

Nguyễn Lê Thanh Hiển

MSSV: 22520418

Ngày hoàn thành bài tập: 27/11

## BÁO CÁO THỰC HÀNH BÀI TẬP 18/11

### Ứng dụng ToDoList: Thêm vào Menu điều chỉnh

Môn: Nhập môn ứng dụng di động (SE114.Q11)

\*Link Github source code:

<https://github.com/BeginnerCoder52/ToDoList>

### I. Mô tả ứng dụng

Ứng dụng ToDoList là một ứng dụng quản lý công việc cá nhân được thiết kế theo phong cách hiện đại với giao diện màu xanh dương chủ đạo, bo tròn mềm mại, lấy cảm hứng từ các ứng dụng quản lý thói quen phổ biến. Ứng dụng cho phép người dùng thêm công việc mới kèm tiêu đề, mô tả, thời hạn và trạng thái, đồng thời hỗ trợ đánh dấu hoàn thành nhanh chóng thông qua Checkbox.

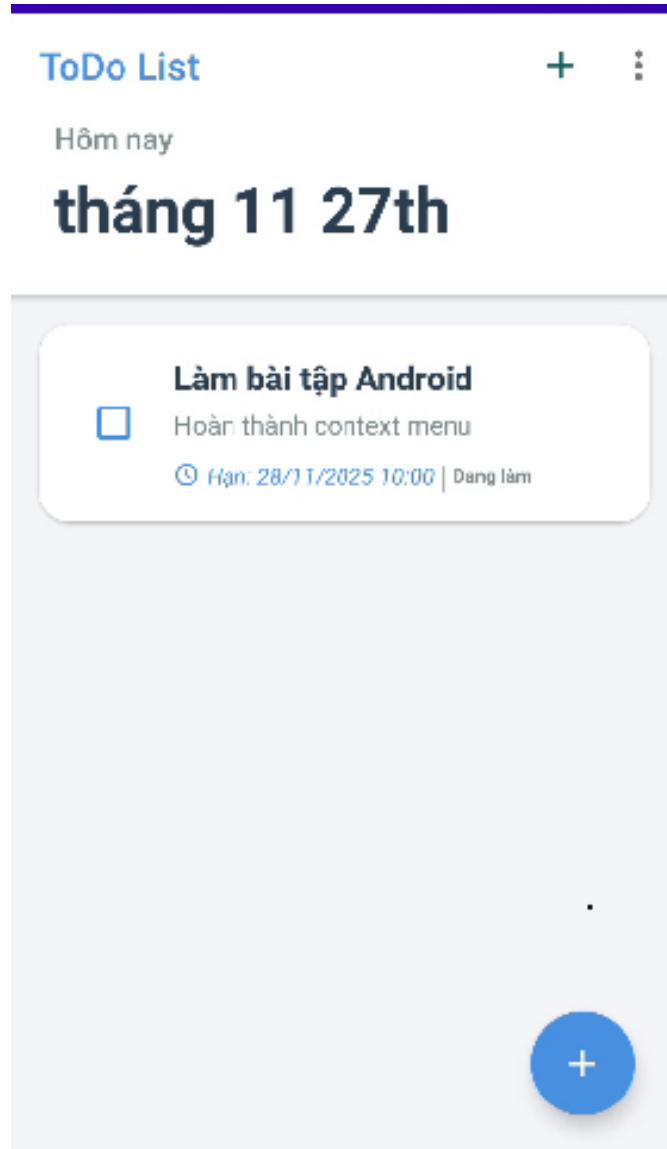
Trong bài thực hành này, ứng dụng được nâng cấp thêm các tính năng điều khiển thông qua Option Menu (trên thanh tiêu đề) và Context Menu (nhấn giữ vào công việc).

### II. Cách hoạt động của hệ thống

Quy trình hoạt động của ứng dụng được mô tả theo thứ tự thao tác thực tế như sau:

**Bước 1: Giao diện chính và Option Menu** Khi mở ứng dụng, màn hình chính hiển thị danh sách các công việc cần làm cùng với ngày tháng hiện tại.

