## Useful links

<https://harikrishnan-devloper.blogspot.com/2020/05/kony-push-notification-registration-and.html>

<http://codingwithgireesh.blogspot.com/2019/06/kony-push-notification-using-fcm.html>

<https://abdulkareemkony.blogspot.com/2018/10/kony-overview.html>

<https://basecamp.temenos.com/s/question/0D56A00000gu8soSAA/how-to-extract-kar-file-manually>

<https://chandkony.blogspot.com/>

**More details:**

3) Objects Services

4) Matrix mutiplication, amstrong number,

5) Deeplink

6) Orchestration

7) MVC

1.Application life cycle

What is init() in this life cycle

What is a Appservice and DeepLink

2.Segment Methods AddAt()

3.Splice method in javascript var arr = [1,2,4,5,6] Above array should be returned as [1,2,3,4,5,6] by using arr.splice(3,0,2)

4.What is the return type of preprocessor and postprocessor

5.what is the method we use in preprocessor and postprocessor

6.What is a integration service

7.What is a Identity service

8.Push notifications

9.javascript concepts and Java concepts

2.How to get current location in kony

4.What is the difference between MVC normal project and DBX project

5.How to stop preprocessor when the input value is less than 20

7.Is it possible to invoke integration service without object service in DBX project

8.Do you know how to configure MF in the client server

9.FFI implemention

1. Session handling and authenticating the user based on the session id in the java code

2. What is the use of Identity Service

3. What is the difference between DataPreProcessor and DataPreProcessor2

4. What is the use od Enable pass-through

5. How to integrate a service with out using MF Addatdata: Setdataat: Alternateskin in segment: { "lat": "17.4414", "lon": "78.3809" } Aplicable widgets in segmen

## Difference between splice and slice in javascript:

<https://www.tothenew.com/blog/javascript-splice-vs-slice/>

|  |  |
| --- | --- |
| **SPLICE** | **SLICE** |
| returns the removed item(s) in an array | slice() method returns the selected element(s) in an array, as a new array object. |
| Original array is altered | Original array is unaltered bcoz it creates a new array object |
| Can take n number of arguments: | Only 2 arguments |
| var array=[1,2,3,4,5];  console.log(array.splice(2));  // shows [3, 4, 5], returned removed item(s) as a new array object.    console.log(array);  // shows [1, 2], original array altered.    var array2=[6,7,8,9,0];  console.log(array2.splice(2,1));  // shows [8]    console.log(array2.splice(2,0));  //shows [] , as no item(s) removed.    console.log(array2);  // shows [6,7,9,0]    var array3=[11,12,13,14,15];  console.log(array3.splice(2,1,"Hello","World"));  // shows [13]    console.log(array3);  // shows [11, 12, "Hello", "World", 14, 15] | var array=[1,2,3,4,5]  console.log(array.slice(2));  // shows [3, 4, 5], returned selected element(s).    console.log(array.slice(-2));  // shows [4, 5], returned selected element(s).  console.log(array);  // shows [1, 2, 3, 4, 5], original array remains intact.    var array2=[6,7,8,9,0];  console.log(array2.slice(2,4));  // shows [8, 9] |

## What is the difference between Let and Var in Javascript

var is function scoped and let is block scoped (less scope).

let a = 'hello'; // globally scoped

var b = 'world'; // globally scoped

console.log(window.a); // undefined

console.log(window.b); // 'world'

var a = 'hello';

var a = 'world'; // No problem, 'hello' is replaced.

let b = 'hello';

let b = 'world'; // SyntaxError: Identifier 'b' has already been declared

## ****Check**** if Network Is Available:

//return true or false

function checkIfNetworkIsAvailable() {

return kony.net.isNetworkAvailable(constants.NETWORK\_TYPE\_ANY);

}

## Read Raw file data using Kony visualizer

function fileReadOptions(){

try{

var destFilePath = kony.io.FileSystem.getDataDirectoryPath()+"/getdetails.json";

var myFile = new kony.io.File(destFilePath).createFile();

alert("Path is "+destFilePath);

var fileObj = null;

try{

var file = new kony.io.File(destFilePath);

//copyBundledRawFileTo API overrides the destination file with new one.

