

Camera Device API Guide Program

Outline

Compass guide program is a guide application for eGov Device API, using the mobile device API framework to be used as a tool and a reference when developing hybrid applications. It supports the inquiry of Compass related functions of mobile smart devices through JavaScript-based Compass DeviceAPI. Also, it connects with web server applications based on eGov standard framework to send necessary information to server and inquire them, using photos taken with device's built-in camera and images saved in device's memory.

Feature

This Guide Program provides **send photographs taken with built-in camera to server, send images from the gallery to server, and receive image list from server** features.

Preconditions

Category	Description
Local Device Environments	Xcode 6.3.2, PhoneGap 4.3.0
Server-side Developmental Environment	eGov Standard Framework Develeopment Environment 3.5
Works in sync with Mash up Open API	N/A
Test Device	iPhone4, iPhone6
Test Platform	iOS 7.1.2, iOS 8.3
Libraries Added	N/A

Restrictions

Supported devices and platforms

For iPhone devices, there may be issues due to device's processing power.

- Problem: PhoneGap error.
- Solution: delay PhoneGap loading sequence with setTimeout() function.

```
document.addEventListener('DOMContentLoaded', function () { setTimeout(loaded, 200); }, false);
```

- Problem: iScroll5 content height calculation error.
- Solution: use setTimeout() to ensure iscroll is generated after css application to contents is complete.

```
setTimeout(function()
```

```

{
    myScroll = new iScroll(thisPage,
        {
            checkDOMChanges: true,
            onBeforeScrollStart:function(e)
            {
            }
        });
    },
    500);

```

Problems may occur if alert() is included in CallBack function. (phoneGap)

- Problem: error calling alert() message from CallBack function saved in PhoneGap.
- Solution: Use asynchronous function or avoid using functions that use thread like alert().

Using cross domain

When using certain outside domains or its subdomains on PhoneGap, add such domains on <access origin="ExternalHosts"/> at [Project_Name]/Supproting Files/config.xml

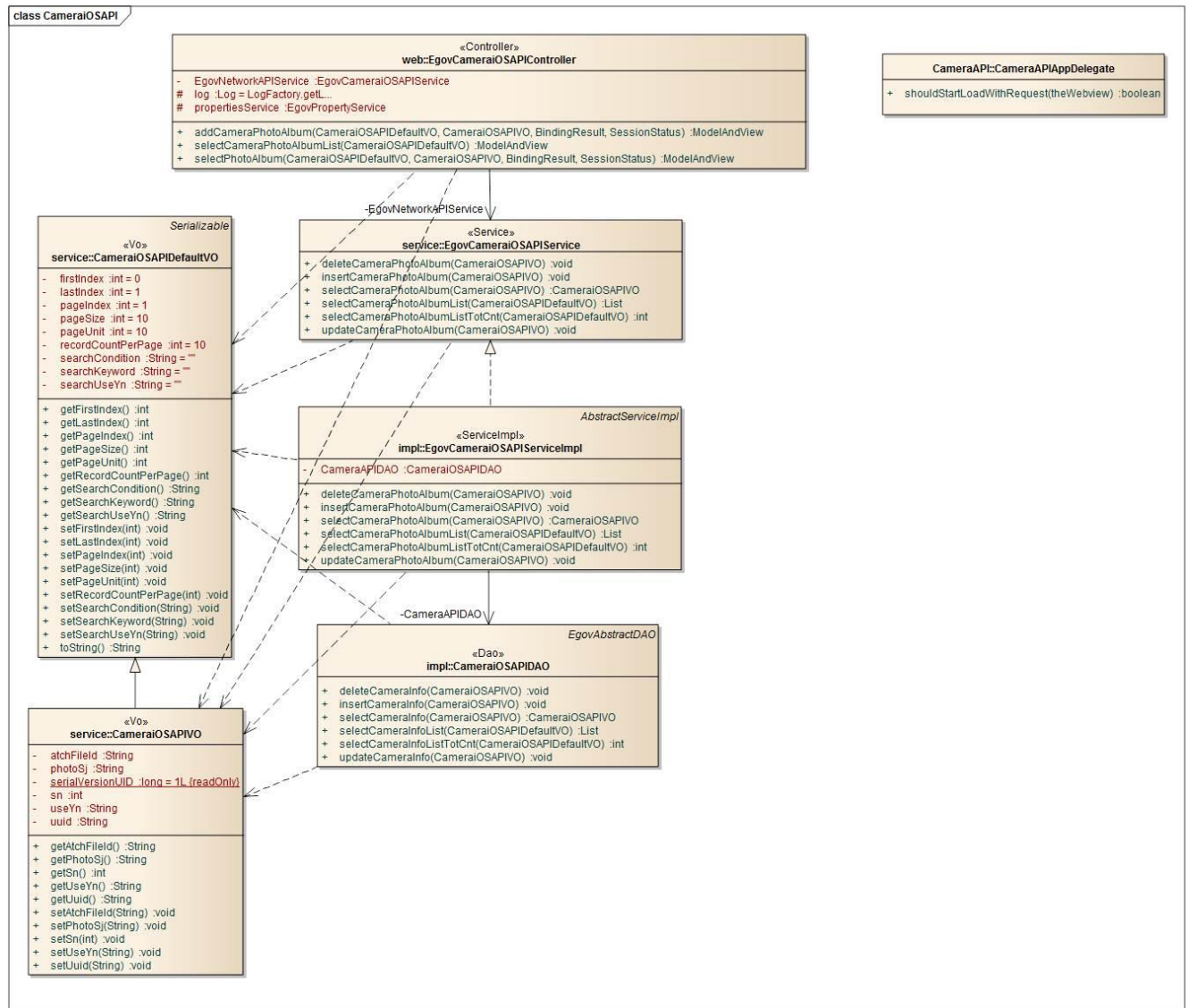
License

N/A

Description

Camera Device API Guide Program consists of send photographs taken with built-in camera to server, send images from the gallery to server, and receive image list from server features. (refer to related functions section)

