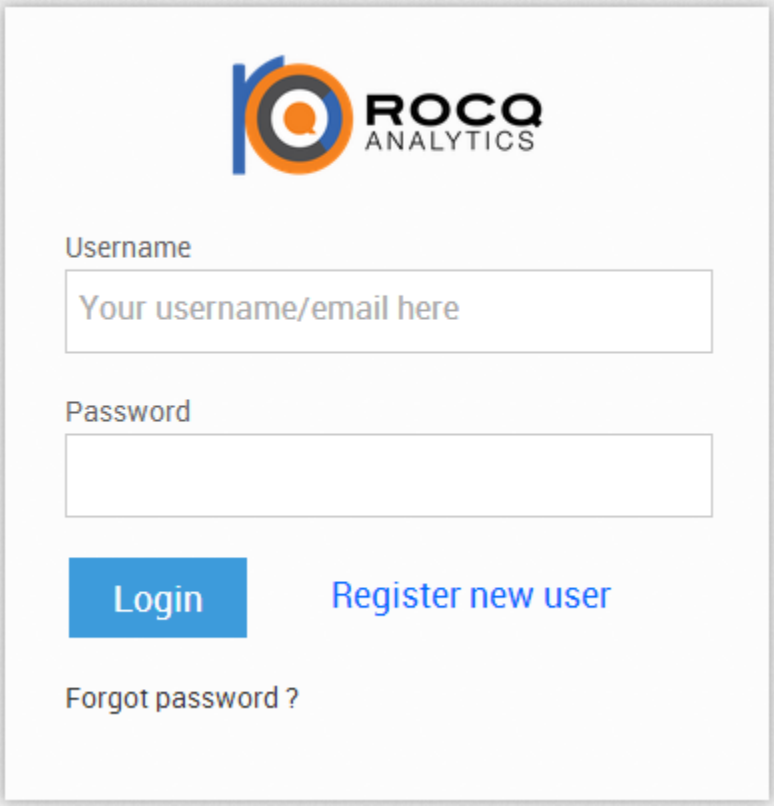
rocQ ANALYTICS

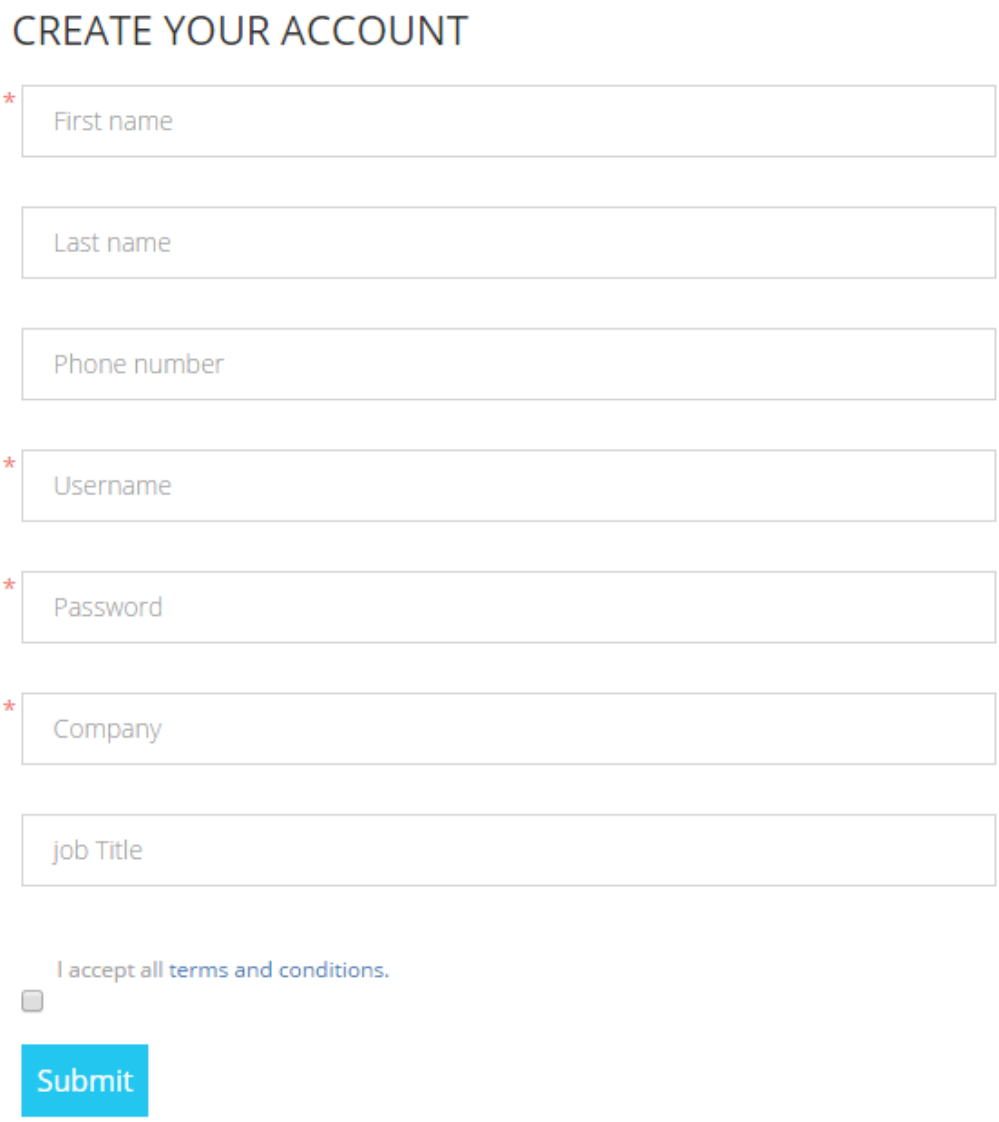
iOS API INREGRATION

Setting rocQ SDK in iOS Project

