



AuraTester User Guide

Automated testing tool for Oracle Forms based systems



Table of Contents

Introduction	3
Creating a Test Case	4
Creating an Test Group	10
Creating a new Test Group	10
Adding Test Cases to a Test Group	12
Running a Test Group	13
Test Monitoring	13
Creating a New Monitoring Agent	13
Activating/Pausing a Monitoring Agent	14
Monitoring Results.....	15
Recording Oracle Forms Simulations	16
Edit Parameters Dialog	20
View Scenario Dialog.....	23
Run Simulator Dialog	25
Debug Option.....	28
Using Junit with AuraPlayer Scripts	29



Introduction

AuraTester is an automated testing tool for Oracle Forms based systems. It allows users to record test scenarios using the user friendly toolbar then convert them to test scripts. Then advanced QA can be performed by reading in parameters from a batch file and comparing received results with expected values. Users can then run either one test case or a group of test cases one after the other. Each test case/group test cases has its own 'Report URL' that can be even be executed via external tools such as Continues Integration tools (Jenkins/Bamboo/etc.).

Autatester gives you the best of both worlds:

1. The ability to record customized test scripts to silently automate to run Oracle Forms business scenarios from any scripting language such as Ruby or JavaScript. These scripts can then be used from Junit in Eclipse or with any third party scripting tool. The functionality of this tool discovered in your player user guide
2. A user friendly functional testing tool to create automated Oracle Forms test scenarios & testing groups, and to monitor and analyze projected results. This is for users who do not want to manage their own scripting language or integrations.

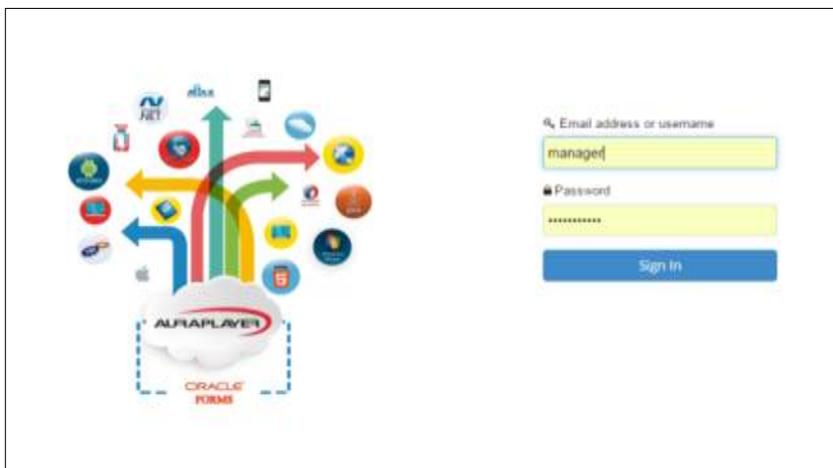
AuraTester's open and flexible architecture allows it to plug into any existing testing framework (such as Taza and JUnit), without additional Forms development or changes to the environment. It can also be used to successfully run multiple concurrent Oracle Forms sessions and measure system response time and server performance using any load testing software (such as WebLoad and JMeter).

* For manual testing, please see the 'Testing' section of the 'Service Manager User Guide'

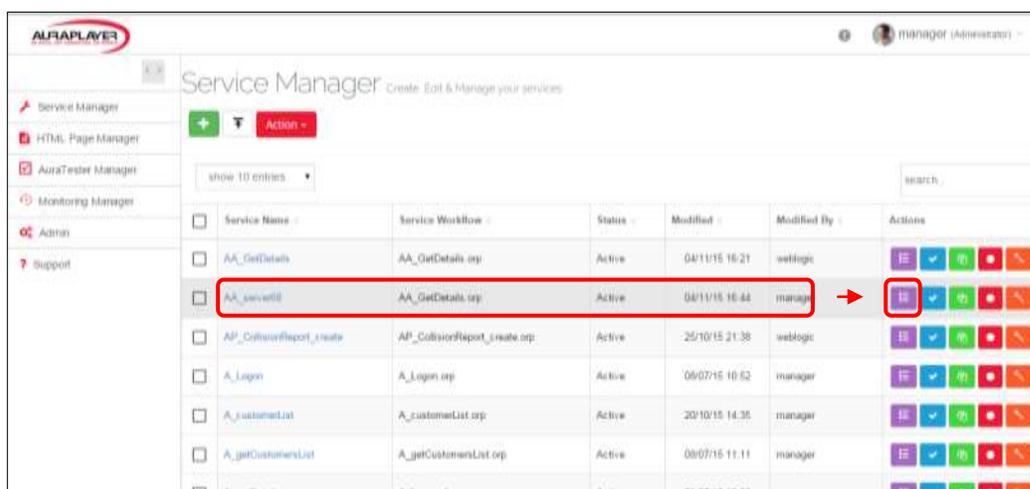
Creating a Test Case

A test case is a set of conditions under which AuraTester will determine whether the business process tested is working. The test case runs a web service with specific input parameters making sure all functions performed on Oracle Forms (Insert/Delete/Etc.) are done correctly by validating the web service responses and execution time.

