

INFORMATION TECHNOLOGIES FOR SHIFT TO RAIL

D5.4 – Travel companion Core Integration Report

Due date of deliverable: 31/08/2016

Actual submission date: 26/09/2016

Leader/Responsible of this Deliverable: Nora Winninger (SNCF)

Reviewed: Y

Document status		
Revision	Date	Description
0	19/05/2016	Template creation
1	31/08/2016	Test campaign execution report
2	13/09/2016	Modification to be compliant with another template
3	21/09/2016	Final version after TMC approval

Project funded from the European Union's Horizon 2020 research and innovation programme		
Dissemination Level		
PU	Public	X
CO	Confidential, restricted under conditions set out in Model Grant Agreement	
CI	Classified, information as referred to in Commission Decision 2001/844/EC	

Start date of project: 01/05/2015

Duration: 30 months



Contract No. H2020 – 636078



EXECUTIVE SUMMARY

This document is the integration report of the Travel Companion for the Core Release. It describes the campaign strategy to understand the motives behind the test cases. Then, it details all the test case for this campaign. Finally, it gives result for the first run of the test campaign.

TABLE OF CONTENTS

Executive Summary	2
List of Figures	5
List of Tables	5
1. Introduction	6
2. TEST MATERIALS DESCRIPTION	6
2.1 Configuration of the client side	6
2.1.1 Infrastructure and Hardware	6
2.1.2 Setup & configuration	6
2.1.3 Tested system	7
2.1.4 System data parameters.....	7
2.1.5 Simulators	7
2.1.6 Personnel	7
2.2 Configuration of the server side	7
2.2.1 Infrastructure and Hardware	7
2.2.2 Setup & configuration	7
2.2.3 Tested system	7
2.2.4 System data parameters.....	7
2.2.5 Simulators	8
2.2.6 Personnel	8
3. Test Descriptions	8
3.1 functional tests	8
3.1.1 User account creation and first search for an offer.....	9
3.1.2 User login and Get Preferences.....	11
3.1.3 Search and pay offer	13
3.1.4 Set and retrieve preference	14
3.1.5 Start tracking trip	18
3.2 Technical Tests.....	19
3.2.1 Reinstall application.....	19
3.3 Error tests	21
3.3.1 Existing email	21
3.3.2 Wrong credentials.....	23

3.3.3 Not Corresponding password	24
4. TEST EXECUTION.....	26
4.1 functional tests	26
4.1.1 User account creation and first search.....	26
4.1.2 User login and Get Preferences.....	31
4.1.3 Search and pay offer	38
4.1.4 Set and retrieve preference	44
4.1.5 Start tracking trip	49
4.2 Technical Tests.....	52
4.2.1 Reinstall Application	52
4.3 Error tests	59
4.3.1 Existing email	59
4.3.2 Wrong credentials.....	62
4.3.3 Not Corresponding password	65

LIST OF FIGURES

NA

LIST OF TABLES

NA

1. INTRODUCTION

The objective of the C-Rel test strategy is to thoroughly validate the functionalities developed for this release. In particular, the test concerns the functionalities of the client application and the calls that can be made to all relevant web services, either to travel companion cloud or to other WPs. The goals of the tests are:

- To check that the components developed within WP5 can work together coherently. For instance, to check that we can search an offer after a login, or after creating an account. Hence, test scenarios are designed by chaining atomic operations together, such as searching for an offer, logging in, or setting a preference.
- To check the coherence between the specification of the web services and their implementation (if a web service is not compliant with its interface contract, using it in the client application can become impossible).
- To test the availability of web services for external clients (the travel companion personal application will access web service through a 4G mobile network, so we need to check that web services used by it are accessible outside of private networks).

Our test campaign will also investigate, if possible, error conditions for web services (e.g., the creation of an account associated with an already existing email).

The tests presented in this document focus on the Travel companion, and should not be considered as integration tests for other WPs. In particular, we will not carry out any stress tests for these web services.

2. TEST MATERIALS DESCRIPTION

2.1 CONFIGURATION OF THE CLIENT SIDE

In this section we detail the configuration of the mobile device used to carry out the test scenarios in practice.

2.1.1 Infrastructure and Hardware

An Android mobile device, compatible with Android 6.0. This device must be able to access a 4G network.

2.1.2 Setup & configuration

The device does not need any special configuration, besides having the travel companion personal application installed. The only requirement is for the mobile device to be able to run a test version of the application (i.e., the operation system must be configured in development mode).

2.1.3 Tested system

Travel companion Personal application.

2.1.4 System data parameters

NA

2.1.5 Simulators

No Android simulators are used, as there may be some subtle differences compared with the behaviour of a real phone.

2.1.6 Personnel

NA

2.2 CONFIGURATION OF THE SERVER SIDE

In this section we detail the configuration of the travel companion cloud, i.e., the server side of the application.

2.2.1 Infrastructure and Hardware

The cloud environment is constituted by two connected servers based on Linux CentOS 6.7 (64 bits). The first one includes a Wildfly v8.1 Application Server that manages all the interfaces between the travel companion cloud and WP5 modules (such as the travel companion mobile device) as well as other WPs. The second one includes a PostgreSQL database to store user information. The two servers interact to exchange user data, storing and retrieving user information from the database, in order to support other WPs' modules.

2.2.2 Setup & configuration

No special configuration is necessary, aside from the need to have the server-side applications installed. The only requirement is related to remote access to the cloud, being essential to use a VPN client to connect to it in order to ensure environment security. VPN connection supports SSL connection and IPsec and it is possible to access VPN throughout java code using a Runtime call to execute Linux commands.

2.2.3 Tested system

Access Manager and Preference Wallet modules of the Travel companion cloud.

2.2.4 System data parameters

NA

2.2.5 Simulators

No simulators are needed, since all modules interacting with the tested ones are operative and the interface contract has been defined before the definition of the tests cases. However, SOAP UI has been defined as the simulator for testing cloud response in case of interface failure, in order to ensure cloud operation.

2.2.6 Personnel

NA

3. TEST DESCRIPTIONS

3.1 FUNCTIONAL TESTS

These tests are those that are directly inspired by the specification

3.1.1 User account creation and first search for an offer

WP5TestCase1-1	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that the user can create an account, and that he is connected after the account creation.
Description	The user will create an account, set the associated preferences, and then search an offer.
Status	
% passed	

Normal client and server configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: - Application installed on mobile device				

Id	Step description	Expected result	Observed result	State	Associated defect
1	<p>Account Creation :</p> <p>Tester launches the application, then goes to the create account screen.</p> <p>Tester creates an account: he set testCrel@sncf20160830.fr in login field, testpwd in the password and password verification field, and touch the register button.</p> <p>Remark: the login can be used only once. So if needed, increment the login : testCrel@sncf20160830-2.fr</p>	<p>Create account confirmation message in the application (demonstration)</p> <p>Account present in the wallet (inspection)</p>			
2	<p>Preference Setting :</p> <p>Tester goes to the preference screen, and set some preferences: he checks:</p> <ul style="list-style-type: none"> the cross in front of Fast Long Distance Train the one in front of Trenitalia the one in front of AirFrance 	<p>Preferences changed on screen (demonstration)</p> <p>Preferences present in the wallet (inspection)</p>			
3	<p>Search an offer :</p> <p>Tester goes to the search offer screen, and searches an offer, with a round trip. He selects</p> <ul style="list-style-type: none"> Origin Piazza Navona Destination :Van Gogh Museum in Amsterdam 	<p>Offer list returned by Shopping module shown in list offer activity (demonstration)</p>			

3.1.2 User login and Get Preferences

WP5TestCase1-2	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that a user can use an account after its creation
Description	The user will empty the cache, then login, and then set some preferences. Then he will empty the cache again, login again, and checks the preferences he set are still here
Status	
% passed	

Normal client and server configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: User is able to empty application cache				

Id	Step description	Expected result	Observed result	State	Associated defect
1	<p>Empty Application Cache</p> <p>Tester close the application</p> <p>Tester empty the application cache in Android preference</p> <p>Tester launches the application</p>	<p>The application launches on the device</p>			
2	<p>Login :</p> <p>Tester goes to the login activity</p> <p>Tester enters these credentials :</p> <ul style="list-style-type: none"> - Login : arya@got.org - Password : stark <p>The login in the login field, and the password in the password field. Then user touches the connect button.</p>	<p>Login message confirmation (Demonstration)</p>			
3	<p>Preference Setting :</p> <p>Tester goes to the preference screen, and set some preferences: he checks:</p> <ul style="list-style-type: none"> • the cross in front of Fast Long Distance Train • the one in front of Trenitalia • the one in front of Air France 	<p>Preferences changed on screen (demonstration)</p> <p>Preferences present in the wallet (inspection)</p>			
4	<p>Empty Application Cache</p> <p>Tester repeats step 1</p>	<p>The application launch on the device</p>			

Id	Step description	Expected result	Observed result	State	Associated defect
5	Login Tester repeats step 2	Login message confirmation (Demonstration)			
6	Check Preferences Tester goes to the preference screen. Tester checks that preferences set in step 3 are still present.				

3.1.3 Search and pay offer

WP5TestCase1-3	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that the user can search for an offer after a login, done in test 1-2
Description	The user will search for an offer, and then book and pay it
Status	
% passed	

Normal client and server configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
Preconditions: User logged in, after WP5TestCase1-2					
1	Search an offer Tester searches an offer: <ul style="list-style-type: none"> Origin Piazza Navona Destination Van Gogh Museum 	The application will show 3 offers in the offer list (Demonstration)			
2	Check offers Prices are 320, 330 and 320.	All Offers are coherent with tester's request (Demonstration)			
3	Book the most expensive offer Tester touch the book button in the offer list for the offer with a cost of 330 €.	Pay screen opens			
4	Pay the offer Check the price in the pay screen is 330 €. The user touches the pay button.	User get a confirm payment message (Demonstration) Offer and Entitlement present in Wallet (not for the Core Release)s			

3.1.4 Set and retrieve preference

WP5TestCase1-4	
Method Of Test	Demonstration and Inspection
Type of test	Manual

WP5TestCase1-4	
Objectives	Check that the user can set and retrieve preferences
Description	The user will set the preferences, change screen, and come back to the preference screen. The preferences are not cached. They are reloaded from the server every time.
Status	
% passed	

Normal client and server configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: User logged in, either after test WP5TestCase1-1 or WP5TestCase1-2 ; and tester must know this login credential				
1	Set Preferences Tester goes to the preference activity. Then he set some preferences : he check Long Distance Train	The preferences set by tester are in the Wallet (Inspection)			
2	Change screen Tester goes to the search offer screen				

Id	Step description	Expected result	Observed result	State	Associated defect
3	Check preferences Tester goes to the preference activity, and checks that Long distance train is still checked	Long Distance train still checked.			

Id	Step description	Expected result	Observed result	State	Associated defect
5	<p>Check all preferences</p> <p>Tester reproduces step 1 to 4, with consecutively :</p> <ul style="list-style-type: none"> • Train • Commuter Train • Suburban Railway • Subway • Tram • Bus • Hailed Shared Taxi • Ferry • Taxi • Car • Car sharing • Carpooling • Bike • Bike sharing • Walk • Trenitalia • SNCF • Air France • Lufthansa • Cartafreccia • Flying Blue • Direct Travel • Each PRM type • Each PRM Parameters • Each class • Each seat 	<p>Selected preference still checked.</p>	<p>26/09/2016</p>		

3.1.5 Start tracking trip

This test will check the ability of the application to contact the trip tracker module.

WP5TestCase1-5	
Method Of Test	Inspection
Type of test	Manual
Objectives	Test the availability of the trip tracker interface
Description	User will just click a button on a dedicated window. A log message will be written by the application
Status	
% passed	

Standard configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Go to Start Tracking activity Touch Start Tracking button	A log message like this one : StartTracking Retrofit response success: retrofit2.Response@3d99d6d**200**StartTrackingResponse{activationResponseMessage='OK'} (Address may change) (Introspection)			

3.2 TECHNICAL TESTS

These tests are intended to check problems that can arise with the implementation of any mobile application (i.e., they're not specific to the travel companion).

3.2.1 Reinstall application

WP5TestCase2-1	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that the application will still work after being updated, and cached data are still available
Description	User will login, set preference, uninstall application, and then install the application with a greater version number

WP5TestCase2-1	
Status	
% passed	

Standard configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: 2 version of the application, the older and the newer				
1	Install older version Tester launches the application He log in with this credential : Login arya@got.org Password : stark He goes to the preference screen. He checks the following preferred transport mode: bus, tram, subway.	User logged in, and preferences set			

Id	Step description	Expected result	Observed result	State	Associated defect
2	Uninstall the application The tester uninstalls the application	Application no more present in the device			
3	Install newer version of the application Tester will install the other, more recent version of the application.	Application present again in the device			
4	Login, and check Tester launches the application. Tester log into the application, with the credentials of step1. Tester goes to the preference screen. Tester checks that the preferred transport mode bus, tram and subway are checked.	Tester is able to launch the application and preferences are still here.			

