gupshup

Embed Android SDK in to a workforce marketplace app.

Objective:

To embed smart messaging in to an existing Android app for workforce marketplace

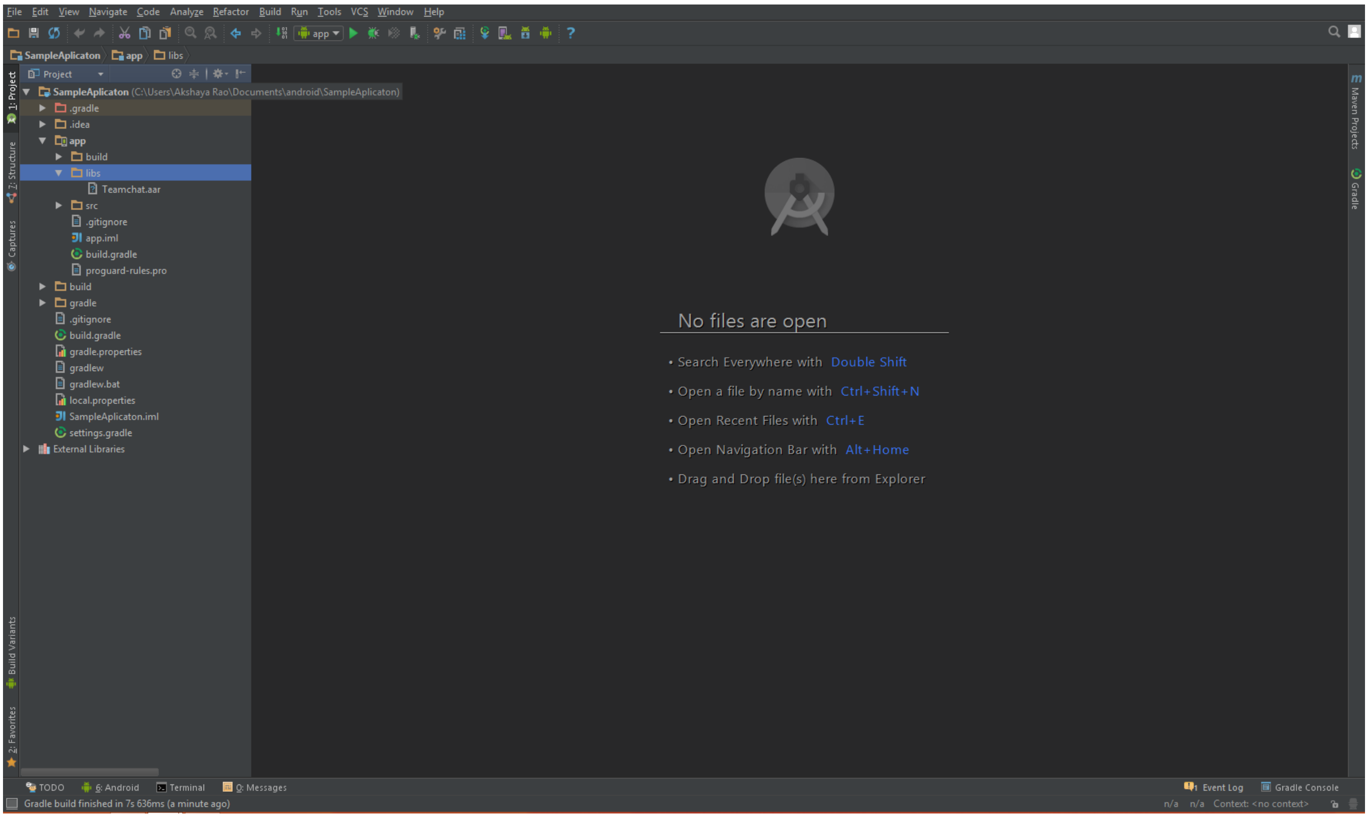
Implementation:

