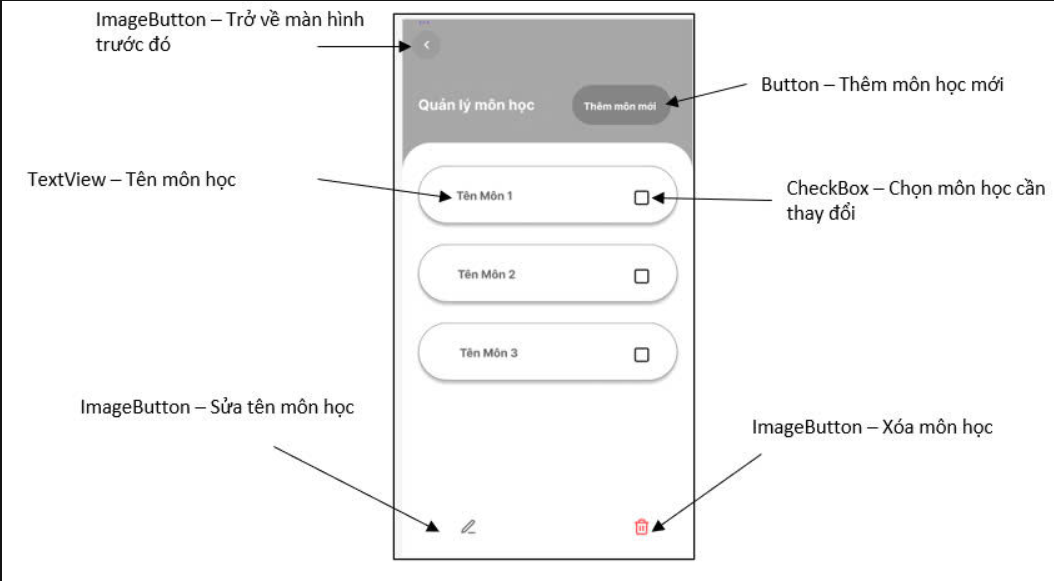
2.3.2.Nguyễn Hữu Đức - Quản lý môn học

2.3.2.1.Màn hình quản lý môn học



*Hinh ? :Nguyễn mẫu và mô tả giao diện quản lý môn học (SubjectActivity)*

* Các chức năng chính của màn hình :

- Chức năng thêm môn học: Nhấn vào ImageButton Thêm môn mới (Biểu tượng dấu “+”) .Ứng dụng sẽ hiện ra 1 Dialog yêu cầu người dùng nhập tên của môn học mới.Người dùng nhập tên môn học mới sau đó bấm xác nhận.

* Mã nguồn của chức năng thêm môn học:

addSubjectBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

try {

addSubjectLayout.setVisibility(View.VISIBLE);

addSubjectBox.setVisibility(View.VISIBLE);

check = 0;

}

catch (Exception e) {

Toast.makeText(SubjectActivity.this, "Không thể mở giao diện thêm môn học!", Toast.LENGTH\_SHORT).show();

}

}

});

if (subjectName.getText().toString().isEmpty()) {

Toast.makeText(SubjectActivity.this, "Bạn phải nhập tên môn học!", Toast.LENGTH\_SHORT).show();

return;

}

if (check == 0) {

subjectTable.addNewSubject(subjectName.getText().toString(), userId);

Log.d("onClick:", "add subject!");

Toast.makeText(SubjectActivity.this, "Thêm môn học thành công!", Toast.LENGTH\_SHORT).show();

}

subjects.clear();

subjects.addAll(subjectTable.getSubjectsOfUserID(userId));

Log.d("add", subjects.toString());

subjectAdapter.notifyDataSetChanged();

addSubjectLayout.setVisibility(View.GONE);

addSubjectLayout.setClickable(false);

subjectName.setText("");

addSubjectBox.setVisibility(View.GONE);

-Chức năng sửa môn học : Nhấn vào 1 học, màn hình sẽ bôi xanh nền môn học được chọn, người dùng click vào ImageButton sửa để sửa môn học.Ứng dụng sẽ hiện ra 1 Dialog yêu cầu người dùng nhập tên mới của môn học.Người dùng nhập tên môn học mới sau đó bấm xác nhận.

