**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO MÔN ANDROID**

**XÂY DỰNG ỨNG DỤNG DỰ BÁO THỜI TIẾT**

**Nguyễn Vân Trường An**

**Đỗ Quang Ân**

**Thành phố Hồ Chí Minh – 5/2020**

1. Tổng quan
2. Đề tài: Xây dựng ứng dụng android dự báo thời tiết
3. Phạm vi đề tài

* Xây dựng ứng dụng với các chức năng cơ bản phụ vụ việc demo cho yêu cầu của đề tài
* Mục đính chính: học tập và tham khảo

1. Yêu cầu
2. Về cơ sở lý thuyết

* Nghiên cứu các thư viện cung cấp dữ liệ thời tiết, từ đó lựa chọn thư viện phù hợp
* Tìm hiểu cách đọc dữ liệu và tích hợp vào app từ thư viện đã chọn
* Thiết kê giao diện phù hợp

1. Về chức năng

* Tìm, gợi ý tên thành phố
* Thêm thành phố muốn xem thời tiết
* Xóa thành phố, phục hồi nếu cần
* Xem dự báo thời tiết trong 7 ngày tới

1. Về kỹ thuật

* Ngôn ngữ lập trình **Kotlin**.
* Nền tảng **Android**.
* Cơ sở dữu liệu: **Room Database và SharePreference**
* Xây dựng theo mô hình **MVP (Model-View-Presenter)**

1. Chọn công nghệ - công cụ phát triển

* Công cụ phát triển Android (**Android** **Studio**)
* Kotlin plugins.
* Thư viện chính - cung cấp dữ liệu thời tiết (**OpenWeatherMap** **API**)
* Thư viện HTTP Client (**Retrofit**)
* Các thư viện khác (**Material, Glide, Kprogresshud…)**

1. Cơ sở lý thuyết (Sơ lược về OpenWeatherMap API):

* OpenWeatherMap là một trong những nhà cung cấp thông tin thời hàng đầu, được thành lập vào năm 2014 bởi một nhóm các kỹ sư và chuyên gia về Big Data, xử lý dữ liệu và xử lý hình ảnh vệ tinh
* OpenWeatherMap bao gồm cả gói miễn và có các gói trả phí dành cho các nhà phát triển yêu cầu hỗ trợ và dữ liệu thời tiết cao hơn
* Sản phẩm:
  + **Current & Forecast collection**
* [Current weather for any geolocation](https://openweathermap.org/current)
* [4-days/hourly weather forecast](https://openweathermap.org/api/hourly-forecast)
* [5-days/3 hour weather forecast](https://openweathermap.org/forecast5)
* [16-days/daily weather forecast](https://openweathermap.org/forecast16)
* [Climate forecast for 30 days](https://openweathermap.org/api/forecast30)
* [One Call API](https://openweathermap.org/api/one-call-api)
* [Bulk downloading](https://openweathermap.org/bulk)

#### Historical collection

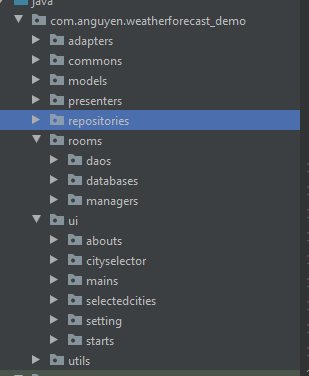
* [Historical weather API](https://openweathermap.org/history)
* [History Bulk](https://openweathermap.org/history-bulk)
* [Historical Forecast Bulk](https://openweathermap.org/api/history-forecast-bulk)
* [Statistical weather data API](https://openweathermap.org/api/statistics-api)

#### Maps and other weather API's collection

* [Weather maps 2.0: Forecast, Historical, Current](https://openweathermap.org/api/weather-map-2)
* [Weather maps 1.0](https://openweathermap.org/api/weathermaps)
* [Relief maps](https://openweathermap.org/api/relief)
* [Weather alerts](https://openweathermap.org/triggers)
* [Network of weather stations](https://openweathermap.org/stations)
* [UV index](https://openweathermap.org/api/uvi)

1. Kết quả xây dựng:
2. Câu trúc tổng thể của project:

* Xây dựng dựa trên mô hình MVP (Model – Presenter – View)

