

# BÀI TẬP LỚN

LẬP TRÌNH THIẾT BỊ DI ĐỘNG

## XÂY DỰNG DỰNG DỤNG QUIZZ APP TOÁN LỚP 3

**Nhóm sinh viên thực hiện:**

1. Võ Lê Ngọc Thuận
2. Nguyễn Hữu Thọ

**Lớp:** 64.HTTT

**Cán bộ giảng dạy:** Mai Cường Thọ

## MỤC LỤC

1. GIỚI THIỆU .....	4
1.1 Mục tiêu của dự án.....	4
1.2. Tầm quan trọng của ứng dụng .....	4
1.3 Đặc điểm chính của ứng dụng:.....	5
PHẦN 2: PHÂN TÍCH YÊU CẦU .....	6
2.1. Yêu cầu chức năng .....	6
2.2. Yêu cầu phi chức năng.....	6
PHẦN 3: THIẾT KẾ ỨNG DỤNG .....	8
3.1. Kiến trúc tổng thể.....	8
3.2 Thiết kế giao diện người dùng (UI Design) .....	8
3.3 Thiết kế cơ sở dữ liệu (nếu có).....	10
PHẦN 4: CÀI ĐẶT CÁC XỬ LÝ, CÁC CHỨC NĂNG CHÍNH.....	11
4.1. Tạo cơ sở dữ liệu bằng lệnh .....	11
4.1.1. Phân xử lý đăng nhập bằng firebase.....	11
4.1.1.1. Phân xử lý tạo tài khoản bằng firebase .....	11
4.1.1.2. Phân xử lý khôi phục mật khẩu bằng firebase.....	12
4.2. Cài đặt các xử lý, các chức năng chính .....	12
PHẦN 5: KẾT QUẢ .....	16
5.1. Màn hình sau khi chạy ứng dụng .....	16
5.2. Màn hình sau khi hoạt ảnh kết thúc .....	17
5.2.1. Màn hình sau khi người dùng click vào tùy chọn “Đăng ký” .....	18
5.2.2. Màn hình sau khi người dùng click vào tùy chọn “Quên mật khẩu?” .....	19
5.3. Màn hình sau khi đăng nhập .....	20
5.4. Màn hình sau khi click vào lựa chọn Bài tập .....	21
5.5. Màn hình sau khi click vào 1 trong 5 chương công, trừ, nhân, chia và nâng cao (chức năng của các chương giống nhau, chỉ khác nhau ở các câu hỏi theo chương).....	22
5.6. Màn hình sau khi hoàn thành Quiz trong trang GiaoDienQuiz.....	24
5.7. Màn hình sau khi click vào lựa chọn luyện tập tự do.....	25
5.8. Màn hình sau khi click vào lựa chọn Bảng cửu chương .....	27
5.9. Màn hình sau khi click vào lựa chọn thử thách.....	28
5.9.1. Màn hình sau khi hết thời gian trong phần Luyện tập.....	30

5.9.2. Màn hình sau khi hoàn thành 10 câu hỏi trong phần Luyện tập .....	31
5.10. Màn hình mở mục cài đặt.....	32
5.10.1. Màn hình sau khi nhấn vào lựa chọn “Thông tin người dùng” .....	33
5.10.2. Màn hình sau khi nhấn vào lựa chọn “Giới thiệu ứng dụng” .....	34
5.10.3. Màn hình sau khi nhấn vào lựa chọn “Đăng xuất”.....	35
PHẦN 6: ĐÁNH GIÁ VÀ KẾT LUẬN .....	36
6.1. Đánh giá kết quả .....	36
6.2. Kết luận.....	36
PHẦN 7: PHỤ LỤC.....	37

# 1. GIỚI THIỆU

## 1.1. Mục tiêu của dự án

Mục tiêu của bài tập lớn là xây dựng một ứng dụng di động Quiz Toán học dành cho học sinh lớp 3 trên ứng dụng Android Studio nhằm hỗ trợ các em học sinh lớp 3 ôn tập và rèn luyện kiến thức toán học một cách sinh động và hiệu quả. Thông qua ứng dụng, học sinh có thể luyện tập các dạng toán cơ bản như cộng, trừ, nhân, chia, và giải các bài toán có lời văn, từ đó nâng cao tư duy logic và kỹ năng giải toán.

Lý do chọn chủ đề ứng dụng di động

- Thiết bị di động phổ biến: Điện thoại thông minh và máy tính bảng đang được sử dụng rộng rãi, kể cả với học sinh tiểu học, tạo điều kiện thuận lợi cho việc học mọi lúc mọi nơi.

- Tính tiện lợi và linh hoạt: Việc lựa chọn chủ đề này xuất phát từ nhu cầu thực tế trong việc học trực tuyến, tự học tại nhà ngày càng phổ biến – đặc biệt ở lứa tuổi tiểu học, khi việc học cần kết hợp giữa kiến thức và sự hứng thú.

- Hướng tới học tập hiện đại: Việc học thông qua ứng dụng giúp tăng tính tương tác, trực quan và phù hợp với xu hướng chuyển đổi số trong giáo dục hiện nay.

- Tăng hứng thú học tập: Trẻ em thường yêu thích công nghệ, do đó học qua ứng dụng sẽ tạo sự hứng thú và giảm cảm giác áp lực khi học toán theo cách truyền thống.

## 1.2. Tầm quan trọng của ứng dụng

Trong bối cảnh giáo dục hiện đại, đặc biệt sau đại dịch COVID-19, việc học trực tuyến và tự học tại nhà ngày càng trở nên phổ biến. Tuy nhiên, học sinh tiểu học – đặc biệt là lớp 3 – vẫn còn hạn chế về khả năng tự giác và tập trung. Do đó, cần có một công cụ hỗ trợ học tập vừa mang tính giáo dục, vừa tạo hứng thú học tập cho các em.

Ứng dụng QuizApp Toán lớp 3 ra đời nhằm giải quyết các vấn đề sau:

- Thiếu công cụ luyện tập Toán phù hợp lứa tuổi: Nhiều ứng dụng hiện có không tối ưu cho học sinh tiểu học hoặc quá phức tạp. QuizApp được thiết kế đơn giản, thân thiện, dễ sử dụng với học sinh lớp 3.

- Tạo môi trường học tập vui nhộn, tương tác: Với hình ảnh minh họa đáng yêu, tính năng câu hỏi trắc nghiệm có điểm số, thời gian, ứng dụng giúp các em cảm thấy học Toán giống như đang chơi trò chơi.

- Giúp phụ huynh và giáo viên theo dõi tiến độ học tập: Thông qua điểm số và số lần luyện tập, phụ huynh có thể biết được con mình đang tiến bộ như thế nào mà không cần kiểm tra trực tiếp.

### 1.3. Đặc điểm chính của ứng dụng

- Đăng nhập, đăng ký và quên mật khẩu: Đây là các chức năng cơ bản được thấy trong hầu hết các app, làm gia tăng bảo mật thông tin của người dùng.

- Ôn tập theo chương: Người dùng có thể chọn các chương cụ thể (Cộng, Trừ, Nhân, Chia, bài toán có lời văn) để luyện tập các câu hỏi phù hợp với nội dung đang học.

- Luyện tập tự do: Chức năng này cho phép người dùng làm bài không giới hạn số lượng câu hỏi, không giới hạn thời gian cũng như không hiển thị số điểm đạt được mà chức năng này chọn ngẫu nhiên từ toàn bộ ngân hàng câu hỏi để rèn luyện kỹ năng tổng hợp.

- Chế độ thử thách: Người dùng làm bài (tự luận) trong thời gian giới hạn, được chấm điểm ngay sau khi hoàn thành. Giúp rèn luyện phản xạ nhanh và tạo cảm giác thi đua.

- Chế độ tra cứu bảng cửu chương: Ứng dụng cho phép người dùng tra cứu bảng cửu chương (cộng, trừ, nhân, chia  $\leq 10$ ) giúp học sinh dễ dàng tra cứu để học thuộc và hỗ trợ ghi nhớ kiến thức cơ bản.

- Giao diện thân thiện – minh họa bằng hình ảnh dễ thương: Giúp học sinh nhỏ tuổi dễ tiếp cận và có hứng thú học tập.

- Cài đặt: Xem thông tin của ứng dụng, thông tin người dùng và đăng xuất ra khỏi phần đăng nhập.

- Giao diện dễ sử dụng: Thiết kế thân thiện với học sinh tiểu học, sử dụng hình ảnh và màu sắc sinh động giúp học sinh nhỏ tuổi dễ tiếp cận và có hứng thú học tập

## PHẦN 2: PHÂN TÍCH YÊU CẦU

### 2.1. Yêu cầu chức năng

- Quản lý tài khoản người dùng:
  - + Đăng nhập: Cho phép người dùng (học sinh hoặc phụ huynh) đăng nhập vào hệ thống.
  - + Đăng ký: Cho phép người dùng đăng ký tài khoản mới.
  - + Quên mật khẩu: Cho phép người dùng đặt lại mật khẩu trong email sau khi mã được gửi tới.
- Ôn tập theo chương: Cho phép người dùng chọn chương (cộng, trừ, nhân, chia, bài toán lời văn) để làm bài trắc nghiệm theo từng chủ đề.
- Luyện tập tự do: Cho phép học sinh làm bài không giới hạn thời gian, không giới hạn số lượng câu hỏi tùy chọn.
- Chế độ thử thách: Học sinh làm 10 câu hỏi tự luận trong thời gian giới hạn (ví dụ 15 giây mỗi câu), ứng dụng tự động chấm điểm.
- Chế độ tra cứu bảng cửu chương: Cho phép học sinh tra các bảng cửu chương cộng, trừ, nhân, chia từ 1-10 để xem lại (nếu quên) hoặc học thuộc.
- Xáo trộn câu hỏi và đáp án: Hệ thống tự động trộn câu hỏi, câu trả lời để tránh trùng lặp và tạo tính ngẫu nhiên cho mỗi lần làm bài.
- Hiển thị kết quả: Khi người dùng chọn kết quả đúng thì ứng dụng sẽ tô xanh kết quả vừa chọn và ngược lại khi người dùng chọn kết quả sai thì ứng dụng sẽ tô đỏ kết quả người dùng vừa chọn sau đó hệ thống sẽ tô xanh kết quả đúng của câu hỏi đó. Khi làm bài, ứng dụng hiển thị số câu đúng, sai và tổng điểm (theo thang điểm 10). Sau khi kết thúc, ứng dụng sẽ hiện ra đánh giá “Đạt” nếu đúng từ 5 câu trở lên và “chưa đạt” nếu sai hơn 5 câu.
- Quay lại màn hình chính trước đó: Cho phép người dùng quay lại trang trước đó.
- Giao diện thân thiện: Thiết kế đơn giản, dễ sử dụng, phù hợp với học sinh tiểu học.

