



AI & ML DRIVEN  
MARKETING  
AUTOMATION



# Integration document

## Cordova Development

*This document will help you integrate the appICE SDK in your Android/iOS Projects.*

---

## **Table of Contents**

<b>Setting up your app on //appice.io</b>	<b>4</b>
Sign-up	4
Setup your App	4
<b>Requirements</b>	<b>7</b>
<b>Installing appICE plugin in your project</b>	<b>8</b>
appICE plugin installation	8
Get The Keys	11
Initialise SDK	12
Send Events	12
Set User	13
SetUserId	13
Setup Push Notification	13
<b>Verify Push Notifications</b>	<b>15</b>
<b>Install App on your Android phone</b>	<b>22</b>
<b>Verify integration on appICE panel</b>	<b>23</b>

---

## Setting up your app on //appice.io

### Sign-up

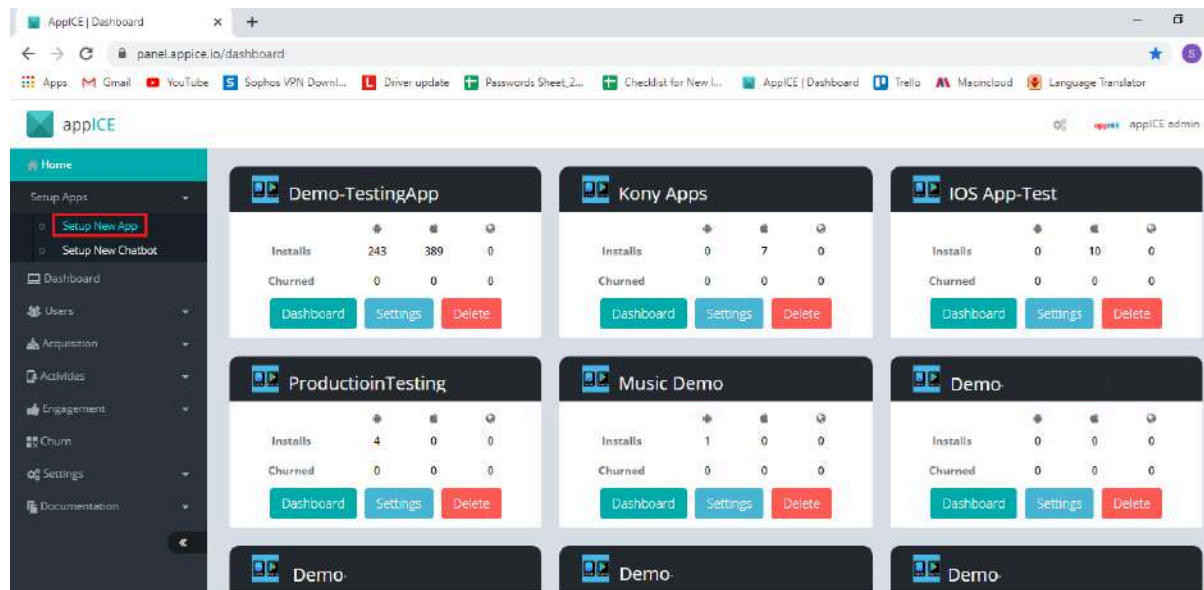
You need to sign-up and create an account with [www.appice.io](http://www.appice.io).

1. Visit <https://panel.appice.io/signup> to Register.
2. This brings you to the “Sign up” page.
3. Provide your Name, Email address and your chosen password and create your account.
4. Login using your username and password credentials once your account is approved.

### Setup your App

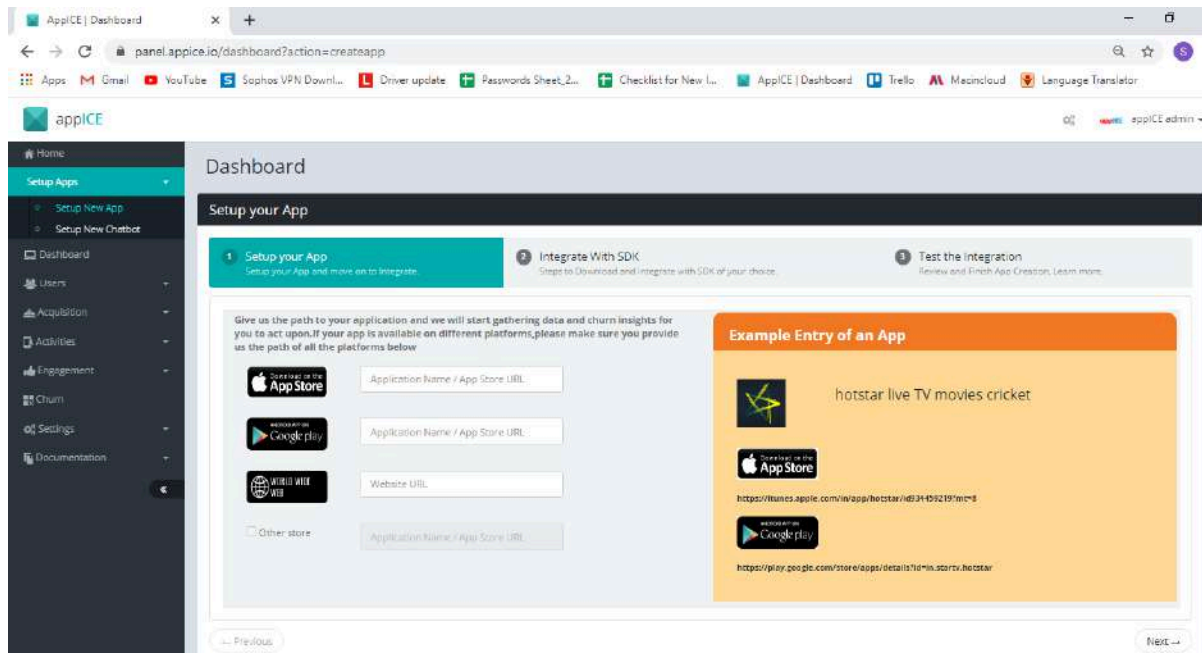
To start the process of integrating appICE in your app, you will first need to setup your app on the appICE dashboard.

1. Click on Setup New App in left panel.



2. Provide the link to your mobile app on the Google Play Store as well as Apple store if you have both versions of your app.

Click Next to go to next page.



AppICE | Dashboard

panel.appice.io/dashboard?action=createapp

Apps Gmail YouTube Sophos VPN Downl... Driver update Passwords Sheet\_2... Checklist for New L... AppICE | Dashboard Trello Macincloud Language Translator

appice

Home

Setup Apps

Setup New App

Setup New Chatbot

Dashboard

Users

Acquisition

Activities

Engagement

Churn

Settings

Documentation

Dashboard

Setup your App

1 Setup your App  
Setup your App and move on to Integrate.

2 Integrate With SDK  
Steps to Download and integrate with SDK of your choice.

3 Test the Integration  
Review and Finish App Creation, Learn more.

Give us the path to your application and we will start gathering data and churn insights for you to act upon. If your app is available on different platforms, please make sure you provide us the path of all the platforms below

Download on the App Store

Application Name / App Store URL

Get it on Google play

Application Name / App Store URL

Website URL

Other store

Application Name / App Store URL

Example Entry of an App

hotstar live TV movies cricket

Download on the App Store

https://itunes.apple.com/in/app/hotstar/id3445219?nc=s

Get it on Google play

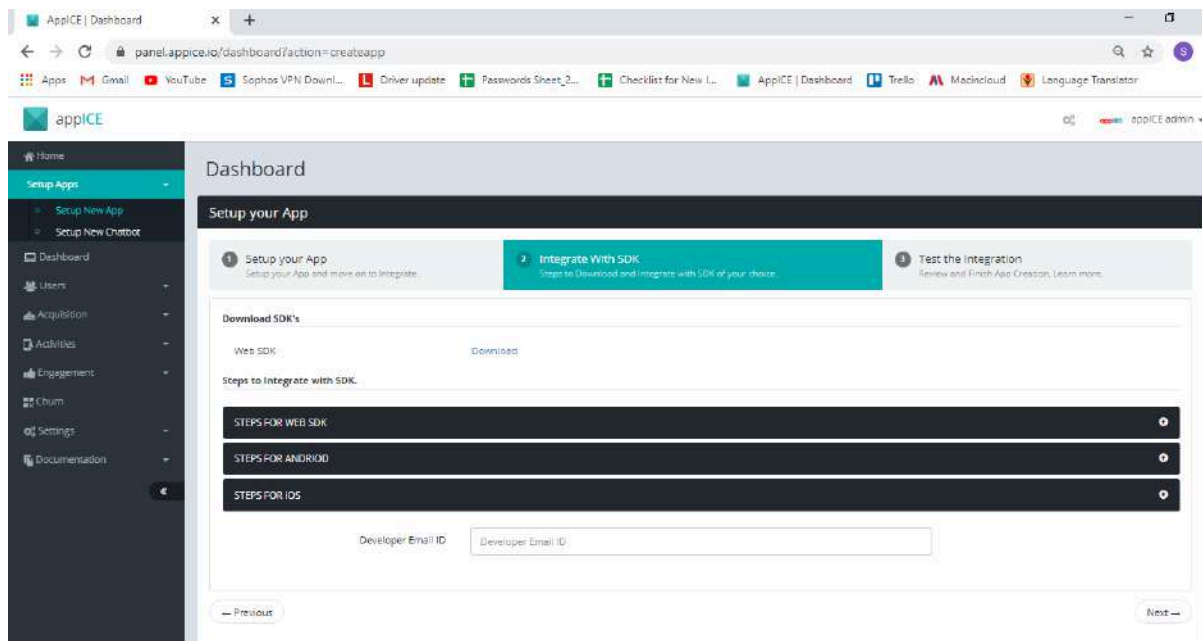
https://play.google.com/store/apps/details?id=in.startv.hotstar

Previous

Next

Click Next to go to next page.

