Adding contacts to a group in Symbian Web Runtime

This code snippet shows how to add an existing contact to a contact group using the Contacts Platform Service for Web Runtime. It is supported from S60 5th Edition onwards.

Overview

To obtain access to the service object for the Contacts Service API, the method `getServiceObject("Service.Contact", "IDataSource")` is used.

In the application startup, the list of contacts and contact groups is loaded using the function `Service.IDataSource.GetList()`. The `criteria` object is used to determine the type of data.

After choosing the contact and contact group, the function `Service.IDataSource.Organise()` is used to add the contact to the contact group. To set up correct values for the function `Service.IDataSource.Organise()`, the `criteria` object is used.

The `criteria` object contains `Type`, `Data`, and `OperationType` to define the organising operation.

The `groupdata` object is used to define the contact group and contacts for the `criteria.Data` object.

Source file: Relevant HTML components

```
<select id="contacts" size="5">
    <option></option>
</select><br/>
<select id="groups" size="5">
    <option></option>
</select><br/>
<input type="button" onclick="doAdd();" value="Add to group" />
```

Source file: JavaScript file

```
// Declare the service object
var so = null;
```

```
// Initializes the widget
function init() {
    try {
        //Get the service object
        so = device.getServiceObject("Service.Contact", "IDataSource");
    } catch (e) {
        alert("init: " + e );
    }

    // Clean the contact list
    var contactList = document.getElementById("contacts");
    while (contactList.length != 0){
        contactList.remove(0);
    }
    // Clean the contact group list
    var groupList = document.getElementById("groups");
    while (groupList.length != 0){
        groupList.remove(0);
    }
    // Get the list of contacts
```


(C) Copyright Nokia 2014. All rights reserved.

```
    getContactList();
    // Get the list of contact groups
    getGroupsList();
    // Select the first contact group
    groupList.options[0].selected = true;
    // Select the first contact
    contactList.options[0].selected = true;

    // ...
}
```

```
/**
 * Java Script function
 * Get the list of contacts
 */
function getContactList() {
    // Criteria for Service.IDataSource.GetList() function
    var criteria = new Object();
    // Type of list
    criteria.Type = "Contact";
    try {
        var result;
        // Service.IDataSource.GetList() function - method retrieves a list of
        // contacts
        result = so.IDataSource.GetList(criteria);
        if (result.ErrorCode == 0) {
            //Fill the list of contacts
            fillContactsList(result.ReturnValue);
        } else {
            alert("Error in getting contact list");
        }
    } catch (e) {
        alert ("getContactList: " + e);
    }
}

/**
 * Java Script function
 * Fill the list of contacts
 * @param iterator List of contacts, equals result.ReturnValue of
 * Service.IDataSource.GetList() function return value
 */
function fillContactsList(iterator) {
    var contactList = document.getElementById("contacts");
    try{
        iterator.reset();
        var item;
        while ((item = iterator.getNext()) != undefined) {
            // Last name of contact
            var lastName = "";
            if (item.LastName != undefined) {
                lastName = item.LastName.Value;
            }
            // First name of contact
            var firstName = "";
```


(C) Copyright Nokia 2014. All rights reserved.

```
            if (item.FirstName != undefined) {
                firstName = item.FirstName.Value;
            }
            // Contact id
            var id = item.id;
            var node = document.createElement("option");
            node.value = id;
            node.appendChild(document.createTextNode(lastName + " " +
                firstName));
            contactList.appendChild(node);
        }
    } catch (e) {
        alert("fillContactsList" + e);
    }
}

/**
 * Java Script function
 * Get the list of contact groups
 */
function getGroupsList() {
    // Criteria for Service.IDataSource.GetList() function
    var criteria = new Object();
    // Type of list
    criteria.Type = "Group";
    try {
        var result;
        // Service.IDataSource.GetList() function - method retrieves a list of
        // contact groups
        result = so.IDataSource.GetList(criteria);
        if (result.ErrorCode == 0) {
             // Fill the list of contact groups
            fillGroupsList(result.ReturnValue);
        } else {
            alert("Error in getting contact list: " + result.ErrorCode);
        }
    } catch (e) {
        alert("getContactList: " + e);
    }
}

/**
 * Java Script function
 * Fill the list of groups
 * @param iterator List of groups, equals result.ReturnValue of
 * Service.IDataSource.GetList() function return value
 */
function fillGroupsList(iterator) {
    var groupList = document.getElementById("groups");
    try {
        iterator.reset();
        var item;
        while ((item = iterator.getNext()) != undefined) {
            // Name of group
            var groupName = item.GroupLabel;
            // Group's id
            var id = item.id;
```


(C) Copyright Nokia 2014. All rights reserved.

```
            var node = document.createElement ("option");
            node.value = id;
            node.appendChild(document.createTextNode(groupName));
            groupList.appendChild(node);
        }
    } catch (e) {
        alert("fillContactsList" + e);
    }
}

// Add selected contact to the selected contact group
function doAdd() {
    var contact;
    var group;
    // Get the selected contact group
    var groupList = document.getElementById("groups");
    for (var i = 0; i < groupList.options.length; i++) {
        if (groupList.options[i].selected) {
            group = groupList.options[i];
        }
    }
    // Get the selected contact
    var contactList = document.getElementById("contacts");
    for (var i = 0; i < contactList.options.length; i++) {
        if (contactList.options[i].selected) {
            contact = contactList.options[i];
        }
    }
    // Create the contact and contact group data to update
    var groupdata = new Object();
    // The id of selected contact group
    groupdata.id = group.value;
    groupdata.IdList = new Array();
    // The id of selected contact
    groupdata.IdList[0] = contact.value;
    // Criteria for update contact and contact group
    var criteria = new Object();
    criteria.Type = "Group";
    criteria.Data = groupdata;
    // Add the contact to group, "Disassociate" to delete
    criteria.OperationType = "Associate";

    try {
        // Organise method adds contacts to a contact group (association)
        var result = so.IDataSource.Organise(criteria);
        if (result.ErrorCode == 0) {
            alert("Contact added to group successfully");
        } else {
            alert("Error occurs while organizing contacts");
        }
    } catch (e) {
        alert("Exception occure while organizing contacts: " + e);
    }
}
```


(C) Copyright Nokia 2014. All rights reserved.

Postconditions

At startup, the "Contacts" and "Contact Groups" listboxes are shown.

To associate a contact with a contact group, use the "Add to group" button.

Supplementary material

This code snippet is part of the stub concept, which means that it has been patched on top of a template application in order to be more useful to developers. The version of the WRT stub application used as a template in this snippet is v1.1.

- The patched, executable application that can be used to test the features described in this snippet is available for download at Media:AddingContactsToGroup.zip.
- You can view all the changes that are required to implement the above-mentioned features. The changes are provided in unified diff and colour-coded diff (HTML) formats in Media:AddingContactsToGroup.diff.zip.
- For general information on applying the patch, see Using Diffs.
- For unpatched stub applications, see Example app stubs with logging framework.

(C) Copyright Nokia 2014. All rights reserved.