Date : 26-Sep-21

Fingerprint Authentication (1.0.4)

1. Overview

Fingerprint Authentication allows users to confirm their identity with a single touch. With this component, you can add fingerprint recognition to your app. Drag and drop the component in your app, configure few properties and you are ready to go.

The following sections help you use the Fingerprint-Authentication component in your app.

## Use case

### Fingerprint login your App

### Fingerprint to pay in your App

### B. Features

### Ready to use UI

## C. Percentage of re-use:

100% (Data can be customizable)

# Getting Started

## Prerequisites

Before you start using the Fingerprint Authentication component, ensure you have the following:

* [HCL Foundry](https://manage.hclvoltmx.com/)
* Volt MX Iris

## Platforms Supported

### Mobile

#### iOS

#### Android

### Tablets

## Importing the Component

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

## **To import the Fingerprint Authentication component, do the following:**

## Open your app project in Volt MX Iris.

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

## Graphical user interface, text Description automatically generated

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

Graphical user interface, text, application

Description automatically generated

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.

Text

Description automatically generated

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#add-a-component-to-a-form).

# 3.References

## A. Dynamic Usage

You can also add a **Fingerprint Authentication** 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 which is similar to the sample code mentioned below.

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

/\* creating a component's Object \*/

initFingerPrintComponent: function (){

var touchid = new com.voltmxmp.fingerprint(

{

"autogrowMode": voltmx.flex.AUTOGROW\_NONE,

"clipBounds": true,

"left": "0dp",

"top": "0dp",

"width": "100%",

"height": "100%",

"id": "touchid",

"isVisible": true,

"layoutType": voltmx.flex.FREE\_FORM,

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

"skin": "slFbox",

}, {}, {});

/\* Setting the component's properties \*/

touchid.touchIcon = "reusablelogin\_touch\_id\_icon.png";

touchid.buttonText = "Cancel";

/\*Adding the Touch ID component to a Form\*/

this.view.add(touchid);

},

## B. Properties

The properties provided on the **Component** tab allow you to customize the UI elements in the **Fingerprint Authentication** component. You can set the properties directly on the **Component** tab or by writing a JavaScript.

[Open](javascript:void(0);)1. Touch ID Icon

|  |  |
| --- | --- |
| **Category:** | General |
| **Description:** | Specifies the Touch ID icon for Fingerprint Authentication. |
| Syntax: | touchIcon |
| **Type:** | String |
| **Read/Write:** | Read + Write |
| **Example:** | this.view.componentID.touchIcon = "touch\_id\_icon.png"; |

2.[Open](javascript:void(0);)Button Text

|  |  |
| --- | --- |
| **Category:** | Android Popup Properties |
| **Description:** | Specifies the text of the cancel button. |
| Syntax: | buttonText |
| **Type:** | String |
| **Read/Write:** | Read + Write |
| **Example:** | this.view.componentID.buttonText = "Cancel" |

## C. Events

### 1. onTouchCallback Event

|  |  |
| --- | --- |
| **Description:** | This event is invoked when there is a touch action performed by the user on the fingerprint reader. |
| **Syntax:** | onTouchCallback(eventobject) |
| **Example:** | this.view.componentID.onTouchCallback = function(eventobject){alert(eventobject);} |

## APIs

-- None of the methods are exposed.

# 4.Revision History

App Version 1.0.4

## A. Limitation

## -- NA

## B. Known Issues

-- NA