Lycia Development Suite

Lycia Mobile
Version 1.00 – July 2014
Revision number 1.00

Lycia Suite

Lycia Mobile

Part Number: 021-001-001-014

Olga Gusarenko

Technical Writer

Last Updated July 2014

Lycia Mobile

Part Number: 021-001-001-014

Published by:

Querix Ltd, 50 The Avenue, Southampton, SO17 1XQ, UK

Printing History:

July 2014: Beta edition

Last Updated:

July 2014

Documentation written by:

Olga Gusarenko

Notices

The information contained within this document is subject to change without notice. If you find any problems in the documentation, please, submit your comments to documentation@querix.com

Table of Contents

Introduction

Nowadays, the number of platforms, browsers and devices grow extremely fast, and the corresponding adoption of applications becomes one of the core tasks for the developers. With a view to your comfort, the Querix team provide you with the support of the mobile development framework, which enables to have a single application source for multiple target environment.

Moreover, we have developed a new thin client, named Lycia Touch, which is able to deploy your application to various mobile devices including smart phones and tablets on iOS or Android operating systems. On executing an app with Lycia Touch, standard graphical widgets such as date, time, combo and others are replaced with touch controls by the native mobile device system.

Among the advantages of Lycia Touch is that it, as well as Lycia DC or Lycia Web, is a middleware thin client, and, as a result, there is no difference between the deployment of a 4GL application to the server for the mobile or any other Lycia client, you have already experienced working with.

This document will give you answers to the questions relating to the process of a mobile application creation and its execution using Lycia Touch thin client, such as follows:

- What aspects must I focus on when creating an app?
- How can Lycia Touch be delivered and installed on my mobile device?
- How can I execute my app with Lycia Touch?

Please, be informed, that this guide is a work-in-progress one and may not reflect all of the latest changes relating to the subject. You are welcome to express your ideas concerning its improvement to our technical writer.

If you face any technical problems while using Lycia Touch, please, feel free to ask for technical service from our team.

App Development Peculiarities

To make an application look and feel properly when run on a mobile device with Lycia Touch, several aspects responsible for the responsive design must be taken into consideration within the process of its creation.

In order to achieve an optimal view of an application, that supposes easy-to-read and navigate through a close to an original program layout with a minimum of scrolling, resizing and panning the corresponding classes should be used.

Note, that no special file formats are required while an application for running on a mobile client creation. A mobile application is developed as a standard one with all the adaptive changes taking place within the existing file formats (*.4gl, *.qxtheme). Moreover, the same application can be successfully executed on the mobile client as well as on the desktop one.

Full-screen Windows

We provide you with a possibility of a user interface customization in several ways, without time-consuming recoding every time the user intends to utilize another device or thin client. These methods are described below: the 4GL source code modification (STYLE attribute and ui.Interface.getFrontEndName() method) and Theme Designer usage.

Such modifications in your application result in a native look of its windows, depending on the device it has been running on.

The STYLE attribute

Adding a special argument to an ATTRIBUTE clause of the OPEN WINDOW statement results in a full-screen view of a window it specifies.

The syntax of such a clause is as follows:

```
ATTRIBUTE(BORDER,STYLE="full-screen")
```

where:

full-screen	a STYLE attribute value, standing for a special class name, that enables the full-screen mode

Usage:

```
OPEN WINDOW w_full_screen WITH FORM "window_full_screen"   ATTRIBUTE(BORDER,
STYLE="full-screen")

# opens the window titled "w_full_screen" in a full-screen mode
```

ui.Interface.getFrontEndName() method

This method is generally used to retrieve the type of the front-end. Depending on a client or device an application is being run on, it may return different values.

If an application is run on a smartphone, tablet or any other mobile device, the method returns *lyciamobile* string.

The syntax of the method invocation is as follows:

```
LET client = CALL ui.Interface.getFrontEndName()
      IF client = "lycia_mobile" THEN
            DISPLAY "A mobile device is used"
      END IF
```

where:

lyciamobile	a returned value indicating that an application is running on a mobile device
client	any variable

Theme style modification

The user interface may also be adapted for the device utilized by the user by means of Theme Designer, a separate graphical tool developed by the Querix team, that is intended to manage an application view and functionality.