* Mã nguồn của chức năng sửa môn học:

subjectList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

try {

selectedList.add(i);

editSubject = i;

if (selectedList.size() > 1) {

for (int j = 0; j < subjectList.getCount(); j++) {

View reBackgroundView = subjectList.getChildAt(j);

if (reBackgroundView != null) {

reBackgroundView.setBackgroundColor(Color.TRANSPARENT);

}

}

}

view.setBackgroundColor(Color.parseColor("#00BCD4"));

Log.d("selectedList:", selectedList.toString());

}

catch (Exception e) {

Toast.makeText(SubjectActivity.this, "Không chọn được môn cần sửa!", Toast.LENGTH\_SHORT).show();

}

}

});

if (check == 1) {

if (selectedList.isEmpty()) {

Toast.makeText(SubjectActivity.this, "Bạn phải chọn môn học cần sửa trước!", Toast.LENGTH\_SHORT).show();

return;

}

subjectTable.updateSubjectName(subjects.get(editSubject).getSubjectID(), subjectName.getText().toString(), userId);

Log.d("onClick:", "update subject!");

Toast.makeText(SubjectActivity.this, "Sửa môn học thành công!", Toast.LENGTH\_SHORT).show();

}

subjects.clear();

subjects.addAll(subjectTable.getSubjectsOfUserID(userId));

Log.d("add", subjects.toString());

subjectAdapter.notifyDataSetChanged();

addSubjectLayout.setVisibility(View.GONE);

addSubjectLayout.setClickable(false);

subjectName.setText("");

addSubjectBox.setVisibility(View.GONE);

-Chức năng xóa môn học : Chọn check box vào 1 hoặc nhiều môn học, sau đó click vào ImageButton xóa để xác nhận xóa môn học.Ứng dụng sẽ xóa hết các môn học được người dùng checkBox.

* Mã nguồn của chức năng xóa môn học :

// Xử lý sự kiện khi checkbox thay đổi trạng thái

checkBox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(CompoundButton compoundButton, boolean isChecked) {

try {

// Cập nhật trạng thái checkbox vào đối tượng dữ liệu

myList.get(position).setChecked(isChecked);

// Cập nhật mảng checkedItems khi checkbox được chọn hoặc bỏ chọn

if (isChecked) {

// Nếu checkbox được check, thêm chỉ số vào mảng

SubjectActivity.getCheckList.add(position);

} else {

// Nếu checkbox uncheck, loại bỏ chỉ số khỏi mảng

SubjectActivity.getCheckList.remove(Integer.valueOf(position));

}

} catch (Exception e) {

Toast.makeText(context, e.getMessage(), Toast.LENGTH\_SHORT).show();

}

}

});

delBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

try {

Log.d("checkboxes:", getCheckList.toString());

for (int i = 0; i < getCheckList.size(); i++) {

//Day la truy van xoa

Log.d("name:", subjects.get(getCheckList.get(i)).getSubjectName());

subjectTable.deleteSubjectByName(subjects.get(getCheckList.get(i)).getSubjectName(), userId);

}

getCheckList.clear();

subjects.clear();

subjects.addAll(subjectTable.getSubjectsOfUserID(userId));

subjectAdapter.notifyDataSetChanged();

subjectName.setText("");

}

catch (Exception e) {

Toast.makeText(SubjectActivity.this, "Lỗi khi xóa môn học!", Toast.LENGTH\_SHORT).show();

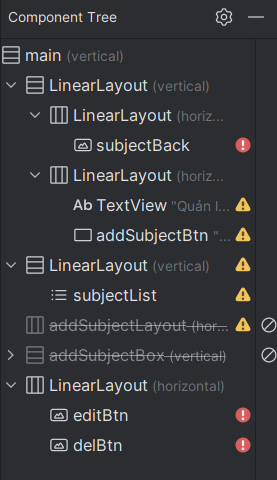
Toast.makeText(SubjectActivity.this, e.getMessage(), Toast.LENGTH\_SHORT).show();

}

}

});

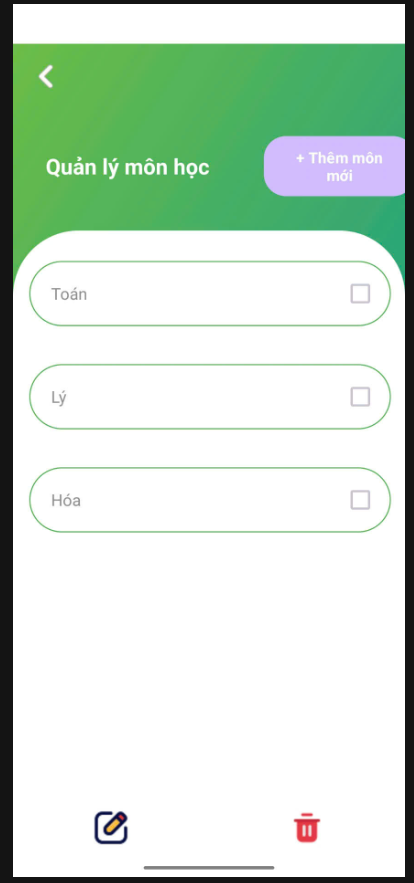
}



*Hình ? : Component Tree của giao diện quản lý môn học*

*(activity\_subject.xml)*

* Giao diện trên thiết bị:



*Hình ? :Giao diện màn hình quản lý môn học trên thiết bị thực tế*