|  |
| --- |
| BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**  **BÁO CÁO TỔNG KẾT**  **Ứng Dụng BlogApp Sử Dụng**  **FireBase**  **Đề tài: Số 15**  **Thành viên:**  **Phạm Hải Châu – MSSV: 1651012019**  **Lê Quang Đại Lộc – MSSV**  **Khoa: Công Nghệ Thông Tin**  **TP Hồ Chí Minh, 04/2019** |

[**Chương I:** **GIỚI THIỆU VỀ FIREBASE** 3](#_Toc6267740)

[**Chương II:** **CÁC CHỨC NĂNG CỦA BLOGAPP** 5](#_Toc6267741)

[**2.1** **Chức năng đăng ký, đăng nhập trong ứng dụng** 5](#_Toc6267742)

[**2.1.1** **Kết nối Firebase** 5](#_Toc6267743)

[**2.1.2** **Kích hoạt email và password trong Firebase** 6](#_Toc6267744)

[**2.1.3** **Tạo account trên Firebase** 7](#_Toc6267745)

[**2.1.4** **Đăng nhập với tài khoản đã có trên Firebase** 8](#_Toc6267746)

[**2.2** **Chức năng chỉnh sửa ảnh đại diện và tên hiển thị** 9](#_Toc6267747)

[**2.2.1** **Cắt ảnh** 9](#_Toc6267748)

[**2.2.2** **Lưu thông tin tài khoản trên Firebase** 13](#_Toc6267749)

[**2.3** **Chức năng đăng bài viết** 17](#_Toc6267750)

[**2.3.1** **HomeActivity** 17](#_Toc6267751)

[**2.3.2** **Post Activity** 19](#_Toc6267752)

[**2.3.3** **Cắt Ảnh** 21](#_Toc6267753)

[**2.3.4** **Update dữ liệu bài viết lên Firebase** 22](#_Toc6267754)

[**2.3.5** **Đổ dữ liệu về HomeActivity** 24](#_Toc6267755)

[**2.4** **Chức năng like, comment** 25](#_Toc6267756)

[**2.4.1** **Chức năng Like** 25](#_Toc6267757)

[**2.4.2** **Chức năng Comment** 26](#_Toc6267758)

1. **GIỚI THIỆU VỀ FIREBASE**

Firebase là một dịch vụ hệ thống backend được Google cung cấp sẵn cho ứng dụng Mobile của bạn, với Firebase bạn có thể rút ngắn thời gian phát triển, triển khai và thời gian mở rộng quy mô của ứng dụng mobile mình đang phát triển. Hỗ trợ cả 2 nền tảng Android và IOS, Firebase mạnh mẽ, đa năng, bảo mật và là dịch vụ cần thiết đầu tiên để xây dưng ứng dụng với hàng triệu người sử dụng. Sử dụng Firebase bạn sẽ có được hưởng các lợi ích sau:

* Xây dựng ứng dụng nhanh chóng mà không tốn thời gian, nhân lực để quản lý hệ thống và cơ sơ sở hạ tầng phía sau: Firebase cung cấp cho bạn chức năng như phân tích, cơ sở dữ liệu, báo cáo hoạt động và báo cáo các sự cố lỗi để bạn có thể dễ dàng phát triển, định hướng ứng dụng của mình vào người sử dụng nhằm đem lại các trải nghiệm tốt nhất cho họ.
* Uy tín chất lượng đảm bảo từ Google: Firebase được google hỗ trợ và cung cấp trên nền tảng phần cứng với quy mô rộng khắp thế giới, được các tập đoàn lớn và các ưng dụng với triệu lượt sử dụng từ người dùng.
* Quản lý cấu hình và trải nghiệm các ứng dụng của Firebase tập trung trong một giao diện website đơn giản, các ứng dụng này hoạt động độc lập nhưng liên kết dữ liệu phân tích chặt chẽ

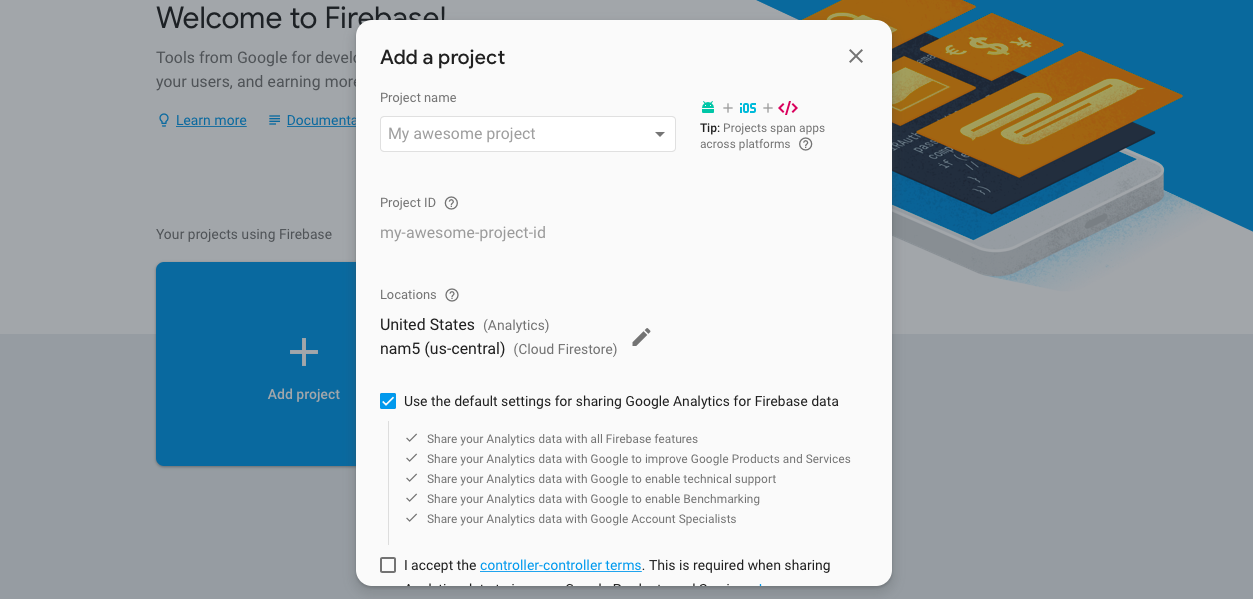
**Tổng quan các dịch vụ mà Firebase cung cấp:**

Firebase cung cấp cho chúng ta 2 nhóm sản phẩm chính tập trung vào 2 đối tượng là:

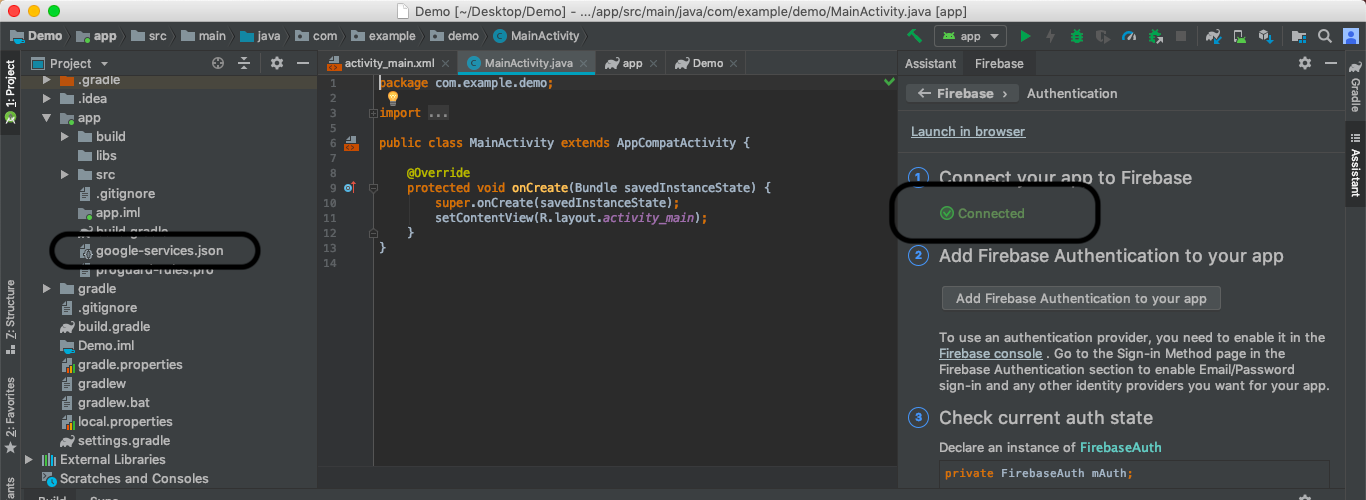
* Develop & test your app: phát triển và kiểm thử các ứng dụng được thiết kế.
* Grow & engage your audience: Phân tích dữ liệu và tối ưu hóa trải nghiệm đối với người dùng.

Kết nối với ứng dụng với Firebase

* Tạo 1 project ở Android Studio.
* Vào <https://console.firebase.google.com> chọn Add Project .



* Đặt tên cho project và nhấn Create Project.
* Để kết nối Firebase và app mà không cần làm các bước giống Firebase hướng dẫn ta vào Android Studio chọn Tool -> Firebase -> Authentication -> Connect to Firebase -> Chọn Firebase mình vừa tạo ở trên nhấn Connect. Lúc này, app mình đã tự động kết nối thành công với firebase.



