

Contacts Device API Guide Program

Outline

Contacts Device API Guide Program supports JavaScript-based control and checking of Contacts related features of mobile smart device.

Also, it serves as a reference and a tool for implementing "back up Contacts information to server" and "restore to back up Contacts information" features.

Feature

This Guide Program provides **Contacts inquiry, back up Contacts information, and restore to backup Contacts** features.

Preconditions

Category	Description
Local Device Environments	eGovFramework Runtime Environment 3.5, Android SDKAPI 22(version 5.0 Lollipop)
Server-side Developmental Environment	Runtime Environment for eGov Standard Framework 3.5
Works in sync with Mash up Open API	N/A
Test Device	Galaxy S2
Test Platform	Android 2.3
Libraries Added	N/A

Restrictions

New ID is given when Contacts are created or altered.

- Problem: when restoring Contacts using PhoneGap library, new ID is given to restored Contacts.
- Solution: sync new ID to server.

```
function fn_egov_contactIdUpdate(param,isComplete) {
    var url = "/ctt/xml/updateContacts.do";
    var acceptType = "xml";
    var params = {contactsList : {contactsList : param}};

    window.plugins.EgovInterface.post(url, acceptType, params, function(xmldata) {
        console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess request Completed');
        $.mobile.hidePageLoadingMsg('a');
    });
}
```

```

        if($(xmldata).find("resultState").text() == "OK"){
            if(isComplete) {
                jAlert($(xmldata).find("resultMessage").text() +'contact information
synchronized.', 'information', 'c');
            } else {
                jAlert('Number of contacts saved on Server is different from the number of
contacts recovered.\n Please reinitiate recovery.', 'information', 'c');
            }
        }else{
            jAlert($(xmldata).find("resultMessage").text(), 'Error', 'c');
        }
    });
}

```

Supported devices and platforms

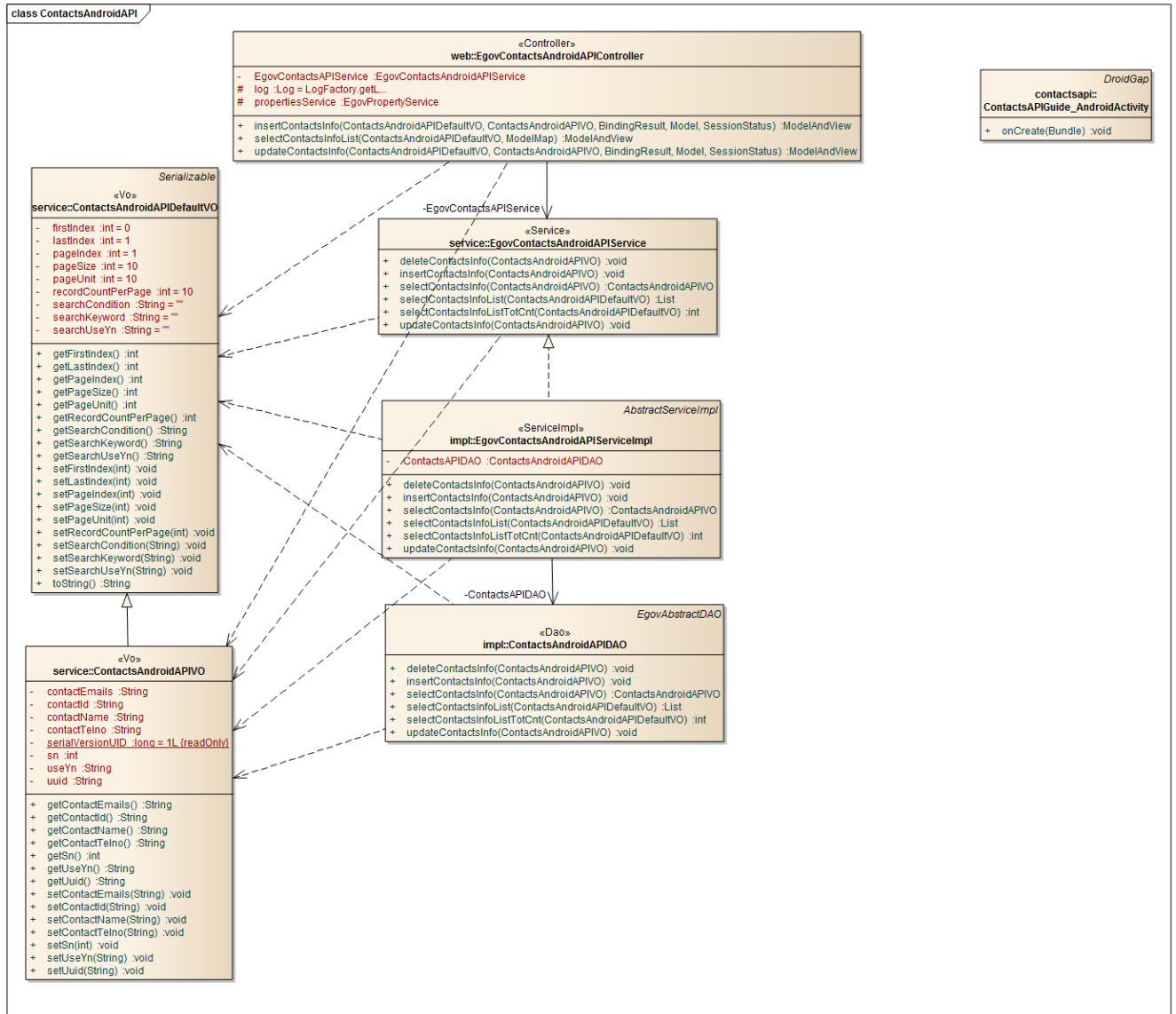
Using cross domain

When using certain outside domains or its subdomains on PhoneGap, add such domains on `<access origin="" />` at `Res/xml/config.xml`.