3.3 ERROR TESTS

These tests are intended to check what happens in case of misuses of the application by the user.

3.3.1 Existing email

WP5TestCase3-1	
Method Of Test	Demonstration
Type of test	Manual

WP5TestCase3-1	
Objectives	Check that the user cannot create an account with an already existing email
Description	The tester will try to create an account with test credentials, to trigger the error in the ID Manager Stub
Status	
% passed	

NB as Identity manager in cloud was not in CRel perimeter, a stub has been developed. This stub returns the existing email only with the credentials above

Standard configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed on the device				

Id	Step description	Expected result	Observed result	State	Associated defect
1	<p>Account Creation :</p> <p>Tester launches the application, then goes to the create account screen.</p> <p>Tester creates an account with these credentials:</p> <p> Login: cersei@got.org</p> <p> Password: testpwd</p> <p> Password confirmation: testpwd</p> <p>Tester touches the register button.</p> <p>Notice : the login to choose must not have been selected for another test. Also tester must remember the credential use for step 3</p>	Error message			

3.3.2 Wrong credentials

WP5TestCase3-2	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that valid credentials are necessary to connect
Description	Tester will try to login with invalid credentials
Status	
% passed	

Standard configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed in the device, and cache emptied				
1	Tester will log in with a wrong credential: Tester launches the application. He goes to the login screen. He uses this credentials: Login: arya@got.org Password : WrongPassword	Error Message			

3.3.3 Not Corresponding password

WP5TestCase3-2	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that a correct password confirmation is necessary to create an account
Description	Tester will try to create an account with no corresponding password

WP5TestCase3-2	
Status	
% passed	

Standard configuration	
Regression	
Test Case Tester	

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed in the device, and cache emptied				
1	Account Creation : Tester launches the application, then goes to the create account screen. Tester creates an account: he set testCrel@sncf20160830.fr in login field, testpwd in the password and testpawdtestpwd in password verification field, and touches the register button.	Error Message			

4. TEST EXECUTION

4.1 FUNCTIONAL TESTS

4.1.1 User account creation and first search

WP5TestCase1-1	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that the user can create an account, and that he is connected after the account creation.
Description	The user will create an account, set the associated preferences, and then search an offer.
Status	OK
% passed	100%

Normal client and server configuration	
Regression	NA
Test Case Tester	FChenebit and HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: - Application installed on mobile device				

Id	Step description	Expected result	Observed result	State	Associated defect
1	<p>Account Creation :</p> <p>Tester launches the application, then goes to the create account screen.</p> <p>Tester creates an account: he set testCrel@sncf20160830.fr in login field, testpwd in the password and password verification field, and touch the register button.</p> <p>Remark: the login can be used only once. So if needed, increment the login : testCRel@sncf20160830-2.fr</p>	<p>Create account confirmation message in the application (demonstration)</p> <p>Account present in the wallet (inspection)</p>	Cf. fig 1 and 2	Passed	
2	<p>Preference Setting :</p> <p>Tester goes to the preference screen, and set some preferences: he checks:</p> <ul style="list-style-type: none"> the cross in front of Fast Long Distance Train the one in front of Trenitalia the one in front of AirFrance 	<p>Preferences changed on screen (demonstration)</p> <p>Preferences present in the wallet (inspection)</p>	Cf. fig 3	Passed	
3	<p>Search an offer :</p> <p>Tester goes to the search offer screen, and searches an offer, with a round trip. He selects</p> <ul style="list-style-type: none"> Origin Piazza Navona Destination :Van Gogh Museum in Amsterdam 	<p>Offer list returned by Shopping module shown in list offer activity (demonstration)</p>	Cf. fig 4 & 5	Passed	

Fig 1 Register screen

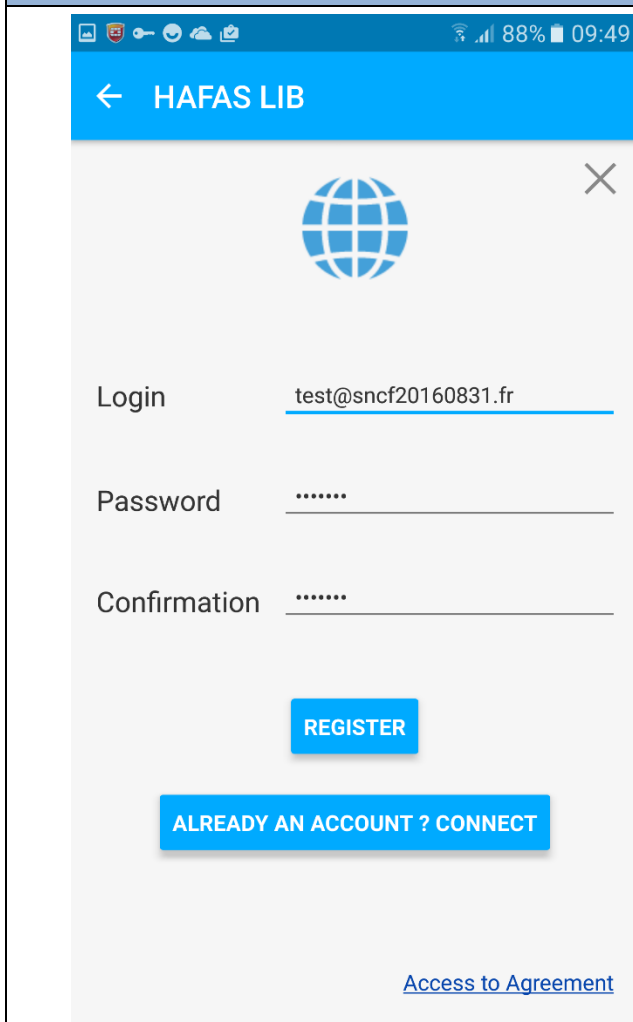


Fig 1 shows the Register screen of the HAFAS LIB application. The screen has a blue header with a back arrow and the text "HAFAS LIB". Below the header is a globe icon with a close button (X). The main form contains three input fields: "Login" with the text "test@sncf20160831.fr", "Password" with six dots, and "Confirmation" with six dots. Below the input fields are two buttons: "REGISTER" and "ALREADY AN ACCOUNT ? CONNECT". At the bottom right, there is a link "Access to Agreement". The status bar at the top shows the time as 09:49 and battery level at 88%.

Fig 2 Register screen answer

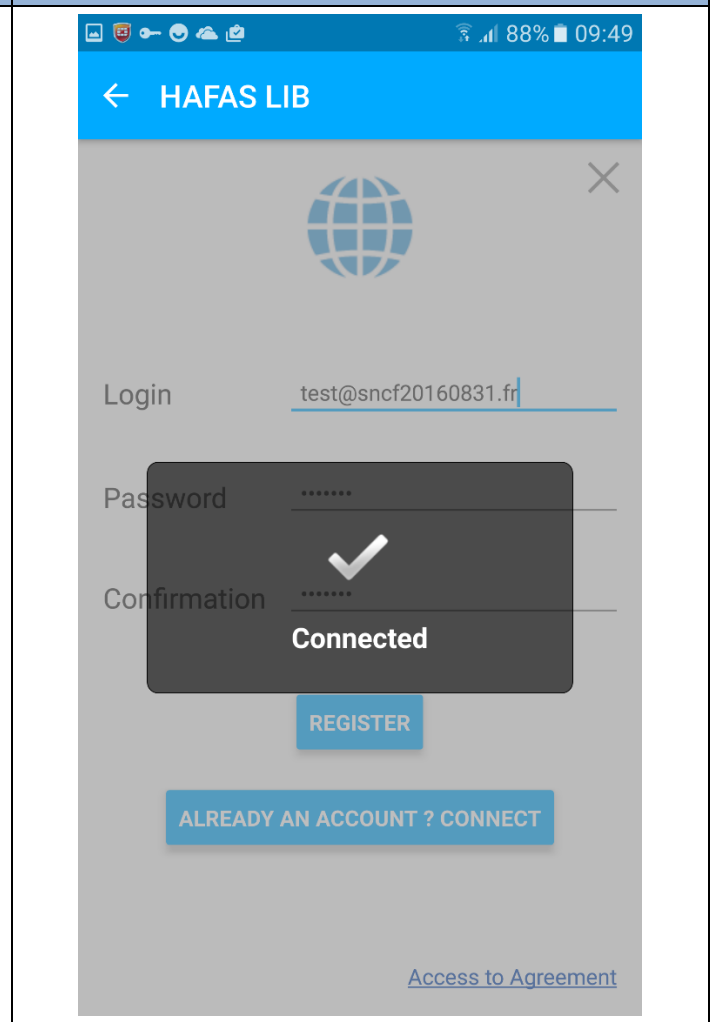
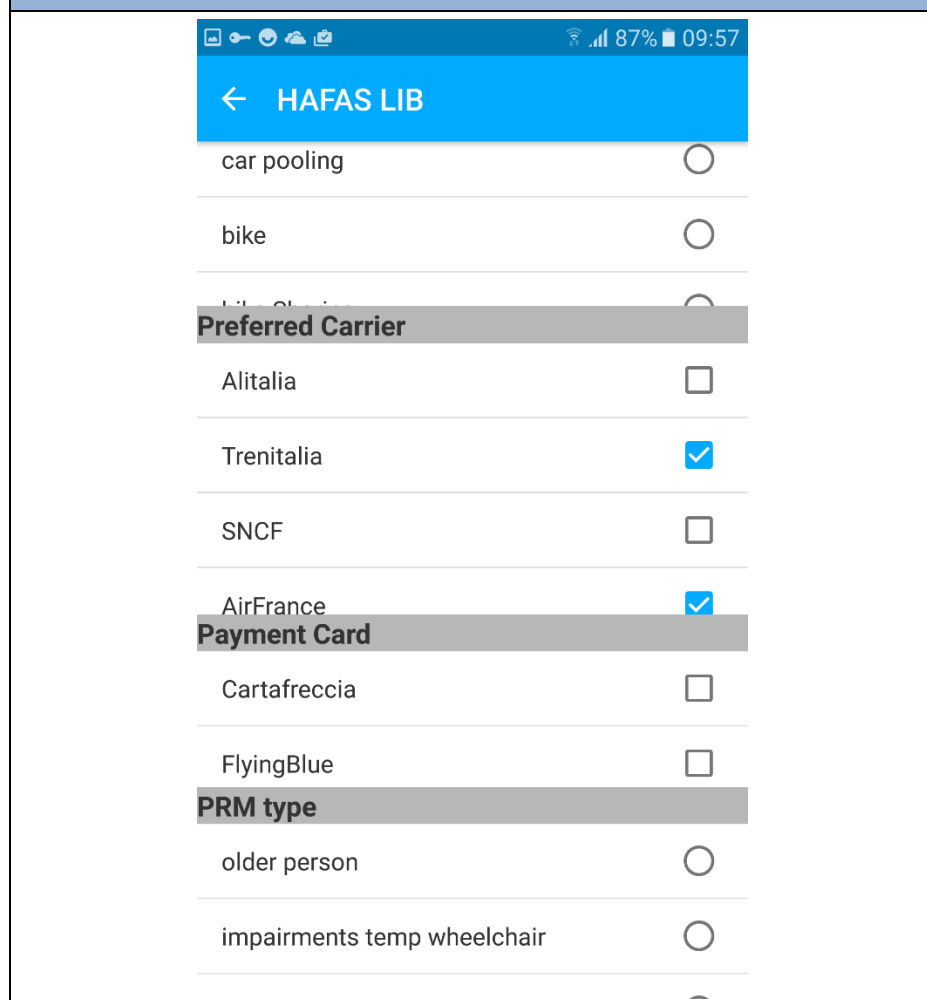


Fig 2 shows the Register screen of the HAFAS LIB application after a successful registration. The screen is identical to Fig 1, but a dark grey modal dialog box is overlaid in the center. The dialog box contains a white checkmark icon and the text "Connected". The background of the screen is dimmed. The status bar at the top shows the time as 09:49 and battery level at 88%.