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')
   .
   .
}
```

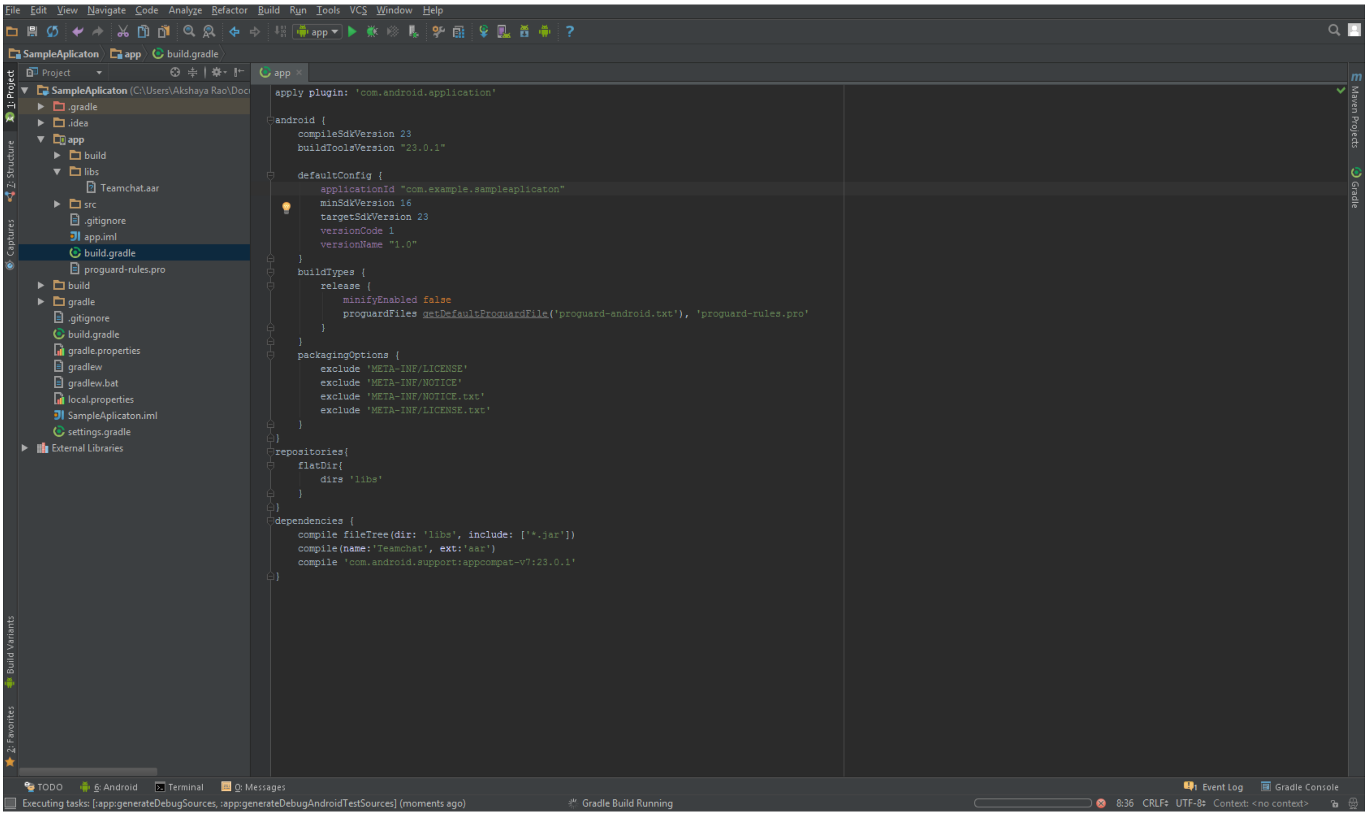


No files are open

- Search Everywhere with Double Shift
- Open a file by name with Ctrl+Shift+N
- Open Recent Files with Ctrl+E
- Open Navigation Bar with Alt+Home
- Drag and Drop file(s) here from Explorer

Gradle build finished in 7s 636ms (a minute ago)

Your Gradle file will look something like this



```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 23
    buildToolsVersion "23.0.1"

    defaultConfig {
        applicationId "com.example.sampleaplicaton"
        minSdkVersion 16
        targetSdkVersion 23
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    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 fileTree(dir: 'libs', include: ['*.jar'])
    compile(name:'Teamchat', ext:'aar')
    compile 'com.android.support:appcompat-v7:23.0.1'
}
```

Executing tasks: [:app:generateDebugSources, :app:generateDebugAndroidTestSources] (moments ago)

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
            }
    }
    });
```