//Hence check before copying

if(!file.exists()){

fileObj = kony.io.FileSystem.copyBundledRawFileTo ("getdetails.json", destFilePath);

}else{

// fileObj = file;

fileObj = kony.io.FileSystem.copyBundledRawFileTo ("getdetails.json", destFilePath);

var reading = new kony.io.File(destFilePath).read();

alert(JSON.stringify(reading));

alert("readAsText() "+reading.readAsText());

}

}catch(e) {

kony.print("Exception "+e);

}

}catch(err){

alert("Error : "+JSON.stringify(err));

}

}

## Kony-Local Storage NameSapce API Elements:

|  |
| --- |
| Syntax :-  **1. kony.store.setItem(key, value)**    key [string] - Mandatory  Specifies the keyname for which the item needs to be set  value [object] - Mandatory  Specifies the value that must be set at the given index. This value can be a number,  string,Boolean  Example :  kony.store.setItem("keyValue5", "this is a key value"); //string  kony.store.setItem("keyValue4", true); //boolean    kony.store.**getItem**(keyname)  Specifies the keyname from which the item needs to be fetched.  Example :    var myValue = kony.store.getItem("name");  alert("name is "+myValue);    kony.store.**removeItem**(keyname)    Specifies the keyname for which the item needs to be removed.  Example :  kony.store.removeItem("name");  alert("name removed");    kony.store.**clear**()  clearing all the key-value pairs    kony.store.**key**(index)    Specifies the index for which the key name is to be returned.  Example :    var keyName = kony.store.key(0);  alert("first key name is "+keyName);  kony.store.**length**()  Returns the length of the local storage. |

1) Pre-Processor,Post processor

## App Life Cycle Events

**Pre Appinit. Pre-initialization.**

This event consists of any logic that needs to be executed before the initialization of forms.

**Init**

Is generated by the code and consists of the form and skin initialization data

**Post Appinit. Post-initialization**

This function to define logic that needs to be executed before the first form is shown and after the application is initialized

**App Service (eg. GPay, Paytm etc.,)**

This function also returns the form handle and launching of background services(launching another form)

**Deeplink (eg,: push notification)**

Refers to creating services links in your app that enable third-party native and browser applications to connect to your app.

## Form LifeCycle Event

1) init

a) Initializations required for the form.

b) Init initializes the form and any widgets.

2) preShow

a) called just before a form is visible on the screen.

3) postShow

a) called immediately after the form is visible on the screen.

4) onHide

a) called when the form goes out of the screen. A form can go out of the screen when another form is to be shown.

5) onDestroy

a) called when a form is destroyed.

## Why using Gesture in Kony?

A gesture is an action associated with **movement of a mouse or a touch screen** action

Currently, Kony Platform supports only Tap, Swipe, Longpress, Pan, Rotation, Pinch, and 3D Touch gesture types. 3D Touch is supported only in iOS 9.0 and later

The gesture methods are:

1. addGestureRecognizer

2. removeGestureRecognizer

## Cryptography:

1. kony.crypto.createHash
2. kony.crypto.newKey
3. kony.crypto.encrypt
4. *kony.crypto.decrypt*

### kony.crypto.createHash(hashingAlgorithm,inputstring,options)

| Hashing Algorithm | Result Length (in bytes) | Result Length (in hexadecimal characters) |
| --- | --- | --- |
| *sha1* | *20* | *40* |
| *sha224* | *28* | *56* |
| *sha256* | *32* | *64* |
| *sha384* | *48* | *96* |
| *sha512* | *64* | *128* |
| *md2* | *16* | *32* |
| *md4* | *16* | *32* |
| *md5* | *16* | *32* |

*sha512: Secure Hash Algorithm 224 (SHA-512).*

*md2: Message-Digest Algorithm 2 (MD2).*

**Example Code :**

*Key :: md5 algorithm*

*function genhashKEY() {*

*try {*

*var algorithm\_key = "md5";*

*var hashKEY = kony.crypto.createHash(algorithm\_key, "chandrupass");*

*alert("Result : " + hashKEY);*

*} catch (err) {*

*alert("Error" + err);*

*}*

*}*

### kony.crypto.newKey

function genAEGALGM()

{

var algo = "aes";

// kony.crypto.newKey

var encryptDecryptKey = kony.crypto.newKey(

"passphrase",

128, {

passphrasetext: ["inputstring1"],

subalgo: "aes",

passphrasehashalgo: "aes"