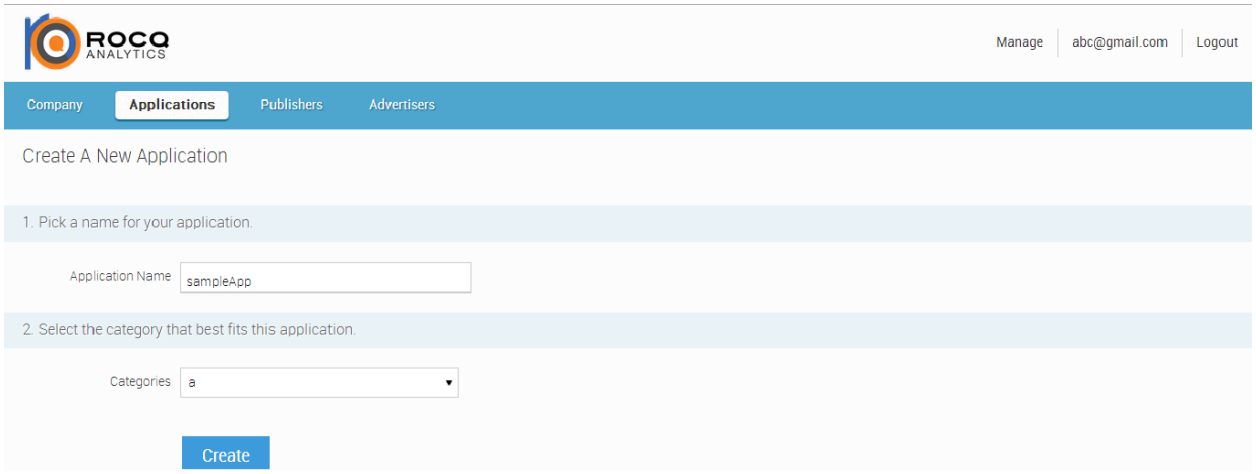
- Go to http://rocq.io/login
- Click on **Register Now** to get yourself registered.



