Cooperation – Mobile offline web service

Summary

The mobile offline web service provides the function for the users to make list inquiry and detailed inquiry and for the managers to manage posts online and offline.

This component is relatively similar to the online ebook service.

The user can read ebook offline the same as online.

The offline web service is realized using the offlineweb technology of HTML5.

The page that will be viewed offline can be designated using offlineweb.

In terms of offlineweb utilization of HTML5, the ‘mobile synchronization service’ component and ‘mobile offline web service’ starts from the same service of using offlineweb. However, the ‘synchronization service’ expanded the utilization by adding web storage on the offlineweb of HTML5; whereas, the offlineweb service is created only by the HTML5 offileweb technology.

\* Function Flow



The development and testing of the mobile screen of this component is completed. It is also noted in development since the supports on HTML5 and CSS3 vary between the PC browser and mobile browser in the same brand.

The tested device and browsers are GalaxyS2, GalaxyS, iPad2, iPhone4, GalaxyTab1 and the web browsers for Android, Firefox, Safari and Opera Mobile.

The test results are as following.

Android

The operation of HTML5, CSS3 and components of the Firefox(6.xx) and opera mobile was stable in android OS 2.3 (gengerbread).

However, in the lower version browsers, android 2.1, 2.2, the operation of components was unstable and UI realization was not smooth due to insufficient HTML5 and CSS3 supports.

For Opera Mobile, it caused problems in colors and event processing in UI. Therefore, it was excluded from the subject. Galaxy, GalxyS2 and GalaxyTab1 were chosen as test mobile devices.

iOS

The iOS is one of three supported browsers and was tested in versions 4.2 and 4.3. The support for HTML5 and CSS3 were relatively smooth.

iPhone4 and iPad2 were chosen as test mobile devices.

Description

The offline web service is composed of management functions – offline web list inquiry, detailed inquiry, register, update and delete – and user functions – offline web list inquiry and detailed inquiry. (Refer to Related Functions)

Class Diagram



Related sources

Related Tables

|  |  |  |
| --- | --- | --- |
| Table name | Table name (English) | Notes |
| Offline web | C OMTNOFFLINEWEB | Manage offlineweb |

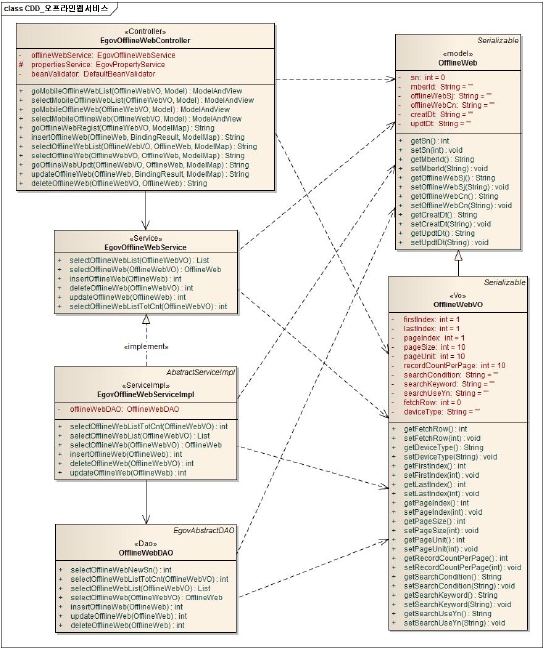
Login / membership management / permission management

Offline web service requires the administrator authentication in offline post list inquiry , detail inquiry, registration , modification and deletion.

For authentication, you can connect with login, membership management and permission management components. If you use Spring Security for authentication, make change as in the followings..

src/main/java/resources/egovframework/spring/com/context-common.xml

<!-- Service class that can check the authenticated user’s LoginVO, permissions and authentication -->



|  |  |  |
| --- | --- | --- |
| Type | Target sources | Notes |
| C ontroller | egovframework.mbl.com.ows.web.EgovOfflineWebC ontroller.java | Offline web C ontroller C lass |
| Service | egovframework.mbl.com.ows.service.EgovOfflineWebService.java | Offline web Service C lass |
| ServiceImpl | egovframework.mbl.com.ows.service.impl.EgovOfflineWebServiceImpl.java | Offline web ServiceImpl C lass |
| Model | egovframework.mbl.com.ows.service.OfflineWeb.java | Offline web Model C lass |
| VO | egovframework.mbl.com.ows.service.OfflineWebVO.java | Offline web VO C lass |
| DAO | egovframework.mbl.com.ows.service.impl.OfflineWebDAO.java | Offline web Dao C lass |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovMobileOfflineWebList.jsp | Offline web mobile list inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovMobileOfflineWeb.jsp | Offline web mobile detail inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovOfflineWebList.jsp | Offline web list inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovOfflineWebDetail.jsp | Offline web detail inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovOfflineWebInsert.jsp | Offline web registration page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/ows/EgovOfflineWebUpdate.jsp | Offline web modification page |
| QUERY XML | resources/egovframework/sqlmap/mbl/com/ows/EgovOfflineWeb\_SQL\_XXX.xml | Offline web QUERY XML |
| Validator Rule XML | resources/egovframework/validator/validator-rules.xml | XML that defines Validator Rule |
| Validator XML | resources/egovframework/validator/mbl/com/ows/EgovOfflineWeb.xml | Offline web Validator XML |