Description

Contacts Device API Guide Program is comprised of **Contacts inquiry, back up Contacts information, and restore to backup Contacts** features.

Related Class Diagram



Device Application

Source

Type	Target Source	Remark
Activity	kr.go.egovframework.hyb.contacts.ContactsAPIGuide_AndroidActivity	ContactsAPI Guide Program Activity Class
CSS	assets/www/css/egovframework/mbl/hyb/ContactsAPI.css	ContactsAPI Guide Program Core Cascading Style Sheets
IMAGE	assets/www/images/egovframework/mbl/hyb/	ContactsAPI Guide Program Core Image Folder
JS	assets/www/js/egovframework/mbl/hyb/ContactsAPI.js	ContactsAPI Guide Program Core JavaScript

RES	assets/www/res/	ContactsAPI Guide Program Core Resource Folder
XML	AndroidManiFest.xml	Configuration XML for Android
HTML	assets/www/ContactsAPI.html	ContactsAPI Main Page
HTML	assets/www/Intro.html	ContactsAPI Intro Page
HTML	assets/www/license.html	ContactsAPI License Page
HTML	assets/www/overview.html	ContactsAPI Function Description Page

APIs Used

contacts.create

- Create a new contact object

```
var myContact = navigator.contacts.create({"displayName": "Test User"});
contacts.find
```

- Finds a contact

```
navigator.contacts.find(contactFields, contactSuccess, contactError, contactFindOptions);
var options = new ContactFindOptions();
options.filter="Bob";
var fields = ["displayName", "name"];
navigator.contacts.find(fields, onSuccess, onError, options);
```

- Saves a contact

```
// create a new contact object
var contact = navigator.contacts.create();
contact.displayName = "Plumber";
contact.nickname = "Plumber"; //specify both to support all devices
```

```
// populate some fields
var name = new ContactName();
name.givenName = "Jane";
name.familyName = "Doe";
contact.name = name;
```

```
// save to device
contact.save(onSuccess,onError);
```

- Clones a contact

```
var clone = contact.clone();
clone.name.givenName = "John";
console.log("Original contact name = " + contact.name.givenName);
console.log("Cloned contact name = " + clone.name.givenName);
```

- Removes a contact

```

function onSuccess() {
    alert("Removal Success");
};

function onError(contactError) {
    alert("Error = " + contactError.code);
};

// remove the contact from the device
contact.remove(onSuccess,onError);

```

- contactFields

Properties	Object Structure	Remark
ID	string	Global Unique Identifier (DOMString)
displayName	(DOMString)	Display name of a contact
name	(ContactName)	Object that contains the entire components
nickname	(DOMString)	Casual name where contact can be made
phoneNumbers	(ContactField [])	Phone contact field
email	(ContactField [])	E-mail contact field
addresses	(ContactAddress [])	Address field
ims	(ContactField [])	IM Address field
organizations	(ContactOrganization [])	Organization field
birthday	Date	Birth date of the contact (birthdate)
note	(DOMString)	Remark on the contact
photo	(ContactField [])	Photo field
categories	(ContactField [])	Customized array
urls	(ContactField [])	Web page array

- ContactName

Properties	Object Structure	Remark
formatted	(DOMString)	Full name of the contact
familyName	(DOMString)	
givenName	(DOMString)	

middleName (DOMString)

honorificPrefix (DOMString) Honorific prefix

honorificSuffix (DOMString) Honorific suffix

- ContactField

Properties	Object Structure	Remark
type	(DOMString)	Field category arrays
value	(DOMString)	Field value
pref	(Boolean)	Preferred value

- ContactAddress

Properties	Object Structure	Remark
pref	(boolean)	Representative value of ContactAddress
type	(DOMString)	Field types
formatted	(DOMString)	Address information for output
streetAddress	(DOMString)	Street information
locality	(DOMString)	City / County
region	(DOMString)	State / Region
postalCode	(DOMString)	Postal code
country	(DOMString)	Area code

- ContactOrganization

Properties	Object Structure	Remark
pref	(Boolean)	Representative value of ContactAddress
type	(DOMString)	Definition of field types (e.g. 'work')
name	(DOMString)	Name of the affiliation
department	(DOMString)	Department in the affiliation
title	(DOMString)	Title

- ContactFindOption

Properties	Object Structure	Remark
filter	(DOMString)	Defines search terms or conditions.?(Default: "")
multiple	(Boolean)	Loading of multiple contact information

- ContactError

Properties	Object Structure	Remark
ContactError.UNKNOWN_ERROR	(DOMString)	Unknown error
ContactError.INVALID_ARGUMENT_ERROR	(DOMString)	Wrong argument
ContactError.TIMEOUT_ERROR	(DOMString)	Timeout error
ContactError.PENDING_OPERATION_ERROR	(DOMString)	Invalid command
ContactError.IO_ERROR	(DOMString)	I/O error
ContactError.NOT_SUPPORTED_ERROR	(DOMString)	Not supported
ContactError.PERMISSION_DENIED_ERROR	(DOMString)	Permission denied

