

---

# **BÁO CÁO THỰC HÀNH**

**Môn học: Phát triển ứng dụng trên thiết bị di động**

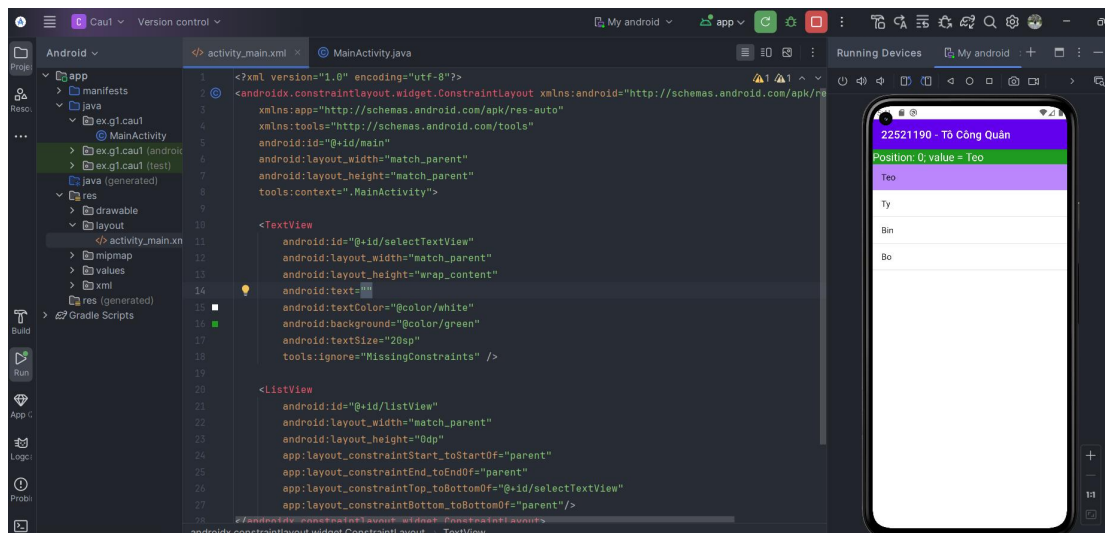
**Bài báo cáo: Lab02**

**Tên chủ đề: LISTVIEW và RECYCLERVIER**

**Lớp: NT118.P13.2**

# BÁO CÁO CHI TIẾT

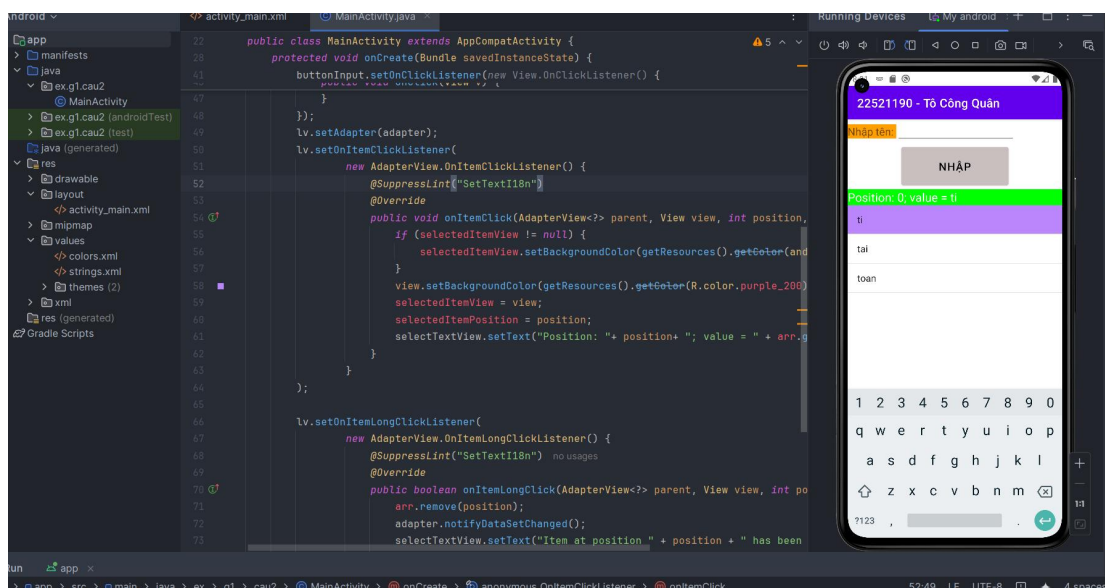
## Câu 1: Sử dụng ListView control với mảng dữ liệu định sẵn



