**BỘ NÔNG NGHIỆP VÀ MÔI TRƯỜNG**

**TRƯỜNG ĐẠI HỌC THỦY LỢI**

A blue and white logo

AI-generated content may be incorrect.

**BÀI TẬP THỰC HÀNH SỐ 4**

**PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG**

**NỘI DUNG BỔ SUNG: ỨNG DỤNG VỚI CSDL**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã sinh viên** | **Họ và tên** | **Lớp** |
| 1 | 2251061822 | Nguyễn Thị Khánh Linh | 64CNTT2 |

**Hà Nội, năm 2025**

**BÀI TẬP 1: SHARED PREFERENCE**

**Mục tiêu:**

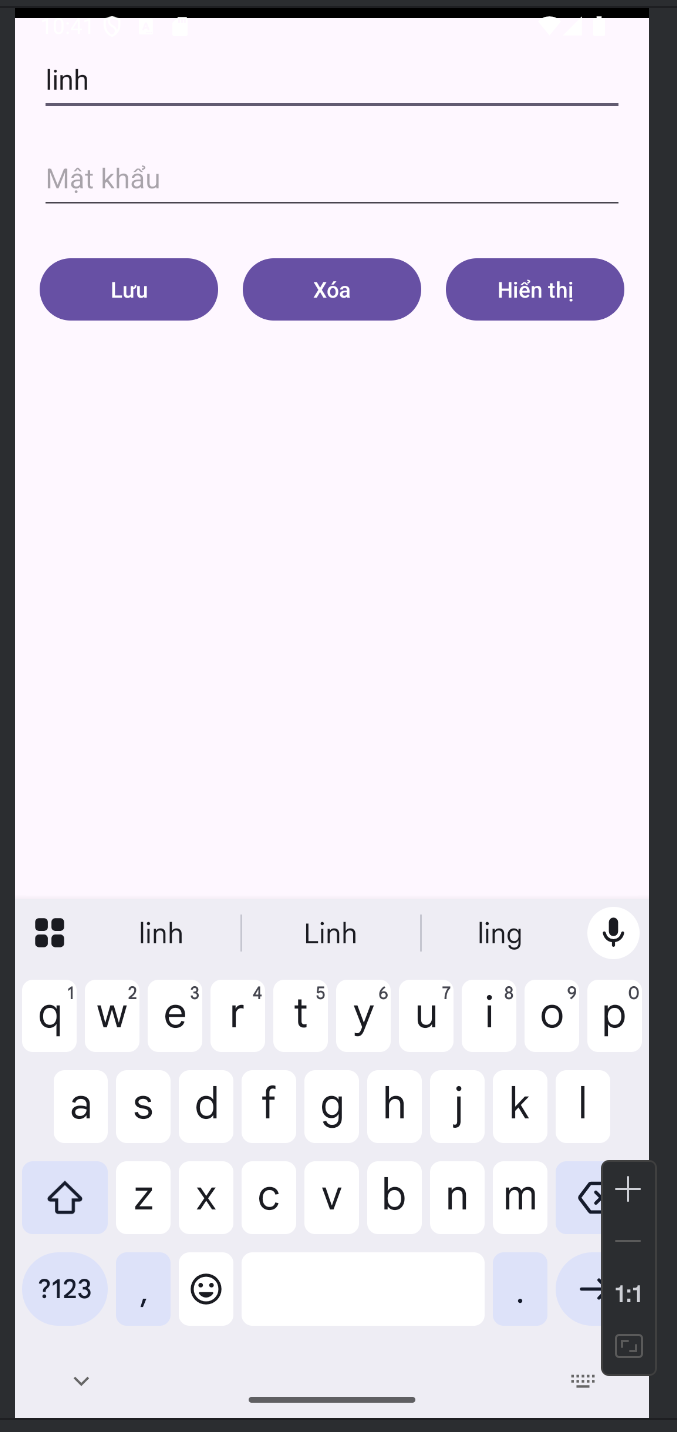
* Hiểu cách sử dụng Shared Preference để lưu trữ dữ liệu cục bộ trong ứng dụng Android.
* Thực hành lưu trữ và đọc dữ liệu từ Shared Preference.