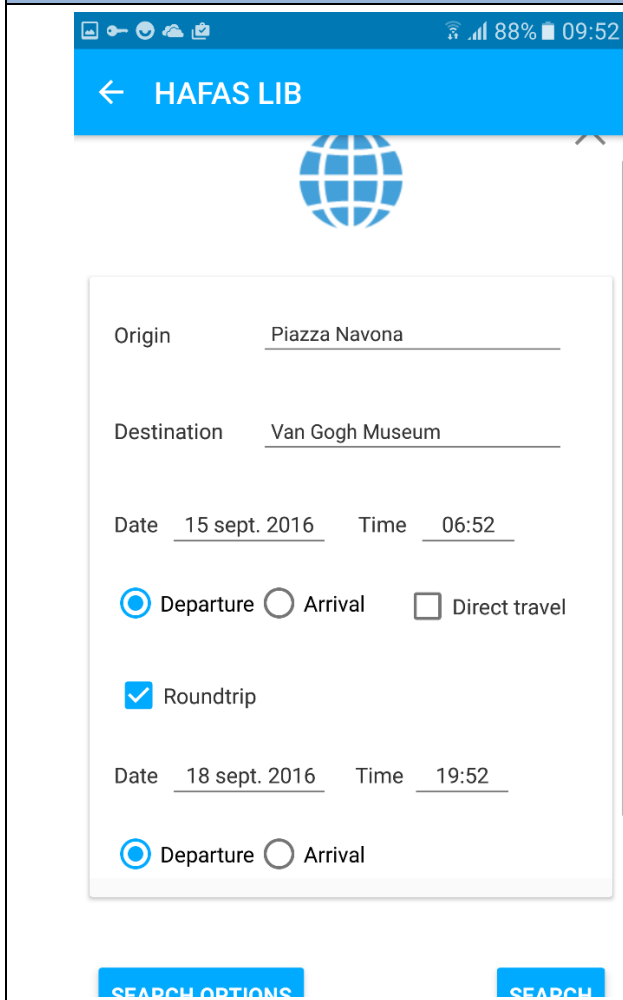
Fig 3 Preference screen



The screenshot shows a mobile application interface for 'HAFAS LIB'. The status bar at the top indicates 87% battery and the time 09:57. The app header is blue with a back arrow and the text 'HAFAS LIB'. The main content area lists various preferences with checkboxes or radio buttons:

- car pooling: ☐
- bike: ☐
- Preferred Carrier** (Section Header)
- Alitalia: ☐
- Trenitalia: ☒
- SNCF: ☐
- AirFrance: ☒
- Payment Card** (Section Header)
- Cartafreccia: ☐
- FlyingBlue: ☐
- PRM type** (Section Header)
- older person: ☐
- impairments temp wheelchair: ☐

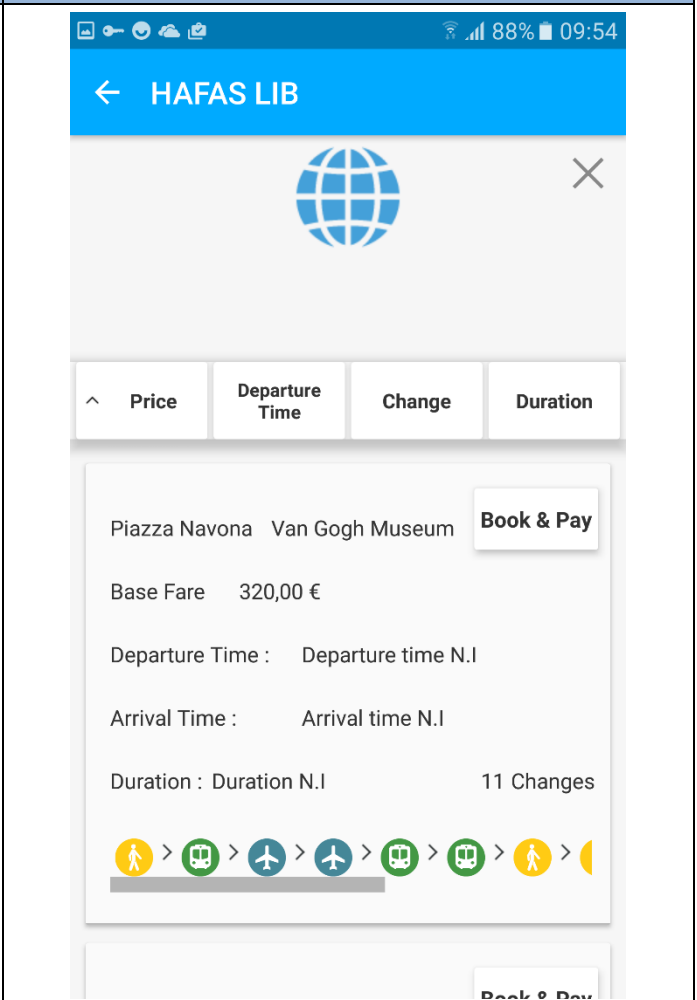
Fig 4 Search offer



Mobile app interface for searching travel offers. The screen shows a search form with the following fields and options:

- Origin:** Piazza Navona
- Destination:** Van Gogh Museum
- Date:** 15 sept. 2016 **Time:** 06:52
- Options:**
 - ☒ Departure ☐ Arrival ☐ Direct travel
 - ☒ Roundtrip
 - Date:** 18 sept. 2016 **Time:** 19:52
 - ☒ Departure ☐ Arrival
- Buttons:** SEARCH OPTIONS, SEARCH

Fig 5 Search offer result list



Mobile app interface showing search results. The screen displays a list of results with the following details:

- Header:** HAFAS LIB
- Buttons:** Price, Departure Time, Change, Duration
- Results:**
 - Origin:** Piazza Navona **Destination:** Van Gogh Museum **Book & Pay**
 - Base Fare:** 320,00 €
 - Departure Time:** Departure time N.I
 - Arrival Time:** Arrival time N.I
 - Duration:** Duration N.I **11 Changes**
- Visuals:** A sequence of icons representing the travel route: a person, a train, an airplane, a train, a train, a train, a person, and a train.
- Buttons:** Book & Pay

4.1.2 User login and Get Preferences

WP5TestCase1-2	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that a user can use an account after its creation
Description	The user will empty the cache, then login, and then set some preferences. Then he will empty the cache again, login again, and checks the preferences he set are still here
Status	OK
% passed	100 %

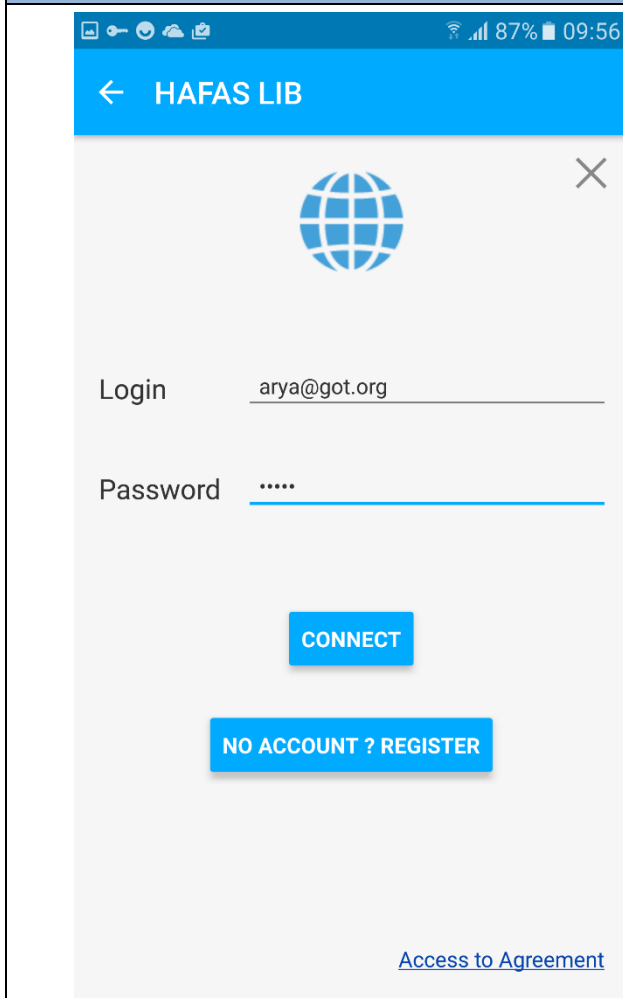
Normal client and server configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: User is able to empty application cache				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Empty Application Cache Tester close the application Tester empty the application cache in Android preference Tester launches the application	The application launches on the device	Application launched	Passed	
2	Login : Tester goes to the login activity Tester enters these credentials : <ul style="list-style-type: none"> - Login : arya@got.org - Password : stark The login in the login field, and the password in the password field. Then user touches the connect button.	Login message confirmation (Demonstration)	Cf fig 1 & fig 2	Passed	
3	Preference Setting : Tester goes to the preference screen, and set some preferences: he checks: <ul style="list-style-type: none"> • the cross in front of Fast Long Distance Train • the one in front of Trenitalia • the one in front of Air France 	Preferences changed on screen (demonstration) Preferences present in the wallet (inspection)	CF fig 3	Passed	
4	Empty Application Cache Tester repeats step 1	The application launch on the device	Application launches	Passed	

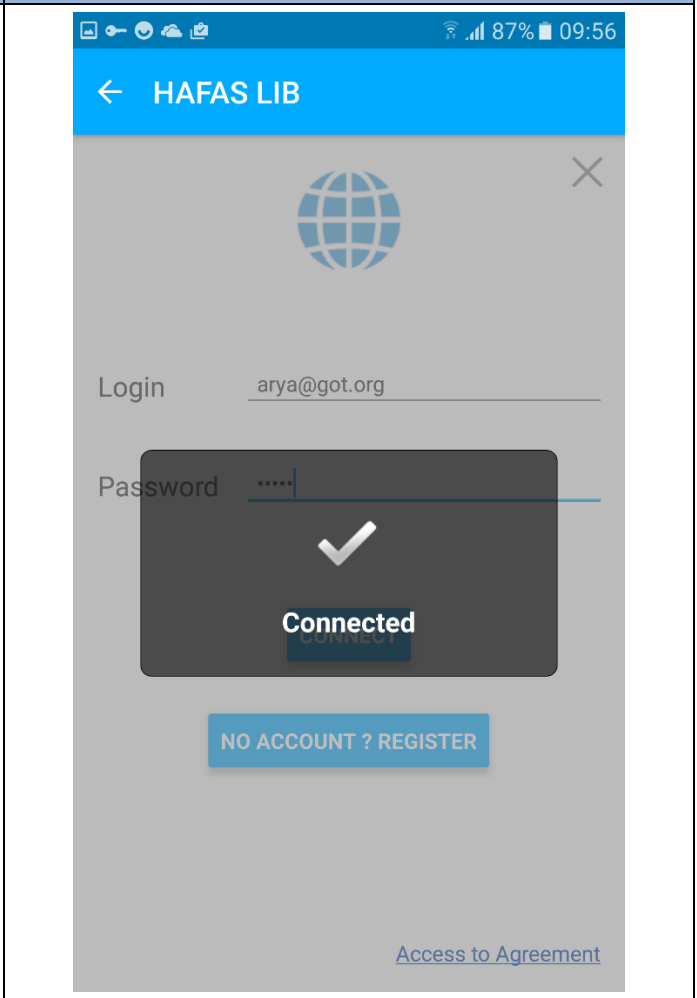
Id	Step description	Expected result	Observed result	State	Associated defect
5	Login Tester repeats step 2	Login message confirmation (Demonstration)	Cf fig 4 & 5	Passed	
6	Check Preferences Tester goes to the preference screen. Tester checks that preferences set in step 3 are still present.		Cf fig 6	Passed	

Fig 1 login



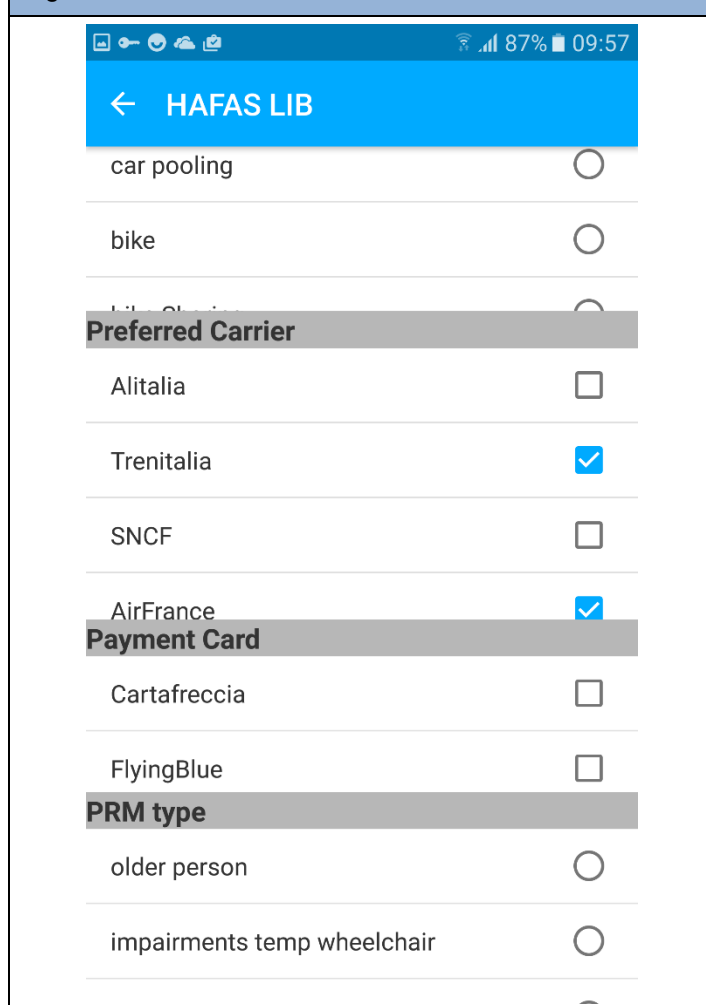
The screenshot shows the login interface of the HAFAS LIB mobile application. At the top, there is a blue header with a back arrow and the text "HAFAS LIB". Below the header is a grey area containing a globe icon and a close button (X). The main form has two input fields: "Login" with the email "arya@got.org" and "Password" with masked characters "....". Below the password field is a blue "CONNECT" button. Further down is a blue button labeled "NO ACCOUNT ? REGISTER". At the bottom right, there is a blue link that says "Access to Agreement". The status bar at the top shows a battery level of 87% and the time 09:56.

