# *Deliverable Number D44.3*
# *Factsheet Number 01*
# *Mobile Development Module*

| | |
| --- | --- |
| Document Owner: | Giovanni Casella – Softeco |
| Contributors: | Paolo Giovene, Enrico Morten, Simona Stringa - Softeco |
| Dissemination: | Confidential |
| Contributing to: | WP44 |
| Date: | 29/03/2013 |
| Revision: | 1.0 |

## Table of Contents

**1 AVAILABILITY AND CONTACTS 3**

**2 ARCHITECTURE AND FUNCTIONALITIES 4**

**3 TECHNICAL INFORMATION 5**

3.1 REQUIREMENTS ..... 5
3.2 TECHNICAL DETAILS ..... 5

**4 LICENSING 6**

4.1 LICENSE ..... 6
4.2 THIRD PARTY LICENSES ..... 6

**5 TECHNICAL MANUAL 6**

**6 USER MANUAL 6**

6.1 ACCESS ..... 6
6.2 USER MANUAL ..... 7

**7 FUTURE PLANS 13**

**8 REFERENCES 13**

# 1 Availability and Contacts

This table describes how to reach the prototype and the contact person in case of questions.

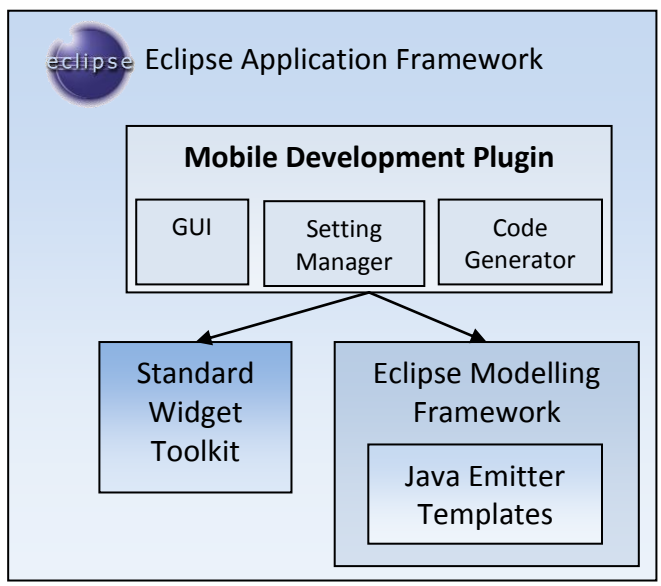
| | |
| --- | --- |
| **Version** | Final version |
| **Availability** | *http://www.msee-ip.eu/intranet/sp4-workspace/wp44-msee-generic-mobile-business-platform/d44.3-mobile-platform-first-prototype/* |
| **Accompanying specification and design document** | *MSEE D44.1 – MSEE D44.3* |
| **Source control** | *http://engrep.eng.it/svn/msee/SP4/* |
| **Contact person** | *Giovanni Casella – Softeco Sismat* – giovanni.casella@softeco.it |

## 2 Architecture and Functionalities

The goal of the Mobile Development module is to extend the MSEE Development Platform [1] addressing the development of applications targeting mobile devices. In particular the module supports the developers in the implementation of mobile applications based on back-and REST services.
The module, implemented as an Eclipse plug-in, supports the development of:

- REST services implemented with the Apache Jersey[1] framework.
- Web Applications designed for mobile phones implemented with the jQuery[2] and jQuery Mobile[3] libraries.

The following figure shows the internal high level architecture of the module:




Further details can be found in the document MSEE D44.1 “Mobile business platform specification and architecture” [1].

[1] Apache Jersey: jersey.java.net
[2] jQuery: jquery.com
[3] jQueryMobile: jquerymobile.com

# 3 Technical Information

## 3.1 Requirements

In order to use the module it is required:

- **Eclipse Juno**: http://eclipse.org/juno/
- **Apache Jersey 1.12**: http://maven.java.net/service/local/artifact/maven/redirect?r=releases&g=com.sun.jersey&a=jersey-archive&v=1.12&e=zip
- **jQuery 1.7.2**: http://code.jquery.com/jquery-1.7.2.js
- **jQuery Mobile 1.1**: http://code.jquery.com/mobile/1.1.1/jquery.mobile-1.1.1.js

## 3.2 Technical details

| **Nature** | **Eclipse plug-in** |
| --- | --- |
| **Programming Language** | Java |
| **Development Framework** | Eclipse Juno |
| **Additional libraries** | - |
| **Application Server** | - |
| **Database** | - |

# 4 Licensing

## 4.1 License

Proprietary licensing scheme.

