

Embedding Android SDK to build a business chat app

Objective:

To embed smart messaging and build an Android app for business communication between employees

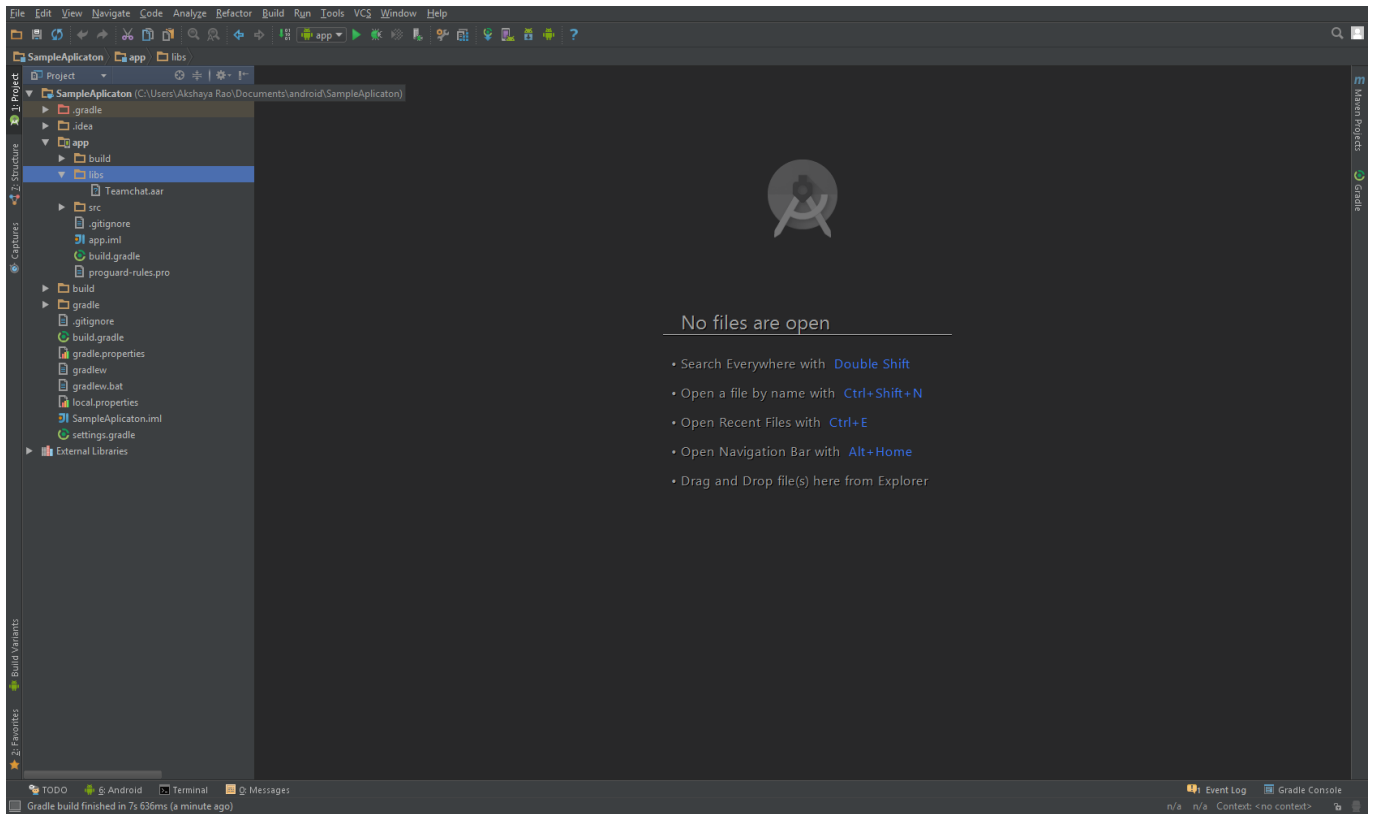
Implementation:

Method 1:

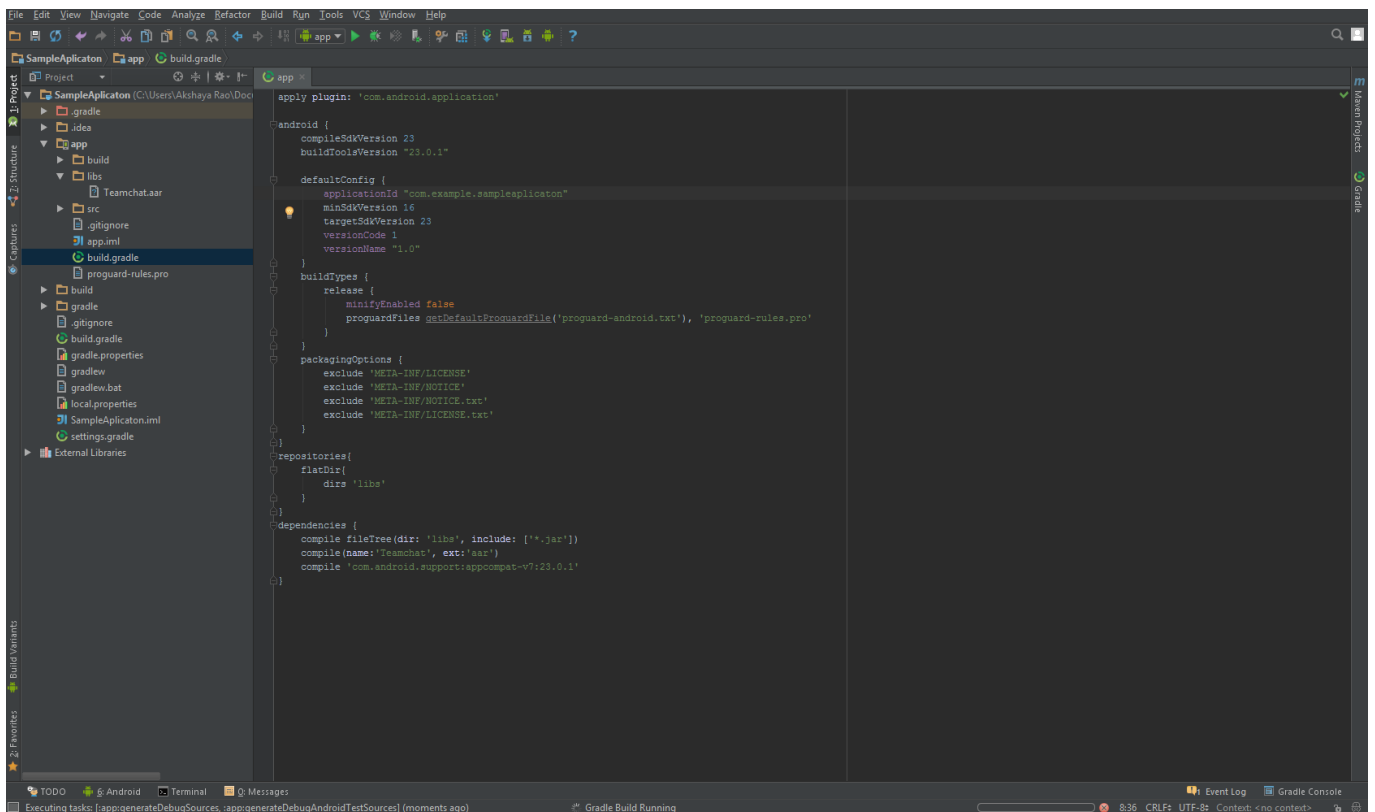
Place Teamcht.aar file in project's lib folder and add the following lines to app's gradle file.:

```
android {
    .
    .
    packagingOptions {
        exclude 'META-INF/LICENSE'
        exclude 'META-INF/NOTICE'
        exclude 'META-INF/NOTICE.txt'
        exclude 'META-INF/LICENSE.txt'
    }
    .
    .
}
repositories{
    flatDir{
        dirs 'libs'
    }
}

dependencies {
    .
    .
    compile(name:'Teamchat', ext:'aar')
    .
    .
}
```



Your Gradle file will look something like this



Method 2:

Follow the below mentioned steps:

Step 1 : Right click on project in project structure.

