1.

Utils ->RoundedNetworkImageView

2.

manifest

<**uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"**/>  
<**uses-permission android:name="android.permission.CAMERA"**/>  
  
<**uses-feature android:name="android.hardware.camera"**/>  
<**uses-feature android:name="android.hardware.camera.autofocus"**/>

3.gradle app

compile **'com.google.firebase:firebase-storage:10.0.1'**

4.

google android firebase -> Storage

copy link:

*//for img***storageRef** = FirebaseStorage.*getInstance*().getReferenceFromUrl(**"gs://polocrazychat-4d8af.appspot.com/"**);

5.

content\_home.xml

<**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"**>  
  
 <**com.kpf.sujeet.chat.Utils.RoundedNetworkImageView  
 android:id="@+id/top\_network\_imageview"  
 android:layout\_gravity="center"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"** />  
  
</**LinearLayout**>

6.

code

#

1.initialise var ->

2.link from storage of firebase->

3.top\_network\_imageview.setOnClickListener->

4. selectImage()->

5. onRequestPermissionsResult ->

6. onActivityResult->

7. Query … = **databaseReference->**

8.upLoadProfileImage(Uri file)

1.

*//for img***private int REQUEST\_CAMERA** = 0;  
**private int SELECT\_FILE** = 1;  
**private** Bitmap **thumbnail**;  
**private** String **selectedImagePath**;  
BitmapFactory.Options **options**;  
  
StorageReference **storageRef**;

2.(get address from google firebase database->storage.)

*//for img***storageRef** = FirebaseStorage.*getInstance*().getReferenceFromUrl(**"gs://polocrazychat-4d8af.appspot.com/"**);

3.

*//for uploading img from camera or gallery* **top\_network\_imageview**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 selectImage();  
  
 }  
 });

7. Query query1 = **databaseReference**.child(**"users"**).child(**mAuth**.getCurrentUser().getUid()).child(**"photoUrl"**);  
 query1.addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 **if**(dataSnapshot.getKey().equals(**"photoUrl"**)){  
 **img\_logged\_user\_dp**.setImageUrl(dataSnapshot.getValue().toString(), **imageLoader**);  
 **top\_network\_imageview**.setImageUrl(dataSnapshot.getValue().toString(), **imageLoader**);  
  
 }  
 }  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
}

6.

*//for img upload*@Override  
**protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
 **if** (resultCode == ***RESULT\_OK***) {  
 **if** (requestCode == **REQUEST\_CAMERA**) {  
 *//for uploading from camera* **thumbnail** = (Bitmap) data.getExtras().get(**"data"**);  
 ByteArrayOutputStream bytes = **new** ByteArrayOutputStream();  
 **thumbnail**.compress(Bitmap.CompressFormat.***PNG***, 90, bytes);  
  
 File destination = **new** File(Environment.*getExternalStorageDirectory*(), System.*currentTimeMillis*() + **".PNG"**);  
 FileOutputStream fo;  
 **try** {  
 destination.createNewFile();  
 fo = **new** FileOutputStream(destination);  
 fo.write(bytes.toByteArray());  
 fo.close();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
  
  
 **if** (**firebaseUser**.getUid() != **null** && **thumbnail** != **null**) {  
 *//for profile img* Bitmap bitmap = **thumbnail**;  
  
 String path = MediaStore.Images.Media.*insertImage*(getContentResolver(), bitmap, **"Title"**, **null**);  
 upLoadProfileImage(Uri.*parse*(path));  
  
 } **else** {  
  
 }  
 }**else if**(requestCode ==**SELECT\_FILE**){  
 *//for selecting from gallery* Uri selectedImageUri = data.getData();  
 String[] projection = {MediaStore.MediaColumns.***DATA***};  
 Cursor cursor = managedQuery(selectedImageUri,projection,**null**,**null**,**null**);  
 **int** column\_index = cursor.getColumnIndexOrThrow(MediaStore.MediaColumns.***DATA***);  
 cursor.moveToFirst();  
 **selectedImagePath** = cursor.getString(column\_index);  
  
 **options** = **new** BitmapFactory.Options();  
 **options**.**inJustDecodeBounds** = **true**;  
 BitmapFactory.*decodeFile*(**selectedImagePath**,**options**);  
  
 **final int** REQUIRED\_SIZE = 200;  
 **int** scale =1;  
 **while**(**options**.**outWidth** / scale / 2 >= REQUIRED\_SIZE && **options**.**outHeight** / scale / 2 >= REQUIRED\_SIZE)  
 scale \*=2;  
 **options**.**inSampleSize** = scale;  
 **options**.**inJustDecodeBounds** = **false**;  
 **thumbnail** = BitmapFactory.*decodeFile*(**selectedImagePath**,**options**);  
  
 **if**(**firebaseUser**.getUid() !=**null** && **thumbnail** != **null**){  
 Bitmap bitmap = **thumbnail**;  
  
 String path = MediaStore.Images.Media.*insertImage*(getContentResolver(),bitmap,**"Title"**,**null**);  
 upLoadProfileImage(Uri.*parse*(path));  
  
 }**else**{  
  
 }  
 }  
  
 }  
}

4.