Server Application

Source

Type	Target Source	Remark
Controller	egovframework.hyb.add.ctt.web.EgovContactsAndroidAPIController.java	Classes for Contacts information administration.
Service	egovframework.hyb.add.ctt.service.EgovContactsAndroidAPIService.java	Service interface for Contacts information administration.
VO	egovframework.hyb.add.ctt.service.ContactsAndroidAPIDefaultVO.java	VO Class for Contacts administration.
VO	egovframework.hyb.add.ctt.service.ContactsAndroidAPIVO.java	VO Class for Contacts administration.
DAO	egovframework.hyb.add.ctt.service.impl.ContactsAndroidAPIDAO.java	Data processing classes for Contacts administration.
DAO	egovframework.hyb.add.ctt.service.impl.EgovContactsAndroidAPIServiceImpl.java	Class for processing services requested.
Query ML	X resources/egovframework/sqlmap/hyb/add/ctt/EgovContactsAndroidAPIGuide_SQL_mysql.xml	Query file for Contacts administration.

Related Tables

Title	Table	Remark
-------	-------	--------

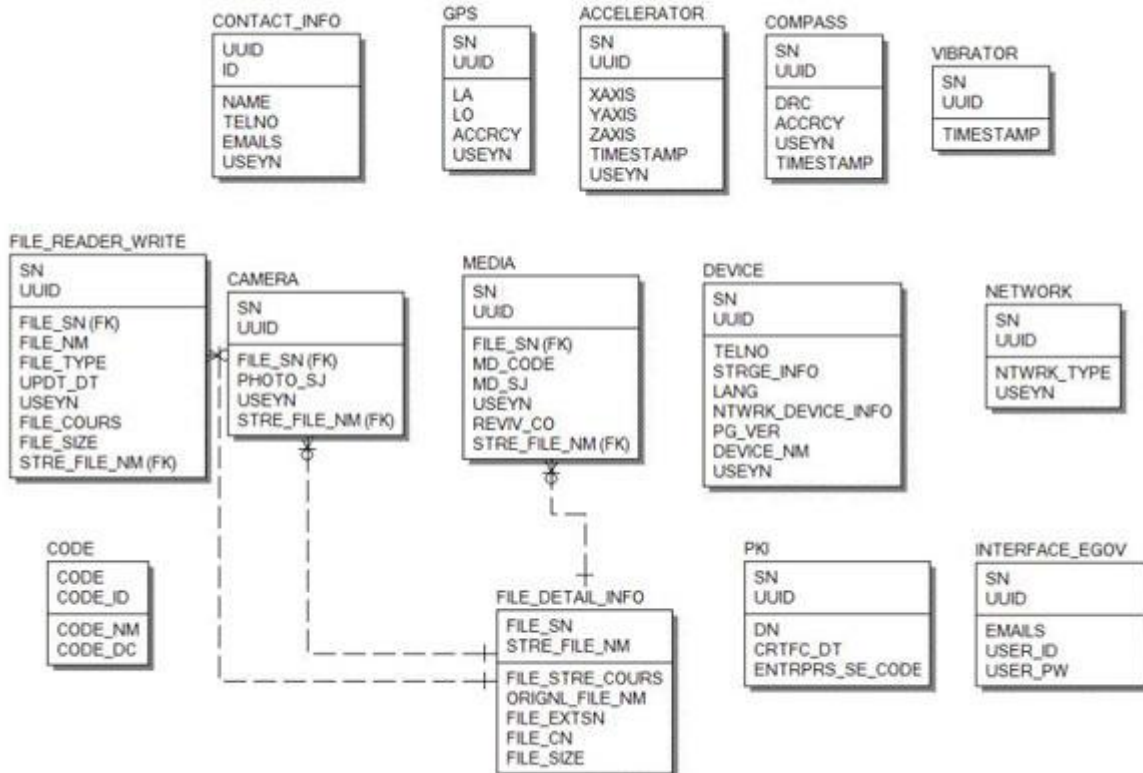
CONTACT	CONTACT_INFO	Contacts information administration.
---------	--------------	--------------------------------------

Table Definition

- CONTACT

No.	Column	Title of Column	Type	Length	Null	KEY
1	ID	Serial No.	VARCHAR	20	NotNull	pk
2	UUID	UUID	VARCHAR	50	NotNull	pk
3	NAME	Title	VARCHAR	60	Null	
4	TELNO	Phone contact	VARCHAR	20	Null	
5	EMAILS	Email	VARCHAR	50	Null	
6	USEYN	Activation	CHAR	1	Null	

ERD



Configuration

Contacts Device properties required for use of GPS-related functions of Contacts Device API Guide Program are as follows:

Device Application

res/xml/plugins.xml

```
<plugins>
<!-- PhoneGap Plugin for eGov Interface Device API Class -->
<pluginname="EgovInterfacePlugin" value="kr.go.egovframework.hyb.plugin.EgovInterfacePlugin"/>
<!-- Phonegap Plugin Class for eGov Contacts Device API -->
<pluginname="Contacts" value="org.apache.cordova.ContactManager"/>
</plugins>
```

res/values/serverinfo.xml

```
<!-- Server Directory for eGov Interface Device API Class -->
<?xmlversion="1.0"encoding="utf-8"?>
<resources>
  <stringname="SERVER_URL">http://192.168.100.222:8080/DeviceAPIGuideTotal_Web_V1.7.1</string>
</resources>
```

