Microsoft

Integrating Power Apps with mobile Native Apps

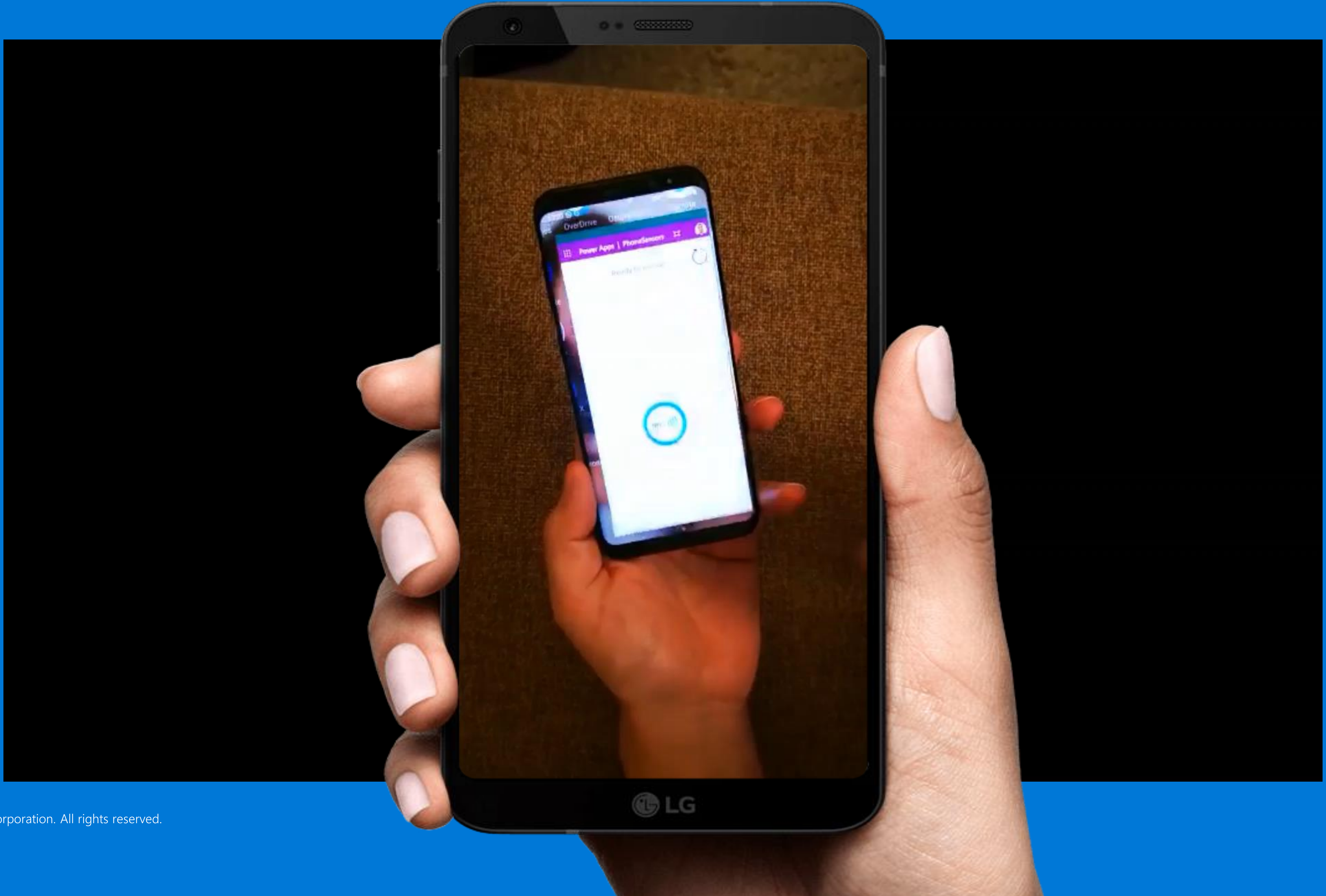
(NFC and Bluetooth in Power Apps)

Rui.Santos@Microsoft.com – TS CE D365

Let’s start with the Demos !!!

Read NFC Cards

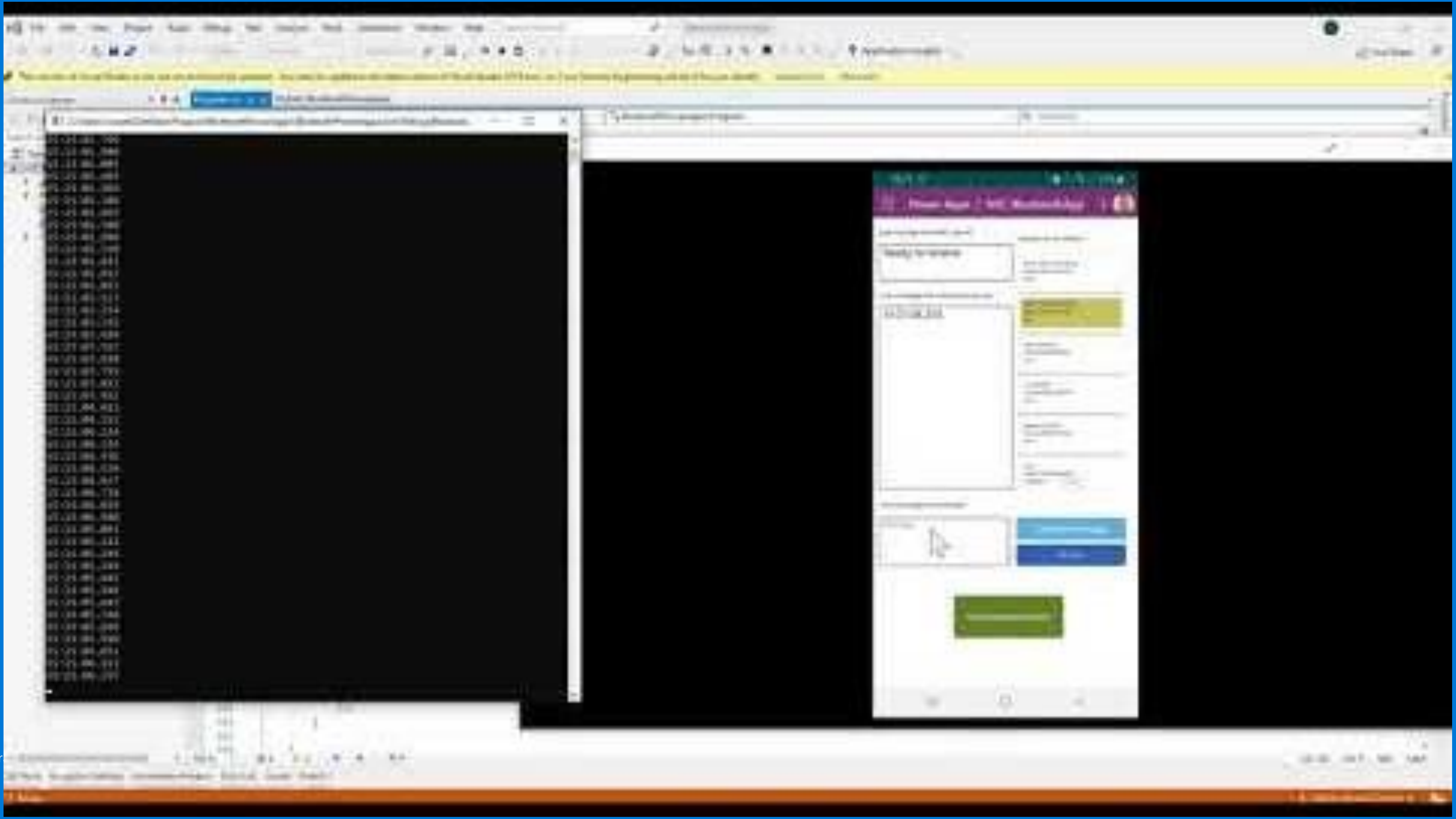
https://www.linkedin.com/posts/ruisantosnor_android-nfc-powerapps-activity-6624795233949360128-gf0-/



