



**FPT POLYTECHNIC**



**android**

**LẬP TRÌNH ANDROID VỚI REST API**

**FIREBASE**

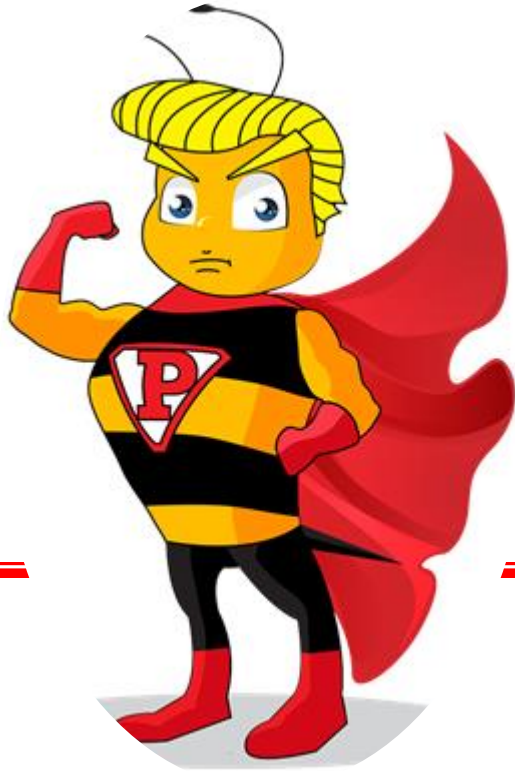
[www.poly.edu.vn](http://www.poly.edu.vn)

- ☐ Tổng quan về Firebase
- ☐ Tạo dự án trên Firebase
- ☐ Authentication

# MỤC TIÊU

- ◎ TỔNG QUAN VỀ FIREBASE
- ◎ TẠO DỰ ÁN TRÊN FIREBASE
- ◎ AUTHENTICATION





---

FIREBASE

---

...

- ❑ **Firestore** là một nền tảng giúp phát triển các ứng dụng di động. Bên cạnh đó, Firestore còn được xem là một dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây cloud với hệ thống máy chủ mạnh mẽ của Google.
- ❑ **Firestore** chứa cơ sở dữ liệu mang đến khả năng code nhanh và thuận tiện hơn. Lập trình viên có thể dễ dàng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu sẵn có.




- ❑ Vào năm 2011, James Tamplin và Andrew Lee đã cho ra mắt **Evolve**. Đây là một nền tảng có cấu trúc khá đơn giản chuyên cung cấp các API cần thiết để tích hợp tính năng trò chuyện vào các trang web. Tuy nhiên, họ nhận ra rằng nền tảng này đang được sử dụng để truyền dữ liệu ứng dụng chứ không đơn giản là chat. Sau đó, họ đã phát triển **Envole** và tạo nên **Firestore**.
- ❑ Đến tháng 4 năm 2012, Firestore đã được công bố trên toàn cầu dưới dạng một công ty riêng biệt. Những năm sau đó, **Firestore** đã thực hiện nhiều cuộc huy động vốn và ra mắt các sản phẩm mới.
- ❑ Đến tháng 10 năm 2014, **Firestore** đã chính thức được **Google** mua lại và trở thành một ứng dụng đa năng trên nền tảng di động và website.

## ❑ Realtime Database

- ❑ Realtime Database là một cơ sở dữ liệu thời gian thực được lưu trữ dưới dạng JSON và được đồng bộ hóa theo thời gian thực đối với mọi kết nối.
- ❑ Đối với các ứng dụng được xây dựng trên đa nền tảng như Android, iOS và WebApp, tất cả client sẽ cùng sử dụng một cơ sở dữ liệu. Bên cạnh đó, hệ thống dữ liệu này sẽ tự động cập nhật khi lập trình viên phát triển ứng dụng. Sau đó, tất cả dữ liệu này sẽ được truyền tải thông qua các kết nối SSL có 2048 bit.



## Authentication

-  Authentication là tính năng giúp xác thực danh tính của người dùng ứng dụng. Firebase cung cấp các bước xác thực thông qua Email, Facebook, Twitter, GitHub hay Google. Điều này giúp cho các thông tin cá nhân của khách hàng được bảo vệ một cách tốt nhất, hạn chế được tình trạng bị hacker đánh cắp. Đồng thời việc xác thực danh tính qua Firebase sẽ giúp người dùng tiếp cận sản phẩm nhanh chóng và an toàn hơn.