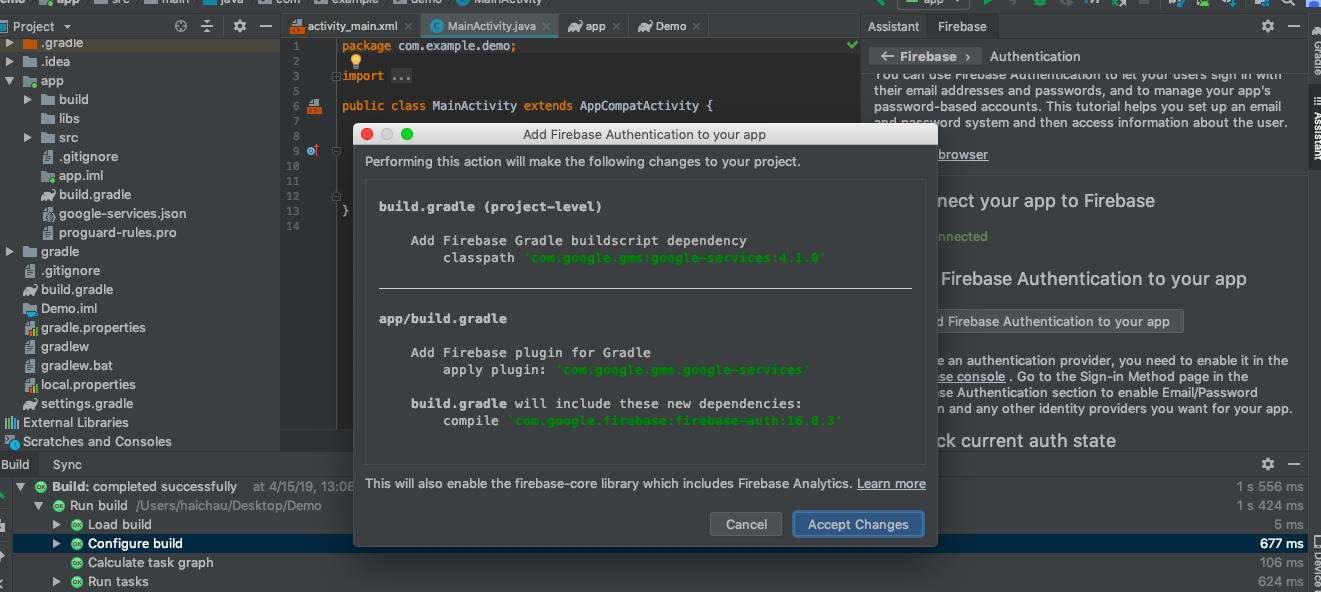
1. **CÁC CHỨC NĂNG CỦA BLOGAPP**

Blog App gồm 4 chức năng:

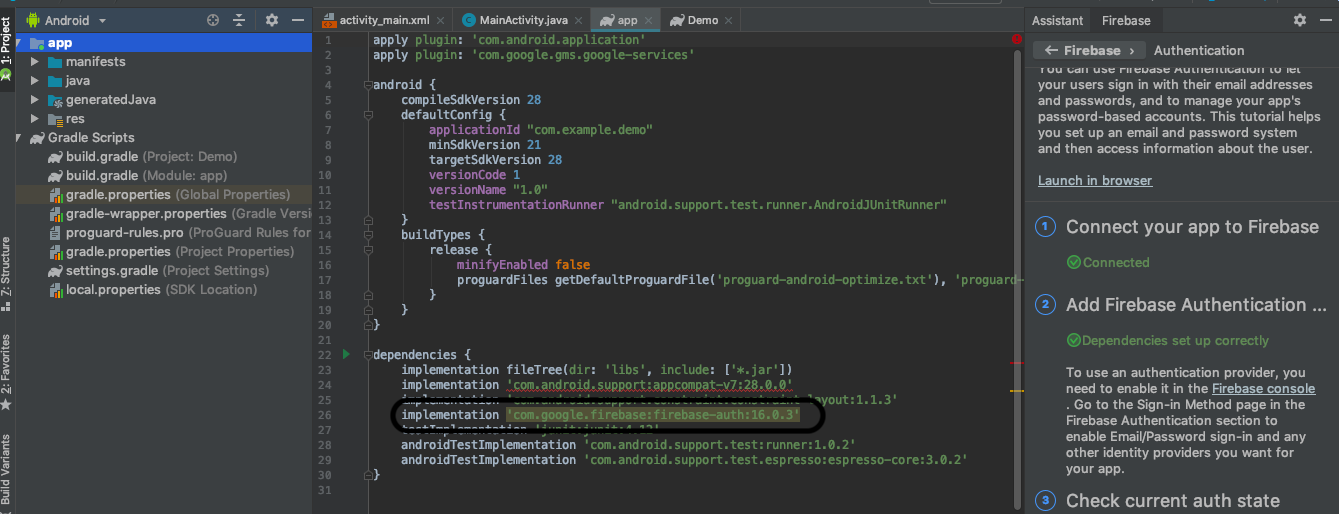
* Đăng ký tài khoản và dung tài khoản đã đăng ký đăng nhập vào Blog App.
* Chỉnh sửa tên hiển thị và ảnh đại diện của tài khoản.
* Đăng bài viết lên Blog App
* Thích và Bình luận các bài viết khác.

1. **Chức năng đăng ký, đăng nhập trong ứng dụng**
2. **Kết nối Firebase**

* Để có thể sử dụng được dịch vụ đăng ký, đăng nhập của Firebase, vào Android Studio chọn Tool -> Firebase -> Authentication -> Add Firebase Authentication to your app.



* Nhấn Accept Changes. Lúc này ở file build.gradle(app) đã tự động implement thư viện để có thể sử dụng dịch vụ đăng ký của Firebase

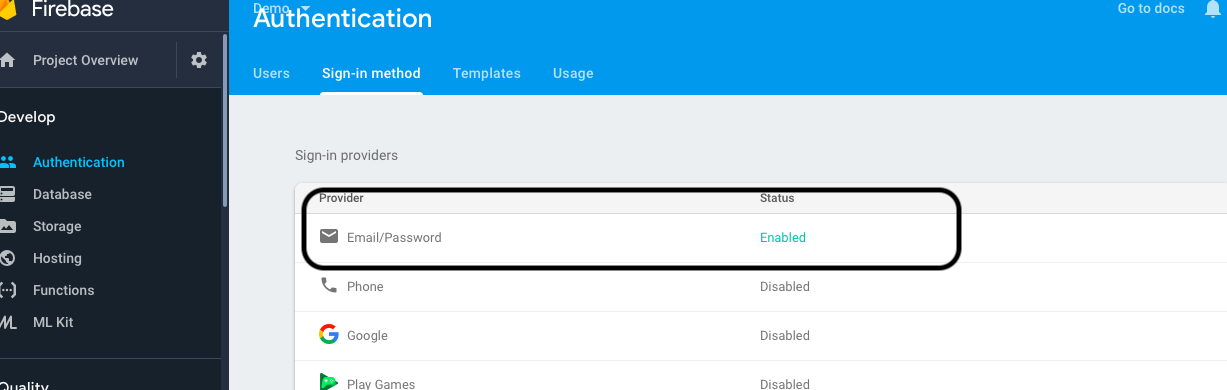


1. **Kích hoạt email và password trong Firebase**

* Trước khi đăng ký tài khoản ở firebase, ta phải vào project ta vừa tạo ở web firebase -> Chọn Authenication

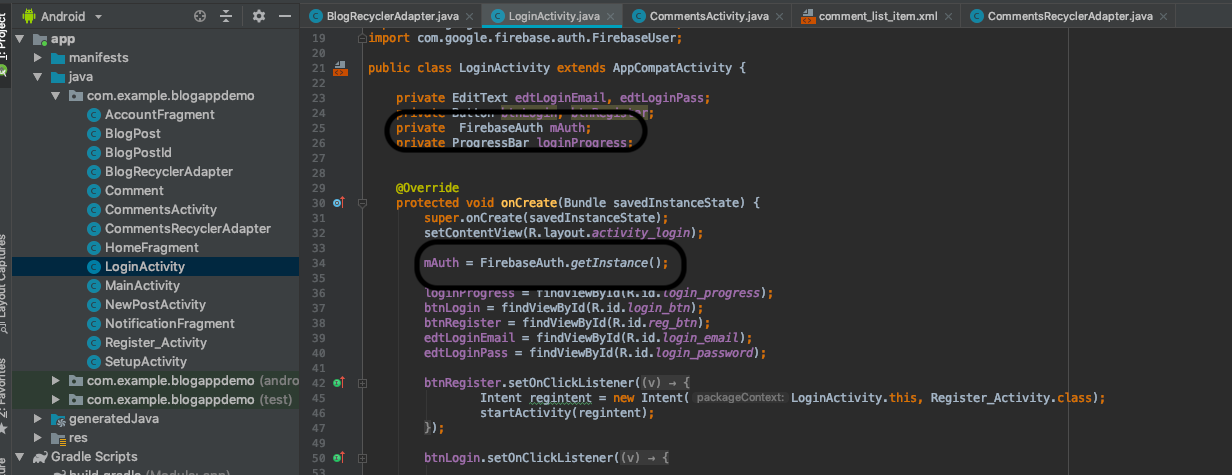


* Chọn Set up sign-in method, sau đó Enable Email/Password.



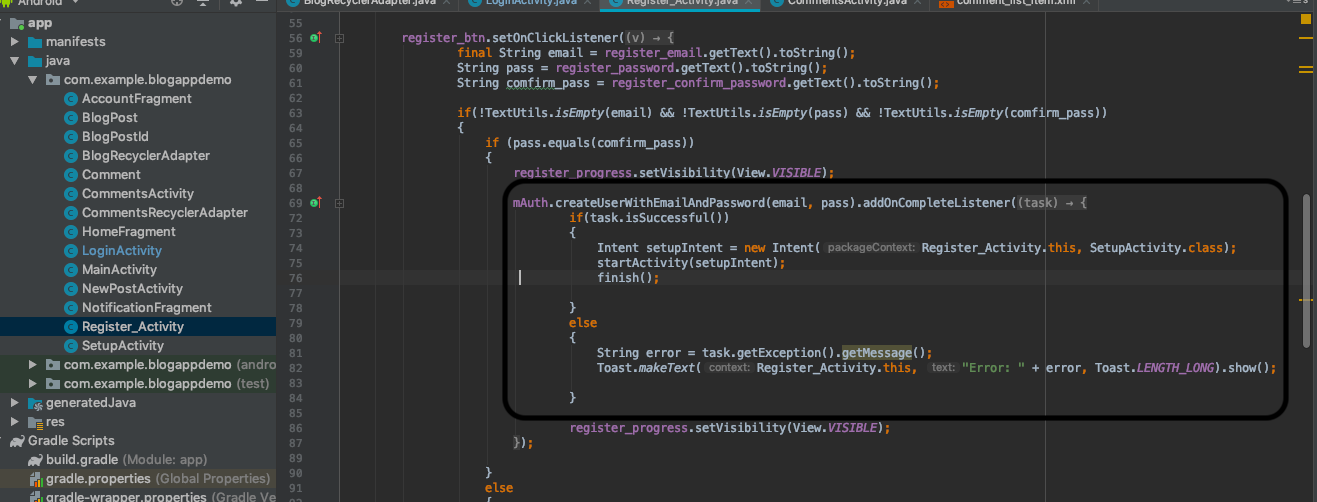
1. **Tạo account trên Firebase**

* Để tạo chức năng đăng ký tài khoản trong app, ta chỉ cần khởi tạo đối tượng FirebaseAuth

