Manager component typical input coming from Money Saving Option Manager, paying attention to cover the exceptional and edge cases related to the RedistributionInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Money Saving Option Manager driver must have been already implemented and I2T7 must have already been performed.

3.2.14 Money Saving Option Integration Test

Test Case Identifier	I14T1
Test Item(s)	Money Saving Option Manager → Requests Manager
Input Specification	Simulate Money Saving Option Manager component typical input coming from Requests Manager, paying attention to cover the exceptional and edge cases related to the MoneySaveInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Requests Manager driver must have been already implemented and I2T6 must have already been performed.

Test Case Identifier	I14T2
Test Item(s)	Special Parking Area → Money Saving Option Manager
Input Specification	Simulate Special Parking Area component typical input coming from Money Saving Option Manager, paying attention to cover the exceptional and edge cases related to the FreeSlotsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Money Saving Option Manager driver must have been already implemented.

3.2.15 Minor Issue Report Manager Integration Test

Test Case Identifier	I15T1
Test Item(s)	Minor Issue Report Manager → Requests Manager
Input Specification	Simulate Minor Issue Report Manager component typical input coming from Requests Manager, paying attention to cover the exceptional and edge cases related to the MinorReportInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Requests Manager driver must have been already implemented and I2T8 must have already been performed.

3.2.16 By-Pass Manager Integration Test

Test Case Identifier	I16T1
Test Item(s)	By-Pass Manager → Requests Manager
Input Specification	Simulate By-Pass Manager component typical input coming from Requests Manager, paying attention to cover the exceptional and edge cases related to the By-PassInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Requests Manager driver must have been already implemented.

Test Case Identifier	I16T2
Test Item(s)	Car → By-Pass Manager
Input Specification	Simulate Car component typical input coming from By-Pass Manager, paying attention to cover the exceptional and edge cases related to the CarCommandsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The By-Pass Manager driver must have been already implemented.

3.2.17 Request Manager Integration Test

Test Case Identifier	I17T1
Test Item(s)	Request Manager → Car
Input Specification	Simulate Requests Manager component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarRequestsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car driver must have been already implemented and I2T8, I12T1, I14T1, I15T1, I16T1 must have already been performed.

3.3 Car Client Core Subsystem

3.3.1 Timer Module Integration Test

Test Case Identifier	I1T1
Test Item(s)	Timer Module → Ending Reservation Manager
Input Specification	Simulate Timer Module component typical input coming from External Ending Reservation Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Ending Reservation Manager driver must have been already implemented.

Test Case Identifier	I1T2
Test Item(s)	Timer Module → Driving Module
Input Specification	Simulate Timer Module component typical input coming from External Driving Module, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Driving Module driver must have been already implemented.

3.3.2 Sensors Manager Integration Test

Test Case Identifier	I2T1
Test Item(s)	Sensors Manager → Ending Reservation Manager
Input Specification	Simulate Sensors Manager component typical input coming from External Ending Reservation Manager, paying attention to cover the exceptional and edge cases related to the SensorInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Ending Reservation Manager driver must have been already implemented.

Test Case Identifier	I2T2
Test Item(s)	Sensors Manager → Driving Module
Input Specification	Simulate Sensors Manager component typical input coming from External Driving Module, paying attention to cover the exceptional and edge cases related to the SensorInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Driving Module must have been already implemented.

Test Case Identifier	I2T3
Test Item(s)	Sensors Manager → Screen Manager
Input Specification	Simulate Sensors Manager component typical input coming from External Screen Manager, paying attention to cover the exceptional and edge cases related to the SensorInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Screen Manager driver must have been already implemented.

3.3.3 Authentication Manager Integration Test

Test Case Identifier	I3T1
Test Item(s)	Authentication Manager → Driving Module
Input Specification	Simulate Authentication Manager component typical input coming from External Driving Module, paying attention to cover the exceptional and edge cases related to the AuthorizationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Driving Module driver must have been already implemented.

Test Case Identifier	I3T2
Test Item(s)	Authentication Manager → Screen Manager
Input Specification	Simulate Authentication Manager component typical input coming from External Screen Manager, paying attention to cover the exceptional and edge cases related to the AuthorizationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Screen Manager driver must have been already implemented.

3.3.4 Tow Truck Call Module Integration Test

Test Case Identifier	I4T1
Test Item(s)	Tow Truck Call Module → Screen Manager
Input Specification	Simulate Tow Truck Call Module component typical input coming from External Screen Manager, paying attention to cover the exceptional and edge cases related to the ECInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Screen Manager driver must have been already implemented.