Microsoft

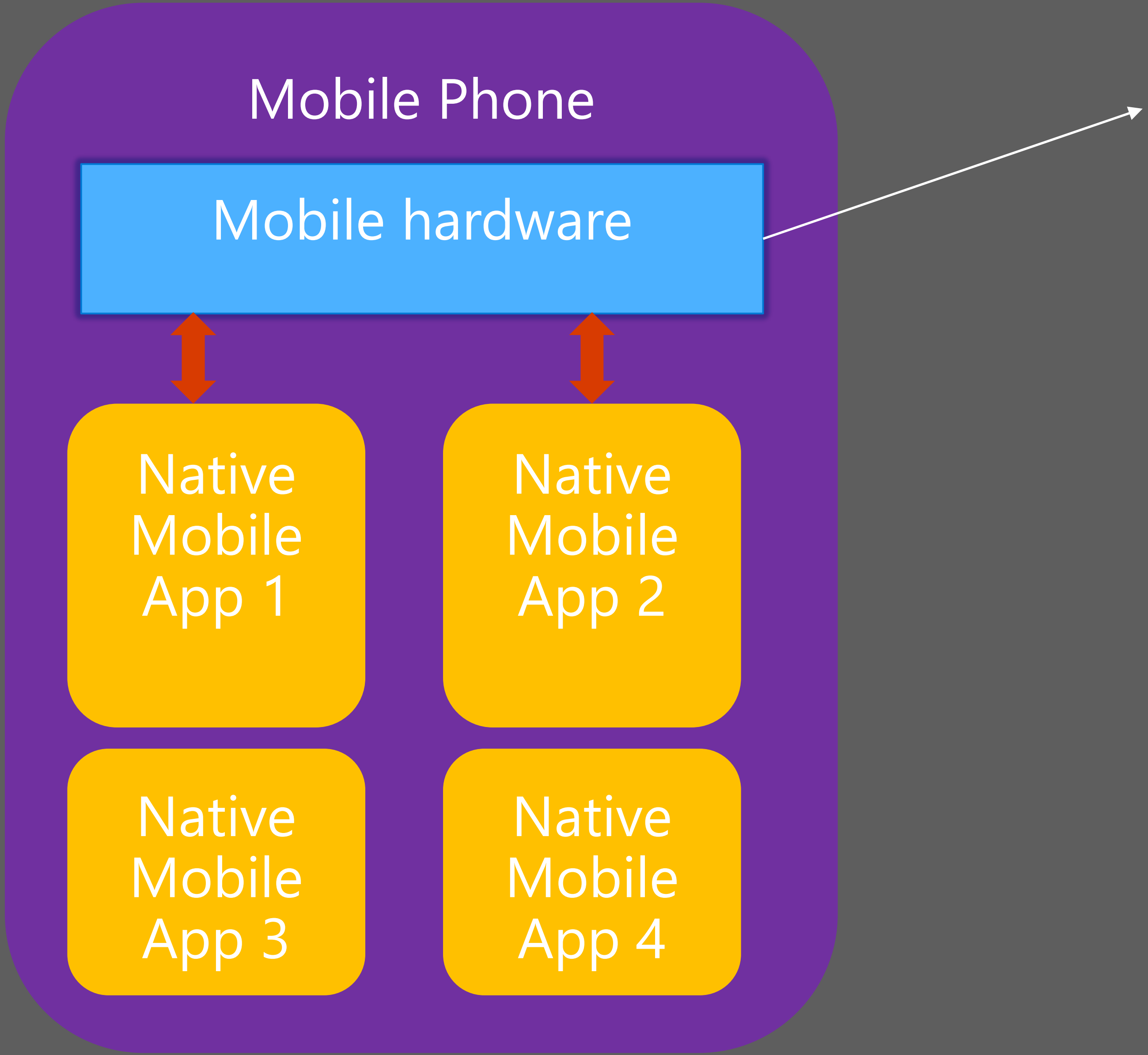
Use Bluetooth to communicate offline

https://www.linkedin.com/posts/ruisantosnor_bluetooth-healthcaretechnology-warehousemanagement-activity-6656217968759894016-LKjl



Microsoft

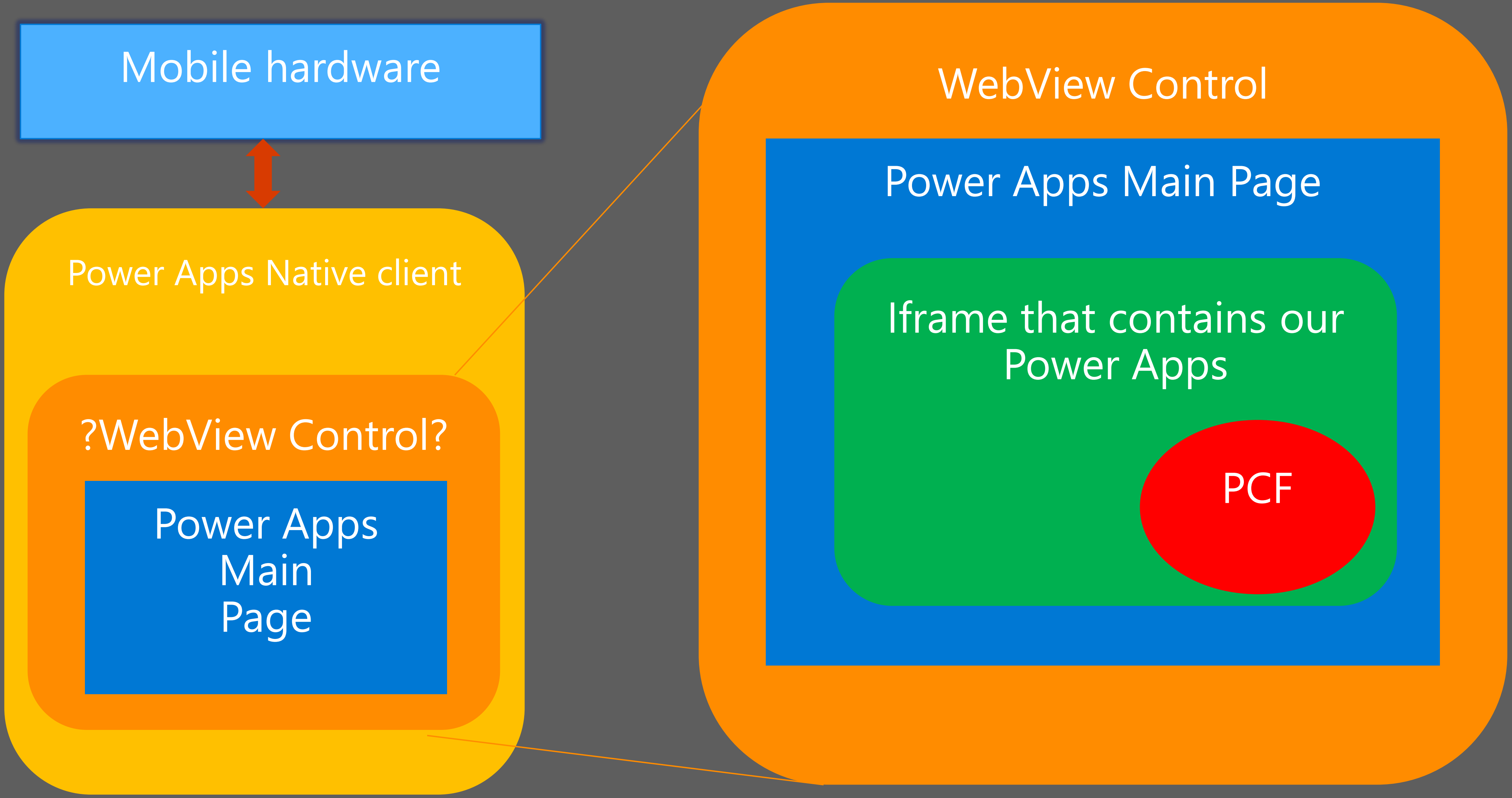
Structure of a native mobile App



- This is where the sensors are, like fingerprint, NFC, Heartbeat….