<bean id="egovUserDetailsHelper" class="egovframework.com.cmm.util.EgovUserDetailsHelper">

<property name="egovUserDetailsService">

<ref bean="egovUserDetailsSecurityService" />

</property>

</bean>

<!-- Bean that will use Security-based authentication -->

<bean id="egovUserDetailsSecurityService" class="egovframework.com.sec.ram.service.impl.EgovUserDetailsSecurityServiceImpl"/>

web.xml

<!-- 2. Spring Security Setup -->

<filter>

<filter-name>springSecurityFilterChain</filter-name>

<filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>

</filter>

<filter-mapping>

<filter-name>springSecurityFilterChain</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

<listener>

<listener-class>org.springframework.security.ui.session.HttpSessionEventPublisher</listener-class>

</listener>

<!-- security end -->

<!-- 4. Filter setup for Spring Security logout setup -->

<filter>

<filter-name>EgovSpringSecurityLogoutFilter</filter-name>

<filter-class>egovframework.com.sec.security.filter.EgovSpringSecurityLogoutFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>EgovSpringSecurityLogoutFilter</filter-name>

<url-pattern>/uat/uia/actionLogout.do</url-pattern>

</filter-mapping>

<!-- 4. Filter setup for Spring Security login setup -->

<filter>

<filter-name>EgovSpringSecurityLoginFilter</filter-name>

<filter-class>egovframework.com.sec.security.filter.EgovSpringSecurityLoginFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>EgovSpringSecurityLoginFilter</filter-name>

<url-pattern>\*.do</url-pattern>

</filter-mapping>

As in the above, add Spring Security filter to web.xml.

For details on login/membership management/permission management, refer to the followings.

login: general login

Membership management: user management

Permission management: Permission management

The following pages need the user/administrator permission.

Permission type Page path

Description

Administrator

/mbl/com/syn/selectOfflineWebList.mdo Screen of list inquiry of the offline post

Administrator

/mbl/com/syn/selectOfflineWeb.mdo

Screen of detail inquiry of the offline post

Administrator

/mbl/com/syn/goOfflineWebRegist.mdo Screen of offline post registration

Administrator

Administrator

Administrator

Administrator

/mbl/com/syn/insertOfflineWeb.mdo

/mbl/com/syn/goOfflineWebUpdt.mdo

/mbl/com/syn/updateOfflineWeb.mdo

/mbl/com/syn/deleteOfflineWeb.mdo

offline post registration

offline post modification screen

offline post modification

offline post deletion

Offline settings and notes for offline web

\* The currently distributed mobile common component adopts HTML5 and CSS3 and this means that it doesn’t support the standards for mobile web site validity check of Mobile OK and W3C. For details on correct validity check, please contact the competent professional institute.

\* Security verification is under progress on the currently distributed mobile component. We plan to complete security verification as soon as possible to release a new security-patched version.

To use html5 Offline web, some setting information is required.

- In web.xml, set the mime type.

- Create the xxxx.manifest file under the root and write <html manifest=”/xxxx.manifest”> in the top of the page to be used offline.

- Write the page to be applied in /xxxx.manifest.

For the standard framework, since the jsp file is written in the WEB-INF folder, if it is written in a format of /WEB-INF/xx/xx.jsp, it will not be a subject of offline work.

The xx.css, xx.js, xx.jpg and xxxx.manifest, including jsp used under root can be subjects of offline work.

When applying the files for offline work to xxxx.manifest file, you will not have to write css and js files and imported file into the jsp file

in the xxxx.manifest file.

But if the path of the imported file is not correct, offline application may not work. Pay attention to the path of the file to be imported.

.

When you modify and test the file set up in the xxxx.manifest file, you have to delete the browser cache.