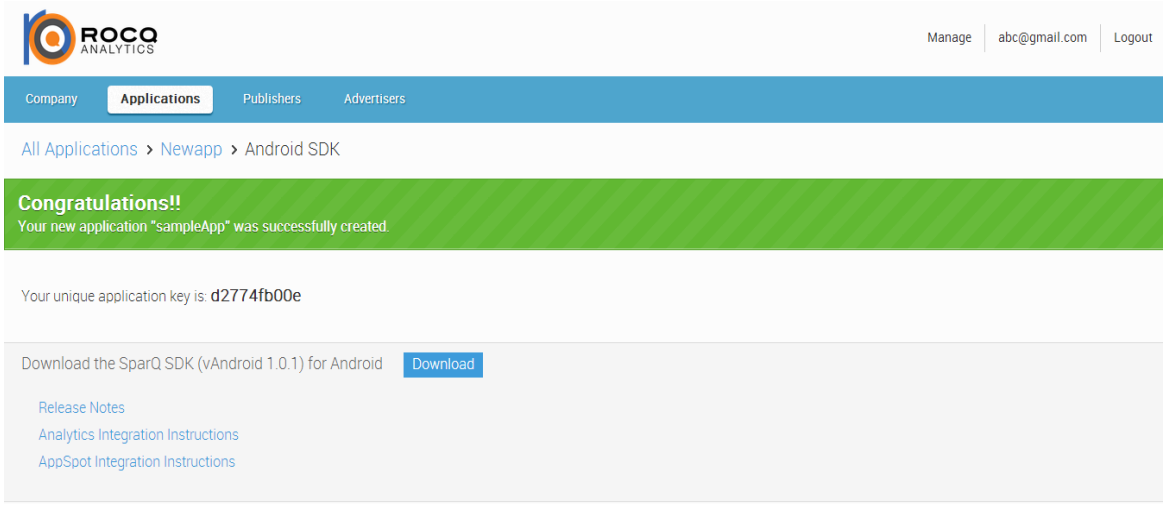
- Fill in all the required credentials and **Submit** to create your account.



- Once you have successfully registered, you will be directed to your first page. Now, click on **Create A New Application**.



- You will be assigned with a Unique Application Key of your app. This key helps rocQ identify your app. Now, download the rocQ SDK from the **Download** button provided at the bottom of the page.



- Now you have to integrate the rocQ iOS SDK in your Application.

rocQ iOS SDK integration steps

1) Importing headers
2) Linking static and dynamic libraries

Step 1: Importing Headers

i) Add **rocqAnalytics.h** file to your project.

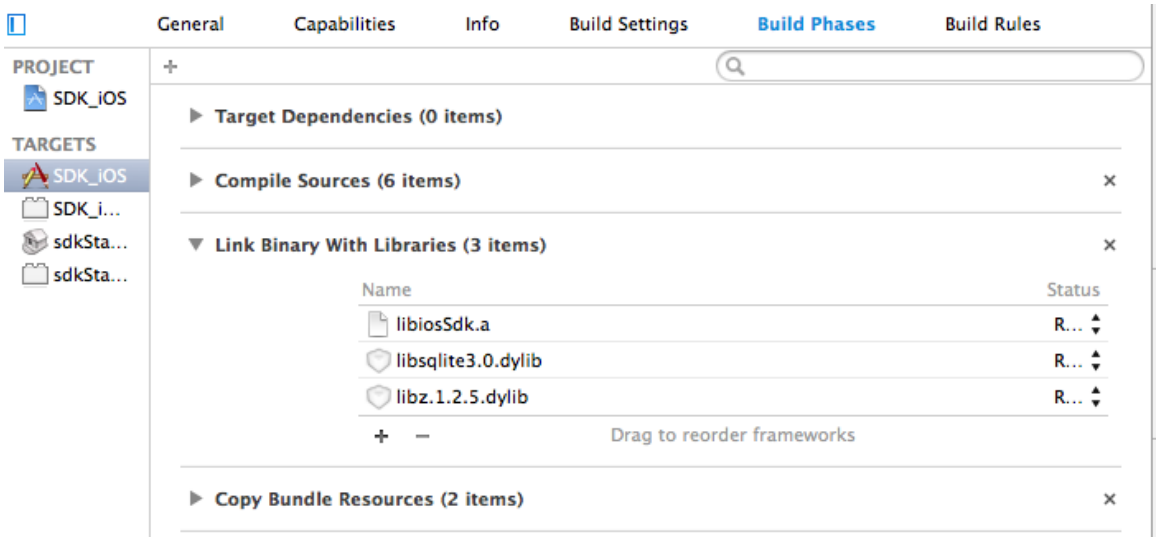
Step 2: Linking static and dynamic libraries

Go to the **Build Phases** tab, followed by **Link Binary With Libraries** and add the following files

1) Static library file ROCQiosSdk.a that is in **Downloads** folder.