## ❑ Cloud Storage

- ❑ Cloud Storage là tính năng cho phép lưu trữ và quản lý nội dung đã tạo ra như ảnh, video, nội dung, văn bản,... Firebase Storage cung cấp các API hỗ trợ bạn upload và download các file từ ứng dụng một cách trơn tru mà không cần quan tâm đến chất lượng đường truyền mạng với độ bảo mật cao.



**Cloud Storage**  
for Firebase

## **Hosting**

-  Hosting được phân phối thông qua tiêu chuẩn công nghệ bảo mật SSL từ hệ thống mạng CDN. CDN là một mạng lưới máy chủ giúp lưu trữ các bản sao của các nội dung tĩnh trên website. Thông qua CDN, người dùng có thể truy cập và sử dụng các dịch vụ trên web khi cài Firebase Hosting một cách nhanh chóng và ổn định hơn.



## □ Analytics

- Analytics giúp bạn có thể phân tích hành vi của người sử dụng ứng dụng của bạn. Qua đó, bạn sẽ biết được khách hàng thường xuyên truy cập tính năng nào và các thông tin về hiệu quả quảng cáo, tình trạng trả phí,... để có thể đưa ra được chiến lược phát triển phù hợp. Để thực hiện tính năng Analytics của Firebase, bạn cần cài đặt Software Development Kit (SDK).



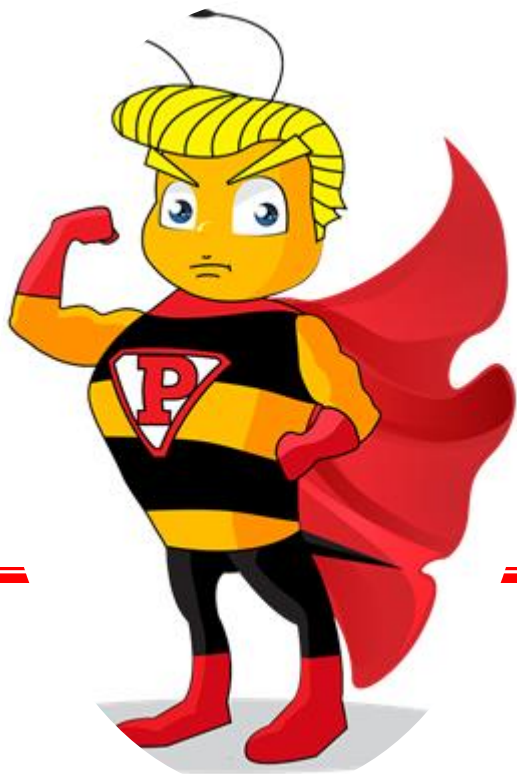
**Google Analytics**  
for Firebase

## □ Ưu điểm

- Sử dụng miễn phí và thuận tiện
- Dễ sử dụng và tích hợp
- Đáp ứng nhu cầu của người dùng
- Cập nhật liên tục và đa nền tảng

