**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**PBL4: DỰ ÁN HỆ ĐIỀU HÀNH & MẠNG MÁY TÍNH**

**Đề tài 103: Xây dựng ứng dụng BKZalo**

**cho phép người dùng có thể trò chuyện trên di động**

|  |  |
| --- | --- |
| **GIẢNG VIÊN HƯỚNG DẪN:** | **Ths.Nguyễn Thế Xuân Ly** |
| **DOANH NGHIỆP HƯỚNG DẪN: Trung tâm điều độ điện lực miền Trung** | |
| **KS. Nguyễn Thế Hiển** | **KS. Đinh Văn Huy** |
| **SINH VIÊN THỰC HIỆN:** | |
| 1. **Trần Tấn Đạt** | **MSSV: 102190206** |
| 1. **Phan Thị Thu Sương** | **MSSV: 102190237** |
| **LỚP SH: 19TCLC-DT5** | **NHÓM HP: 19Nh13B** |

**Đà Nẵng 12/ 2021**

MỤC LỤC

[1](#_Toc91193779)

[MỤC LỤC 2](#_Toc91193780)

[DANH SÁCH HÌNH VẼ 3](#_Toc91193781)

[MỞ ĐẦU 4](#_Toc91193782)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 5](#_Toc91193783)

[1.1. Hệ điều hành 5](#_Toc91193784)

[1.1.1. Khái niệm 5](#_Toc91193785)

[1.1.2. Sơ lược về hệ điều hành Android 5](#_Toc91193786)

[1.1.3. Tìm hiểu về Firebase 6](#_Toc91193787)

[1.2. Mô hình Client-Server 8](#_Toc91193788)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 10](#_Toc91193789)

[2.1.1. Sơ đồ Usercase 10](#_Toc91193790)

[2.2. Các chức năng chỉnh của ứng dụng 10](#_Toc91193791)

[2.2.1. Đăng kí 10](#_Toc91193792)

[2.2.2. Đăng nhập 12](#_Toc91193793)

[2.2.3. Trò chuyện 13](#_Toc91193794)

[2.3. Sơ đồ lớp 14](#_Toc91193795)

[2.3.1. Cơ sở dữ liệu 14](#_Toc91193796)

[CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 16](#_Toc91193797)

[3.1. Yêu cầu hệ thống 16](#_Toc91193798)

[3.2. Đánh giá kết quả 16](#_Toc91193799)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 16](#_Toc91193800)

[TÀI LIỆU THAM KHẢO 17](#_Toc91193801)

DANH SÁCH HÌNH VẼ

[Hình 1: Mô hình Client-Server 11](#_Toc91193666)

[Hình 2: Sơ đồ usercase 12](#_Toc91193667)

[Hình 3: Giao diện Đăng kí 12](#_Toc91193668)

[Hình 4: Sơ đồ Activity-Đăng kí 13](#_Toc91193669)

[Hình 5: Sơ đồ tuần tự-Đăng kí 13](#_Toc91193670)

[Hình 6: Giao diện Đăng nhập 14](#_Toc91193671)

[Hình 7: Sơ đồ Activity-Đăng nhập 14](#_Toc91193672)

[Hình 8: Sơ đồ tuần tự-Đăng nhập 15](#_Toc91193673)

[Hình 9: Giao diện Chat 15](#_Toc91193674)

[Hình 10: Sơ đồ Activity-Chat 16](#_Toc91193675)

[Hình 11: Sơ đồ tuần tự-Chat 16](#_Toc91193676)

[Hình 12: Authentication 17](#_Toc91193677)

[Hình 13: Realtime Database 17](#_Toc91193678)

MỞ ĐẦU

Ngày nay, thiết bị di động rất phổ biến và đóng vai trò quan trọng trong cuộc sống của con người. Điện thoại di động giúp kết nối với mọi người trên khắp mọi nơi, là công cụ hỗ trợ đắc lực cho con người trong mọi phương diện của cuộc sống. Tiêu biểu phải kể đến vai trò quan trọng là giúp con người có thể kết nối với nhau, trò chuyện và chia sẻ thông tin. Vì vậy, các ứng dụng có khả năng cho phép người dùng trò chuyện, nhắn tin rất phổ biến và được sử dụng rộng rãi.