**Yêu cầu:**

1. **Tạo ứng dụng mới:**
   * Tạo một dự án Android mới bằng Kotlin.
   * Thiết kế giao diện người dùng với hai trường nhập (EditText) cho tên người dùng và mật khẩu, và ba nút bấm: "Lưu", "Xóa", và "Hiển thị".
2. **Sử dụng Shared Preference:**
   * Tạo một lớp helper **PreferenceHelper** để quản lý Shared Preference.
   * Khi người dùng nhấn nút "Lưu", lưu tên người dùng và mật khẩu vào Shared Preference.
   * Khi người dùng nhấn nút "Xóa", xóa dữ liệu đã lưu trong Shared Preference.
   * Khi người dùng nhấn nút "Hiển thị", đọc dữ liệu từ Shared Preference và hiển thị lên màn hình.
3. **Thực hành:**
   * Viết mã Kotlin để thực hiện các chức năng trên.
   * Sử dụng getSharedPreferences để truy cập Shared Preference và edit() để lưu dữ liệu.
   * Sử dụng commit() hoặc apply() để lưu thay đổi.
4. **Kết quả**

<<Sinh viên chụp Ảnh màn hình kết quả và mã nguồn chính tại đây>>

**A screen shot of a phone

Description automatically generated  A screenshot of a phone

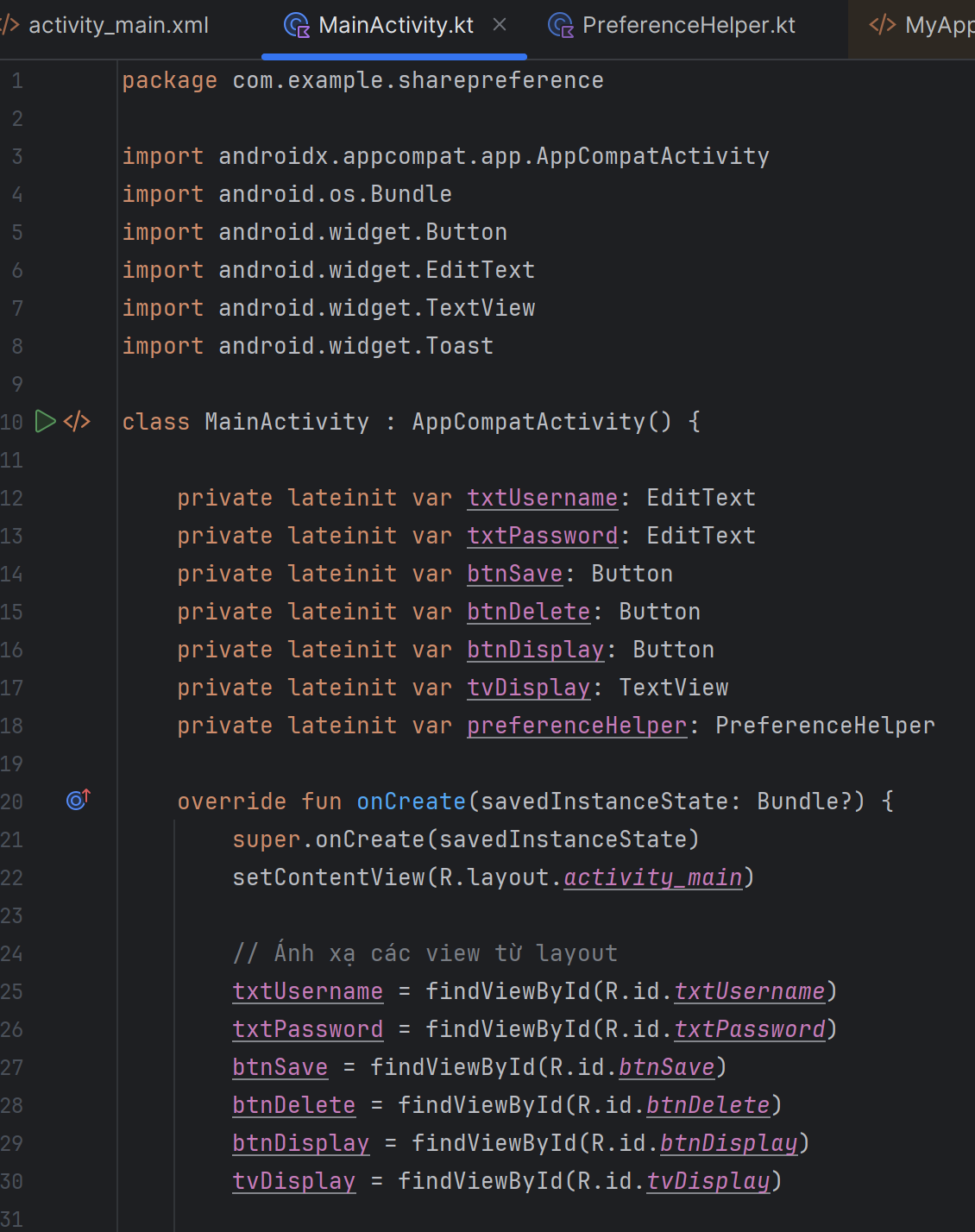
Description automatically generated**

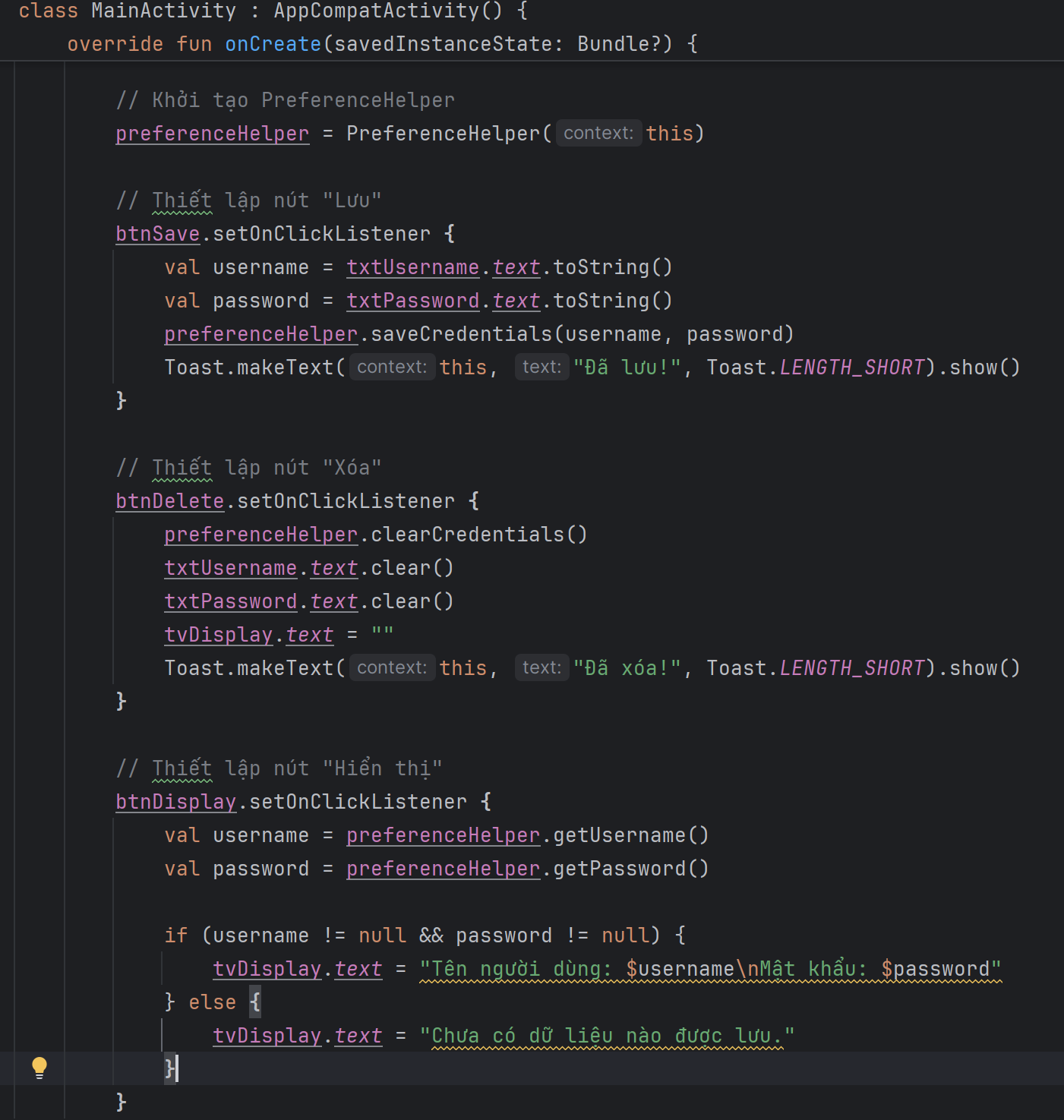
**A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated**

****

****

**BÀI TẬP 2: SQLite**

**Mục tiêu:**

* Hiểu cách sử dụng SQLite để lưu trữ dữ liệu trong ứng dụng Android.
* Thực hành tạo cơ sở dữ liệu SQLite, thêm, sửa, xóa dữ liệu.