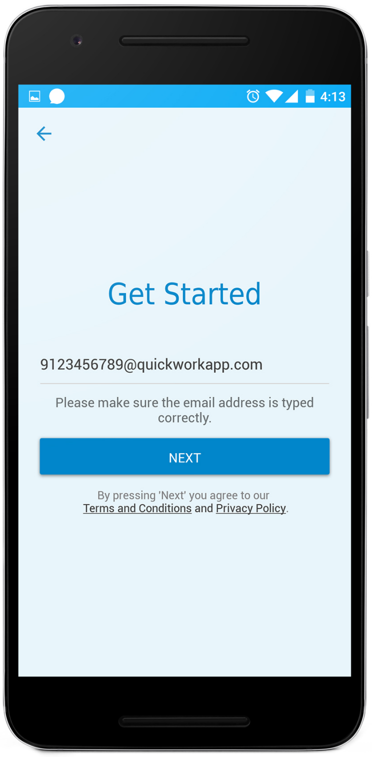
To add members to a Group, you can use the following API:

```
//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)
                {
                        //Sucsess
                }
                else
                {
                        //Failure
                }
        }
});
```

QuickWork is a 'quality workforce marketplace', where companies can get quality certified workers for short-term assignments quickly, and students, graduates and workers can get various types of short-term work opportunities (e.g. temporary, part-time, externship, internship, apprenticeship, work-from-home) from qualified companies, quickly.



4:13

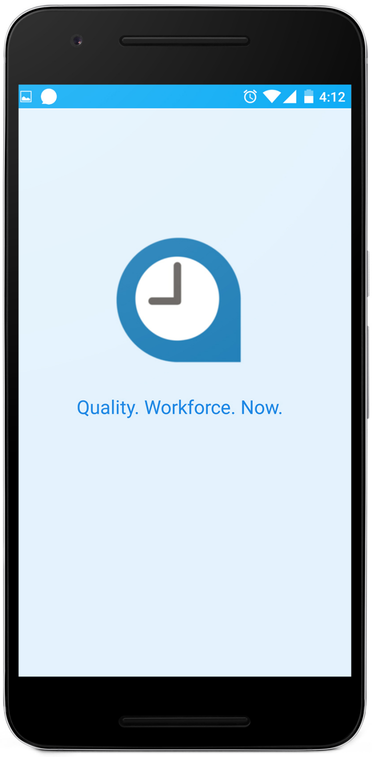
Get Started

9123456789@quickworkapp.com

Please make sure the email address is typed correctly.

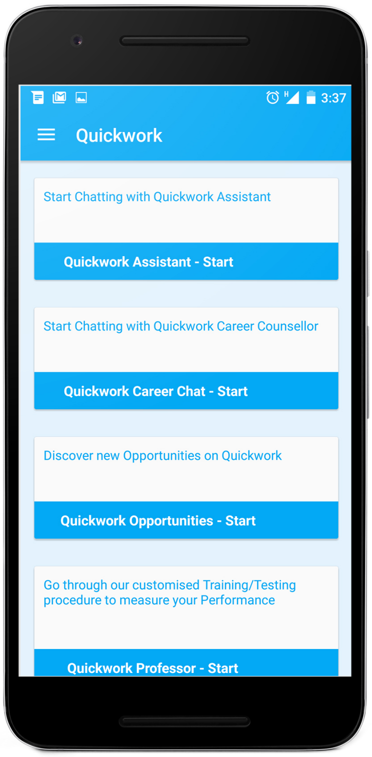
NEXT

By pressing 'Next' you agree to our Terms and Conditions and Privacy Policy.



4:12

Quality. Workforce. Now.