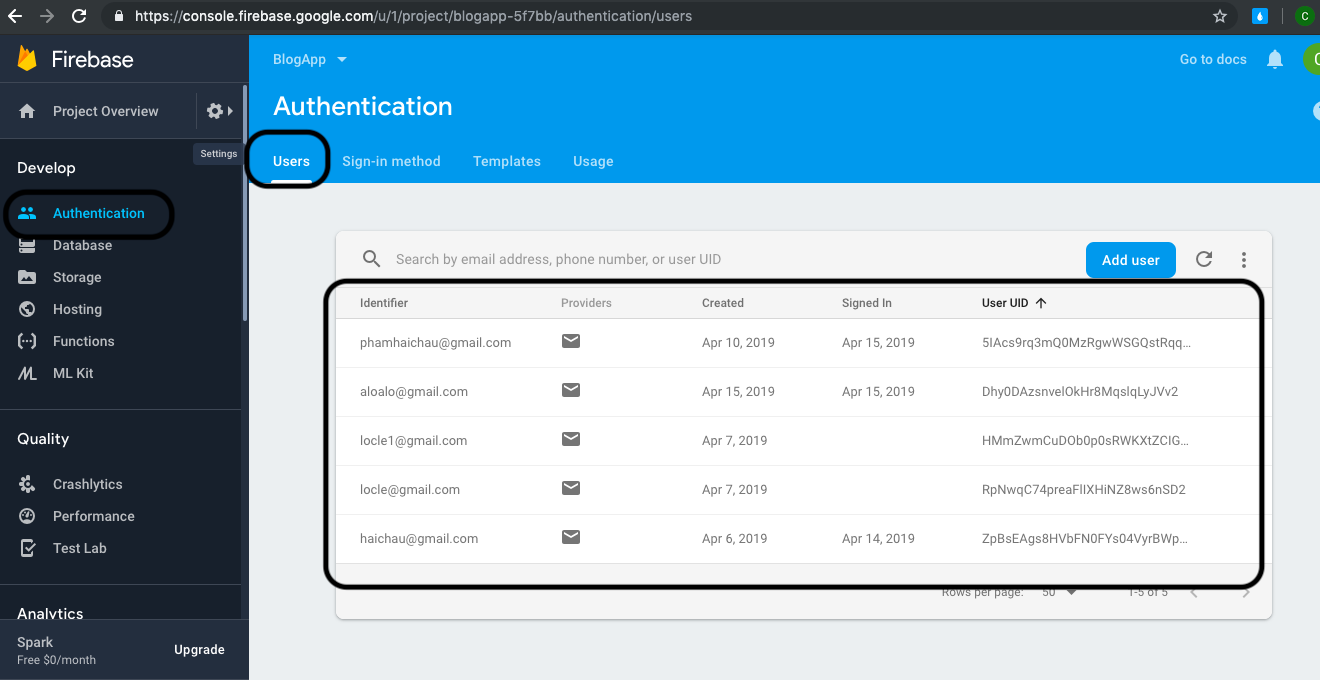


* Sử dụng phương thức

createUserWithEmailAndPassword(email, password) để đăng ký tài khoản lên Firebase.

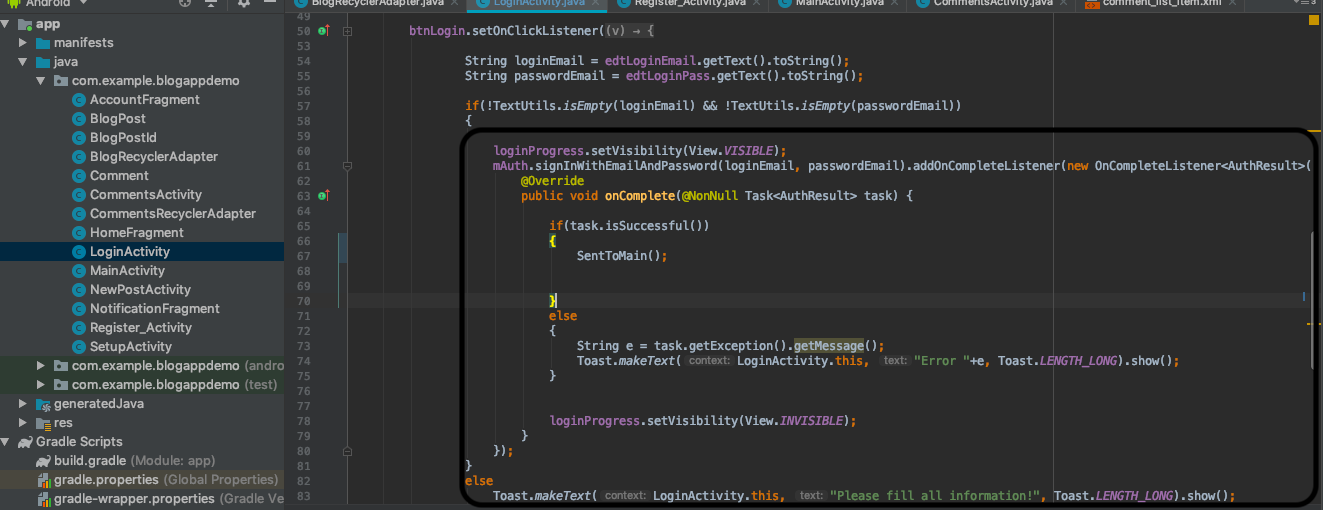


* Sau khi tạo thành công tài khoản Firebase sẽ lưu thông tin tài khoản như hình bên dưới.



1. **Đăng nhập với tài khoản đã có trên Firebase**

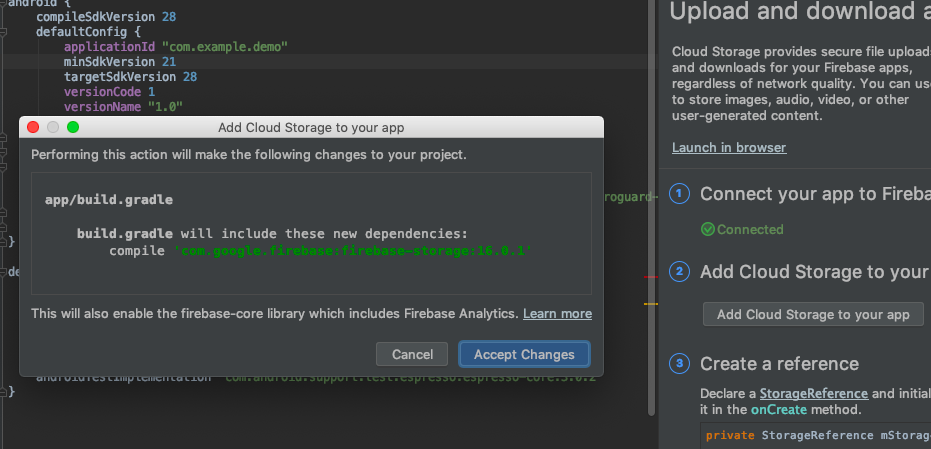
* Với chức năng đăng nhập ta cũng khởi tạo đối tượng FirebaseAuth, sau đó sử dụng phương thức signInWithEmailAndPassword(email, password) là đã có thể đăng nhâp vào BlogApp.



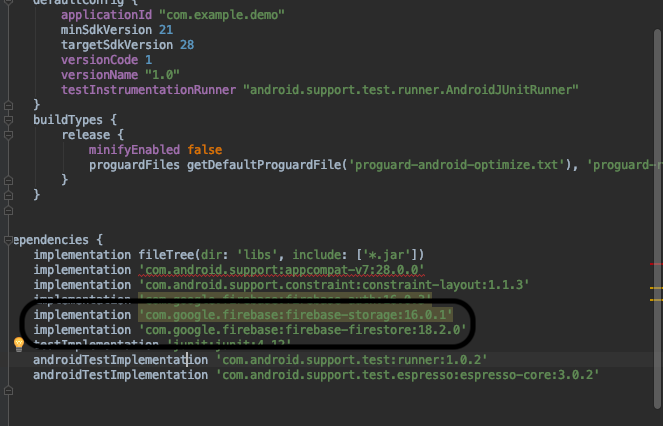
1. **Chức năng chỉnh sửa ảnh đại diện và tên hiển thị**
2. **Cắt ảnh**

* Đầu tiên ta kết nối app đến Storage trên Firebase, vào Tool -> Firebase

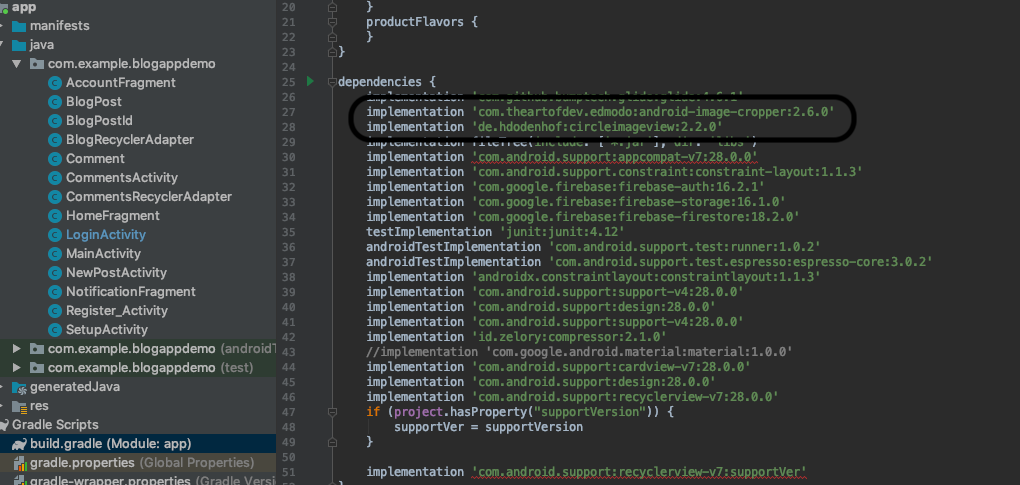
-> Storage, chọn Add Cloud Storage to your app.