3.3.5 Actuators Module Integration Test

Test Case Identifier	I5T1
Test Item(s)	Actuators Module → Decision Manager
Input Specification	Simulate Actuators Module component typical input coming from External Decision Manager, paying attention to cover the exceptional and edge cases related to the DriverInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Decision Manager driver must have been already implemented.

3.3.6 Decision Manager Integration Test

Test Case Identifier	I6T1
Test Item(s)	Decision Manager → Driving Module
Input Specification	Simulate Decision Manager component typical input coming from External Driving Module, paying attention to cover the exceptional and edge cases related to the EngineOnOffInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Driving Module driver must have been already implemented and I5T1 must have already been performed.

Test Case Identifier	I6T2
Test Item(s)	Decision Manager → Commands Manager
Input Specification	Simulate Decision Manager component typical input coming from External Commands Manager, paying attention to cover the exceptional and edge cases related to the ControllerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Commands Manager driver must have been already implemented and I5T1 must have already been performed.

3.3.7 Ending Reservation Manager Integration Test

Test Case Identifier	I7T1
Test Item(s)	Server → Ending Reservation Manager
Input Specification	Simulate Server components typical input coming from External Ending Reservation Manager, paying attention to cover the exceptional and edge cases related to the CarRequestInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Ending Reservation Manager driver must have been already implemented, I1T1 and I2T1 must have already been performed.

3.3.8 Screen Manager Integration Test

Test Case Identifier	I8T1
Test Item(s)	Server → Screen Manager
Input Specification	Simulate Server components typical input coming from External Screen Manager, paying attention to cover the exceptional and edge cases related to the CarRequestInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Screen Manager driver must have been already implemented, I2T3, I3T2 and I4T1 must have already been performed.

Test Case Identifier	I8T2
Test Item(s)	Screen Manager → Commands Manager
Input Specification	Simulate Screen Manager components typical input coming from External Commands Manager, paying attention to cover the exceptional and edge cases related to the OutputInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Commands Manager driver must have been already implemented, I2T3, I3T2 and I4T1 must have already been performed.

3.3.9 Commands Manager Integration Test

Test Case Identifier	I9T1
Test Item(s)	Commands Manager → Server
Input Specification	Simulate Commands Manager component typical input coming from External Server, paying attention to cover the exceptional and edge cases related to the CarCommandInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Server driver must have been already implemented, I6T2 and I8T2 must have already been performed.

3.4 Operator Server Core Subsystem

3.4.1 DBMS API Manager Integration Test

Test Case Identifier	I1T1
Test Item(s)	DBMS API Manager → Cars Availability Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Cars Availability Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Availability Manager driver must have been already implemented.

Test Case Identifier	I1T2
Test Item(s)	BDMS API Manager → Car Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented.

Test Case Identifier	I1T3
Test Item(s)	DBMS API Manager → Car Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Reservation Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

Test Case Identifier	I1T4
Test Item(s)	DBMS API Manager → Cars Redistribution Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Cars Redistribution Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Redistribution Manager driver must have been already implemented.

Test Case Identifier	I1T5
Test Item(s)	DBMS API Manager → Minor Issues Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Minor Issue Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Minor Issue Manager driver must have been already implemented.

Test Case Identifier	I1T6
Test Item(s)	DBMS API Manager → Notification Dispatcher
Input Specification	Simulate DBMS API Manager component typical input coming from Notification Dispatcher, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Notification Dispatcher driver must have been already implemented.

Test Case Identifier	I1T7
Test Item(s)	DBMS API Manager → Log in Module
Input Specification	Simulate DBMS API Manager component typical input coming from Log in Module, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Log in Module driver must have been already implemented.

Test Case Identifier	I1T8
Test Item(s)	DBMS API Manager → Ride Report Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Ride Report Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Ride Report Manager driver must have been already implemented.

Test Case Identifier	I1T9
Test Item(s)	DBMS API Manager → Paperworks Manager
Input Specification	Simulate DBMS API Manager component typical input coming from Paperworks Manager, paying attention to cover the exceptional and edge cases related to the DBInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Paperworks Manager driver must have been already implemented.

3.4.2 E-Mail API Manager

Test Case Identifier	I2T1
Test Item(s)	E-Mail API Manager → Ride Report Manager
Input Specification	Simulate E-Mail API Manager component typical input coming from Ride Report Manager, paying attention to cover the exceptional and edge cases related to the E-MailInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Ride Report Manager driver must have been already implemented.

3.4.3 Timer Module Integration Test

Test Case Identifier	I3T1
Test Item(s)	Timer Module → Car Manager
Input Specification	Simulate Timer Module component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented.