Related Class Diagram



Device Application

Source

Type	Target	Source	Remark
CSS	www/css/egovframework/mbil/hyb/CameraAPI.css	CameraAPI Guide Program main	Cascading Style Sheets
IMAGE	www/images/egovframework/mbil/hyb/	CameraAPI Guide Program main	Image folder
JS	www/js/egovframework/mbil/hyb/CameraAPI.js	CameraAPI Guide Program main	JavaScript
HTML	www/CameraAPI.html	CameraAPI	main page
HTML	www/license.html	CameraAPI	license page
HTML	www/overview.html	CameraAPI	feature description page

APIs Used

camera.getPicture

- Loads picture shot or in album
- Loads images encoded in base64 or in album

```
navigator.camera.getPicture( cameraSuccess, cameraError, [ cameraOptions ] );
                           cameraOption
```

```
var cameraOption = { };
cameraOption = {
    quality: 50,
    destinationType : Camera.DestinationType.FILE_URI,
    sourceType : Camera.PictureSourceType.CAMERA,
    targetWidth: 200,
    targetHeight: 200
};
```

Option	Description	Remark
quality	Defines quality (resolution) of image in percentage terms	
destinationType	Defines type of the destination value	navigator.camera.DestinationType
sourceType	Configures type of source for photo library and picture shot	navigator.camera.PictureSourceType
allowEdit	Defines editability of the pre-selected image	
encodingType	Defines encoding type	navigator.camera.EncodingType
targetWidth	Configures target width (pixel) of image Rate fixed	
targetHeight	Configures target height (pixel) of image Rate fixed	
mediaType	Configures type of media when pictureSourceType is configured either for PHOTOLIBRARY or SAVEDPHOTOALBUM	navigator.camera.MediaType
correctOrientation	Orientation of image for the images being loaded	
saveToPhotoAlbum	Saving image for the storage in the device	

FileTransfer [upload]

- File transfer object that enables file upload to server.

```
var options = new FileUploadOptions();
options.fileKey="file";
options.fileName=fileURI.substr(fileURI.lastIndexOf('/')+1);
options.mimeType="text/plain";
```

```
var params = new Object();
params.value1 = "test";
params.value2 = "param";
```

```
options.params = params;
```

```
var ft = new FileTransfer();
ft.upload(fileURI, "http://some.server.com/upload.php", transferSuccess, transferFail, options);
    FileUploadOptions
```

- Upload options object that defines transfer parameters for file uploads

Option	Description	Remark
fileKey	Name of the file to be uploaded	Default is "file."
fileName	Name of the file to be saved on server	Default is "image.jpg."
contentType	Sets contentType when uploading data.	Default is "image/jpeg."
params	Additional parameters to be sent to server.	
chunkedMode	Whether or not to use large streaming.	Default is "true."

Server Application

Source

Type	Target Source	Remark
Controller	egovframework.hyb.ios.cmr.web.EgovCameraiOSAPIController.java	Classes for image information administration.
Service	egovframework.hyb.ios.cmr.service.EgovCameraiOSAPIService.java	Service interface for image information administration.
VO	egovframework.hyb.ios.cmr.service.CameraiOSAPIDefaultVO.java	VO Class for image administration.
VO	egovframework.hyb.ios.cmr.service.CameraiOSAPIFileVO.java	VO Class for image file administration
VO	egovframework.hyb.ios.cmr.service.CameraiOSAPIVO.java	VO Class for image administration.
VO	egovframework.hyb.ios.cmr.service.CameraiOSAPIXMLVO.java	VO Class for image administration.
DAO	egovframework.hyb.ios.cmr.service.impl.CameraiOSAPIDAO.java	Data processing Class for image administration.
DAO	egovframework.hyb.ios.cmr.service.impl.EgovCameraiOSMngUtil.java	Classes for image file administration
DAO	egovframework.hyb.ios.cmr.service.impl.EgovCameraiOSAPIServiceImpl.java	Class for processing services requested.

Query X resources/egovframework/sqlmap/com/cop/adb/EgovNcrd_SQ Query file for Contacts administration
ML L.xml

ID Generation

- In order to use ID Generation Service, FILE (FILE information ID) entry must be added to COMTECOPSEQ, a sequence saving table.