Xuất phát từ nhu cầu kết nối giữa con người với nhau trong xã hội, chúng em đã tìm hiểu và phát triển một ứng dụng BKZalo cho phép người dùng trò chuyện, nhắn tin trên thiết bị di động.

Nhóm chúng em cũng xin gửi lời cảm ơn đến thầy Nguyễn Thế Xuân Ly đã hỗ trợ và giúp đỡ chúng em hoàn thành đề tài của nhóm.

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## Hệ điều hành

### Khái niệm

* **Hệ điều hành (Operating System-OS):** là một phần mềm dùng để điều hành, quản lý toàn bộ tất cả thành phần (bao gồm cả phần cứng và phần mềm) của thiết bị điện tử. Có vai trò trung gian trong việc giao tiếp giữa người sử dụng và thiết bị.
* Hệ điều hành dành cho máy tính:
* Được thiết kế để sử dụng trên những thiết bị như: Máy tính để bàn, laptop,… Đây là hệ điều hành sử dụng GUI.
* GUI cho phép bạn sử dụng chuột để nhấp vào mọi thứ trên màn hình hiển thị (bao gồm các biểu tượng , nút, menu,…) và được diễn tả bằng cách kết hợp hình ảnh cùng với văn bản.
* Bao gồm các hệ điều hành phổ biến như: Windows, Linux, macOS
* Hệ điều hành dành cho thiết bị di động:
* Được thiết kế dành cho nhứng thiết bị như: điện thoại di động, máy tính bảng,… Được thiết kế với nhu cầu giải trí, liên lạc,…
* Bao gồm các hệ điều hành phổ biến như: Android, iOS.

### Sơ lược về hệ điều hành Android

* Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng.….
* Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp,sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo giãn và thu lại để xử lý các đối tượng trên màn hình. Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng.
* Android có lượng ứng dụng của bên thứ ba ngày càng nhiều, được chọn lọc và đặt trên một cửa hàng ứng dụng như Google Play hay Amazon Appstore để người dùng lấy về, hoặc bằng cách tải xuống rồi cài đặt tập tin ''APK'' từ trang web khác
* Các ứng dụng cho Android được phát triển bằng ngôn ngữ Java sử dụng Bộ phát triển phần mềm Android (SDK). SDK bao gồm một bộ đầy đủ các công cụ dùng để phát triển, gồm có công cụ gỡ lỗi, thư viện phần mềm, bộ giả lập điện thoại dựa trên QEMU, tài liệu hướng dẫn, mã nguồn mẫu, và hướng dẫn từng bước.

### Tìm hiểu về Firebase

* Firebase là một nền tảng để phát triển ứng dụng di động và trang web, bao gồm các API đơn giản và mạnh mẽ mà không cần backend hay server.
* Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu..