It allows an easy deployment of a number of special classes that adjust an application layout depending on the client or device. Here is the list of the available classes with an indication of the front-end an application is expected to be opened in:

lycia_normal	browser or desktop client, NOT on a mobile device
lycia_web	browser
lycia_desktop	LyciaDesktop
lycia_touch	a standalone mobile application
lycia_mobile	a mobile device (browser or a standalone application)
lycia_html5	HTML client, that includes all three - web, desktop, mobile

for browsers:

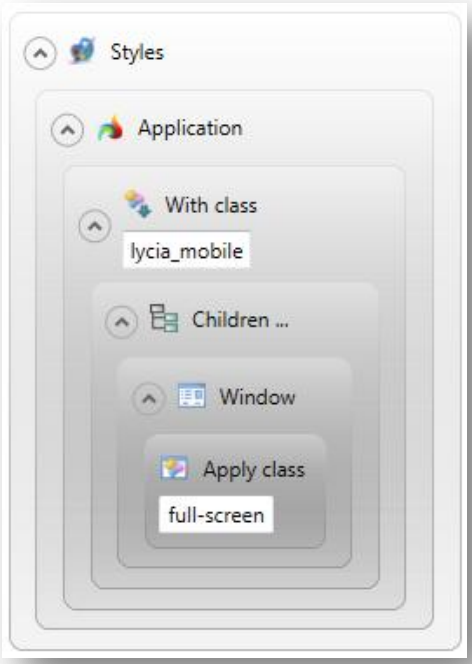
lycia_firefox	Mozilla Firefox®
lycia_opera	Opera®
lycia_safari	Safari®
lycia_ie	Windows Internet Explorer®

for mobile operating systems:

lycia_iphone	iOS
lycia_android	Android

Below is an illustration of how a *full-screen* class can be applied to all the windows of an application filtered with the *lycia_mobile* class filter. Note, that such a modification, based on applying the *lycia_mobile* class filter to the theme, effects an application only, if it is run on a mobile device:

Theme Styles diagram



XML source code extract

```
<ElementFilter ElementName="Application">
<StyleSheet>
<WithClassFilterClassName="lycia_mobile">
  <StyleSheet>
  <ChildFilter>
    <StyleSheet>
    <ElementFilter ElementName="Window">
       <StyleSheet>
        <DoStyleAction>
          <ApplyClass Name="full-screen" />
        </DoStyleAction>
        </StyleSheet>
     </ElementFilter>
     </StyleSheet>
   </ChildFilter>
   </StyleSheet>
</WithClassFilter>
</StyleSheet>
</ElementFilter>
```

Be aware of, that a single line of code of an already implemented 4GL application cannot be changed this way, as it simply defines the user theme, that takes precedence over the 4GL source code at runtime.

Orientation

One of the most important aspects of an application building for various devices is taking into consideration possible screen orientations and user interface changes this may cause.

In order to save your time when coding, we advise you to set an OnOrientationChanged event, that is application scoped. It is triggered each time a mobile device is rotated by the user. For its handling you should use the ON ACTION clause. Its syntax is as follows:

```
ON ACTION ("orientation_name")
```

where

orientation_name	a string, standing for the orientation name and can be one of the following *landscape*, *portrait*, *portraitup*, *portraitdown*, *landscapecw* or *landscapeccw*.

In order an application could know how the user interface should be changed depending on a screen position (horizontal or vertical) the *fgl_getproperty* function can be used.

Depending on the property name passed to it, the function can return the following values:

```
fgl_getproperty("gui", "application.deviceorientation")
```

- returns an integer that stands for the mobile device rotation degree: 90, -90 for the landscape and 0, 180 for the portrait orientation.

```
fgl_getproperty("gui", "application.deviceorientationname")
```

- returns a string that stands for the orientation of a mobile device. The last can be one of the following:

portraitup	screen is rotated 0
portraitdown	screen is rotated 180
lanscapecw	screen is rotated -90
landscapeccw	screen is rotated 90

Lycia Touch Installation

The process of the installation of Lycia Touch on your mobile device is simple and does not require any special knowledge base. It can be installed on your smartphone or tablet as any other mobile software.