Related Tables

Title

Title	Table	Remark
Camera	CAMERA	Manage image information
File details	FILE_DETAIL_INFO	Manage file's detailed information

Tables Breakdown

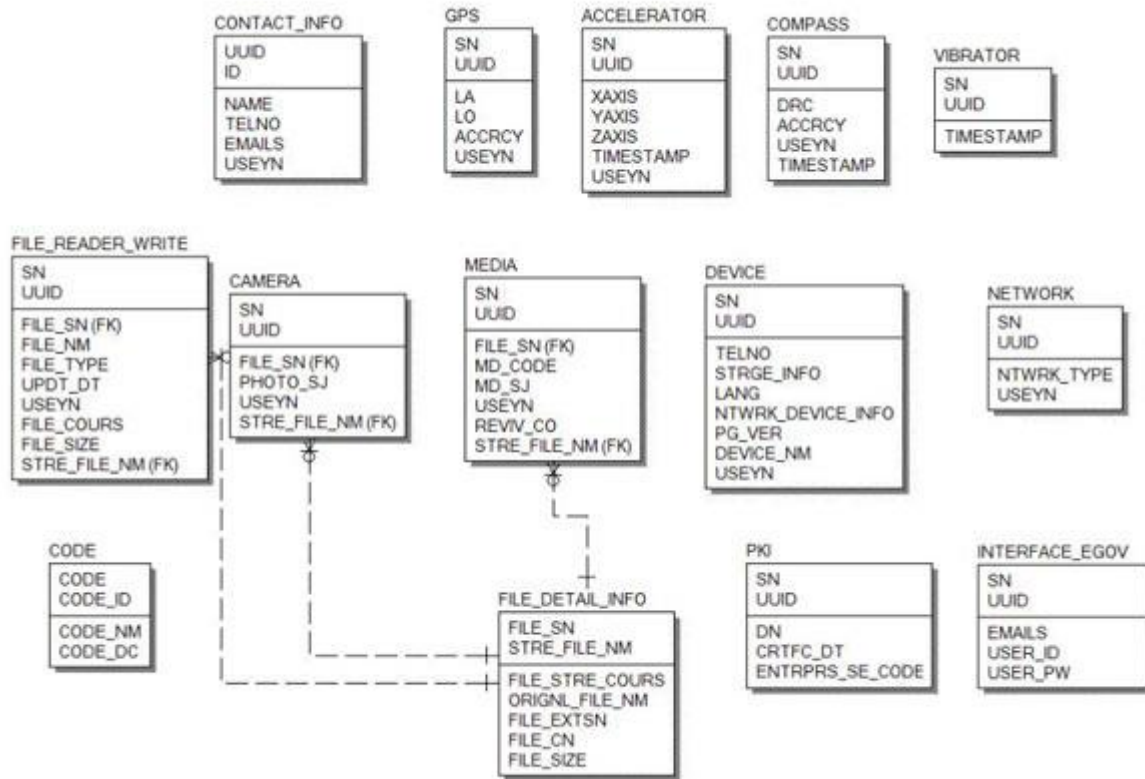
Camera

No.	Column	Title of Column	Type	Length	Null	KEY
1	SN	Serial No.	NUMERIC	6	NotNull	pk
2	UUID	UUID	VARCHAR	50	NotNull	pk
3	PHOTO_SJ	Title of photo	VARCHAR	255	Null	
4	FILE_SN	File Serial Number	NUMERIC	20	NotNull	Fk
5	USEYN	Activation	CHAR	1	Null	pk

File details

No.	Column	Title of Column	Type	Length	Null	KEY
1	FILE_SN	File Serial Number	NUMERIC	20	NotNull	pk
2	FILE_STRE_COURS	File directory	VARCHAR	2000	Null	
3	STRE_FILE_NM	Stored file name	VARCHAR	255	Null	pk
4	ORIGNL_FILE_NM	Original file name	VARCHAR	255	NotNull	
5	FILE_EXTSN	File extension	VARCHAR	20	Null	
6	FILE_CN	File contents	BLOB		Null	
7	FILE_SIZE	File size	NUMERIC	8	Null	

ERD



Configuration

Camera Device properties required for use of Camera-related functions of Camera Device API Guidance Program are as follows:

Device Application

config.xml

ExternalHosts

```
<accessorigin="ExternalHosts"/>
```

Plugin

```
<featurename="Camera">
<paramname="ios-package" value="CDVCamera"/>
</feature>
<featurename="FileTransfer">
<paramname="ios-package" value="CDVFileTransfer"/>
</feature>
```

Server Application

context-properties.xml

```
<entrykey="serverContext" value="Server Directory "/>
```

```
<entrykey="fileStorePath"value="File_Save_Path"/>
    resource/egovframework/sqlmap/sql-map-config_[DB_NAME].xml
```

```
<sqlMapresource="egovframework/sqlmap/hyb/ios/cmr/EgovCameraiOSAPIGuide_SQL_[DB
NAME].xml"/>
```

Related Features

Camera Device API Guide Program is comprised of a function to send necessary information to server and inquire them, using photos taken with device's built-in camera and images saved in device's memory.

Send photographs taken with built-in camera to server

Business Logic

1. Check the network. If the network is not Wi-Fi, function based on user approval.
2. Image title must be minimum two characters and maximum 20. (Validation Rule)
3. If the titles are the same when transferring to server, determine whether or not to overwrite based on user approval.