### 2.2. Yêu cầu phi chức năng

- Giao diện người dùng thân thiện: Thiết kế giao diện đơn giản, trực quan, sử dụng màu sắc tươi sáng và font chữ dễ đọc, phù hợp với học sinh tiểu học.

- Tính ổn định: Ứng dụng cần hoạt động mượt mà, không bị giật, lag hoặc lỗi khi chuyển giữa các màn hình hoặc khi làm bài.

- Bảo mật dữ liệu: Tài khoản người dùng, dữ liệu câu hỏi, đáp án và kết quả làm bài phải được lưu trữ an toàn, không bị mất hoặc thay đổi ngoài ý muốn.

- Khả năng mở rộng: Ứng dụng cần dễ dàng nâng cấp để thêm câu hỏi mới, thêm chương học hoặc cập nhật chức năng mới trong tương lai.

- Tương thích tốt: Ứng dụng phải chạy ổn định trên nhiều thiết bị Android với các kích thước màn hình khác nhau

## PHẦN 3: THIẾT KẾ ỨNG DỤNG

### 3.1. Kiến trúc tổng thể

Ứng dụng Quiz Toán lớp 3 được thiết kế theo mô hình kiến trúc Mobile – Backend Firebase, đảm bảo tính đơn giản, hiệu quả và dễ mở rộng. Firebase là nền tảng backend mạnh mẽ của Google, hỗ trợ nhiều tính năng phù hợp với ứng dụng giáo dục trên di động, ứng dụng bao gồm các thành phần chính sau:

#### a. Ứng dụng Mobile (Client side)

- Phát triển bằng Android Studio sử dụng ngôn ngữ Java.
- Chạy trực tiếp trên các thiết bị Android (điện thoại, máy tính bảng).
- Xử lý giao diện người dùng (UI), logic hiển thị câu hỏi, tính điểm, xáo trộn câu trả lời, và phản hồi người dùng.
- Tương tác với Firebase để:
  - + Xác thực người dùng (đăng nhập, đăng ký, quên mật khẩu).
  - + Lưu và truy xuất dữ liệu (câu hỏi, điểm số, thông tin người dùng).

#### b. Firebase (Backend as a Service - BaaS)

Firebase đóng vai trò là backend của hệ thống, bao gồm:

- Firebase Authentication:
  - + Hỗ trợ đăng ký, đăng nhập, gửi lại mật khẩu khi quên mật khẩu thông qua email.
  - + Xác thực bằng email/mật khẩu.
- Firebase Realtime Database / Firestore:
  - + Lưu trữ dữ liệu câu hỏi, đáp án, kết quả làm bài.
  - + Lưu trữ thông tin người dùng (ví dụ: điểm cao nhất, lịch sử học tập).

### 3.2 Thiết kế giao diện người dùng (UI Design)

Mục tiêu thiết kế giao diện: Giao diện của ứng dụng được thiết kế đơn giản, trực quan và sinh động, phù hợp với lứa tuổi học sinh tiểu học. Sử dụng màu sắc tươi sáng, biểu tượng dễ nhận biết và chữ rõ ràng để tạo cảm giác thân thiện và dễ sử dụng.



Các màn hình chính trong ứng dụng:

- Màn hình hoạt ảnh xuất hiện: Làm tăng tính thẩm mỹ cho ứng dụng, làm cho ứng dụng thêm sôi động, tạo cảm giác chuyên nghiệp cho ứng dụng.
- Màn hình đăng nhập: Cho phép người dùng nhập email và mật khẩu.
- Màn hình đăng ký: Cho phép người dùng đăng ký tài khoản.
- Màn hình quên mật khẩu: Cho phép người dùng lấy lại mật khẩu khi nhập email (tài khoản đã đăng ký trước đó), mã sẽ được gửi về email để người dùng đổi mật khẩu.
- Màn hình chính: Cho phép chọn chương học hoặc chọn chế độ làm bài (Luyện tập / Thử thách / Tra bảng cửu chương), mỗi chương có biểu tượng minh họa dễ thương, ngoài ra còn có phần cài đặt dùng để đăng xuất, xem thông tin ứng dụng và xem thông tin người dùng.
- Màn hình chọn chương: Gồm các chương: Cộng, Trừ, Nhân, Chia, và chương nâng cao (các bài toán lời văn).
- Màn hình làm bài: Hiển thị câu hỏi và 4 đáp án (nếu là các câu hỏi trắc nghiệm), hoặc hiển thị câu hỏi và ô nhập kết quả (nếu là các câu hỏi tự luận), kèm đồng hồ đếm ngược nếu ở chế độ thử thách.
- Màn hình kết quả: Hiển thị tổng số câu đúng, sai và điểm số với các tùy chọn làm lại hoặc quay về trang chủ.
- Màn hình tra cứu bảng cửu chương: Cho phép xem bảng cộng, trừ, nhân, chia từ 1 đến 10.
- Màn hình cài đặt: Hiển thị thông tin tài khoản, nút đăng xuất, xem giới thiệu ứng dụng.

Thiết kế phong cách giao diện:

- Màu chủ đạo: Lavender, tím, và trắng – tạo cảm giác vui tươi, màu sắc hài hòa, hình ảnh vui nhộn, giúp người dùng học tập tích cực.
- Font chữ: To, rõ ràng, font chữ kiểu trò chơi, phù hợp cho trẻ em.
- Biểu tượng: Các nút có hình minh họa.
- Tương tác: Hiệu ứng tô màu câu trả lời đúng/sai (xanh/đỏ), hiệu ứng chuyển màn hình mượt mà.

### 3.3 Thiết kế cơ sở dữ liệu

Ứng dụng QuizApp dành cho học sinh lớp 3 sử dụng Firebase Authentication để quản lý người dùng.

Firebase Authentication (xác thực người dùng):

Firebase Authentication được sử dụng để quản lý các chức năng:

- Đăng ký tài khoản (qua email/password).
- Đăng nhập.
- Gửi email đặt lại mật khẩu.

Thông tin người dùng lưu trữ trong Firebase Authentication gồm:

- uid: ID người dùng duy nhất (được Firebase tạo tự động).
- email: Địa chỉ email đăng nhập.
- password: Mật khẩu người dùng (mã hóa).

Chức năng Xác thực người dùng với Firebase Authentication

- Đăng ký tài khoản mới: Khi người dùng đăng ký tài khoản mới, ứng dụng sử dụng Firebase Authentication để tạo tài khoản bằng email và mật khẩu. Sau khi tạo thành công, hệ thống sẽ lấy thông tin người dùng (UID, email) từ đối tượng FirebaseUser và lưu trữ thông tin bổ sung như họ tên, ngày đăng ký, điểm số khởi tạo... vào Firebase Firestore hoặc Realtime Database. Việc này giúp lưu trữ đầy đủ hồ sơ của từng người dùng, phục vụ cho các chức năng như cá nhân hóa nội dung, theo dõi tiến độ học tập và quản lý thống kê.

- Đăng nhập tài khoản: Ở chức năng đăng nhập, người dùng chỉ cần nhập đúng email và mật khẩu đã đăng ký trước đó. Ứng dụng sẽ sử dụng SignInWithEmailPassword của Firebase Authentication để xác thực. Nếu đăng nhập thành công, người dùng được chuyển đến giao diện chính của ứng dụng, nơi họ có thể bắt đầu làm bài quiz. Trong quá trình này, thông tin người dùng cũng được truy xuất từ Firestore để hiển thị tên, điểm số hiện tại hoặc tiến độ làm bài. Việc sử dụng Firebase Authentication giúp đảm bảo bảo mật cho tài khoản và đồng thời cho phép đồng bộ hóa thông tin giữa nhiều thiết bị khi người dùng đăng nhập ở bất kỳ đâu.

- Quên mật khẩu: Chức năng “Quên mật khẩu” giúp người dùng khôi phục lại quyền truy cập nếu quên mật khẩu. Khi người dùng nhập địa chỉ email và xác nhận yêu cầu, hệ thống sẽ sử dụng sendPasswordResetEmail từ Firebase để gửi một email đặt lại mật khẩu. Việc này đảm bảo rằng chỉ người sở hữu email mới có thể thay đổi mật khẩu, tăng tính bảo mật cho ứng dụng.

