

Show pagesource

Old revisions

Print view

Recent changes

Search

Trace: » [Common component guide](#) » [Test 2](#) » [Test 1](#) » [Test 4](#) » [Location Information Connection](#) » [mms Service Connection v1.0](#) » [mms Service Connection v2.0](#) » [Multimedia control](#) » [mcom](#) » Synchronization Service

Table of Contents

- [Cooperation](#) – Synchronization Service
 - [Outline](#)
 - [Description](#)
 - [Synchronization service related offline setup and caution](#)
 - [Offline web related coding tip](#)
 - [Synchronization mobile list inquiry](#)
 - [Synchronization mobile detailed inquiry](#)
 - [Synchronization mobile registration](#)
 - [Synchronization mobile revision](#)
 - [Synchronization list inquiry](#)
 - [Synchronization detailed inquiry](#)
 - [Synchronization registration](#)
 - [Synchroniation revision](#)

[Reference](#)

Cooperation – Synchronization Service

Summary

The synchronization service provides the functions for the users to inquire, register, update and delete posts and for the managers to manage the posts both online and offline.

It is similar to Gmail service provided by the Google. In Gmail, user can check the email list and compose email without the network connection. And the email composed offline is sent when the network is normalized.

The synchronization service used the localStorage of HTML5 for implementation.

To look into the implementation, the offline web is used to designate the page though the network is disconnected and webstorage is used to display data. The data types include base data, new data, modified data and deleted data.

The data is categorized and stored in the webstorage.

- The base data is the data retrieved online and stored in the webstorage
- The new data is the data registered offline, created newly and stored in the webstorage
- The modified data is the base or new data modified offline and stored in the webstorage
- The deleted data is the base, new, modified data deleted offline and stored in the webstorage

In terms of offlineweb utilization by HTML5, 'Mobile synchronization service' and 'Mobile

offlineweb service' build on the same base, that is 'offlineweb utilization'. The mobile offlineweb service added webstorage to HTML5 offlineweb to expand the utilization, while the offlineweb service use HTML5 offlineweb technology only.

* Function flow



This component has been developed and tested on the browsers available in the mobile devices.

Since the browser support is different depending on the device (PC or mobile device), development and application should be carefully carried out.

The test devices include Galaxy S2, Galaxy S, Galaxy Tab 1, iPad2, iPhone 4.

The test mobile browsers include Android basic browser, Firefox, Safari and Opera Mobile.

The results are as follow.

Android

For the Android web browser, Firefox(6.xx), opera mobile, in android OS 2.3(gingerbread), the execution of HTML5, CSS3 and components is supported stably..

However, in Android 2.1, 2.2, the previous version of Android OS 2.3 (gingerbread), the support for HTML5 and CSS3 is not sufficient to implement the component UI properly..

Opera Mobile was excluded from the support as it caused some problems with handling colors and events.

For testing, Galaxy, Galaxy S2 and Galaxy Tab 1 were used.

iOS

iOS is one of the three supported browsers and both version (iOS 4.2, 4.3) supported HTML5 and CSS3 relatively well.