Fig 2 login answer



This screenshot shows the same login interface as Fig 1, but with a dark grey overlay indicating a successful login. The overlay features a white checkmark and the text "Connected". The "CONNECT" button is partially obscured by the overlay. The rest of the interface, including the header, input fields, and bottom links, remains visible. The status bar at the top shows the same battery level and time as in Fig 1.

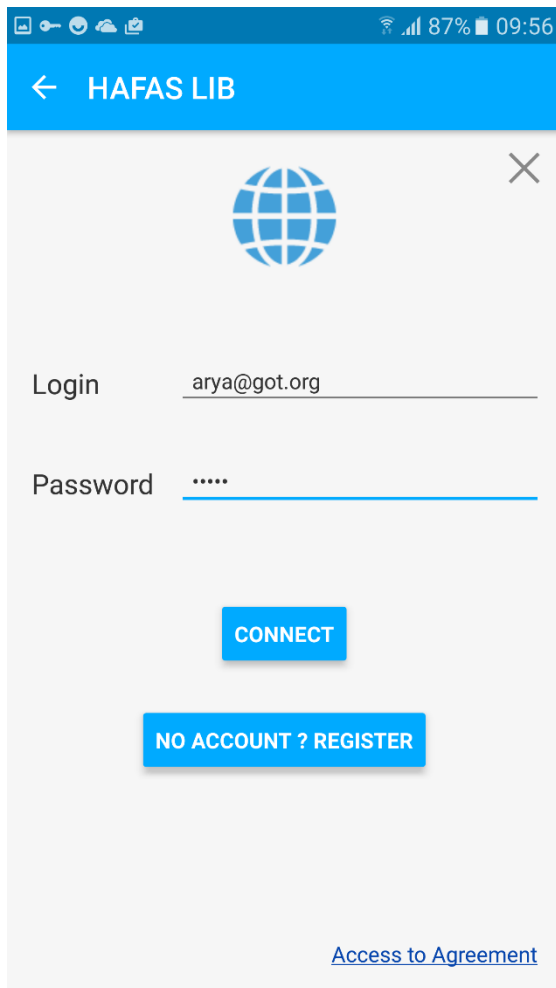
Fig 3 Preference screen



The screenshot shows a mobile application interface for 'HAFAS LIB'. At the top, there is a blue header bar with a back arrow and the text 'HAFAS LIB'. Below this, the screen is divided into several sections, each with a title and a list of options with checkboxes or radio buttons.

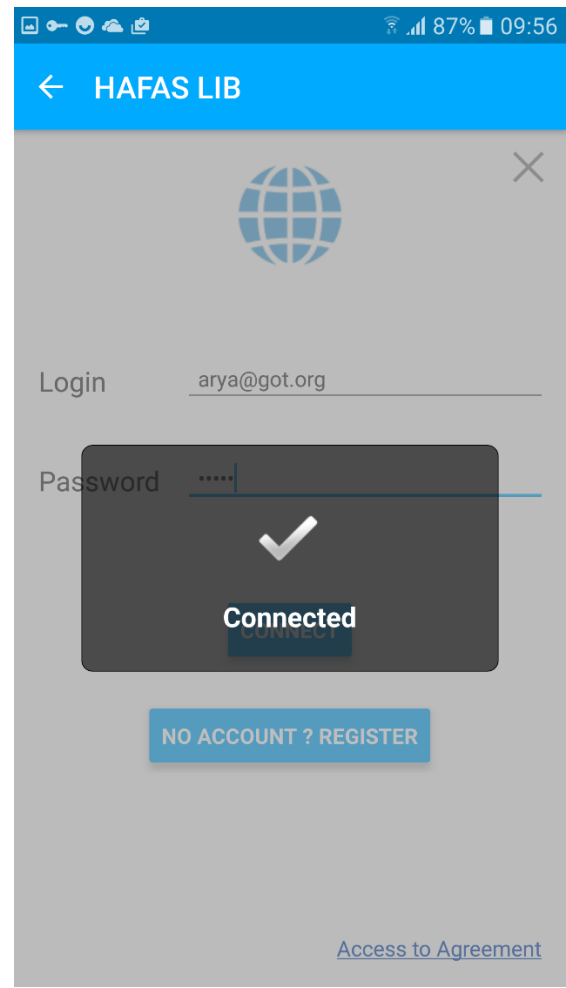
- car pooling** ☐
- bike** ☐
- Preferred Carrier** (Section Header)
 - Alitalia ☐
 - Trenitalia ☒
 - SNCF ☐
 - AirFrance ☒
- Payment Card** (Section Header)
 - Cartafreccia ☐
 - FlyingBlue ☐
- PRM type** (Section Header)
 - older person ☐
 - impairments temp wheelchair ☐

Fig 4 login



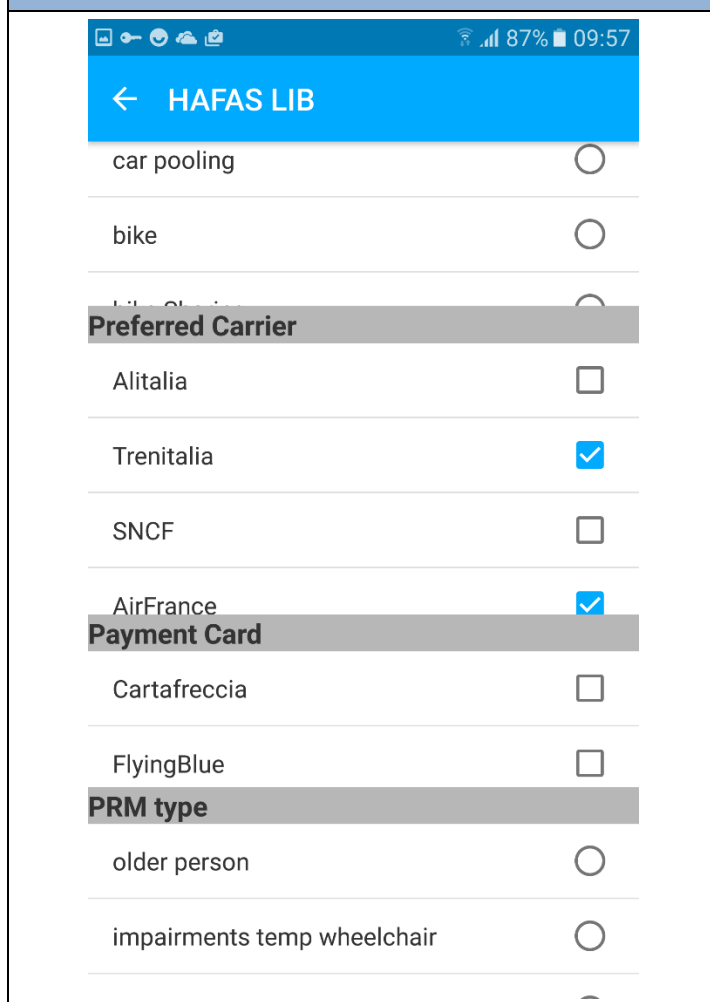
Mobile app login screen. The status bar at the top shows signal strength, 87% battery, and the time 09:56. The app header is blue with a back arrow and the text 'HAFAS LIB'. Below the header is a grey box containing a globe icon and a close button (X). The login form has two fields: 'Login' with the email 'arya@got.org' and 'Password' with masked characters '.....'. Below the fields are two buttons: 'CONNECT' and 'NO ACCOUNT ? REGISTER'. At the bottom right is a link 'Access to Agreement'.

Fig 5 login answer



Mobile app login screen showing a successful login. A dark grey overlay with a white checkmark and the text 'Connected' is centered on the screen. The background is dimmed. The status bar, app header, and login form fields are visible behind the overlay. The 'CONNECT' button is partially obscured by the overlay.

Fig 6 Preference screen



The screenshot shows a mobile application interface for 'HAFAS LIB'. At the top, there is a status bar with icons for home, back, and search, along with signal strength, 87% battery, and the time 09:57. Below the status bar is a blue header with a back arrow and the text 'HAFAS LIB'. The main content area is a list of preferences, each with a radio button or checkbox on the right. The preferences are grouped into sections: 'car pooling', 'bike', 'Preferred Carrier', 'Payment Card', and 'PRM type'. The 'Preferred Carrier' section includes 'Alitalia', 'Trenitalia' (checked), and 'SNCF'. The 'Payment Card' section includes 'Cartafreccia' and 'FlyingBlue'. The 'PRM type' section includes 'older person' and 'impairments temp wheelchair'.

Preference	Selected
car pooling	<input type="radio"/>
bike	<input type="radio"/>
Preferred Carrier	
Alitalia	<input type="checkbox"/>
Trenitalia	<input checked="" type="checkbox"/>
SNCF	<input type="checkbox"/>
AirFrance	<input checked="" type="checkbox"/>
Payment Card	
Cartafreccia	<input type="checkbox"/>
FlyingBlue	<input type="checkbox"/>
PRM type	
older person	<input type="radio"/>
impairments temp wheelchair	<input type="radio"/>

4.1.3 Search and pay offer

WP5TestCase1-3	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that the user can search for an offer after a login, done in test 1-2
Description	The user will search for an offer, and then book and pay it
Status	OK
% passed	100 %

Normal client and server configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: User logged in, after WP5TestCase1-2				
1	Search an offer Tester searches an offer: <ul style="list-style-type: none"> Origin Piazza Navona Destination Van Gogh Museum 	The application will show 3 offers in the offer list (Demonstration)	Cf Fig1 & Fig 2	Passed	

Id	Step description	Expected result	Observed result	State	Associated defect
2	Check offers Prices are 320, 330 and 320.	All Offers are coherent with tester's request (Demonstration)	Cf Fig 3, 4 & 5	Passed	
3	Book the most expensive offer Tester touch the book button in the offer list for the offer with a cost of 330 €.	Pay screen opens	Fig 6	Passed	
4	Pay the offer Check the price in the pay screen is 330 €. The user touches the pay button.	User get a confirm payment message (Demonstration) Offer and Entitlement present in Wallet (not for the Core Release)s	Fig 7 & 8	Passed	

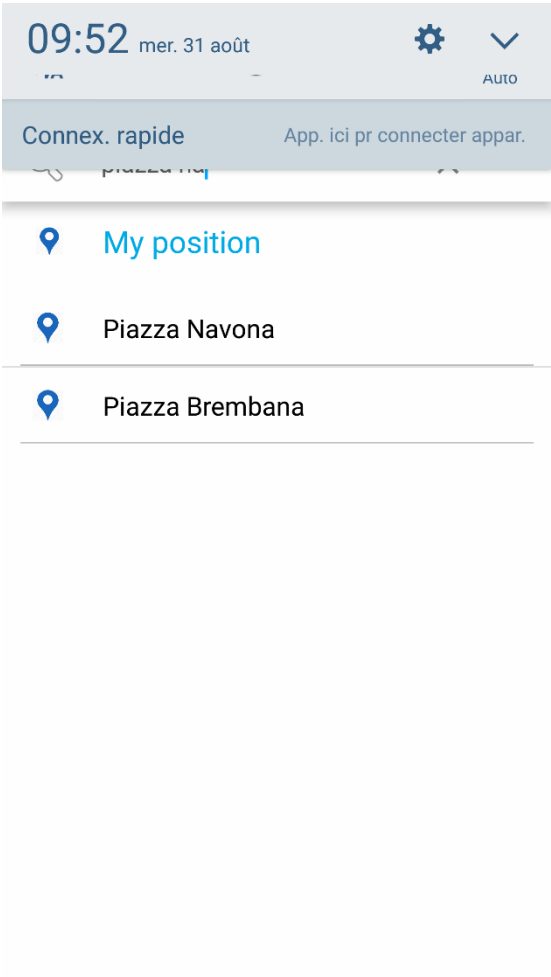
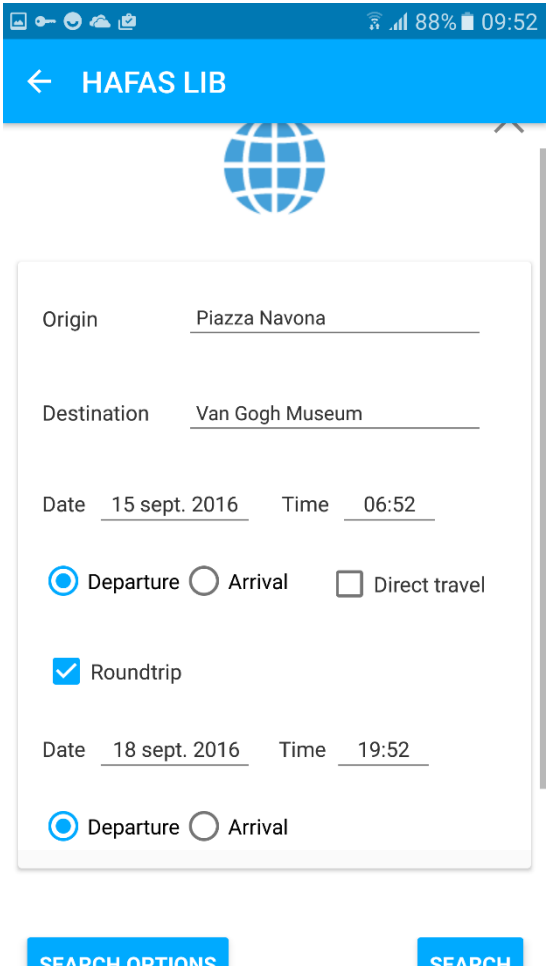
Fig 1 Origin or Destination Choice	Fig 2 Search Offer
	