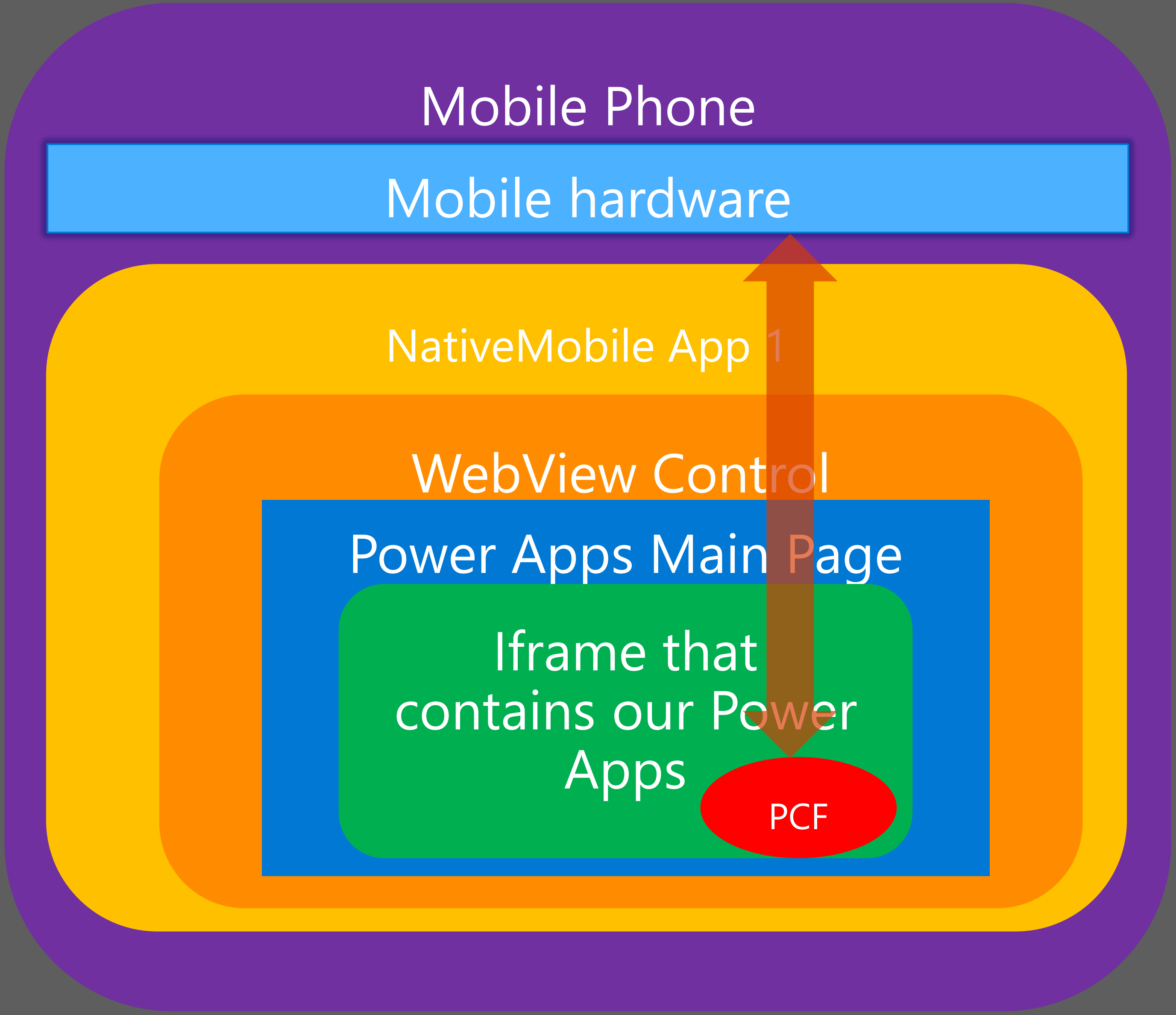
- You can only access the hardware of the phone from a native mobile App

Structure of a Power Apps host



Microsoft

All layers together



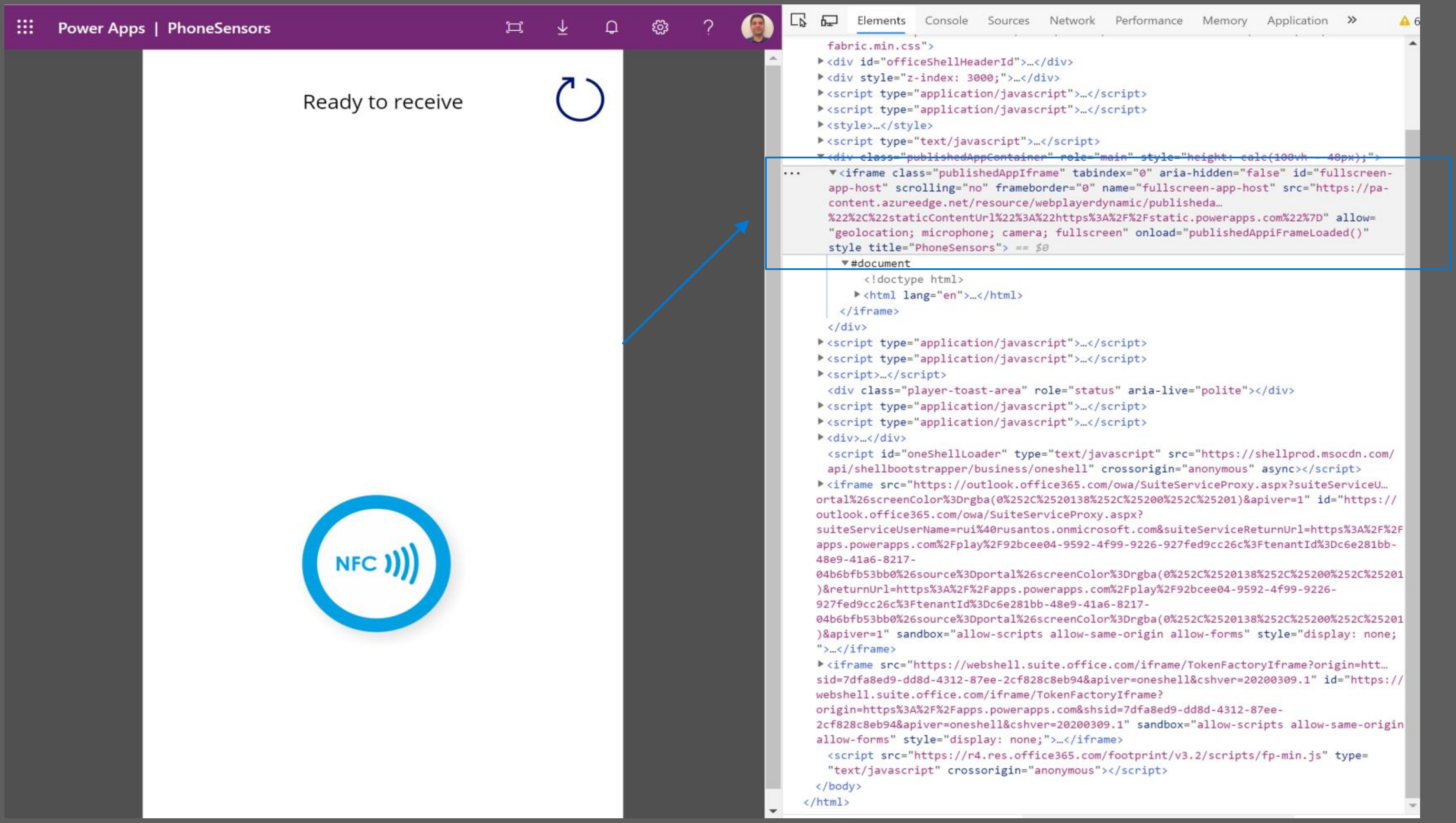
The Goal:

- Have a PCF to receive the messages from hardware sensors

The Challenge:

- Not possible to change the Power Apps native Client
- Inject code in the Power Apps main page
- Pass information between the different layers

Hacking Power Apps



```
    fabric.min.css">
  ▶<div id="officeShellHeaderId">…</div>
  ▶<div style="z-index: 3000;">…</div>
  ▶<script type="application/javascript">…</script>
  ▶<script type="application/javascript">…</script>
  ▶<style>…</style>
  ▶<script type="text/javascript">…</script>
  ▼<div class="publishedAppContainer" role="main" style="height: calc(100vh - 48px);">
...  ▼<iframe class="publishedAppIframe" tabindex="0" aria-hidden="false" id="fullscreen-
    app-host" scrolling="no" frameborder="0" name="fullscreen-app-host" src="https://pa-
    content.azureedge.net/resource/webplayerdynamic/publisheda…
    %22%2C%22staticContentUrl%22%3A%22https%3A%2F%2Fstatic.powerapps.com%22%7D" allow=
    "geolocation; microphone; camera; fullscreen" onload="publishedAppiFrameLoaded()"
    style title="PhoneSensors"> == $0
      ▼#document
          <!doctype html>
        ▶<html lang="en">…</html>
      </iframe>
    </div>
  ▶<script type="application/javascript">…</script>
  ▶<script type="application/javascript">…</script>
  ▶<script>…</script>
    <div class="player-toast-area" role="status" aria-live="polite"></div>
  ▶<script type="application/javascript">…</script>
  ▶<script type="application/javascript">…</script>
  ▶<div>…</div>
    <script id="oneShellLoader" type="text/javascript" src="https://shellprod.msocdn.com/
    api/shellbootstrapper/business/oneshell" crossorigin="anonymous" async></script>
  ▶<iframe src="https://outlook.office365.com/owa/SuiteServiceProxy.aspx?suiteServiceU…
  ortal%26screenColor%3Drgba(0%252C%2520138%252C%25200%252C%25201)&apiver=1" id="https://
  outlook.office365.com/owa/SuiteServiceProxy.aspx?
  suiteServiceUserName=rui%40rusantos.onmicrosoft.com&suiteServiceReturnUrl=https%3A%2F%2F
  apps.powerapps.com%2Fplay%2F92bcee04-9592-4f99-9226-927fed9cc26c%3FtenantId%3Dc6e281bb-
  48e9-41a6-8217-
  04b6bfb53bb0%26source%3Dportal%26screenColor%3Drgba(0%252C%2520138%252C%25200%252C%25201
  )&returnUrl=https%3A%2F%2Fapps.powerapps.com%2Fplay%2F92bcee04-9592-4f99-9226-
  927fed9cc26c%3FtenantId%3Dc6e281bb-48e9-41a6-8217-
  04b6bfb53bb0%26source%3Dportal%26screenColor%3Drgba(0%252C%2520138%252C%25200%252C%25201
  )&apiver=1" sandbox="allow-scripts allow-same-origin allow-forms" style="display: none;
  ">…</iframe>
  ▶<iframe src="https://webshell.suite.office.com/iframe/TokenFactoryIframe?origin=htt…
  sid=7dfa8ed9-dd8d-4312-87ee-2cf828c8eb94&apiver=oneshell&cshver=20200309.1" id="https://
  webshell.suite.office.com/iframe/TokenFactoryIframe?
  origin=https%3A%2F%2Fapps.powerapps.com&shsid=7dfa8ed9-dd8d-4312-87ee-
  2cf828c8eb94&apiver=oneshell&cshver=20200309.1" sandbox="allow-scripts allow-same-origin
  allow-forms" style="display: none;">…</iframe>
    <script src="https://r4.res.office365.com/footprint/v3.2/scripts/fp-min.js" type=
    "text/javascript" crossorigin="anonymous"></script>
  </body>
</html>
```

Create my own Power Apps native mobile App

Inject JavaScript code in the WebView to send information from the native code (sensor) to javascript world

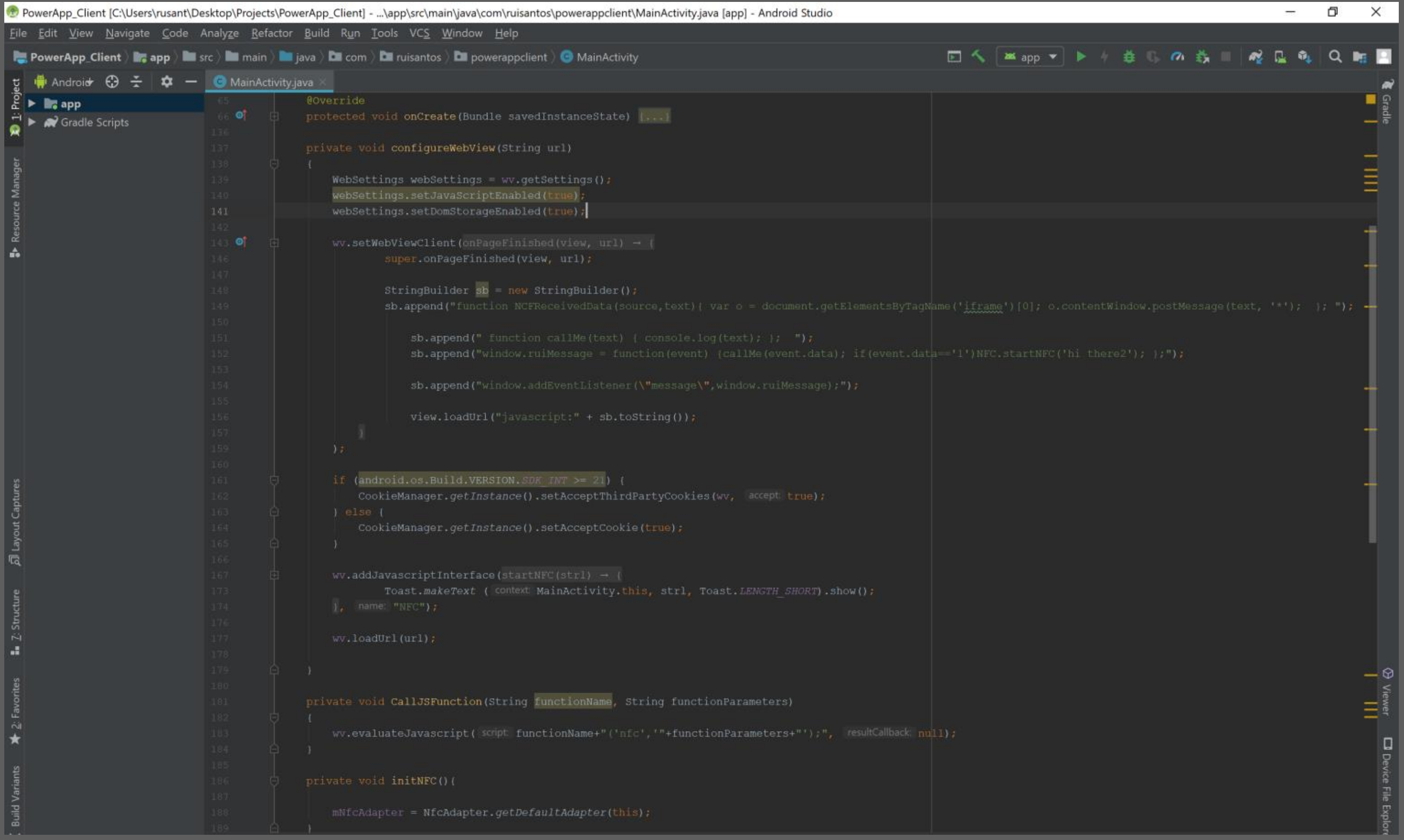
Use PCF to inject JavaScript code to receive the messages from the previous step (iframe <-> host communication)