If you just refresh the browser after modifying the code, the code may not be applied.

web.xml Settings

<mime-mapping>

<extension>manifest</extension>

<mime-type>text/cache-manifest</mime-type>

</mime-mapping>

xxxx.manifest Settings

CACHE MANIFEST

# version 1.0.01

# 2011.09.21

# Explicitly cached

CACHE:

./css/egovframework/mbl/cmm/jquery.mobile-1.0b3.min.css

./css/egovframework/mbl/cmm/EgovMobile.css

./css/egovframework/mbl/mcomd/egovMcomd.css

./js/egovframework/mbl/cmm/jquery-1.6.2.min.js

./js/egovframework/mbl/cmm/jquery.mobile-1.0b3.min.js

./js/egovframework/mbl/cmm/EgovMobile.js

./js/egovframework/mbl/cmm/datepicker/jquery.mobile.datebox.js

./mbl/com/ows/goMobileOfflineWebList.mdo

./mbl/com/ows/goMobileOfflineWeb.mdo

# Display fallback.html for all uncached pages

FALLBACK:

# Following resources require network connection.

NETWORK:

\*

Offline web coding Tip

In implementing HTML5 Offline web services, by right-clicking on the Chrome browser and selecting the Inspect Element to use the developer’s tool, you can reduce the problems caused by the browser cache or xxxx.manifest file.

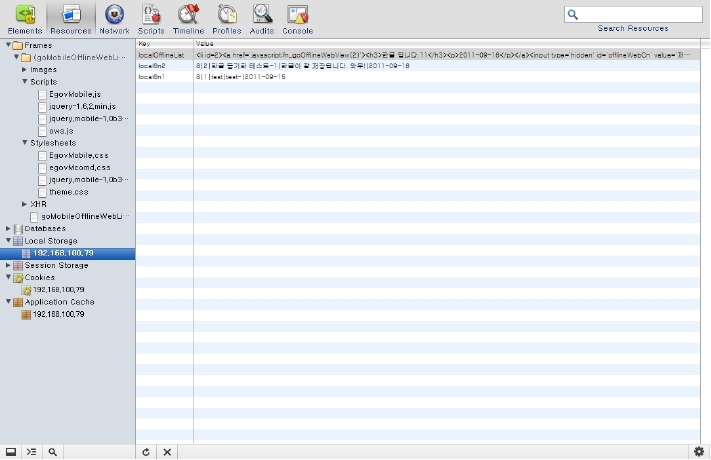
For instance, by selecting Resource tab of the Chrome browser’s developer tool, you can check Local Storage and Application Cache.

By clicking and expanding Application Cache, you can view the pages for offline web written in xxxx.manifest.

If there is nothing written in xxxx.manifest in Application Cache, it means that the offline web is not applied.

If you use debugging using the Chrome browser’s developer tool in the development, you can have more stable offline web coding.

The following figure shows samples applied to Local Storage and Application Cache.



Offline web mobile list inquiry

Business rules

Offline web service uses ajax communication based json and form transfer types to process data.

Page navigation is totally based on the form type.

You can retrieve message lists for Offline web service.

By clicking View More button, you can add another five messages to a retrieved list.

You can move to Offline web service registration screen.

If converted to the offline environment, the list inquiry and detail inquiry are carried out as in the online environment.

But the list inquiry should be carried out at least once in the online environment.

When the list inquiry is carried out online, the information contained in the list is stored the web storage

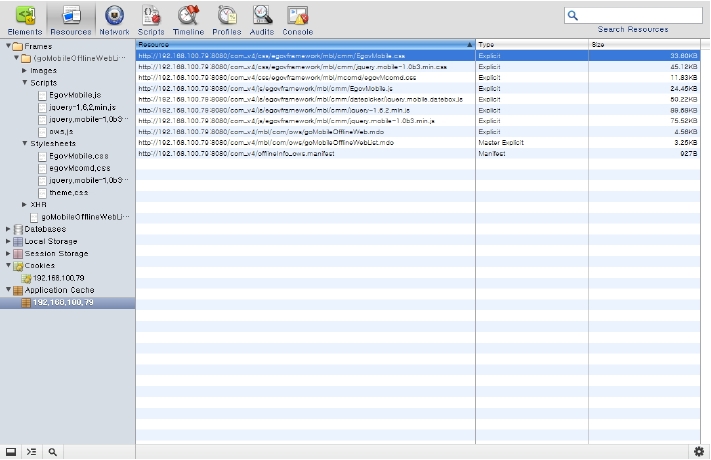
and this is used offline.