## 4.2 Third party licenses

| | |
| --- | --- |
| Json Simple | Apache License 2.0 |
| Apache XML commons | Apache License 2.0 |
| Apache Xerces | Apache License 2.0 |

# 5 Technical Manual

The distribution file of the Mobile Development plug-in can be installed in Eclipse June simply copying it in the Eclipse "plugin" folder. At the next execution of Eclipse the wizard provided by the plug-in will be available. If new versions of the plug-in will be installed or if the new wizard does not appear start Eclipse using the "-clean" option.

# 6 User Manual

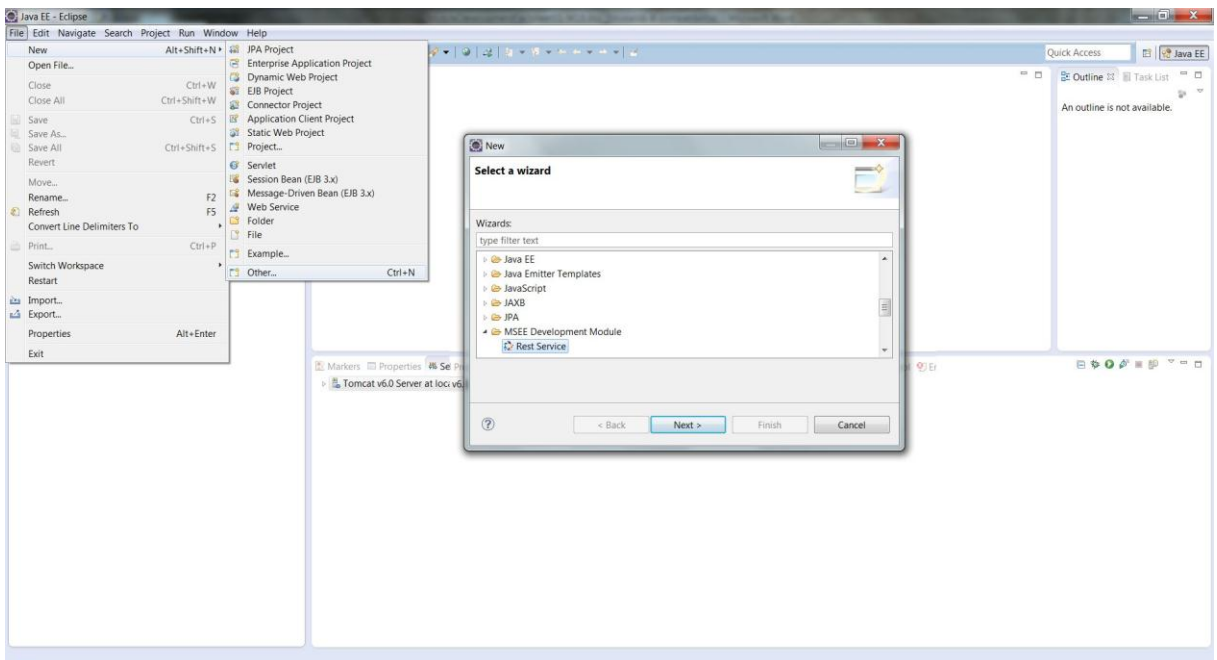
## 6.1 Access

The software component is available for download at:

*http://www.msee-ip.eu/intranet/sp4-workspace/wp44-msee-generic-mobile-business-platform/d44.3-mobile-platform-first-prototype/*