Hacking Power Apps

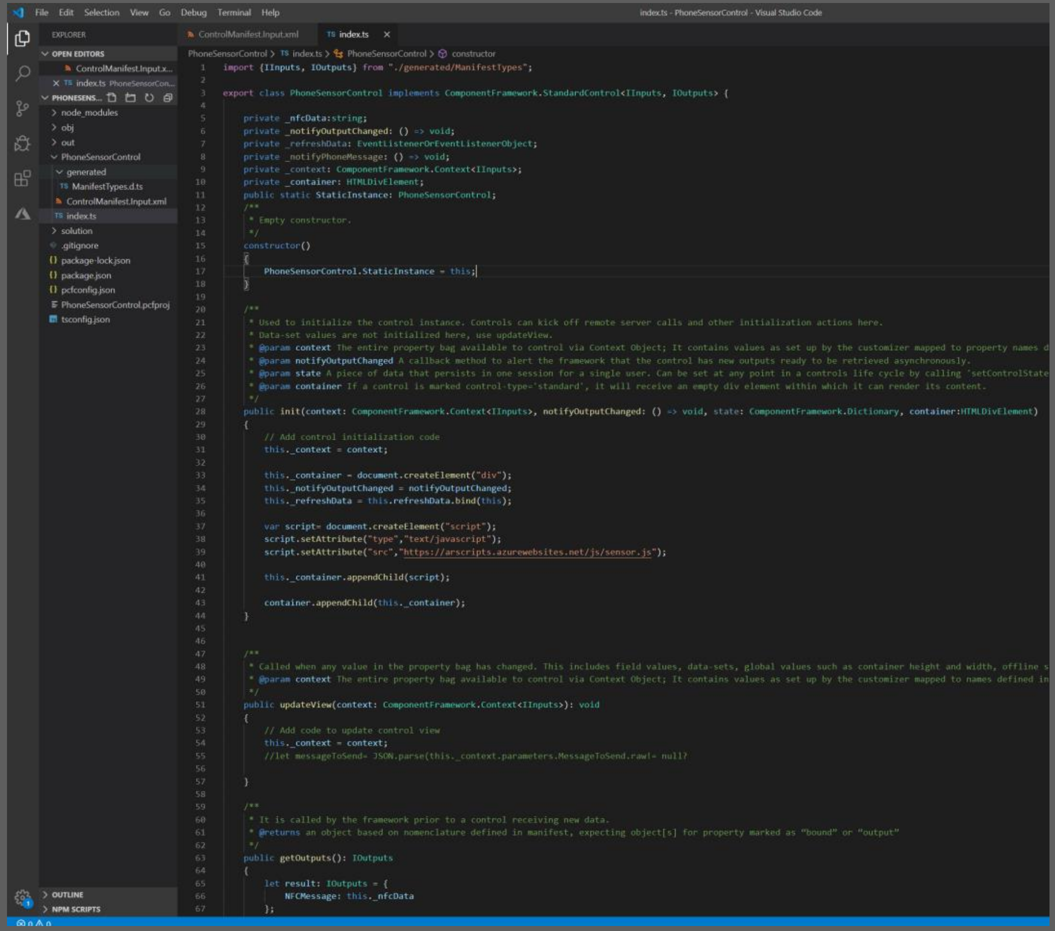
Create my own Power Apps native mobile App

Inject JavaScript code in the WebView to send information from the native code (sensor) to javascript world

Use PCF to inject JavaScript code to receive the messages from the previous step (iframe <-> host communication)



```java
@Override
protected void onCreate(Bundle savedInstanceState) {...}

private void configureWebView(String url)
{
    WebSettings webSettings = wv.getSettings();
    webSettings.setJavaScriptEnabled(true);
    webSettings.setDomStorageEnabled(true);

    wv.setWebViewClient(onPageFinished(view, url) → {
            super.onPageFinished(view, url);

            StringBuilder sb = new StringBuilder();
            sb.append("function NCFReceivedData(source,text){ var o = document.getElementsByTagName('iframe')[0]; o.contentWindow.postMessage(text, '*');  }; ");

                sb.append(" function callMe(text) { console.log(text); };  ");
                sb.append("window.ruiMessage = function(event) {callMe(event.data); if(event.data=='1')NFC.startNFC('hi there2'); };");

                sb.append("window.addEventListener(\"message\",window.ruiMessage);");

                view.loadUrl("javascript:" + sb.toString());
        }
    );

    if (android.os.Build.VERSION.SDK_INT >= 21) {
        CookieManager.getInstance().setAcceptThirdPartyCookies(wv,  accept: true);
    } else {
        CookieManager.getInstance().setAcceptCookie(true);
    }

    wv.addJavascriptInterface(startNFC(strl) → {
            Toast.makeText( context: MainActivity.this, strl, Toast.LENGTH_SHORT).show();
    },  name: "NFC");

    wv.loadUrl(url);

}

private void CallJSFunction(String functionName, String functionParameters)
{
    wv.evaluateJavascript( script: functionName+"('nfc','"+functionParameters+"');",  resultCallback: null);
}

private void initNFC(){

    mNfcAdapter = NfcAdapter.getDefaultAdapter(this);
}
```



```typescript
import {IInputs, IOutputs} from "./generated/ManifestTypes";

export class PhoneSensorControl implements ComponentFramework.StandardControl<IInputs, IOutputs> {

    private _nfcData:string;
    private _notifyOutputChanged: () => void;
    private _refreshData: EventListenerOrEventListenerObject;
    private _notifyPhoneMessage: () => void;
    private _context: ComponentFramework.Context<IInputs>;
    private _container: HTMLDivElement;
    public static StaticInstance: PhoneSensorControl;
    /**
     * Empty constructor.
     */
    constructor()
    {
        PhoneSensorControl.StaticInstance = this;
    }

    /**
     * Used to initialize the control instance. Controls can kick off remote server calls and other initialization actions here.
     * Data-set values are not initialized here, use updateView.
     * @param context The entire property bag available to control via Context Object; It contains values as set up by the customizer mapped to property names d
     * @param notifyOutputChanged A callback method to alert the framework that the control has new outputs ready to be retrieved asynchronously.
     * @param state A piece of data that persists in one session for a single user. Can be set at any point in a controls life cycle by calling 'setControlState
     * @param container If a control is marked control-type='standard', it will receive an empty div element within which it can render its content.
     */
    public init(context: ComponentFramework.Context<IInputs>, notifyOutputChanged: () => void, state: ComponentFramework.Dictionary, container:HTMLDivElement)
    {
        // Add control initialization code
        this._context = context;

        this._container = document.createElement("div");
        this._notifyOutputChanged = notifyOutputChanged;
        this._refreshData = this.refreshData.bind(this);

        var script= document.createElement("script");
        script.setAttribute("type","text/javascript");
        script.setAttribute("src","https://arscripts.azurewebsites.net/js/sensor.js");

        this._container.appendChild(script);

        container.appendChild(this._container);
    }


    /**
     * Called when any value in the property bag has changed. This includes field values, data-sets, global values such as container height and width, offline s
     * @param context The entire property bag available to control via Context Object; It contains values as set up by the customizer mapped to names defined in
     */
    public updateView(context: ComponentFramework.Context<IInputs>): void
    {
        // Add code to update control view
        this._context = context;
        //let messageToSend= JSON.parse(this._context.parameters.MessageToSend.raw!= null?

    }

    /**
     * It is called by the framework prior to a control receiving new data.
     * @returns an object based on nomenclature defined in manifest, expecting object[s] for property marked as "bound" or "output"
     */
    public getOutputs(): IOutputs
    {
        let result: IOutputs = {
            NFCMessage: this._nfcData
        };
```