AndroidManifest.xml

```
<uses-permissionandroid:name="android.permission.INTERNET"/>
<uses-permissionandroid:name="android.permission.READ_CONTACTS"/>
<uses-permissionandroid:name="android.permission.WRITE_CONTACTS"/>
<uses-permissionandroid:name="android.permission.ACCESS_NETWORK_STATE"/>
```

Server Application

resource/egovframework/sqlmap/sql-map-config_[DB NAME].xml

```
<sqlMapresource="egovframework/sqlmap/hyb/add/dvc/EgovContactsAndroidAPIGuide_SQL_[DB NAME].xml"/>
```

Related Features

Contacts Device API Guide Program is comprised of **Contacts inquiry**, **back up Contacts information**, and **restore to backup Contacts** features.

Contacts inquiry

Business Logic

1. Inquires name, phone(mobile) number, and email of all Contacts saved in mobile device.

Related Codes

Inquires a Contact

```
function fn_egov_move_localContactList(){
  $.mobile.showPageLoadingMsg('a');
  var fields = ["displayName", "name", "phoneNumbers", "emails"];
```

```

navigator.contacts.find(fields, fn_egov_contactListSuccess, fn_egov_fail);
}
    Contact inquiry successful

function fn_egov_contactListSuccess(contacts) {
    console.log('DeviceAPIGuide fn_egov_contactListSuccess Success');
    var theList = $('#listView');
    theList.html("");
    contactsList = contacts;
    nPageCount = 0;
    fn_egov_showList(true);

    $.mobile.hidePageLoadingMsg('a');
    $.mobile.changePage("#contactList", "slide", false, false);

    theList.listview("refresh");

    setTimeout(loadScrollList, 1000);
}
    Detailed inquiry of certain Contact

function fn_egov_move_localContactDetail(index)
{
    var displayName = "";
    var phoneNumber = "";
    var email = "";

    displayName = contactsList[index].displayName;

    if(contactsList[index].phoneNumbers != null)
    {
        for(var j=0; j<contactsList[index].phoneNumbers.length; j++)
        {
            if(contactsList[index].phoneNumbers[j].type == "mobile")
            {
                phoneNumber = contactsList[index].phoneNumbers[j].value;
                break;
            }
        }
    }
    if(contactsList[index].emails != null)
    {
        for(var j=0; j<contactsList[index].emails.length; j++)
        {
            email = contactsList[index].emails[j].value;
            break;
        }
    }

    document.getElementById('detailIndex').value = index;
    document.getElementById('detailName').value = (displayName==null)?'noname ':displayName;
    document.getElementById('detailPhone').value = phoneNumber;
    document.getElementById('detailEmail').value = email;
}

```

```

    $.mobile.changePage("#contactDetail", "slide", false, false);
}
    Delete Contact

```

```

function fn_egov_contactLocalDelete()
{
    var index = $("#detailIndex").val();
    contactsList[index].remove(fn_egov_removeSuccess,fn_egov_fail);
}

```

Related Screen and Implementation Manual

Function	_URL_ Controller	method	Display (HTML)
Inquire contact list		fn_egov_contactListSuccess	ContactsAPI.html#contactList
Inquire contact details		fn_egov_move_localContactDetail	ContactsAPI.html#contactDetail
Delete Contact		fn_egov_contactLocalDelete	ContactsAPI.html#contactDetail





List: calls Contact details inquiry.
+ 10 more: displays 10 more Contact lists.
Delete: deletes a Contact.

Back up Contacts information

Business Logic

1. Check the network. If the network is not Wi-Fi, function based on user approval.
2. When backing up Contacts, update Contacts with same ID and adds Contacts with new ID.

Related Codes

Request total number of Contacts backed up in server.

```
function fn_egov_contactBackupCountSuccess(contacts)
{
```

```

console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess Success');
contactsList = contacts;

var url = "/ctt/xml/selectBackupCount.do";
var acceptType = "xml";

var params = {uuid : device.uuid};

// get the data from server
window.plugins.EgovInterface.post(url, acceptType, params, function(xmldata) {
    console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess request Completed');
    $.mobile.hidePageLoadingMsg('a');
    if($(xmldata).find("resultState").text() == "OK"){

        var html = '<div class="ui-block-a"><label align="center"><h3>' +
String(contactsList.length) + '</h3></label></div>';
        html += '<div class="ui-block-b"><label align="center"><h3>' +
$(xmldata).find("totCount").text() + '</h3></label></div>';

        var theList = $('#deviceBackupCount');
        theList.html(html);

        $.mobile.changePage("#contactBackup", "slide", false, false);
    }else{
        jAlert($(xmldata).find("resultMessage").text(), 'Error', 'c');
    }
});
}

    Backs up Contacts information to server.

function fn_egov_contactBackup()
{
    // displays the warning message that data charges will be incurred when using 3G.
    if(!fn_egov_network_check(false)) {
        return;
    }

    if(contactsList.length == 0) {
        return;
    }

    $.mobile.showPageLoadingMsg('a');

    var param = [];
    for(var index = 0; index < contactsList.length ; index++) {
        var displayName = "";
        var phoneNumber = "";
        var email = "";

        if(contactsList[index].displayName != null) {
            displayName = contactsList[index].displayName;
        }

        if(contactsList[index].phoneNumbers != null) {