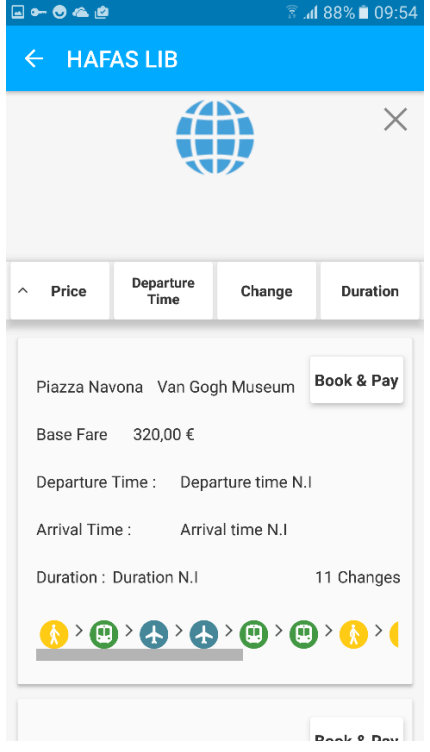
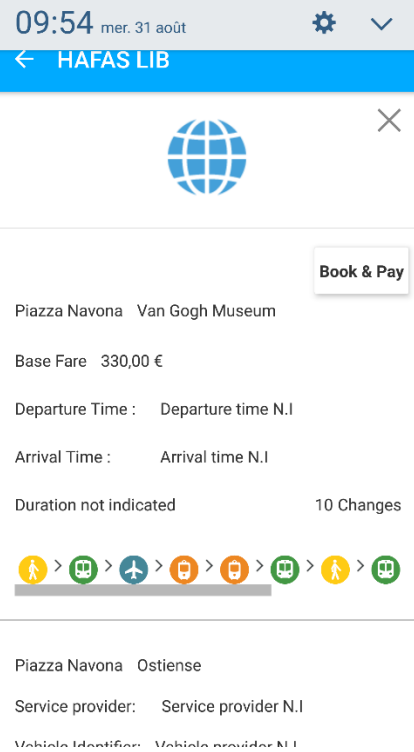
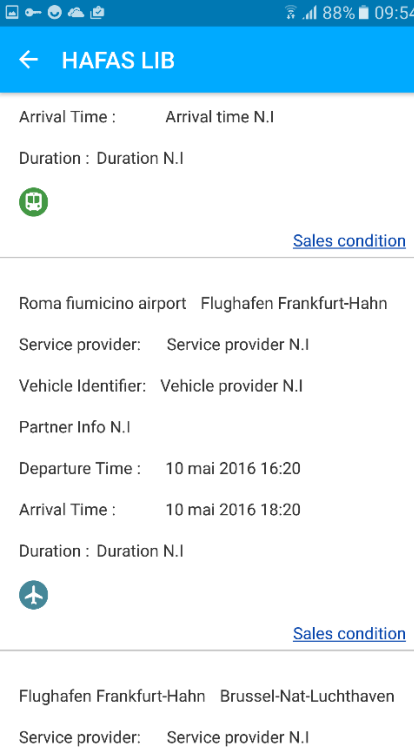




Fig 3 Offer List	Fig 4 Offer detail, part 1 (top of screen)	Fig 5 Offer detail, part 2 (travel episode detail)
 <p>Fig 3 is a screenshot of the 'HAFAS LIB' app showing a list of travel offers. The top bar is blue with a back arrow and the text 'HAFAS LIB'. Below it is a search bar with a globe icon and a close button. A filter bar contains tabs for 'Price', 'Departure Time', 'Change', and 'Duration'. The main content area shows a list item for 'Piazza Navona - Van Gogh Museum' with a 'Book & Pay' button. Below the item, details are listed: 'Base Fare 320,00 €', 'Departure Time: Departure time N.I', 'Arrival Time: Arrival time N.I', and 'Duration: Duration N.I' with '11 Changes'. At the bottom, a horizontal bar displays various transport mode icons (walking, train, plane, bus, etc.).</p>	 <p>Fig 4 shows the top portion of an offer detail screen. The status bar at the top shows the time '09:54' and date 'mer. 31 août'. The app header is 'HAFAS LIB'. Below the search bar, the route 'Piazza Navona - Van Gogh Museum' is displayed. A 'Book & Pay' button is visible. The fare details include 'Base Fare 330,00 €', 'Departure Time: Departure time N.I', 'Arrival Time: Arrival time N.I', and 'Duration not indicated' with '10 Changes'. A horizontal bar shows the transport mode sequence: walking, train, plane, bus, and train. The bottom section shows the route 'Piazza Navona - Ostiense' with 'Service provider: Service provider N.I' and 'Vehicle Identifier: Vehicle provider N.I'.</p>	 <p>Fig 5 displays the travel episode details. The header is 'HAFAS LIB'. The first episode shows 'Arrival Time: Arrival time N.I', 'Duration: Duration N.I', and a train icon, with a 'Sales condition' link. The second episode is for the route 'Roma fiumicino airport - Flughafen Frankfurt-Hahn', showing 'Service provider: Service provider N.I', 'Vehicle Identifier: Vehicle provider N.I', 'Partner Info N.I', 'Departure Time: 10 mai 2016 16:20', 'Arrival Time: 10 mai 2016 18:20', and 'Duration: Duration N.I', with another 'Sales condition' link. The third episode shows the route 'Flughafen Frankfurt-Hahn - Brussel-Nat-Luchthaven' with 'Service provider: Service provider N.I'.</p>

Fig 6 Booking Result



 HAFAS LIB



Book & Pay

ROME FIUMICINO PARIS CHARLES DE GAULLE

Price

Departure Time : 01 janv. 1970 01:00

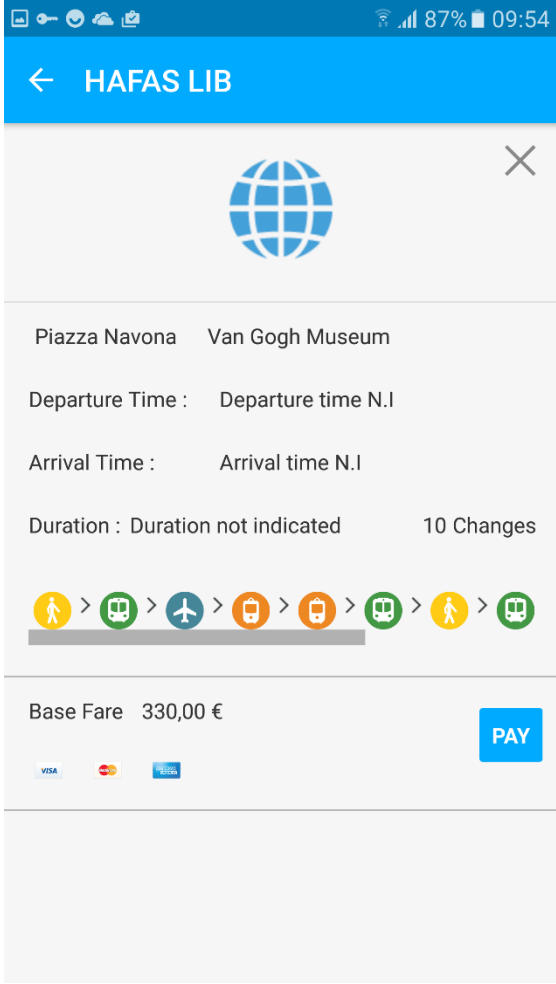
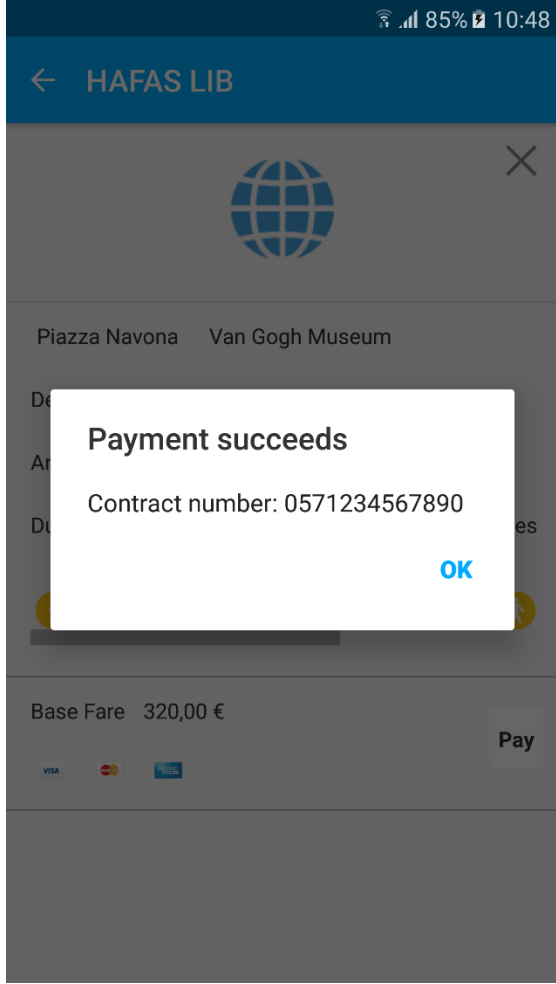
Arrival Time : 01 janv. 1970 01:00

Duration : Changes

ROME FIUMICINO 1 PARIS CHARLES DE GAULLE 2F

Service provider: AIR FRANCE

Vehicle Identifier: AIRBUS INDUSTRIE A321

Fig 7 Payment	Fig 8 Payment confirmation
	

4.1.4 Set and retrieve preference

WP5TestCase1-4	
Method Of Test	Demonstration and Inspection
Type of test	Manual
Objectives	Check that the user can set and retrieve preference
Description	The user will set the preferences, change screen, and come back to the preference screen. The preferences are not cached. They are reloaded from the server every time.
Status	OK
% passed	100 %