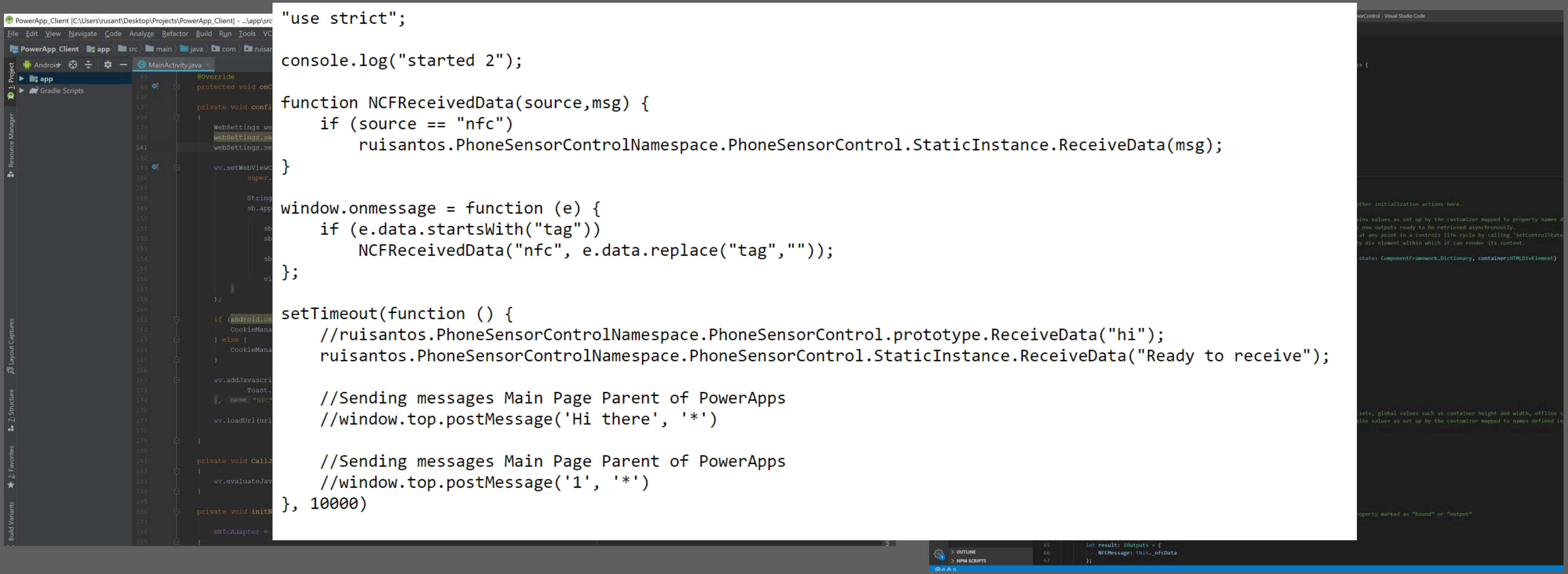
Hacking Power Apps

Microsoft

Create my own Power Apps native mobile App

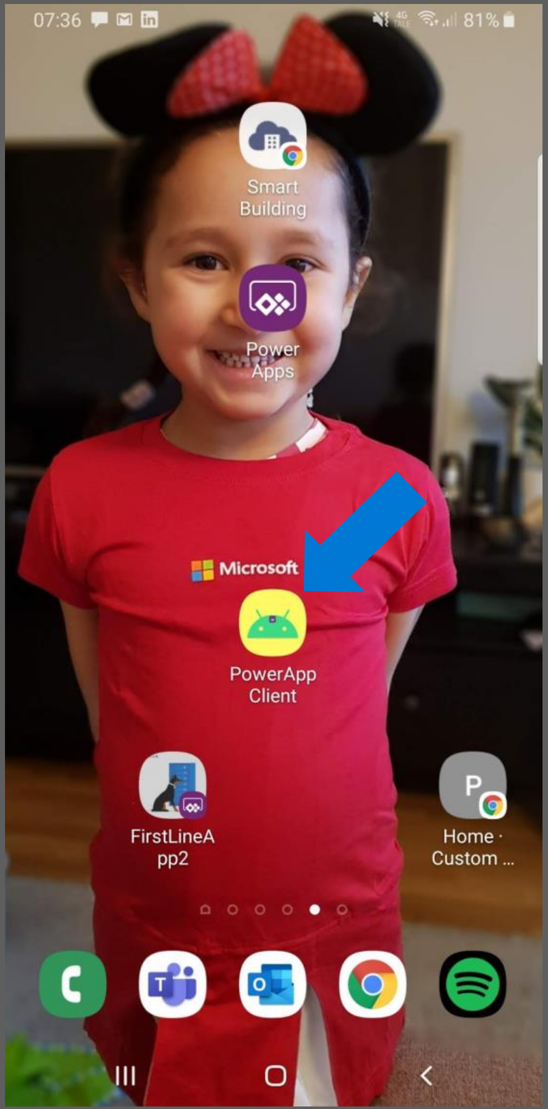
Inject JavaScript code in the WebView to send information from the native code (sensor) to javascript world

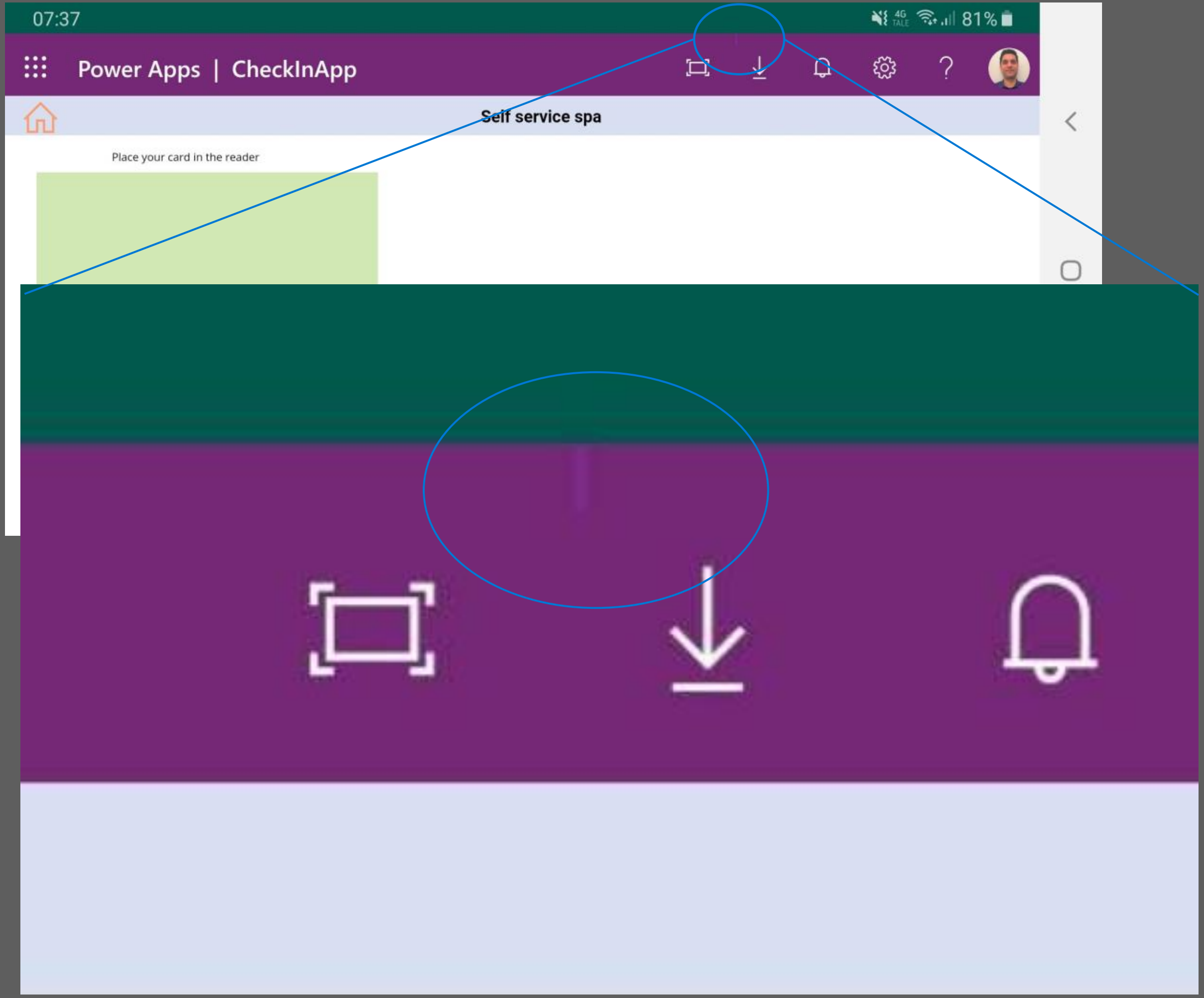
Use PCF to inject JavaScript code to receive the messages from the previous step (iframe <-> host communication)



