**Báo cáo**

1. **Activities**

Hiển thị màn hình những gì mà người dùng làm

Vòng đời của một activity :



Khi một activity được khởi tạo thì ngay sau đó hàm onStart() và onResume() được gọi đến

và khi một activity đang running, giả sử có một cuộc gọi điện thoại đến làm che đi toàn bộ activity thì hàm onPause() và hàm onStop() sẽ được gọi. Nếu như sau đó bạn vẫn muốn tiếp tục với activity lúc trước thì hàm onRestart() được gọi đến sau đó lại gọi tiếp đến hàm onStart() và hàm onResume(). Còn nếu không hàm onDestroy() được gọi đến và vòng đời của activity sẽ dừng lại ở đây

1. **Service**

Service là một thành phần mà nó được chạy ngầm để thực hiện các hoạt động lâu dài mà không cần tương tác với người dùng, nó hoạt động ngay cả khi ứng dụng bị tắt

Có hai trạng thái của service là

* Started : ở trạng thái này khi một activity bắt đầu với hàm startService(). Chỉ cần bắt đầu một lần service có thể chạy ngầm thậm chí khi ứng dụng tắt thì nó vẫn hoạt động
* Bound : ở trạng thái này service phải chạy dựa vào giao diện, buộc phải cung cấp giao diện client-server cho phép các thành phần tương tác với dịch vụ , gửi yêu cầu....

1. **Broadcast** **Receive**

Phản hồi các thông điệp từ ứng dụng khác hay từ chính hệ thống ,Những thông điệp này có thể được gọi là các events hoặc intent, ví dụ như phản hồi khi thiết bị cắm sạc, nhận wifi, pin đã sạc đầy hoặc chuẩn bị hết pin,...

1. **Fragment**

Là một phần của Activity, thường được sử dụng với những Activity có nhiều layout, thay vì mình phải tạo activity khác

Vòng đời của một fragment



* Fragment cũng có layout của riêng của nó, cũng có các hành vi và vòng đời riêng.
* Chúng ta có thể thêm hoặc xóa Fragment trong một Activity trong khi Activity này đang chạy.
* Có thể kết hợp nhiều Fragment trong một Activity để xây dựng giao diện người dùng đa khung.
* Một Fragment có thể được sử dụng trong nhiều Activitiy.
* Vòng đời của Fragment phụ thuộc vào vòng đời của acticity, khi activity tạm dừng thì các fragment cũng dừng lại theo.
* Fragment có thể thực hiện một hành vi mà không có trong thành phần giao diện người dùng.
* Fragment được thêm vào API 11 trở lên

1. **Content** **Provider**

Được dùng để cũng cấp dữ liệu từ một ứng dụng tới các ứng dụng khác theo yêu cầu. Những yêu cầu này được xử lý bởi các phương thức của lớp ContentResolver. Một Content Provider có thể sử dụng các cách khác nhau để lưu giữ dữ liệu và dữ liệu có thể được lưu giữ trong một cơ sở dữ liệu, một file, hoặc có thể thông qua một mạng

Content Provider được sử dụng tương đối giống với database, bạn có thể truy vấn lấy dữ liệu, sửa đổi nội dung hay thêm xóa sửa, Dữ liệu này được lưu trong Sqlite database

1. **Intent** – **filter**

Là một mô tả trừu tượng của một hoạt động để được thực hiện. Nó có thể được sử dụng với startActivity để chạy một activity, broadcastIntent gửi nó tới bất kì thành phần BroadcastReceiver nào quan tâm đến và với startService(Intent) hoặc blindService() có thể giao tiếp với service ở background