Giao diện có 2 control, ListView và TextView. ListView sẽ hiện ra một danh sách tên từ một mảng có sẵn. Khi nhấn vào một item trên ListView thì TextView sẽ hiện ra vị trí và giá trị của item đang được chọn. Vị trí được bắt đầu từ 0. Khi một item được chọn thì background cũng sẽ được tô màu để dễ nhận biết

Demo: [tại đây](#)

## Câu 2: Sử dụng ArrayList và Listview control

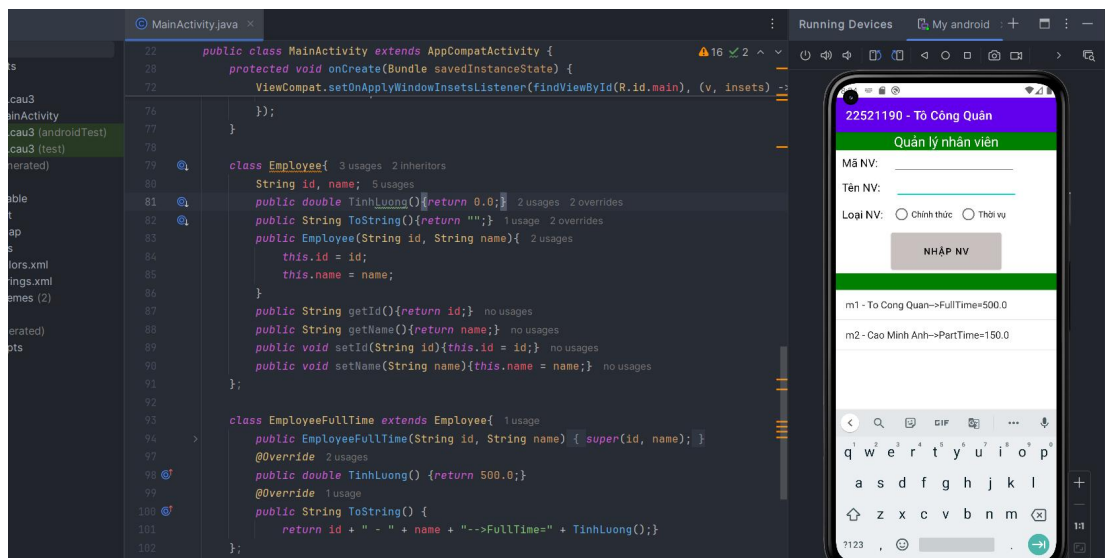


Giao diện tương tự như câu 1, tuy nhiên có thêm 3 view là TextView: Nhập tên, EditText để nhập dữ liệu tên và Button để nhập. Khi điền tên và nhập thì sẽ

được thêm vào danh sách. Khi chọn vào một item (tên), thì background sẽ đổi màu để biết tên mình đang chọn và TextView bên trên sẽ cho ta biết vị trí và giá trị của nó. Thêm nữa, khi giữ lâu vào tên thì tên đó sẽ bị xóa khỏi danh sách.

Demo: [tại đây](#)

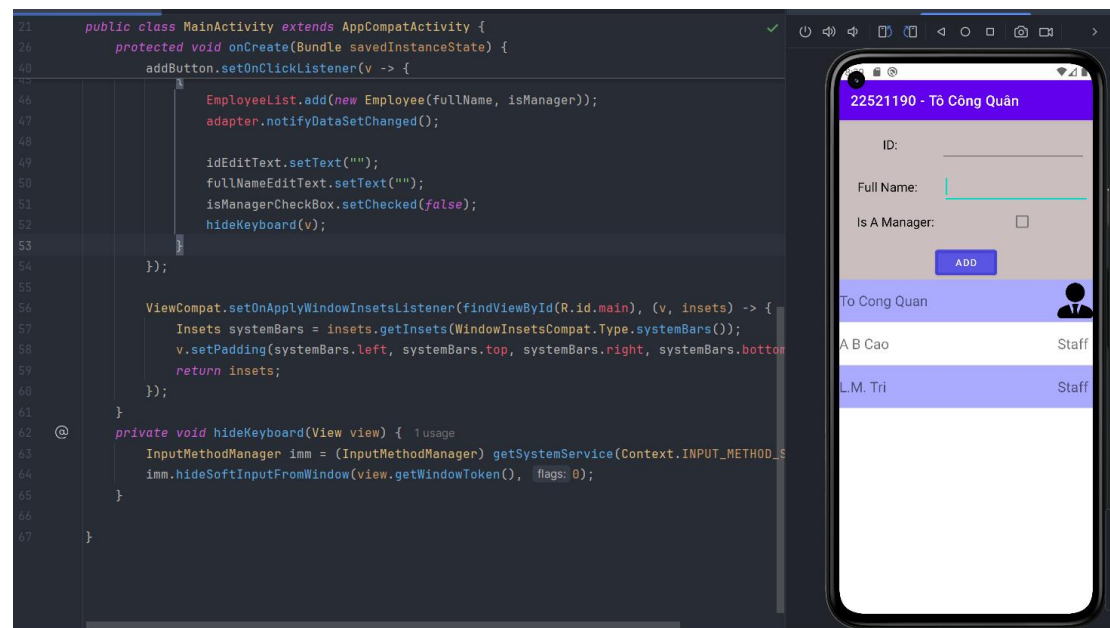
### Câu 3: Sử dụng ArrayList và ListView mà từng phần tử trong ArrayList là các Object bất kỳ