Related Codes

Calls photo capture API.

```
function fn_egov_capture_photo(sel)
{
    var cameraOption = {};
    cameraOption = {
        quality: 50,
        destinationType : Camera.DestinationType.FILE_URI,
        sourceType : Camera.PictureSourceType.CAMERA,
        targetWidth: 200,
        targetHeight: 200
    };

    navigator.camera.getPicture(fn_egov_upload_photo,null,cameraOption);
};

    Loads captured image
```

```
function fn_egov_upload_photo(data)
{
    image = data;
    if(myScroll != null)
    {
        myScroll.destroy();
    }
    fn_egov_wait_iscroll('wrapper_image');
    toast("uploadPhoto success!");
};

    Request for image overlap check
```

```
var params = {
    photoSj : $("#photoSj").val(),
```



```

        uuid : device.uuid
    };

$.mobile.showPageLoadingMsg('a');
// depending on device performance, delay using setTimeout to ensure next code is executed after
ProgressDialog Show is completely loaded.
setTimeout(function()
    {
        fn_egov_sendto_server("/cmr/cameraPhotoAlbumCheckiOS.do",params);
    },
    500);
    Image overlap check result

var cameraiOSAPIVO = result["cameraiOSAPIVO"];
var currentImageSn = cameraiOSAPIVO["sn"];

if(currentImageSn != null && currentImageSn != "")
{
    jConfirm("File with the same name already exists..\nEdit image?", "Alert", "b",
        function(result)
        {
            if(result)
            {
                fn_egov_update_PhotoGraph(currentImageSn);
            }
        });
}
else
{
    jConfirm("Save?", "Alert", "b",
        function(result)
        {
            if(result)
            {
                fn_egov_insert_PhotoGraph();
            }
        });
}
}

    Send image to server

function fn_egov_insert_PhotoGraph()
{
    $.mobile.showPageLoadingMsg();

    setTimeout(function()
        {
            var options = new FileUploadOptions();
            options.fileKey = "file";
            options.fileName = new Date().format("YYYYMMDDhhmmss") + ".jpg";
            options.mimeType = "image/jpeg";

            var params = { };

```

```

        options.params = params;
        var ft = new FileTransfer();
        var serverPath = downloadContext +
"/cmr/photoAlbumImageUploadiOS.do?uuid="+ encodeURIComponent(device.uuid)+"&photoSj="+
encodeURIComponent("Image Title ");
        ft.upload(image, serverPath, fn_insertSuccess, fn_error, options);
    },
    500);
}

function fn_egov_update_PhotoGraph(currentImageSn) {

    $.mobile.showPageLoadingMsg();

    setTimeout(function()
    {
        var options = new FileUploadOptions();
        options.fileKey = "file";
        options.fileName = new Date().format("YYYYMMDDhhmmss") + ".jpg";
        options.mimeType = "image/jpeg";

        var ft = new FileTransfer();

        var params = { };

        options.params = params;
        var serverPath = downloadContext +
"/cmr/photoAlbumImageUpdateiOS.do?sn="+currentImageSn;
        ft.upload(image, serverPath, fn_insertSuccess, fn_error, options);
    },
    500);
}

```

Related Screen and Implementation Manual

Function	URL	Controller	method	Display (HTML)
Image overlap eck	ch /cmr/cameraPhotoAlbumChec kiOS.do	EgovCameraiOSAPICo ntroller	selectPhotoAlbum PhoSj	CameraAPI.html#addPhoto Album
Image upload (in sert)	/cmr/photoAlbumImageUploa diOS.do	EgovCameraiOSAPICo ntroller	fileUpload	CameraAPI.html#addPhoto Album
Image upload (up date)	/cmr/photoAlbumImageUpdat eiOS.do	EgovCameraiOSAPICo ntroller	fileUpdate	CameraAPI.html#addPhoto Album





Title: minimum 2 characters, maximum 20 characters rule is in effect.

Take photograph: calls camera device.

Save: conducts image overlap check on server and uploads.

Send images from the gallery to server

Business Logic

1. Check the network. If the network is not Wi-Fi, function based on user approval.
2. Image title must be minimum two characters and maximum 20. (Validation Rule)
3. If the titles are the same when transferring to server, determine whether or not to overwrite based on user approval.

Related Codes

Calls gallery API.

```
function fn_egov_capture_photo(sel)
{
    var cameraOption = {};
    cameraOption = {
        quality: 50,
        destinationType : Camera.DestinationType.FILE_URI,
        sourceType : Camera.PictureSourceType.PHOTOLIBRARY,
        targetWidth: 200,
        targetHeight: 200
    };

    navigator.camera.getPicture(fn_egov_upload_photo,null,cameraOption);
};
    Loads selected image from gallery

function fn_egov_upload_photo(data)
{
    image = data;
    if(myScroll != null)
    {
        myScroll.destroy();
    }
    fn_egov_wait_iscroll('wrapper_image');
    toast("uploadPhoto success!");
};
    Request for image overlap check

var params = {
    photoSj : $("#photoSj").val(),
    uuid : device.uuid
};

$.mobile.showPageLoadingMsg('a');
// depending on device performance, delay using setTimeout to ensure next code is executed after
ProgressDialog Show is completely loaded.
setTimeout(function()
{
    fn_egov_sendto_server("/cmr/cameraPhotoAlbumCheckiOS.do",params);
},
500);
```

Image overlap check result

```
var cameraiOSAPIVO = result["cameraiOSAPIVO"];
var currentImageSn = cameraiOSAPIVO["sn"];
```

```
if(currentImageSn != null && currentImageSn != "")
{
    jConfirm("File with the same name already exists..\nEdit image?", "Alert", "b",
        function(result)
        {
            if(result)
            {
                fn_egov_update_PhotoGraph(currentImageSn);
            }
        });
}
else
{
    jConfirm("Save?", "Alert", "b",
        function(result)
        {
            if(result)
            {
                fn_egov_insert_PhotoGraph();
            }
        });
}
}
```