- Now you have access to the SDK's for Cordova including the instructions for integration. To make it easier, we have also provided the option to provide your developer's email address so that those instructions can be emailed to them directly.



AppICE | Dashboard

panel.appice.io/dashboard?action=createapp

Apps Gmail YouTube Sophos VPN Downl... Driver update Passwords Sheet\_2... Checklist for New L... AppICE | Dashboard Trello Macincloud Language Translator

appice

Home

Setup Apps

Setup New App

Setup New Chatbot

Dashboard

Users

Acquisition

Activities

Engagement

Churn

Settings

Documentation

Dashboard

Setup your App

1 Setup your App  
Setup your App and move on to Integrate.

2 Integrate With SDK  
Steps to Download and integrate with SDK of your choice.

3 Test the Integration  
Review and Finish App Creation, Learn more.

Download SDK's

Web SDK

Download

Steps to Integrate with SDK

STEPS FOR WEB SDK

STEPS FOR ANDROID

STEPS FOR IOS

Developer Email ID

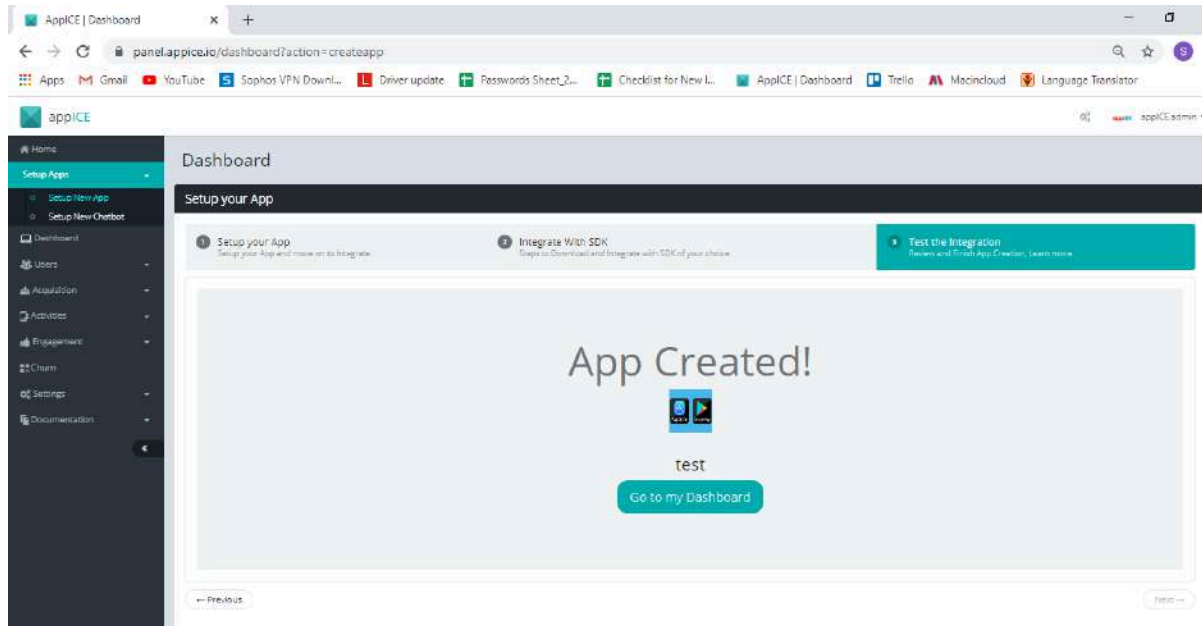
Developer Email ID

Previous

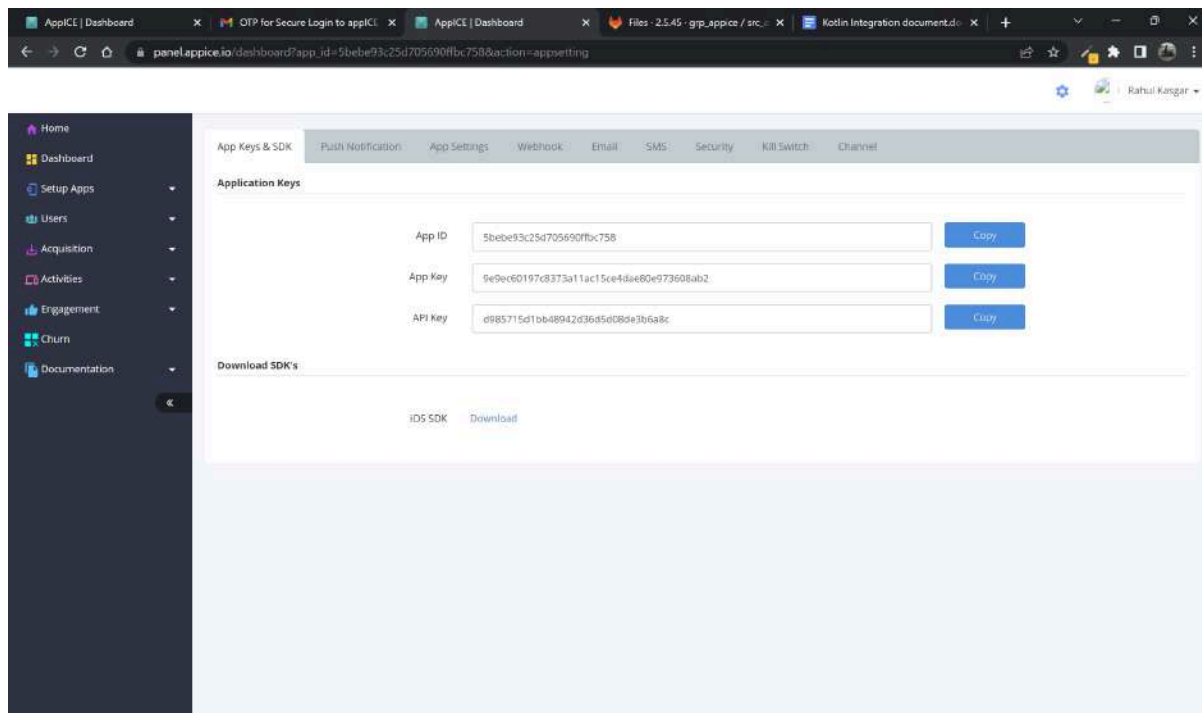
Next

Click Next to go to next page.

- Now you are all ready to start receiving data from your mobile app once the developer completes the integration and publishes the app again on the app store.



- Open your panel and select your app > App Settings  
You will find the App ID, App Key, API Key here.



## Requirements

To get started with Cordova App, the only requirement is a Node.js & npm environment.

Additionally:

1. Android Studio
2. Xcode for iOS
3. Visual Studio Code (recommended)
4. JDK 1.8

Install cocoapods

<https://guides.cocoapods.org/using/getting-started.html>

---

---

## Installing appICE plugin in your project

### appICE plugin installation

To add appICE Plugin in your project, run this command:

```
cordova plugin add cordova-plugin-appice --variable APPICE_APP_ID="APP_ID"
--variable APPICE_API_KEY="API_KEY" --variable APPICE_APP_KEY="APP_KEY"
```

Now we have successfully added the Appice plugin.

#### **NOTE:**

1. Add these lines in the android manifest file for ssl pinning, encryption.

***..\android\app\src\main\AndroidManifest.xml***

```
<service android:exported="false" android:name="semusi.activitysdk.Api"
android:protectionLevel="signature" />
```

```
<meta-data android:name="APPICE_ENCRYPTION" android:value="1" />
```

```
<meta-data android:name="APPICE_SSL_PINNING" android:value="1" />
```

2. For apps targeting Android 13 and above, you must add the com.google.android.gms.permission.AD\_ID permission in the AndroidManifest.xml

```
"<uses-permission
android:name="com.google.android.gms.permission.AD_ID"/>"
```

### Add the below snippet in following Class:

In app's Application class

@Override

```
public void onCreate() {
super.onCreate();
ContextApplication.initSdk(context, application);
}
```



---

Add these lines in build.gradle app level. If it already exists, skip.

```
repositories {  
    maven { url "https://jitpack.io" }  
}
```

Also add these dependencies:

1. implementation "androidx.work:work-runtime:2.7.0"
2. implementation "com.google.android.gms:play-services-location:21.0.1"
3. implementation  
"com.gitlab.grp\_appice.src\_android\_appice-sdk:cordova:2.5.56"
4. Implementation "com.gitlab.grp\_appice.src\_android\_appice-sdk:sdk:2.5.56"

## iOS

There are two ways to integrate ios sdk-PODSPEC & manually.