**Yêu cầu:**

1. **Tạo ứng dụng mới:**
   * Tạo một dự án Android mới bằng Kotlin.
   * Thiết kế giao diện người dùng với hai trường nhập (EditText) cho tên và số điện thoại, và bốn nút bấm: "Thêm", "Sửa", "Xóa", và "Hiển thị".
2. **Sử dụng SQLite:**
   * Tạo một lớp helper để quản lý cơ sở dữ liệu SQLite.
   * Tạo bảng dữ liệu với hai cột: tên và số điện thoại.
   * Viết các hàm để thêm, sửa, xóa dữ liệu từ cơ sở dữ liệu.
   * Khi người dùng nhấn nút "Hiển thị", đọc dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình.
3. **Thực hành:**
   * Viết mã Kotlin để thực hiện các chức năng trên.
   * Sử dụng SQLiteOpenHelper để tạo và quản lý cơ sở dữ liệu.

**4. Kết quả**

<<Sinh viên chụp Ảnh màn hình kết quả và mã nguồn chính tại đây>>

**A screenshot of a cell phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a chat

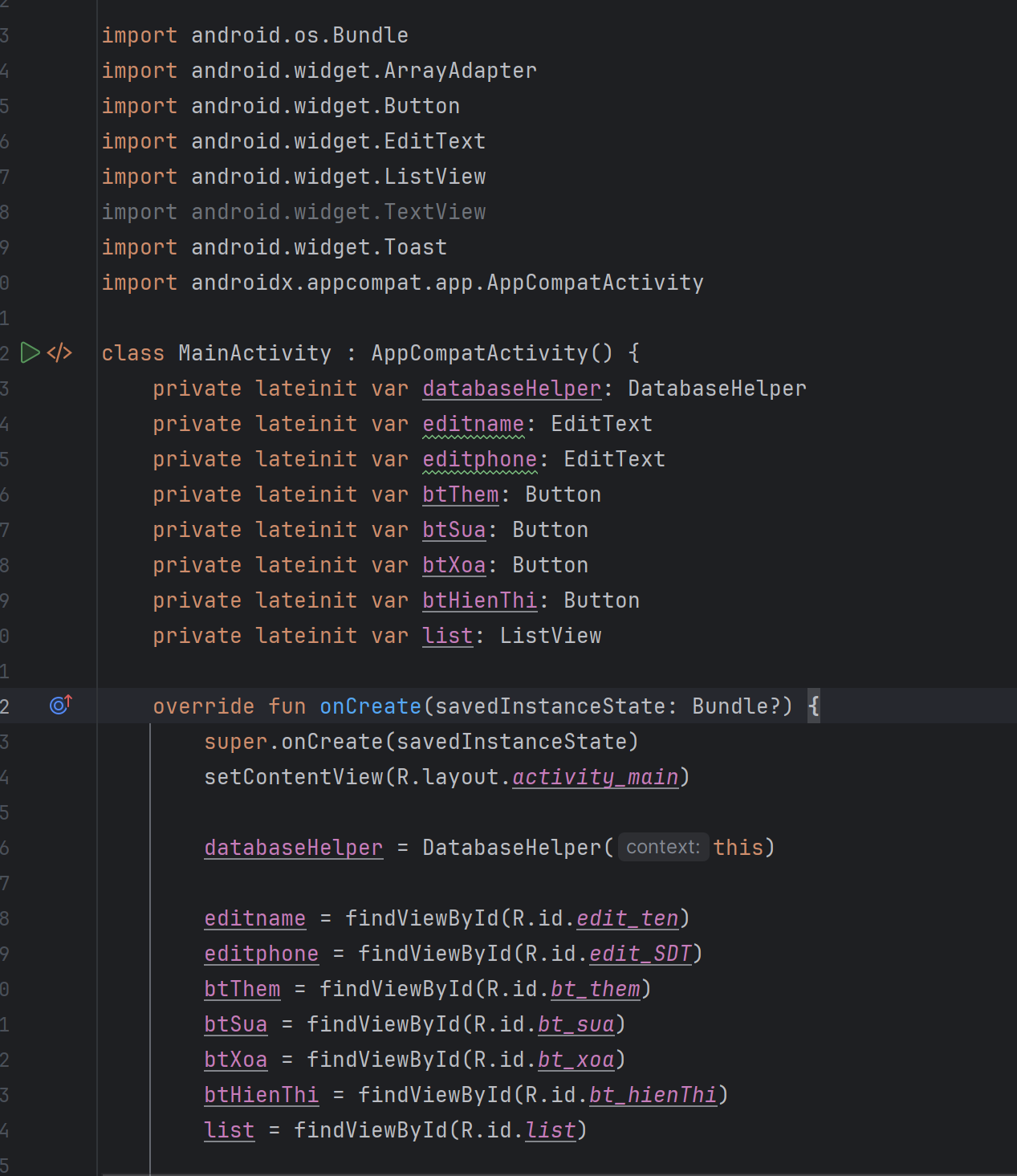
Description automatically generated**

**A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated**

****

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**BÀI TẬP 3: HỆ SINH THÁI FIREBASE**

**Mục tiêu:**

* Hiểu rõ về các dịch vụ chính của Firebase.
* Biết cách tích hợp Firebase vào dự án phát triển ứng dụng.

**Yêu cầu:**

1. **Tìm hiểu các dịch vụ chính của Firebase:**
   * Firebase Authentication: Xác thực người dùng.
   * Firebase Realtime Database và Cloud Firestore: Cơ sở dữ liệu thời gian thực và NoSQL.
   * Firebase Cloud Functions: Chạy mã backend serverless.
   * Firebase Cloud Messaging (FCM): Gửi thông báo đẩy.
   * Firebase Storage: Lưu trữ tệp tin trên đám mây.
   * Firebase Machine Learning (ML): Tích hợp trí tuệ nhân tạo vào ứng dụng.
2. **Viết báo cáo:**
   * Giới thiệu tổng quan về Firebase và lịch sử phát triển.
   * Mô tả chi tiết từng dịch vụ chính của Firebase.
   * Thảo luận về lợi ích và ứng dụng của Firebase trong phát triển ứng dụng.

