Aqui vamos recuperar direto do storage com o nome da imagem

# Primeira maneira de fazer

binding.btnRecuperar.setOnClickListener **{** recuperarImagemFirebase()  
 **}**}  
  
private fun recuperarImagemFirebase() {  
 val idUsuarioLogado = FirebaseAuth.getInstance().*currentUser*?.*uid* if (idUsuarioLogado != null ) {  
  
 armazenamento.getReference("fotos")  
 .child(idUsuarioLogado) // pasta com id do usuário.  
 .child("foto.jpg") // Nesse caso vamos estar recuperando pelo proprio nome do arquivo.  
 .*downloadUrl* // Aqui vai bixar a URL da imagem para a gente  
 .addOnSuccessListener **{** urlFirebase **->** // Se for sucesso - vai retornar a URL la do firebase  
 Toast.makeText(this, "URL recuperado: $urlFirebase", Toast.*LENGTH\_SHORT*).show()  
 binding.imageRecuperada.setImageURI(urlFirebase)  
 **}** }  
}

Agora a outra maneira melhor de fazer

# Usando Picaso

Essa maneira ele vai baixar a imagem, fazer tratamento de erros, fazer um cash da imagem para você não precisar ficar baixando ela toda hora...

Pra isso vamos jogar no google Picaso Android

Vamos implementar

// Picaso  
*implementation* ("com.squareup.picasso:picasso:2.8")

Implementando

private fun recuperarImagemFirebase() {  
 val idUsuarioLogado = FirebaseAuth.getInstance().*currentUser*?.*uid* if (idUsuarioLogado != null ) {  
  
 armazenamento.getReference("fotos")  
 .child(idUsuarioLogado)  
 .child("foto.jpg") // Nesse caso vamos estar recuperando pelo proprio nome do arquivo.  
 .*downloadUrl* // Aqui vai bixar a URL da imagem para a gente  
 .addOnSuccessListener **{** urlFirebase **->** // Se for sucesso - vai retornar a URL la do firebase  
 Toast.makeText(this, "URL recuperado: $urlFirebase", Toast.*LENGTH\_SHORT*).show()  
  
 **Picasso.get()  
 .load(urlFirebase)  
 .resize(200, 160)  
 .centerCrop()  
 .into(binding.imageRecuperada)**  
   
 **}** }  
}

Ali vamos passar o url

Tamanho

Como vai ser adicionado

E onde esta o arquivo...

Mas o ideal e recuperar sempre do banco de dados o firestore...