## PHẦN 4: CÀI ĐẶT CÁC XỬ LÝ, CÁC CHỨC NĂNG CHÍNH

### 4.1. Tạo cơ sở dữ liệu bằng lệnh

#### 4.1.1. Phần xử lý đăng nhập bằng firebase

```
// Gửi yêu cầu đăng nhập đến Firebase Authentication
 mAuth.signInWithEmailAndPassword(emailStr, passwordStr)
    .addOnCompleteListener(task -> {
        if (task.isSuccessful()) {
            FirebaseUser user = mAuth.getCurrentUser();
            Intent intent = new Intent(LoginActivity.this,
Man_hinh_chinh.class);
            intent.putExtra("email", user.getEmail());
            startActivity(intent);
            finish();
        } else {
            Toast.makeText(LoginActivity.this, "Đăng nhập thất
bại: "
                                + task.getException().getMessage(),
Toast.LENGTH_SHORT).show();
        }
    });
};
```

#### 4.1.1. Phần xử lý tạo tài khoản bằng firebase

```
//Gửi yêu cầu tạo tài khoản đến Firebase
 mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(task -> {
        if (task.isSuccessful()) {
            Toast.makeText(this, "Đăng ký thành công!",
Toast.LENGTH_SHORT).show();
            startActivity(new Intent(this, LoginActivity.class));
            finish();
        } else {
            Toast.makeText(this, "Đăng ký thất bại: "
                                + task.getException().getMessage(),
Toast.LENGTH_LONG).show();
        }
    });
};
```

#### 4.1.2. Phân xử lý khôi phục mật khẩu bằng firebase

```
//Gửi yêu cầu khôi phục mật khẩu đến Firebase.
 mAuth.sendPasswordResetEmail(email)
    .addOnCompleteListener(task -> {
        if (task.isSuccessful()) {
            Toast.makeText(this, "Email khôi phục đã được gửi!",
                Toast.LENGTH_SHORT).show();
            finish();
        } else {
            Toast.makeText(this, "Lỗi: "
                + task.getException().getMessage(),
                Toast.LENGTH_SHORT).show();
        }
    });
```

### 4.2. Cài đặt các xử lý, các chức năng chính

#### 4.2.1. Chức năng hoạt ảnh xuất hiện 4 giây

Vào AndroidManifest.xml đặt filter cho trang .Item2.quiz\_xuat\_hien để trang .Item2.quiz\_xuat\_hien xuất hiện đầu tiên

```
public void run() {
    Intent intent = new Intent(quiz_xuat_hien.this,
        LoginActivity.class);
    startActivity(intent);
    finish(); // Đóng màn hình hiện tại
}
}, 4000); // 4000 milliseconds = 4 giây
```

```
<activity
    android:name=".Item2.quiz_xuat_hien"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

#### 4.2.2. Chức năng khởi tạo câu hỏi khi bắt đầu 1 QUIZ mới (Trong phần ôn tập theo chương).

```
//Lấy chương cần làm Quiz từ Intent
int chapter = getIntent().getIntExtra("CHAPTER", 1);
loadQuestions(chapter); // Load danh sách các câu hỏi theo từng chương
// Lấy ngẫu nhiên 10 câu hỏi trong danh sách
Collections.shuffle(questionList);
questionList = new ArrayList<>(questionList.subList(0, 10));
showQuestion(); // Hàm hiển thị câu hỏi đầu tiên
//Gắn sự kiện cho từng thẻ
//Khi người dùng click vào một đáp án, sẽ gọi checkAnswer(String
selectedAnswer) để kiểm tra đúng/sai.
card1.setOnClickListener(v -> checkAnswer(answer1.getText().toString()));
card2.setOnClickListener(v -> checkAnswer(answer2.getText().toString()));
card3.setOnClickListener(v -> checkAnswer(answer3.getText().toString()));
card4.setOnClickListener(v -> checkAnswer(answer4.getText().toString()));
```

#### 4.2.3. Chức năng xáo trộn vị trí các đáp án (Trong phần ôn tập theo chương).

```
//Xáo trộn thứ tự đáp án và gắn vào UI
//Tạo một danh sách rỗng để chứa các đáp án
List<String> shuffledAnswers = new ArrayList<>();
shuffledAnswers.add(current.getAnswer1()); // đáp án A
shuffledAnswers.add(current.getAnswer2()); // đáp án B
shuffledAnswers.add(current.getAnswer3()); // đáp án C
shuffledAnswers.add(current.getAnswer4()); // đáp án D
Collections.shuffle(shuffledAnswers); // Xáo trộn thứ tự các đáp án
// Gắn đáp án vào các card
answer1.setText(shuffledAnswers.get(0));
answer2.setText(shuffledAnswers.get(1));
answer3.setText(shuffledAnswers.get(2));
answer4.setText(shuffledAnswers.get(3));
```

#### 4.2.4. Chức năng kiểm tra đáp án và cập nhật điểm số khi người dùng chọn câu trả lời (Trong phần ôn tập theo chương).

```
Question_Quiz current = questionList.get(currentQuestionIndex); // Lấy câu hỏi
hiện tại
resetCardColors(); // Đặt lại màu nền các card về màu trắng
MaterialCardView selectedCard = getCardByAnswerText(selectedAnswer); // Lấy
card người dùng chọn
MaterialCardView correctCard =
getCardByAnswerText(current.getCorrectAnswer()); // Lấy card đáp án đúng
if (selectedAnswer.equals(current.getCorrectAnswer())) { // Nếu chọn đúng
    correctCount++; // Tăng số câu đúng

selectedCard.setCardBackgroundColor(getResources().getColor(R.color.green)); //
Tô màu xanh lá cho card chọn
} else {
    wrongCount++;
    selectedCard.setCardBackgroundColor(getResources().getColor(R.color.red));
```

```
correctCard.setCardBackgroundColor(getResources().getColor(R.color.green));
}
// Cập nhật lại số liệu hiển thị lên giao diện
tvSoCauDung.setText("✓ Đúng: " + correctCount);
tvSoCauSai.setText("✗ Sai: " + wrongCount);
tvDiemSo.setText("🏆 Điểm: " + correctCount + "/" + questionList.size());
```

#### 4.2.5. Chức năng chuyển sang câu hỏi kế tiếp.

```
// Sau 1 giây, chuyển sang câu hỏi kế tiếp
card1.postDelayed(() -> {
    currentQuestionIndex++;
    resetCardColors();
    showQuestion();
}, 1000);
```

#### 4.2.6. Chức năng tránh lặp lại các câu hỏi cũ (Trong phần thử thách).

```
//Tạo một đối tượng Random để sinh số ngẫu nhiên.
Random random = new Random();
//Trả về một số nguyên ngẫu nhiên từ 0 đến dsCauHoi.size()
int index = random.nextInt(dsCauHoi.size());
//Lấy câu hỏi tại vị trí index và đồng thời xóa câu hỏi đó khỏi danh sách
dsCauHoi
cauHoiHienTai = dsCauHoi.remove(index);
//Cập nhật câu hỏi lên giao diện
tvCauHoi.setText("Câu hỏi: " + cauHoiHienTai noiDung);
//Xóa nội dung cũ trong EditText và đặt lại nó thành rỗng
edtTraLoi.setText("");
tvKetQua.setText("");
edtTraLoi.setEnabled(true);
btnKiemTra.setEnabled(true);
tvThoiGian.setVisibility(View.VISIBLE);
demNguocThoiGian();
```

## 4.2.7. Chức năng đếm ngược thời gian trả lời câu hỏi (Trong phần thử thách).

```
private void demNguocThoiGian() {
    //Điều kiện nếu đang có bộ đếm thời gian chạy thì hủy nó để chuẩn bị
    cho bộ đếm mới
    if (countDownTimer != null) countDownTimer.cancel();
    //Tạo một bộ đếm mới với tổng thời gian là 15.000 ms (15 giây).
    countDownTimer = new CountDownTimer(15000, 1000) {
        //Mỗi giây, cập nhật TextView tvThoiGian hiển thị số giây còn
        lại.
        public void onTick(long millisUntilFinished) {
            tvThoiGian.setText("Thời gian: " + millisUntilFinished /
1000);
        }
    }
}
```

## 4.2.8. Chức năng thiết lập một spinner hiển thị danh sách các số từ 1 đến 10 (Trong phần bảng cửu chương).

```
//Thiết lập Spinner chứa số từ 1 đến 10
List<Integer> listSo = new ArrayList<>();
for (int i = 1; i <= 10; i++) listSo.add(i);
//Thiết lập Adapter cho Spinner để đưa danh sách vào Spinner
ArrayAdapter<Integer> adapter = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_item, listSo);
adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
item);
spinnerSo.setAdapter(adapter);
```

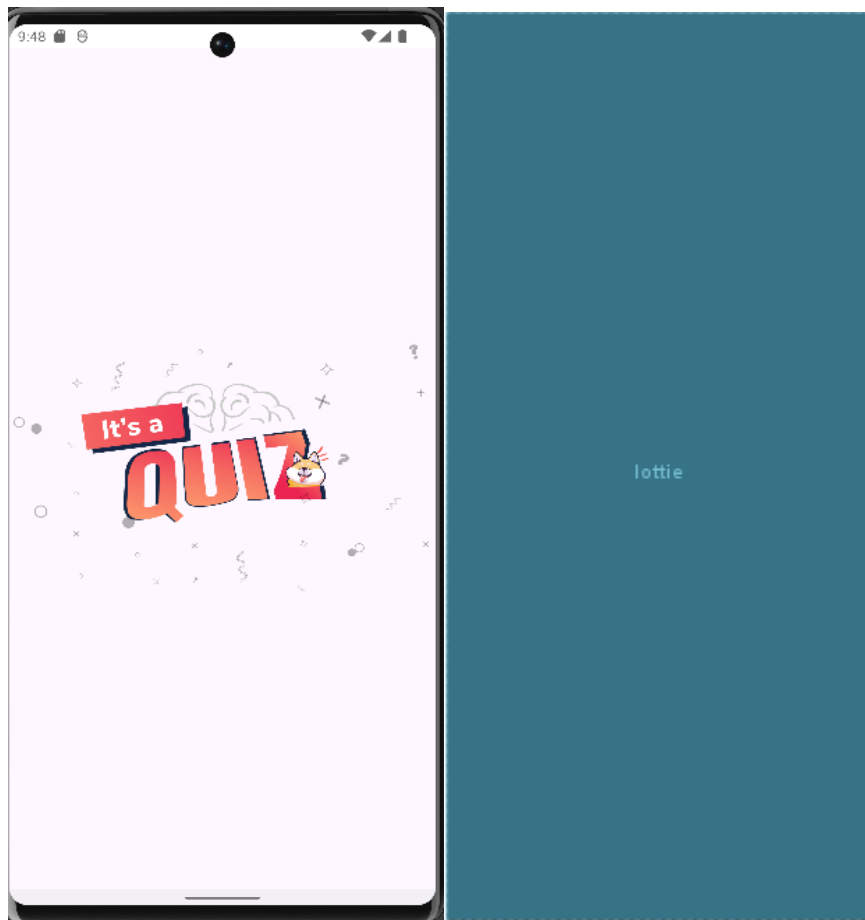
## 4.2.9. Chức năng quay lại.