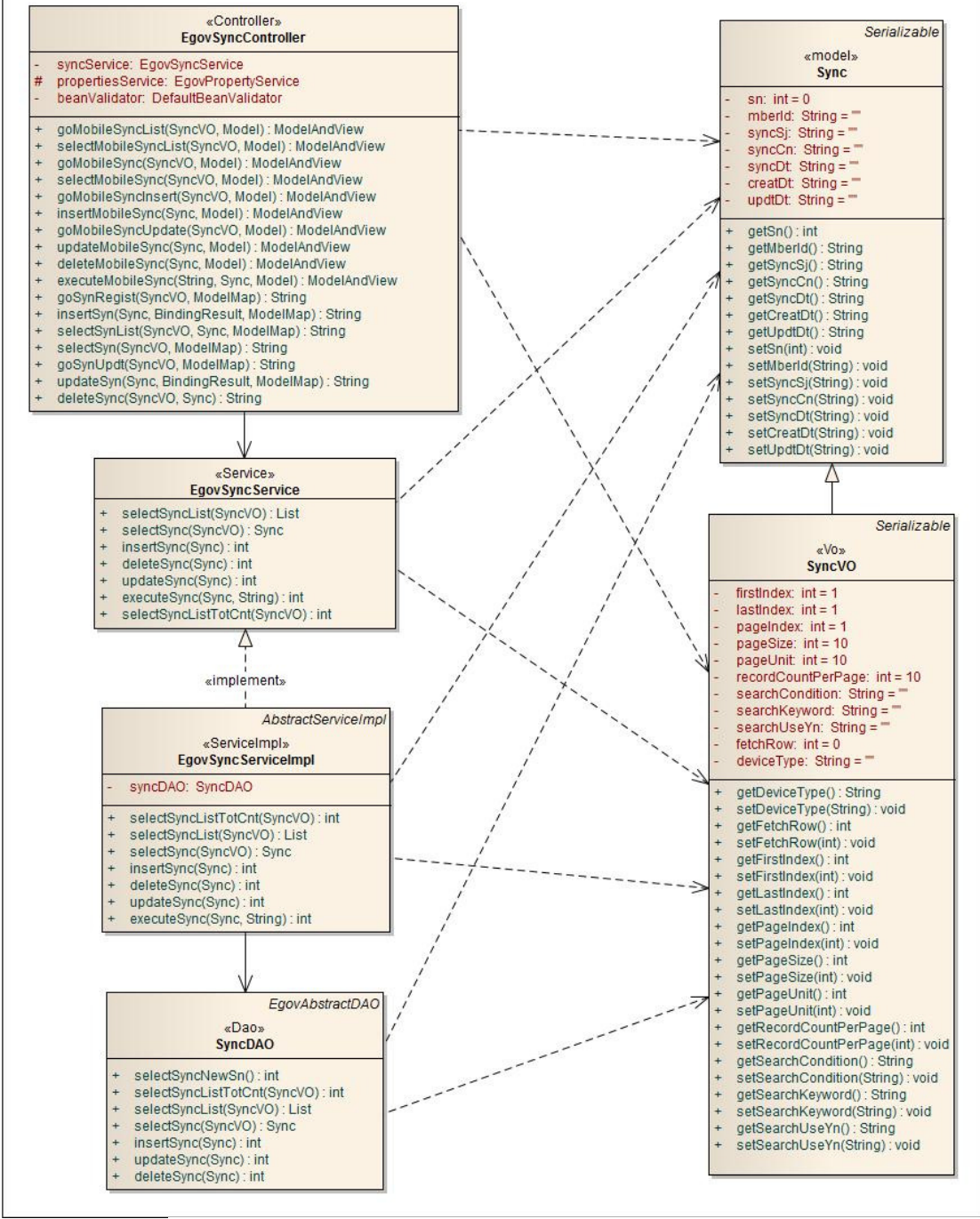
iPhone4 and iPad2 were used for testing..

Descriptions

Synchronization service is composed of management functions (**Synchronization list inquiry, detail inquiry, registration, modification and deletion**) and user functions (**Synchronization service list inquiry, detail inquiry, registration, modification and deletion**)(refer to the relevant functions)

Class diagram

class CDD_동기화서비스



Related sources

| Type | Target source names | Note |
|-------------|---|-------------------------------------|
| Controller | egovframework.mbl.com.syn.web.EgovSyncController.java | Synchronization on Controller Class |
| Service | egovframework.mbl.com.syn.service.EgovSyncService.java | Synchronization on Service Class |
| ServiceImpl | egovframework.mbl.com.syn.service.impl.EgovSyncServiceImpl.java | Synchronization |

| | | |
|--------------------|---|--|
| | | ServiceImpl Class |
| Model | egovframework.mbl.com.syn.service.Sync.java | Synchronization Model Class |
| VO | egovframework.mbl.com.syn.service.SyncVO.java | Synchronization VO Class |
| DAO | egovframework.mbl.com.syn.service.impl.SyncDAO.java | Synchronization Dao Class |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovMobileSyncList.jsp | Synchronization mobile list inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovMobileSync.jsp | Synchronization mobile detail inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovMobileSyncInsert.jsp | Synchronization mobile registration page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovMobileSyncUpdate.jsp | Synchronization mobile modification page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovSyncList.jsp | Synchronization list inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovSyncDetail.jsp | Synchronization detail inquiry page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovSyncInsert.jsp | Synchronization registration page |
| JSP | /WEB-INF/jsp/egovframework/mbl/com/syn/EgovSyncUpdate.jsp | Synchronization modification page |
| QUERY XML | resources/egovframework/sqlmap/mbl/com/syn/EgovSync_SQL_XXX.xml | Synchronization QUERY XML |
| Validator Rule XML | resources/egovframework/validator/validator-rules.xml | XML that defines Validator Rule |
| Validator XML | resources/egovframework/validator/mbl/com/syn/EgovSync.xml | Synchronization Validator XML |

Related tables

| Table name | Table name (English) | Notes |
|-----------------|-----------------------|------------------------|
| Synchronization | COMTNSYNC | Manage Synchronization |

Login/membership management/permission management

Synchronization service requires the user/administer authentication in **Synchronization post list inquiry, detail inquiry, registration, modification and deletion**. or authentication, you can use connect to login, membership management, and permission management component. To use Spring Security for authentication, make changes as in the following.