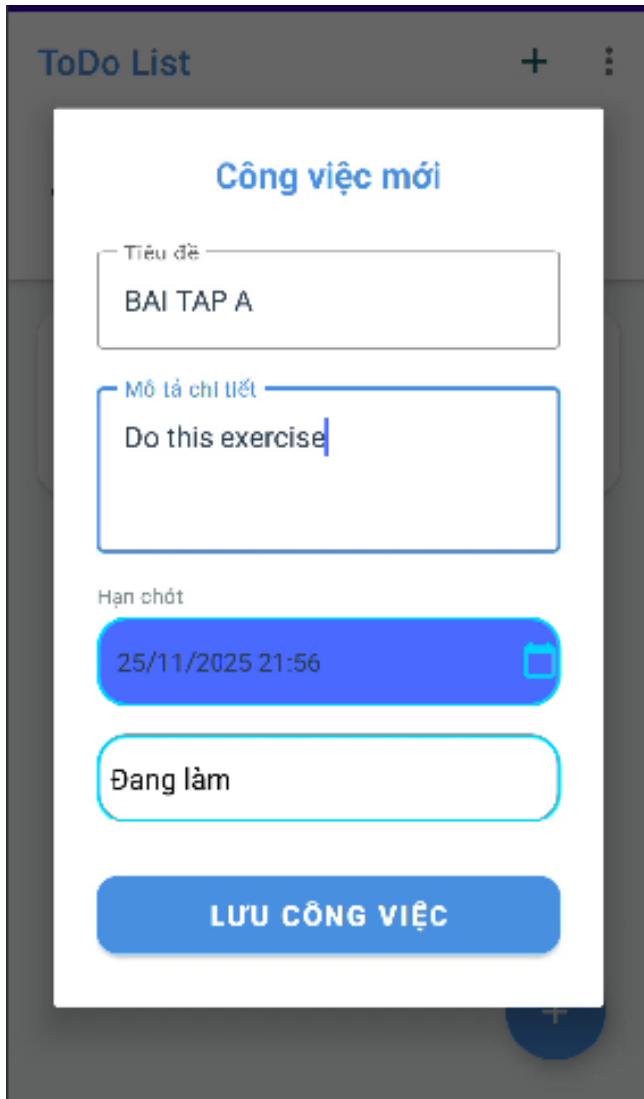
- Góc trên bên phải có thanh Toolbar chứa nút **Thêm (+)** và dấu ba chấm để mở **Option Menu**.



*Giao diện chính hiển thị danh sách công việc.*

**Bước 2: Thêm công việc mới (New Task)** Người dùng nhấn vào nút Floating Action Button (FAB) ở góc dưới hoặc chọn "New" trên Option Menu. Hộp thoại nhập liệu xuất hiện:

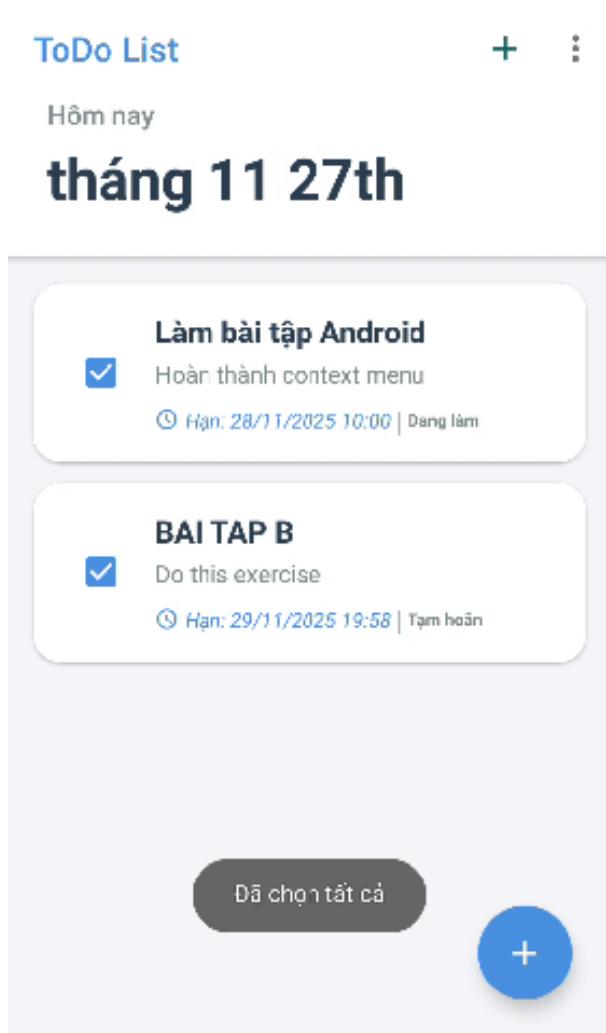
- Người dùng nhập tiêu đề: "BAI TAP A".
- Mô tả: "Do this exercise".
- Chọn hạn chót và trạng thái ban đầu.