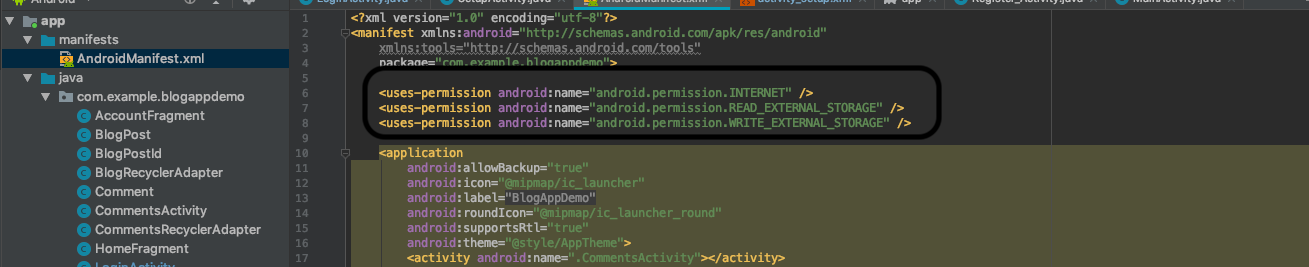
* Sau đó gõ implementation ‘com.google.firebase:firestore:18.2.0. File gradle(app) sẽ như thế này



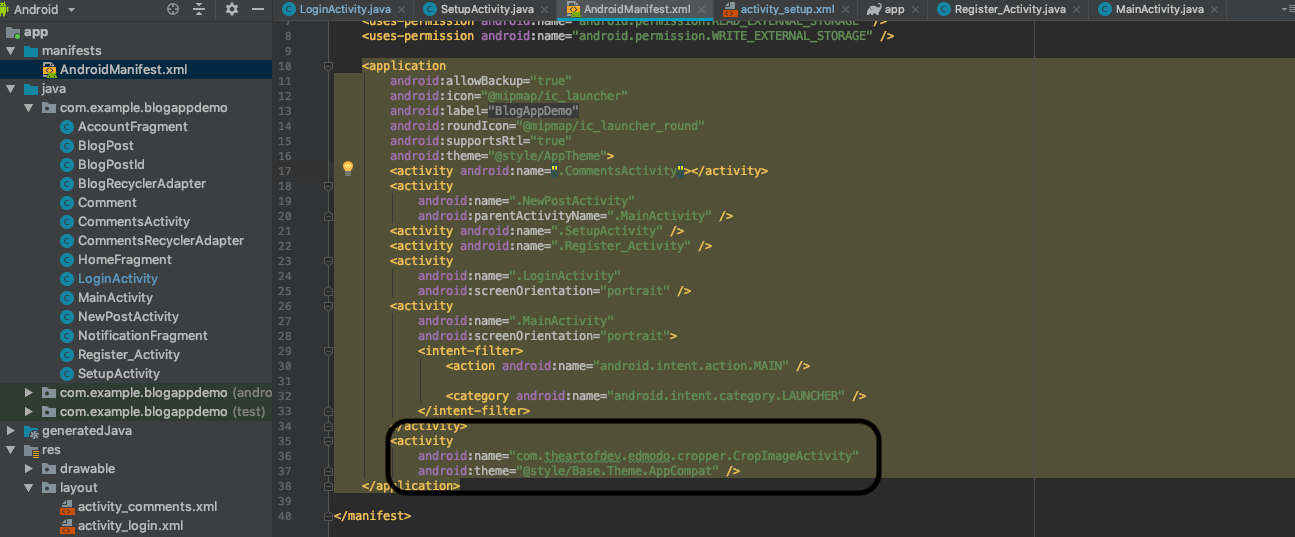
* Ở phần này trước khi vào làm ta cần phải implement 2 thư viện



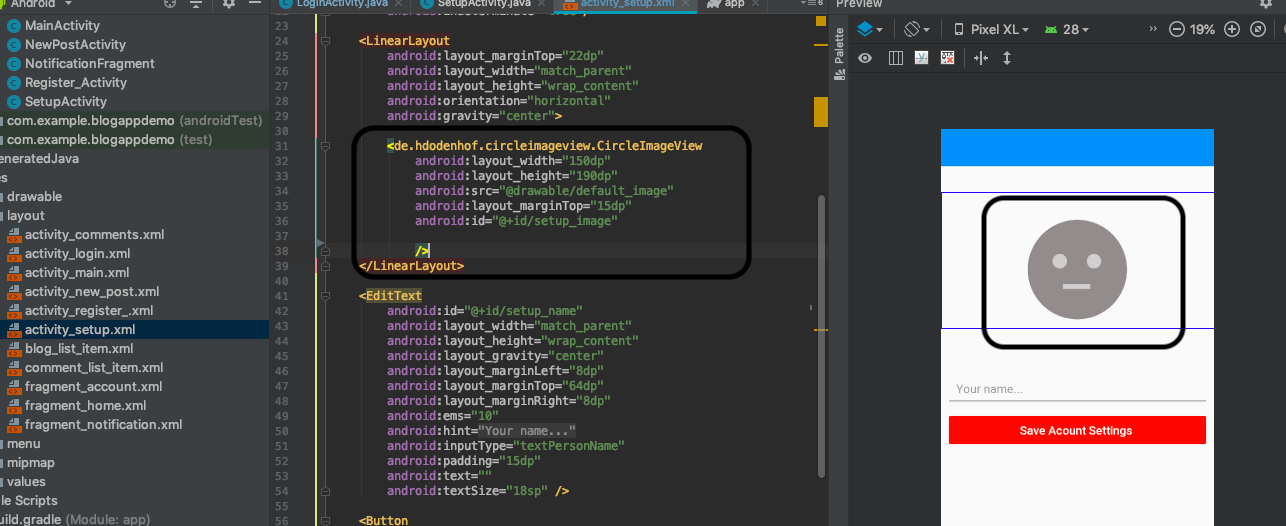
* Xin cấp quyền đọc và ghi dữ liệu với bộ nhớ ngoài ở file AndroidManifest.xml



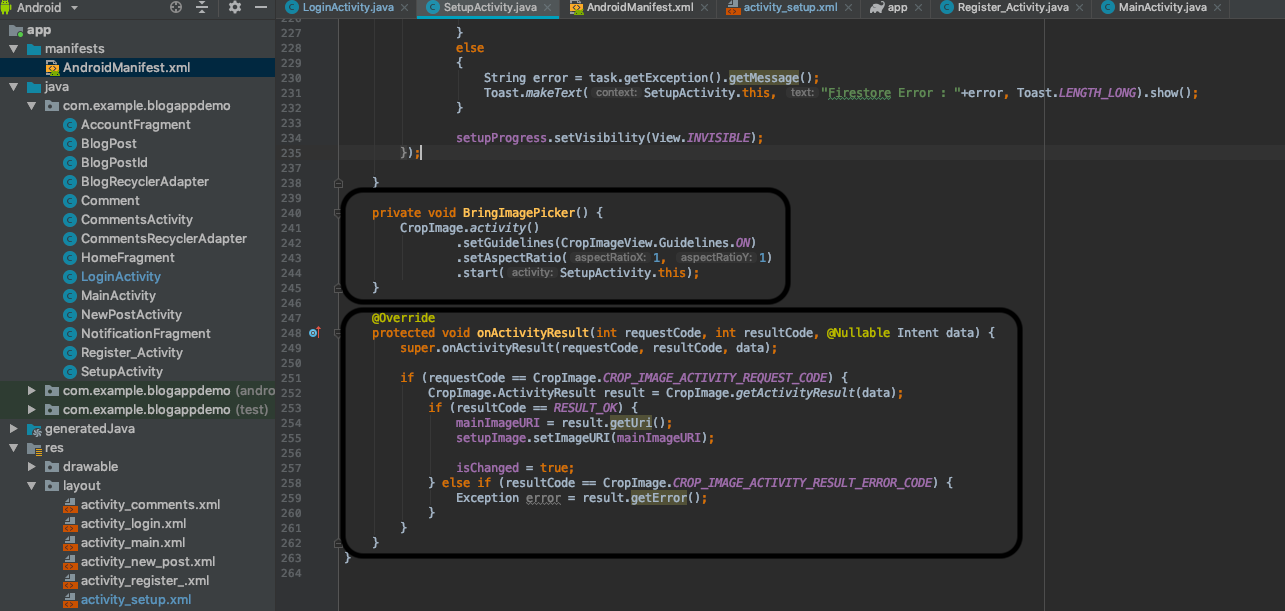
* Thêm CropImageActivity vào file AndroidManifest.xml

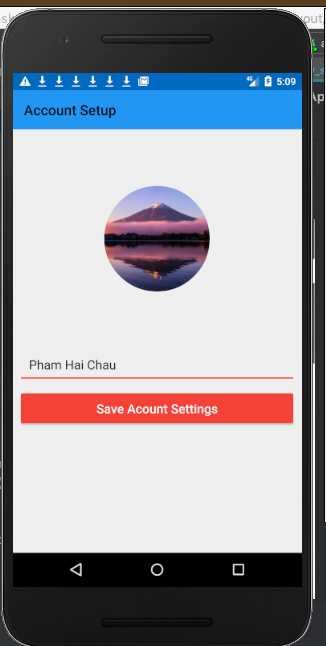


* Thư viện ‘de.hdodenhof:circleimageview:2.2.0’ giúp ta có thể sử dụng thẻ CircleImageView ở file xml.



* Thư viện ‘com.theartofdev.edmodo:android-image-cropper:2.6.0’ giúp chúng ta cắt ảnh và trả ảnh về activity mà ta muốn.