Test Case Identifier	I3T2
Test Item(s)	Timer Module → Reservation Manager
Input Specification	Simulate Timer Module component typical input coming from Reservation Manager, paying attention to cover the exceptional and edge cases related to the TimerInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager driver must have been already implemented.

3.4.4 Cars Redistribution Manager Integration Test

Test Case Identifier	I4T1
Test Item(s)	Cars Redistribution Manager → Notification Dispatcher
Input Specification	Simulate Cars Redistribution Manager component typical input coming from Notifications Dispatcher, paying attention to cover the exceptional and edge cases related to the RedistributionInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Notifications Dispatcher driver must have been already implemented and I1T4 must have already been performed.

3.4.5 Notifications Dispatcher Integration Test

Test Case Identifier	I5T1
Test Item(s)	Notification Dispatcher → Car Manager
Input Specification	Simulate Notifications Dispatcher component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the RedistributionInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I1T6 must have already been performed.

Test Case Identifier	I5T2
Test Item(s)	Operator Application → Notification Dispatcher
Input Specification	Simulate Operator Application component typical input coming from Notifications Dispatcher, paying attention to cover the exceptional and edge cases related to the NotificationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Notifications Dispatcher driver must have been already implemented.

3.4.6 Reservation Manager - Client Side Integration Test

Test Case Identifier	I6T1
Test Item(s)	Reservation Manager - Client Side → Cars Availability Manager
Input Specification	Simulate Reservation Manager – client side component typical input coming from Cars Availability Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Cars Availability Manager driver must have been already implemented and I1T3, I3T2 must have already been performed.

Test Case Identifier	I6T2
Test Item(s)	Reservation Manager - Client Side → Car Manager
Input Specification	Simulate Reservation Manager – client side component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I1T3, I3T2 must have already been performed.

Test Case Identifier	I6T3
Test Item(s)	Reservation Manager Client - Side → Car
Input Specification	Simulate Reservation Manager – client side component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarRequestsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car driver must have been already implemented and I1T3, I3T2 must have already been performed.

Test Case Identifier	I6T4
Test Item(s)	Reservation Manager - Client Side → Operator Manager
Input Specification	Simulate Reservation Manager – client side component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the ReservationInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T3, I3T2 must have already been performed.

3.4.7 Cars Availability Manager Integration Test

Test Case Identifier	I7T1
Test Item(s)	Cars Availability Manager → Car Manager
Input Specification	Simulate Cars Availability Manager component typical input coming from Car Manager, paying attention to cover the exceptional and edge cases related to the AvailabilityInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car Manager driver must have been already implemented and I1T2, I6T1 must have already been performed.

Test Case Identifier	I7T2
Test Item(s)	Cars Availability Manager → Operator Manager
Input Specification	Simulate Cars Availability Manager component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the AvailabilityInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T2, I6T1 must have already been performed.

3.4.8 Log in Module Integration Test

Test Case Identifier	I8T1
Test Item(s)	Log In Module → Reservation Manager - Server Side
Input Specification	Simulate Log in Module component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I1T7 must have already been performed.

Test Case Identifier	I8T2
Test Item(s)	Log In Module → Operator Manager
Input Specification	Simulate Log in Module component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the LogInInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T7 must have already been performed.

3.4.9 Paperworks Manager Integration Test

Test Case Identifier	I9T1
Test Item(s)	Paperworks Manager → Operator Manager
Input Specification	Simulate Paperworks Manager component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the InfoInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T9 must have already been performed.

Test Case Identifier	I9T2
Test Item(s)	Paperworks Manager → Minor Issues Manager
Input Specification	Simulate Paperworks Manager component typical input coming from Minor Issue Manager, paying attention to cover the exceptional and edge cases related to the InfoInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Minor Issue Manager driver must have been already implemented and I1T9 must have already been performed.

3.4.10 Car Manager Integration Test

Test Case Identifier	I10T1
Test Item(s)	Car Manager → Minor Issues Manager
Input Specification	Simulate Car Manager component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the CarInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I1T2, I3T1, I5T1, I6T1, I7T1 must have already been performed.

Test Case Identifier	I10T2
Test Item(s)	Car Manager → Car
Input Specification	Simulate Car Manager component typical input coming from Car, paying attention to cover the exceptional and edge cases related to the CarRequestsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Car driver must have been already implemented and I1T2, I3T1, I5T1, I6T1, I7T1 must have already been performed.

3.5 Reservation Manager - Server Side Integration Test

Test Case Identifier	I11T1
Test Item(s)	Car → Reservation Manager - Server Side
Input Specification	Simulate Car component typical input coming from Reservation Manager – server side, paying attention to cover the exceptional and edge cases related to the CarCommandsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Reservation Manager – server side driver must have been already implemented and I1T5, I3T2, I8T1, I9T1 must have already been performed.