2) Dynamic library libsqlite3.0.dylib

3) Dynamic library libz.1.2.5.dylib



Using ROCQ iOS SDK for tracking:

Step 1: Import **rocqAnalytics.h** in **AppDelegate.h** file

```
#import “rocqAnalytics.h”
```

Step 2: Configure app secret that you got by registering your application in rocQ website. Add the below given code in **Appdelegate.m** file in **didFinishLaunchingWithOptions** method.

```
[rocqAnalytics setAppsecret:@”app secret”];
```

In the above code you have to replace “app secret” string with your Unique Application Key you have been provided with by registering your application at the rocQ website.

Step 3: For Tracking Sessions

For tracking sessions and screens put the following code in every **ViewController** (screen) in **ViewWillAppear** method.

```
[rocqAnalytics trackScreen:NSStringFromClass([self class])];
```

You can track the screen by their names and parameters, whenever required.

To track the screen, pass Screen name and properties in key-value pairs.

```
[rocqAnalytics trackScreen:“screen name” properties:
@{@"screen":@"screen 1",@"content":@"ht media"}];
```

Parameters:

- **String <Screen Name>:** It is screen name in human-readable format such as Show Details" or "Store page".
- **properties <key>,<value> (optional)**: The key-value pair consists of properties of screens, for example <items>,<10> or <movie>,<krish>. There can be N numbers of parameters.

Step 4: For tracking Event

You can track the events by their names and parameters, whenever required. To track the event, pass Event name, and properties in key-value pairs.

```
[rocqAnalytics trackEvent:@"event Name"
properties:@{@"key1":@"value1",@"key2":@"value2"} position:
CENTRE];
```

Parameters:

- **String <Event Name> :** It is event name in human-readable format such as "Buy Button" or "news list item".
- **properties (optional)** : A dictionary which contains properties of the event. There can be N numbers of parameters in it.
- **Position (optional):** You can pass the position of an item where it is placed on the screen. It will be helpful in finding its relevant position. (TOP, LEFT, RIGHT, BOTTOM, CENTER).

Step 5:- Tracking User Identity/Information.

You can track the users with their username, email, age and other parameters, whenever required. To track the user, pass user name, and properties in key-value pairs.

```
[rocqAnalytics identity:@"anonym"
properties:@{@"key":@"value",@"key2":@"value2"}];
```

Parameters:

- **String <User-Name>:** It is a user name in human-readable format like "abhi23".
- **properties (optional)** : It carries the properties of the user in key-value pairs. Example @{@"email ID":@"*******@gmail.com",@"city":@"gurgaon",@"age":@"23"} It can be N numbers of parameters.

Push notification Quick start Guide:

Tracking Push notifications:

Follow the below given steps to receive push notifications;

Step 1:

Please mail your push certificate and password for opening key file to **shashank.agarwal@hindustantimes.com**

Step 2:

Insert the following line of code in **didFinishLaunchingWithOption** method of **AppDelegate.m** file to register for push notifications.

```
[[UIApplication sharedApplication]registerForRemoteNotificationTypes:
(UIRemoteNotificationTypeBadge | UIRemoteNotificationTypeSound |
UIRemoteNotificationTypeAlert)];
```

Step 3:

Insert the following method in **AppDelegate.m** file to set device token

```
-(void)application:(UIApplication*)application
didRegisterForRemoteNotificationsWithDeviceToken:(NSData*)deviceToke
n{

[rocqAnalytics setDeviceToken:deviceToken];

}
```

Step 4:

Insert the following method in **AppDelegate.m** file to track push notifications.

```
- (void)application:(UIApplication*)application
didReceiveRemoteNotification:(NSDictionary*)userInfo{

[rocqAnalytics trackPush:userInfo];

}
```

Deep Linking:

Insert the following method in **AppDelegate.m** file to track deep linking.

```
-(BOOL)application:(UIApplication *)application openURL:(NSURL
*)url sourceApplication:(NSString *)sourceApplication
annotation:(id)annotation{

[rocqAnalytics deepLink:[url host]];

NSDictionary *deepLinkData=[[NSDictionary alloc]init];

deepLinkData=[rocqAnalytics parseQueryString:[url query]];

return YES;

}
```

Setting Debug Mode:

To set the debug mode, insert the below given code in **Appdelegate.m** file in **didFinishLaunchingWithOptions** method:

```
[rocqAnalytics setRQDebugMode:YES];
```