## *6.2 User Manual*

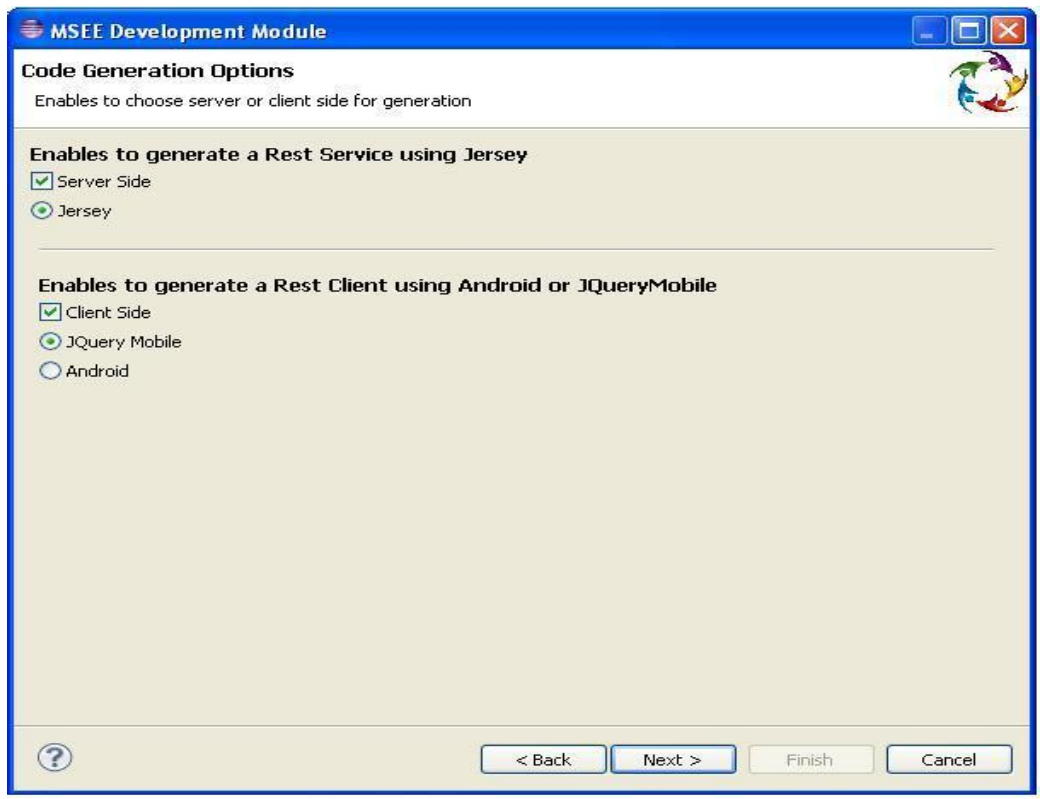
The wizard provided by the Mobile Development plug-in is composed by several pages. To start the wizard use the "File -> New -> Other...-> MSEE Development Module -> REST Service" menu.




In the following the pages of the wizard are described.