## □ Nhược điểm

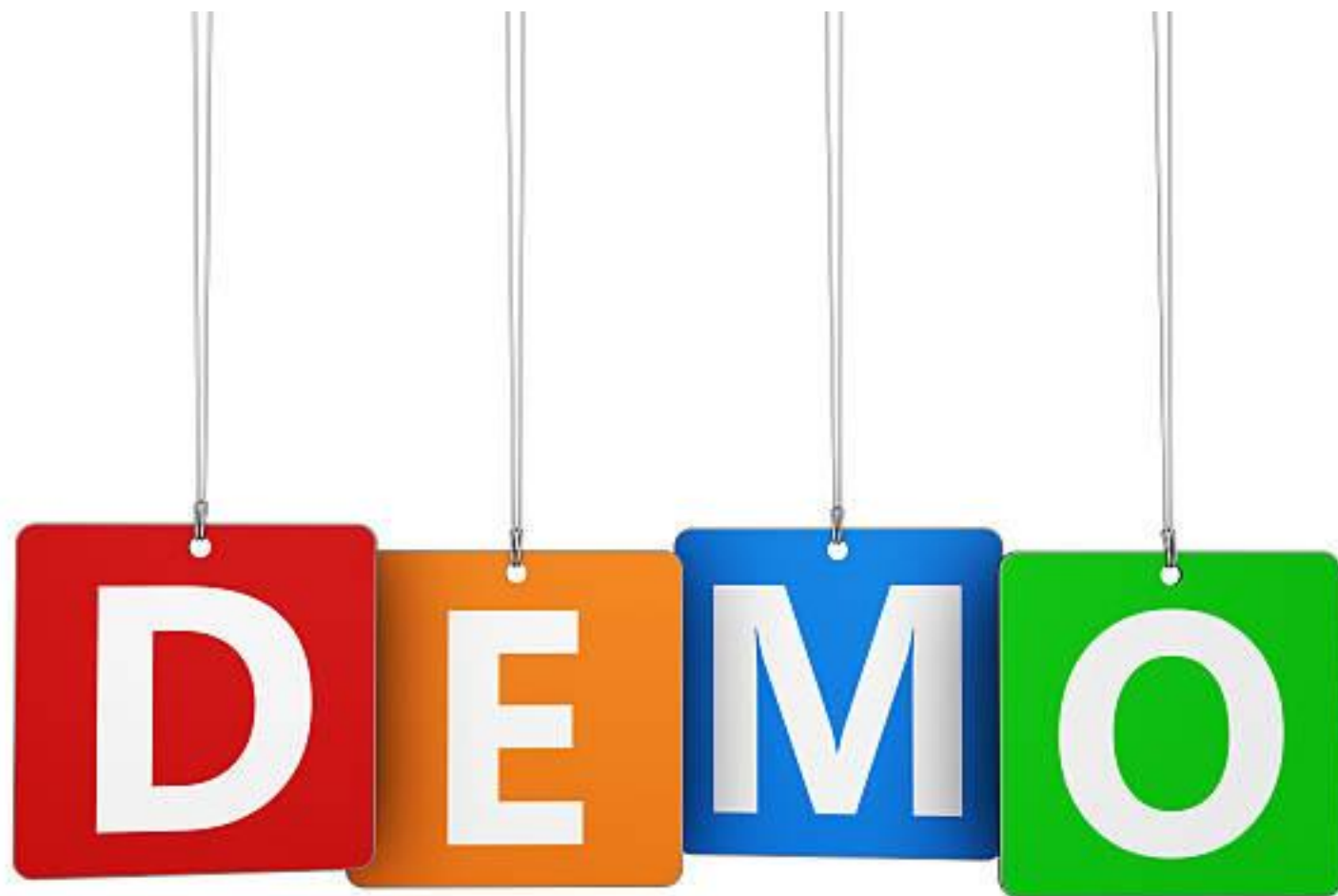
- ❖ Giới hạn về quy mô ứng dụng
- ❖ Khả năng tùy chỉnh hạn chế
- ❖ Giá các dịch vụ của Firebase

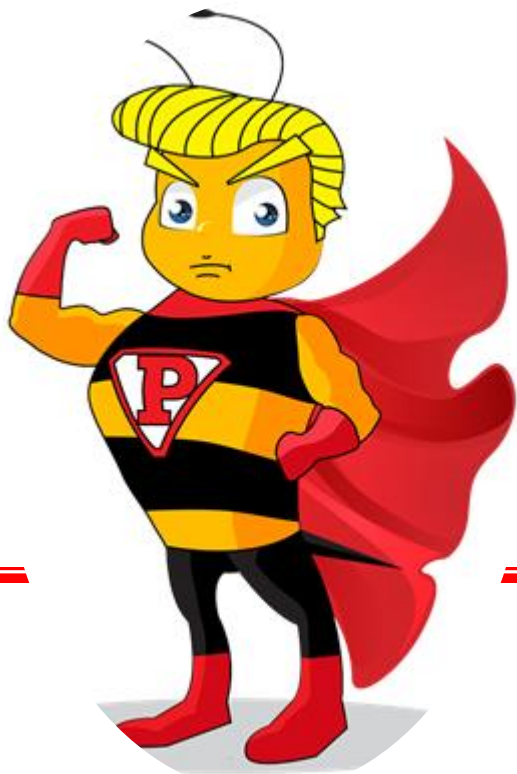


# TẠO DỰ ÁN TRÊN FIREBASE

---

...





# AUTHENTICATION TRÊN FIREBASE

...



---

XÁC THỰC BẰNG EMAIL/PASSWORD

---

...