1. Log into the 'Service Manager'



2. Choose the web service you wish to test and click on its 'Details' button



Service Name	Service Workflow	Status	Modified	Modified By	Actions
<input type="checkbox"/> AA_GetDetails	AA_GetDetails.asp	Active	04/11/15 16:21	weblogic	[E] [D] [U] [C] [R]
<input type="checkbox"/> AA_servicel	AA_GetDetails.asp	Active	04/11/15 16:44	manager	[E] [D] [U] [C] [R]
<input type="checkbox"/> AP_CollisionReport_create	AP_CollisionReport_create.asp	Active	25/10/15 21:38	weblogic	[E] [D] [U] [C] [R]
<input type="checkbox"/> A_Login	A_Login.asp	Active	09/07/15 10:52	manager	[E] [D] [U] [C] [R]
<input type="checkbox"/> A_customerList	A_customerList.asp	Active	20/10/15 14:35	manager	[E] [D] [U] [C] [R]
<input type="checkbox"/> A_getCustomersList	A_getCustomersList.asp	Active	08/07/15 11:11	manager	[E] [D] [U] [C] [R]
<input type="checkbox"/> A_getDetails	A_GetDetails.asp	Active	09/07/15 10:53	manager	[E] [D] [U] [C] [R]

Note: In order to create a test case/test group you must have existing web services. If you don't have any existing web services, please "create a new service" and refer to the 'Service Manager User Guide - Creating a Webservice'

3. In the 'Details' page, click 'Create Test Case' button at the top



4. In the 'Test Case Configuration' page set the following:

- a. **Batch Configuration** – Here you can choose a batch CSV file to enable you to run multiple occurrences of a test with different parameters for either input or to validate expected results.

Batch Configuration

Parameters file

Use first line as header

Number of iterations

Run until end of file

If you are not using a batch file for this test, please skip to section 'b'.

- i. Upload a CSV file – Click on 'Choose File' to upload the CSV file
- ii. Configure the batch test

Use first line as header:	Check the check box to define first line as header
Number of iterations:	Define how many lines from the CSV file you wish to run as test case
Run until end of file:	Instead of 'Number of iterations', this option will run the entire CSV file

Test Case Details

Test Case Details
➤

Test case name	TestA
Description	description
Response format	JSON ▼
Response level	ERROR ▼

Choose the 'Test Case Details':

Test Case Name:	Name your test case
Description:	Describe your test case (optional)
Response Format:	Choose the test response format (JSON/XML)
Response Level:	Choose the amount of detail you wish to see: Error – Shows the description of the Errors if they occur Debug – Shows input parameters that were sent to the web service and all the responses received.

- c. **Service Details** (Read Only) – These are the Name, Description and Form URL of the web service tested

Service Details
➤

Service Name	AA_GetDetails
Description	
Form URL	http://ec2-54-203-157-52.us-west-2.compute.amazonaws.com:9001/forms/ffmservlet?config=ComplexScenario

b. Input Parameters

In this section we define the input parameters needed to run the scripts.



Name	Label	Default Value	Parameterized Value
MAIL_USERNAME_0	Name	MA	choose parameterized value
MAIL_PASSWORD_0	Password	ORACLE	choose parameterized value
S_CUSTOMER_ID_0	Id	200	choose parameterized value

Set Input Parameters: You can input parameters either as static values or choose values as parameters from a CSV file.

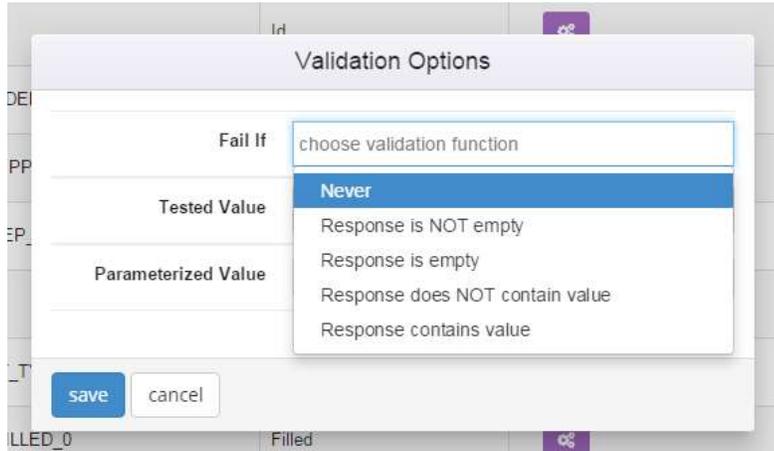
Default Value:	Choose the default input parameters you wish to test by editing them in the 'Default Value' field.
Parameterized Value: (* batch testing only)	From the drop down, choose the corresponding column from the CSV file.

c. **Output Parameters** – This section lists the output parameters that we will see as the response from our script. Here we can set validations to compare results received with expected results. Validations can be chosen from a set of predefined rules or chosen from values contained in the CSV file. To configure test validation, click on the 'Validations' button next to the field whose parameter you wish to test



Name	Label	Validations
S_CUSTOMER_NAME_0	Name Required	cc
S_CUSTOMER_PHONE_0	Phone	cc
S_CUSTOMER_ADDRESS_0	Address	cc
S_CUSTOMER_ID_0	Id	cc

Configure the validation - Choose the rule on which the validation will fail



Failure rules include:

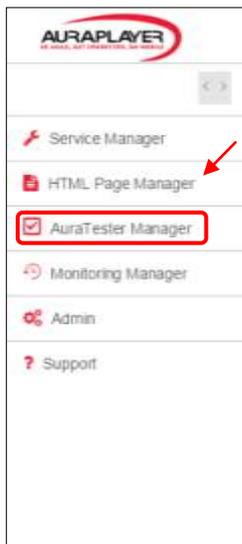
Fail If	Define when test will fail -	
	Never:	Test will never fail (ignore result)
	Response is NOT empty:	If any value is returned
	Response is empty:	If no value is returned
	Response does not contain value:	Fail if the field has a specific value in it. Either a value you set (in section ii below) or a value from a parameters list that you set in parameterized value (batch file).
	Response contains value:	Fail if the field is empty (does not have a record/value in it)
Tested Value:	Value to be validated against	
Parameterized Value: (* batch testing only)	From the drop down, choose the corresponding column from the CSV file.	