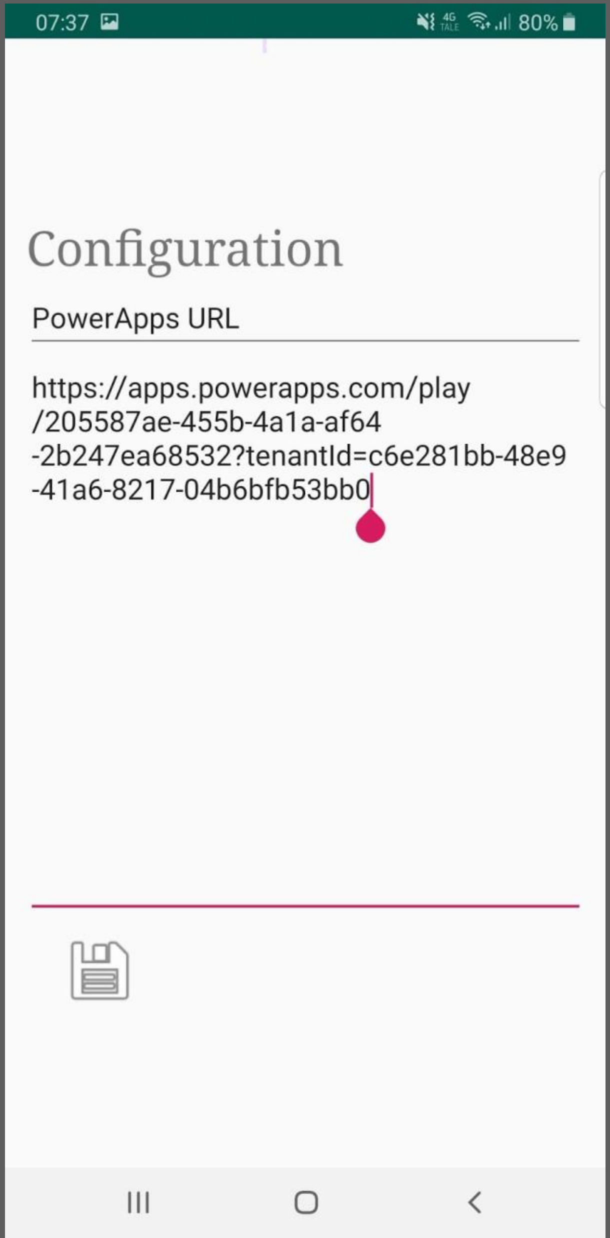
```
"use strict";

console.log("started 2");

function NCFReceivedData(source,msg) {
    if (source == "nfc")
        ruisantos.PhoneSensorControlNamespace.PhoneSensorControl.StaticInstance.ReceiveData(msg);
}

window.onmessage = function (e) {
    if (e.data.startsWith("tag"))
        NCFReceivedData("nfc", e.data.replace("tag",""));
};

setTimeout(function () {
    //ruisantos.PhoneSensorControlNamespace.PhoneSensorControl.prototype.ReceiveData("hi");
    ruisantos.PhoneSensorControlNamespace.PhoneSensorControl.StaticInstance.ReceiveData("Ready to receive");

    //Sending messages Main Page Parent of PowerApps
    //window.top.postMessage('Hi there', '*')

    //Sending messages Main Page Parent of PowerApps
    //window.top.postMessage('1', '*')
}, 10000)
```