Giao diện sẽ bao gồm nhiều view. Sẽ có hai EditText để nhập dữ liệu vào: MÃ NV và Tên NV. Ngoài ra, còn có 2 radio button. Khi điền đủ 2 trường thông tin và chọn Loại NV, nhấn nút Nhập NV thì nhân viên đó sẽ được thêm vào danh sách và hiện mã, loại nhân viên và lương của họ. Nếu như là nhân viên chính thức thì lương là 500.0, còn nếu là thời vụ thì sẽ là 150.0.

Demo: [tại đây](#)

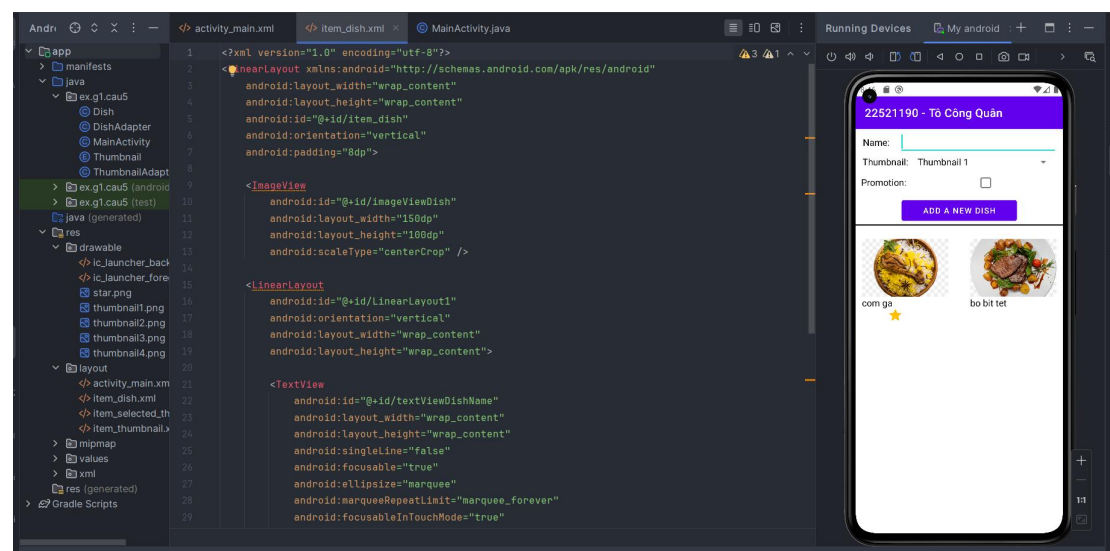
## Câu 4: Sử dụng CustomAdapter cho Listview



Giao diện sẽ có TextView, EditText, Button, Listview và ImageView. Khi nhập 2 thông tin về ID và tên, sau đó nhấn nút Add thì thông tin sẽ thêm vào danh sách bên dưới. Nếu như checkbox Is A Manager được click thì ImageView sẽ được hiện, hiểu là đây là Manager, còn nếu như không được click thì thay vì là ImageView sẽ là TextView “Staff”, được hiểu là nhân viên. Ngoài ra, từng item trong danh sách sẽ được tô màu xen kẽ nhau để dễ nhìn hơn.

Demo: [tại đây](#)

## Câu 5: Sử dụng GridView, Spinner



---

Giao diện sẽ được kết hợp nhiều view, lần này sẽ sử dụng GridView thay vì ListView. Ban đầu cần nhập tên món ăn, trong menu thumbnails sẽ có hình của các món ăn. Checkbox Promotion sẽ được hiểu là món ăn đó có khuyến mãi hay không. Khi nhấn nút Add thì món ăn sẽ được thêm vào GridView gồm hình, tên và một hình ngôi sao để thể hiện là được khuyến mãi, nếu checkbox không được click thì sẽ không có hình ngôi sao. GridView sẽ chia thành hai cột để hiển thị món ăn.