Send image to server

```
function fn_egov_insert_PhotoGraph()
{
    $.mobile.showPageLoadingMsg();

    setTimeout(function()
    {
        var options = new FileUploadOptions();
        options.fileKey = "file";
        options.fileName = new Date().format("YYYYMMDDhhmmss") + ".jpg";
        options.mimeType = "image/jpeg";

        var params = { };

        options.params = params;
        var ft = new FileTransfer();
        var serverPath = downloadContext +
"/cmr/photoAlbumImageUploadiOS.do?uuid="+ encodeURIComponent(device.uuid)+"&photoSj="+
encodeURIComponent("Image Title ");
        ft.upload(image, serverPath, fn_insertSuccess, fn_error, options);
    },
    500);
}
```

```
function fn_egov_update_PhotoGraph(currentImageSn) {
```

```

$.mobile.showPageLoadingMsg();

setTimeout(function()
{
    var options = new FileUploadOptions();
    options.fileKey = "file";
    options.fileName = new Date().format("YYYYMMDDhhmmss") + ".jpg";
    options.mimeType = "image/jpeg";

    var ft = new FileTransfer();

    var params = { };

    options.params = params;
    var serverPath = downloadContext +
"/cmr/photoAlbumImageUpdateiOS.do?sn="+currentImageSn;
    ft.upload(image, serverPath, fn_insertSuccess, fn_error, options);
},
500);
}

```

Related Screen and Implementation Manual

Function	URL	Controller	method	Display (HTML)
Image overlap ck	che /cmr/cameraPhotoAlbumChec kiOS.do	EgovCameraiOSAPICo ntroller	selectPhotoAlbum PhoSj	CameraAPI.html#addPhot oAlbum
Image upload (ins ert)	/cmr/photoAlbumImageUploa diOS.do	EgovCameraiOSAPICo ntroller	fileUpload	CameraAPI.html#addPhot oAlbum
Image (update)	Upload /cmr/photoAlbumImageUpdat eiOS.do	EgovCameraiOSAPICo ntroller	fileUpdate	CameraAPI.html#addPhot oAlbum





Title: minimum 2 characters, maximum 20 characters rule is in effect.

Gallery: calls Device's gallery image information.

Save: conducts image overlap check on server and uploads.

Inquire server's image list

Business Logic

1. Check the network. If the network is not Wi-Fi, function based on user approval.

Related Codes

Request server's image list

```
function fn_egov_show_ImgList()
{
    if(fn_egov_network_check(false))
    {
```

```

var params = null;
if(ImageArray === null)
{
    params = {pageIndex:1};
}
else
{
    if(ImageArray.length % 10 === 0)
    {
        params = {pageIndex:1+ImageArray.length/10};
    }
    else
    {
        jAlert('Last Page.', 'Alert', 'c');
        return;
    }
}

$.mobile.showPageLoadingMsg('a');
setTimeout(function()
{
    fn_egov_sendto_server("/cmr/cameraPhotoAlbumListiOS.do",params);
},
500);
}

    Inquire details about the selected image

function fn_detailPhotoAlbum(VOsn)
{
    if(fn_egov_network_check(false))
    {
        currentImageSn = VOsn;
        var params = {
            sn : VOsn
        };

        $.mobile.showPageLoadingMsg('a');
        setTimeout(function()
        {
            fn_egov_sendto_server("/cmr/cameraPhotoAlbumDetailiOS.do",params);
        },
        500);
    }
}

    Delete selected image

function fn_deletePhotoAlbum()
{
    if(fn_egov_network_check(false))
    {
        jConfirm("Delete?", "Alert", "b",
            function(result)
            {

```

```

        if(result)
        {
            var params = {
                sn : currentImageSn
            };

            $.mobile.showPageLoadingMsg('a');
            setTimeout(function()
            {

fn_egov_sendto_server("/cmr/deleteCameraPhotoAlbumiOS.do",params);
                },
                500);
            }
        });
    }
}

```

Request server's image list inquiry

Related Screen and Implementation Manual

Function	URL	Controller	method	Display (HTML)
Request server's image list	/cmr/cameraPhotoAlbumListiOS.do	EgovCameraiOSAPI Controller	selectCameraPhotoAlbumList	CameraAPI.html#listPhotoAlbum
Details about the selected image	/cmr/cameraPhotoAlbumDetailiOS.do	EgovCameraiOSAPI Controller	selectPhotoAlbum	CameraAPI.html#detailPhotoAlbum
Delete selected image	/cmr/deleteCameraPhotoAlbumiOS.do	EgovCameraiOSAPI Controller	deleteCameraPhotoAlbum	CameraAPI.html#detailPhotoAlbum





List: requests details of the selected list.
View more: loads 10 more image lists on server.
List: calls server's image list page.
Delete: deletes selected image.