Save the 'Validations' configured

Creating an Test Group

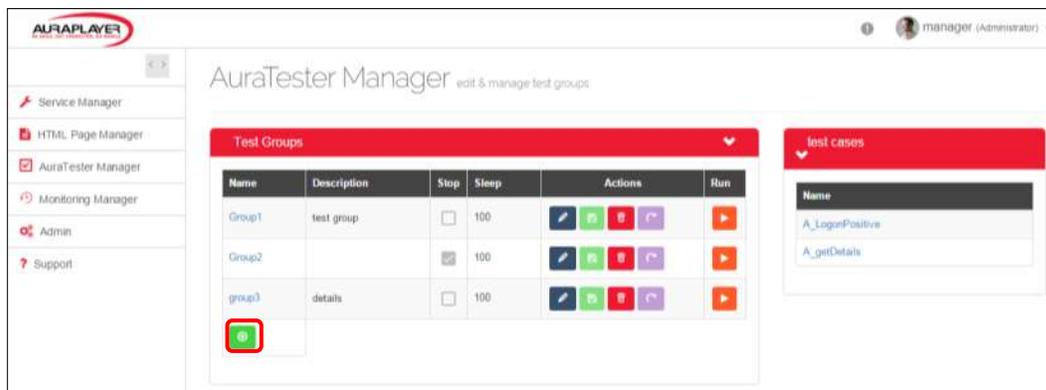
A 'Test Case Group' is a group of test cases bound together as one group. When the test group is run it runs the test cases automatically one after the other. These are viewed as a unit for testing more complex inter-dependent test cases.

1. Click the 'AuraTester Manager' tab on the left side of the 'Service Manager'



Creating a new Test Group

1. Click the green 'Add' button to create a new 'Test Group'



2. Fill in the 'Test Group' details

Name	Description	Stop	Sleep	Actions	Run
Group1	test group	<input type="checkbox"/>	100	   	
Group2		<input checked="" type="checkbox"/>	100	   	
group3	details	<input type="checkbox"/>	100	   	
NewGroup	group descript	<input type="checkbox"/>	100	   	

Name:	Name the new Test Group in the 'Name' column
Description:	Add a Test Group description in the 'Description' column (<i>*optional</i>)
Stop:	Check this box if you would like to stop the test scenario if any of the use cases fail. Meaning all test cases must succeed for the group to continue.
Sleep:	If you wish to pause the testing process in between each use case. Measured in milliseconds

The Action Menu buttons for Test Group :

Edit:		Edit the Test Group configuration
Save:		Save the Test Group configuration
Delete:		Delete the Test Group (once deleted, the test group can't be retrieved)
Cancel:		If in edit mode this will exit

Adding Test Cases to a Test Group

Once the test group has been create, we then add test cases that we would like to be included as part of the test group. These test cases must exist in the system prior to be added to a test group.

1. You can add test cases to a test group by clicking the green 'Add' button from the 'Test Cases' section



2. Reorder Test Cases – Use the reordering cursor to drag and drop the different use cased so they are in the order you wish them to be in so the test are run in the particular order.

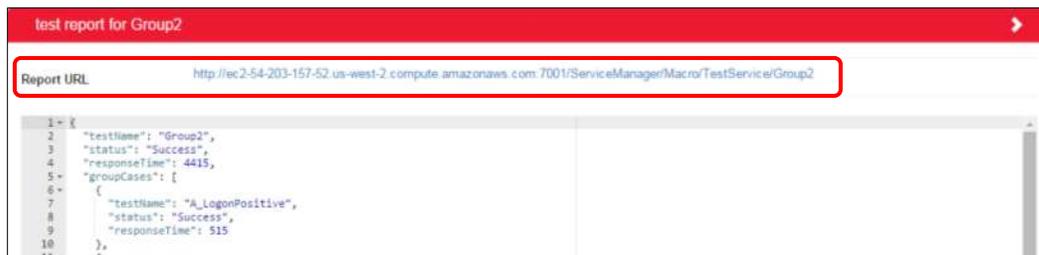


3. Delete Test Case – To remove a test case, click the 'Delete' button 
3. Save the Group Test Case by clicking the 'Save' button in the actions column



Running a Test Group

1. To run the Group Test click the 'Run' button; the test will run in the background and the page will refresh with the results 
2. A 'Report URL' will be presented on top of the results. This URL can be used to run the test from external tools; for example CI (Continues Integration) tools such as Jenkins and Bamboo