```

```

        for(var j=0; j<contactsList[index].phoneNumbers.length; j++) {
            if(contactsList[index].phoneNumbers[j].type == "mobile"){
                phoneNumber = contactsList[index].phoneNumbers[j].value;
                break;
            }
        }
    }
    if(contactsList[index].emails != null) {
        for(var j=0; j<contactsList[index].emails.length; j++) {
            email = contactsList[index].emails[j].value;
            break;
        }
    }
    param.push({ uuid:device.uuid, contactId:contactsList[index].id, name:displayName,
telNo:phoneNumber, emails:email, useYn:'Y'});
}

var url = "/ctt/xml/addContactsInfo.do";
var acceptType = "xml";
var params = {contactsList : {contactsList : param}};

window.plugins.EgovInterface.post(url, acceptType, params, function(xmldata) {
    console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess request Completed');
    $.mobile.hidePageLoadingMsg('a');
    if($(xmldata).find("resultState").text() == "OK"){
        jAlert($(xmldata).find("resultMessage").text(), 'information', 'c');

        var html = '<div class="ui-block-a"><label align="center"><h3>' +
String(contactsList.length) + '</h3></label></div>';
        html += '<div class="ui-block-b"><label align="center"><h3>' +
$(xmldata).find("totCount").text() + '</h3></label></div>';

        var theList = $('#deviceBackupCount');
        theList.html(html);
    }else{
        jAlert($(xmldata).find("resultMessage").text(), 'Error', 'c');
    }
});
}

```

Related Screen and Implementation Manual

Function	_URL_	Controller	method	Display (HTML)
Inquires total number of backed up Contacts	/ctt/xml/selectBackupCount.do	EgovContactsAndroidAPIController	selectContactsCount	ContactsAPI.html#contactBackup
Contact backup request	/ctt/xml/addContactsInfo.do	EgovContactsAndroidAPIController	addContactsInfoXml	ContactsAPI.html#contactBackup



Back up Contacts: backs up contacts saved in device to server.

Restore to backup Contacts

Business Logic

1. Check the network. If the network is not Wi-Fi, function based on user approval.
2. When restoring Contacts, delete Contacts in device if ID of Contacts in device and backed up Contacts are the same, and restores to backed up Contacts.

Related Codes

Request total number of Contacts backed up in server.

```
function fn_egov_contactRecoveryCountSuccess(contacts) {
    console.log('DeviceAPIGuide fn_egov_contactRecoveryCountSuccess Success');
    contactsList = contacts;
}
```



```

var url = "/ctt/xml/selectBackupCount.do";
var acceptType = "xml";
var params = {uuid : device.uuid};

// get the data from server
window.plugins.EgovInterface.post(url, acceptType, params, function(xmldata) {
    console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess request Completed');
    $.mobile.hidePageLoadingMsg('a');
    if($(xmldata).find("resultState").text() == "OK"){
        var html = '<div class="ui-block-a"><label align="center"><h3>' +
String(contactsList.length) + '</h3></label></div>';
        html += '<div class="ui-block-b"><label align="center"><h3>' +
$(xmldata).find("totCount").text() + '</h3></label></div>';

        var theList = $('#deviceRecoveryCount');
        theList.html(html);

        $.mobile.changePage("#contactRecovery", "slide", false, false);
    } else {
        jAlert($(xmldata).find("resultMessage").text(), 'Error', 'c');
    }
});
}

Restore to backup Contacts

function contactRecoveryCallBack(ContactData)
{
    var id = ContactData["id"];
    var name = ContactData["name"];
    var telNo = ContactData["telNo"];
    var email = ContactData["emails"];

    var options = new ContactFindOptions();
    options.filter=String(id);
    var fields = ["id"];
    navigator.contacts.find(fields,
        function(contacts)
        {
            if(contacts.length != 0)
            {
                contacts[0].remove(function(contact){ },fn_egov_fail);
            }

            var contactname = new ContactName();
            contactname.middleName = name;

            var myContact =
navigator.contacts.create({"name":contactname});

            var phoneNumbers = [2];
            phoneNumbers[0] = new ContactField('work', telNo, false);
            phoneNumbers[1] = new ContactField('mobile', telNo, false);
            phoneNumbers[2] = new ContactField('home', telNo, false);

```

```

        myContact.phoneNumbers = phoneNumbers;

        var emails = [2];
        emails[0] = new ContactField('work', email, false);
        emails[1] = new ContactField('home', email, false);
        myContact.emails = emails;

        myContact.save(function(contact)
            {

recoveryParam.push({ uuid:device.uuid, contactId:id, newId:contact.id});
                                                                    if(listCount ===
recoveryParam.length)
                                                                    {

$.mobile.hidePageLoadingMsg('a');
                                                                    confirm(String(listCount) +
'contacts recovered. Synchronizing saved contact information with server. ');

$.mobile.showPageLoadingMsg('a');
                                                                    fn_egov_contactIdUpdate();
                                                                    }
                                                                    },fn_egov_fail);

            }, function(contactError){ }, options);
    }

    Sync restored Contacts to server

function fn_egov_contactIdUpdate(param,isComplete) {
    var url = "/ctt/xml/updateContacts.do";
    var acceptType = "xml";
    var params = {contactsList : {contactsList : param}};

    window.plugins.EgovInterface.post(url, acceptType, params, function(xmldata) {
        console.log('DeviceAPIGuide fn_egov_contactBackupCountSuccess request Completed');
        $.mobile.hidePageLoadingMsg('a');
        if($(xmldata).find("resultState").text() == "OK"){
            if(isComplete) {
                jAlert($(xmldata).find("resultMessage").text() +'contacts synchronized.',
'information', 'c');
            } else {
                jAlert('Number of contacts saved on Server and number of contacts recovered are
different..\nRe-initiate Recovery.', 'information', 'c');
            }

            var fields = ["displayName", "name", "phoneNumbers", "emails"];
            navigator.contacts.find(fields, function(contacts){
                contactsList = contacts;
                var html = '<div class="ui-block-a"><label align="center"><h3>' +
String(contactsList.length) + '</h3></label></div>';
                html += '<div class="ui-block-b"><label align="center"><h3>' +
$(xmldata).find("totCount").text() + '</h3></label></div>';

                var theList = $('#deviceRecoveryCount');