**Những dịch vụ nổi bật của Firebase:**

* Dịch vụ Firebase Cloud Messaging (FCM) là một giải pháp nhắn tin đa nền tảng, nó cho phép bạn gửi tin nhắn một cách đáng tin cậy mà không mất phí.
* Sử dụng FCM bạn có thể gửi thông báo cho ứng dụng khi có email mới hoặc có dữ liệu mới để thực hiện đồng bộ hóa. Với các trường hợp sử dụng nhắn tin real time thì tin nhắn có thể có lưu lượng lên đến 4KB.
* Một FCM implement hai thành phần chính để gửi và nhận: 1. Một môi trường đáng tin cậy như Cloud Function cho Firebase hoặc một app server để build và gửi tin nhắn. 2. Một client app như IOS, Android hoặc Web(JavaScrip) để nhận tin nhắn qua các service dành riêng cho từng nền tảng.
* Dịch vụ Cloud Firestore
* Cloud Firestore là cơ sở dữ liệu NoSQL được lưu trữ trên nền tảng đám mây mà các ứng dụng web, IOS, Android có thể truy cập trực tiếp thông qua SDK native. Cloud Firestore cũng được dùng trong Node.js, Java, Python và Go SDKs. Theo mô hình dữ liệu NoSQL của Cloud Firestore, bạn lưu trữ dữ liệu trong documents, nó chứa các trường mà có thể map tới các giá trị. Những documents này lại được lưu trữ trong các collections. Documents hỗ trợ nhiều kiểu data khác nhau,. Bạn cũng có thể tạo subcollections trong các documents và xây dựng các cấu trúc dữ liệu phân cấp mở rộng khi cơ sở dữ liệu của bạn phát triển. Cloud Firestore data model hỗ trợ bất kỳ cấu trúc dữ liệu nào hoạt động tốt nhất cho app của bạn.
* Truy vấn trong Cloud Firestore là linh hoạt và hiệu quả
* Dịch vụ Firebase Analytics
* Google Analytics giúp bạn hiểu cách người dùng sử dụng các web, iOS hay Android app. SDK sẽ tự động ghi lại một số sự kiện(events) và thuộc tính người dùng (user properties) và cũng cho phép bạn định nghĩa các custom events để đo lường những sự kiện quan trọng liên quan tới business. Khi dữ liệu được ghi lại, nó được lưu trong dashboard trên Firebase console.
* Analytics cũng tích hợp với một số tính năng khác của Firebase.
* Dịch vụ Firebase Remote Config
* Remote Config bao gồm thư viện client xử lý các tác vụ quan trọng như fetching parameter values và lưu vào bộ đệm, trong khi vẫn cho phép bạn kiểm soát khi các giá trị mới được kích hoạt để không làm ảnh hưởng đến trải nghiệm người dùng.
* Các phương thức get() cung cấp quyền truy cập duy nhất đến các giá trị tham số.
* Dịch vụ Firebase Crashlytics: là một công cụ báo cáo lightweight, realtime. Nó giúp bạn theo dõi và phát hiện sửa các lỗi làm ảnh hưởng đến chất lượng app của bạn.
* Dịch vụ Firebase Dynamic Links: Bạn tạo một Dynamic Links dùng Firebase console, dùng một REST API, iOS hoặc Android Builder API, hoặc tự tạo bằng cách thêm các tham số vào các miền cụ thể theo format. Các tham số này chỉ định các link bạn muốn mở, tùy thuộc vào nền tảng của người dùng và liệu ứng dụng của bạn có được cài đặt hay không.

**Realtime Database**

* Truy cập Cơ sở dữ liệu thời gian thực Firebase là cơ sở dữ liệu NoQuery được lưu trữ trên đám mây cho phép bạn lưu trữ và đồng bộ hóa giữa những người dùng của bạn trong thời gian thực.
* Cơ sở dữ liệu thời gian thực thực sự chỉ là một đối tượng JSON lớn mà các developer có thể quản lý trong thời gian thực.
* Chỉ với một API duy nhất, cơ sở dữ liệu Firebase cung cấp cho ứng dụng của bạn cả giá trị hiện tại của dữ liệu và mọi cập nhật cho dữ liệu đó.
* Đồng bộ hóa thời gian thực giúp người dùng của bạn dễ dàng truy cập dữ liệu của họ từ bất kỳ thiết bị nào, có thể là web hoặc di động. Cơ sở dữ liệu thời gian thực cũng giúp người dùng của bạn hợp tác, trao đổi với nhau.
* Một lợi ích đáng kinh ngạc khác của Cơ sở dữ liệu thời gian thực là nó cung cấp SDK di động và web, cho phép bạn xây dựng ứng dụng của mình mà không cần máy chủ.
* Khi người dùng của bạn ngoại tuyến, SDK cơ sở dữ liệu thời gian thực sử dụng bộ đệm cục bộ trên thiết bị để phục vụ và lưu trữ các thay đổi. Khi thiết bị trực tuyến, dữ liệu local sẽ được tự động đồng bộ hóa.
* Cơ sở dữ liệu thời gian thực cũng có thể tích hợp với Firebase Authentication để cung cấp quy trình xác thực đơn giản và trực quan.