Step 2 : Click on New->Module.

Step 3 : Click on Import .Jar or .Aar package.

Step 4 : Click next.

Step 5 : Now browse and select the Teamchat.aar file.

Step 6 : Click Finish. Now the module will be created.

Step 7: Click on project structure and go in dependency section of the module of your app.

Step 8 : Click on plus sign on right side and add teamchat module in the dependencies.

Note:

1. This SDK uses the following libraries.

```
com.android.support:design:23.0.1
com.android.support:appcompat-v7:23.0.1
com.google.android.gms:play-services-maps:8.1.0
com.google.android.gms:play-services-gcm:8.1.0
aws-android-sdk-2.1.6-autoscaling.jar
aws-android-sdk-2.1.6-core.jar
universal-image-loader-1.9.3.jar
```

Please add the following under dependencies of gradle file

```
dependencies {
    .
    .
    compile 'com.android.support:appcompat-v7:23.0.1'
    compile 'com.google.android.gms:play-services-gcm:8.1.0'
    compile 'com.google.android.gms:play-services-maps:8.1.0'
    compile 'com.android.support:design:23.0.1'
    .
    .
}
```

2. Make sure to use a theme with no actionbar as your app base theme. Customize this in styles.xml.

Example: Using Light theme with no actionbar.

```
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
    <!-- Customize your theme here. -->
    </style>
</resources>
```

3. To use location services, set up the app to use google maps.

Refer <https://developers.google.com/maps/documentation/android-api/start>

4. To use GCM Push Notification services, set up the app to use GCM.

Refer <https://developers.google.com/cloud-messaging/android/client>

5. SDK uses the following permissions. So no need of including these permissions again in your application Manifest file.

Network/ internet related

```
android.permission.ACCESS_NETWORK_STATE
android.permission.ACCESS_WIFI_STATE
android.permission.INTERNET
android.permission.INTERACT_ACROSS_USERS_FULL
```

Contacts read and write

```
android.permission.READ_CONTACTS
android.permission.WRITE_CONTACTS
```

File storage related

```
android.permission.WRITE_EXTERNAL_STORAGE
android.permission.READ_EXTERNAL_STORAGE
```

Call/ SMS(for verification) related

```
android.permission.CALL_PHONE
android.permission.READ_PHONE_STATE
android.permission.RECEIVE_SMS
```

Push notification related

```
android.permission.WAKE_LOCK
com.google.android.c2dm.permission.RECEIVE
android.permission.VIBRATE
android.permission.GET_TASKS
```

Location related

```
android.permission.ACCESS_FINE_LOCATION
android.permission.ACCESS_COARSE_LOCATION
com.google.android.providers.gsf.permission.READ_GSERVICES
```

Mic

```
android.permission.RECORD_AUDIO
```

Camera access

```
android.permission.CAMERA
```

Set host URL before initializing using the following API. Make sure the URL ends with '/' (ex. <http://demo.teamchat.com/>)

```
TeamChat.setHostURL("HOST_URL", this);
```

Now you need to initialize the Teamchat object with appID before making any other API calls.

```
Teamchat.initializeWithAppID("your app ID here", "your api key here", this, new
Teamchat.TeamchatCompletionHandler()
{
    @Override
    public void onTeamchatCompletion(boolean success, String error, String message)
    {
        if(success)
        {
            //Initialized successfully
        }
        else
        {
            //Initialization failed. error is holding the error description.
        }
    }, this
);
```

Enable remote notifications by setting GCM ID.:

```
GoogleCloudMessaging gcm = GoogleCloudMessaging.getInstance(MainActivity.this);
regId = gcm.register("your_GCM_SenderId");
Teamchat.setRemoteNotificationsDeviceToken(regId, MainActivity.this);
```

Refer to the link below, how to obtain Registration ID.