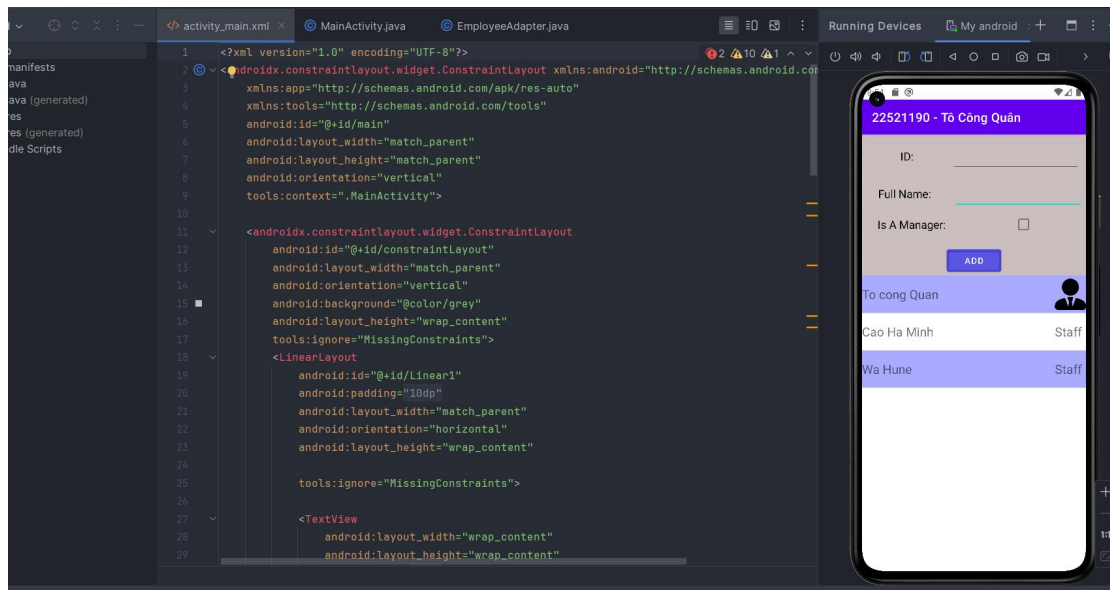
**Demo:** [tại đây](#)

## **Câu 6: Sử dụng CustomAdapter cho ListView) bằng RecyclerView**

RecyclerView là một ViewGroup mới được giới thiệu trong Android L (Android API 21). Đây là một ViewGroup có chức năng tương tự ListView nhưng nó tỏ ra mạnh mẽ, linh hoạt hơn rất nhiều. ListView chỉ hỗ trợ bạn scroll các item trong ListView theo chiều dọc (vertical) mà không hỗ trợ scroll theo chiều ngang (horizontal) . RecyclerView support được tất cả những thứ đó và hơn thế nữa.

### ***Sự khác nhau giữa ListView và RecyclerView***

- Yêu cầu sử dụng ViewHolder Pattern trong Adapter: Với ListView chúng ta không cần sử dụng ViewHolder pattern để cải thiện performance của ListView nhưng ngược lại khi chúng ta tạo một Adapter sử dụng với RecyclerView bắt buộc phải sử dụng ViewHolder để cải thiện performance. Mục đích sử dụng ViewHolder để tái sử dụng View nhằm tránh việc tạo View mới và findViewById quá nhiều
- ListView chỉ support cho chúng ta danh sách dạng scroll dọc. Nhưng với RecyclerView cung cấp cho chúng ta RecyclerView.LayoutManager cho phép Layout các item trong listView theo các kiểu khác nhau (ngang, dọc, dạng grid, dạng staggered grid).
- Với ListView việc xử lý animation cho các item trong ListView không hề dễ dàng. Nhưng đối với RecyclerView có hỗ trợ ItemAnimator giúp chúng ta có thể xử lý animation khi add hay remove một item ra khỏi RecyclerView một cách dễ dàng. Mặc định RecyclerView sử dụng DefaultItemAnimator.
- Với ListView việc sử dụng divider không được linh hoạt nhưng với RecyclerView có hỗ trợ ItemDecoration cho phép chúng ta draw divider một cách tùy thích.
- ListView có support các phương thức phương thức.setOnItemClickListener và setOnLongItemClickListener để chọn 1 item trong ListView. Nhưng RecyclerView chỉ support một phương thức đó là onItemClick.



Giao diện giống như câu 4, nhưng thay vì sử dụng ListView thì sẽ sử dụng RecyclerView.

**Demo:** [tại đây](#)

---