**POD update-enter into this path /platforms/ios install pod and update.**

//To go inside ios platform

**cd /platforms/ios**

//to initialize pod

**pod init or pod install**

//to update pod

**pod update**

1. libiosAppiceSDK\_v1.7.26.a is our sdk for ios it should be present in appicesdk folder.
  2. We should have APPICESDK folder inside src/ios folder where will have all files which are imported in plugin.xml.
-

## inside plist location permission

```
<key>NSLocationAlwaysUsageDescription</key>
<string>App needs your location to provide enhanced
features.</string>
<key>NSLocationUsageDescription</key>
<string>App needs your location to provide enhanced
features.</string>
<key>NSLocationWhenInUseUsageDescription</key>
<string>App needs your location to provide enhanced
features.</string>
```

***The SSL Pinning certificate should be added inside the copy bundle Resources module.***

### ***App Metadata -***

```
<key>AppICEAppID</key>
<string>6042074e12c29c18fb963b9f</string>
<key>AppICEAppKey</key>
<string>8d92800c02e405c303aca5b4d00acae6cdb342a7</string>
<key>AppICEApiKey</key>
<string>73e76506c9d58f351eed67fac534e2e1</string>
```

---

## For SSL pinning -

```
<key>APPICE_SSL_PINNING</key>

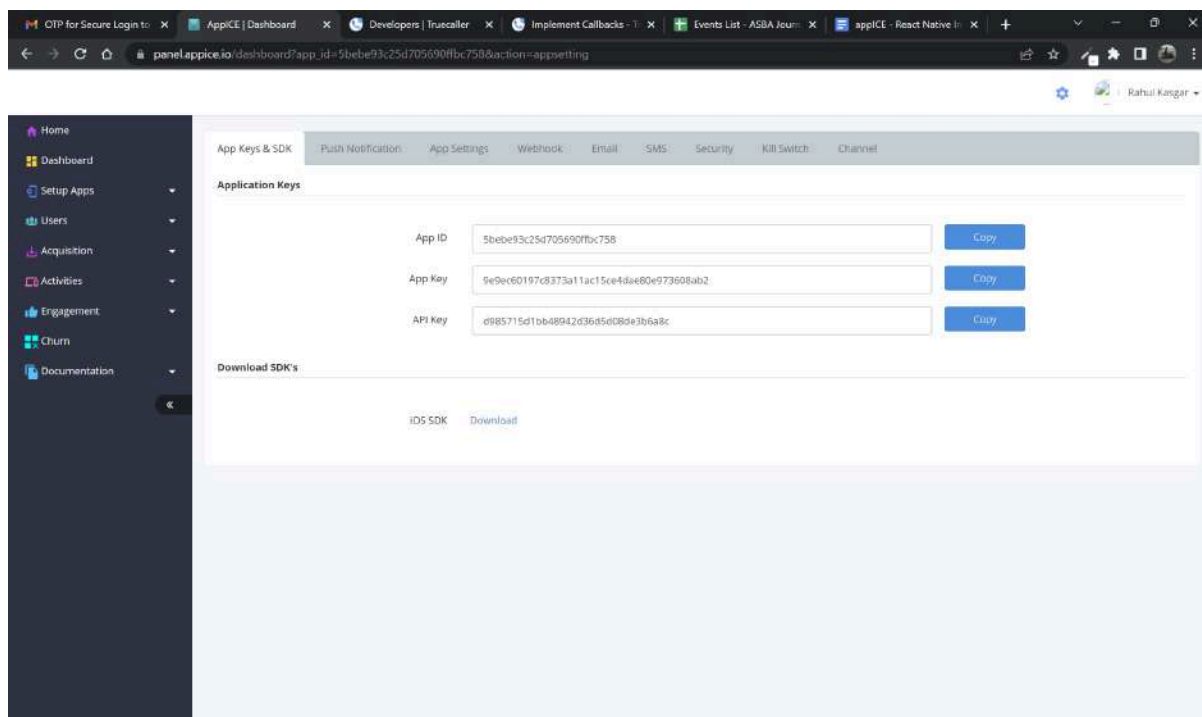
<true/>
```

## For Session Encryption, we have:

```
<key>SESSION_ENCRYPTION</key>

<true/>
```

## Get The Keys



Now Copy the keys and paste them into code to initialise the SDK

---

## **Initialise SDK**

```
var platformName =window.cordova.platformId
var cert=[];
if( platformName ==="android"){
  cert=["assets/ai_android.pem"];
}else{
  cert=["Resources/ai_ios.der"];
}
```

These values are APP\_ID, APP\_KEY,API\_KEY respectively which you can get from app metadata.

i.e from android manifest in Android and info.plist in iOS.

```
AppICE.initSdk(
"5bebe93c25d705690ffbc758", //use the App Metadata for this that you created above
"9e9ec60197c8373a11ac15ce4dae80e973608ab2",
"D985715d1bb48942d36d5d08de3b6a8c" ,"",
"US",
"https://a.appice.io",
cert,
function(success) {
  console.log("init success: " + JSON.stringify(success));

}, function(error) {
  console.error("Init Error : " + error);
});
```

## **Send Events**

```
var dataObj = {
  "ClickedMenuBtn":"false",
  "CLickedSubmitBtn":"false",
  "AppName":"SampleApp"
};
AppICE.tagEvent("SampleApp", dataObj,
function(success) {
  console.log("tagEvent success: " + JSON.stringify(success));

}, function(error) {
```

---

```
console.error("tagEvent Error : " + error);  
});
```

### **Set User**

```
AppICE.setUser("", phone,  
email,  
function (success) {  
  console.log("set user success",success);  
},  
function (error) {  
  console.log(" set user error",error);  
});
```

### **SetUserId**

```
const userids = [userid];  
  
AppICE.setUserId(userids, function (success) {  
  console.log("set user id success");  
},  
function (error) {  
  console.log("set user Failed");  
});
```

### **Setup Push Notification**

#### **Add google-services.json in *platforms/android/app* for android**

In order to setup push we need to Integrate mfp in our application

In config file these entries for mfp

```
<preference name="FIREBASE_UPDATE" value="false" />
```

---

```
<preference name="APPICE_APPLICATIONCLASS_INCLUDE" value="false" />
<preference name="MFP_PUSH" value="true" />
<preference name="SSL_PINNING" value="true" />
```

**Add these entries in android manifest:**

```
<service android:exported="false"
android:name="com.appice.appicemfppush.AppICEMFPPush"
tools:replace="android:exported">
  <intent-filter>
    <action android:name="com.google.firebase.MESSAGING_EVENT" />
  </intent-filter>
</service>
```

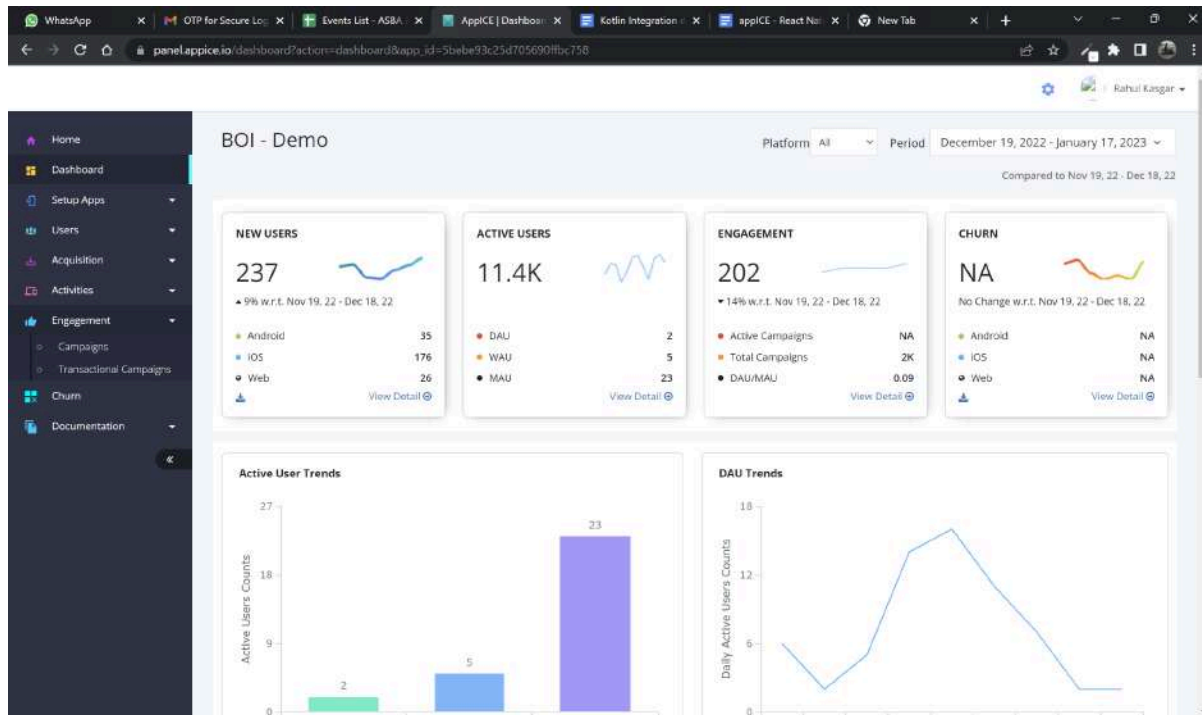
**Add Firebase dependencies in Android build.gradle file:**

```
appice.io.android:libmfp:1.0.0
```