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

```
<!--Service class that can check the authenticated user's LoginVO, permissions and authentication-->
<bean id="egovUserDetailsHelper" class="egovframework.com.cmm.util.EgovUserDetailsHelper">
  <property name="egovUserDetailsService">
    <ref bean="egovUserDetailsService" />
  </property>
</bean>

<!-- Bean that will use Security-based authentication -->
<bean
class="egovframework.com.sec.ram.service.impl.EgovUserDetailsServiceServiceImpl"
id="egovUserDetailsService" />
```

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 permissions.

| Permission type | page path | Description |
|-----------------|-------------------------------------|---|
| User | /mbl/com/syn/goMobileSyncList.mdo | mobile Synchronization post list inquiry screen |
| User | /mbl/com/syn/goMobileSync.mdo | mobile Synchronization post detail inquiry screen |
| User | /mbl/com/syn/goMobileSyncInsert.mdo | mobile Synchronization post registration screen |
| User | /mbl/com/syn/goMobileSyncUpdate.mdo | mobile Synchronization post modification screen |
| User | /mbl/com/syn/deleteMobileSync.mdo | mobile Synchronization post deletion |
| Administrator | /mbl/com/syn/selectSyncList.mdo | Synchronization post list inquiry screen |
| Administrator | /mbl/com/syn/selectSync.mdo | Synchronization post detail inquiry screen |
| Administrator | /mbl/com/syn/goSyncRegist.mdo | Synchronization post registration screen |
| Administrator | /mbl/com/syn/insertSync.mdo | Synchronization post registration |
| Administrator | /mbl/com/syn/goSyncUpdt.mdo | Synchronization post modification screen |
| Administrator | /mbl/com/syn/updateSync.mdo | Synchronization post modification |
| Administrator | /mbl/com/syn/deleteSync.mdo | Synchronization post deletion |

Synchronization service offline Settings and Notes

* 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.

* The currently distributed mobile common component V 2.0 has completed the security verification on the source code.

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/com/syn/syn.js
./mbl/com/syn/goMobileSynclList.mdo
./mbl/com/syn/goMobileSync.mdo
./mbl/com/syn/goMobileSyncInsert.mdo
./mbl/com/syn/goMobileSyncUpdate.mdo

# Display fallback.html for all uncached pages

FALLBACK:

# Following resources require network connection.

NETWORK:
```

*

Since the code ("window.navigator.onLine") that determines whether to be online or offline on some mobile devices (Galaxy S and S2) doesn't work, use the following code.

src/main/webapp/js/egovframework/mbi/com/syn/syn.js

```
// Online status
var onLine = true;
// True if the device is Galaxy S or S2
var deviceStatus = false;

// if the device is Galaxy S or S2:
if (navigator.userAgent.indexOf("SHW-M110") > -1 ||
    navigator.userAgent.indexOf("SHW-M250") > -1) {
    deviceStatus = true;
}

if(deviceStatus==true){
    // check the initial connection status
    testConnection();
    // check the connection status regularly
    setInterval(function () {
        // check the connection status
        testConnection();
    }, 10000); // at a 10 seconds internal
}

// propagate events.
function triggerEvent(type) {
    var event = document.createEvent('HTMLEvents');
    event.initEvent('onLine', type, false);
    event.eventName = type;
    window.dispatchEvent(event);
}
```



```

        if(type==true){
            try{
                synchronizer();
            }catch(e){
            }
        }
    }

// return results after connecting with actual HTTP .
function testConnection() {
    if(deviceStatus!=true){
        onLine = window.navigator.onLine;
        return;
    }
    onLine = true;

    var xhr = new XMLHttpRequest();
    xhr.open('HEAD', '/', false); // async=false (synch mode)
    try {
        xhr.send(null);
        onLine = true;
    } catch (e) {
        // If connection fails, NETWORK_ERR occurs.
        onLine = false;
    }