3.6 Ride Report Manager Integration Test

Test Case Identifier	I12T1
Test Item(s)	Ride Report Manager → Operator Manager
Input Specification	Simulate Ride Report Manager component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the ReportInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T8, I2T1 must have already been performed.

3.7 Minor Issue Manager Integration Test

Test Case Identifier	I13T1
Test Item(s)	Minor Issues Manager → Operator Manager
Input Specification	Simulate Minor Issues Manager component typical input coming from Operator Manager, paying attention to cover the exceptional and edge cases related to the MinorIssuesInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Manager driver must have been already implemented and I1T5, I9T2 must have already been performed.

3.8 Operator Manager Integration Test

Test Case Identifier	I14T1
Test Item(s)	Operator Manager → Operator Application
Input Specification	Simulate Operator Manager component typical input coming from Operator Application, paying attention to cover the exceptional and edge cases related to the OperationsInt interface.
Output Specification	Check if the correct methods are invoked, along with the correct parameter types and values.
Environmental Needs	The Operator Application driver must have been already implemented and I6T4, I7T2, I8T2, I9T1, I12T1, I13T1 must have already been performed.

4 Tools and test equipment required

4.1 Test tools

In the integration phase we plan on using a set of different tools with respect to the different testing phases. In particular:

- **JUnit** (http://junit.org): one of the most popular testing frameworks for Java programming language. We will use it for the first phase of tests, wich is unnit testing, not discussed in depth in this document.

- **Arquillian** (http://arquillian.org): a testing tool designed for the JVM, expecially valid in the integration testing.

- **JMeter** (http://jmeter.apache.org): a Apache distribution designed for having accurate measures of the system's performances. To be used in the last phase of testing.

Furthermore, we plan on using a set of different tools for the measurement of the app performances and other non-functional properties. In particular we refer to the three main mobile operating systems:

- **Android**: the main tool we will use is Traceview Walkthrough, a framework embedded in Android Studio useful for monitoring methods execution time. Other useful tools for Android testing are Battery Profiler and Memory Profiler, which are respectively designed for monitoring battery consuption and memory usage.

- **iOS**: the kit provided by the Xcode IDE will be the main tool for the analysis. In particular the framework for performance profiling we will use is Instruments, which provides a quite general description of the performances of the app.

- **Windows Phone**: as a counterpart to the tools used for the previous two operating systems, Microsoft provides the Windows Performance Analyzer tool as part of the Windows Phone Application Analysis toolkit.

The greatest amount of manual work will be spent in preparing the test datasets, since all the tests will be performed by authomatic tools, as listed before.

4.2 Test equipment

In the previous paragraph we described the software tools necessary for testing. Now we point out which devices we want to use for the tests, trying to cover a descriptive subset of all the target devices for the project.
For what concerns the mobile app, the target devices will be:

- At least one Android smartphone for each display size from 4" to 6" with steps of 1/2".

- At least one Android tablet for each display size from 7" to 10" with steps of 1/2".

- At least one iOS smartphone for each product of the iOS product family.

- At least one iOS tablet for each product of the iOS product family.
- At least one Windows Phone smartphone for each display size from 4” to 7” with steps of 1/2”.

Both the mobile app and the mobile version of the website will be tested on these devices, while the desktop version of the website will be tested on devices with monitors of standard sizes (13”, 15”, 17”) and operating system Windows (version 7, 8.1, 10) and Linux (Linux Ubuntu 15), but also on the most popular iOSX devices, such as the family of MacBook.
The testing phase of the software to be embedded in the car system will be tested directly on a set of at least three sample cars.

5 Program stubs and test data required

We plan on creating a script capable of populating our database with data consistent with those that will be generated as a reaction to the actions of the users and to the communication with the cars’ sensors. The amount of data necessary for this task should be big enough to perform a stress test on the database, that is to say that the database should have a load close to the one it will handle after the complete deployment.
There are components in the architecture we developed in the DD that are provided by external sources (those modules related to APIs in the DD, such as the Navigation API and the Payment API): such components must be temporarily substituted by apposite stubs while performing the tests, so that similar functionalities are provided to the subsystems before they are put together. All these components will be integrated only after the creation of appropriate capable of implementing the communication between the API interfaces and the complete system.

6 Hours of work

The team divided the work into equivalent parts, even when modeling different parts of the document. In the following table we present a resume of the work division.

Lo Bianco Riccardo	3h 26/12, 2h 27/12, 5h 28/12
Manzoni Mirco	xxh
Mascellaro Giuseppe	xxh