});

var inputstring = "chandrupass";

alert("Before Encrypt : " + inputstring);

var prptobj = {

padding: "pkcs5",

mode: "cbc",

initializationvector: "1234567890123456"

};

### kony.crypto.encrypt

var myEncryptedText = kony.crypto.encrypt(

algo,

encryptDecryptKey,

inputstring,

prptobj);

alert("ChiperText : " + JSON.stringify(myEncryptedText));

### kony.crypto.decrypt

var myClearText = kony.crypto.decrypt(algo, encryptDecryptKey, myEncryptedText, prptobj);

alert("Original Text : " + myClearText);

}

## How to get the Application Version Code using Kony NFI

function getVersionControl (){

try{

var KonyMain = java.import("com.konylabs.android.KonyMain");

var packageManager = java.import("android.content.pm.PackageManager");

packageManager = KonyMain.getAppContext().getPackageManager();

if (packageManager !== null) {

try {

packageInfo = packageManager.getPackageInfo(KonyMain.getAppContext().getPackageName(), 0);

var versionName = packageInfo.versionName;

kony.print("Result print is : "+versionName);

var versionCode = packageInfo.versionCode;

kony.print("Result print is : "+versionCode);

} catch (eee) {

kony.print("Exception is : "+JSON.stringify(eee));

}

}

}catch(err){

kony.print("Error Exception is : "+JSON.stringify(err));

}

}

## How to open PDF using Intent in KONY Visualizer?

var pdfFilePath = "storage/emulated/0/filename.pdf";  
 function openPDFFile(pdfFilePath){   
 try{  
        var KonyMain = java.import("com.konylabs.android.KonyMain");  
        var File = java.import("java.io.File");  
        var Intent = java.import("android.content.Intent");  
        var Uri = java.import("android.net.Uri");  
        var file = new File(pdfFilePath);  
        var intentObject = new Intent(Intent.ACTION\_VIEW);     
        intentObject.setDataAndType(Uri.fromFile(file), "application/pdf");  
        intentObject.setFlags(Intent.FLAG\_ACTIVITY\_NO\_HISTORY);  
        var contextObject = KonyMain.getActContext();  
        contextObject.startActivity(intentObject);  
      }catch(err){  
        kony.print("Error in "+JSON.stringify(err));  
      }  
    }

## How to call Another Controller function

**firstControllerfrom.js**  
  
define(function(){  
  return {  
    onNavigate : function(context){  
      alert("Entering");  
    },  
    passDataAnother : function(){  
      return "chandru";  
    }  
  };  
});  
  
**secondControllerfrom.js**  
  
define(function(){  
  
  return {  
    onNavigate : function(context){  
      kony.print("Entering");  
    },  
    callAnotherContr : function(){  
  
      var controllerRef = **require**("firstControllerfrom");  
      var result = controllerRef.passDataAnother ();  
  
      alert("welcome "+result);  
    }  
  };  
});

## Hit the service without Mobile Fabric in Kony visualizer?

define({

onNavigate : function(objectEvent){

this.view.btnServiceCallEvent.onClick = this.onServiceCallEventFun;

},

onServiceCallEventFun : function(){

var httpclient = new kony.net.HttpRequest();

httpclient.open(constants.HTTP\_METHOD\_POST, "https://neutrinoapi.com/email-validate");

httpclient.setRequestHeader("user-id", "chandru");

httpclient.setRequestHeader("api-key", "5wPeyo4c9dFcUIyWMTghQP80nj3k0uyJzgals0MNKQ9wAjyf");

httpclient.setRequestHeader("Content-Type", "application/json");

var postdata = {

"email":"chandbecse@gmail.com",

"fix-typos":true

};

httpclient.timeout = 20000;

httpclient.onReadyStateChange = function(response){

kony.print("Result valid ==> "+response.valid);

kony.print("Result provider ==> "+response.provider);

kony.print("Result domain ==> "+response.domain);

kony.print("Result Response ==> :"+JSON.stringify(response));

};

httpclient.send(postdata);

}

});

## Kony Push Notification using FCM

Steps

1) Create new project in kony visualizer 8.x version.

2) Go to firebase console, create a new project and select android.

Then provide the package name taken from kony visualizer->settings->native->android.

3) From firebase messaging part, note down the server key and sender id.