    // propagate the event
    triggerEvent(onLine);
}

```

Offlineweb 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.

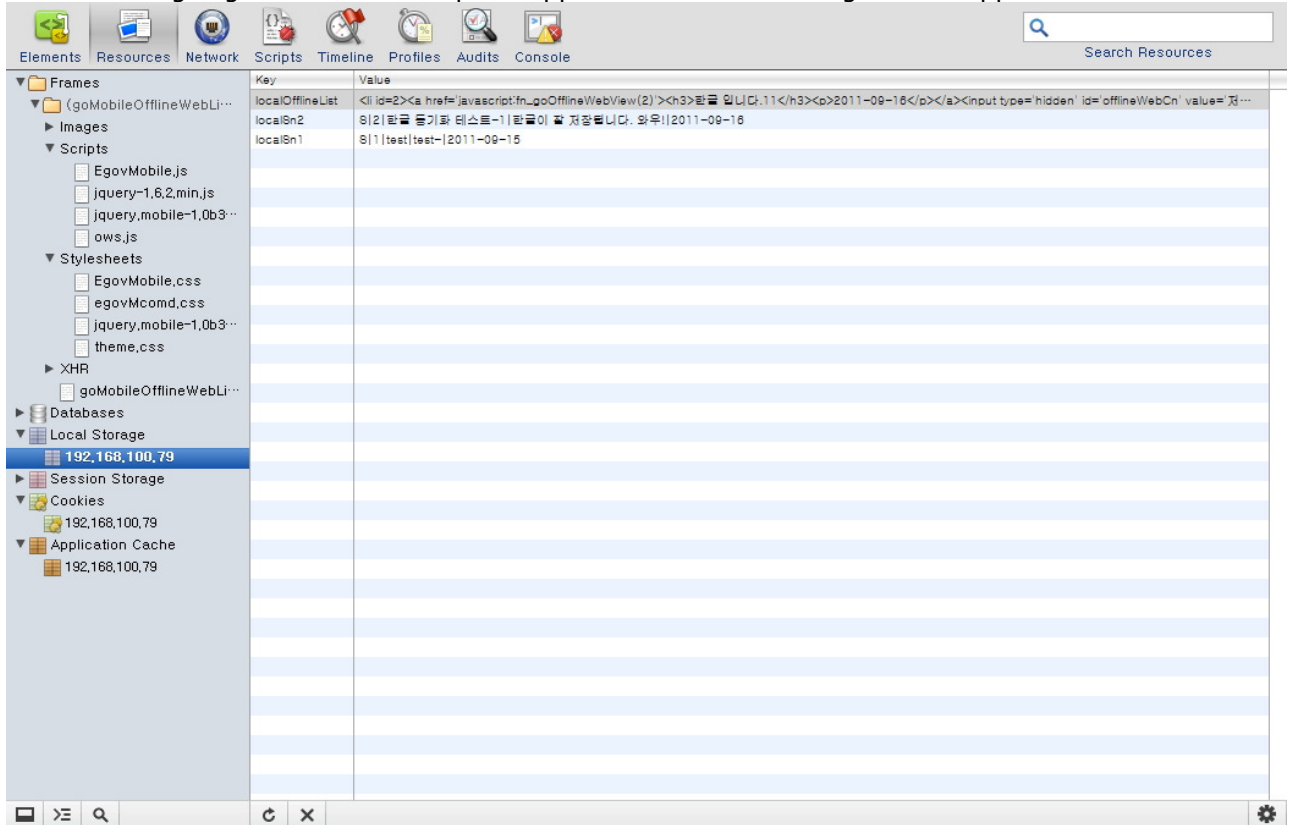
For instance, by selecting Resource tab of the Chrome browser's developer tool, you can check Local Storage and Application Cache.

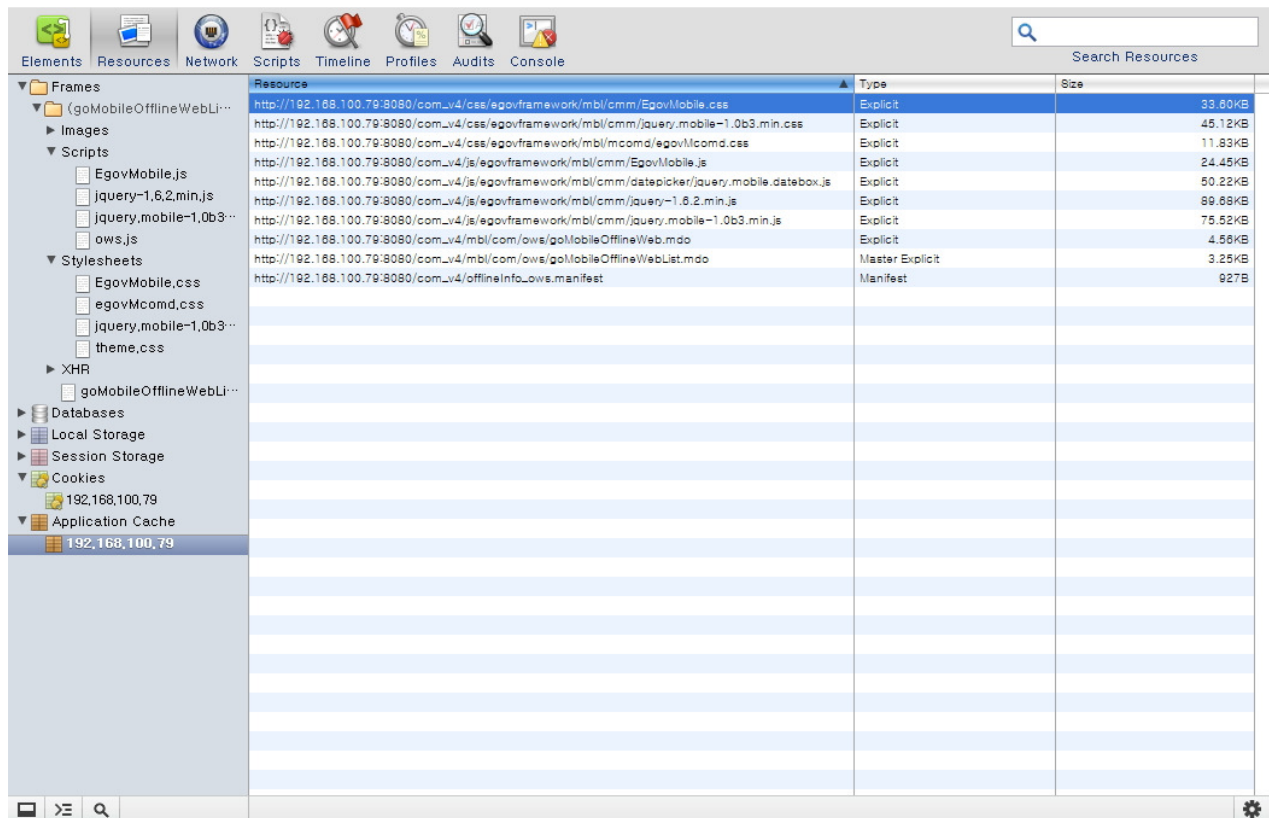
By clicking and expanding Application Cache, you can view the pages for offline web written in xxxx.manifest.

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.





Synchronization mobile list inquiry

Business rules

Synchronization 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 Synchronization service.

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

By clicking Write button, you can move to Synchronization service registration screen.

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

It is designed that data done offline (CRUD) are stored in the webstorage and, when converted to the online environment, all are reflected to DBMS.

But the list inquiry should be carried out at least once in the online environment.(assign data to the webstorage once or more by using the list inquiry.)

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