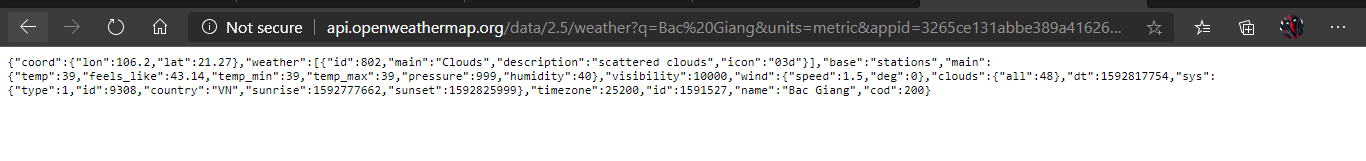


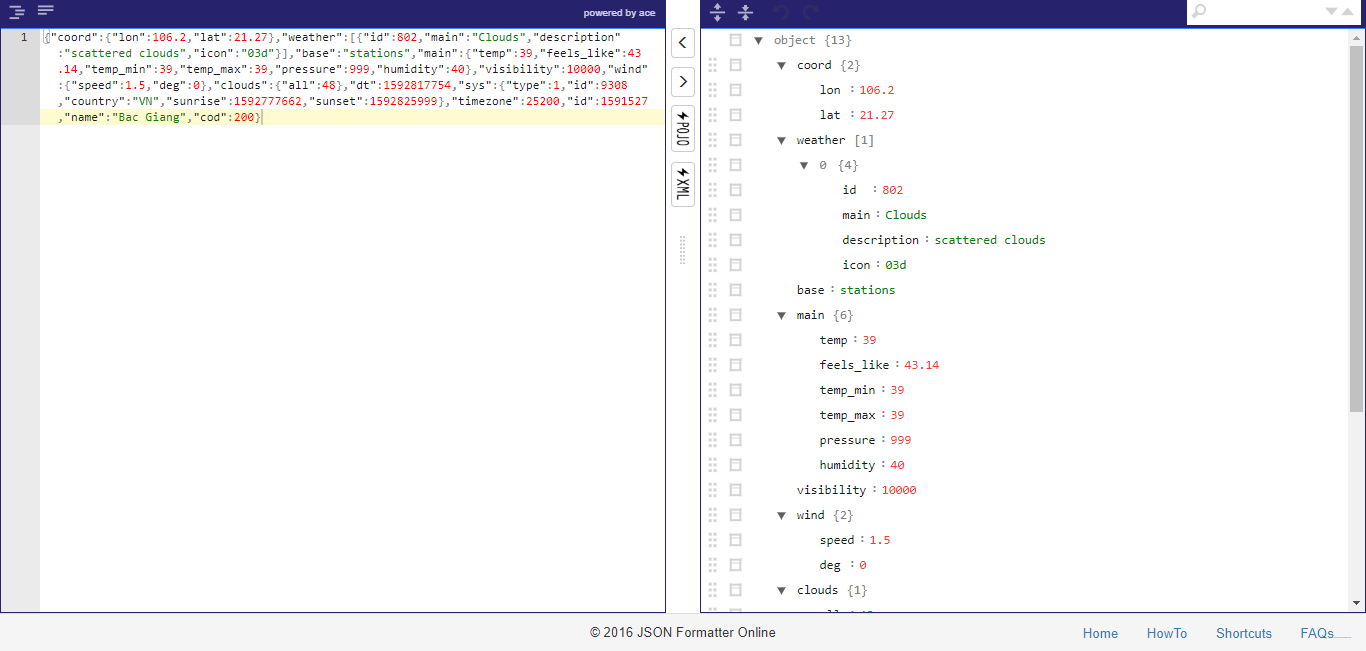
* Chú thích:
* Adapters: Chứa các adapter class của RecycleView hoặc ViewPager
* Commons: Các const value, phương thức, class hỗ trợ được dùng chung
* Models: Chứa các data class
* Presenter: Chứa các class phục phụ việc giao tiếp giữa giao diện và cơ sở dữ liệu
* Repositories: Chứa các class và interface thực hiện các phương thức đọc Api thông qua Retrofit và Api client được build trong Utils
* Rooms: bao gồm daos chứa interface đọc, ghi cơ sở dữ liệu; databases chứa các class khởi tạo cơ sở dữ liệu; manager chứa class thực hiện các phương thức nhận yêu cầu đọc hoặc ghi cơ sơ dữ liệu từ presenter thông qua daos
* Uis: Chứa các thư mục giao diện, một thư mục là một Activity hoặc Fragment và một View interface tương ứng
* Utils: Chứa các class build service Retrofit, đọc Api’s Url, interface giao tiếp với Repositories
* Khi người dùng thao tác trên Giao diện người dùng, Activiy hoặc Fragment gửi yêu cầu lấy dữ liệu thông qua Presenter
* Presenter lấy dữ liệu từ Manger (từ Room Database lấy dữ liệu cục bộ) hoặc WeatherReposity (đọc Api thông qua Retrofit lấy dữ liệu thời tiết). Sau đó trả kết quả theo Model tương ứng về cho View
* Nếu lấy dữ liệu thất bại, View chuyển đến phương thức thông báo thất bại
* Nếu lấy dữ liệu thành công chuyển đến phương thức thành công có trả về dữ liệu vừa lấy
* Từ kết quả trả về của View, Activiy hoặc Fragment (có implent interface View tương ứng) chuyển cho Adapter nếu cần và hiển thị lên Giao diện người dùng