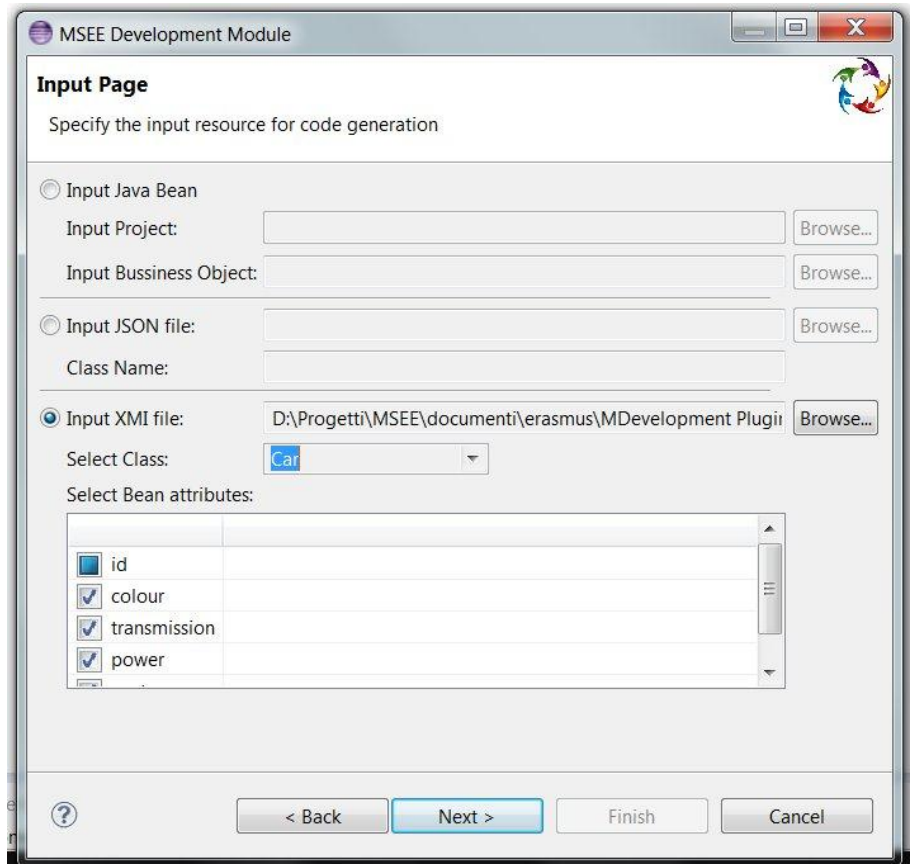
## Start Page

The first page enables the developer to decide if to generate only the REST service and/or a mobile application exploiting the service (REST Service). Moreover the page enables the developer to choose the base technology to use.



## Input Page

In this page the user must choose the business object that will be handled by the REST service (refer to [1] for additional details).




A REST resource can be the representation of a car, a user, a product, etc.

The user can provide the resource exploiting different ways:

- Input Java Bean: the resource is represented by a Java bean. A Java bean is a reusable software component for Java. In practice it is a Java class conforming to a particular convention. The user can select the project (Input Project) where this bean is available and the bean itself. The "Browse" buttons enables the developer to browse projects and classes.