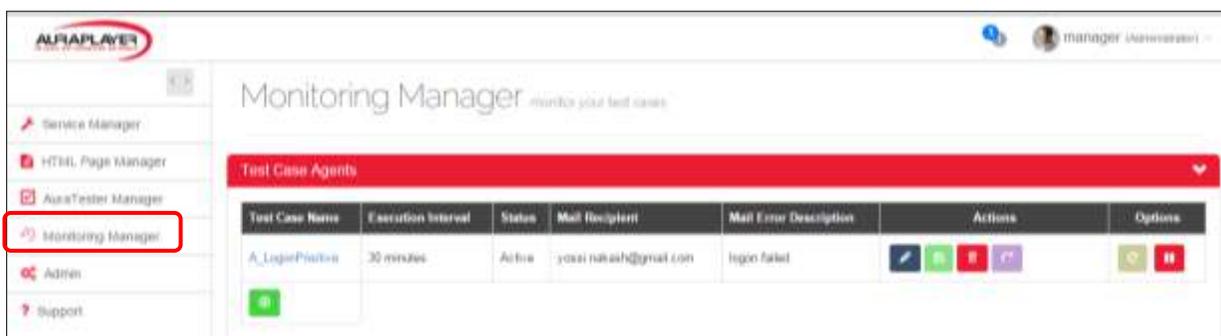


Test Monitoring

Test monitoring is available for checking the scenario performance and response time. Test cases created in the AuraTester manager can be monitored for time and performance from here.

Creating a New Monitoring Agent for a Test Case

1. Click the 'Monitoring Manager' tab



2. Click the green 'Add' button to create a new monitoring agent 

3. Configure the new agent

Test Case Agents					
Test Case Name	Execution Interval	Status	Mail Recipient	Mail Error Description	
A_LogonPositive	30 minutes	Active	yossi.nakash@gmail.com	logon failed	
AA_GetDetail	10 seconds	Paused	recipient	description	

Test Case Name	Choose the test case from the drop down list that you wish to monitor
Execution Interval	Determines how often the monitoring agent should run
Status	Shows if the agent is Active or Paused (this field is set after configuration and is read only)
Mail Recipient	If the test being monitored return a not successful result, , an email will be sent to this email recipient
Mail Error Description	This is the email 'subject' for the results

Functionality of Monitoring Agent 'Action' buttons:

Edit:		Edit the Agent's configuration
Save:		Save the Agent's configuration
Delete:		Delete the Monitoring Agent (once deleted, it can't be retrieved)
Cancel:		Cancel a creation of a new agent

4. Save the new agent by clicking the 'Save' button

Activating/Pausing a Monitoring Agent

A monitoring agent has two modes: Active and Paused.

Activate		Runs the agent in the intervals set and reports if there's an error
Pause		Pause agent. No monitoring will be done on the use case until re-activated.
Reset		Deletes the saved logs of the agent. Once reset button is pressed all old records are wiped clean.

Monitoring Results

Once an agent is set to 'Active' it will run as configured in the background. To see the results visually, click on the agent name.

Monitoring Manager monitor your test cases

Test Case Agents

Test Case Name	Execution Interval	Status	Mail Recipient	Mail Error Description	Actions	Options
A_LoginPositive	30 minutes	Active	yossi.nakash@gmail.com	login failed	[edit] [stop] [start] [refresh]	[clear] [stop]
AA_GetDetails	10 seconds	Active	yossi.nakash@gmail.com	Get Details Error	[edit] [stop] [start] [refresh]	[clear] [stop]

Monitor Chart - shows the time it takes for each run (this agent was set to run every 10 seconds)



Logs table – Show the response time and the status of run (if success or false) with the exact time stamp

AA_GetDetails logs (Active)

show 10 entries

search

Response Time	Status	Timestamp	Fail Reason
0.81	Success	01-12-2015 21:44:20	
1.05	Success	01-12-2015 21:44:11	
0.83	Success	01-12-2015 21:44:09	

10 20 50 all

Advanced Details – Showing average response time over all runs logged, minimum and maximum response time.

Advanced Details	
Average Response Time	0.85 seconds
Max Response Time	1.03 seconds
Min Response Time	0.78 seconds

Recording Oracle Forms Simulations

Start the AuraPlayer Toolbar: Start->All Programs->AuraPlayer->AuraPlayer

1. Click the “Record” button



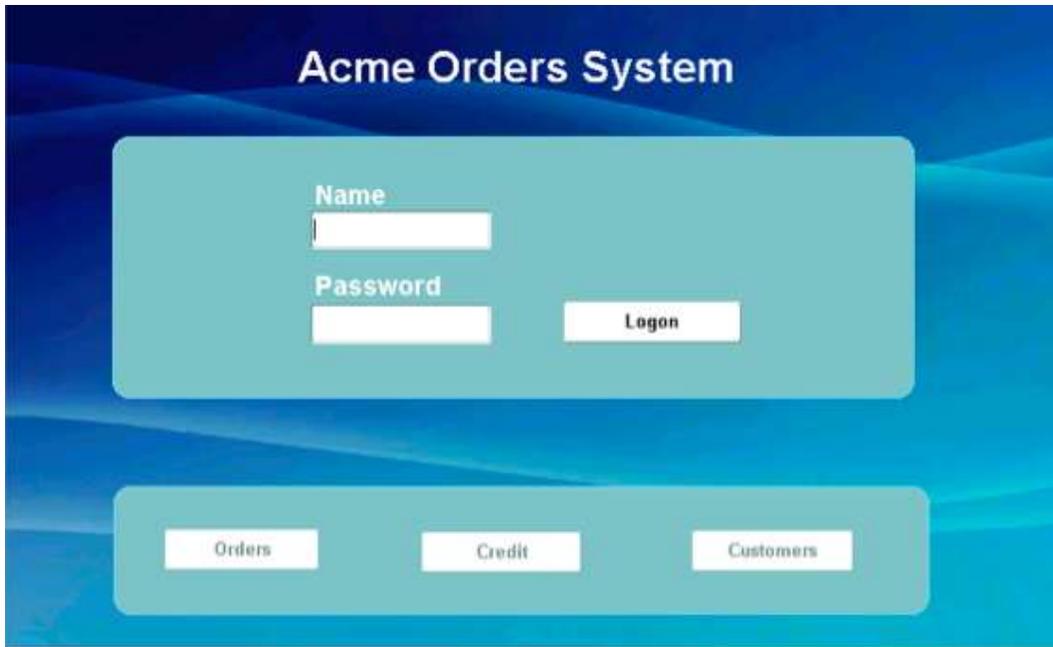
2. Check ‘Record as Web Service’ checkbox
3. Write in the URL TextBox the URL of your Form application (for example: http://oraplayervm/forms/frmservlet?config=bank_login)