|  |
| --- |
| **Tổng quan về Firebase:**   * Firebase là một nền tảng phát triển ứng dụng di động và web do Google phát triển. Nó cung cấp một loạt các công cụ và dịch vụ để giúp các nhà phát triển xây dựng, cải thiện và phát triển ứng dụng của họ. * Firebase hoạt động như một Backend-as-a-Service (BaaS), nghĩa là nó cung cấp các dịch vụ backend sẵn sàng sử dụng, giúp các nhà phát triển tập trung vào việc xây dựng giao diện người dùng và logic ứng dụng mà không cần lo lắng về việc quản lý máy chủ hoặc cơ sở dữ liệu. * Các dịch vụ chính của Firebase:   + Firebase Authentication: Xác thực người dùng.   + Firebase Realtime Database và Cloud Firestore: Cơ sở dữ liệu thời gian thực và NoSQL.   + Firebase Cloud Functions: Chạy mã backend serverless.   + Firebase Cloud Messaging (FCM): Gửi thông báo đẩy.   + Firebase Storage: Lưu trữ tệp tin trên đám mây.   + Firebase Machine Learning (ML): Tích hợp trí tuệ nhân tạo vào ứng dụng.   **Lịch sử phát triển của Firebase:**   * Firebase ban đầu được phát triển bởi James Tamplin và Andrew Lee vào năm 2011 với tên gọi "Envolve". Ban đầu, nó là một nền tảng cung cấp các API để tích hợp tính năng chat vào các trang web. * Vào năm 2012, Envolve được đổi tên thành Firebase và ra mắt công khai với Cơ sở dữ liệu thời gian thực (Realtime Database) là sản phẩm chính. * Năm 2014, Google mua lại Firebase và bắt đầu mở rộng nền tảng với nhiều tính năng và dịch vụ mới. * Từ đó đến nay, Firebase đã phát triển mạnh mẽ và trở thành một trong những nền tảng phát triển ứng dụng phổ biến nhất, được sử dụng bởi hàng triệu nhà phát triển trên toàn thế giới.   **Firebase Authentication (Xác thực người dùng)**   * Firebase Authentication cung cấp một hệ thống xác thực người dùng hoàn chỉnh, giúp bạn dễ dàng tích hợp tính năng đăng nhập, đăng ký và quản lý người dùng vào ứng dụng của mình. * Hỗ trợ nhiều phương thức xác thực phổ biến như email/mật khẩu, đăng nhập bằng tài khoản Google, Facebook, Twitter, GitHub, và nhiều nhà cung cấp khác. * Cung cấp các công cụ để quản lý người dùng, đặt lại mật khẩu, và bảo mật tài khoản   **Firebase Realtime Database và Cloud Firestore (Cơ sở dữ liệu thời gian thực và NoSQL)**   * **Firebase Realtime Database:**   + Là một cơ sở dữ liệu NoSQL lưu trữ dữ liệu dưới dạng JSON và đồng bộ hóa dữ liệu thời gian thực giữa các thiết bị.   + Phù hợp cho các ứng dụng yêu cầu cập nhật dữ liệu liên tục và tương tác thời gian thực như ứng dụng trò chuyện, trò chơi trực tuyến. * **Cloud Firestore:**   + Là một cơ sở dữ liệu NoSQL linh hoạt và có khả năng mở rộng, lưu trữ dữ liệu dưới dạng tài liệu và bộ sưu tập.   + Hỗ trợ truy vấn phức tạp và phù hợp cho các ứng dụng có cấu trúc dữ liệu phức tạp.   + Cả 2 loại cơ sở dữ liệu này đều giúp việc lưu trữ, truy xuất dữ liệu trở nên nhanh và tiện lợi hơn.   **Firebase Cloud Functions (Chạy mã backend serverless)**   * Firebase Cloud Functions cho phép bạn chạy mã backend trên đám mây mà không cần quản lý máy chủ. * Mã được kích hoạt bởi các sự kiện từ Firebase hoặc các dịch vụ Google Cloud khác. * Phù hợp cho các tác vụ backend như xử lý dữ liệu, gửi thông báo, và tích hợp với các API bên ngoài.   **Firebase Cloud Messaging (FCM) (Gửi thông báo đẩy)**   * Firebase Cloud Messaging (FCM) là một dịch vụ nhắn tin đa nền tảng cho phép bạn gửi thông báo đẩy đến người dùng trên Android, iOS và web. * Hỗ trợ gửi thông báo đơn lẻ hoặc theo nhóm, và có thể tùy chỉnh nội dung thông báo. * Giúp người dùng không bỏ lỡ những thông tin quan trọng từ ứng dụng.   **Firebase Storage (Lưu trữ tệp tin trên đám mây)**   * Firebase Storage cho phép bạn lưu trữ và truy xuất các tệp tin như hình ảnh, video và tài liệu trên đám mây. * Tích hợp chặt chẽ với Firebase Authentication và Cloud Firestore, giúp bạn kiểm soát quyền truy cập tệp tin. * Việc lưu trữ trên đám mây giúp giải phóng dung lượng cho thiết bị của người dùng.   **Firebase Machine Learning (ML) (Tích hợp trí tuệ nhân tạo vào ứng dụng)**   * Firebase Machine Learning (ML) cung cấp các công cụ để tích hợp trí tuệ nhân tạo (AI) vào ứng dụng của bạn. * Hỗ trợ các mô hình ML tùy chỉnh và các API ML sẵn có của Google. * Phù hợp cho các ứng dụng yêu cầu phân tích hình ảnh, xử lý ngôn ngữ tự nhiên, và các tác vụ ML khác.   **Lợi ích của Firebase:**   * **Tăng tốc độ phát triển:**   + Firebase cung cấp các dịch vụ backend sẵn có, giúp nhà phát triển tập trung vào việc xây dựng giao diện người dùng và logic ứng dụng mà không cần lo lắng về việc quản lý máy chủ.   + Việc sử dụng các dịch vụ như Realtime Database và Cloud Firestore giúp đồng bộ hóa dữ liệu thời gian thực, giảm thiểu thời gian phát triển các tính năng tương tác. * **Tiết kiệm chi phí:**   + Mô hình serverless của Firebase giúp giảm chi phí quản lý máy chủ và cơ sở hạ tầng.   + Nhiều dịch vụ của Firebase có gói miễn phí, phù hợp cho các dự án nhỏ và giai đoạn thử nghiệm. * **Khả năng mở rộng:**   + Firebase được xây dựng trên cơ sở hạ tầng của Google, đảm bảo khả năng mở rộng và hiệu suất cao.   + Các dịch vụ như Cloud Functions và Cloud Firestore có khả năng tự động mở rộng để đáp ứng nhu cầu tăng trưởng của ứng dụng. * **Tăng cường tương tác người dùng:**   + Firebase Cloud Messaging (FCM) giúp gửi thông báo đẩy hiệu quả, tăng cường tương tác và giữ chân người dùng.   + Realtime Database và Cloud Firestore cho phép xây dựng các ứng dụng tương tác thời gian thực, mang lại trải nghiệm người dùng tốt hơn. * **Phân tích và tối ưu hóa:**   + Firebase Analytics cung cấp thông tin chi tiết về hành vi người dùng, giúp nhà phát triển hiểu rõ hơn về cách người dùng tương tác với ứng dụng.   + Dựa trên dữ liệu phân tích, nhà phát triển có thể tối ưu hóa ứng dụng để cải thiện hiệu suất và trải nghiệm người dùng.   **Ứng dụng của Firebase:**   * **Ứng dụng trò chuyện và mạng xã hội:**   + Realtime Database và Cloud Firestore cho phép xây dựng các tính năng trò chuyện thời gian thực, chia sẻ nội dung và tương tác giữa người dùng.   + Firebase Authentication giúp quản lý người dùng và bảo mật tài khoản. * **Ứng dụng thương mại điện tử:**   + Firebase Storage giúp lưu trữ hình ảnh và video sản phẩm.   + Cloud Functions giúp xử lý các tác vụ backend như thanh toán và quản lý đơn hàng.   + Firebase Analytics giúp theo dõi hành vi khách hàng và tối ưu hóa trải nghiệm mua sắm. * **Ứng dụng trò chơi:**   + Realtime Database cho phép đồng bộ hóa dữ liệu trò chơi thời gian thực, tạo ra trải nghiệm chơi game trực tuyến mượt mà.   + Cloud Functions giúp xử lý các logic trò chơi trên máy chủ. * **Ứng dụng giáo dục:**   + Firebase Storage giúp lưu trữ tài liệu và video học tập.   + Firebase Cloud Messaging giúp gửi thông báo về các khóa học và sự kiện.   + Firebase Authentication giúp quản lý tài khoản học viên. * **Ứng dụng doanh nghiệp:**   + Firebase có thể được dùng để xây dựng các ứng dụng nội bộ doanh nghiệp.   + Giúp việc quản lý dữ liệu, và tương tác giữa các nhân viên trong công ty. |