Related codes

localStorage.setItem("localOfflineList", html);

$('div[id="list"] ul[data-role="listview"]').html(html).listview('refresh');

Screen and execution manual



|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| Moves to inquiry page | /mbl/com/ows/goMobileOfflineWebList.mdo | goMobileOfflineWebList |  |
| Inquiry | /mbl/com/ows/selectMobileOfflineWebList.mdo | selectMobileOfflineWebList | “OfflineWebDAO.selectOfflineWebList” |
| View more | /mbl/com/ows/selectMobileOfflineWebList.mdo | selectMobileOfflineWebList | “OfflineWebDAO.selectOfflineWebList” |
| Moves to detail page | /mbl/com/ows/goMobileOfflineWeb.mdo | goMobileOfflineWeb |  |

Offline web mobile detail inquiry

Business Rules

Offline web service uses ajax communication based json and form transfer types to process data.

Page navigation is totally based on the form type.

Offline web service detail posts are being retrieved.

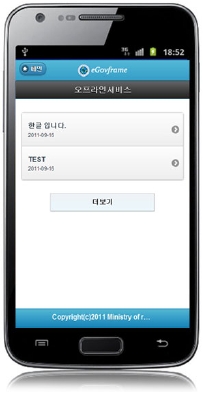
The detail posts are process by using web storage information acquired during the online list inquiry.

Deletion is being carried out.

Related codes

N/A

Screen and execution manual



|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| List | /mbl/com/ows/goMobileOfflineWebList.mdo | goMobileOfflineWebList |  |

Offline web list inquiry

Business Rules

You can use search conditions and key words to retrieve offline web service post lists.

10 rows are displayed on a page. When exceeding 10 rows, paging at the bottom is used.

Related codes

N/A

Screen and execution manual

|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| Inquiry | /mbl/com/ows/selectOfflineWebList.mdo | selectowsList | “OfflineWebDAO.selectOfflineWebList” |
|  |  | “OfflineWebDAO.selectOfflineWebTotC nt” |  |
| Moves to the registration page | /mbl/com/ows/goOfflineWebRegist.mdo | goowsRegist |  |



Offline web detail inquiry

Business Rules

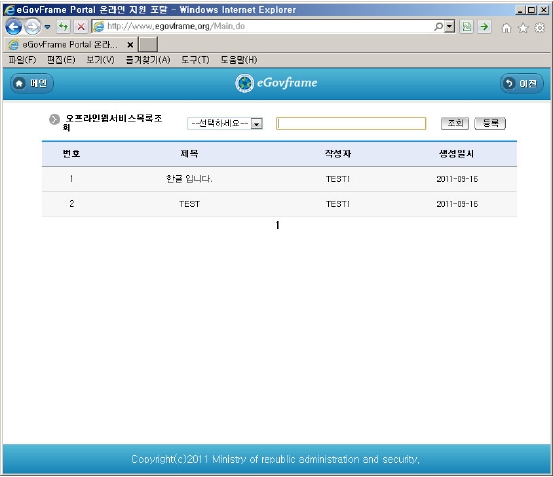
Offline web service detail posts are being retrieved.

Related codes

N/A

Screen and execution manual

|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| Delete | /mbl/com/ows/deleteOfflineWeb.mdo | deleteOfflineWeb | “OfflineWebDAO.deleteOfflineWeb” |
| Moves to modification page | /mbl/com/ows/goOfflineWebUpdt.mdo | goowsUpdt |  |
| List | /mbl/com/ows/selectOfflineWebList.mdo | selectowsList | “OfflineWebDAO.selectOfflineWebList” |



Offline web registration

Business Rules

Offline web service posts are registered and titles and contents are inputted.

The title and content are required items.

Related codes

N/A

Screen and execution manual



|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| Register | /mbl/com/ows/insertOfflineWeb.mdo | insertOfflineWeb | “OfflineWebDAO.insertOfflineWeb” |
| Cancel |  |  |  |
| List | /mbl/com/ows/selectOfflineWebList.mdo | selectowsList | “OfflineWebDAO.selectOfflineWebList” |

Offline web modification

Business Rules

Offline web service posts are modified and titles and contents are inputted.

The title and content are required items.

Related codes

N/A

Screen and execution manual



|  |  |  |  |
| --- | --- | --- | --- |
| Action | URL | Controller method | QueryID |
| Cancel |  |  |  |
| Store | /mbl/com/ows/updateOfflineWeb.mdo | updateows | “OfflineWebDAO.updateOfflineWeb” |
| List | /mbl/com/ows/selectOfflineWebList.mdo | selectowsList | “OfflineWebDAO.selectOfflineWebList” |

References

Distribution and test: Mobile Common Component System and distribution package plan

Related database structure and schema: related database structure and schema

e govfram e work /m com /Offline web service.tx t · Last m odifie d: 2011/10/10 15:24 by m com

Except where otherwise noted, content on this wiki is licensed under the following license:CC Attribution-Noncommercial-Share Alike 3.0 Unported

[http://creativecommons.org/licenses/by-nc-sa/3.0/]