4. Once you've clicked the "Record" button a new Record Tab will open, click the "Record" button on the "Recording Toolbar" to start the recording



5. IE will be opened automatically with the Form system running:



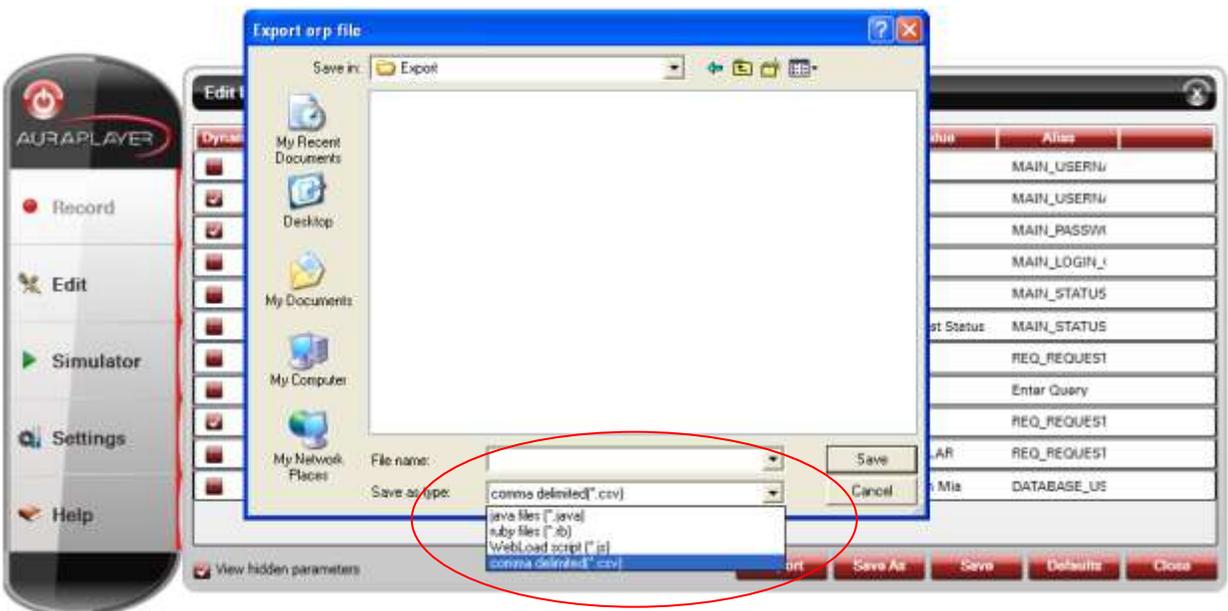
6. Perform the business process that you want to record: For example the login screen below

- a. We entered the User name: MIA
- b. We enter the Password: ORACLE
- c. We then press the Login Button

7. Click on the “Stop” button to **end the recording** then click on the “Edit” button to see the scenario and to edit the input and output parameters



8. Once you click “Edit” the Edit Parameters dialog will open. To complete the simulation recording click the “Save” button to choose a directory where you will save your scenario file. This file should be accessible to the AuraPlayer Service Manager if you need to create Webservices from these scenarios.



Edit Parameters Dialog

Can be accessed from:

Recording toolbar

Edit->Edit Parameters.

Simulator->Edit Parameters.

Click the "Edit" button and the "Edit Tab" will open and allow you to edit pre-existing scenarios

You can view scenarios, edit parameters or export the scenario as a file. In addition if you have AuraPlayer ServiceManager you can launch it from here using the "Deploy to Webservice"



button

Acme Customer List

Id	Name	Phone	Address	City	State	Country	Zip	Rating	Comments
201	Unisports	55-2066101	72 Via Bahia	Sao Paulo		Brazil		EXCELLE	Customer usually order
202	Simms Athletics	81-2010199	6741 Takashi Blvd	Osaka		Japan		POOR	Customer should always
203	Delhi Sports	91-10351	11368 Chanakya	New Delhi		India		GOOD	Customer specializes in
204	Womansport	1-206-104-0103	281 King Street	Seattle	Washington	USA	98101	EXCELLE	
205	Kam's Sporting Go	852-3692888	15 Henessey Road	Hong Kong				EXCELLE	
206	Sportique	33-2257201	172 Rue de Rivoli	Cannes		France		EXCELLE	Customer specializes in
207	Sweet Rock Sports	234-6036201	6 Saint Antoine	Lagos		Nigeria		GOOD	
208	Muench Sports	49-527454	435 Gruenstrasse	Stuttgart		Germany		GOOD	Customer usually pays t
209	Beisbol Sil	809-352689	792 Playa Del Mar	San Pedro de Mac		Dominican Repu		EXCELLE	
210	Futbol Sonora	52-404562	3 Via Saguaro	Nogales		Mexico		EXCELLE	Customer is difficult to r