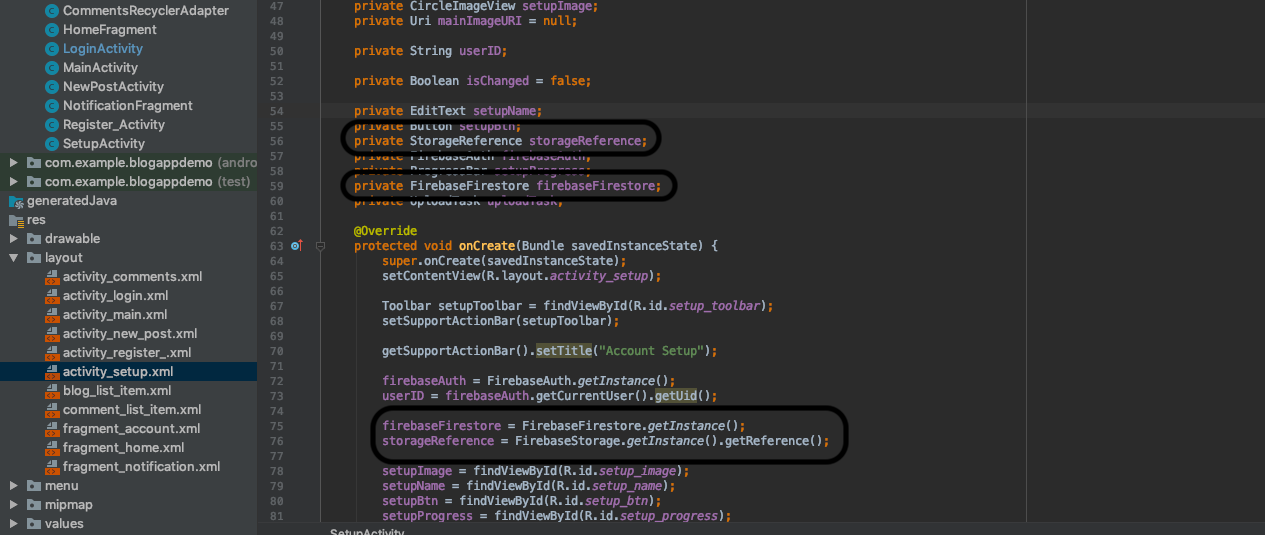


1. **Lưu thông tin tài khoản trên Firebase**

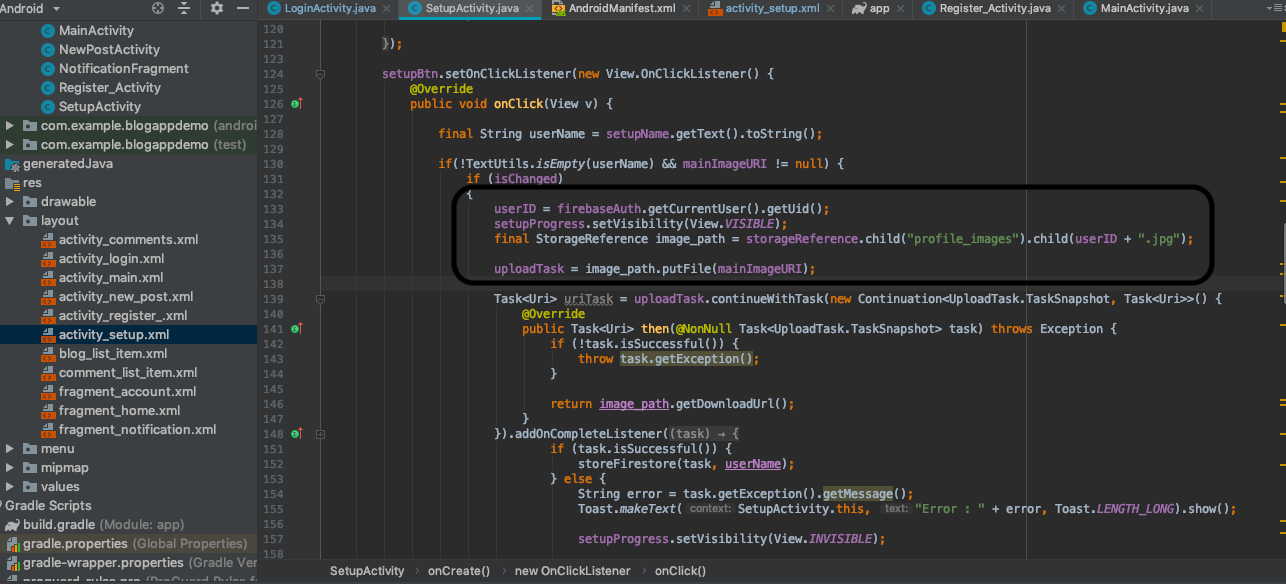
* Khi nhấn nút Save Account Settings thì app sẽ lấy ra ID tài khoản đang dùng và truy cập vào firebase lưu lại tên hiển thị và ảnh đại diện vừa được thay đổi. Ở firebase ảnh đại diện sẽ được lưu ở Storage. Để lưu được ảnh ở Storage, vào firebase chọn Storage -> Rules và sửa lại như ảnh ảnh bên dưới.

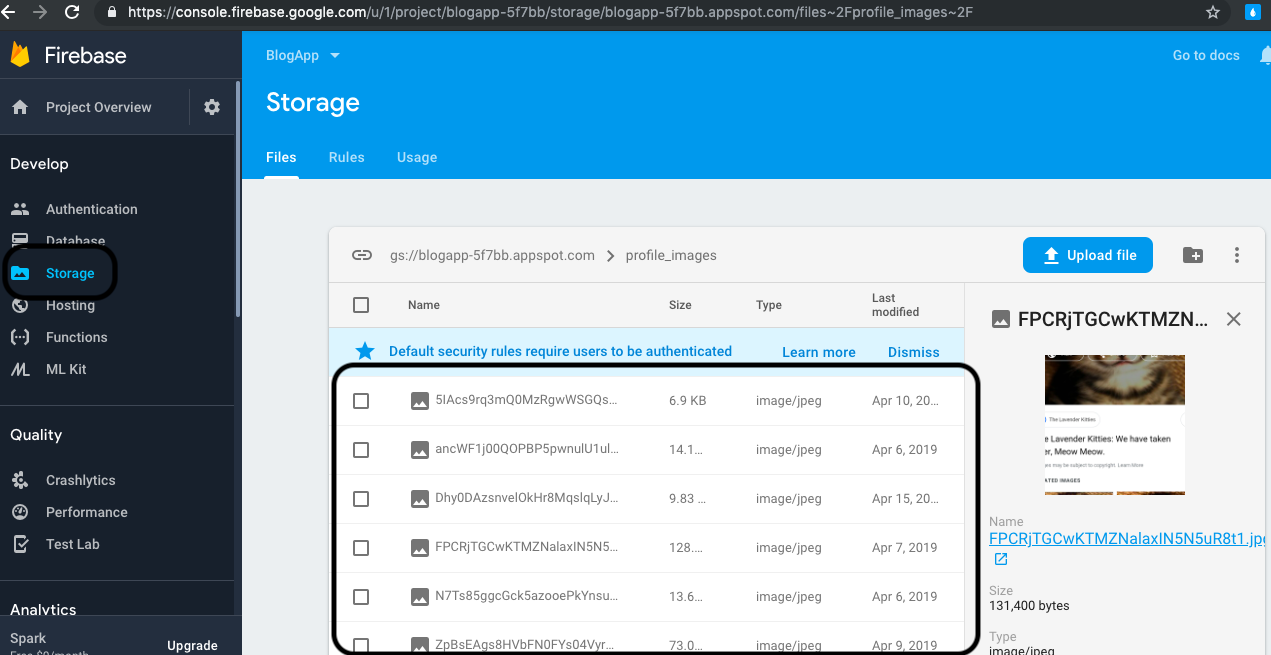


* Tạo đối tượng StorageReference và FirebaseFirestore

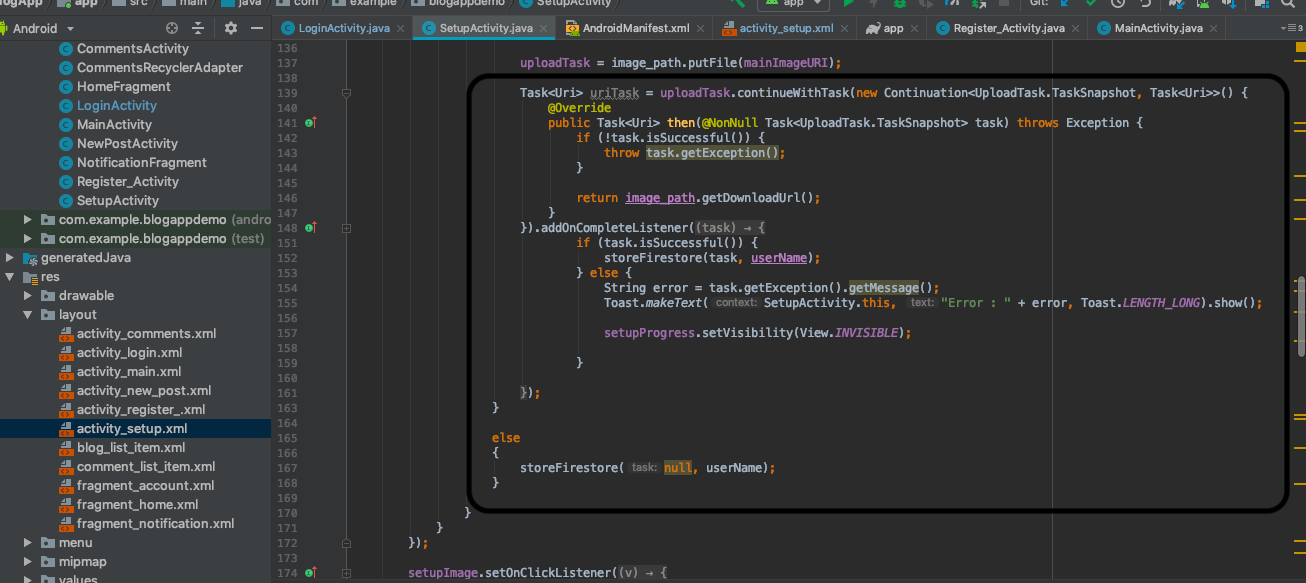


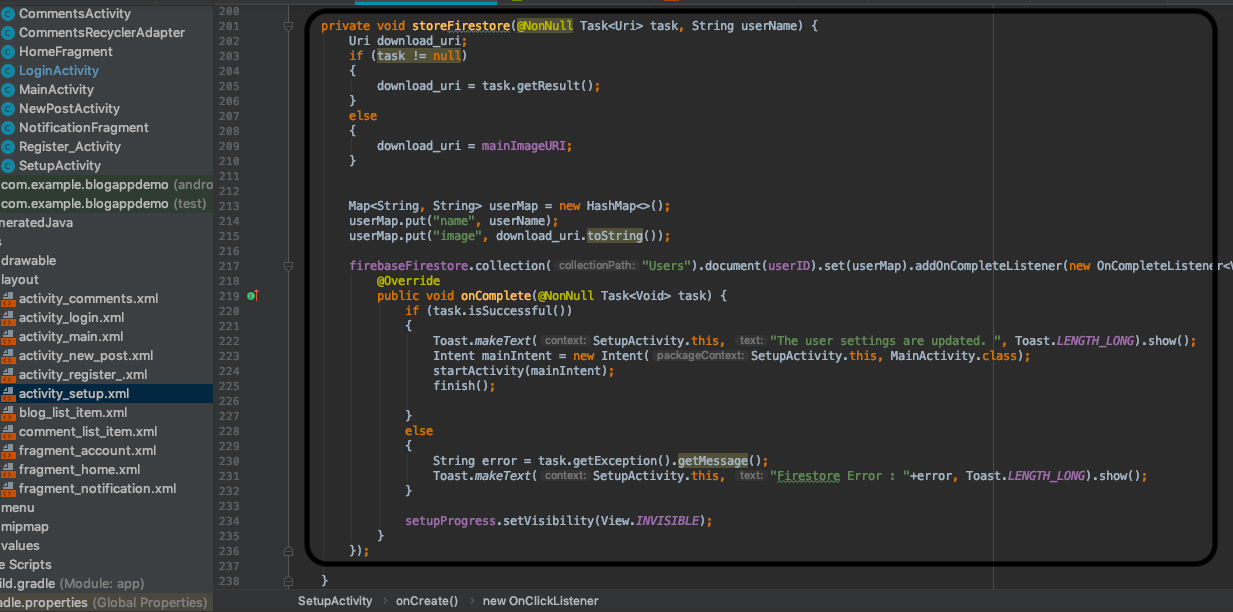
* Storage sẽ lưu ảnh đại diện dưới tên được lấy từ ID của email đang dùng và đuôi là đuôi .jpg