```
//Khi người dùng nhấn nút, activity hiện tại sẽ kết thúc, trở về màn hình
trước đó
btnQuayLai.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        finish(); // Quay lại activity trước đó
    }
});
```

## PHẦN 5: KẾT QUẢ

### 5.1. Màn hình sau khi chạy ứng dụng

Hoạt ảnh xuất hiện (quizz\_xuat\_hien): Khi chạy ứng dụng, ứng dụng sẽ hiện lên 1 hoạt ảnh động để tạo hiệu ứng, mục đích làm cho ứng dụng thêm sinh động hơn. Trang này sẽ được bao phủ bởi lottie. Hoạt ảnh này được thêm vào trong raw (được tạo từ res).





## 5.2. Màn hình sau khi hoạt ảnh kết thúc

- Sau khi đăng nhập, ứng dụng sẽ chuyển hướng người dùng sang trang đăng nhập (từ trang quiz\_xuat\_hien sang trang LoginActivity).

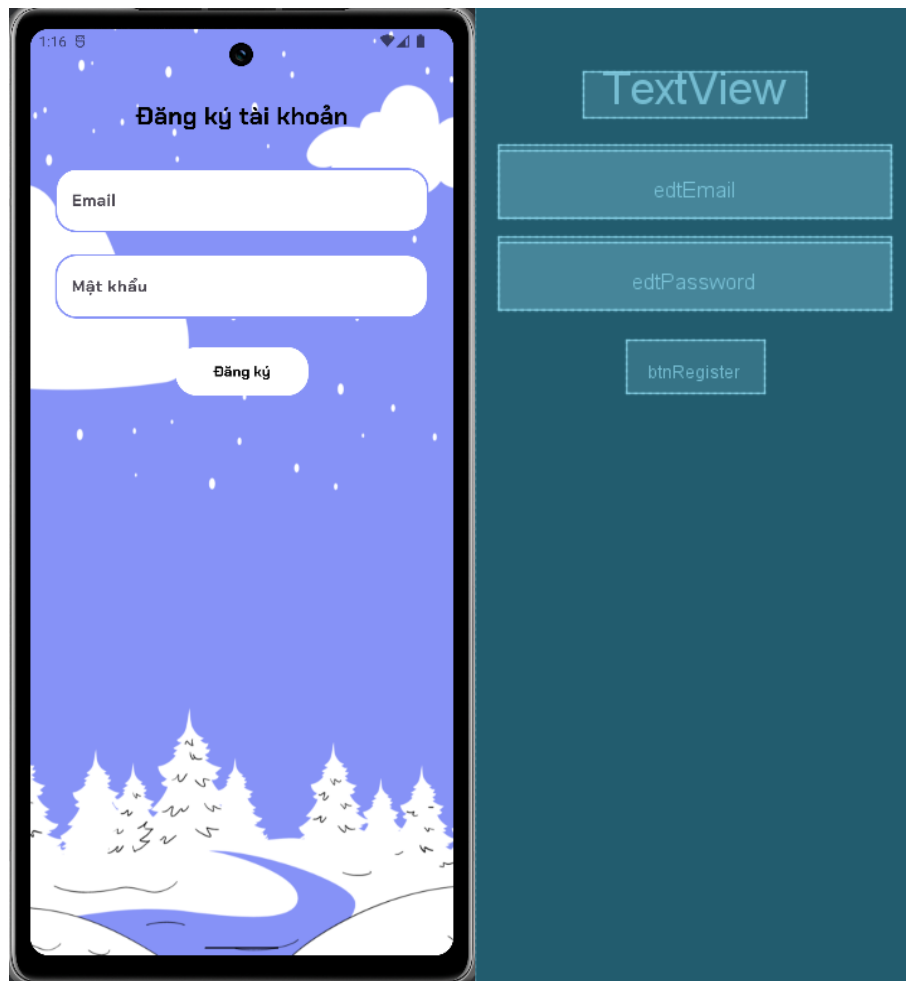
- Tại trang này, ứng dụng yêu cầu người dùng phải điền đầy đủ thông tin bao gồm Email và mật khẩu vào chỗ trống, sau khi điền xong thì click vào button Bắt đầu để tiếp tục, nếu nhập sai hoặc bỏ trống thông tin chưa nhập thì ứng dụng sẽ thông báo nhập lại. Nếu người dùng chưa có tài khoản hoặc đã quên mật khẩu thì click vào 2 tùy chọn bên dưới button bắt đầu.



### 5.2.1. Màn hình sau khi người dùng click vào tùy chọn “Đăng ký”

- Sau khi người dùng click vào tùy chọn đăng ký, ứng dụng sẽ chuyển hướng người dùng sang trang đăng ký (từ trang LoginActivity sang trang RegisterActivity).

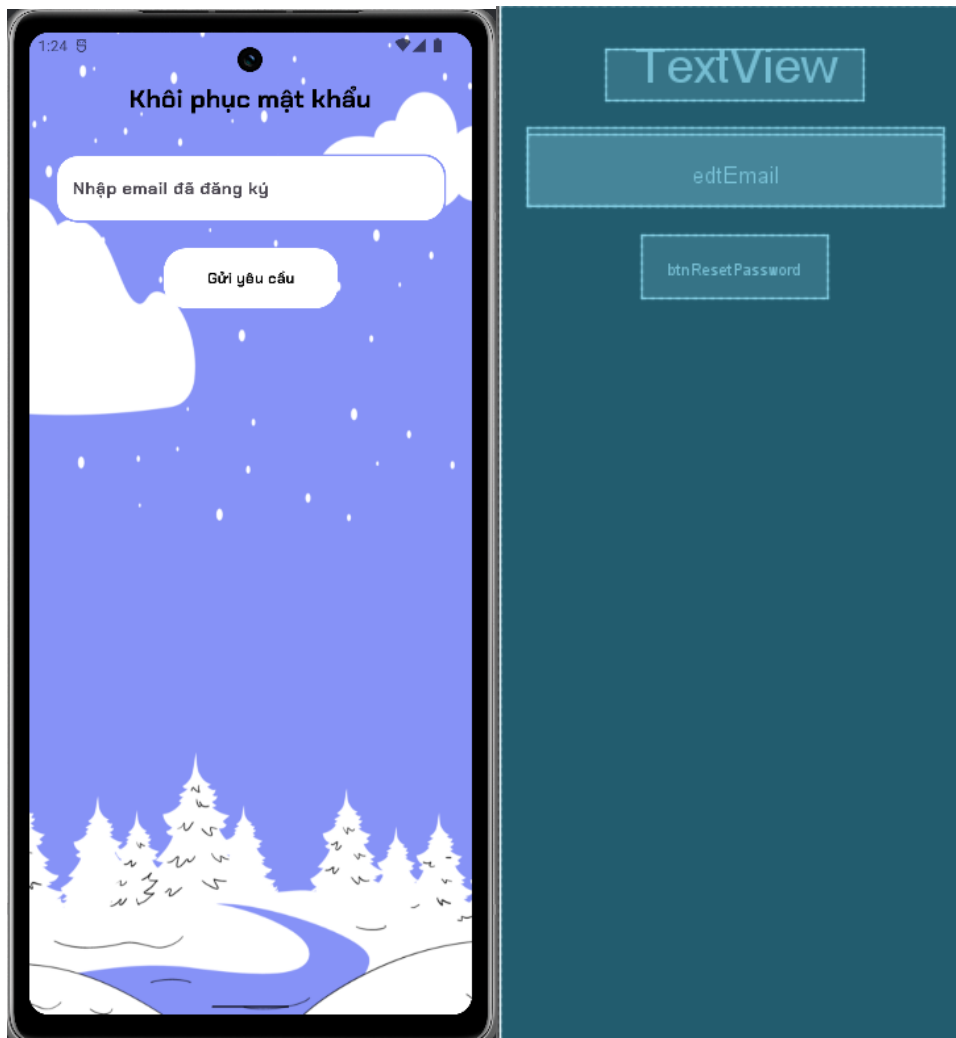
- Tại trang này, ứng dụng yêu cầu người dùng phải điền đầy đủ thông tin bao gồm Email (nên dùng email thật để dễ dàng lấy lại mật khẩu khi quên) và mật khẩu vào chỗ trống, sau khi điền xong thì click vào button đăng ký để tiếp tục, nếu nhập sai hoặc bỏ trống thông tin chưa nhập thì ứng dụng sẽ thông báo nhập lại. Ứng dụng sẽ update tài khoản mới lên firebase và trả người dùng về lại trang đăng nhập để tiến hành đăng nhập.



### 5.2.2. Màn hình sau khi người dùng click vào tùy chọn “Quên mật khẩu?”

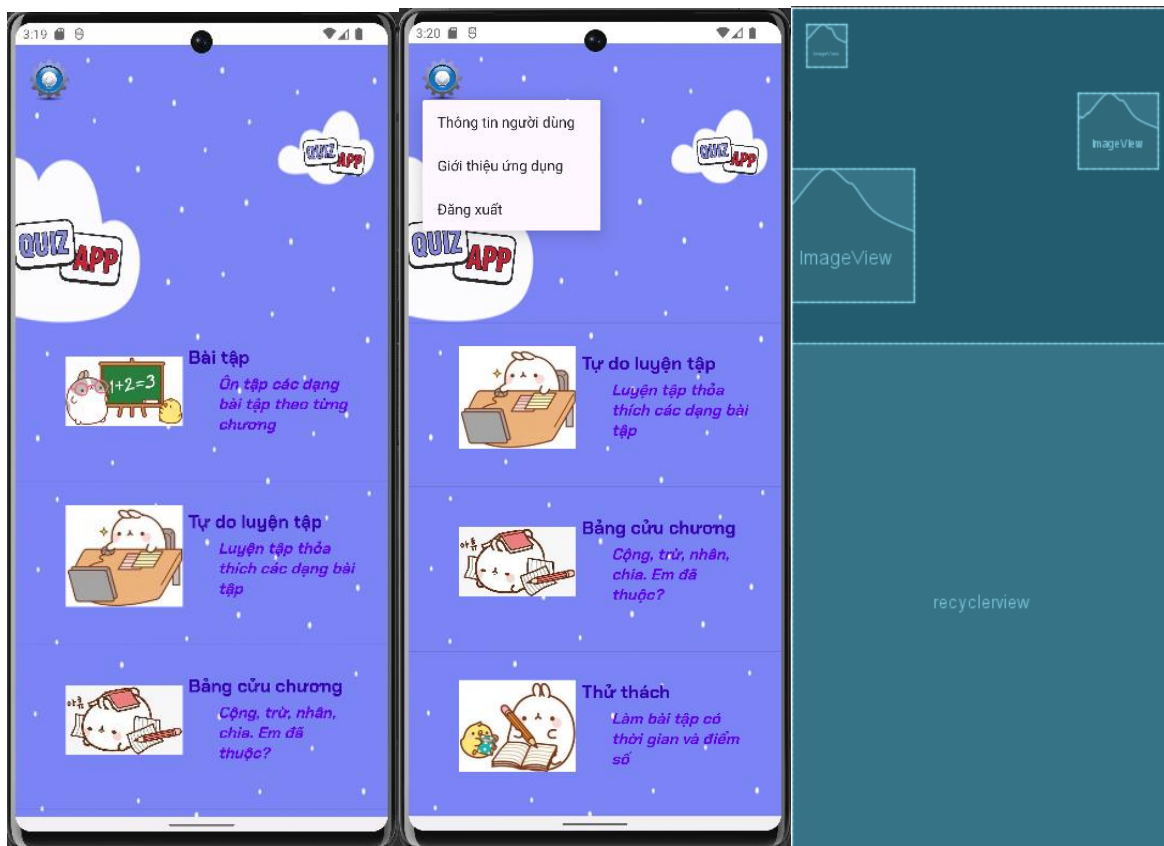
- Sau khi người dùng click vào tùy chọn quên mật khẩu, ứng dụng sẽ chuyển hướng người dùng sang trang quên mật khẩu (từ trang LoginActivity sang trang ForgotPasswordActivity).