---

# Verify Push Notifications

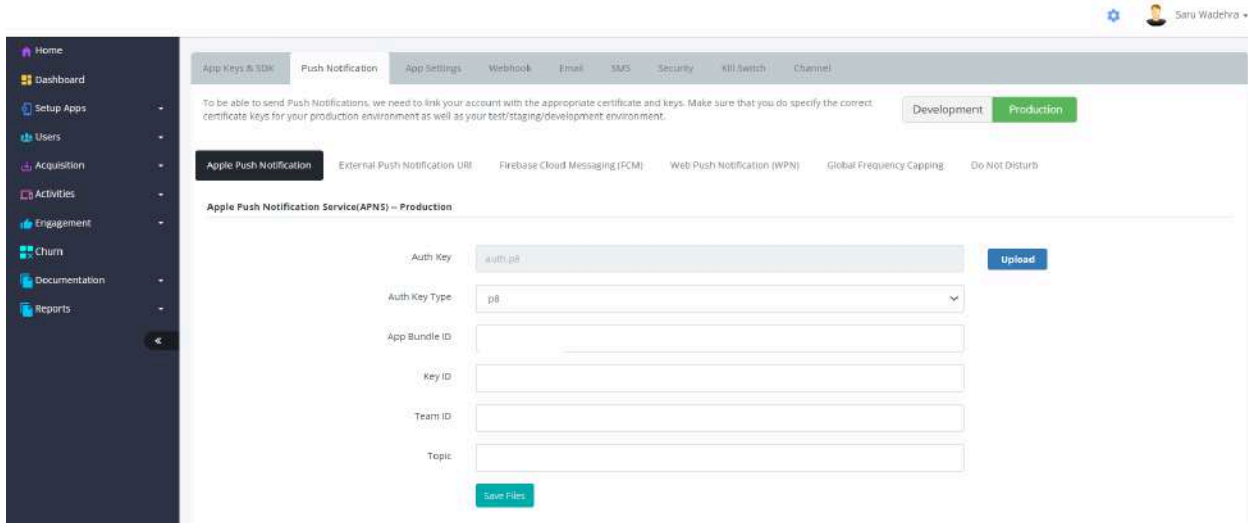
1. <https://panel.appice.io/login>



2. Go to Settings → App Setting



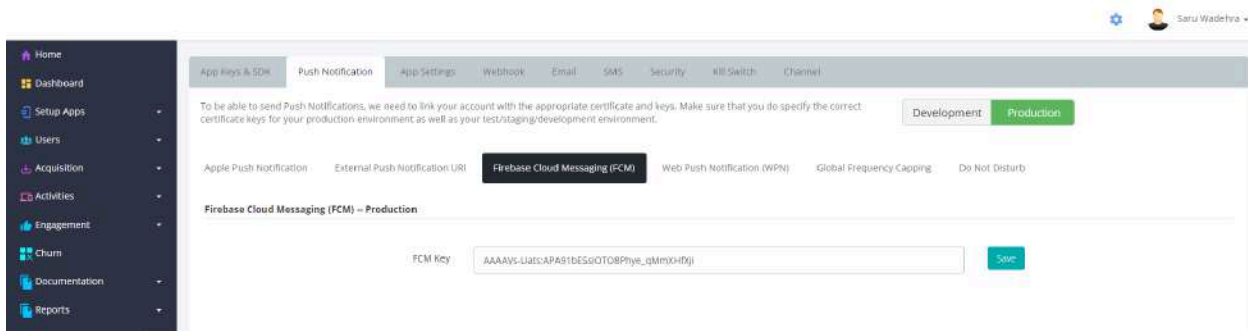
### 3. Goto Push Notification → Apple Push Notification and put required details.



The screenshot shows the AppIC8 interface with the 'Push Notification' tab selected. The 'Apple Push Notification' sub-tab is active, and the 'Production' environment is chosen. The form contains the following fields:

- Auth Key:** A text input field containing 'auth.p8' and an 'Upload' button.
- Auth Key Type:** A dropdown menu set to 'p8'.
- App Bundle ID:** An empty text input field.
- Key ID:** An empty text input field.
- Team ID:** An empty text input field.
- Topic:** An empty text input field.
- Save Files:** A green button at the bottom.

### 4. Goto Push Notification → Firebase Cloud Messaging (FCM) and put the required key.



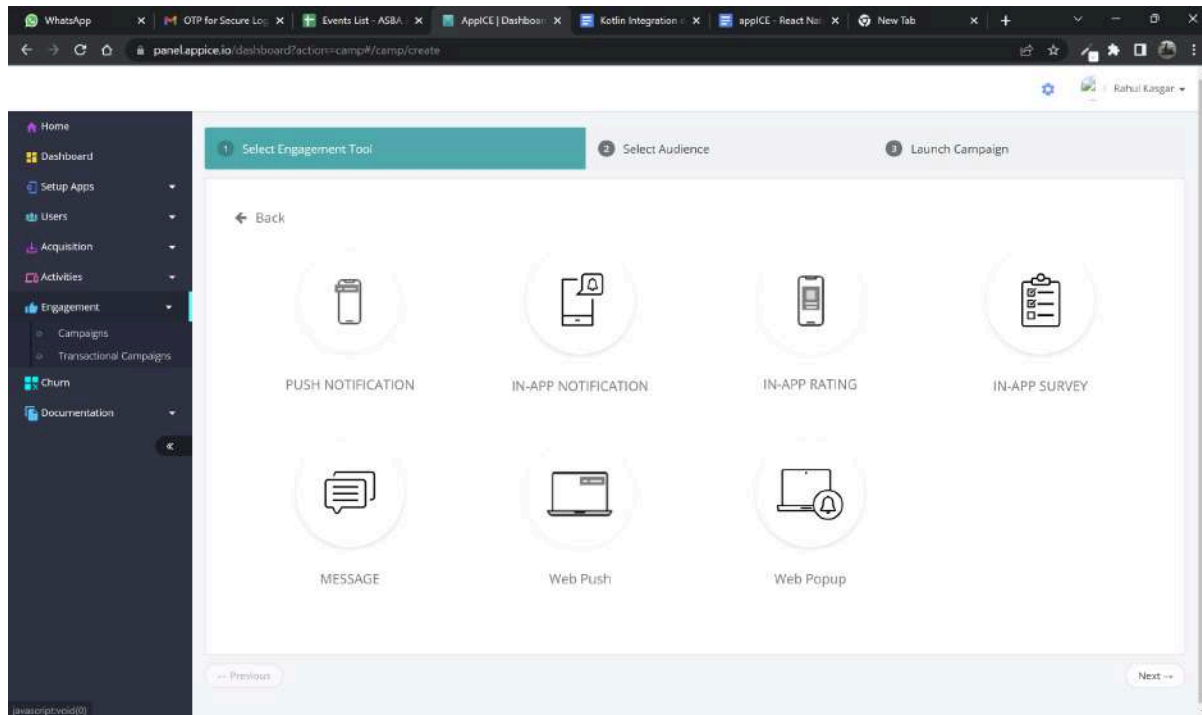
The screenshot shows the AppIC8 interface with the 'Push Notification' tab selected. The 'Firebase Cloud Messaging (FCM)' sub-tab is active, and the 'Production' environment is chosen. The form contains the following fields:

- FCM Key:** A text input field containing 'AAAAVS-UAtc:APA81bE5oOT08Phye\_qMmX0Hfji' and a 'Save' button.

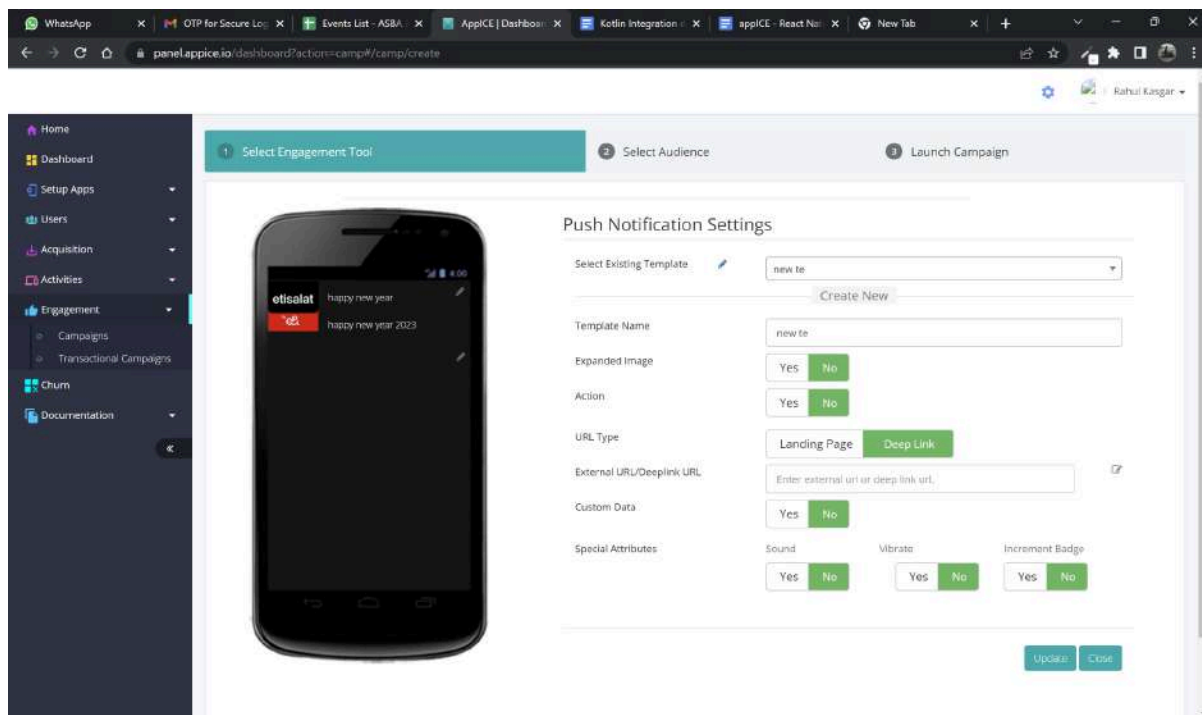
Step 3, 4# are required for push notifications to work on devices.