```

```

        theList.html(html);
    }, fn_egov_fail);

    }else{
        jAlert($(xmldata).find("resultMessage").text(), 'Error', 'c');
    }
});
}

```

Related Screen and Implementation Manual

Function	_URL_	Controller	method	Display (HTML)
Inquires total number of backed up Contacts	/ctt/xml/selectBackupCount.do	EgovContactsAndroidAPIController	selectContactsCount	ContactsAPI.html#contactBackup
Restore to backup Contacts	/ctt/xml/contactsInfoList.do	EgovContactsAndroidAPIController	selectContactsInfoListXml	ContactsAPI.html#contactRecovery
Sync restored Contacts to server	/ctt/xml/updateContacts.do	EgovContactsAndroidAPIController	updateContactsXml	ContactsAPI.html#contactRecovery



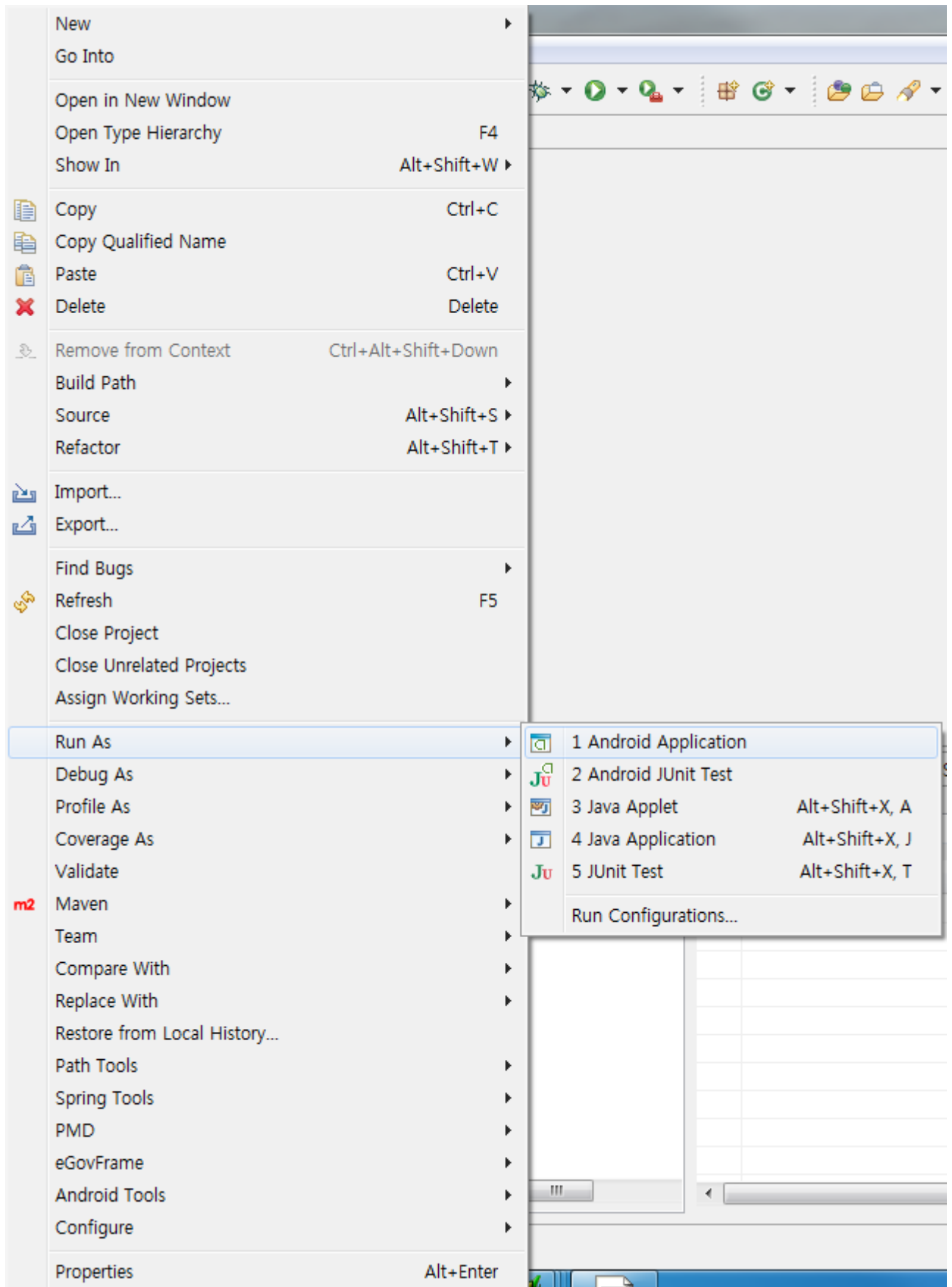
Restore Contacts: restores with Contacts backed up in server.