- Tại trang này, ứng dụng yêu cầu người dùng phải điền đầy đủ thông tin bao gồm Email (Email mà người dùng dùng để đăng ký tài khoản) vào chỗ trống, sau khi điền xong thì click vào button gửi yêu cầu để tiếp tục, nếu nhập sai hoặc bỏ trống thông tin chưa nhập thì ứng dụng sẽ thông báo nhập lại. Ứng dụng sẽ trả người dùng về trang đăng nhập và gửi 1 đường link đến Email của tài khoản mà người dùng đã quên, sau đó người dùng sẽ click vào đường link đó để đến với trang đổi mật khẩu, sau khi đổi xong, người dùng quay trở vào phần đăng nhập để nhập lại email và mật khẩu vừa đổi trước đó.



### 5.3. Màn hình sau khi đăng nhập

- Sau khi đăng nhập, ứng dụng sẽ chuyển hướng người dùng sang trang màn hình chính của ứng dụng (từ trang LoginActivity sang trang Man\_hinh\_chinh) của ứng dụng.
- Tại trang này, người dùng sẽ thấy nó bao gồm các lựa chọn như là: Bài tập, Tự do luyện tập, Bảng cửu chương và Thử thách.
- Ngoài ra, bên góc trái màn hình chính của ứng dụng là phần cài đặt với 3 lựa chọn là thông tin người dùng, giới thiệu ứng dụng và đăng xuất (thoát ra khỏi ứng dụng và tắt ứng dụng).



#### 5.4. Màn hình sau khi click vào lựa chọn Bài tập

- Sau khi click vào lựa chọn Bài tập thì ứng dụng sẽ chuyển hướng người dùng sang trang các chương Quiz (từ trang Man\_hình\_chính sang trang QuizActivity).
- Tại trang này, người dùng sẽ thấy nó bao gồm các mục lựa chọn như Chương bài tập cộng, Chương bài tập trừ, Chương bài tập nhân, Chương bài tập chia và Chương bài tập nâng cao.
- Ngoài ra, có còn có nút “<” quay lại trang trước đó nếu người dùng có chọn chế độ luyện tập khác.



### 5.5. Màn hình sau khi click vào 1 trong 5 chương công, trừ, nhân, chia và nâng cao (chức năng của các chương giống nhau, chỉ khác nhau ở các câu hỏi theo chương)

- Sau khi người dùng chọn được chương bài tập mà mình mong muốn luyện tập thì ứng dụng sẽ chuyển hướng người dùng sang trang chính của Quiz (từ trang QuizActivity sang trang GiaoDienQuiz).

- Tại trang này, người dùng sẽ thấy nó bao gồm phần hiển thị câu hỏi, 4 lựa chọn đáp án và các hiển thị như là hiển thị số đáp án đúng, số đáp án sai và hiển thị tổng số điểm đạt được.



- Nếu chọn sai kết quả của câu hỏi thì ứng dụng sẽ tô màu xanh đáp án đúng của câu và tô đỏ lựa chọn mình vừa chọn, nếu chọn đúng kết quả của câu hỏi thì ứng dụng sẽ tô xanh lựa chọn mình vừa chọn. Làm xong 1 câu ứng dụng sẽ tự động cập nhật phần hiển thị đúng, sai, điểm số.



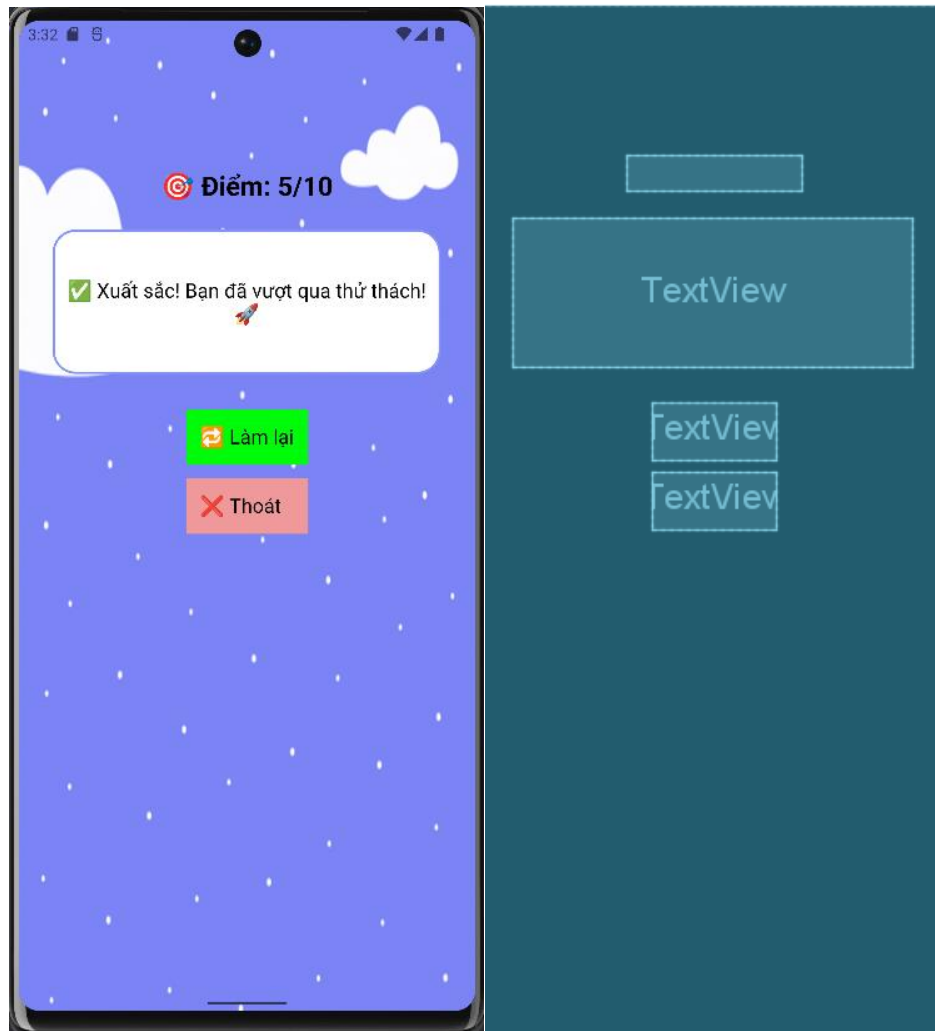
Màn hình hiển thị khi chọn sai kết quả



Màn hình hiển thị khi chọn đúng kết quả