Acme Customer List

Id	Name	Phone	Address	City	State	Country	Zip	Rating	Comments
201	Unisports	55-2066101	72 Via						
202	Simms Athletics	81-2010199	6741						
203	Delhi Sports	91-10351	1136						
204	Womansport	1-206-104-0103	281						
205	Kam's Sporting Go	852-3692888	15 H						
206	Sportique	33-2257201	172						
207	Sweet Rock Sports	234-6036201	6 Sai						
208	Muench Sports	49-527454	435 C						
209	Beisbol Sil	809-352689	792						
210	Futbol Sonora	52-404562	3 Via						

Dynamic	Window Title	Action	Field Name	Label	Recorded Value	Value	Alias
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Set Focus	MAIN_USERNA	Name		null	MAIN_USERNA
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Set Value	MAIN_USERNA	Name	mia		MAIN_USERNA
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Set Value	MAIN_PASSWOF	Password	oracle		MAIN_PASSWC
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Button Pressed	MAIN_LOGIN_0	Logon		Logon	MAIN_LOGIN_C
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Set Focus	MAIN_OPEN_0	Customers		null	MAIN_OPEN_0
<input checked="" type="checkbox"/>	COMPLEX_LOGI	Button Pressed	MAIN_OPEN_0	Customers		Customers	MAIN_OPEN_0
<input checked="" type="checkbox"/>	C	Set Focus	S_CUSTOMER_ID	Id		201	S_CUSTOMER
<input checked="" type="checkbox"/>	C	Event Menu	Enter Query	Enter Query			Enter Query
<input checked="" type="checkbox"/>	C	Set Focus	S_CUSTOMER_S	State		null	S_CUSTOMER
<input checked="" type="checkbox"/>	C	Set Value	S_CUSTOMER_S	State	CA		S_CUSTOMER
<input checked="" type="checkbox"/>	C	Set Focus	S_CUSTOMER_ID	Id		43228	S_CUSTOMER
<input checked="" type="checkbox"/>	C	Set Focus	S_CUSTOMER_N	Name		AAA Electronic	S_CUSTOMER
<input checked="" type="checkbox"/>	C	Set Focus	S_CUSTOMER_P	Phone		518-891-3637	S_CUSTOMER

View hidden parameters

Columns

Dynamic

Window Title

The Title of the Form in which the Action was performed.

Action

The action the user performed while recording.

For example: Button press, event menu, focus, set value and so on.

- Parameters that their action is **SetValue** and are set as dynamic, can be set during playback with dynamic values. And therefore play the scenario with different set values.
- Parameters that their action is **SetFocus**, can set as output parameters. After playback their updated value would be returned

Field Name

The name of the Field that the action was performed on, as it appears in the Form development environment (FMB file).

Label

The name of the Field that the action was performed on, as it appears on the Form – The label that the user sees when he performs actions on the Form.

Recorded Value

There is recorded value only on Parameters that their action is **SetValue**. This is the value that the user set in the Forms while he was recording the scenario with AuraPlayer.

Value

- In Parameters that their action is **SetValue**, the value is the default value to be played back. By default it is empty, but it can be set to be any value.
- In Parameters that their action is **SetFocus**, it is the value of the Field that user focused on.

Alias - The name of the Field to be set in input parameters. The default is the **Field Name**.



Dynamic	Window Title	Action	Field Name	Label	Recorded Value	Value	Alias
<input type="checkbox"/>	MAIN	Set Focus	MAIN_USERNAM	Name		null	MAIN_USERNAM
<input checked="" type="checkbox"/>	MAIN	Set Value	MAIN_USERNAM	Name	MIA		MAIN_USERNAM
<input checked="" type="checkbox"/>	MAIN	Set Value	MAIN_PASSWOF	Password	ORACLE		MAIN_PASSWOF
<input type="checkbox"/>	MAIN	Set Focus	MAIN_STATUS_0	Status		null	MAIN_STATUS_0
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS	Request ID		745	REQ_REQUESTS
<input checked="" type="checkbox"/>	Status Window	Set Value	REQ_REQUESTS	Request ID	732		REQ_REQUESTS
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS	Name		Rimon Elman	REQ_REQUESTS
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS	Priority		REGULAR	REQ_REQUESTS
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS	Status		COMPLETED	REQ_REQUESTS
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS	Telephone		3-6895598	REQ_REQUESTS
<input type="checkbox"/>	Status Window	Set Focus	CTL_SHOW_DET	Show Detail		null	CTL_SHOW_DET
<input type="checkbox"/>	העברת אובייקטים	Set Focus	DATABASE_USEI	Requestor		Urman Mia	DATABASE_USEI

View Hidden Parameters CheckBox

Filter the Parameters according to Action.

When it is unchecked, it shows only Parameters with actions **SetValue**, **SetFocus**.

When it's checked, it shows all the actions that were performed during recording (including **ButtonPressed**, **EventMenu**, etc.)