Compiling, debugging, distributing

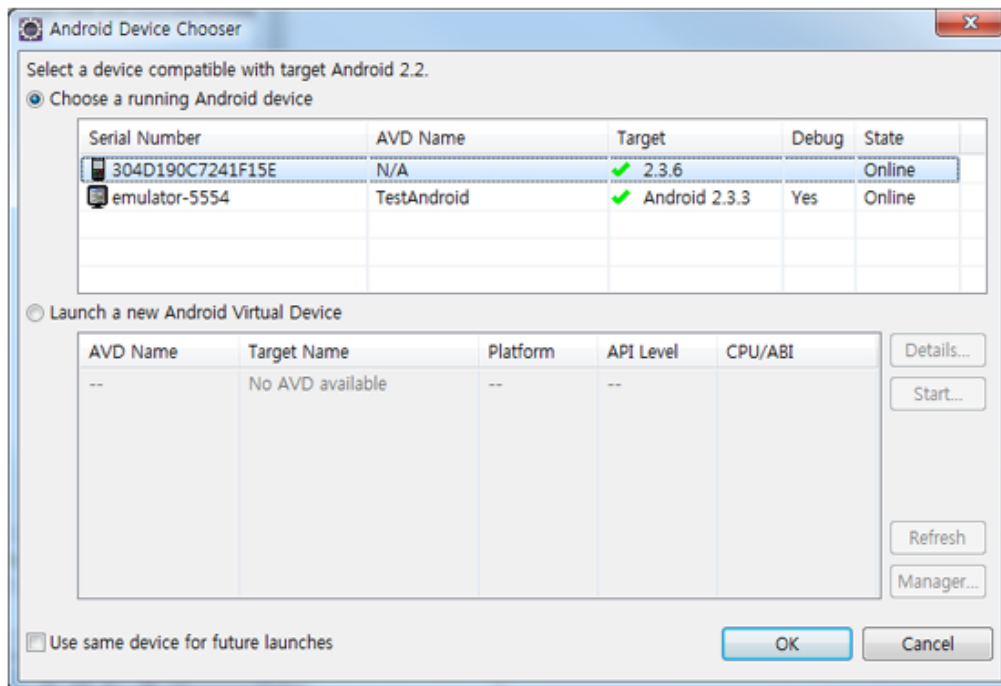
Compiling

How to compile ContactsAPI Device Application

- Right-click on the Device API Guide(Android) project, and click on the "Android Application" at the "Run As" tab. The guide program will be built and installed into the Android device.



- When “Android Device Chooser” window appears, select appropriate device and click on the "OK" button.



- Program display on the emulator



- Program display on the device



Contacts API 디바이스
가이드 프로그램 

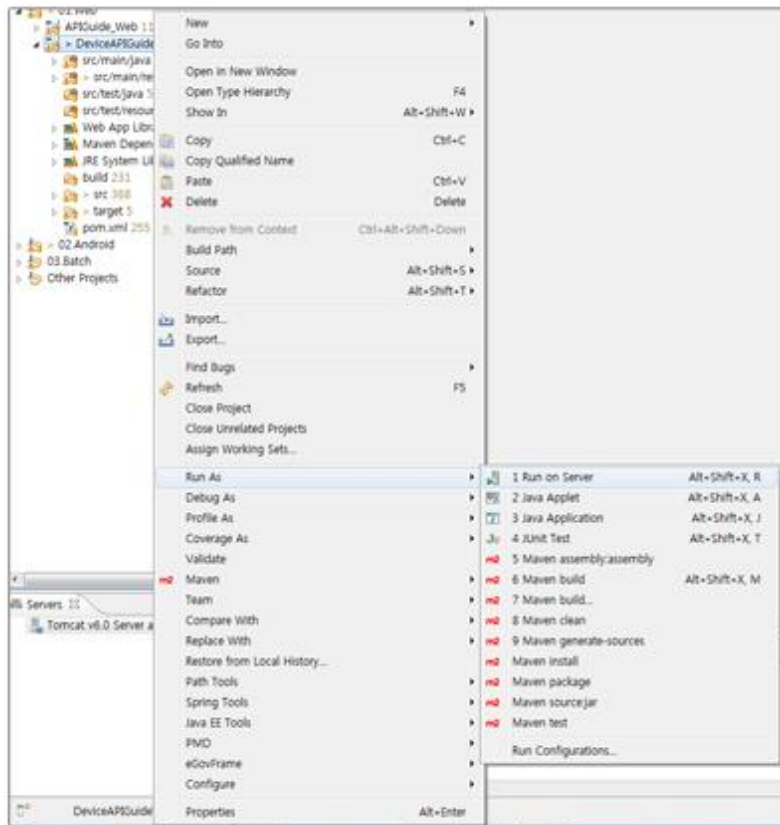


행정안전부



How to compile ContactsAPI Server Applicaton

- Right-click on the project and click on Run As>Run On Server in order to run the ContactsAPI server-side Guide Program.