### 5.6. Màn hình sau khi hoàn thành Quiz trong trang GiaoDienQuiz

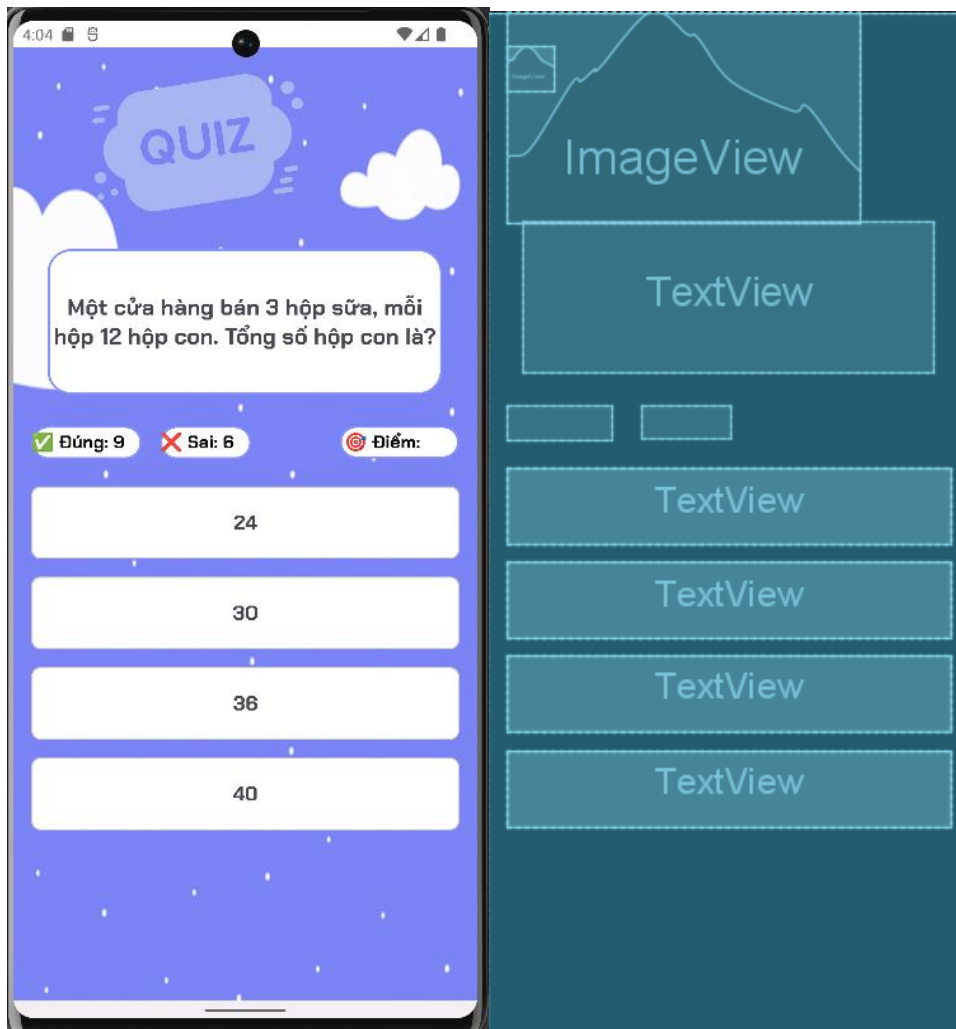
- Sau khi người dùng hoàn thành bài tập gồm 10 câu được lựa chọn ngẫu nhiên trong mã code thì ứng dụng sẽ chuyển hướng người dùng sang trang kết quả (từ trang GiaoDienQuiz sang trang KetQuaActivity).
- Tại trang này, người dùng sẽ thấy tổng điểm mình đạt được (trên thang điểm 10).
- Nếu tổng số điểm  $\geq 5$  thì ứng dụng sẽ chúc mừng bạn đã vượt qua thử thách và nếu tổng số điểm  $< 5$  thì ứng dụng sẽ khích lệ người dùng.
- Ngoài ra, còn có 2 lựa chọn là thoát, trả người dùng về lại trang chọn chương bài tập (QuizActivity) và làm lại (làm lại chương vừa rồi với các câu hỏi ngẫu nhiên).





### 5.7. Màn hình sau khi click vào lựa chọn luyện tập tự do

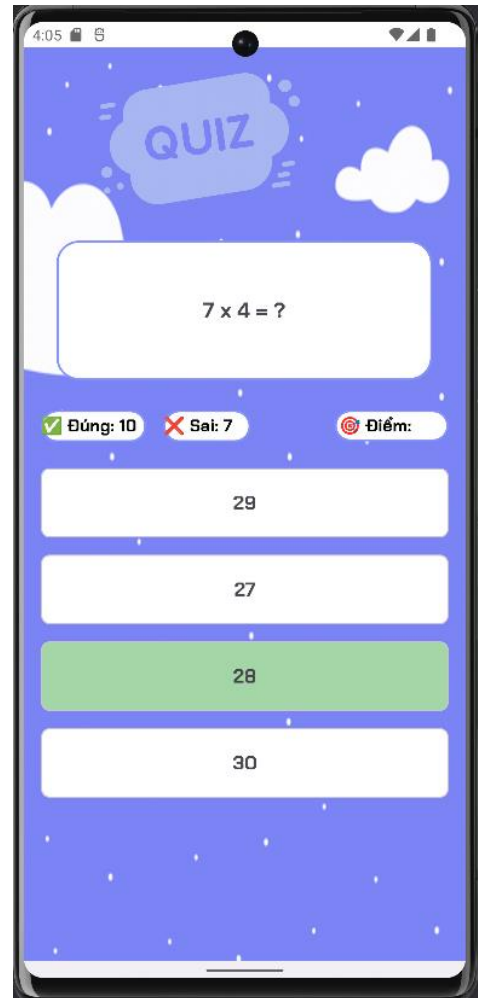
- Sau khi click vào lựa chọn Tự do luyện tập thì ứng dụng sẽ chuyển hướng người dùng sang trang luyện tập các Quiz (từ trang Man\_hinh\_chinh sang trang TuDoLuyenTapActivity).
- Tại trang này, người dùng sẽ thấy nó bao gồm phần hiển thị câu hỏi, 4 lựa chọn đáp án và các hiển thị như là hiển thị số đáp án đúng, số đáp án sai.
- Khác ở phần GiaoDienQuiz là phần TuDoLuyenTapActivity có thể luyện tập đa dạng các câu hỏi, không tính điểm số, tạo cảm giác luyện tập thỏa thích, không tạo áp lực cho các bé khi làm bài.
- Ngoài ra, có còn có nút “<” quay lại trang trước đó nếu người dùng có lựa chọn khác.



- Nếu chọn sai kết quả của câu hỏi thì ứng dụng sẽ tô màu xanh đáp án đúng của câu và tô đỏ lựa chọn mình vừa chọn, nếu chọn đúng kết quả của câu hỏi thì ứng dụng sẽ tô xanh lựa chọn mình vừa chọn. Làm xong 1 câu ứng dụng sẽ tự động cập nhật phần hiển thị đúng, sai.



Màn hình hiển thị khi chọn sai kết quả



Màn hình hiển thị khi chọn đúng kết quả

### 5.8. Màn hình sau khi click vào lựa chọn Bảng cửu chương

- Sau khi click vào lựa chọn Bảng cửu chương thì ứng dụng sẽ chuyển hướng người dùng sang trang bảng cửu chương (từ trang Man\_hinh\_chinh sang trang BangCuuChuongActivity).
- Tại trang này, người dùng sẽ thấy nó bao gồm 1 Spinner chứa số từ 1 đến 10 và 1 button xem bảng sau khi người dùng đã chọn được số mình mong muốn tra cứu bảng cửu chương.
- Sau khi click vào nút button thì ứng dụng sẽ hiển thị ra 4 bảng bao gồm các bảng cửu chương cộng, trừ, nhân, chia phù hợp với số người dùng đã chọn.