- Input JSON file: the resource is represented by a JSON[4] document specifying its attributes. The "Browse" button enables the developer to select the document. Additionally the name of the resource must be provided in the "Class Name" text field.
- Input XMI file: the resource is represented as a class in a UML class diagram[5] exported as an XMI[6] document. Since a class diagrams can contain several class the developer must choose the class to use from the "Select Class" list box.

[4] JavaScript Object Notation: www.json.org
[5] UML class diagram: en.wikipedia.org/wiki/Class_diagram
[6] XML Metadata Interchange: www.omg.org/spec/XMI/

It is important to note that in order to be processed by the wizard the resource must include an id attribute. This is required to generate the URLs of the REST service. The developer can select only some attributes of the resource exploiting the "Select Attributes" table.

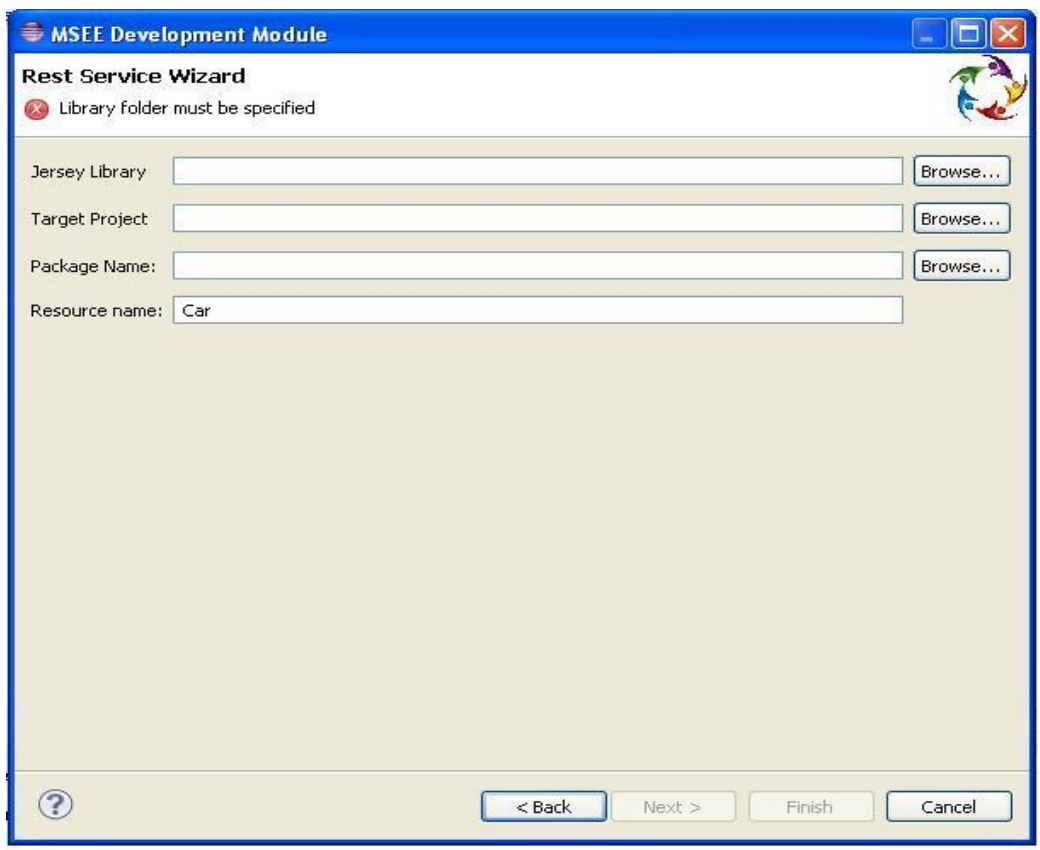
The module is able to take as input the UML class diagrams (available as XMI files) realized using the SLM Toolbox [1] as part of the BSM/TIM/TSM service models.