*Hộp thoại thêm mới đang được điền thông tin.*

**Bước 3: Chức năng Chọn tất cả (Select All)** Sau khi đã có danh sách công việc, người dùng mở Option Menu và chọn "Select All".

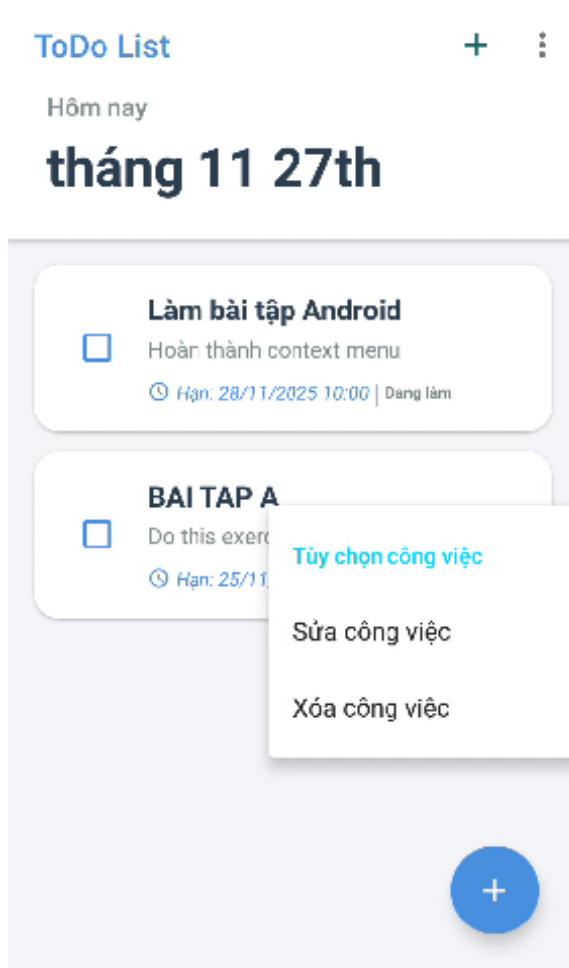
- Hệ thống tự động đánh dấu tick (Checked) vào tất cả các ô Checkbox của các công việc hiện có.



Danh sách công việc sau khi nhấn "Select All", tất cả đều được chọn.

**Bước 4: Context Menu (Tùy chọn cho từng công việc)** Để thao tác riêng với một công việc cụ thể (ví dụ: công việc "BAI TAP A"), người dùng thực hiện thao tác **nhấn giữ (Long Click)** vào thẻ công việc đó.

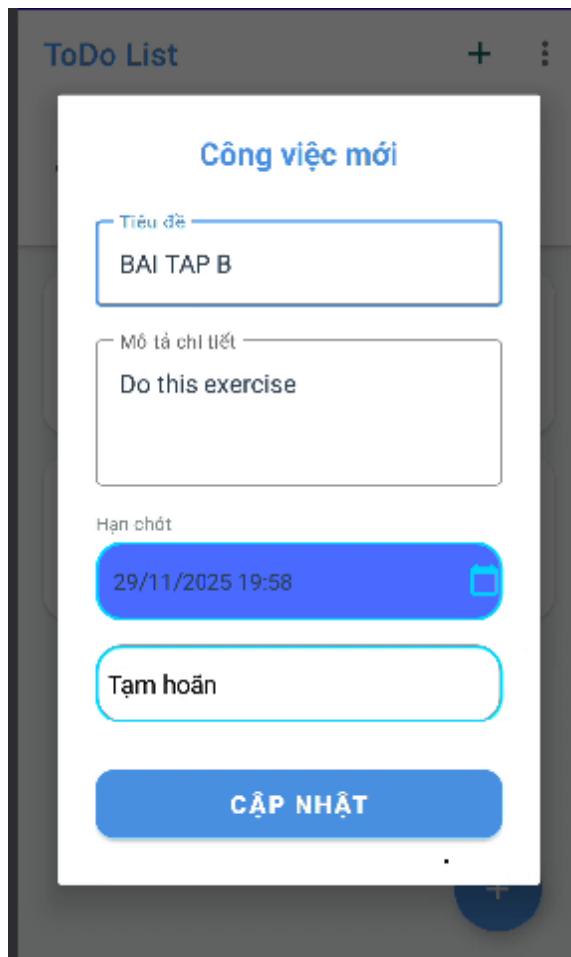
- Một Context Menu hiện lên với 2 tùy chọn: "**Sửa công việc**" và "**Xóa công việc**".