Dynamic	Window Title	Action	Field Name	Label	Recorded Value	Value	Alias
<input type="checkbox"/>	MAIN	Set Focus	MAIN_USERNAME_0	Name		null	MAIN_USERN/
<input checked="" type="checkbox"/>	MAIN	Set Value	MAIN_USERNAME_0	Name	MIA	MIA	MAIN_USERN/
<input checked="" type="checkbox"/>	MAIN	Set Value	MAIN_PASSWORD_0	Password	ORACLE	ORACLE	MAIN_PASSW/
<input type="checkbox"/>	MAIN	Set Focus	MAIN_STATUS_0	Status		null	MAIN_STATUS
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS_REQ_ID	Request ID		745	REQ_REQUEST
<input checked="" type="checkbox"/>	Status Window	Set Value	REQ_REQUESTS_REQ_ID	Request ID	732	732	REQ_REQUEST
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS_REQ_U	Name		Rimon Elman	REQ_REQUEST
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS_REQ_P	Priority		REGULAR	REQ_REQUEST
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS_REQ_S	Status		COMPLETED v	REQ_REQUEST
<input type="checkbox"/>	Status Window	Set Focus	REQ_REQUESTS_USER_	Telephone		3-6895598	REQ_REQUEST
<input type="checkbox"/>	Status Window	Set Focus	CTL_SHOW_DETAILS_0	Show Detail		null	CTL_SHOW_DE
<input type="checkbox"/>	העברת אובייקטים	Set Focus	DATABASE_USER_NAM	Requestor		Urman Mia	DATABASE US

View hidden parameters Print Save As Save **Defaults** Close

Defaults Button

Set a default value in all the parameters that their action is **SetValue**. The value that is set is the recorded value.

You might want to use this option, if during playback most of the dynamic values should be the same as in the recording.

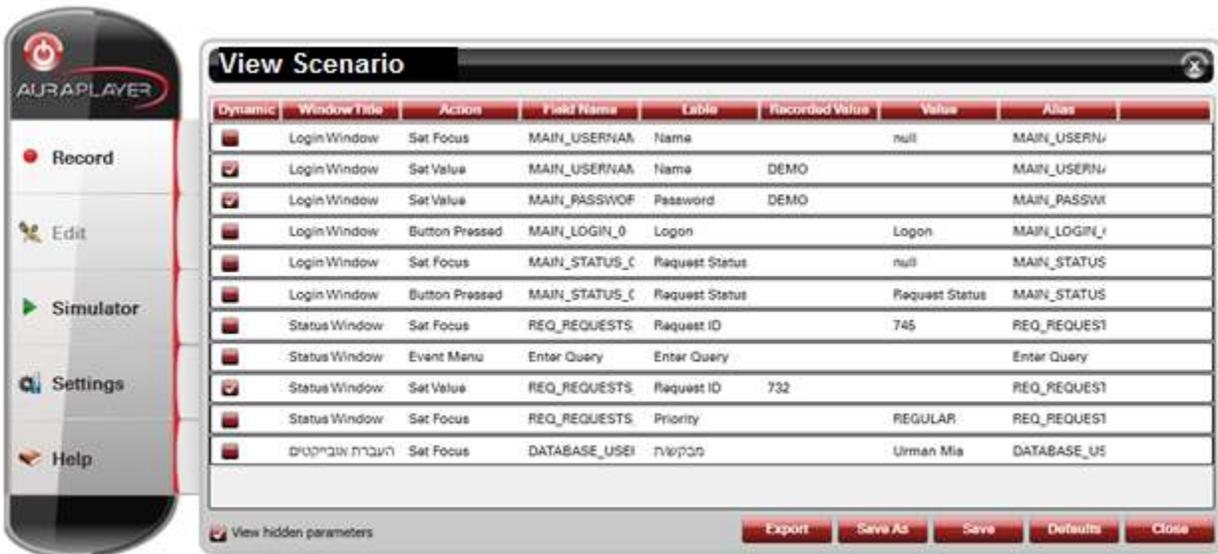
View Scenario Dialog

Can be accessed from:

Edit->View Scenario. Simulator-> View Scenario.

A summary of all the actions that were performed in the recorded scenario, with all the values in all the fields. It is a very convenient way to see the business process that was performed in the Form.

This dialog is Read-only. Click the “View Scenario” button



Run Simulator Dialog

Can be accessed from:

Simulator-> Run

Used for GUI playback only

Plays the recorded scenario in GUI mode.

Should be used on orp files that were recorded in GUI mode (not as Web Service)

