|  |  |
| --- | --- |
| **Test case identifier** | A1 |
| **Tested items** | QueueManager -> GPS stub |
| **Input Specification** | Request the GPS position of a taxi driver |
| **Output specification** | The coordinates are correctly received, and belong to the correct taxi |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | A2 |
| **Tested items** | MessageBroker -> ClientStub |
| **Input Specification** | 1. Send an email to a specific user 2. Send a notification, update or other type of messages to the mobile client of a specific user 3. Send a notification, update or other type of messages to the web client interface of a specific user |
| **Output specification** | 1. The email have been prepared and addressed correctly to the appropriate account 2. The message have been correctly formulated and sent 3. The message have been correctly formulated and sent |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | A3 |
| **Tested items** | DatabaseManager -> DatabaseStub |
| **Input Specification** | 1. Create, modify and delete data on the database (some relevant data: users account, rides, taxis) 2. Formulate a query and retrieve data from the database. |
| **Output specification** | 1. The data have been correctly created, modified and deleted. 2. The data have been correctly retrieved from the database |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | A4 |
| **Tested items** | UsersManager -> MessageBroker |
| **Input Specification** | Create and send a message or notification for one or more users |
| **Output specification** | The message or notification is correctly received, together with any required additional information (e.g. name of the user that must receive the message). |
| **Environmental needs** | The MessageBroker must have been tested. |

|  |  |
| --- | --- |
| **Test case identifier** | A5 |
| **Tested items** | UsersManager -> DatabaseManager |
| **Input Specification** | 1. Require the creation, modification or elimination of an user’s account. 2. Retrieve all the information about a certain user. |
| **Output specification** | 1. The creation, modification or elimination request have been correctly received and executed. 2. The requested information are retrieved and returned. |
| **Environmental needs** | The DatabaseManager must have been tested. |

|  |  |
| --- | --- |
| **Test case identifier** | A6 |
| **Tested items** | RideManager -> QueueManager |
| **Input Specification** | 1. Add or remove a taxi from a queue. 2. Require to update the position of a taxi. 3. Move a taxi to the bottom of a queue. |
| **Output specification** | 1. The taxi have been added or removed correctly. 2. The position of the taxi have been correctly updated. 3. The taxi have been moved to the bottom of the queue. |
| **Environmental needs** | The QueueManager must have been tested. |

|  |  |
| --- | --- |
| **Test case identifier** | A7 |
| **Tested items** | RideManager -> MessageBroker |
| **Input Specification** | Create and send a message or notification for one or more users related to a ride or taxi. |
| **Output specification** | The message or notification is correctly received, together with any required additional information. |
| **Environmental needs** | The MessageBroker must have been tested. |

|  |  |
| --- | --- |
| **Test case identifier** | A8 |
| **Tested items** | RideManager -> UsersManager |
| **Input Specification** | Retrieve the information about a certain user. |
| **Output specification** | The information is correctly retrieved and returned. |
| **Environmental needs** | The UsersManager must have been tested (as well as all the components needed to test the UserManager itself). |

|  |  |
| --- | --- |
| **Test case identifier** | A9 |
| **Tested items** | RideManager -> DatabaseManager |
| **Input Specification** | 1. Create, modify or delete a ride. 2. Retrieve the information about a specific ride. |
| **Output specification** | 1. The ride has been correctly created, modified or deleted. 2. The information related to the ride is correctly returned. |
| **Environmental needs** | The DatabaseManager must have been tested. |

|  |  |
| --- | --- |
| **Test case identifier** | A10 |
| **Tested items** | ApplicationServer -> RideManager |
| **Input Specification** | 1. Create or delete a ride. 2. Retrieve the information about a specific ride. 3. Associate a user to a ride. 4. Retrieve the rides associated to a user. |
| **Output specification** | 1. The ride has been correctly created or deleted. 2. The information has been correctly returned. 3. The user have been correctly associated to the ride (as a customer or driver). 4. The rides have been correctly returned. |
| **Environmental needs** | The RideManager must have been tested (as well as all the components needed to test the RideManager itself). |

|  |  |
| --- | --- |
| **Test case identifier** | A11 |
| **Tested items** | ApplicationServer -> UsersManager |
| **Input Specification** | 1. Create, modify or delete a user. 2. Check if some information (i.e. the couple email and password) are valid and belongs to a user. |
| **Output specification** | 1. The user have been correctly create, modified or deleted. 2. The validation is executed correctly. |
| **Environmental needs** | The UsersManager must have been tested (as well as all the components needed to test the UserManager itself). |

|  |  |
| --- | --- |
| **Test case identifier** | A12 |
| **Tested items** | ClientDriver -> ApplicationServer |
| **Input Specification** | 1. Register a new user. 2. Login a registered user. 3. Create, update or delete taxi rides. 4. Require the rides related to a user. 5. Require specific information about a user or a ride. 6. Communicate the availability, the acceptance and refusal of rides (by taxi drivers). 7. Perform functions reserved to administrators: create and delete driver’s accounts, modify rides’ status and drivers’ status. |
| **Output specification** | 1. The user is correctly registered. 2. The user has correctly performed the log-in. 3. The taxi ride has been correctly created, updated or deleted. 4. The rides have been correctly returned. 5. The ride’s information have been correctly returned. 6. The availability, acceptance and refusal have been correctly communicated. 7. All the functions reserved to the administrators have been correctly performed. |
| **Environmental needs** | The ApplicationServer must have been tested (as well as all the components needed to test the ApplicationServer itself). |

**Table legend**

Test case identifier – Identifies the integration test case in the image.

Tested items – Identifies the component to be tested in the following format: *Component1 -> Component2*, where *Component1* uses *Component2* (“->” symbolize the *dependency* relationship)

Input Specification – Input or context that is required to reconstruct in order to perform the integration testing. On a practical point of view, may consist in one or more method calls performed by the first component towards the second component. More inputs (i.e. different method calls that must be tested between the two components) are represented with a numbered list.

Output specification – Output, or final context, that the integration must produce. More outputs are represented with a numbered list and are related to the correspondent inputs.

Enviromental needs – Preconditions needed before proceeding with the integration testing of these components.

[Stub and high level view comment]

|  |  |
| --- | --- |
| **Test case identifier** | W1 |
| **Tested items** | WebServer -> AppServerStub |
| **Input Specification** | Require the execution of a service requested by the client. |
| **Output specification** | The service requested by the client have been correctly mapped to the correspondent ApplicationServer’s service, which have been invoked. |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | W2 |
| **Tested items** | WebServer –> ClientStub |
| **Input Specification** | Create and send a web page to the client. |
| **Output specification** | The web page is correctly received. |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | W3 |
| **Tested items** | Client Driver -> WebServer |
| **Input Specification** | Require the execution of a service. |
| **Output specification** | The service has been correctly identified and executed. |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | W4 |
| **Tested items** | AppServerBroker Driver -> WebServer |
| **Input Specification** | Send notification or other messages. |
| **Output specification** | The message or notification have been received and handled accordingly. |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | C1 |
| **Tested items** | MobileClient -> AppServer Stub |
| **Input Specification** | Require the execution of a service. |
| **Output specification** | The service request is sent and identified correctly. |
| **Environmental needs** | None |

|  |  |
| --- | --- |
| **Test case identifier** | C2 |
| **Tested items** | MobileNotifications -> MobileClient |
| **Input Specification** | Send a notification or other message to the mobile client. |
| **Output specification** | The notification or message is correctly received and handled. |
| **Environmental needs** | None |