*Context Menu hiện lên khi nhấp giữ vào item "BAI TAP A".*

**Bước 5: Chính sửa công việc (Edit Task)** Người dùng chọn "Sửa công việc" từ Context Menu.

- Hộp thoại xuất hiện trở lại nhưng điền sẵn thông tin cũ.
- Người dùng tiến hành sửa Tiêu đề thành "BAI TAP B", thay đổi hạn chót và chuyển trạng thái sang "Tạm hoãn".
- Nhấn nút "CẬP NHẬT" để lưu lại thay đổi.



Hộp thoại chỉnh sửa với thông tin đã được cập nhật trước khi lưu.

### III. Cấu trúc các file và chức năng

So với phiên bản cơ bản ban đầu, cấu trúc mã nguồn đã được cập nhật và bổ sung các thành phần sau để xử lý Menu và Dialog:

#### 1. File XML (Giao diện)

- **res/menu/main\_menu.xml (Mới):** File định nghĩa Option Menu gồm 3 item: *New* (Thêm), *Select All* (Chọn hết), *Delete Selected* (Xóa mục đã chọn).
- **res/layout/activity\_main.xml:**
  - Bổ sung Toolbar để hiển thị được Option Menu.
  - Điều chỉnh layout sử dụng CoordinatorLayout để tương thích tốt hơn với các thành phần Material Design.
- **res/layout/item\_todo.xml:**

- Thiết kế lại theo dạng CardView bo tròn.
- Tăng kích thước Checkbox (48dp) để dễ thao tác chạm.
- **res/layout/dialog\_add\_todo.xml:**
  - Sử dụng TextInputLayout để nhập liệu chuyên nghiệp hơn.
  - Tái sử dụng layout này cho cả tính năng Thêm mới (Add) và Chính sửa (Edit).

## 2. File Java (Logic xử lý)

- **MainActivity.java:**
  - **Option Menu:** Thêm hàm onCreateOptionsMenu để nạp file menu xml và onOptionsItemSelected để xử lý sự kiện (Thêm, Chọn hết, Xóa).
  - **Context Menu:** Thêm hàm onContextItemSelected để nhận diện sự kiện Edit/Delete gửi từ Adapter.
  - **Logic Dialog:** Viết lại hàm showTodoDialog chấp nhận tham số đầu vào. Nếu tham số là null -> Chế độ Thêm mới; Nếu có dữ liệu -> Chế độ Chính sửa (điền sẵn thông tin cũ).
  - **Xử lý dữ liệu:** Bổ sung hàm deleteSelectedTasks sử dụng Iterator để xóa toàn các mục đã hoàn thành.
- **TodoAdapter.java:**
  - Implement View.OnCreateContextMenuListener trong TodoViewHolder.
  - Đăng ký Context Menu cho từng itemView để khi nhấn giữ sẽ hiện popup Sửa/Xóa.
  - Sử dụng getAdapterPosition() để truyền vị trí item chính xác về cho MainActivity xử lý.

## IV. Kết quả đạt được

- Hiểu và vận dụng thành công **Option Menu** để thực hiện các thao tác tác động lên toàn bộ danh sách (Batch actions).
- Thực hiện được **Context Menu** thông qua sự kiện Long Click để thao tác trên từng đối tượng cụ thể.

- Xây dựng được ứng dụng ToDoList hoàn chỉnh với đầy đủ các chức năng CRUD (Create, Read, Update, Delete).
- Giao diện được tối ưu hóa theo chuẩn Material Design, dễ nhìn và dễ sử dụng.