4) In kony visualizer, create functions for kony.push.register api with the sender id from firebase. and kony.push.setCallback api for all callback functions.

5) Go to kony mobile fabric -> Engagement. Click android and in the field for server key, provide the server key taken from firebase and save.

6) Publish application in mobile fabric.

7) Associate the fabric application to visualizer project.

8) Select FCM in push notification under Kony visualizer->settings->native->android.

9) Take build for android.

10) Once you call the kony.push.register api with the sender id, Your device will be registered in kony engagement server. So that from Adhoc, you will be able to send push notification to selected devices.

Program

var KMSPROP = {};

var pushCallbacks = {

onsuccessfulregistration: regSuccessCallback,

onfailureregistration: regFailureCallback,

onlinenotification: onlinePushNotificationCallback,

offlinenotification: offlinePushNotificationCallback,

onsuccessfulderegistration: unregSuccessCallback,

onfailurederegistration: unregFailureCallback

};

/\*

\* @function setAllCallbacks

\* this function will set the call backs for push notification

\*/

function setAllCallbacks() {

try {

kony.push.setCallbacks(pushCallbacks);

} catch (err) {

kony.print("KMS Module" + JSON.stringify(err));

}

}

/\*

\* @function registerKMS

\* this function will register the divice for GMS console and APS console using

\* register API

\*/

function registerKMS() {

try {

//#ifdef android

KMSPROP.osType = "androidgcm"; // ostype to sent if it is android

var configRegister = {

senderid: "sender id from firebase"

};

//#endif

//#ifdef iphone

KMSPROP.osType = "iphone";

var configRegister = [0, 1, 2];

//#endif

//this will check whether the device is al

if (kony.store.getItem("isRegisteredForKMS") === undefined || kony.store.getItem("isRegisteredForKMS") === "" || kony.store.getItem("isRegisteredForKMS") === null) {

kony.push.register(configRegister);

}

} catch (err) {

kony.print("KMS Module" + JSON.stringify(err));

}

}

/\*

\* @function regSuccessCallback

\* This function is registration success callback on successful registration of APS or GCM

\* this function is used to register to kony messaging for push notification

\*/

function regSuccessCallback(regId) {

try {

alert("reg successful");

kony.store.setItem("isRegisteredForKMS", "true");

KMSPROP.deviceId = kony.os.deviceInfo().deviceid; // device id

KMSPROP.device\_rec = kony.os.deviceInfo().deviceid; // unique key

var messagingSvc = KNYMobileFabric.getMessagingService();

messagingSvc.register(KMSPROP.osType, KMSPROP.deviceId, regId, KMSPROP.device\_rec, successCallbackForSubscribe, failureCallbackForSubscribe);

function successCallbackForSubscribe(res) {

kony.application.dismissLoadingScreen();

kony.print(JSON.stringify(res));

}

function failureCallbackForSubscribe(err) {

kony.store.removeItem("isRegisteredForKMS");

kony.application.dismissLoadingScreen();

kony.print(JSON.stringify(err));

kony.print("Subscription Failed.");

}

} catch (err) {

kony.print("KMS Module" + JSON.stringify(err));

}

}

/\*

\* @function regFailureCallback

\* This function is registration failure callback on failure registration of APS or GCM

\*/

function regFailureCallback(errormsg) {

kony.print(errormsg);

kony.print("Registration Failed ");

}

/\*

\* @function onlinePushNotificationCallback

\* This function is the callback for online push notification

\*/

function onlinePushNotificationCallback(msg) {

try {

alert("push message received"+JSON.stringify(msg));

} catch (err) {

kony.print("KMS Module" + JSON.stringify(err));

}

}

/\*

\* @function offlinePushNotificationCallback

\* This function is the callback for offline push notification

\*/

function offlinePushNotificationCallback(msg) {

try {

alert("push message received"+JSON.stringify(msg));

} catch (err) {

kony.print("KMS Module" + JSON.stringify(err));

}

}

function unregSuccessCallback() {

kony.print("Unregisterd Succesfully!!");

}

function unregFailureCallback(errormsg) {

kony.print(" Unregistration Failed!!" + errormsg);

}

Note: If you get package generation failed after selecting FCM in push notification, from firebase-> your apps, download the google-services.json file. Create a folder named "fcm" in project location->resources->mobile->native->android and paste the json file. Restart the visualizer and take build.