

## 04 - Cordova Integration (iOS and Android)

### Procedures

---

Here are the steps necessary to integrate the in-app survey iOS & Android SDK plugin (In-App Survey Plugin) in Cordova:

1. Download the MCXInAppSurvey plugin into the desired directory.
2. Open the terminal and navigate to the Cordova project root directory.
3. To install the plugin, use the following command:

```
Cordova plugin add <path to MCXInAppSurvey plugin>
```

4. After successfully adding the plugin in Cordova, open the iOS and Android applications from the **platform-ios** and **platform-android** directories respectively.

5. To call the survey using the plugin, use this function from your JavaScript file:

```
cordova.exec(successCallback, errorCallback, "MCXInAppSurvey", "initiateSurvey", [InputData]);
```

Here is the explanation of each parameter passed to the Cordova function:

1. **successCallback** – This function is called when the In-App Survey Plugin is successfully invoked. Below is a sample declaration of this function. Here we are just printing the result of this function; the developer can check the result of the function and act accordingly.

```
var successCallback = function(succ) {  
    console.log('successCallback : ' + succ);  
}
```

2. **errorCallback** – This function is called when the In-App Survey Plugin fails to initialize the survey. Below is the sample declaration of this function. Here we are just printing the error returned by this function; developer can check the error and act accordingly.

```
var errorCallback = function(err) {  
    console.log('errorCallback : ' + err);  
}
```

3. **MCXInAppSurvey** – This is the plugin class name, use as it is.

**4. initiateSurvey** – This is the function name to initiate survey from the In-App Survey Plugin.

**5. [InputData]** – This is the array of objects which we are passing to the In-App Survey Plugin to initiate the survey or display a custom prompt.

Here is the format of the InputData object:

```
var InputData = {  
  "programToken": <Program Token>,  
  "eventName": <Event alias name>,  
  "customPrompt": <Custom prompt JSON String>,  
  "prePopData": <Pre-Population values JSON String>,  
  "debug": <true/false Optional Boolean value>,  
};
```

**1. programToken** – This is given to the developer by the business owner or manager.

**2. eventName** – This is the event alias name discussed in the [Overview](#) (e.g. feedback, logout, etc.).

**3. customPrompt** – This is JSON in string format which has all the configuration of a custom prompt, e.g. text, button text, colors, etc. (sample JSON is below).

**4. prePopData** – This is JSON in string format which contains the pre-population values discussed in the [Overview](#), also called survey prefill values.

Here is an example of sample JSON:

```
{"FirstName":"FirstNameValue","LastName":"LastNameValue"}
```

**5. debug** – This is an optional parameter used for debugging only. Pass "true" if you want to print a raw request and response in the console logs. The default value is "false" if not passed.

Here is an example of custom prompt JSON:

```
{  
  "headerImageURL":"https://images.pexels.com/photos/676...oms-67636.jpeg",  
  "bodyText":"Your feedback is incredibly valuable and will allow us to focus on those aspects that are most important to you. <br /><br /> Please take our 1 minute survey.",  
  "bodyBackgroundColor":"#ffffff",  
  "mainBackgroundColor":"#7122ff",  
  "noButtonBackgroundColor":"#FF6900",  
  "noButtonText":"No Thanks",  
  "noButtonTextColor":"#ffffff",  
  "yesButtonBackgroundColor":"#FF6900",
```

```
"yesButtonText":"Start Survey",
"yesButtonTextColor":"#ffffff",
"footerBackgroundColor":"#003366",
"footerText":"Footer text here",
"footerTextColor":"#ffffff",
"footerHeaderText":"Thank you very much.",
"optOutButtonText":"",
"optOutButtonTextColor":"#000000",
"disclosureText":"Disclosure text here if you don't want optout button",
"disclosureTextColor":"#000000"
}
```