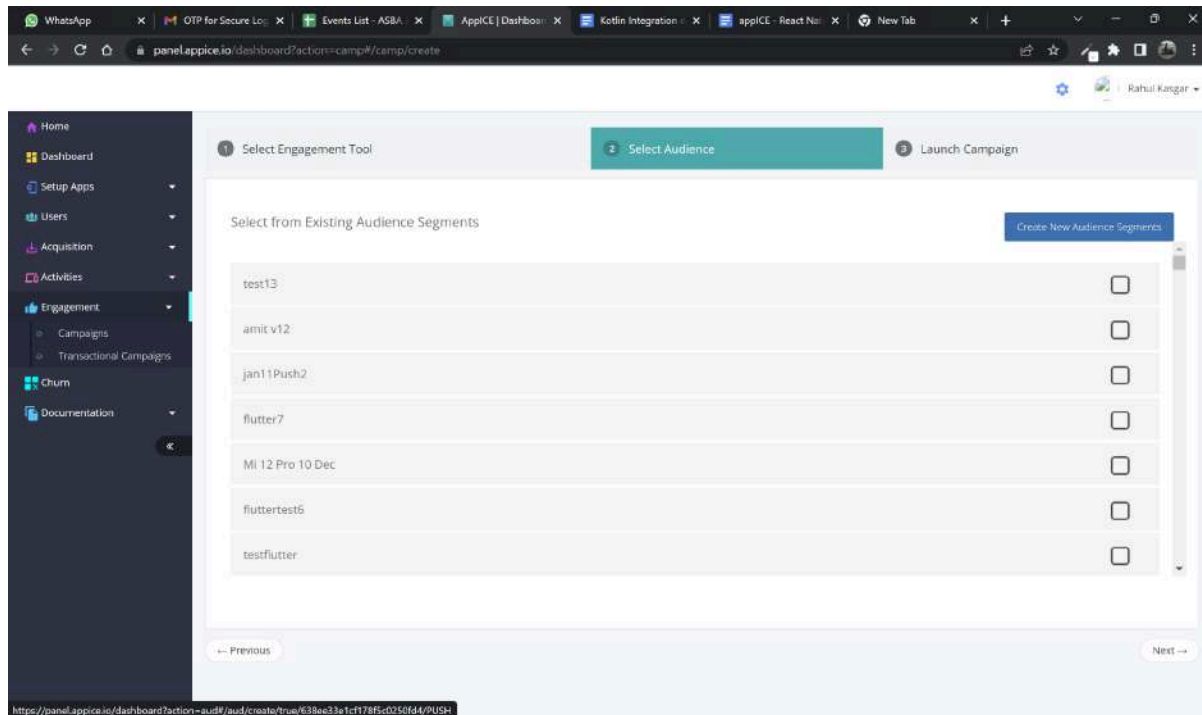
## 5. Goto Engagement > Campaigns > Create New Campaign > PUSH NOTIFICATION



Select from existing template, select details and click on Update button:



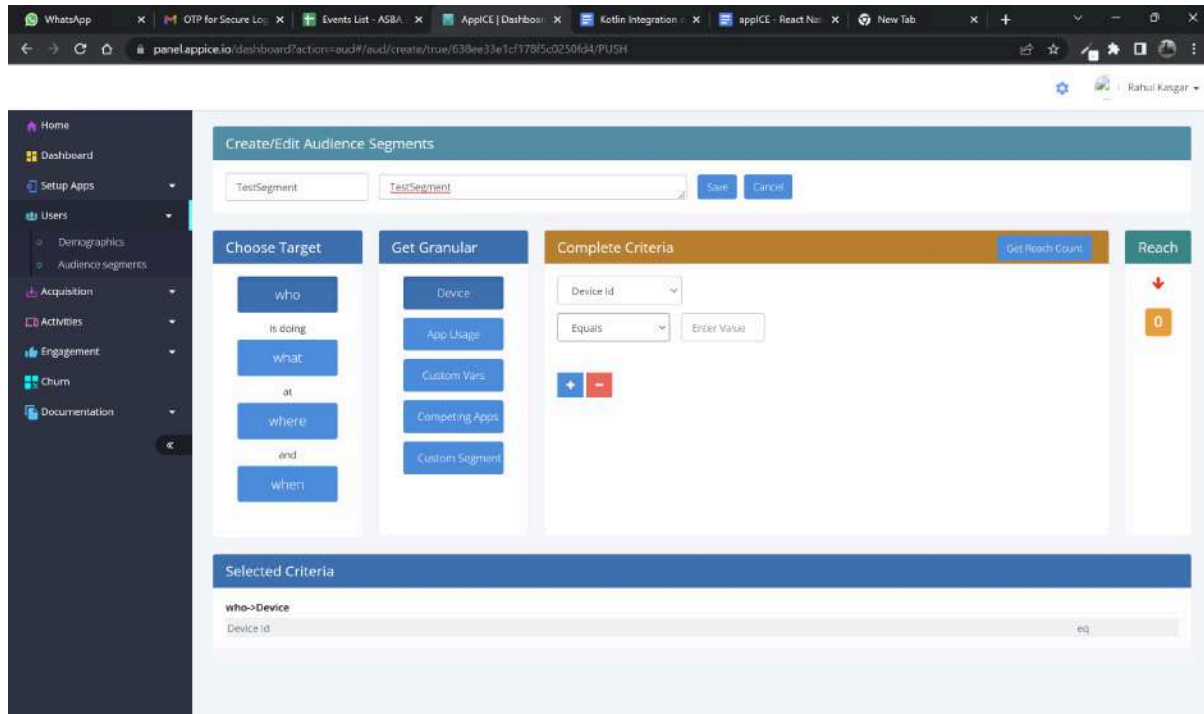
Click on Create New Audience Segment and fill details according to your need.



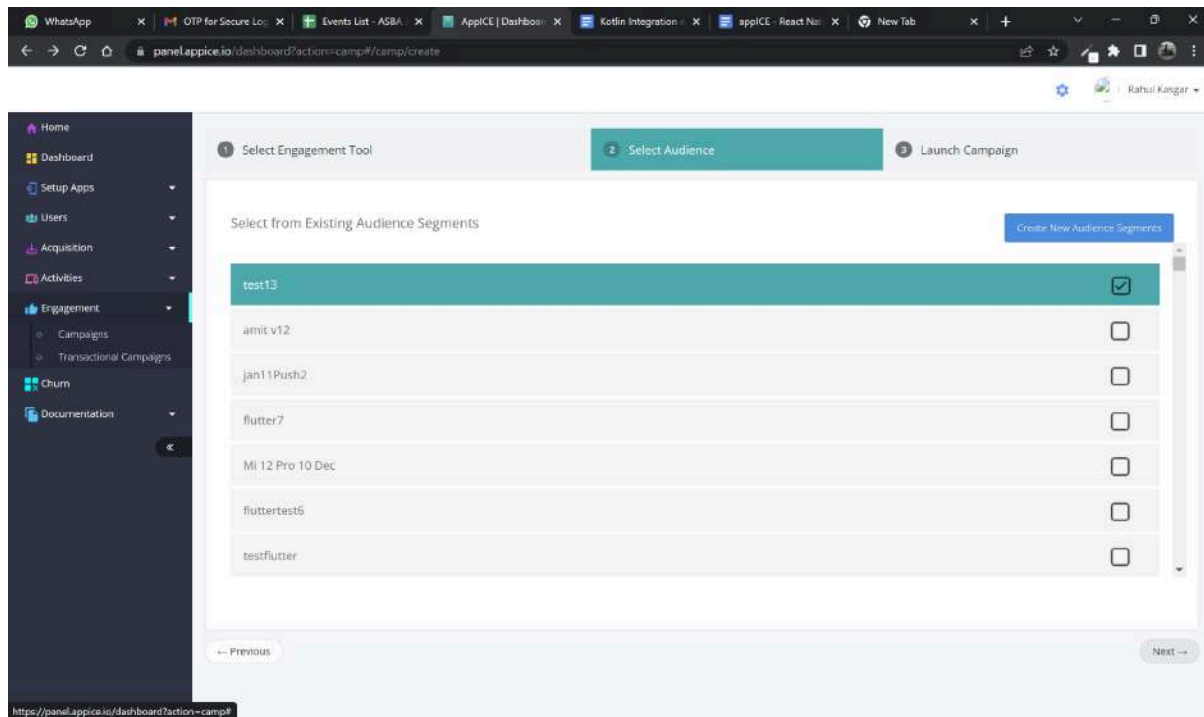
In this example we are targeting an app to specific device id for that we will need to enter our device id and click on Get Reach Count to see the reachable count for the device.