### Rest Service Options Page

This page enables the user to specify the settings needed to generate the REST service.

Description of the text fields:

- Jersey Library: the "Browse…" button enables the developer to choose the Jersey library folder.
- Target project: the "Browse…" button enables the developer to choose the project where the Rest Service code will be generated. Project must be an Eclipse "Dynamic Web Project".
- Package name: the "Browse…" button enables the developer to choose the Java package where the Rest Service code will be generated.
- Resource name: name of the resource managed by the REST service.

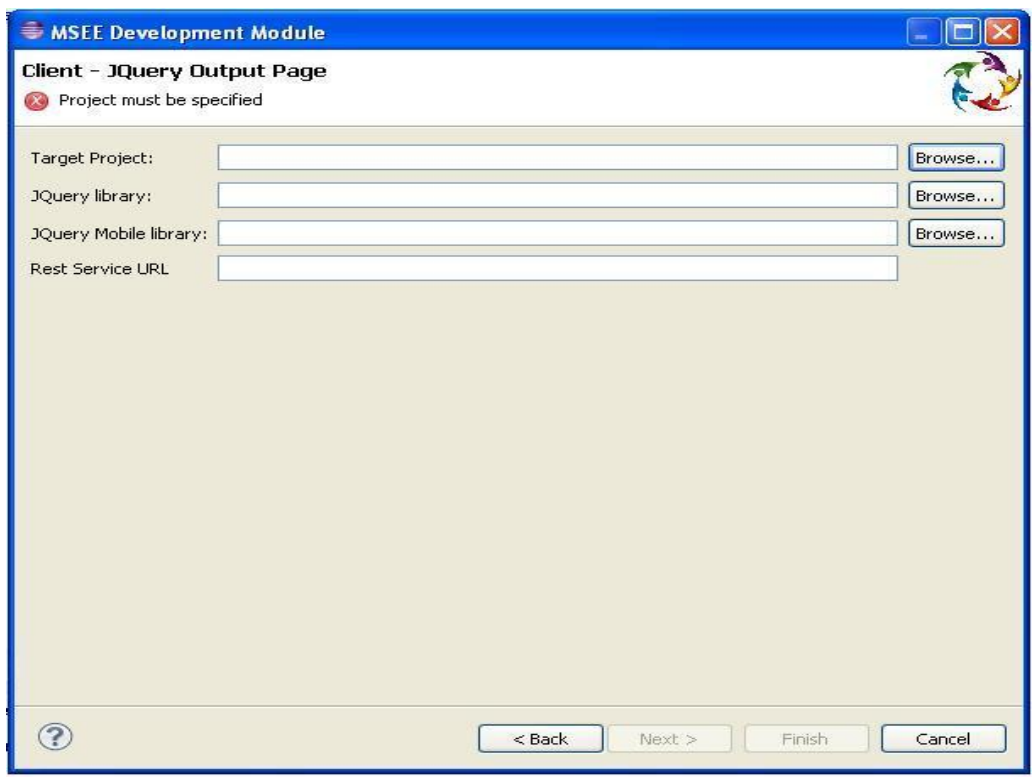


## Mobile Application Wizard Page

This page enables the user to specify the settings needed to generate the REST client, a web application designed for mobile phones based on the jQuery mobile framework.

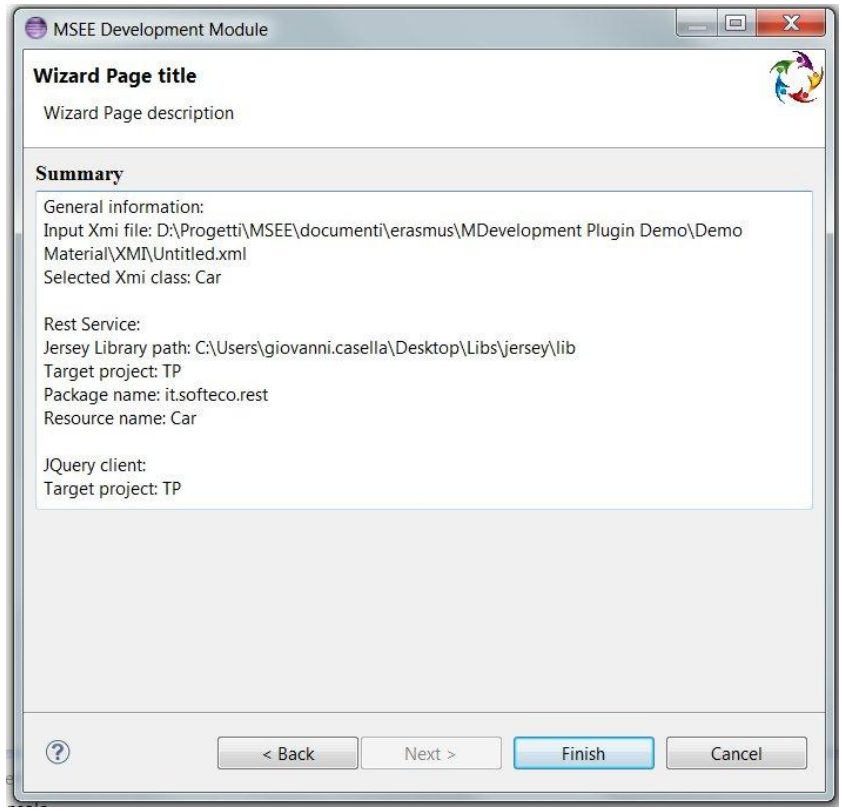
Description of the text fields:

- Target project: the "Browse…" button enables the developer to choose the project where the Rest Client will be generated. Project must be an Eclipse "Dynamic Web Project".
- JQuery Library: the "Browse…" button enables the developer to choose the JQuery library folder.
- JQuery Mobile Library: the "Browse…" button enables the developer to choose the JQuery mobile library folder.
- REST Service URL: address of the rest service to exploit. The field is filled automatically if the developer has chosen to generate the service with the wizard.



## Summary Page

This page enables the user to review selected options. Once the "Finish" button is pressed the plug-in generates all the needed code according user settings. The generated code can be tested executing the target project (the Eclipse dynamic web project where the code has been generated) on a server (i.e. Apache Tomcat 7).



## 7 Future Plans

The prototype will be integrated and used in the framework of the MSEE IT System [5]. Additional developments will be considered during the specification of the final prototype of the MSEE Mobile Business Platform [4] .

## 8 References

[1] D15.3 Specifications and Design of SLM Platform
[2] D42.1 Generic Service Development Platform specifications and architecture.
[3] D44.1 Mobile Business Platform Specification and Architecture.
[4] D44.2 Mobile Business Platform Specification and Architecture (final).
[5] D45.1 MSEE Service-System Integrated.