<https://developers.google.com/cloud-messaging/android/client>

You can login to Teamchat in two ways:

1. Set authentication code, emailID and userID.

```
TeamChat.setAuthenticationCode("authenticationCode", this);
TeamChat.setUserEmailID("userEmailID", this);
TeamChat.setUserID("userID", this);
```

2. Launch Teamchat login activity.

```
Teamchat.login(LoginActivity.this, new Teamchat.LoginCompletionHandler()
{
    @Override
    public void onLoginCompletion(boolean success, String message)
    {
        if (success)
        {
            //Success
        }
        else
        {
            //Failure while logging into Teamchat.
        }
    }
});
```

If you want to configure UI of the Teamchat screens, you can set them as follows:

```
Teamchat.setNavigationTitle(this, "My Title");
Teamchat.setChatListItemSelector(R.drawable.bg, this);
Teamchat.setMediaIcon(R.mipmap.search_view_close, this);
Teamchat.setChatletIcon(R.mipmap.search_view_right_icon, this);
```

To launch the chat groups list Activity, you can use the following API:

```
Teamchat.initWithCompletionHandler(new Teamchat.TeamchatStartCompletionHandler()
{
    @Override
    public void onTeamchatStartCompletionHandler(boolean success, String error,
String messaeg)
    {
        if (success)
        {
            Teamchat.showRoomList(this);
        }
        else
        {
            //error
        }
    }
});
```

```
    }  
    }, this  
);
```

static void initWithCompletionHandler(TeamchatStartCompletionHandler teamchatStartCompletionHandler, Context context) method should be called before launching groups list activity.

To launch the chat window activity, you can use the following API:

```
Teamchat.initWithCompletionHandler(new Teamchat.TeamchatStartCompletionHandler()  
{  
    @Override  
    public void onTeamchatStartCompletionHandler(boolean success, String error,  
String messaeg)  
    {  
        if (success)  
        {  
            Teamchat.openRoom("group_id", this);  
        }  
        else  
        {  
            //error  
        }  
    }, this  
);
```

static void initWithCompletionHandler(TeamchatStartCompletionHandler teamchatStartCompletionHandler, Context context) method should be called before launching chat window activity.

To launch the user profile activity, you can use the following API:

```
Teamchat.initWithCompletionHandler(new Teamchat.TeamchatStartCompletionHandler()  
{  
    @Override  
    public void onTeamchatStartCompletionHandler(boolean success, String error,  
String messaeg)  
    {  
        if (success)  
        {  
            Teamchat.showProfile(this);  
        }  
        else  
        {  
            //error  
        }  
    }, this  
);
```

static void initWithCompletionHandler(TeamchatStartCompletionHandler teamchatStartCompletionHandler, Context context) method should be called before launching user profile activity.

To launch the public groups activity, you can use the following API:

```
Teamchat.initWithCompletionHandler(new Teamchat.TeamchatStartCompletionHandler()
{
    @Override
    public void onTeamchatStartCompletionHandler(boolean success, String error,
String messaeg)
    {
        if (success)
        {
            Teamchat.showPublicGroups(this);
        }
        else
        {
            //error
        }
    }, this
);
```

static void initWithCompletionHandler(TeamchatStartCompletionHandler teamchatStartCompletionHandler, Context context) method should be called before launching public groups activity.