```

window.addEventListener("online", function(e) {

    localStorage.removeItem("localSn");

    var localData = "";

    var rowFlag = "▽";

    for(var i=1; i<=localStorage.length; i++) {

```

```

        var data = localStorage.getItem("localSn" + i).split("|");
        if (data[0] != "S") {
            if (!(data[0] == "D" && data[1] == "0")) {
                if (i == 1) {
                    localData = localStorage.getItem("localSn" + i)
                } else {
                    localData += rowFlag + localStorage.getItem("localSn" +
i);
                }
            }
        }
    }
    if(localData.length > 0) {
        var uri = "<c:url value='/mbl/com/syn/executeMobileSync.mdo'/>";
        $.ajax({
            type      : "POST",
            cache     : false,
            url       : uri,
            data      : {localData:localData},
            dataType  : "json",
            success   :function(json) {
                fn_synList();
            }
        });
    } else {
        fn_synList();
    }
}, true);
function fn_synList() {
    // var uri = "<c:url value='/mbl/com/syn/selectMobileSyncList.mdo'/>";
    var uri = contextPath + "/mbl/com/syn/selectMobileSyncList.mdo";

    $.ajax({

```

```

        type      : "POST",
        cache     : false,
        url       : uri,
        data      : {fetchRow:fetchRowIdx, deviceType:deviceType},
        dataType  : "json",
        success :function(json) {
                var synList = eval(json.synList);
                var html = "";
                localStorage.clear();
                var seperator = "|";
                for(var k=0; k<synList.length; k++) {
                        var data = "S" + seperator + synList[k].sn + seperator +
synList[k].syncSj + seperator + synList[k].syncCn + seperator + synList[k].creatDt;
                        localStorage.setItem("localSn" + (synList.length - k), data);
                }
                fn_showList();
        }
});
};

```

Screen and execution manual

| Action | URL | Controller method | QueryID |
|--------------------------|---------------------------------------|----------------------|--|
| Move to the inquiry page | /mbl/com/syn/goMobileSynclist.mdo | goMobileSynclist | |
| Inquiry | /mbl/com/syn/selectMobileSynclist.mdo | selectMobileSynclist | "SyncDAO.selectSynclist" |
| View more | /mbl/com/syn/selectMobileSynclist.mdo | selectMobileSynclist | "SyncDAO.selectSynclist" |
| Move to the write page | /mbl/com/syn/goMobileSyncInsert.mdo | goMobileSyncInsert | |
| Move to the detail page | /mbl/com/syn/goMobileSync.mdo | goMobileSync | |
| Execute Synchronization | /mbl/com/syn/executeMobileSync.mdo | executeMobileSync | "SyncDAO.insertSync", "SyncDAO.updateSync", "SyncDAO.deleteSync" |



Synchronization mobile detail inquiry

Business rules

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

Page navigation is totally based on the form type.

Synchronization service detail posts are being retrieved.

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

Deletion is being carried out.

Related codes

```
var data = "D" + seperator + rowData[1] + seperator + rowData[2] + seperator + rowData[3] + seperator +  
rowData[4];
```

```
localStorage.setItem(localStorage.getItem("localSn"), data);
```

Screen and execution manual

| Action | URL | Controller method | QueryID |
|-------------------------------|-------------------------------------|--------------------|----------------------|
| Delete | /mbl/com/syn/deleteMobileSync.mdo | deleteMobileSync | "SyncDAO.deleteSync" |
| Move to the modification page | /mbl/com/syn/goMobileSyncUpdate.mdo | goMobileSyncUpdate | |
| List | /mbl/com/syn/goMobileSyncList.mdo | goMobileSyncList | |



Synchronization mobile registration

Business rules

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

Page navigation is totally based on the form type.

Synchronization service posts are inputted and checked for validity on the mobile device. Posts registered offline should be stored in the webstorage separately.

Related codes

```
var dataLength = localStorage.length + 1;
```

```
var data = "I" + seperator + "0" + seperator + syncSj + seperator + syncCn + seperator + getTimeStamp();
```

Screen and execution manual

| Action | URL | Controller method | QueryID |
|----------|-----------------------------------|-------------------|----------------------|
| Register | /mbl/com/syn/insertMobileSync.mdo | insertMobileSync | "SyncDAO.insertSync" |
| Cancel | | | |
| List | /mbl/com/syn/goMobileSyncList.mdo | goMobileSyncList | |



Synchronization mobile modification

Business rules

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

Page navigation is totally based on the form type.

Synchronization service are modified, checked for validity and stored on the mobile device. Posts modified offline should be stored in the webstorage separately.

For the offline modification, the code is composed considering the offline registration.

Related codes

```
var syncStatus = "U";  
  
if (rowData[0] == "I") {  
    syncStatus = "I";  
}  
  
var data = syncStatus + seperator + rowData[1] + seperator + syncSj + seperator + syncCn + seperator +  
rowData[4];  
  