1. Click the "Simulator" button to open the "Simulator Tab" to test existing scenarios

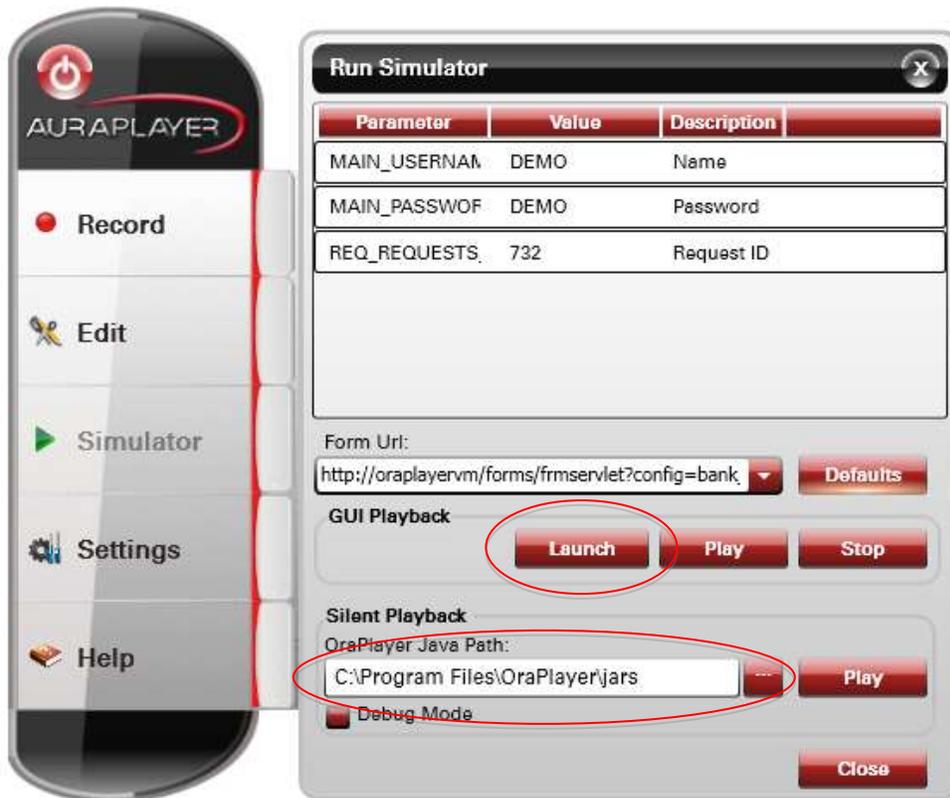
First choose the scenario file you wish to simulate (.orp file) then click on the "Run" button to run the simulation.



2. To begin the simulation, enter the input parameters and the Oracle Forms server URL where you wish to playback the scenario. Please ensure the Oracle Forms server has the same FMX files as the server where the initial scenario was recorded.



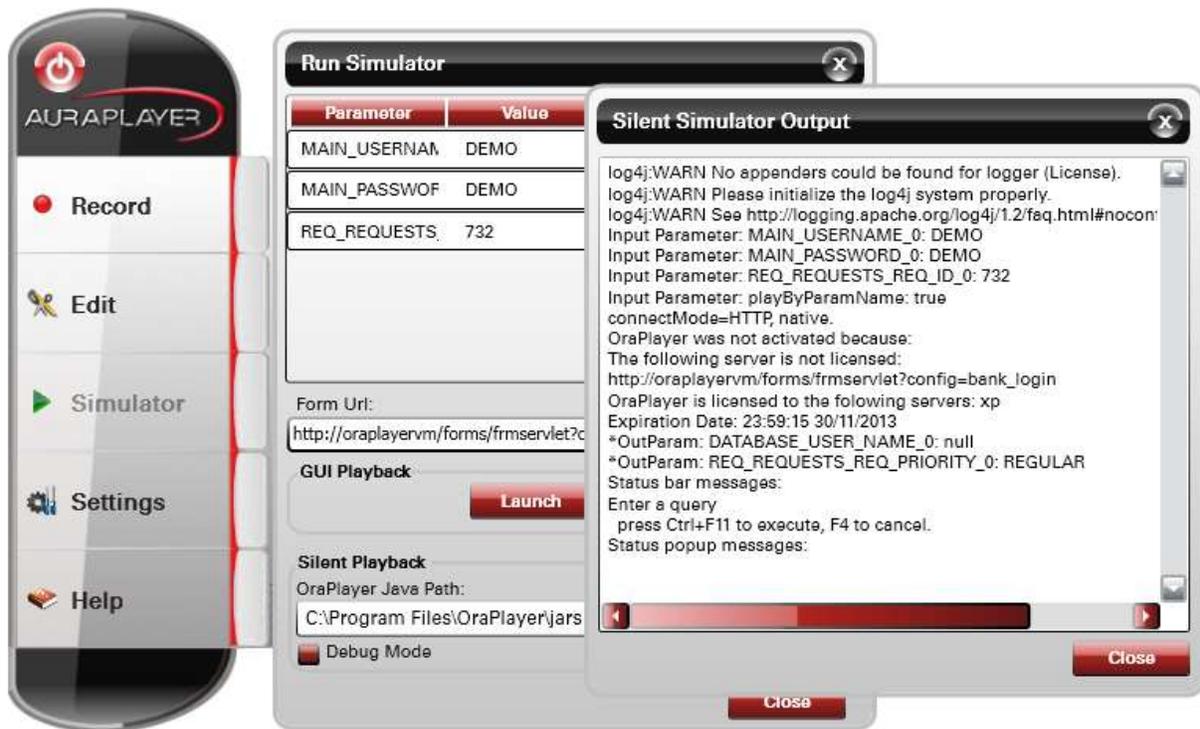
3. Then choose your method of playback. It can be GUI Playback where you will see the forms screen automation. This will require the forms system to be open prior to simulation therefore you must click the "Launch" button and only once the form system loads click the play button to run the simulation.



For silent playback, you must select the directory where the AuraPlayer Jar files are found then click "Play." Additionally you can run the simulation in debug mode to get verbose debugging messages throughout the playback simulation.

Debug Option

If the debug option is chosen you will see the Silent Simulator Output Window. Here you can see all aspects of the running scenario including output parameters received and messages the Forms returned.



Using Junit with AuraPlayer Scripts

In order to create a Java file for Junit testing -

1. Click the "Edit" button and choose the ORP file you wish to create the JavaScript for



2. Once the ORP file is chosen, click the "Export" button and save it as a Java file (Note: file can be saved in different formats such as CSV & Ruby)
3. To see how to run this script in Junit in our training video [Running AuraPlayer Scripts from Junit](#)