**private void** selectImage(){  
 **final** CharSequence[] items = {**"Take Photo"**,**"Choose from gallery"**,**"Cancel"**};//msg inside “” must match exactly  
  
 AlertDialog.Builder builder = **new** AlertDialog.Builder(**this**);//alertdialogbox for displaying popup.  
 builder.setTitle(**"Add Photo"**);  
 builder.setItems(items, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** item) {  
 **if**(items[item].equals(**"Take Photo"**)){  
 *//cheking for marshmellow version* **if**(Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***M***) {  
 **if** (checkSelfPermission(android.Manifest.permission.***CAMERA***) != PackageManager.***PERMISSION\_GRANTED***) {  
  
 requestPermissions(**new** String[]{android.Manifest.permission.***CAMERA***}, **REQUEST\_CAMERA**);  
 }  
 }**else**{  
 Intent intent = **new** Intent(MediaStore.***ACTION\_IMAGE\_CAPTURE***);  
 startActivityForResult(intent,**REQUEST\_CAMERA**);*//handled through onActivityResult* }

}**else if**(items[item].equals(**"Choose from gallery"**)){  
 **if**(Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***M***){  
 **if**(checkSelfPermission(android.Manifest.permission.***WRITE\_EXTERNAL\_STORAGE***) != PackageManager.***PERMISSION\_GRANTED***){  
 requestPermissions(**new** String[]{android.Manifest.permission.***WRITE\_EXTERNAL\_STORAGE***},**SELECT\_FILE**);  
 }  
 }**else**{  
 Intent intent = **new** Intent(Intent.***ACTION\_PICK***,MediaStore.Images.Media.***EXTERNAL\_CONTENT\_URI***);  
 intent.setType(**"image/\*"**);*//picks up* ***every*** *img from gallery,not other formats* startActivityForResult(Intent.*createChooser*(intent,**"Select File"**),**SELECT\_FILE**);  
  
 }  
 } **else if**(items[item].equals(**"Cancel"**)){  
 dialogInterface.dismiss();  
 }  
 }  
 });  
 builder.show();  
  
}

5.

@Override  
**public void** onRequestPermissionsResult(**int** requestCode, @NonNull String[] permissions, @NonNull **int**[] grantResults) {  
 **super**.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 **if**(requestCode == **REQUEST\_CAMERA**){  
 **if**(grantResults[0] == PackageManager.***PERMISSION\_GRANTED***){  
 *//for camera* Intent intent=**new** Intent(MediaStore.***ACTION\_IMAGE\_CAPTURE***);  
 startActivityForResult(intent,**REQUEST\_CAMERA**);  
 }**else**{  
  
 }  
  
 }**else if** (requestCode ==**SELECT\_FILE**){  
 **if**(grantResults[0] == PackageManager.***PERMISSION\_GRANTED***){  
 Intent intent = **new** Intent(Intent.***ACTION\_PICK***,MediaStore.Images.Media.***EXTERNAL\_CONTENT\_URI***);  
 intent.setType(**"image/\*"**);*//picks up every img from gallery,not other formats* startActivityForResult(Intent.*createChooser*(intent,**"Select File"**),**SELECT\_FILE**);  
  
 }**else** {  
  
 }  
  
 }  
}

8.

**public void** upLoadProfileImage(Uri file){  
  
StorageMetadata metadata = **new** StorageMetadata.Builder().setContentType(**"image/jpg"**).build();*//only info of file in metadata* StorageReference storageReference = **storageRef**.child(**firebaseUser**.getUid()).child(**"profileImage"**);  
  
 UploadTask uploadTask= storageReference.child(**"image\_dp.jpg"**).putFile(file,metadata);*//only one img would be replaced everytime* uploadTask.addOnProgressListener(**new** OnProgressListener<UploadTask.TaskSnapshot>() {  
 @Override  
 **public void** onProgress(UploadTask.TaskSnapshot taskSnapshot) {  
  
 }  
 }).addOnSuccessListener(**new** OnSuccessListener<UploadTask.TaskSnapshot>() {  
 @Override  
 **public void** onSuccess(UploadTask.TaskSnapshot taskSnapshot) {  
  
 **databaseReference**.child(**"users"**).child(**firebaseUser**.getUid()).child(**"photoUrl"**).setValue(taskSnapshot.getMetadata().getDownloadUrl().toString());*//not storage ,photourl is database variable* }  
 });  
 }  
}