Enter segment name and description to save

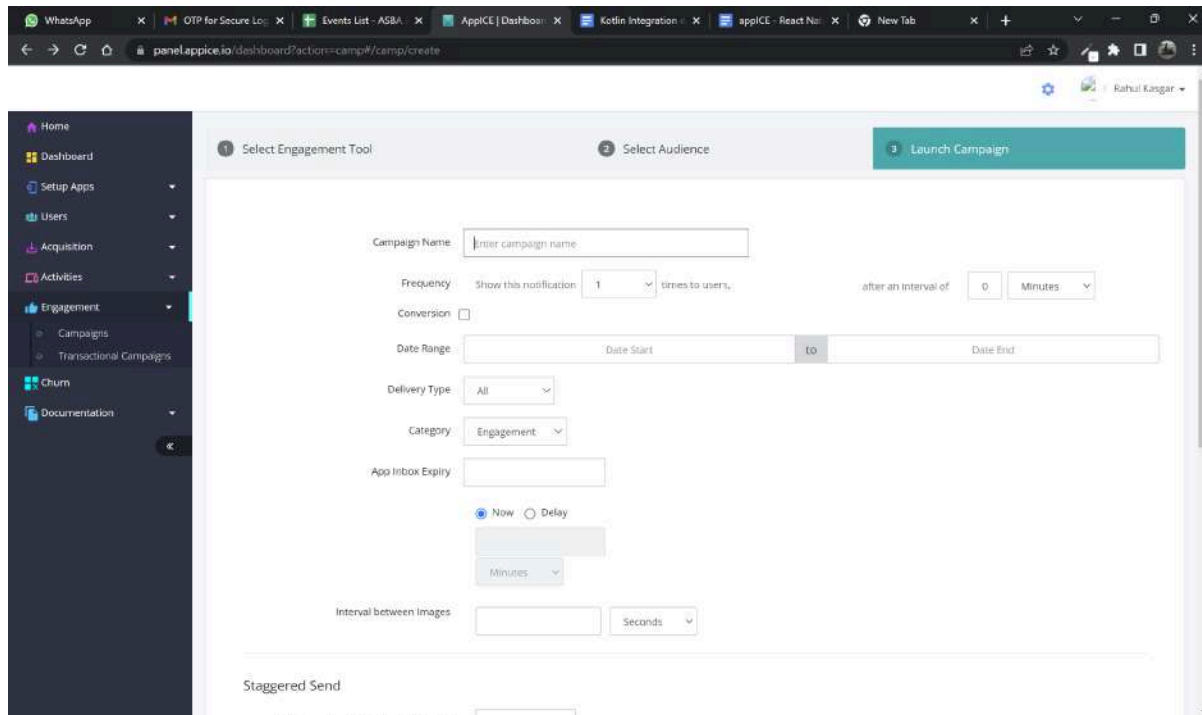
Click on Save button to go next



Select your newly created Audience segment and click next.



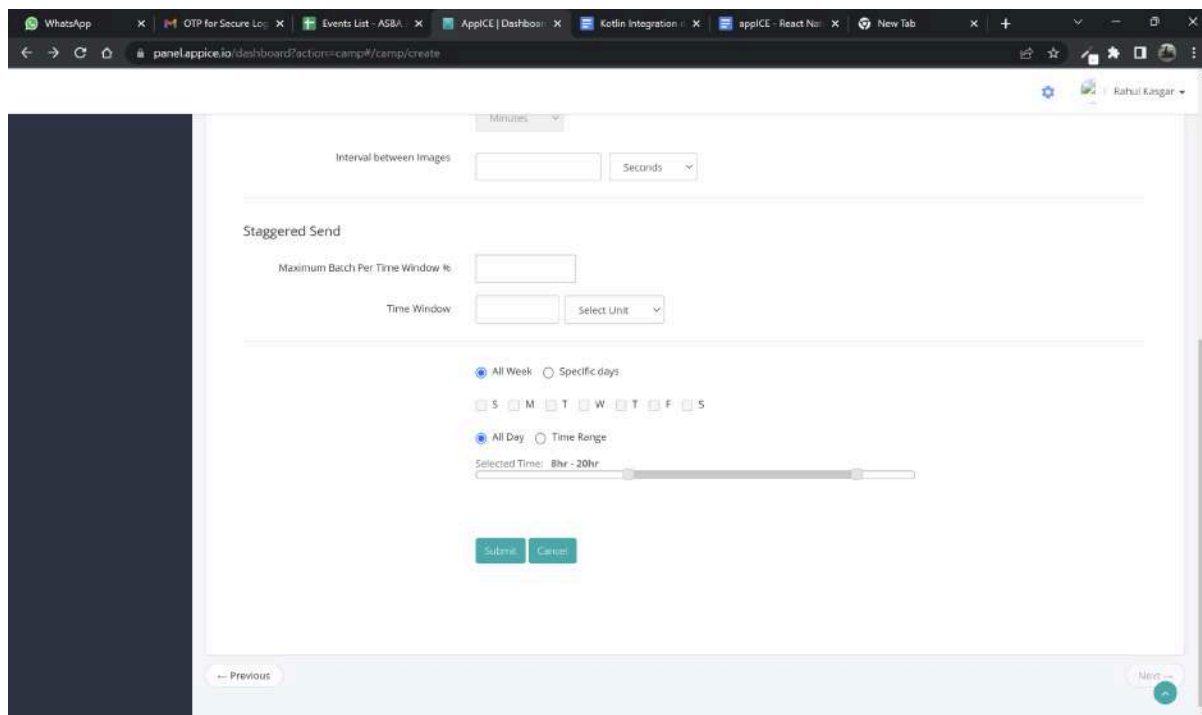
Now Enter your campaign name, select dates and campaign details according to your need.



The screenshot shows the 'Create Campaign' form in the AppICE dashboard. The form is divided into three steps: 1. Select Engagement Tool, 2. Select Audience, and 3. Launch Campaign. The 'Launch Campaign' button is highlighted in green. The form includes the following fields and options:

- Campaign Name:** A text input field with a placeholder 'Enter campaign name'.
- Frequency:** A dropdown menu set to '1' times to users, with an option to 'after an interval of' 0 minutes.
- Conversion:** A checkbox.
- Date Range:** A date range selector with 'Date Start' and 'Date End' fields.
- Delivery Type:** A dropdown menu set to 'All'.
- Category:** A dropdown menu set to 'Engagement'.
- App Inbox Expiry:** A text input field.
- Now/Delay:** Radio buttons for 'Now' (selected) and 'Delay'.
- Interval between Images:** A text input field and a dropdown menu set to 'Seconds'.
- Staggered Send:** A section with a 'Maximum Batch Per Time Window' field and a 'Time Window' dropdown menu.

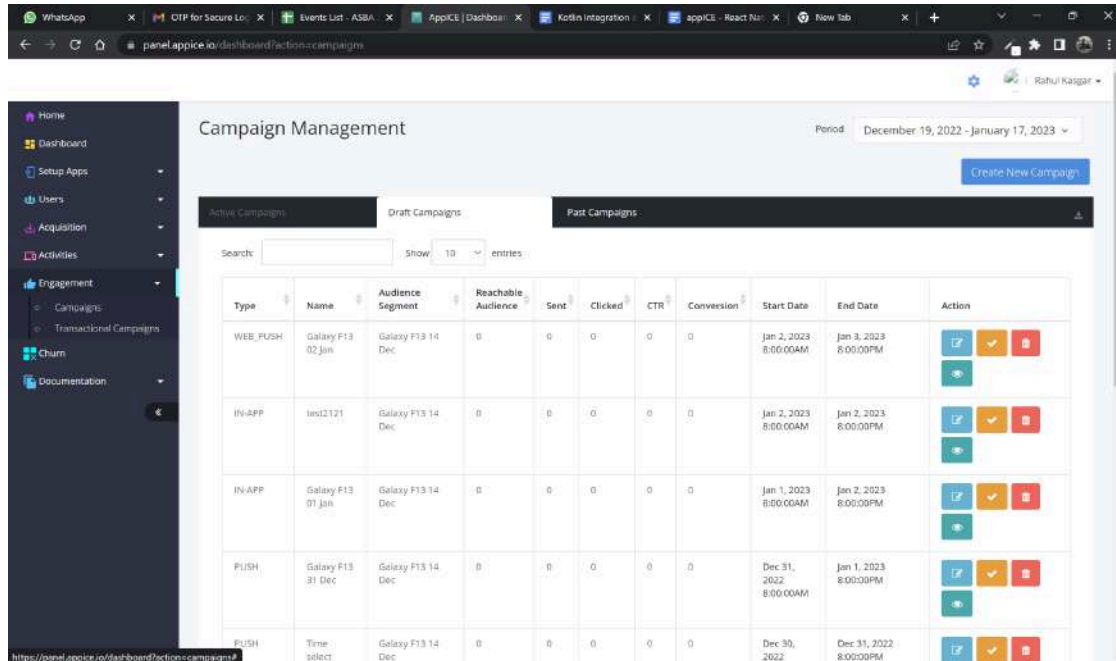
Click on Submit button to create campaign.



The screenshot shows the 'Create Campaign' form in the AppICE dashboard, with the 'Submit' button highlighted in green. The form includes the following fields and options:

- Campaign Name:** A text input field with a placeholder 'Enter campaign name'.
- Frequency:** A dropdown menu set to '1' times to users, with an option to 'after an interval of' 0 minutes.
- Conversion:** A checkbox.
- Date Range:** A date range selector with 'Date Start' and 'Date End' fields.
- Delivery Type:** A dropdown menu set to 'All'.
- Category:** A dropdown menu set to 'Engagement'.
- App Inbox Expiry:** A text input field.
- Now/Delay:** Radio buttons for 'Now' (selected) and 'Delay'.
- Interval between Images:** A text input field and a dropdown menu set to 'Seconds'.
- Staggered Send:** A section with a 'Maximum Batch Per Time Window' field and a 'Time Window' dropdown menu.
- Submit/Cancel:** Two buttons at the bottom of the form.