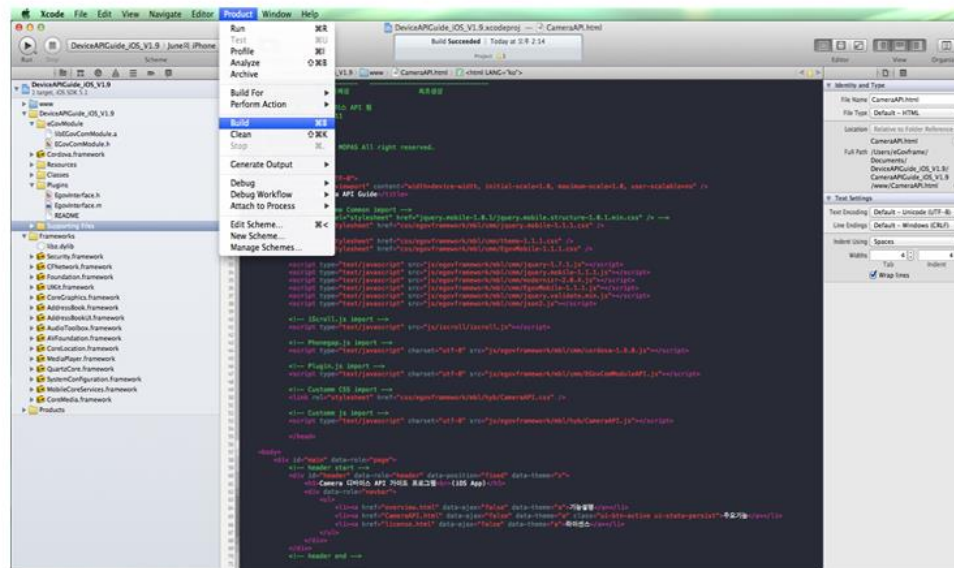
Compiling, debugging, distributing

Compiling

Device Application

1. [CameraAPI installation link](#)

- Click on "build" after checking CameraAPI project's installation status.



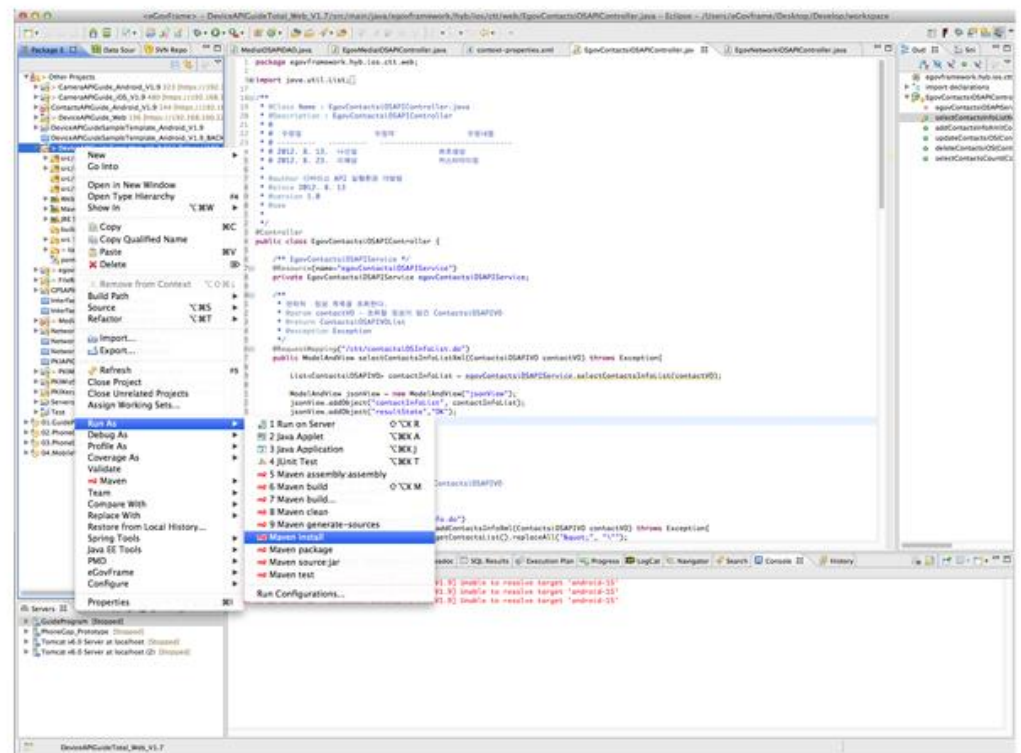
- The following screen will appear upon successful build.



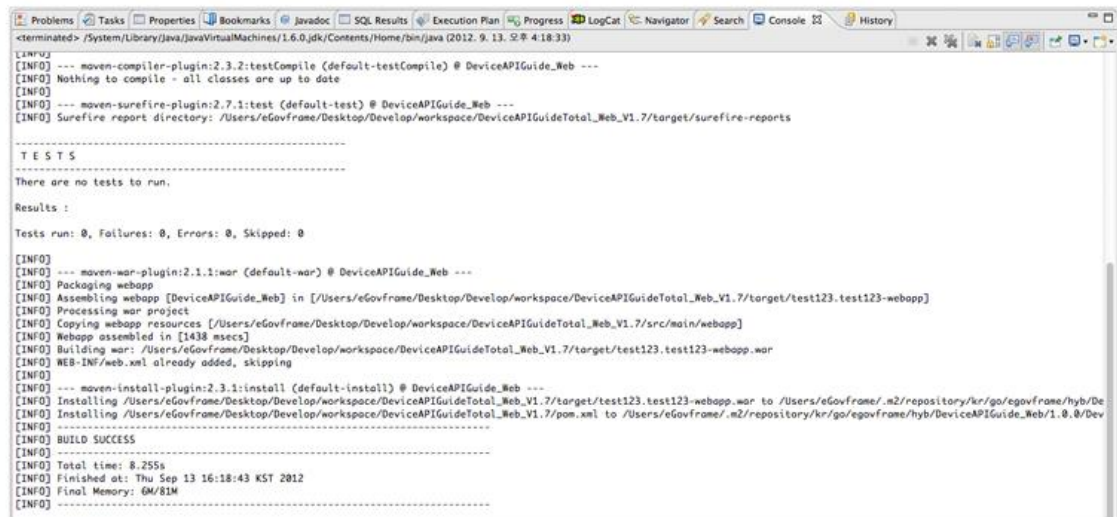
Server Application

1. [Web Server installation link](#)