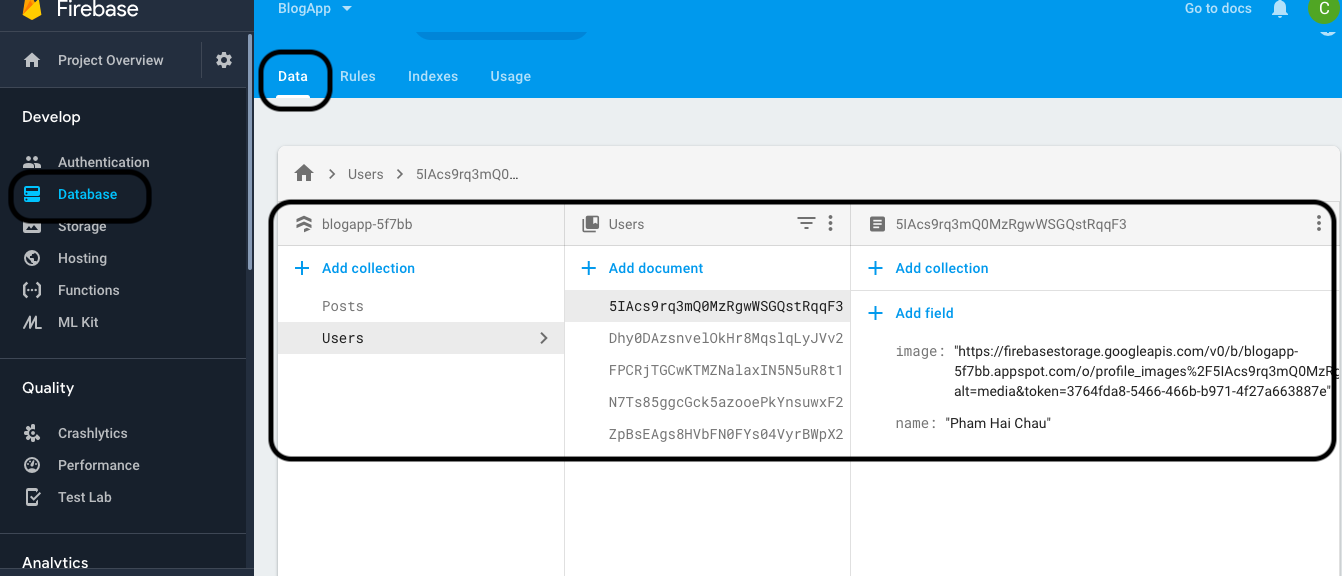




* Sau đó, lấy ảnh đã lưu trên Storage và lưu lại tên hiển thị và ảnh đại diện vào Database

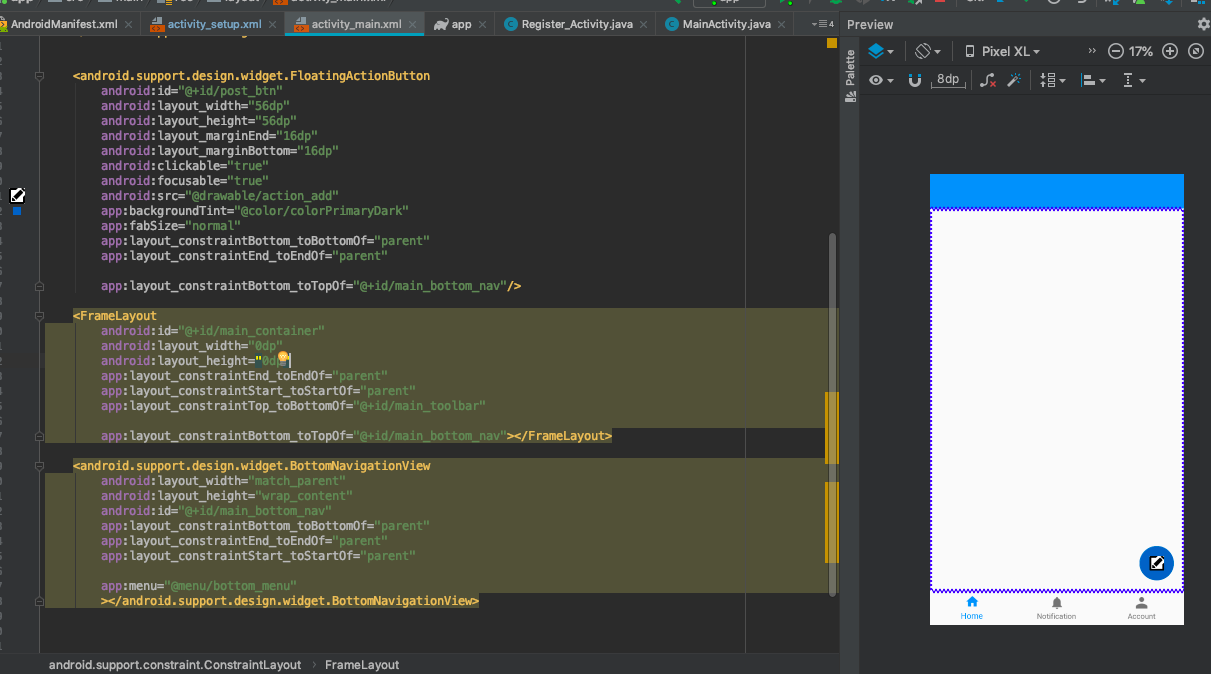




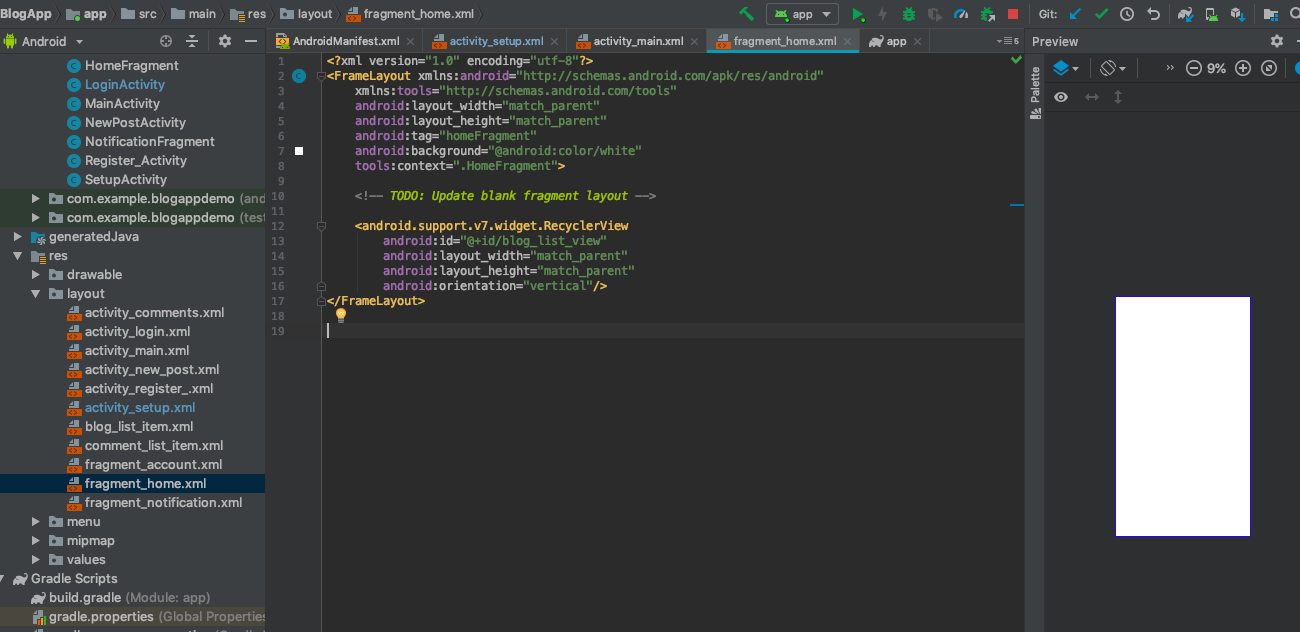
* Database sẽ lưu lại tên hiển thị và đường dẫn ảnh đại diện ở collection(“Users”), Document(“IDEmail”)

1. **Chức năng đăng bài viết**
2. **HomeActivity**

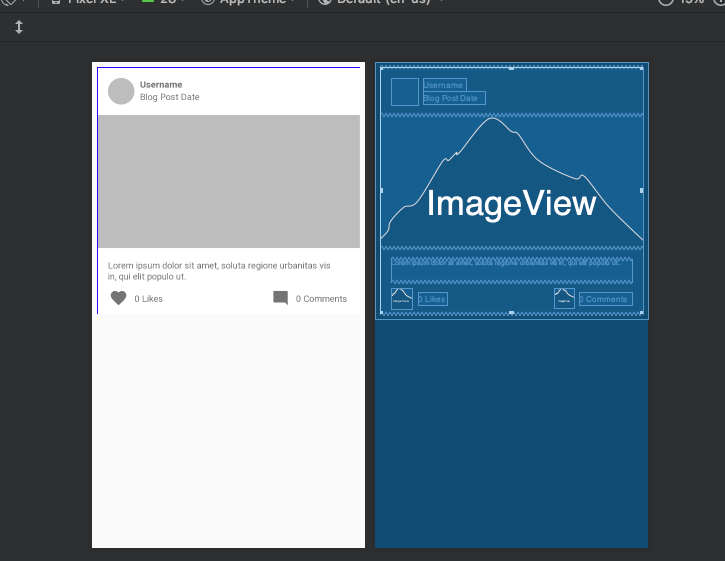
* Sau khi đăng nhập hoặc chỉnh sửa tài khoản xong thì đều sẽ trở về 1 Activity chính. Activity chính như hình bên dưới.