- When the build is successfully completed, a message reading 'Server Startup in xxx ms' will display on the console window on the Eclipse.

```

2012-09-14 09:15:49,759 DEBUG [org.springframework.beans.factory.support.DefaultListableBeanFactory] Returning cached instance of singleton bean 'org.springframework.web.serv
2012-09-14 09:15:49,767 DEBUG [org.springframework.beans.factory.support.DefaultListableBeanFactory] Returning cached instance of singleton bean 'org.springframework.web.serv
2012-09-14 09:15:49,768 DEBUG [org.springframework.beans.factory.support.DefaultListableBeanFactory] Creating instance of bean 'org.springframework.web.servlet.view.DefaultRe
2012-09-14 09:15:49,771 DEBUG [org.springframework.beans.factory.support.DefaultListableBeanFactory] Finished creating instance of bean 'org.springframework.web.servlet.view
2012-09-14 09:15:49,771 DEBUG [org.springframework.web.servlet.DispatcherServlet] Unable to locate RequestToViewNameTranslator with name 'viewNameTranslator': using default
2012-09-14 09:15:49,771 DEBUG [org.springframework.beans.factory.support.DefaultListableBeanFactory] Returning cached instance of singleton bean 'org.springframework.web.serv
2012-09-14 09:15:49,772 DEBUG [org.springframework.web.servlet.DispatcherServlet] Published WebApplicationContext of servlet 'action' as ServletContext attribute with name [o
2012-09-14 09:15:49,772 INFO [org.springframework.web.servlet.DispatcherServlet] FrameworkServlet 'action': initialization completed in 1373 ms
2012-09-14 09:15:49,772 DEBUG [org.springframework.web.servlet.DispatcherServlet] Servlet 'action' configured successfully
2012. 9. 14 오전 9:15:49 org.apache.coyote.http11.Http11Protocol start
정보: Starting Coyote HTTP/1.1 on http-80
2012. 9. 14 오전 9:15:49 org.apache.jk.common.ChannelSocket init
정보: JK: ajp13 listening on /0.0.0.0:8009
2012. 9. 14 오전 9:15:49 org.apache.jk.server.JKMain start
정보: Jk running ID=0 time=0/30 config=null
2012. 9. 14 오전 9:15:49 org.apache.catalina.startup.Catalina start
정보: Server startup in 7209 ms

```

Debugging

Use console.log in order to check the details on any errors on the device application, and to conduct debugging. Debug codes in console.log are available in JavaScript syntaxes that you can use in Eclipse.

See the following for how to code console.log:

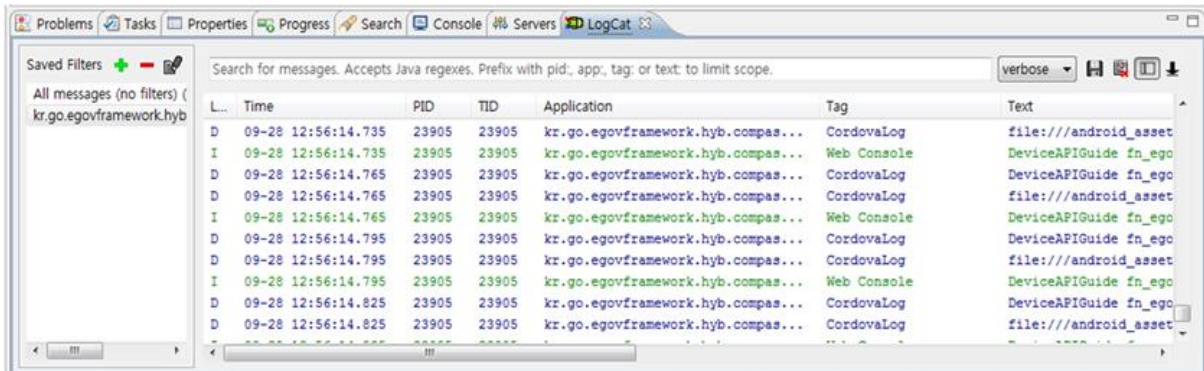
```

function fn_egov_contactListSuccess(contacts) {
    console.log('DeviceAPISGuide fn_egov_contactListSuccess Success');
    var theList = $('#listView');
    theList.html("");
    ...
}

```

}

When the debugging code is executed, check out the following console message appears:



Contacts device API Guide Program will output the following console information for debugging.

Debug code	Debug information
DeviceAPIGuide fn_egov_contactListSuccess	Contacts list in device inquiry successful
DeviceAPIGuide fn_egov_fail + errorCode	Contacts Device API failed due to [error code].
DeviceAPIGuide fn_egov_removeSuccess	Delete Contact successful
DeviceAPIGuide fn_egov_sendto_server Response Completed	Request to server successfully processed
DeviceAPIGuide fn_egov_sendto_server Response Failed	Internal server process error
DeviceAPIGuide fn_egov_sendto_server Request Failed	Communications error
DeviceAPIGuide fn_egov_contactBackupCountSuccess	Successfully inquired the number of Contacts in device
DeviceAPIGuide fn_egov_contactRecoveryCountSuccess	Successfully inquired Contacts before restoration
DeviceAPIGuide EgovInterface.get request Fail	HTTP GET Method request failed
DeviceAPIGuide EgovInterface.post request Fail	HTTP POST Method request failed
DeviceAPIGuide EgovInterface.geturl Fail	geturl function request failed

Distribution

Download Contacts Device API Guide: [Click](#)

References

- UX/UI Library : jQuery Mobile [Click](#)

Phonegap 4.3.0 : [Click](#)