1. **Thực hành:**
   * Tạo một dự án Firebase mới trên Firebase Console.
   * Đăng ký ứng dụng Android vào dự án Firebase.
   * Sử dụng ít nhất hai dịch vụ của Firebase trong dự án (ví dụ: Authentication và Realtime Database).

**Bài tập cụ thể: Tích hợp Firebase Authentication và Realtime Database**

**Yêu cầu:**

1. **Tạo ứng dụng mới:**
   * Tạo một dự án Android mới bằng Kotlin.
   * Thiết kế giao diện người dùng với hai trường nhập (EditText) cho email và mật khẩu, và ba nút bấm: "Đăng ký", "Đăng nhập", và "Hiển thị dữ liệu".
2. **Tích hợp Firebase Authentication:**
   * Sử dụng Firebase Authentication để cho phép người dùng đăng ký và đăng nhập bằng email và mật khẩu.
   * Viết mã để xử lý các sự kiện đăng ký và đăng nhập thành công hoặc thất bại.
3. **Tích hợp Firebase Realtime Database:**
   * Sau khi người dùng đăng nhập thành công, lưu trữ thông tin người dùng vào Firebase Realtime Database.
   * Khi người dùng nhấn nút "Hiển thị dữ liệu", đọc dữ liệu từ Firebase Realtime Database và hiển thị lên màn hình.
4. **Kết quả**

<<Sinh viên chụp Ảnh màn hình kết quả và mã nguồn chính tại đây>>

