2nd May 2023

Scan credit card in offline

# Overview

The OfflineCCScan component scan the credit/debit card through the camera and returns the data

## Use case

 1. Instead of manually entering the card details by using this component details can be scanned very easily.

 B .Percentage of re-use:

Approximate 90% of reuse. It sets an expectation of how much can be used out of the box, and how much needs to be customized for a specific app.

 C. Features

1. The component utilizes CardReader third party library to scan the card.
2. Uses NFI 3.0

2. Getting Started

## A. Prerequisites

 Before you start using the OfflineCCScan component, ensure the following:

 • [HCL Foundry](https://manage.hclvoltmx.com/)

 • Volt MX Iris

## Platforms Supported

### Mobile

#### iOS

### Tablets

## Importing the app

## You can import the Forge components only into the apps that are of the Reference Architecture type.

##  **To import the OfflineCCScan component, do the following:**

## Open your app project in Volt MX Iris.

 2. In the Project Explorer, click the **Templates** tab.

 

 3.Right-click **Components**, and then select **Import Component**. The **Import Component** dialog box appears.



4.Click **Browse** to navigate to the location of the component, select the component, and then click **Import**. The component and its associated widgets and modules are added to your project.

Once you have imported a component to your project, you can easily add the component to a form. For more information, refer [Add a Component to a Form](https://opensource.hcltechsw.com/volt-mx-docs/docs/documentation/Iris/iris_user_guide/Content/C_UsingComponents.html%22%20%5Cl%20%22add-a-component-to-a-form)

## Building and previewing the app

After performing all the above steps, you can build your app and run it on your device. For more information, you can refer to the [Building and Viewing an Application](https://opensource.hcltechsw.com/volt-mx-docs/docs/documentation/Iris/iris_user_guide/Content/Cloud_Build_in_VoltMX_Iris.html#cloud) section of the Volt MX User Guide.

 You can then run your app to see the OfflineCCScan work in real time.

# References

## Dynamic Usage

 You can also add**OfflineCCScan** component dynamically. To do so,

1. In the **Project Explorer**, on the **Projects** tab, click **Controllers** section to access the respective **Form Controller**. Create a method and implement the code snippet similar to the sample code mentioned below.



/\* Creating **OfflineCCScan** component instance \*/

var **OfflineCCScan** = new com.technohub.**OfflineCCScan** ({

 "height": "50%",

 "id": “OfflineCCScan",

 "isVisible": true,

 "left": "0dp",

 "masterType": constants.MASTER\_TYPE\_USERWIDGET,

 "isModalContainer": false,

 "skin": "slFbox",

 "top": "0dp",

 "width": "100%",

 "zIndex": 1,

 "overrides": {

 "OfflineCCScan": {

 "right": "viz.val\_cleared",

 "bottom": "viz.val\_cleared",

 "minWidth": "viz.val\_cleared",

 "minHeight": "viz.val\_cleared",

 "maxWidth": "viz.val\_cleared",

 "maxHeight": "viz.val\_cleared",

 "centerX": "viz.val\_cleared",

 "centerY": "viz.val\_cleared"

 }

 }

 }, {

 "overrides": {}

 }, {

 "overrides": {}

});

 /\*Adding the **OfflineCCScan** component to a Form\*/

this.view.add(**OfflineCCScan**);

 In the code snippet, you can edit the properties of the component as per your requirement. For more information, see Setting Properties.

2. Save the file

Configuring Native Settings (iOS)

To configure the native settings for iOS, follow these steps:

1. From the **Project** explorer, go to **Assets** and expand **Media**.
2. Right-click **Common**, and then select **Resource Location**. Volt MX Iris opens the common resources folder in a file explorer.

3. Open the **infoplist\_configuration.json** file with a text or code editor.
4. At the end of the file, type the following code. You can change the description based on your preference.

"NSCameraUsageDescription" : "Your Description"

 

5. Save the file.

Configuring Deployment Target

1. From the left navigation menu, click **Project Settings**.
2. In the **Project Settings** window, go to **Native** → **iPhone/iPad**.
3. Under **Target Versions**, from the **iOS Version** list, select **11.0** or higher.


### 1. Events:

### 1. resultCallback

|  |  |
| --- | --- |
| **Description:** | Returns the card details within this callback. |
|   Syntax: | resultCallback |
| **Parameters**:  |  Callback as a parameter  |
|  **Remarks:** | None |
|  **Example:** | postShow(){ kony.runOnMainThread(function(){ this.view.OfflineCCScan.resultCallback(this.callBackHandler); }.bind(this), []); }, callBackHandler: function(data) {  var self = this; kony.runOnMainThread(function() { self.view.imgResult.base64 = data; }, []); }   |

 App version 1.0.0:

## Limitations

B. Known Issue