- Ngoài ra, có còn có nút “<” quay lại trang trước đó nếu người dùng muốn chọn chương luyện tập khác

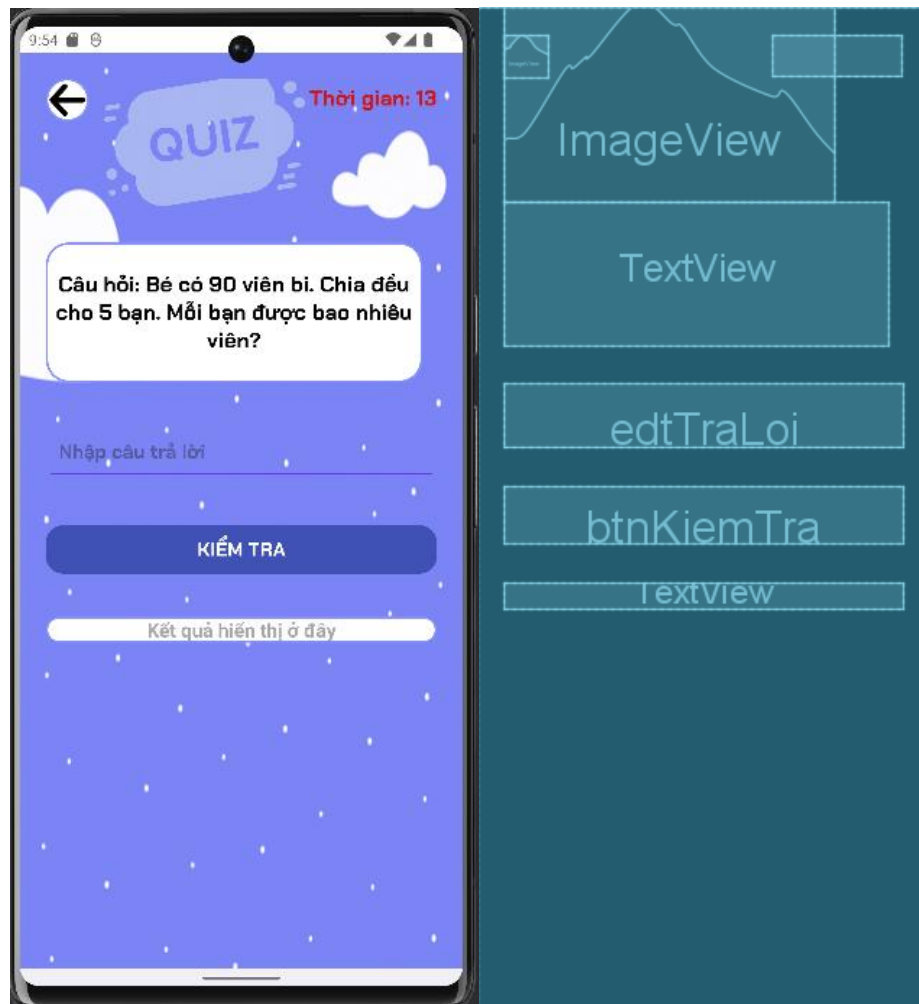


### 5.9. Màn hình sau khi click vào lựa chọn thử thách

- Sau khi click vào lựa chọn Thử thách thì ứng dụng sẽ chuyển hướng người dùng sang trang thử thách (từ trang Man\_hinh\_chinh sang trang ThuThachActivity).

- Tại trang này, người dùng sẽ thấy nó bao gồm phần hiển thị câu hỏi, phần trả lời tự luận (nhập bằng bàn phím), nút kiểm tra dùng để kiểm tra câu trả lời có đúng hay không, phần hiển thị kết quả và thời gian đếm ngược.

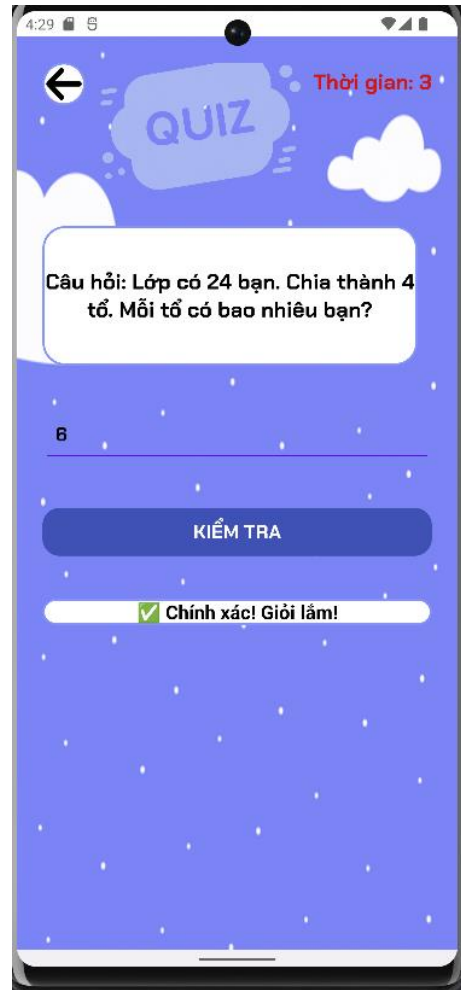
- Ngoài ra, có còn có nút “<-” quay lại trang trước đó nếu người dùng muốn chọn lựa chọn khác.



- Nếu điền sai kết quả của câu hỏi thì ứng dụng sẽ hiển thị ra dòng chữ “sai rồi” kèm đáp án đúng của câu hỏi, nếu điền đúng kết quả của câu hỏi thì ứng dụng sẽ hiển thị ra dòng chữ “Chính xác! Giỏi lắm!”.



Màn hình hiển thị khi nhập sai kết quả



Màn hình hiển thị khi nhập đúng kết quả

### 5.9.1. Màn hình sau khi hết thời gian trong phần Luyện tập

- Sau khi thời gian đếm ngược kết thúc mà người dùng chưa nhập câu trả lời mình vào ô Text thì ứng dụng sẽ hiển thị “Hết giờ!” kèm với đáp án đúng của câu hỏi.



### 5.9.2. Màn hình sau khi hoàn thành 10 câu hỏi trong phần Luyện tập

- Sau khi hoàn thành 10 câu hỏi trong phần luyện tập, ứng dụng sẽ tính điểm cho người dùng trên thang điểm 10, với những câu sai hoặc câu hết thời gian thì sẽ được tính là sai và điểm số sẽ không cộng điểm, còn những câu đúng thì mỗi câu 1 điểm

- Nếu tổng số điểm  $\geq 5$  thì bạn đã đạt và nếu tổng số điểm  $< 5$  thì là bạn chưa đạt và cần cố gắng vào những lần sau.



### 5.10. Màn hình mở mục cài đặt

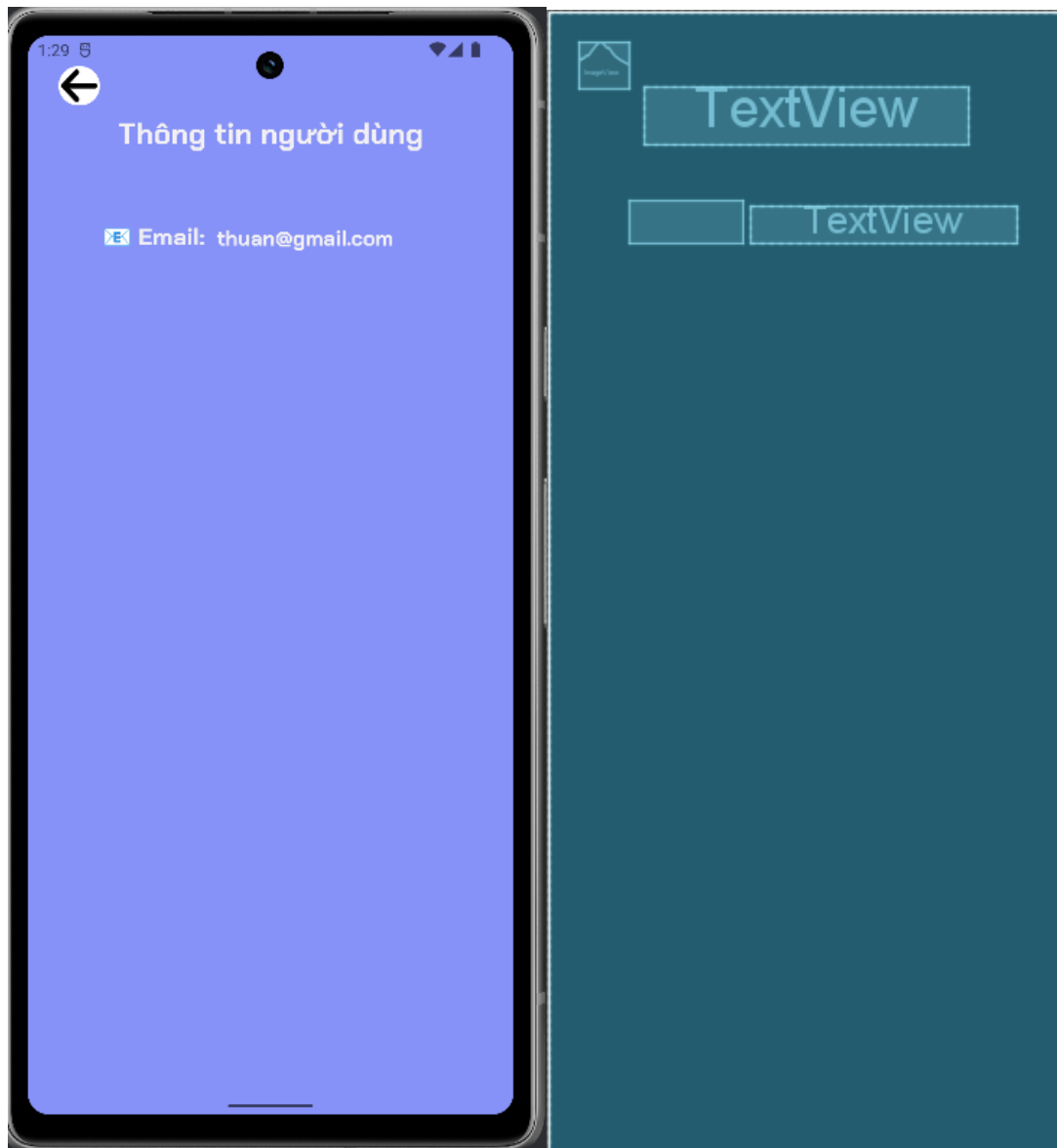
- Sau khi ấn vào biểu tượng cài đặt nó sẽ hiển thị lên 3 lựa chọn là “Thông tin người dùng”, “Giới thiệu ứng dụng” và” Đăng xuất”.





### 5.10.1. Màn hình sau khi nhấn vào lựa chọn “Thông tin người dùng”

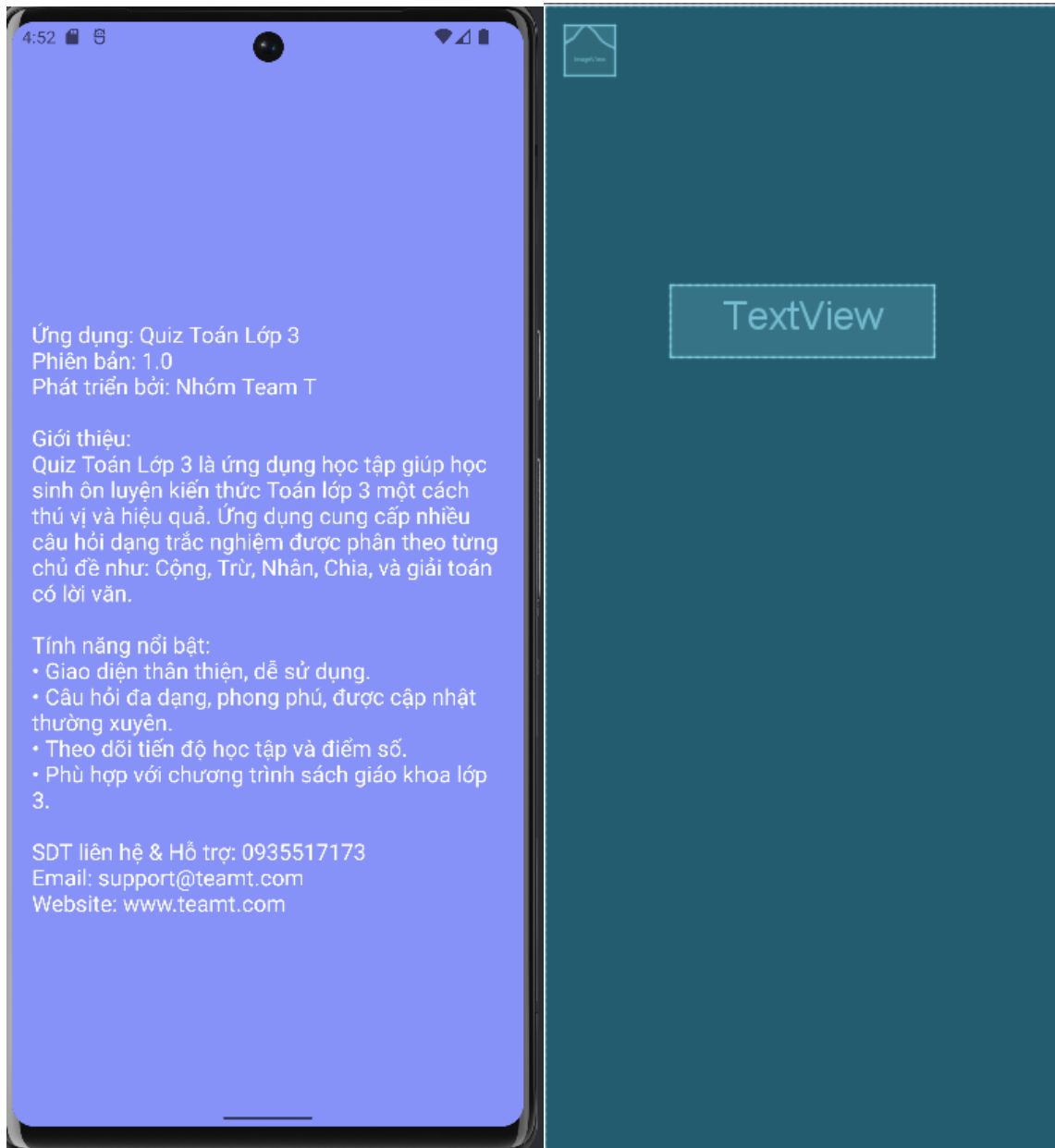
- Sau khi click vào lựa chọn Thông tin người dùng thì ứng dụng sẽ chuyển hướng người dùng sang Thông tin người dùng (từ trang Man\_hinh\_chinh sang trang UserInfoActivity).
- Tại trang này, người dùng sẽ thấy được Email của tài khoản mình đăng nhập.
- Ngoài ra, có còn có nút “<” quay lại trang trước đó nếu người dùng có lựa chọn khác.