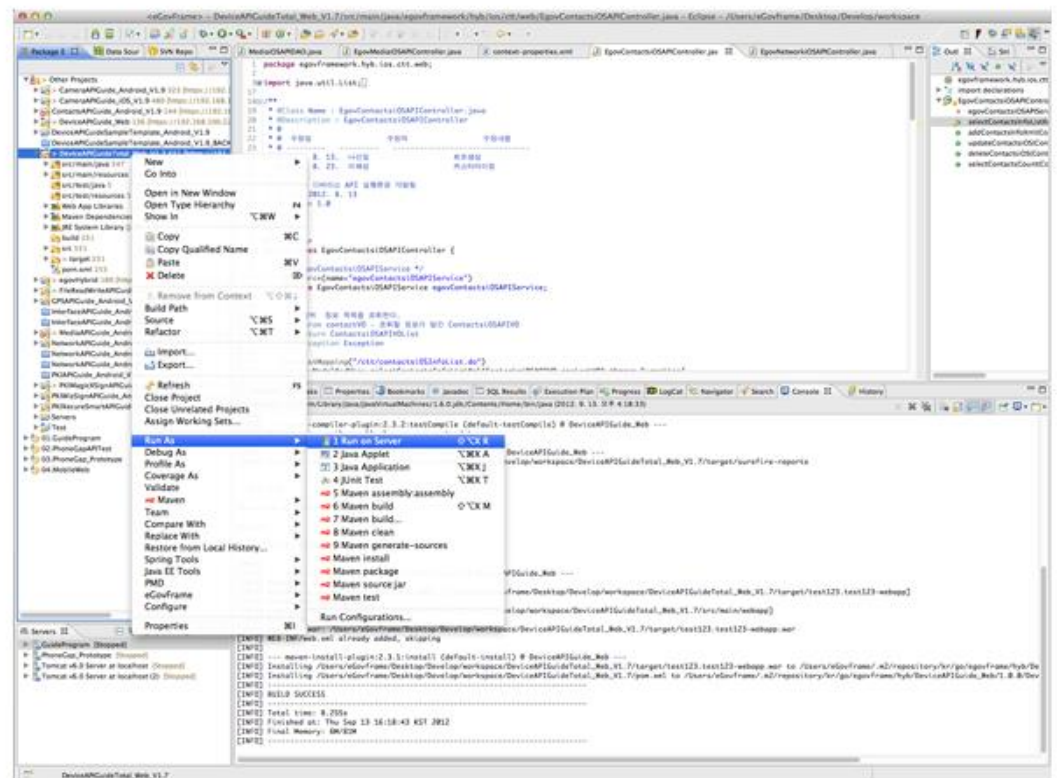
2. Right-click on the project>Run as>Maven install to build.



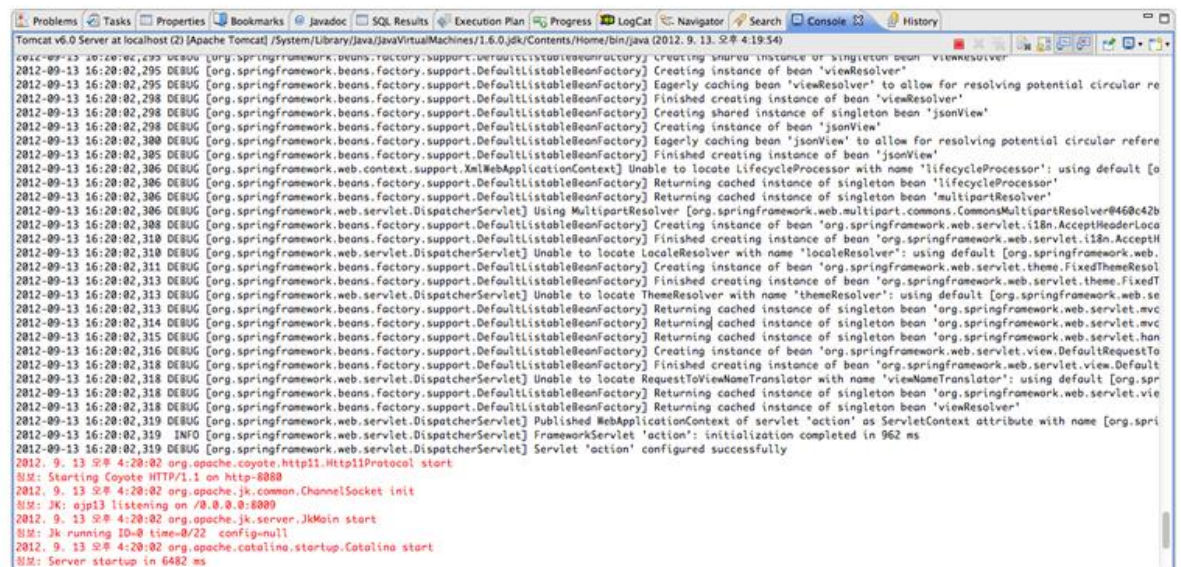
3. Build successful (check console information)