* Thẻ FrameLayout dùng để chứa các Fragment bỏ vào FrameLayout mỗi khi nhấn vào Home, Notification hoặc Account.
* Khi nhấn vào Home, Activity chính sẽ chứa HomeFragment. Trong Fragment này sẽ hiện ra danh sách các bài số, số lượt like, comment của bài viết. Trong file xml của HomeFragment sẽ có thẻ FrameLayout và chứa duy nhất 1 RecyclerView, RecyclerView này sẽ chứa danh sách các bài viết.

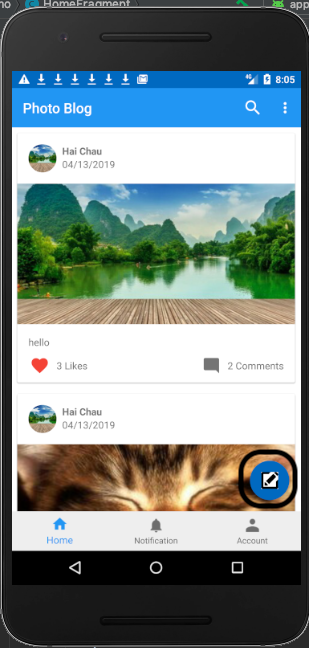


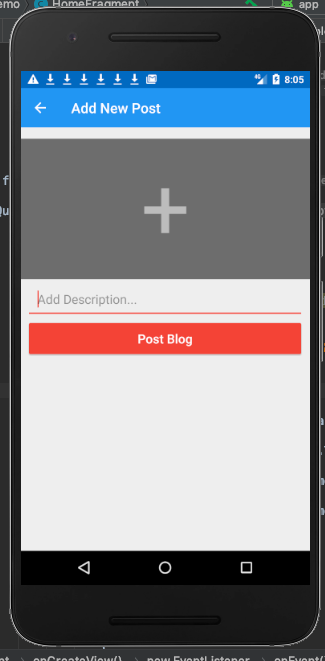
* Các item trong RecyclerView được design như bên dưới.



1. **Post Activity**

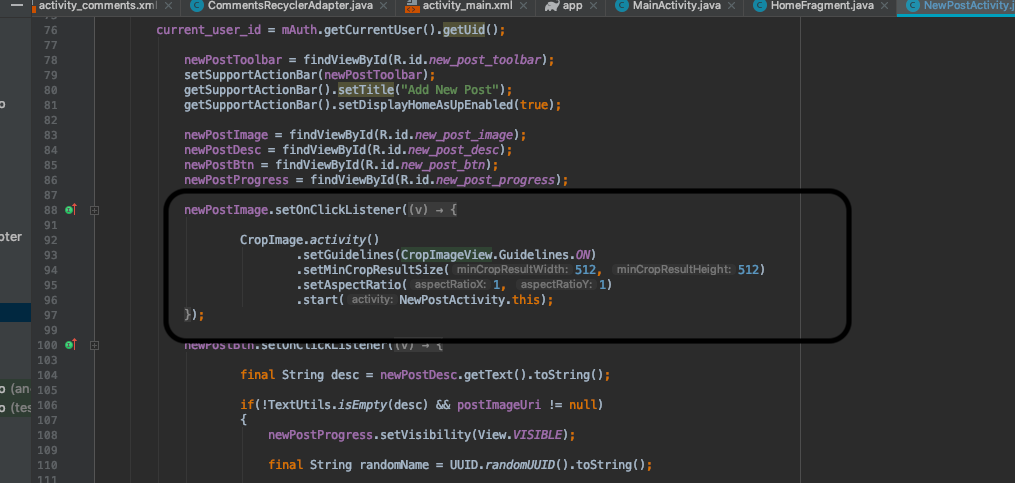
* Khi nhấn vào Button post bài sẽ hiện lên một activity (NewPostActivity)



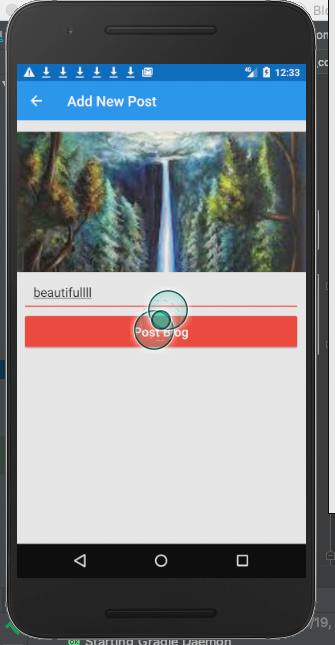


1. **Cắt Ảnh**

* Ở phần thêm ảnh trong NewPostActivity, cũng sẽ sử dụng thư viện ‘com.theartofdev.edmodo:android-image-cropper:2.6.0’ khi nhấn vào thêm ảnh giống ở mục 2.2.1

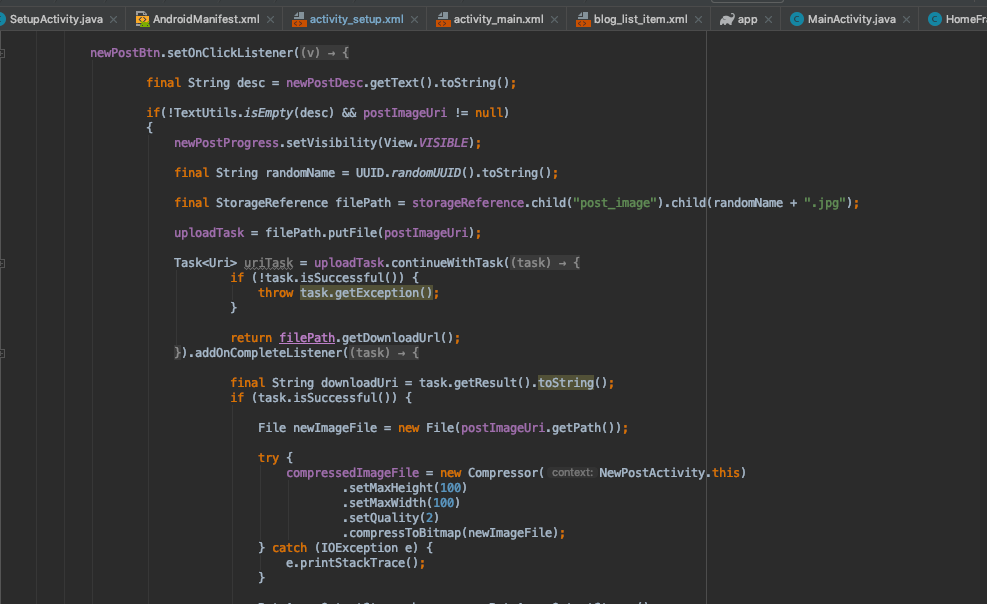


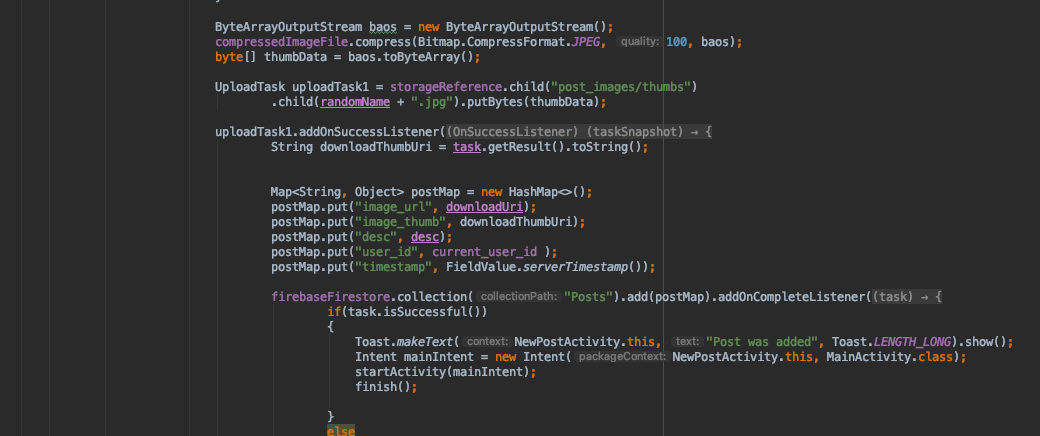


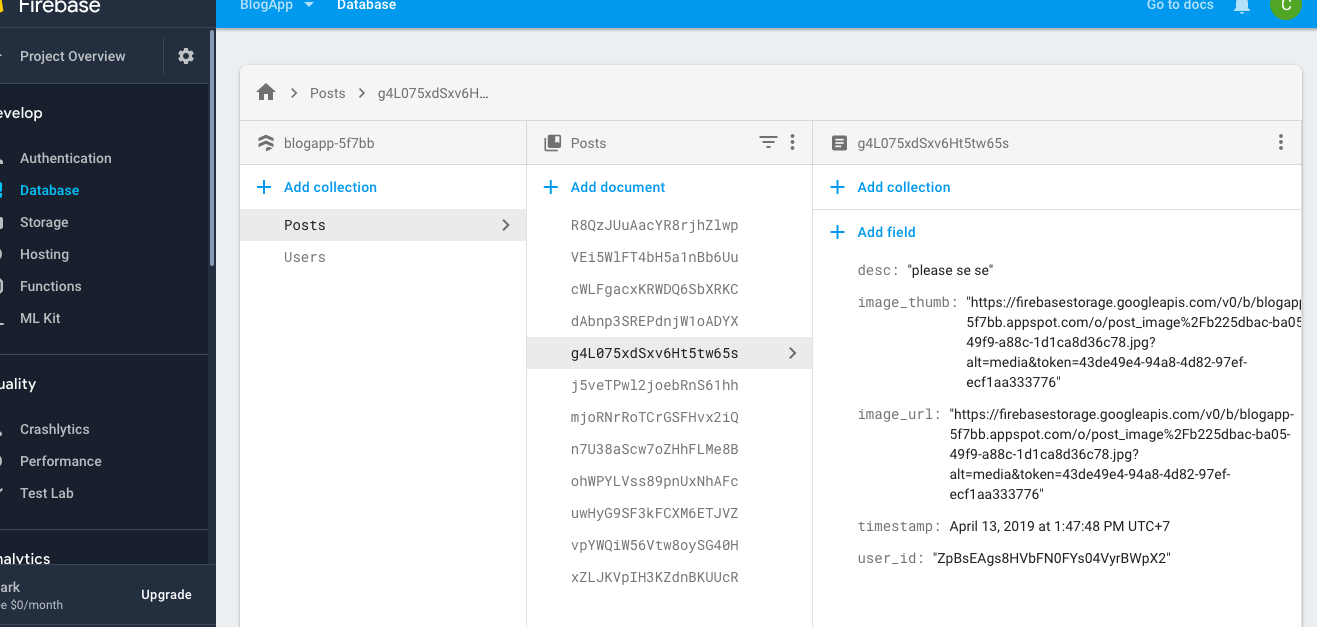


1. **Update dữ liệu bài viết lên Firebase**

* Khi nút Post Blog được nhấn thì các dữ liệu trong bài viết sẽ được update trên Firebase. Database sẽ lưu thông tin cái bài viết ở Collection(“Posts”)