1. Cơ sở dữ liệu:
2. Dữ liệu đọc từ Api:

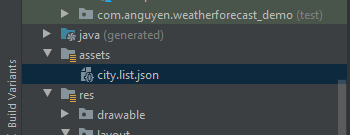
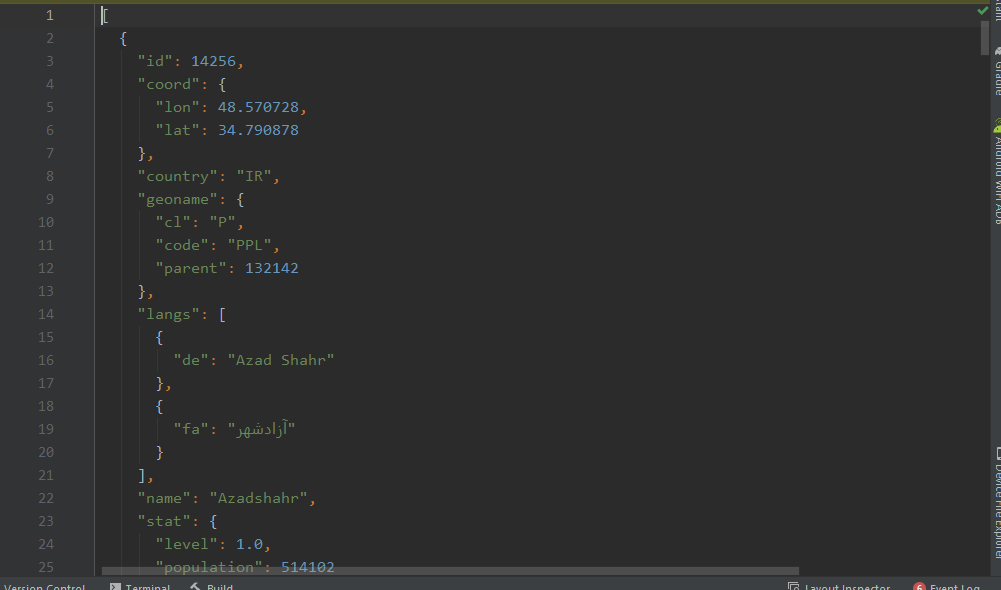
* OpenWeatherMap cung cấp thông tin thời tiết thông qua các Url cụ thể của từng thành phố. Url bao gồm tên miền của api (api.openweathermap.com), sau đó là phần đường dẫn xác định từng thành phố (ví dụ tên thành phố, tọa độ, id, ...) và sau cùng là Api key được cung cấp khi đăng ký tài khoản trên OpenWeatherMap



* Url trên sẽ dẫn tới trang chứa thông tin tiết theo dạng Json. Để sử dụng phải dùng thư viện Http client đọc đường dẫn này sẽ trả về một JsonObject (Trong app này sử dụng **Retrofit**)

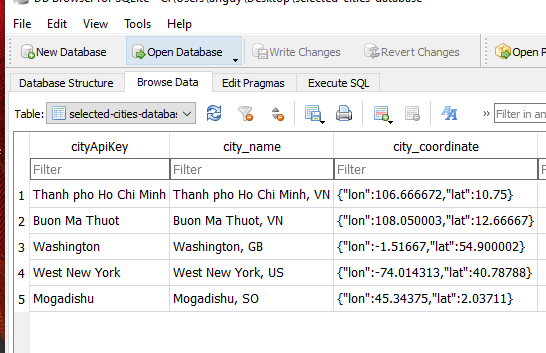


1. Dữ liệu đầu vào:

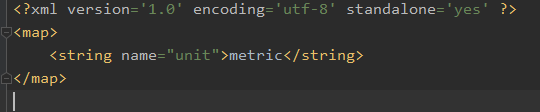
* Lưu danh sách một số thành phố ở Việt Nam trên thế giới
* Cấu trúc theo dạng Json Array, định dạng file \*.txt, lưu trong đường dẫn /main/assets

1. Dữ liệu cục bộ:

* Sử dụng Room Database lưu các thành phố mà người dùng muốn xem thời tiết đã chọn



* Sử dụng SharePreference lưu các thay đổi của chức năng cài đặt



--Kết thúc--