Lycia Touch for iOS and Android devices is available for free on the corresponding distribution services, those are App Store for iOS and Google Play for Android.

To install the program, proceed as it described below:

1. fire up the required service,
2. search for Lycia Touch app once in the store,
3. tap the Install button to initialize an installation.

Under normal completion of the installation process, you can start the application by tapping the Lycia Touch icon from the file drawer of your mobile device.

As both Android and iOS devices support sideloading, you are able to download the latest version of Lycia Touch from the Downloads section of the Querix website (the .ipa file for iOS or the .apk file for Android).

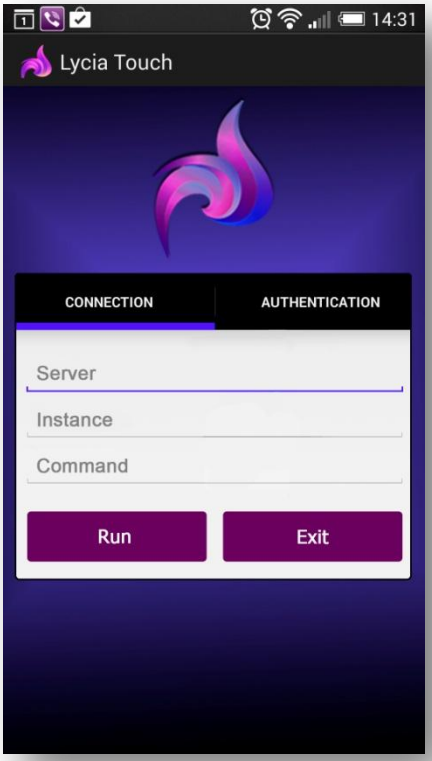
In spite of this, to make the installation process as simple as possible, we strongly recommend you to use the dedicated distribution services.

App Execution with Lycia Touch

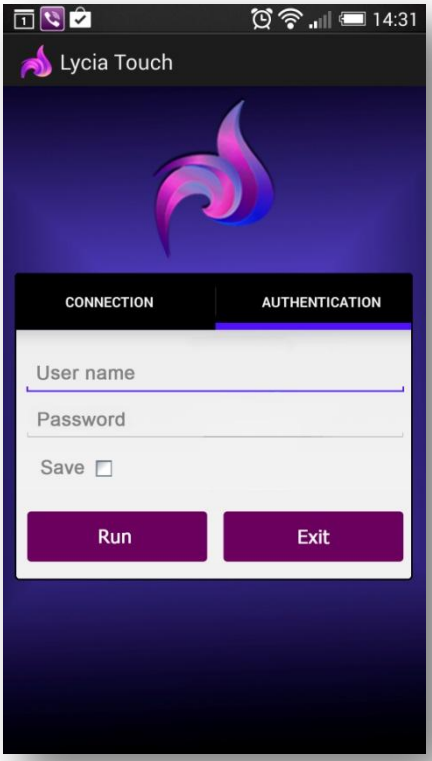
Once installed, Lycia Touch can be started by tapping a corresponding icon from the list of the apps installed on your mobile device.

The Lycia Touch connection dialog provides an interface for calling and running 4GL apps:

Connection Tab



Authentication Tab



As it can be viewed from the screenshots above, the mobile client connection dialog box includes two tabs. The **Connection** one allows entering the connection parameters for the server host name, instance and application name to be executed. There is also an **Authentication** tab asking for the user name and password if a secure service port is used by an application.

The **Server Host Name** field should contain the host name, e.g. 'localhost:9090'.

The Lycia App Server **Instance** stands for the service port or service name, (it is prompted when the Instance field is focused, e.g. '1889').

The **Command** field may contain either the name of the executable 4GL file, or the relative path, if the program file is in a subfolder on the application server. The specified value here can be followed by the '-d <db-name>' argument or any custom argument needed.

Be sure to separate the name of the application and all the arguments by spaces. Here are some examples of the Command field content:

- guidemo
- guidemo5.1/guidemo.exe
- guidemo5.1/guidemo par1 par2 -d oracle

Lycia Touch thin client supports only Lycia II or later versions of an application server, but the same client can be used for different back-end connections.