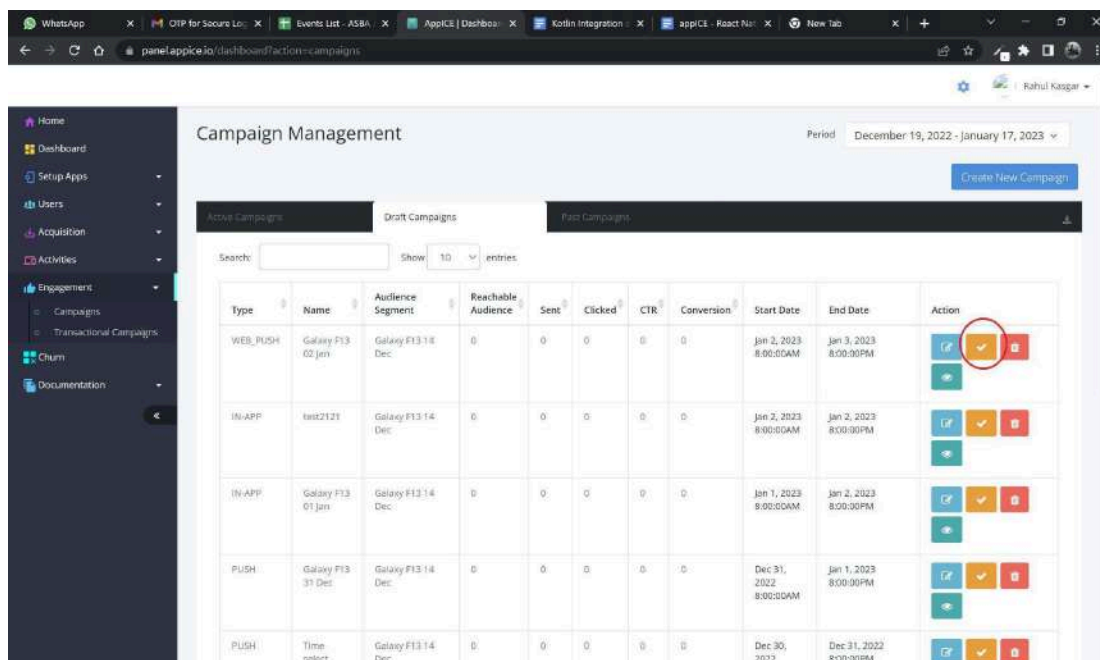
Now our campaign is successfully created but not yet activated we will activate our campaign. By default your campaign will be there in the draft campaigns section.



The screenshot shows the 'Campaign Management' dashboard with a sidebar on the left containing navigation links: Home, Dashboard, Setup Apps, Users, Acquisition, Activities, Engagement (with sub-links for Campaigns, Transactional Campaigns, and Churn), and Documentation. The main content area has tabs for 'Active Campaigns', 'Draft Campaigns', and 'Past Campaigns'. The 'Draft Campaigns' tab is selected, showing a table of campaigns. The table has columns: Type, Name, Audience Segment, Reachable Audience, Sent, Clicked, CTR, Conversion, Start Date, End Date, and Action. There are five campaigns listed, all in the 'Draft' state. The first campaign is 'WEB\_PUSH' for 'Galaxy F13 02 Jan' with an audience segment of 'Galaxy F13 14 Dec'. The 'Action' column for each campaign contains icons for 'Set Active', 'Set Draft', and 'Delete'.

Type	Name	Audience Segment	Reachable Audience	Sent	Clicked	CTR	Conversion	Start Date	End Date	Action
WEB_PUSH	Galaxy F13 02 Jan	Galaxy F13 14 Dec	0	0	0	0	0	Jan 2, 2023 8:00:00AM	Jan 3, 2023 6:00:00PM	[Set Active] [Set Draft] [Delete]
IN-APP	Inst2121	Galaxy F13 14 Dec	0	0	0	0	0	Jan 2, 2023 8:00:00AM	Jan 2, 2023 8:00:00PM	[Set Active] [Set Draft] [Delete]
IN-APP	Galaxy F13 01 Jan	Galaxy F13 14 Dec	0	0	0	0	0	Jan 1, 2023 8:00:00AM	Jan 2, 2023 8:00:00PM	[Set Active] [Set Draft] [Delete]
PUSH	Galaxy F13 31 Dec	Galaxy F13 14 Dec	0	0	0	0	0	Dec 31, 2022 8:00:00AM	Jan 1, 2023 8:00:00PM	[Set Active] [Set Draft] [Delete]
PUSH	Time select	Galaxy F13 14 Dec	0	0	0	0	0	Dec 30, 2022	Dec 31, 2022 8:00:00PM	[Set Active] [Set Draft] [Delete]

You can Activate your campaign by clicking on Set Active Button.



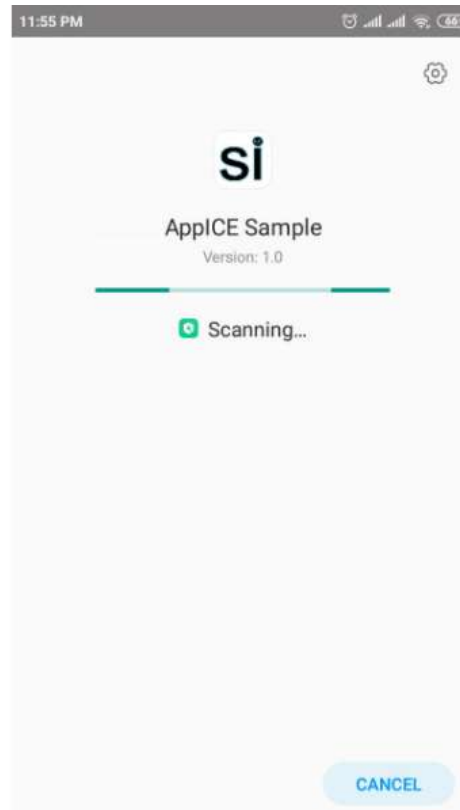
This screenshot is identical to the previous one, but with a red circle highlighting the 'Set Active' button (a blue icon with a white checkmark) in the 'Action' column of the first campaign row.

Once you activate your campaign, push notification should be received on your device.

## Install App on your Android phone

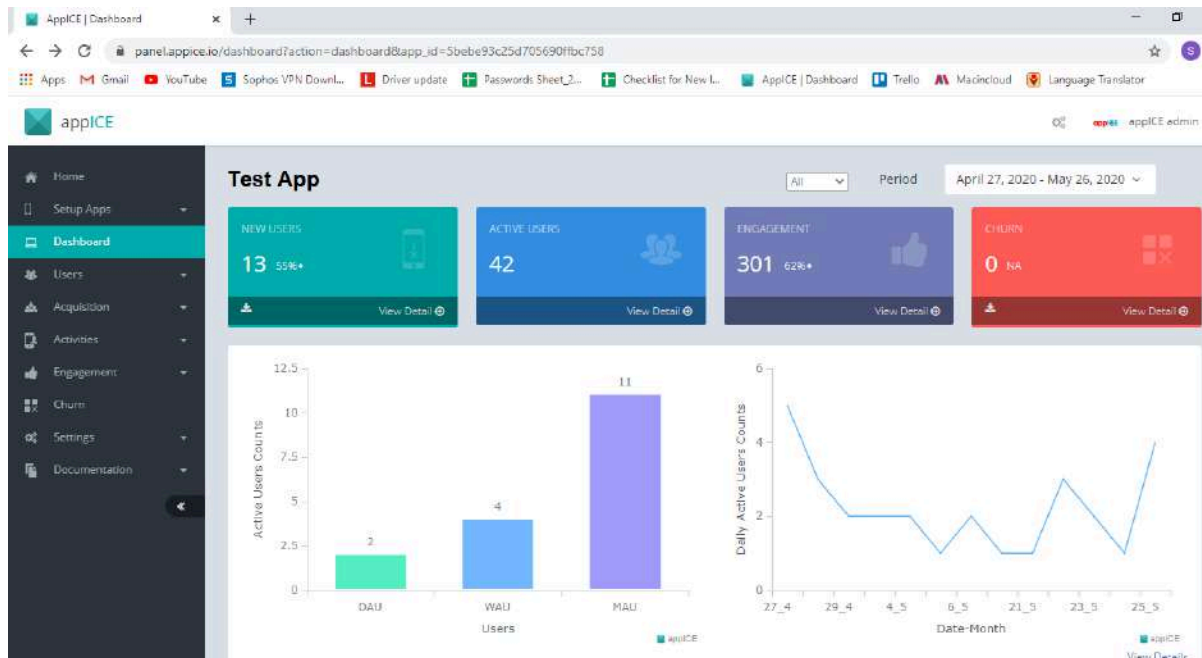
You can install the app from the link below.