3:37

Quickwork

Start Chatting with Quickwork Assistant

Quickwork Assistant - Start

Start Chatting with Quickwork Career Counsellor

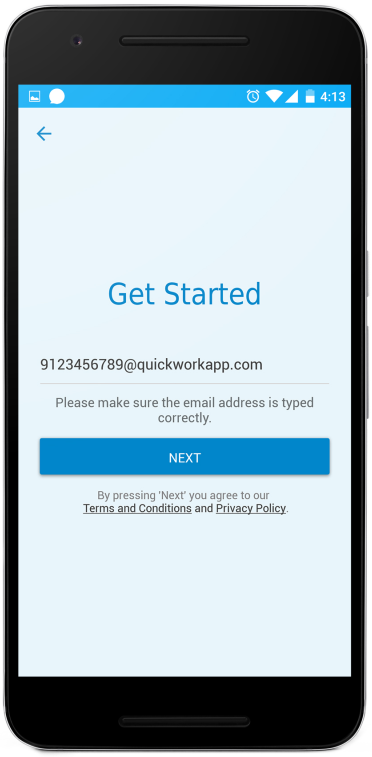
Quickwork Career Chat - Start

Discover new Opportunities on Quickwork

Quickwork Opportunities - Start

Go through our customised Training/Testing procedure to measure your Performance

Quickwork Professor - Start



4:13

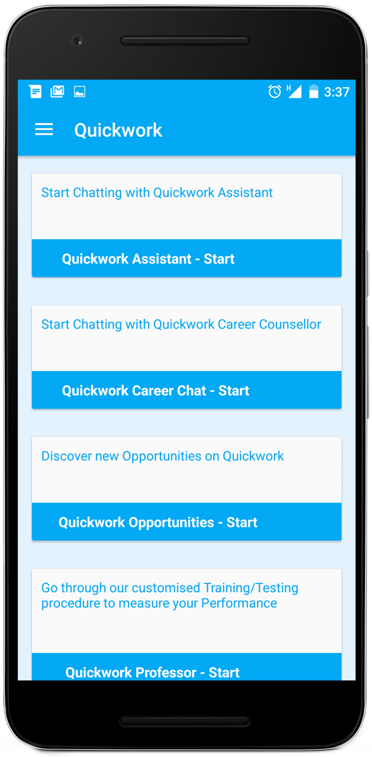
Get Started

9123456789@quickworkapp.com

Please make sure the email address is typed correctly.

NEXT

By pressing 'Next' you agree to our Terms and Conditions and Privacy Policy.



3:37

Quickwork

Start Chatting with Quickwork Assistant

Quickwork Assistant - Start

Start Chatting with Quickwork Career Counsellor

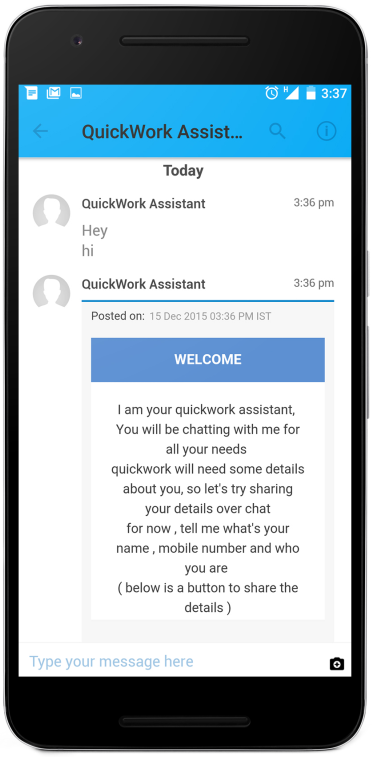
Quickwork Career Chat - Start

Discover new Opportunities on Quickwork

Quickwork Opportunities - Start

Go through our customised Training/Testing procedure to measure your Performance

Quickwork Professor - Start



3:37

QuickWork Assist...

Today

QuickWork Assistant 3:36 pm

Hey
hi

QuickWork Assistant 3:36 pm

Posted on: 15 Dec 2015 03:36 PM IST

WELCOME

I am your quickwork assistant,
You will be chatting with me for
all your needs
quickwork will need some details
about you, so let's try sharing
your details over chat
for now , tell me what's your
name , mobile number and who
you are
(below is a button to share the
details)

Type your message here