localStorage.setItem(localStorage.getItem("localSn"), data);
```

Screen and execution manual

| Action | URL | Controller method | QueryID |
|--------|-----------------------------------|-------------------|----------------------|
| Cancel | | | |
| Store | /mbl/com/syn/updateMobileSync.mdo | updateMobileSync | "SyncDAO.updateSync" |
| List | /mbl/com/syn/goMobileSyncList.mdo | | |



Synchronization list inquiry

Business rules

You can retrieve the synchronization service post lists by using search conditions and key words.

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/syn/selectSyncList.mdo | selectSyncList | "SyncDAO.selectSyncList" |
| | | "SyncDAO.selectSyncTotCnt" | |
| Moves to the registrati | /mbl/com/syn/goSyncRegist.mdo | goSynRegist | |



Synchronization detail inquiry

Business rules

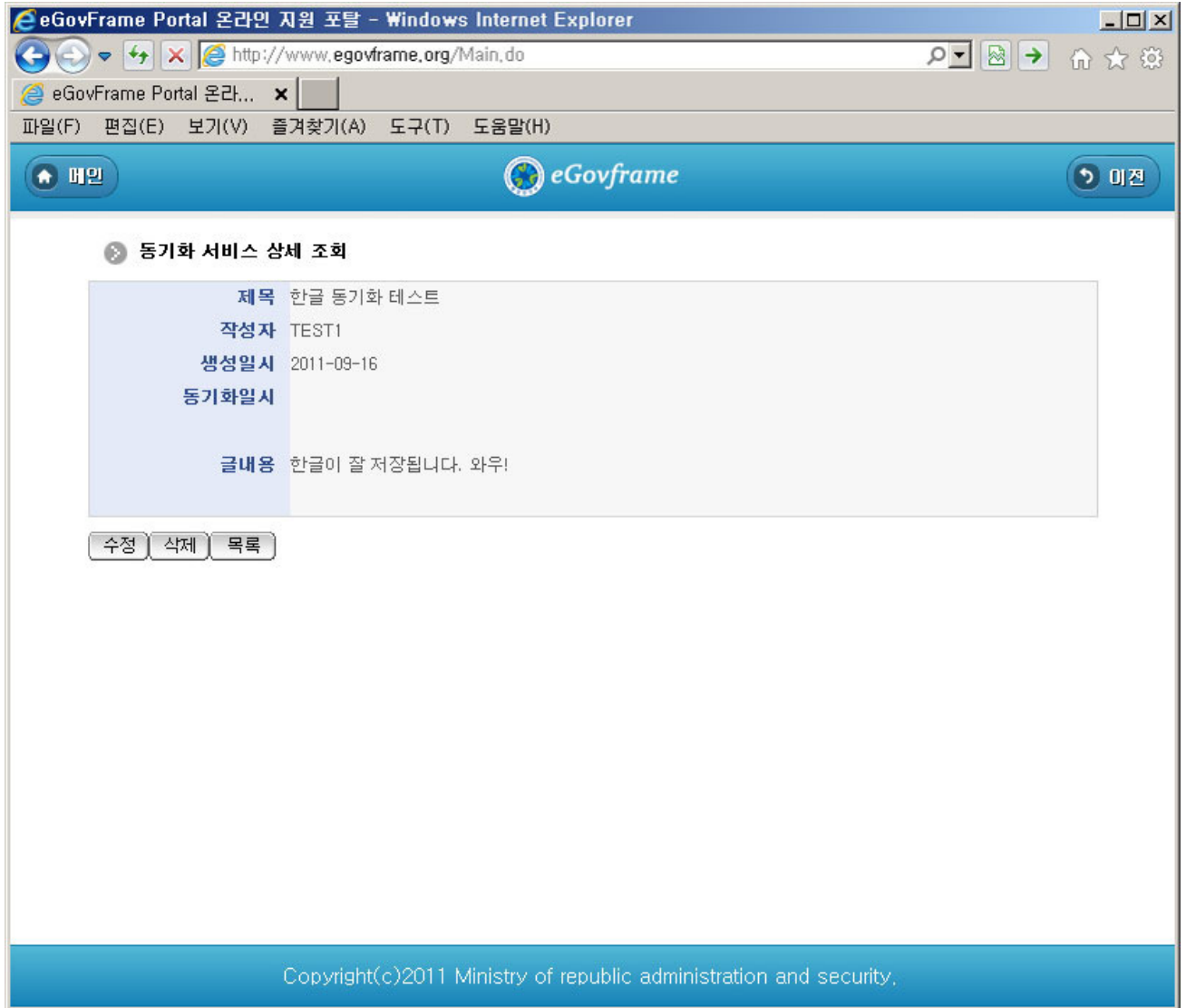
Synchronization service detail posts are being retrieved.

Related codes

N/A

Screen and execution manual

| Action | URL | Controller method | QueryID |
|-------------------------------|---------------------------------|-------------------|--------------------------|
| Delete | /mbl/com/syn/deleteSync.mdo | deleteSync | "SyncDAO.deleteSync" |
| Move to the modification page | /mbl/com/syn/goSyncUpdt.mdo | goSyncUpdt | |