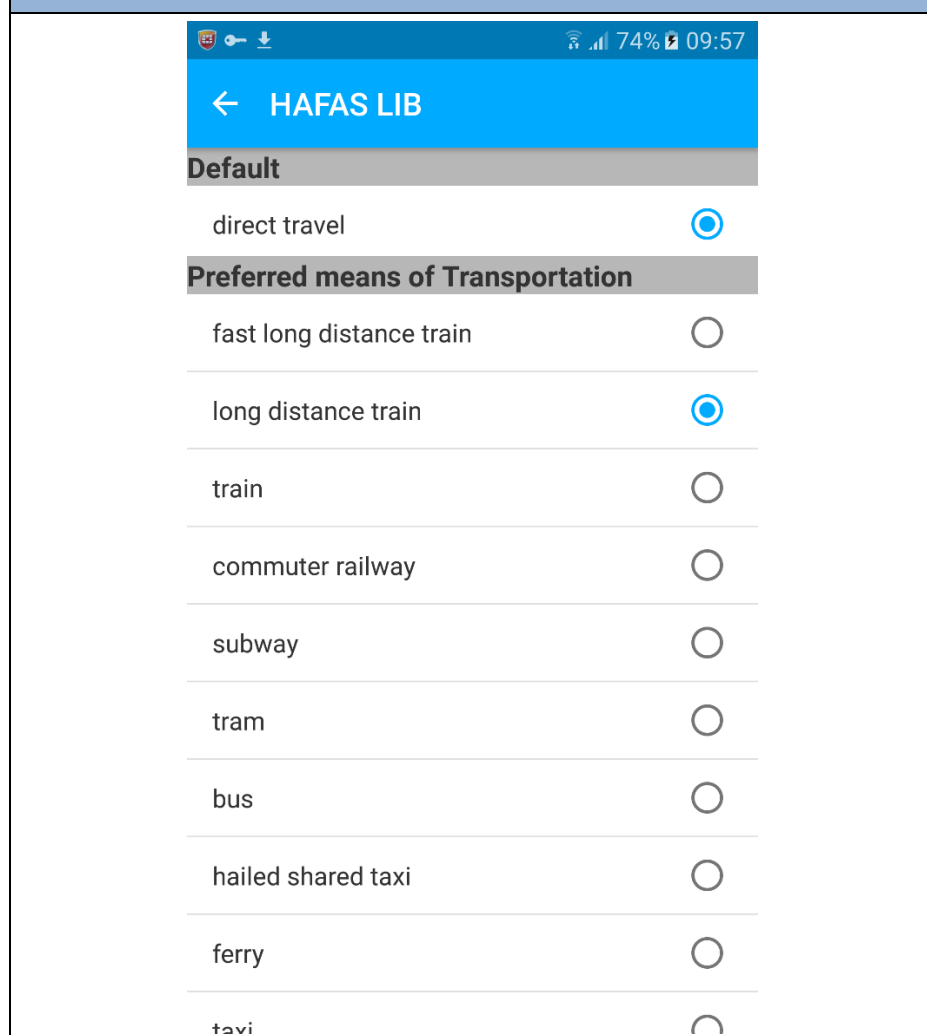
Normal client and server configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: User logged in, either after test WP5TestCase1-1 or WP5TestCase1-2 ; and tester must know this login credential				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Set Preferences Tester goes to the preference activity. Then he set some preferences : he check Long Distance Train	The preferences set by tester are in the Wallet (Inspection)	Cf Fig 1	Passed	
2	Change screen Tester goes to the search offer screen			Passed	
3	Check preferences Tester goes to the preference activity, and checks that Long distance train is still checked	Long Distance train still checked.	Cf Fig 2	Passed	

Id	Step description	Expected result	Observed result	State	Associated defect
5	<p>Check all preferences</p> <p>Tester reproduces step 1 to 4, with consecutively :</p> <ul style="list-style-type: none"> • Train • Commuter Train • Suburban Railway • Subway • Tram • Bus • Hailed Shared Taxi • Ferry • Taxi • Car • Car sharing • Carpooling • Bike • Bike sharing • Walk • Trenitalia • SNCF • Air France • Lufthansa • Cartafreccia • Flying Blue • Direct Travel • Each PRM type • Each PRM Parameters • Each class • Each seat 	<p>Selected preference still checked.</p>	<p>At each step, preference still checked</p>	<p>Passed</p>	

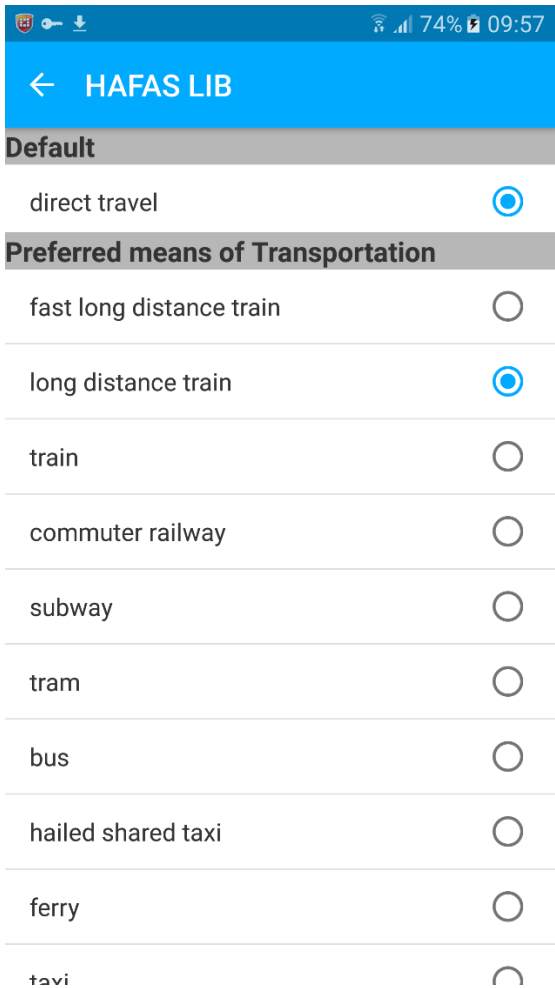
Fig 1 Preference screen with Long distance train checked



The screenshot shows a mobile application interface for HAFAS LIB. At the top, there is a status bar with a signal icon, 74% battery, and the time 09:57. Below this is a blue header bar with a back arrow and the text 'HAFAS LIB'. The main content area has a grey header 'Default' and a list of transportation options. The 'Preferred means of Transportation' section is highlighted with a grey background. The options are: 'direct travel' (selected with a blue circle), 'fast long distance train' (unselected), 'long distance train' (selected with a blue circle), 'train' (unselected), 'commuter railway' (unselected), 'subway' (unselected), 'tram' (unselected), 'bus' (unselected), 'hailed shared taxi' (unselected), 'ferry' (unselected), and 'taxi' (unselected).

Transportation Mode	Selected
direct travel	Yes
fast long distance train	No
long distance train	Yes
train	No
commuter railway	No
subway	No
tram	No
bus	No
hailed shared taxi	No
ferry	No
taxi	No

Fig 2 Preference screen with long distance train still selected



The screenshot shows a mobile application interface for HAFAS LIB. At the top, there is a status bar with a signal icon, a key icon, a download icon, a battery level of 74%, and a time of 09:57. Below the status bar is a blue header with a back arrow and the text "HAFAS LIB". Under the header is a grey bar with the word "Default". Below this is a list of transportation options, each with a radio button. The "long distance train" option is selected, indicated by a blue dot in the center of the radio button. The other options are "direct travel", "fast long distance train", "train", "commuter railway", "subway", "tram", "bus", "hailed shared taxi", "ferry", and "taxi".

Transportation Option	Selected
direct travel	<input type="radio"/>
fast long distance train	<input type="radio"/>
long distance train	<input checked="" type="radio"/>
train	<input type="radio"/>
commuter railway	<input type="radio"/>
subway	<input type="radio"/>
tram	<input type="radio"/>
bus	<input type="radio"/>
hailed shared taxi	<input type="radio"/>
ferry	<input type="radio"/>
taxi	<input type="radio"/>

4.1.5 Start tracking trip

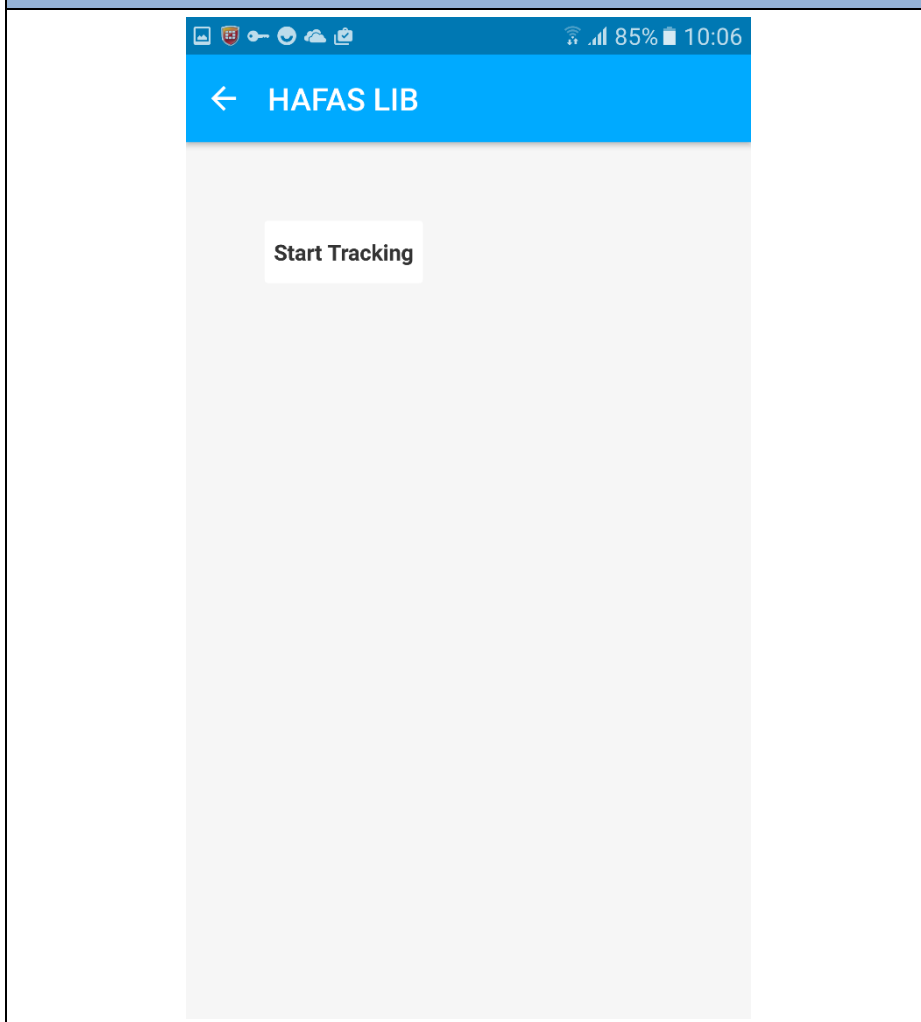
WP5TestCase1-5	
Method Of Test	Inspection
Type of test	Manual
Objectives	Test the availability of the trip tracker interface
Description	User will just click a button on a dedicated window. A log message will be written by the application
Status	OK
% passed	100 %

Standard configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Go to Start Tracking activity Touch Start Tracking button	A log message like this one : StartTracking Retrofit response success: retrofit2.Response@3d99d6d**200**StartTrackingResponse{activationResponseMessage='OK'} (Address may change) (Introspection)	Cf Fig 1	Passed	

Fig 1 start tracking test screen



4.2 TECHNICAL TESTS

4.2.1 Reinstall Application

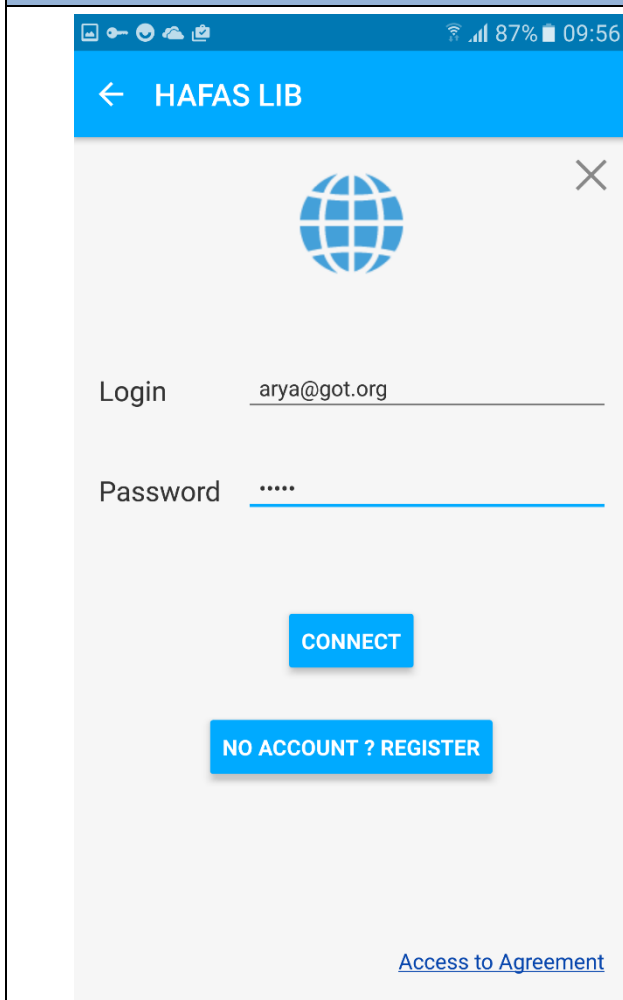
WP5TestCase2-1	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that the application will still work after being updated, and cached data are still available
Description	User will login, set preference, uninstall application, and then install the application with a greater version number
Status	OK
% passed	100 %

Standard configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: 2 version of the application, the older and the newer				