4. Right-click on project>Run as>Run on Server to execute.



5. Check to see if project is successfully executed.



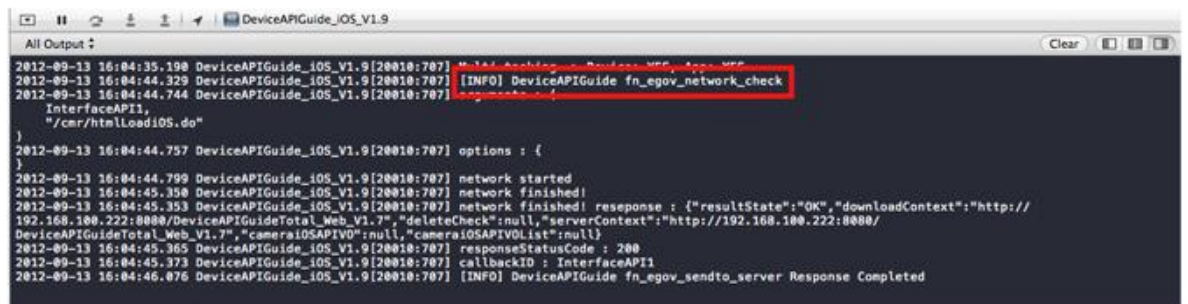
Debugging

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 both Eclipse and Xcode.

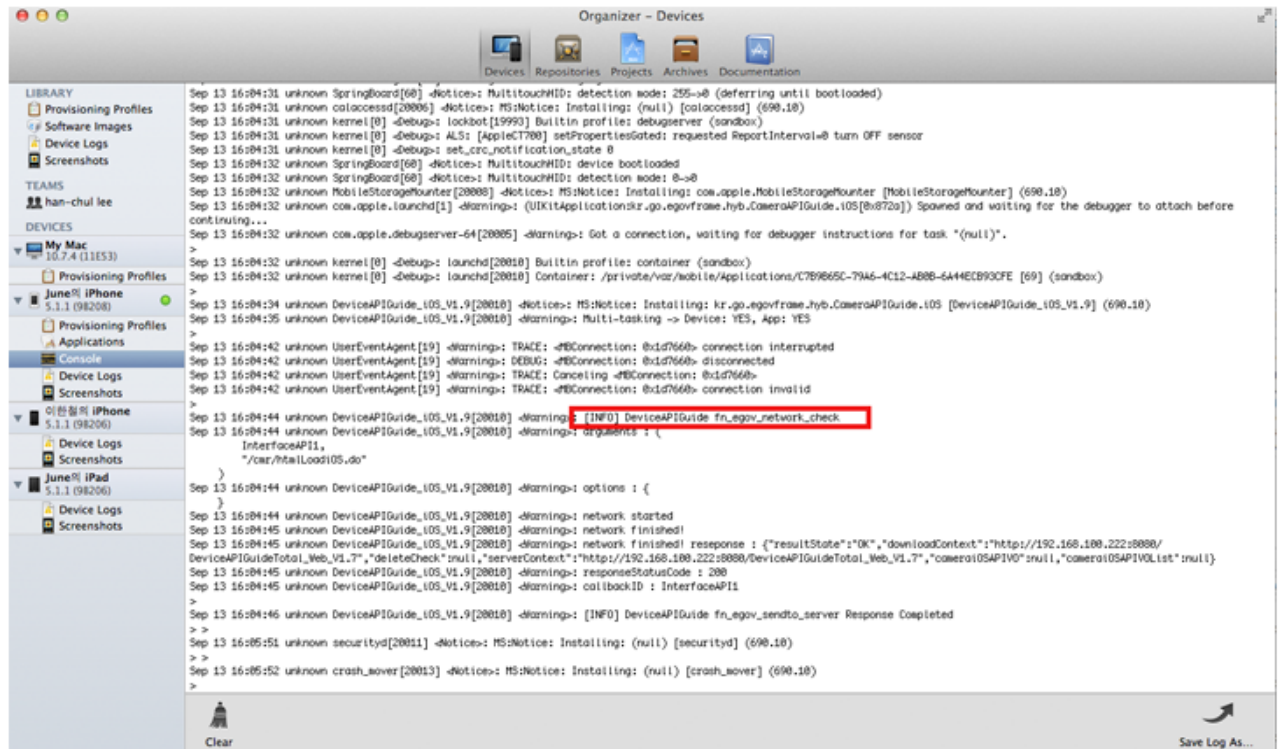
- Example of actual console log

```
function fn_egov_network_check(doCheck)
{
    console.log('DeviceAPIGuide fn_egov_network_check');
    var networkState = navigator.network.connection.type;
    ...
}
```

- xCode console window



- Organizer log window



Log message	Value	Remark
DeviceAPIGuide fn_egov_network_check		Checks network condition
DeviceAPIGuide fn_insertSuccess		Image transfer to server successful
DeviceAPIGuide fn_error		Image transfer to server failed
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

Distribution

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

References

- UX/UI library : jQuery Mobile [Click](#)
- Phonegap 4.3.0 : [Click](#)

