

# Chapter 1

## Integration Strategy

In this chapter the integration strategy will be described. In the section 1.1 the prerequisites for the tests will be presented. In the section 1.2 is dedicated to the required and further components to be integrated in the system to execute some tests.

Finally in the section 1.3 and in the section 1.4 the strategy used to test the integrations will be discussed with paying attention to the order.

### 1.1 Entry Criteria

The criteria which are required to perform each test are easy. The involved components (and eventually needed program stubs) must have been developed and fully tested before executing the integration test.

In addition, for some test, further preconditions may be required. These prerequisites will not be described here, but in the ??, dedicated to test's description.

### 1.2 Elements to be Integrated

Up to now, no further components need to be integrated to perform integration tests.

## 1.3 Integration Testing Strategy

We have decided to test our system's interactions with a bottom-up approach. In the section 1.4 it is possible to see the components' order of implementation defined by us with specific criteria. As soon as two components are available, the tests associated to their interactions have to be applied. Before starting the development of the following part of the system, all the tests should be successful.

Obviously, to perform some tests, we need to use a *non-available* component (it has not been implemented yet): to simulate this components we will use some particular program stubs, if any, and they will be described in detail in the ??.

## 1.4 Sequence of Component/Function Integration

In this section we point out the features related to the order of the components' development. Which is our order of implementation? Why do we need to develop the component *A* before *B*? These two questions are fully discussed from now.

### 1.4.1 Software Integration Sequence

*HERE description of Client and User Handler order of implementation*

*and of Ride Allocator*

The interactions inside these two subsystems are not tested in this project phase, but when the single components are implemented. In fact, each component is tested as soon as its first implementation has been completed and until all the tests are successful the development can not be considered done.

This kind of tests are not defined in this document, but in a specific one called *Unit Test Plan Document*, not available for *myTaxiService*.

## 1.4.2 Subsystem Integration Sequence

In this section we will present the global interactions between the *myTaxiService*'s core subsystems and we will define all the interactions tests and their order.

Opssssss!! Image not available. This is embarrassing!