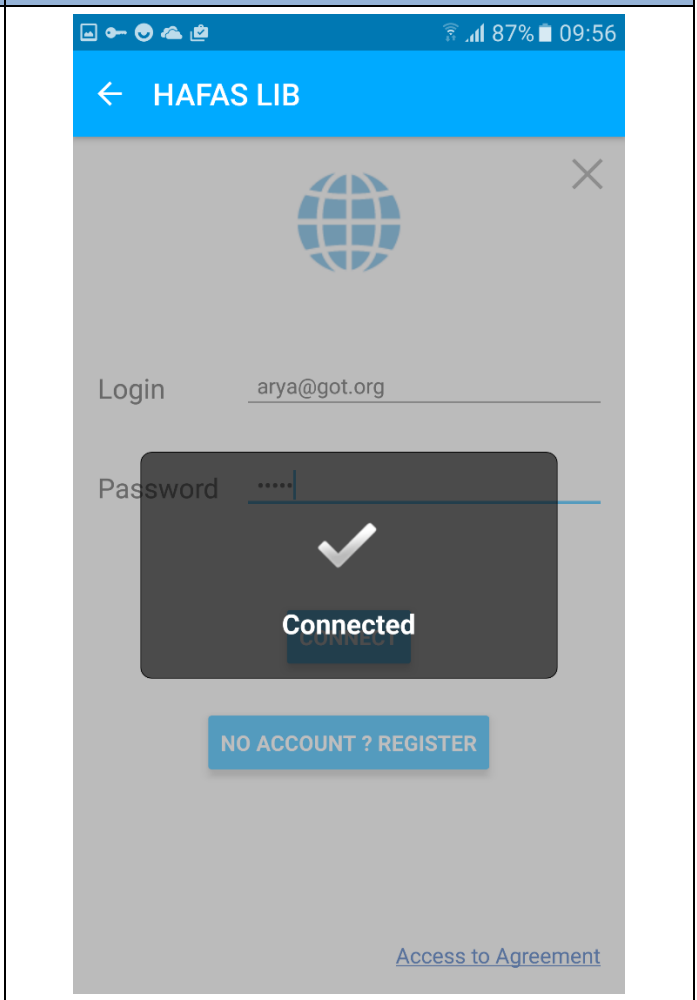
Id	Step description	Expected result	Observed result	State	Associated defect
1	Install older version Tester launches the application He log in with this credential : Login arya@got.org Password : stark He goes to the preference screen. He checks the following preferred transporter: - Trenitalia - SNCF	User logged, and preferences set	Cf fig 1, 2 & 3	Passed	
2	Uninstall the application The tester uninstalls the application	Application no more present in the device	Cf fig 4	Passed	
3	Install newer version of the application Tester will install the other, more recent version of the application.	Application present again in the device	Cf fig 5	Passed	
4	Login, and check Tester launches the application. Tester log into the application, with the credentials of step1. Tester goes to the preference screen. Tester checks that the preferred transport mode bus, tram and subway are checked.	Tester is able to launch the application and preferences are still here.	Cf fig 6	Passed	

Fig 1 login



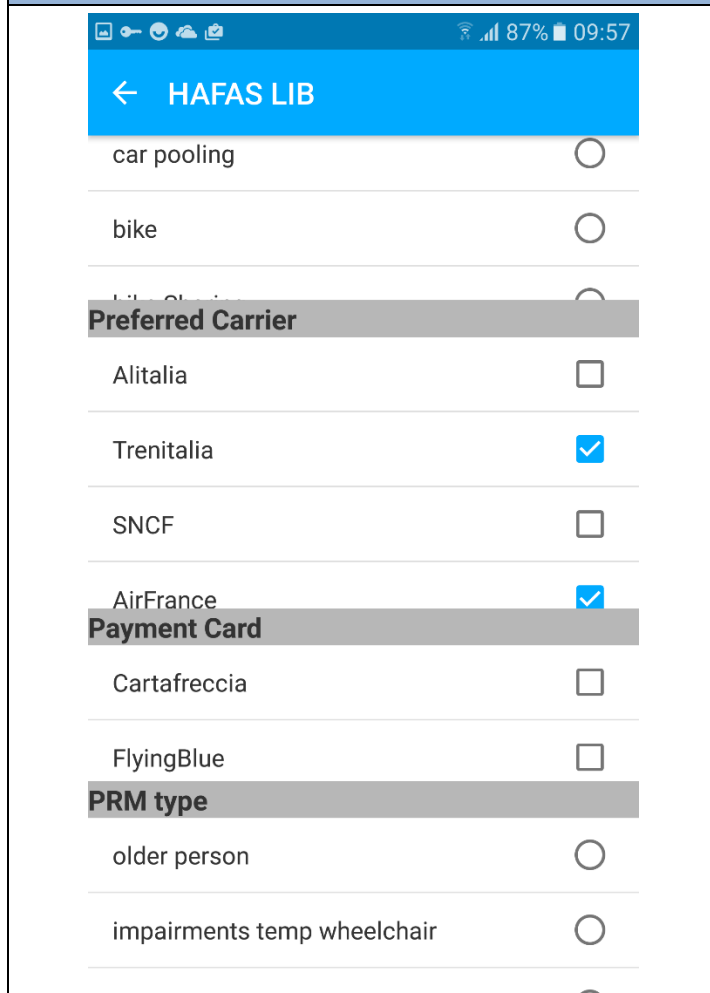
The screenshot shows the login interface of the HAFAS LIB mobile application. At the top, there is a blue header with a back arrow and the text "HAFAS LIB". Below the header is a grey area containing a globe icon and a close button (X). The main form has two input fields: "Login" with the email "arya@got.org" and "Password" with masked characters "....". Below the password field is a blue "CONNECT" button. Further down is a blue button labeled "NO ACCOUNT ? REGISTER". At the bottom right, there is a blue link that says "Access to Agreement". The status bar at the top shows a battery level of 87% and the time 09:56.

Fig 2 login answer



This screenshot shows the same login interface as Fig 1, but with a dark grey overlay indicating a successful login. The overlay features a white checkmark and the word "Connected" in white text. The "CONNECT" button is partially obscured by the overlay. The background elements, including the "HAFAS LIB" header, globe icon, input fields, "NO ACCOUNT ? REGISTER" button, and "Access to Agreement" link, are visible but dimmed. The status bar at the top remains the same, showing 87% battery and 09:56.

Fig 3 Preference screen with Trenitalia and SNCF selected



The screenshot shows a mobile application interface for 'HAFAS LIB'. At the top, there is a status bar with icons for connectivity and battery, showing 87% battery at 09:57. Below the status bar is a blue header with a back arrow and the text 'HAFAS LIB'. The main content area lists various preferences with radio buttons or checkboxes. The 'Preferred Carrier' section is highlighted with a grey bar, and 'Trenitalia' is selected with a blue checkmark. The 'Payment Card' section is also highlighted, and 'AirFrance' is selected. The 'PRM type' section is highlighted, and 'older person' is selected. The 'car pooling' and 'bike' options are also visible at the top.

Option	Selected
car pooling	<input type="radio"/>
bike	<input type="radio"/>
Preferred Carrier	
Alitalia	<input type="checkbox"/>
Trenitalia	<input checked="" type="checkbox"/>
SNCF	<input type="checkbox"/>
AirFrance	<input checked="" type="checkbox"/>
Payment Card	
Cartafreccia	<input type="checkbox"/>
FlyingBlue	<input type="checkbox"/>
PRM type	
older person	<input type="radio"/>
impairments temp wheelchair	<input type="radio"/>

Fig 4 uninstalling application

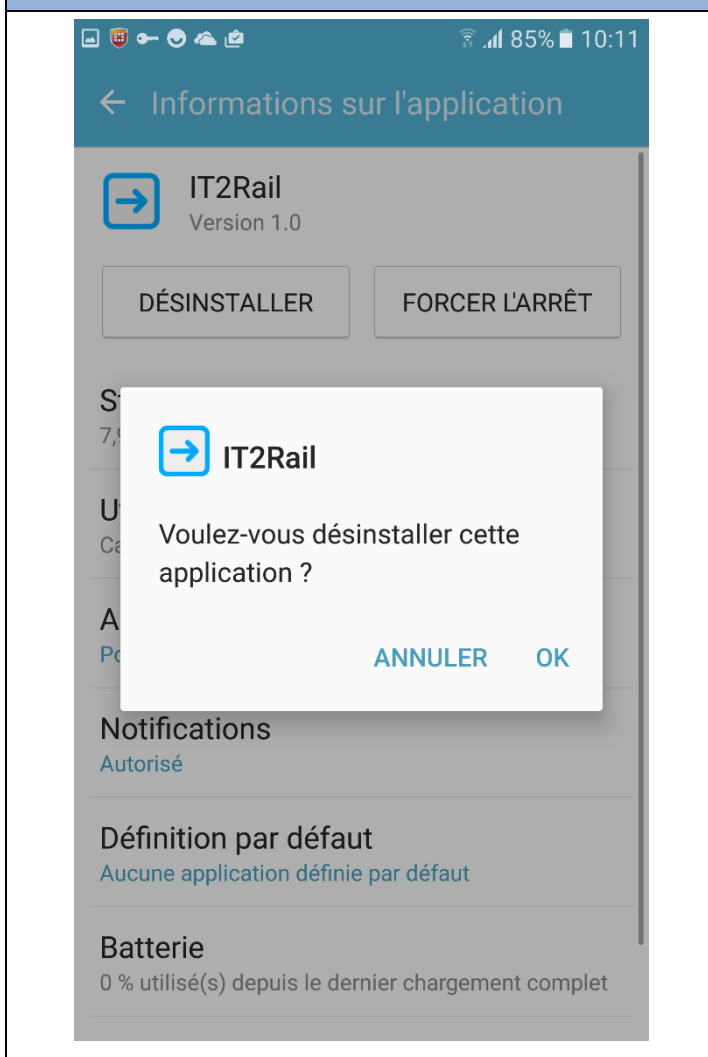


Fig 5 Application installed

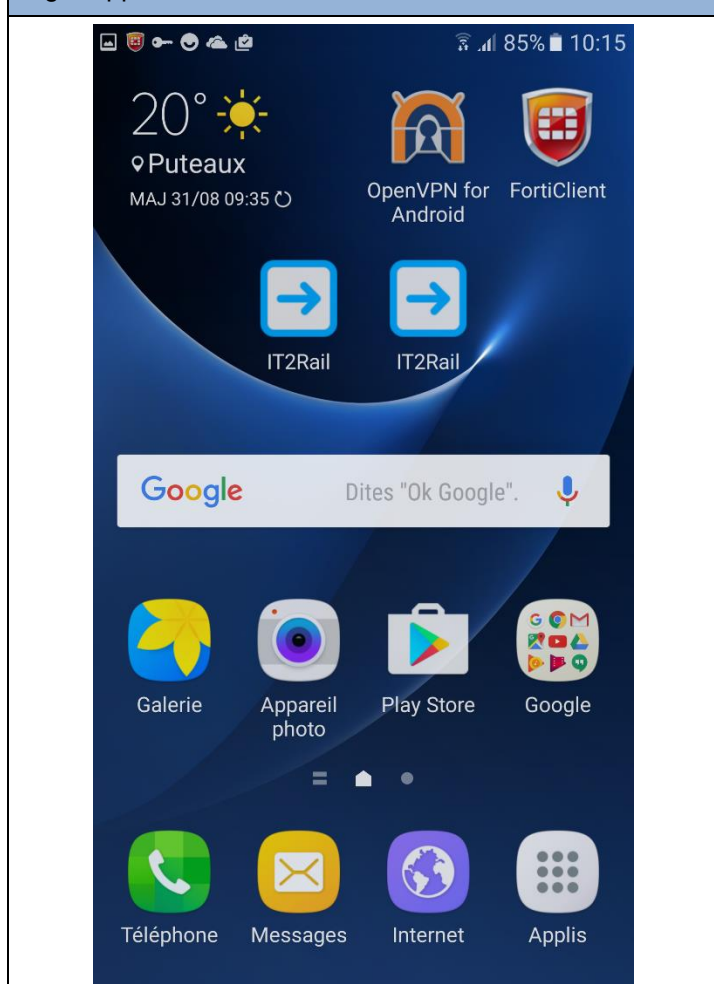


Fig 6 Preference screen with Trenitalia and SNCF still selected

← HAFAS LIB

car pooling ☐

bike ☐

Preferred Carrier

Alitalia ☐

Trenitalia ☒

SNCF ☐

AirFrance ☒

Payment Card

Cartafreccia ☐

FlyingBlue ☐

PRM type

older person ☐

impairments temp wheelchair ☐

4.3 ERROR TESTS

4.3.1 Existing email

WP5TestCase3-1	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check the user cannot create an account with an already existing email
Description	The tester will try to create an account with test credentials, to trigger the error in the simulator
Status	OK
% passed	100 %

Standard configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed on the device				

Id	Step description	Expected result	Observed result	State	Associated defect
1	<p>Account Creation :</p> <p>Tester launches the application, then goes to the create account screen.</p> <p>Tester creates an account with these credentials:</p> <p> Login: cersei@got.org</p> <p> Password: testpwd</p> <p> Password confirmation: testpwd</p> <p>Tester touches the register button.</p> <p>Notice : the login to choose must not have been selected for another test. Also tester must remember the credential use for step 3</p>	Error message	Cf fig 1 & 2	Passed	