| List | /mbl/com/syn/selectSyncList.mdo | selectSynList | "SyncDAO.selectSyncList" |



Synchronization registration

Business rules

Synchronization 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/syn/insertSync.mdo | insertSync | "SyncDAO.insertSync" |
| Cancel | | | |
| List | /mbl/com/syn/selectSyncList.mdo | selectSynList | "SyncDAO.selectSyncList" |



Synchronization modification

Business rules

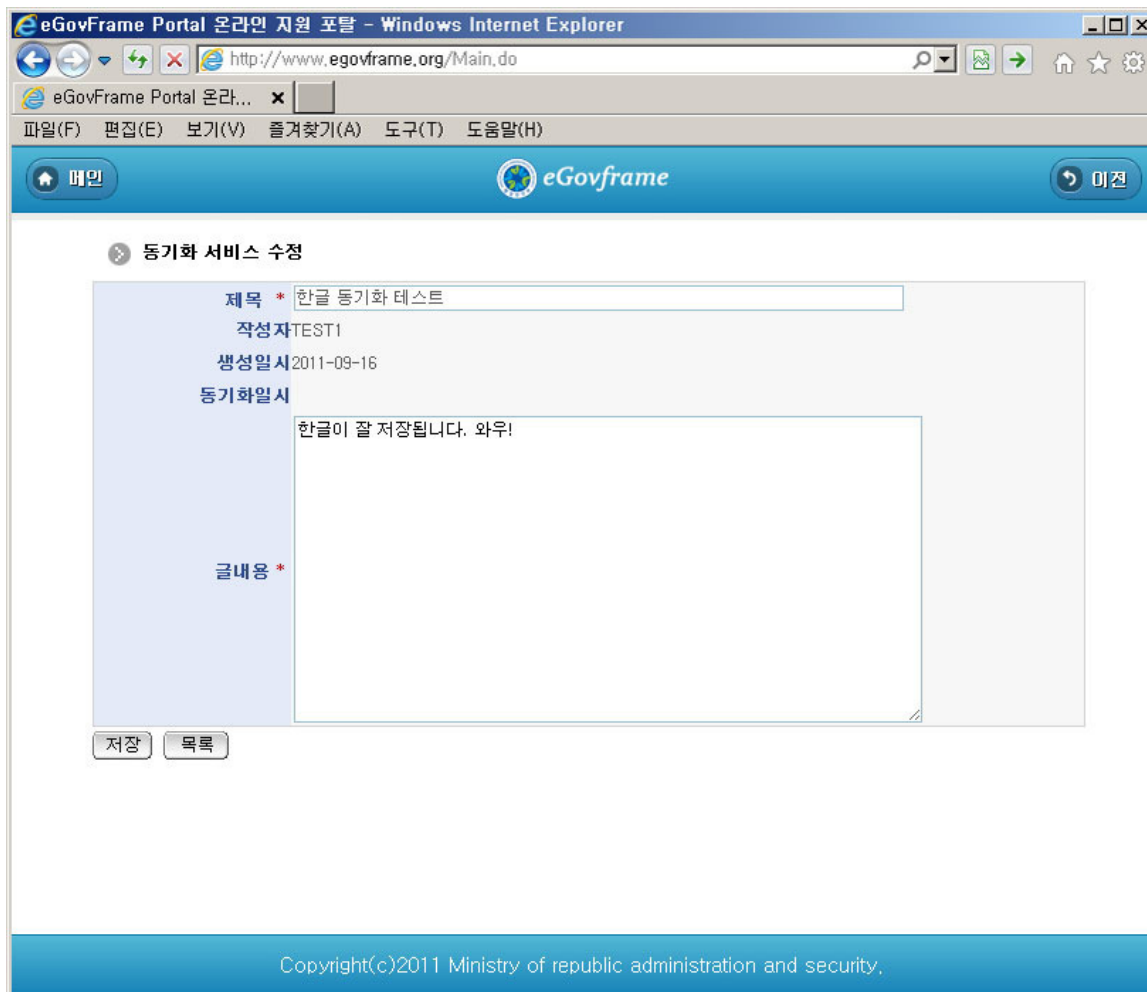
Synchronization service posts are modified and titles and contents are inputted.
The title and content are required items.

Related codes

N/A

Screen and execution manual

| Action | URL | Controller method | QueryID |
|--------|--------------------------------|-------------------|-------------------------|
| Cancel | | | |
| Store | /mbl/com/syn/updateSync.mdo | updateSyn | "SyncDAO.updateSync" |
| List | /mbl/com/syn/selectSynList.mdo | selectSynList | "SyncDAO.selectSynList" |



References

- Distribution and test: [Mobile Common Component System](#) and [distribution package plan](#)
- Related database structure and schema: [related database structure and schema](#)

egovframework/mcom/Synchronizationservice.txt · Last modified: 2011/11/07 19:37 by com

Show pagesource

Old revisions

Login

Index

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