My Native Power Apps client





Place your card in the reader



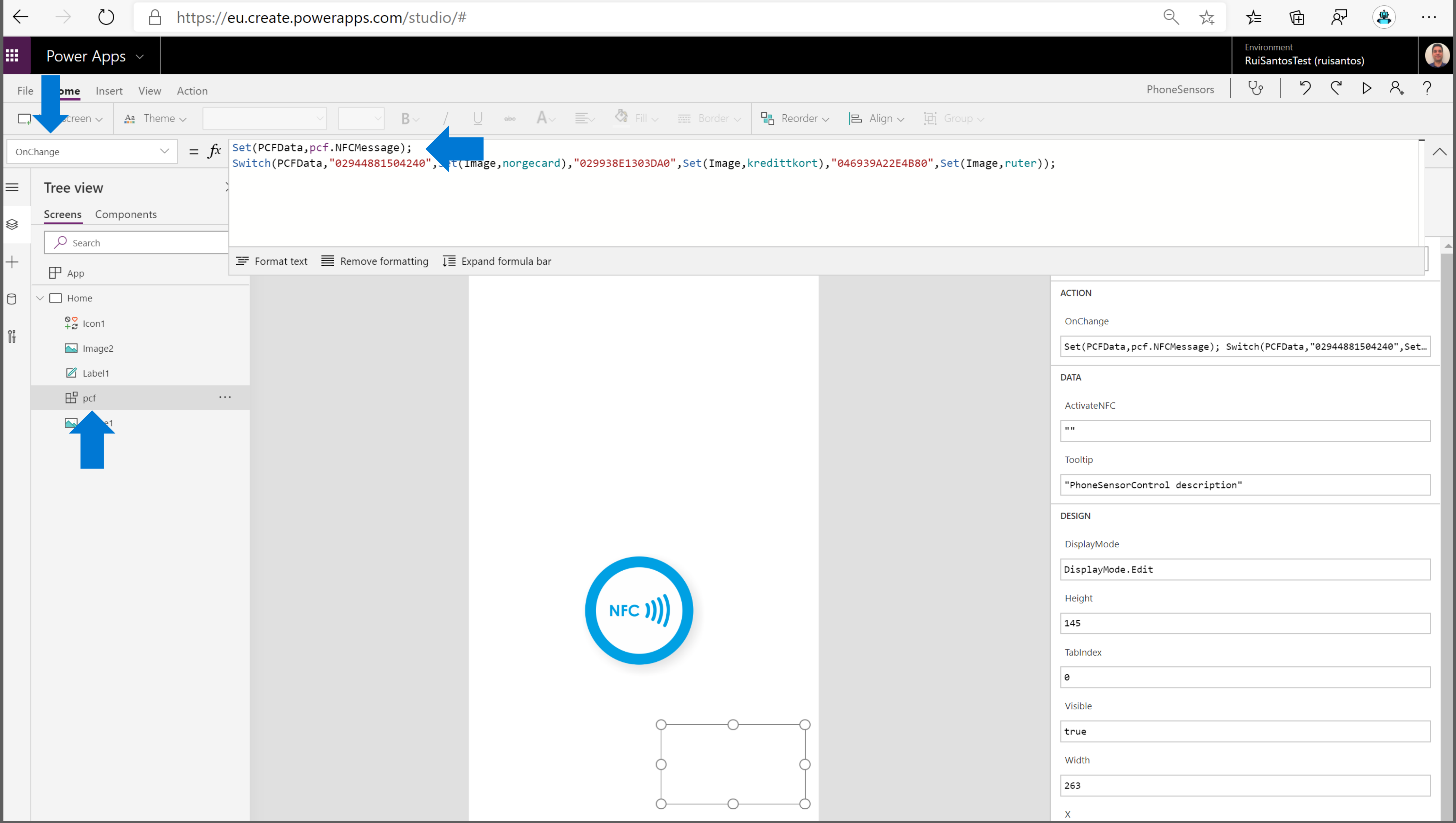
Configuration

PowerApps URL

https://apps.powerapps.com/play/205587ae-455b-4a1a-af64-2b247ea68532?tenantId=c6e281bb-48e9-41a6-8217-04b6bfb53bb0

Microsoft

Power Apps PCF



```
Set(PCFData,pcf.NFCMessage);
Switch(PCFData,"02944881504240",Set(Image,norgecard),"029938E1303DA0",Set(Image,kredittkort),"046939A22E4B80",Set(Image,ruter));
```

ACTION

OnChange: Set(PCFData,pcf.NFCMessage); Switch(PCFData,"02944881504240",Set…

DATA

ActivateNFC: ""

Tooltip: "PhoneSensorControl description"

DESIGN

DisplayMode: DisplayMode.Edit

Height: 145

TabIndex: 0

Visible: true

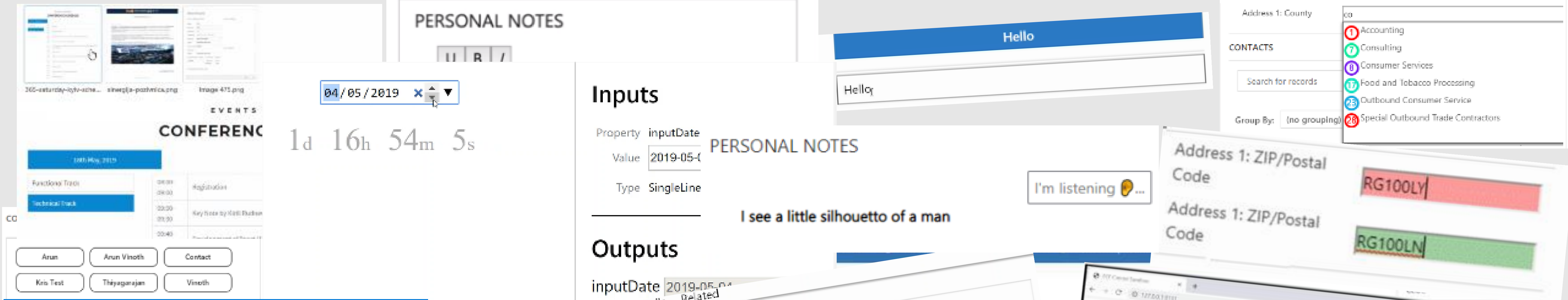
Width: 263

X

Source code

https://github.com/rsantos00/powerapps/

-NativeAppIntegration



PERSONAL NOTES

04/05/2019

1d 16h 54m 5s

EVENTS

CONFERENC

18th May 2019

Functional Track

Technical Track

Registration

Hello

Hello

Address 1: County

co

1 Accounting

7 Consulting

8 Consumer Services

13 Food and Tobacco Processing

25 Outbound Consumer Service

28 Special Outbound Trade Contractors

CONTACTS

Search for records

Group By: (no grouping)

Inputs

Property inputDate

Value 2019-05-0

Type SingleLine

Outputs

inputDate 2019-05-0

PERSONAL NOTES

I'm listening

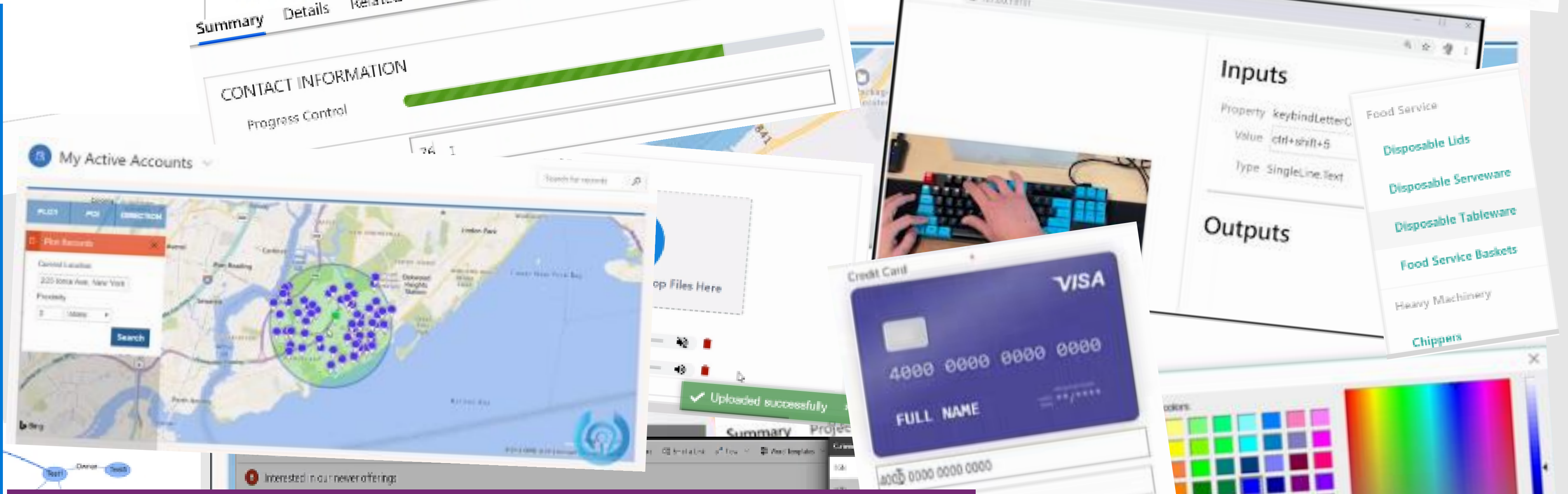
I see a little silhouette of a man

Address 1: ZIP/Postal Code RG100LY

Address 1: ZIP/Postal Code RG100LN

Arun | Arun Vinoth | Contact

Kris Test | Thiyagarajan | Vinoth



Summary Details Related

CONTACT INFORMATION

Progress Control

My Active Accounts

Search

Drop Files Here

Uploaded successfully

Interested in our newer offerings

Inputs

Property keybindLetterC

Value ctrl+shift+6

Type SingleLine.Text

Outputs

Food Service

Disposable Lids

Disposable Serveware

Disposable Tableware

Food Service Baskets

Heavy Machinery

Chippers

Credit Card

VISA

4000 0000 0000 0000

FULL NAME

CVC

Custom colors

Hue: 160 Sat: 240 Lum: 120

Red: 0 Green: 0 Blue: 255

Define Custom Colors >>

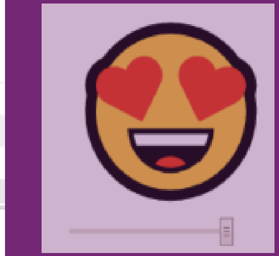
OK Cancel

Add to Custom Colors

Rui Santos

Technical Specialist BizApps (CE)
rui.santos@microsoft.com

LinkedIn



Thank You