### 5.10.2. Màn hình sau khi nhấn vào lựa chọn “Giới thiệu ứng dụng”

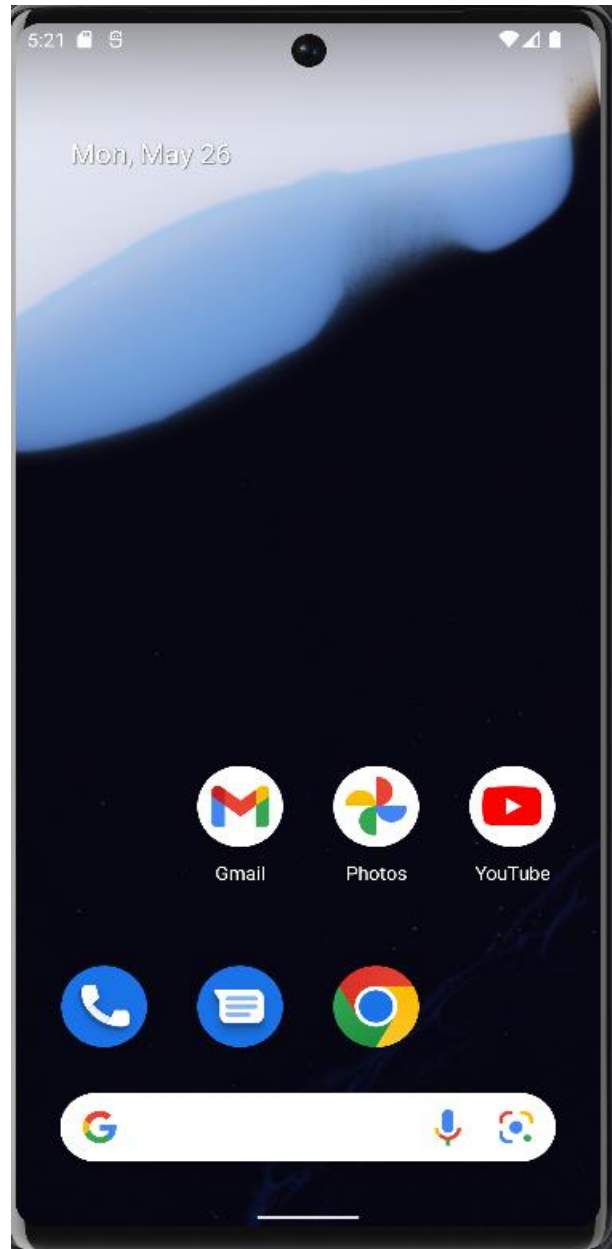
- Sau khi click vào lựa chọn Giới thiệu ứng dụng thì ứng dụng sẽ chuyển hướng người dùng sang giới thiệu ứng dụng (từ trang Man\_hinh\_chinh sang trang AboutActivity).

- Tại trang này, người dùng sẽ thấy nó bao gồm thông tin của ứng dụng cũng như SĐT, Email, Website của admin.



### 5.10.3. Màn hình sau khi nhấn vào lựa chọn “Đăng xuất”

- Sau khi click vào lựa chọn “Đăng xuất” thì ứng dụng sẽ tự động thoát ra ngoài màn hình chính của màn hình (máy tính hoặc điện thoại) và dừng khởi động app.



## PHẦN 6: ĐÁNH GIÁ VÀ KẾT LUẬN

### 6.1. Đánh giá kết quả

Dựa trên các yêu cầu ban đầu của dự án, dự án “Quiz App toán lớp 3” đã hoàn thành được những yêu cầu sau:

- Xây dựng thành công ứng dụng Android hỗ trợ cho học sinh lớp 3 luyện tập theo hình thức tự luận và trắc nghiệm
- Giao diện người dùng hài hòa, bắt mắt, tạo cảm giác vừa học vừa chơi, phù hợp với học sinh tiểu học.
- Các chức năng đăng kí, đăng nhập và quên mật khẩu hoạt động ổn định, được sử dụng thông qua Firebase Authentication.
- Tạo và hiển thị thành công list các câu hỏi, các lựa chọn trả lời và đáp án câu hỏi.
- Hiển thị được điểm số, số câu đúng, câu sai và hiển thị đáp án sau khi trả lời câu hỏi.

Ngoài ra ứng dụng vẫn chưa đáp ứng được yêu cầu lưu lịch sử điểm người dùng sau khi làm xong 1 chương hay 1 bài thử thách hoặc thống kê thời gian bài làm người dùng hoàn thành trong bao nhiêu thời gian. Ứng dụng chưa có chức năng phân loại các câu hỏi theo độ khó khác nhau.

### 6.2. Kết luận

Dự án đã hoàn thành đầy đủ các yêu cầu trọng tâm đề ra, phù hợp với đối tượng học sinh lớp 3. Việc áp dụng Firebase giúp tăng tính ổn định và bảo mật của ứng dụng. Trong tương lai, ứng dụng có thể tiếp tục phát triển với các chức năng mở rộng như:

- Thêm tính năng thống kê điểm theo người dùng.
- Lưu lại lịch sử làm bài và tiến độ học tập.
- Cho phép thêm câu hỏi động từ giáo viên hoặc phụ huynh.
- Hỗ trợ nhiều lớp học hoặc môn học khác.
- Tích hợp các câu hỏi, câu trả lời và các lựa chọn vào firebase.

## PHẦN 7: PHỤ LỤC

### 7.1 Mockups, Wireframes (nếu cần)

### 7.2 Mã nguồn

Link github dự án: [64132410-AndroidProgramming/Project\\_QuizzAppMatchGrade3 at main · VOLENGOCTHUAN/64132410-AndroidProgramming](https://github.com/64132410-AndroidProgramming/Project_QuizzAppMatchGrade3)