## ❑ Đăng ký tài khoản bằng email/password

```
mAuth.createUserWithEmailAndPassword(email, password).addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {  
    @Override  
    public void onComplete(@NonNull Task<AuthResult> task) {  
        if (task.isSuccessful()) {  
            //lấy thông tin tài khoản mới vừa đăng ký  
            FirebaseUser user = mAuth.getCurrentUser();  
            Toast.makeText( context: SignUpActivity.this, text: "Đăng ký thành công", Toast.LENGTH_SHORT).show();  
            finish();  
        } else {  
            Log.w(TAG, msg: "createUserWithEmail:failure", task.getException());  
            Toast.makeText( context: SignUpActivity.this, text: "Đăng ký thật bại", Toast.LENGTH_SHORT).show();  
        }  
    }  
});
```

## ❑ Đăng nhập tài khoản bằng email/password

```
mAuth.signInWithEmailAndPassword(email, password).addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {  
    @Override  
    public void onComplete(@NonNull Task<AuthResult> task) {  
        if (task.isSuccessful()) {  
            //lấy thông tin tài khoản mới vừa đăng nhập  
            FirebaseUser user = mAuth.getCurrentUser();  
            Toast.makeText( context: LoginActivity.this, text: "Đăng nhập thành công", Toast.LENGTH_SHORT).show();  
            startActivity(new Intent( packageContext: LoginActivity.this, LogoutActivity.class));  
        } else {  
            // If sign in fails, display a message to the user.  
            Log.w(TAG, msg: "signInWithEmail:failure", task.getException());  
            Toast.makeText( context: LoginActivity.this, text: "Đăng nhập thất bại", Toast.LENGTH_SHORT).show();  
        }  
    }  
});
```

❑ Chức năng Đăng xuất

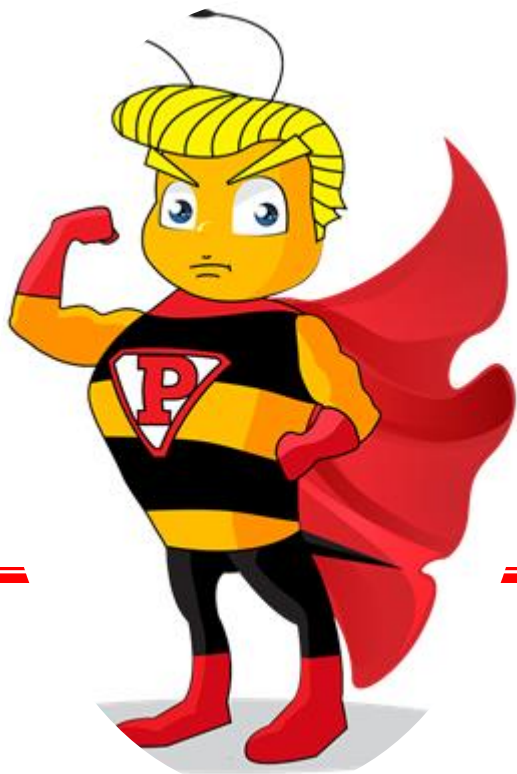
```
firebaseAuth.getInstance().signOut();
```



## ❑ Chức năng quên mật khẩu

```
auth.sendPasswordResetEmail(emailAddress).addOnCompleteListener(new OnCompleteListener<Void>() {  
    @Override  
    public void onComplete(@NonNull Task<Void> task) {  
        if (task.isSuccessful()) {  
            Toast.makeText(context: LoginActivity.this, text: "Vui lòng kiểm tra hộp thư để cập nhật lại mật khẩu", Toast.LENGTH_SHORT).show();  
        } else {  
            Toast.makeText(context: LoginActivity.this, text: "Lỗi gửi mail", Toast.LENGTH_SHORT).show();  
        }  
    }  
});
```