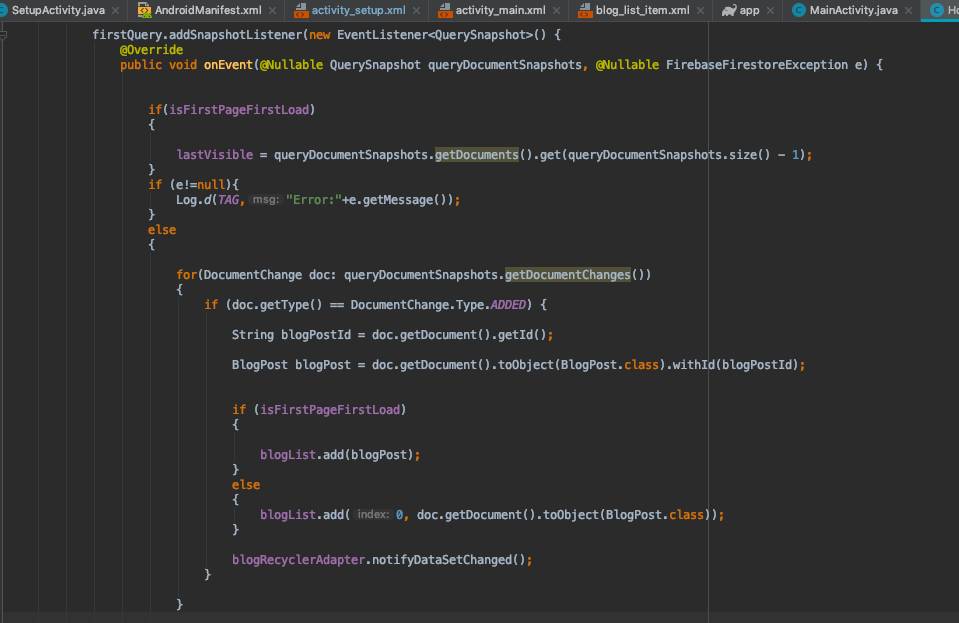






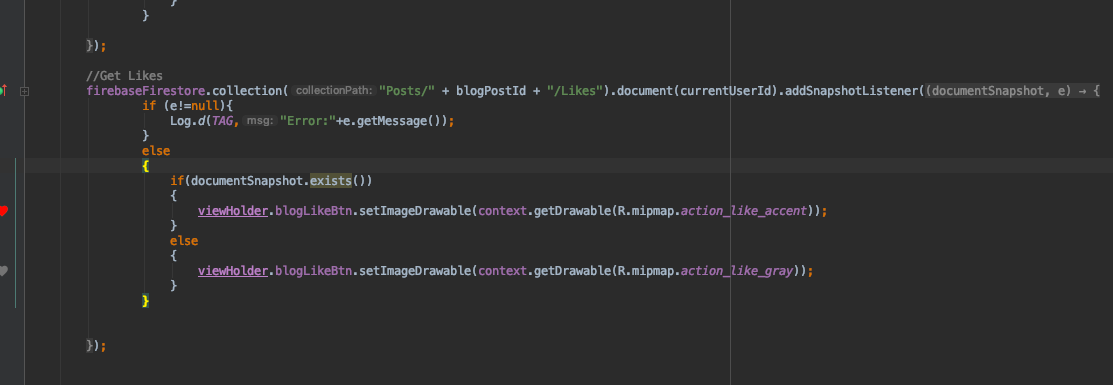
1. **Đổ dữ liệu về HomeActivity**

* Khi Document(“Posts”) có sự thay đổi thì dữ liệu mới sẽ được tự động cập nhật ở HomeFragment

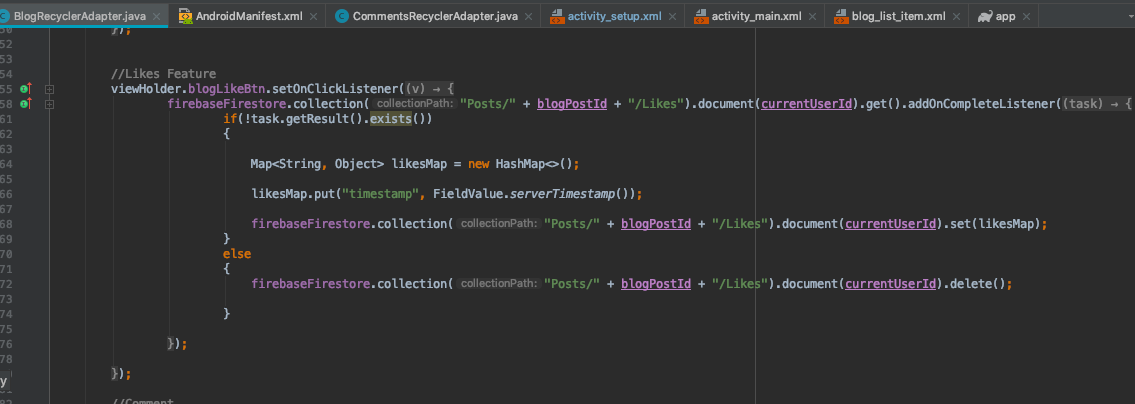


1. **Chức năng like, comment**
2. **Chức năng Like**

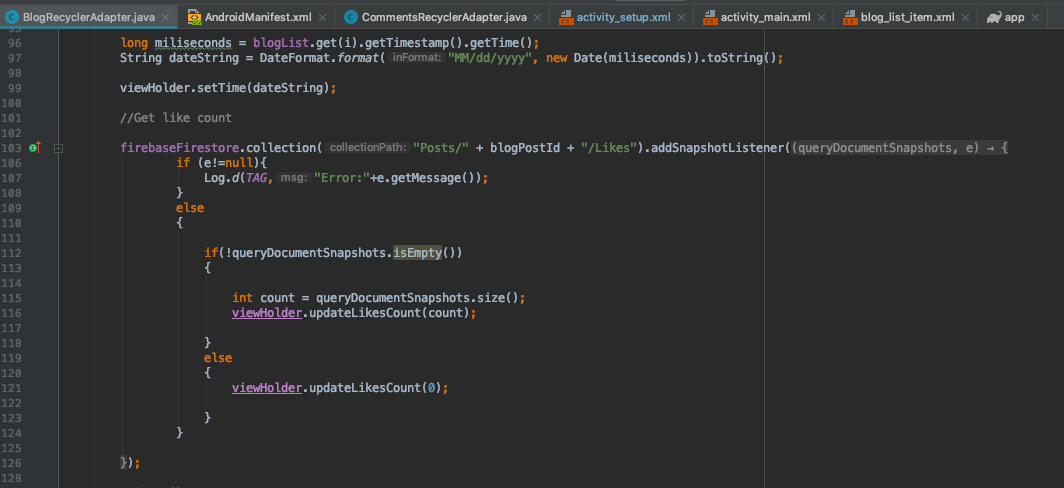
* Khi nhấn like thì nút Like sẽ đỏ lên.



* Thông tin về tên tài khoản like và thời gian like sẽ được update lên Firebase và số lượng tài khoản like sẻ thêm 1 dòng.



* Sau khi Like, số lượng like được đếm từ firebase sẽ được lưu ở biến count và sau đó sẽ update vào Textview bên cạnh biểu tượng like

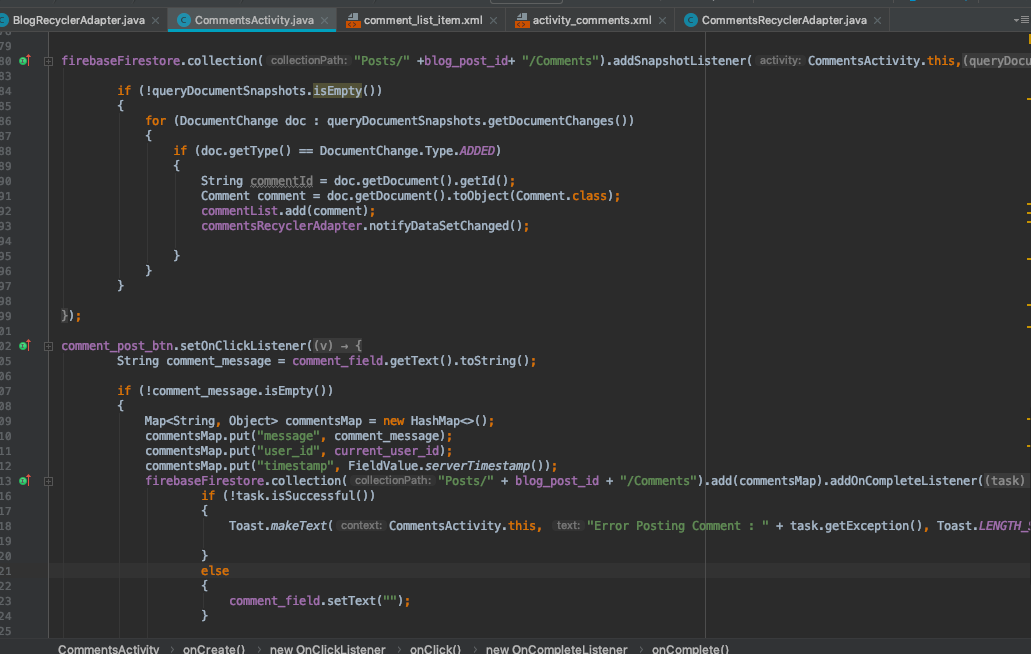


1. **Chức năng Comment**

* Khi nhấn vào biểu tượng comment, thì sẽ hiện lên một CommentActivity. Activity này chứa danh sách các tài khoản đã comment vào phần để comment vào bài viết



* Khi nhập 1 nội dung và nhấn nút Comment thì trên Firebase sẽ cập nhật ở Collection(“Comments”) thêm 1 dòng chứa nội dung gồm: ID email comment, nội dung comment và thời gian comment.





* Sau khi comment, lượt comment sẽ được cập nhật tương tự như số lượng like phía trên