*<link of sample app to be provided separately>*

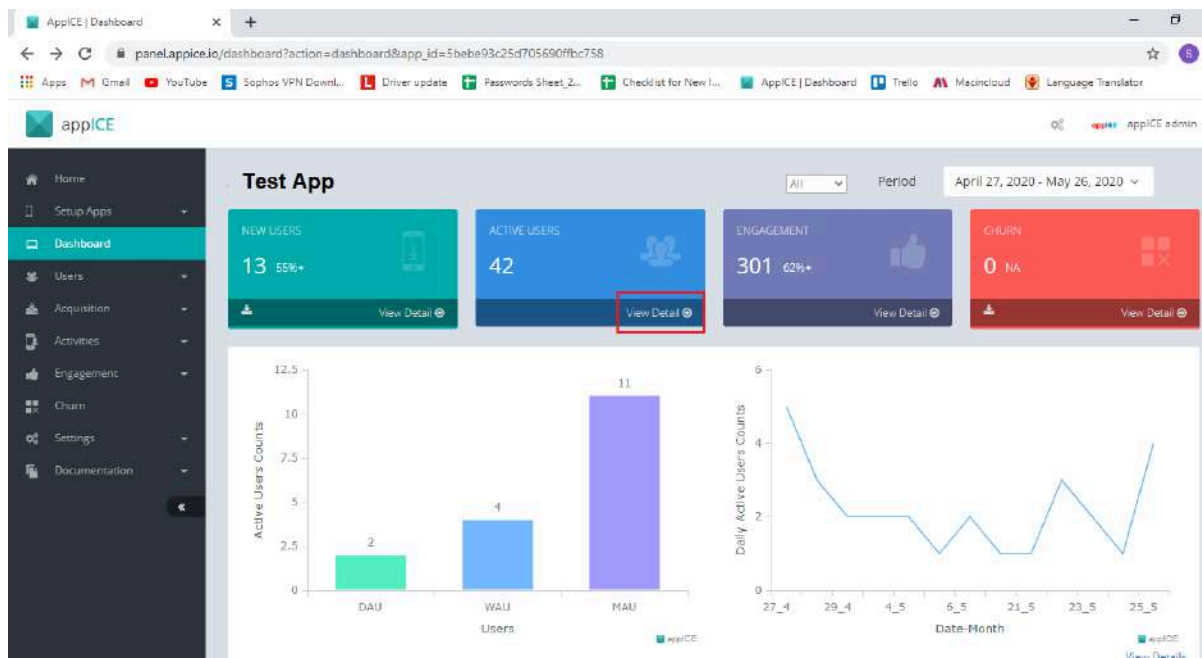


## Verify integration on appICE panel

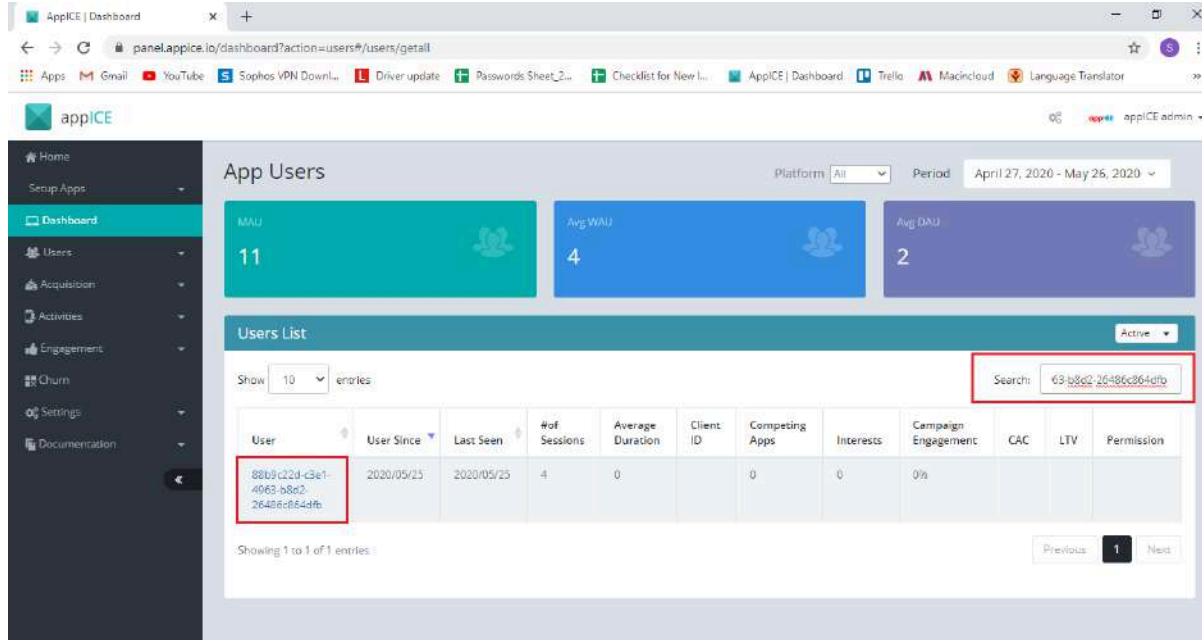
1. Login to panel.appice.io to see these values. Click on your app and go to its dashboard.



2. Click on 'Active Users' ☐ 'View Detail' to see details of your phone app.



3. Clicking on 'View Detail' takes to App Users page. Search your phone Advertising Id in Search box to search for your phone Ads Id.



The screenshot shows the AppICE Dashboard with the 'App Users' section. The search bar is highlighted with a red box, and the search results table is also highlighted with a red box.

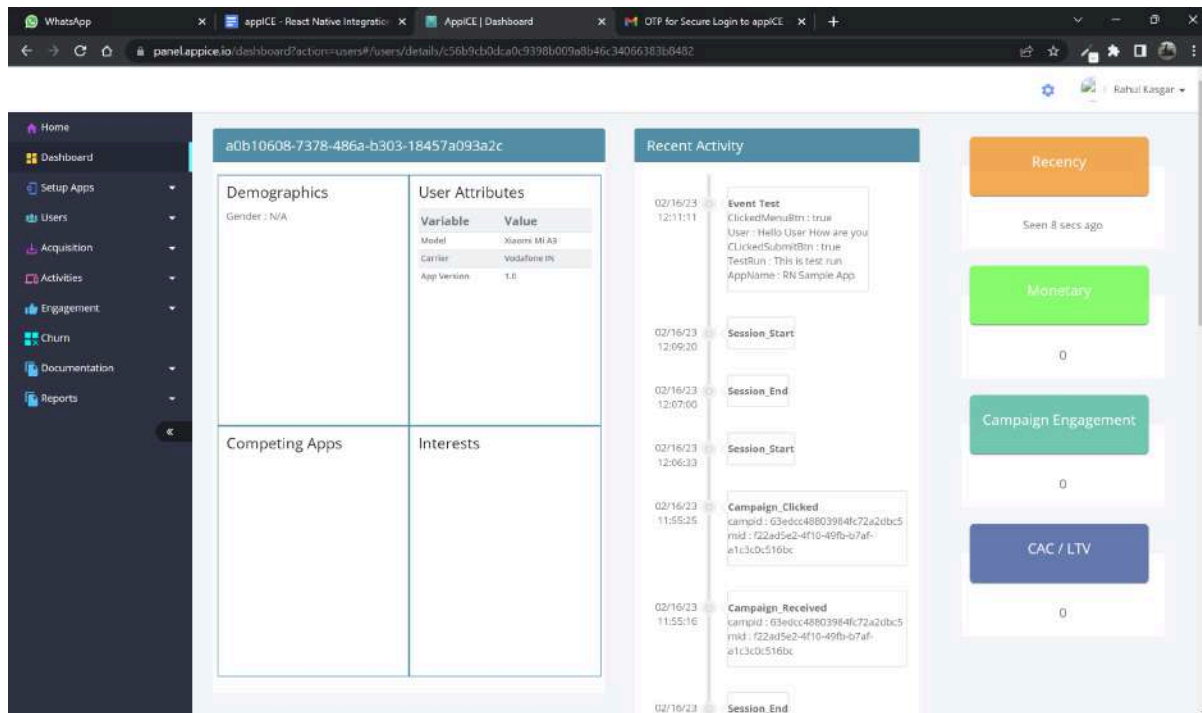
**App Users Summary:**

- MAU: 11
- Avg WAU: 4
- Avg DAU: 2

**Users List:**

User	User Since	Last Seen	# of Sessions	Average Duration	Client ID	Competing Apps	Interests	Campaign Engagement	CAC	LTV	Permission
88b9c2d4-3e1-4063-b8e2-26486c864dfb	2020/05/25	2020/05/25	4	0		0	0	0/0			

4. Click on the searched Ad Id (User) to see details:



The screenshot shows the AppICE Dashboard with the 'App Users' section. The user's details are displayed, including demographics, user attributes, recent activity, and various metrics.

**User Details:**

- Ad Id:** a0b10608-7378-486a-b303-18457a093a2c
- Demographics:** Gender: N/A
- User Attributes:**

Variable	Value
Model	Xiaomi MI A3
Carrier	Vodafone IN
App Version	1.0
- Competing Apps:**
- Interests:**
- Recent Activity:**
  - 02/16/23 12:11:11: Event Test (ClickedMenuBtn: true, User: Hello User How are you, ClickedSubMenuBtn: true, TestRun: This is test run, AppName: RN Sample App)
  - 02/16/23 12:09:20: Session\_Start
  - 02/16/23 12:07:00: Session\_End
  - 02/16/23 12:06:33: Session\_Start
  - 02/16/23 11:55:25: Campaign\_Clicked (campaign: 63e0cc48803964fc72a20bc5, mid: f22a05e2-4f10-49fb-b7af-a1c3cb516bc)
  - 02/16/23 11:55:16: Campaign\_Received (campaign: 63e0cc48803964fc72a20bc5, mid: f22a05e2-4f10-49fb-b7af-a1c3cb516bc)
  - 02/16/23: Session\_End
- Metrics:**
  - Recency: Seen 8 secs ago
  - Monetary: 0
  - Campaign Engagement: 0
  - CAC / LTV: 0