Fig 1 account creation

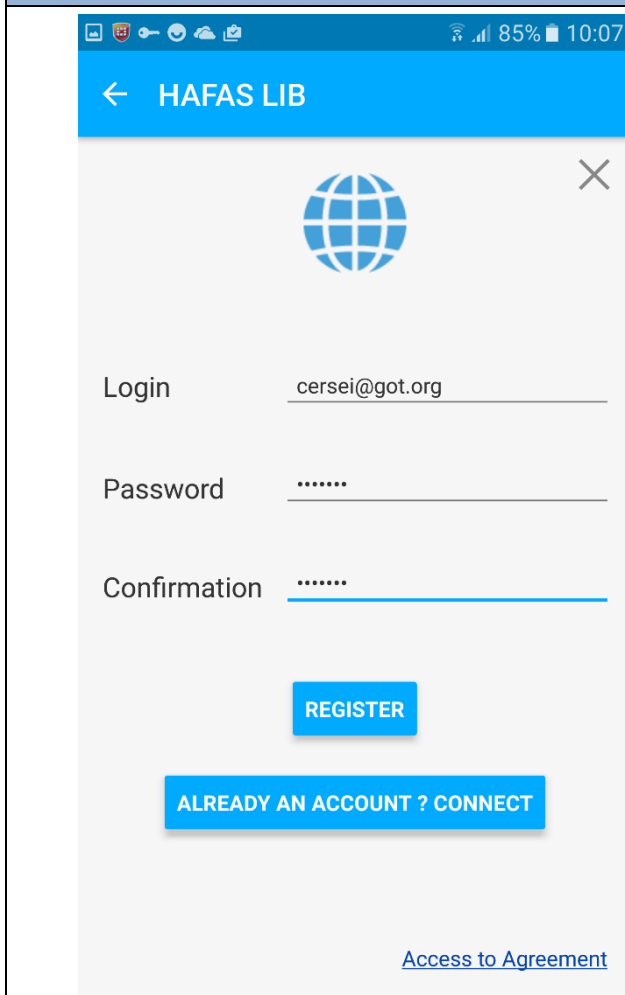
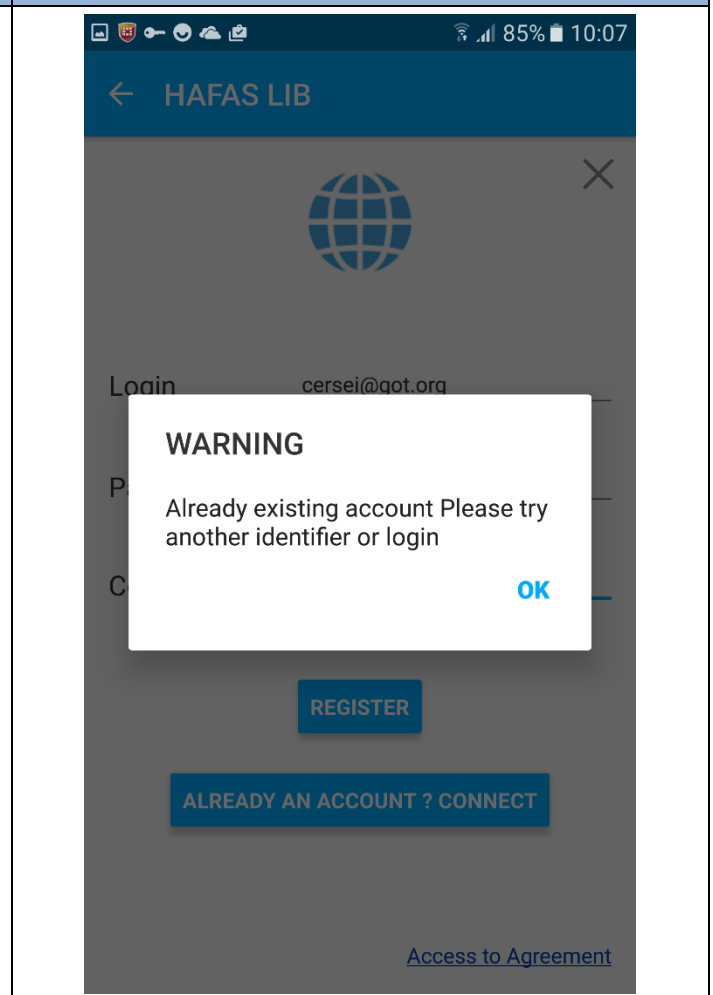


Fig 2 account creation error



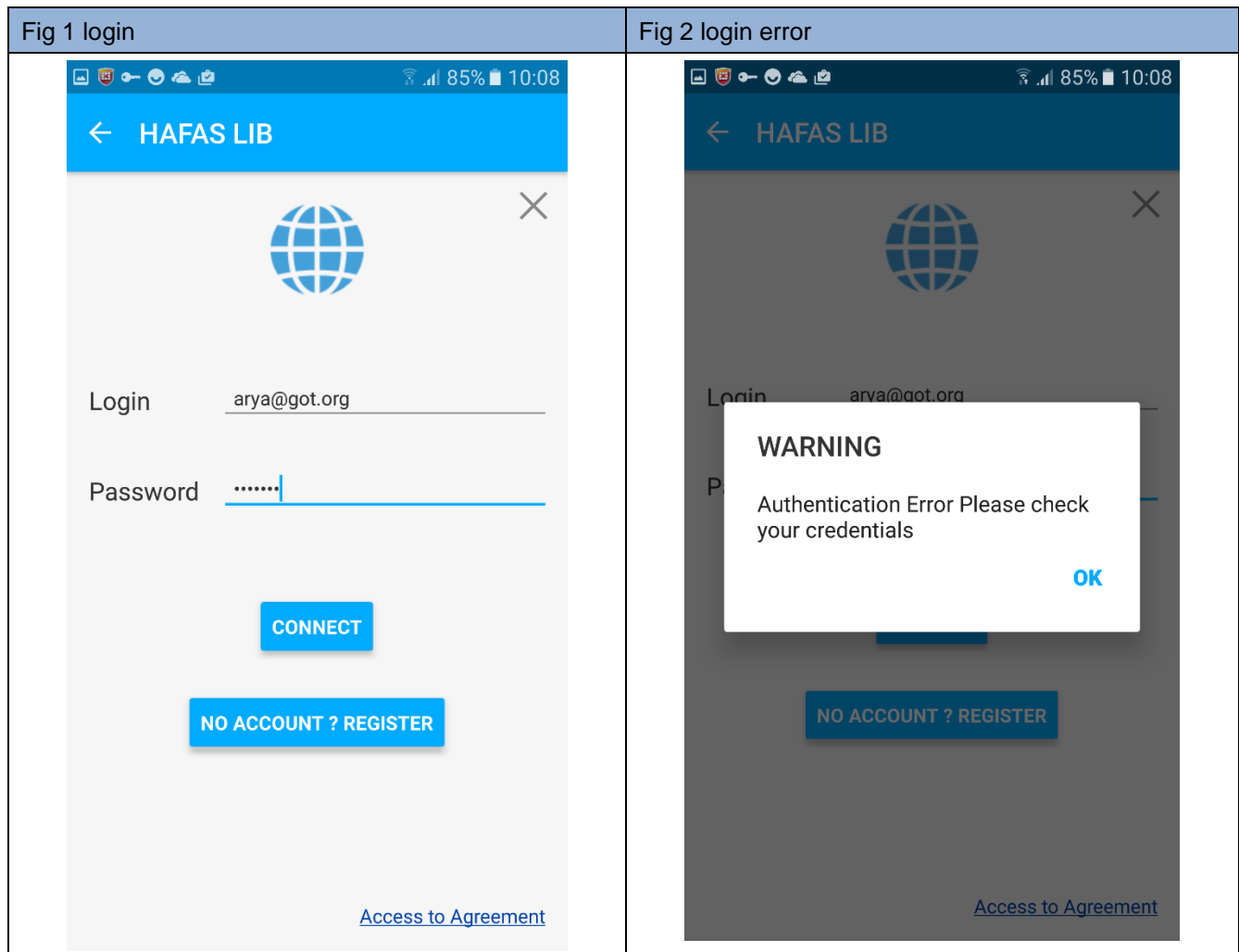
4.3.2 Wrong credentials

WP5TestCase3-2	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that a valid credential is necessary to connect
Description	Tester will try to login with invalid credentials
Status	OK
% passed	100 %

Standard configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed in the device, and cache emptied				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Tester will log in with a wrong credential: Tester launches the application. He goes to the login screen. He uses this credentials: Login: arya@got.org Password : WrongPassword	Error Message	Cf fig 1 & 2	Passed	



4.3.3 Not Corresponding password

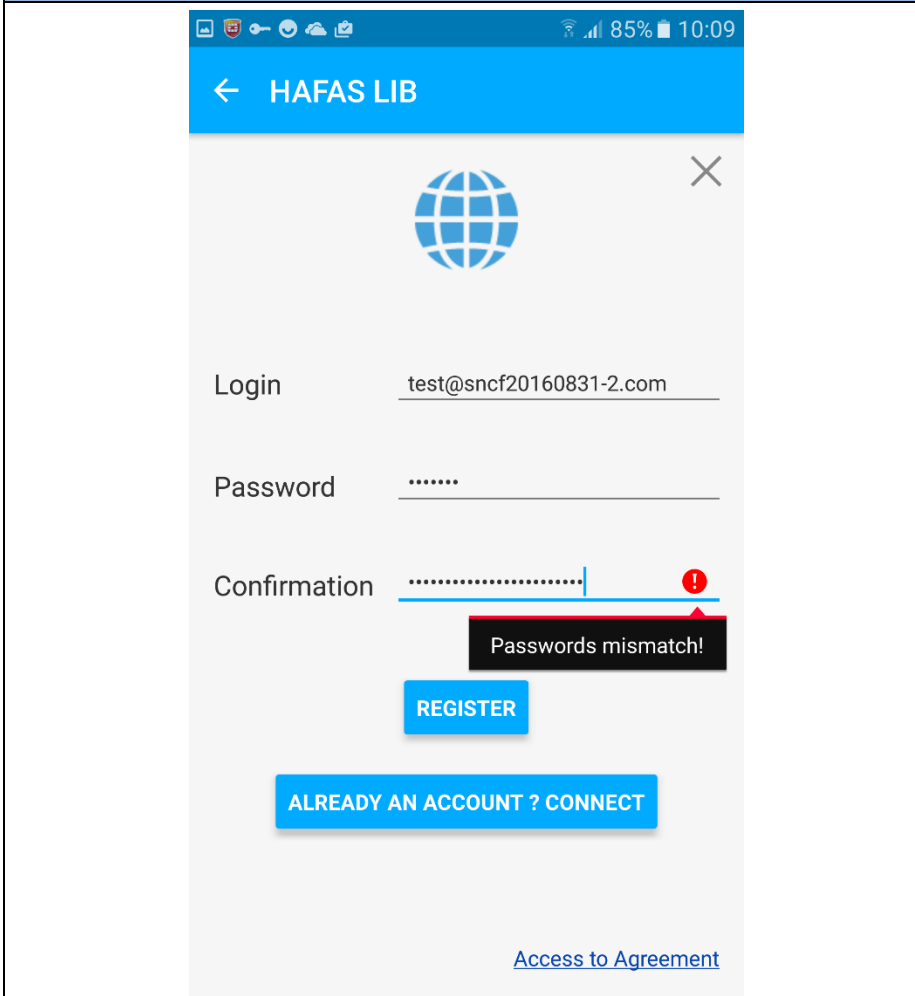
WP5TestCase3-2	
Method Of Test	Demonstration
Type of test	Manual
Objectives	Check that a correct password confirmation is necessary to create an account
Description	Tester will try to create an account with no corresponding password
Status	OK
% passed	100 %

Standard configuration	
Regression	NA
Test Case Tester	FChenebit & HChaabane

Id	Step description	Expected result	Observed result	State	Associated defect
	Preconditions: Application installed in the device, and cache emptied				

Id	Step description	Expected result	Observed result	State	Associated defect
1	Account Creation : Tester launches the application, then goes to the create account screen. Tester creates an account: he set testCrel@sncf20160830.fr in login field, testpwd in the password and testpawdtestpwd in password verification field, and touches the register button.	Error Message directly in the screen	Cf fig 1	Passed	

Fig 1 account creation error



The screenshot shows a mobile application interface for 'HAFAS LIB'. At the top, there is a blue header with a back arrow and the text 'HAFAS LIB'. Below the header is a light gray background with a globe icon and a close button (X). The form contains three input fields: 'Login' with the email 'test@sncf20160831-2.com', 'Password' with masked characters '.....', and 'Confirmation' with masked characters '.....'. A red exclamation mark icon is positioned to the right of the 'Confirmation' field. Below the 'Confirmation' field, a black error message box displays 'Passwords mismatch!'. Underneath the error message is a blue 'REGISTER' button. Further down is a blue button labeled 'ALREADY AN ACCOUNT ? CONNECT'. At the bottom of the screen, there is a blue link labeled 'Access to Agreement'.