---

XÁC THỰC BẰNG SỐ ĐIỆN THOẠI

---

## ❑ Yêu cầu gửi OTP về số điện thoại

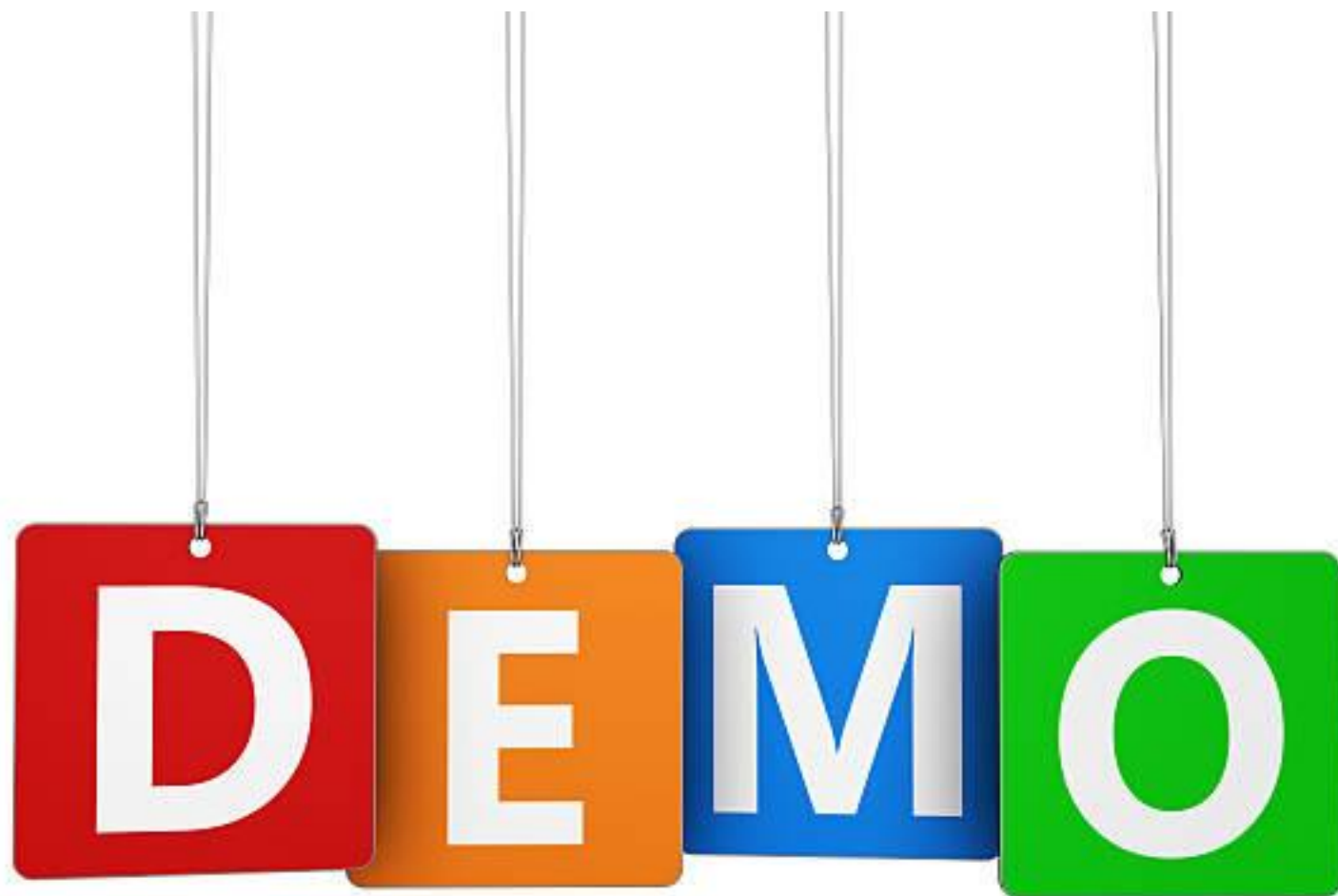
```
private void getOTP(String phoneNumber) {  
    PhoneAuthOptions options =  
        PhoneAuthOptions.newBuilder(mAuth)  
            .setPhoneNumber("+84"+phoneNumber) // Phone number to verify  
            .setTimeout(timeout: 60L, TimeUnit.SECONDS) // Timeout and unit  
            .setActivity(this) // (optional) Activity for callback binding  
            // If no activity is passed, reCAPTCHA verification can not be used.  
            .setCallbacks(mCallbacks) // OnVerificationStateChangedCallbacks  
            .build();  
    PhoneAuthProvider.verifyPhoneNumber(options);  
}
```

## ❑ Xác thực số điện thoại

```
private void verifyOTP(String code)
{
    PhoneAuthCredential credential = PhoneAuthProvider.getCredential(mVerificationId, code);
    signInWithPhoneAuthCredential(credential);
}

1 usage
private void signInWithPhoneAuthCredential(PhoneAuthCredential credential) {
    mAuth.signInWithCredential(credential)
        .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if (task.isSuccessful()) {
                    // Sign in success, update UI with the signed-in user's information
                    Toast.makeText( context: LoginWithPhoneActivity.this, text: "Đăng nhập thành công", Toast.LENGTH_SHORT).show();
                    FirebaseUser user = task.getResult().getUser();
                    startActivity(new Intent( packageContext: LoginWithPhoneActivity.this, LogoutActivity.class));
                    // Update UI
                } else {
                    // Sign in failed, display a message and update the UI
                    Log.w(TAG, msg: "signInWithCredential:failure", task.getException());
                    if (task.getException() instanceof FirebaseAuthInvalidCredentialsException) {
                        // The verification code entered was invalid
                    }
                }
            }
        });
}
```







FPT POLYTECHNIC

**Thank you**