To launch the Teamchat settings activity, you can use the following API:

```
Teamchat.initWithCompletionHandler(new Teamchat.TeamchatStartCompletionHandler()
{
    @Override
    public void onTeamchatStartCompletionHandler(boolean success, String error,
String messaeg)
    {
        if (success)
        {
            boolean result = Teamchat.showSettings(this);
            if(result)
            {
                //Success
            }
            else
            {
                //No active Teamchat session
            }
        }
        else
        {
            //error
        }
    }, this
```



```
);
```

static void initWithCompletionHandler(TeamchatStartCompletionHandler teamchatStartCompletionHandler, Context context) method should be called before launching Teamchat settings activity.

To enable PassLock, you can use the following API:

```
Teamchat.enablePassLock(context);
```

To disable PassLock, you can use the following API:

```
Teamchat.disablePassLock(context);
```

To reset the PassCode, you can use the following API:

```
Teamchat.resetPassCode(context);
```

To check whether PassLock is enabled or not use the following API:

```
Teamchat.isPassLockEnabled();
```

To enable SandBoxing, you can use the following API:

```
Teamchat.enableSandBoxing(context);
```

To disable SandBoxing, you can use the following API:

```
Teamchat.disableSandBoxing(context);
```

To check whether SandBoxing is enabled or not use the following API:

```
Teamchat.isSandBoxingEnabled(context);
```

To create a new Group, you can use the following API:

```
TeamchatGroupCreator creator = new TeamchatGroupCreator();  
  
//Mandatory methods for group creation  
creator.setGroupName("My Group Name");
```

```

//Group Members
ArrayList<Teamchat.TCTeamchatContact> groupMembers = new
ArrayList<Teamchat.TCTeamchatContact> ();

Teamchat.TCTeamchatContact contact1 = new Teamchat.TCTeamchatContact ();
contact1.email = "abc@gmail.com";
contact1.profileName = "abc";

Teamchat.TCTeamchatContact contact2 = new Teamchat.TCTeamchatContact ();
contact2.email = "def@gmail.com";
contact2.profileName = "def";

groupMembers.add(contact1);
groupMembers.add(contact2);

creator.setGroupMembers (groupMembers);

// Optional methods.
creator.setAdminOnly(true);
creator.setShouldHideMemberProfiles (true);

creator.createGroupWithCompletionHandler(this, new
TeamchatGroupCreator.TeamchatGroupCreationCompletionHandler()
{
    @Override
    public void onTeamchatGroupCreationComplete(boolean success, TeamchatError
error, String message, Teamchat.TeamchatGroup createdGroup)
    {
        if(success)
            {
                //success
            }
        else
            {
                //Failure
            }
    }
});

```

To add members to a Group, you can use the following API:

```

//Members to be added.
ArrayList<Teamchat.TCTeamchatContact> membersToBeAdded = new ArrayList<>();
Teamchat.TCTeamchatContact contact1 = new Teamchat.TCTeamchatContact ();
contact1.email = "uvw@gmail.com";
contact1.profileName = "uvw";

Teamchat.TCTeamchatContact contact2 = new Teamchat.TCTeamchatContact ();
contact2.email = "xyz@gmail.com";
contact2.profileName = "xyz";

membersToBeAdded.add(contact1);
membersToBeAdded.add(contact2);
TeamchatGroupCreator.addMembers(this, membersToBeAdded, group, new
TeamchatGroupCreator.TeamchatGroupOperationCompletionHandler()

```

```
{
    @Override
    public void onTeamchatGroupOperationComplete(boolean success, TeamchatError
error, String message)
    {
        if(success)
        {
            //Success
        }
        else
        {
            //Failure
        }
    }
};
```

Teamchat is a messaging app for enterprises. It enables messaging among teams of any size (yes, unlimited!), by structuring the messages to reduce clutter. This is done through "smart" messages that aggregate, organize and summarize the responses, updating themselves without adding new messages to the conversation stream. Examples include polls, surveys, sales reports, registration forms, lookups and more. It's as simple as the other messaging apps, but a lot more powerful. Get your teams productive instantly with Teamchat.