**Authentication**

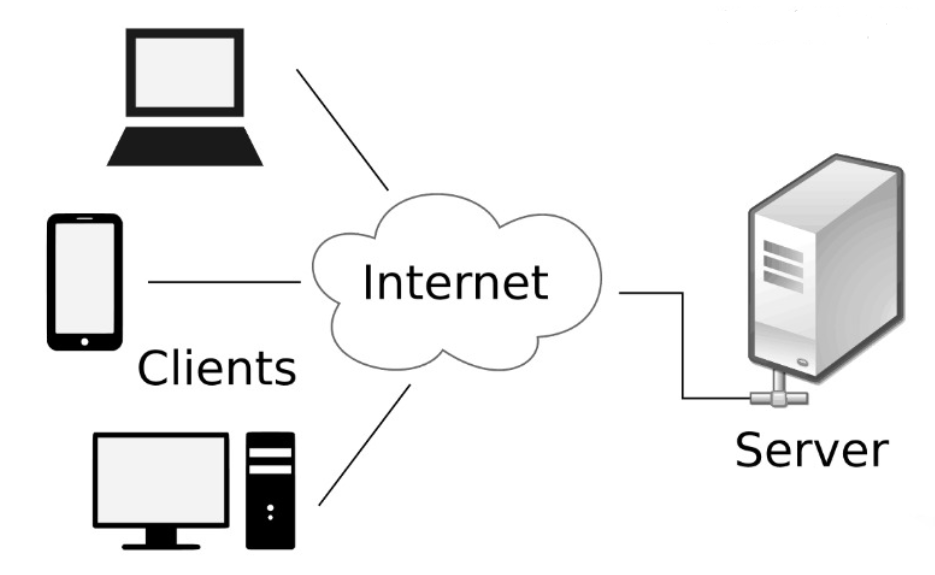
* Xác thực Firebase cung cấp dịch vụ backend, SDK dễ sử dụng và thư viện UI được tạo sẵn để xác thực người dùng với ứng dụng của bạn.
* Bạn có thể xác thực người dùng ứng dụng của bạn thông qua các phương pháp sau:
* Email & Password
* Số điện thoại
* Google
* Facebook
* Twitter
* Và hơn nữa
* Sử dụng Xác thực Firebase giúp xây dựng các hệ thống xác thực an toàn và dễ dàng hơn, đồng thời cải thiện trải nghiệm đăng nhập cho người dùng.
* Firebase Authentication được xây dựng bởi cùng những người đã tạo Đăng nhập Google, Smart Lock và Trình quản lý mật khẩu Chrome.

**Truy cập Firebase Data**

* Dữ liệu chứa trong Firebase Location có thể được truy cập theo nhiều cách khác nhau:
* Firebase client: sử dụng dữ liệu firebase trong ứng dụng của bạn được xây dựng từ các thư viện hỗ trợ của firebase gồm: web, ios, java và nodejs
* REST API: Firebase cung cấp full tính năng REST API, sử dụng mọi ngôn ngữ bạn có quyền đọc, viết, sửa, thêm, xóa dữ liệu firebase của mình trên mọi thiết bị thông qua giao thức HTTPS.
* Web Browser: Cách đơn giản nhất, bạn truy cập firebase data bằng cách mở địa chỉ firebase của bạn trên trình duyệt. Tại đây, có thể tương tác trực tiếp vào dữ liệu với giao diện chuyên nghiệp.

## Mô hình Client-Server

* Client server là mô hình mạng máy tính bao gồm 2 thành phần chính là máy khách (client) và máy chủ (server). Trong mô hình này, server là nơi lưu trữ tài nguyên, cài đặt các chương trình dịch vụ và thực hiện các yêu cầu của client. Client đón vai trò gửi yêu cầu đến server. Client gồm máy tính và thiết bị điện tử nói chung.



Hình 1: Mô hình Client-Server

* Client (Máy khách)

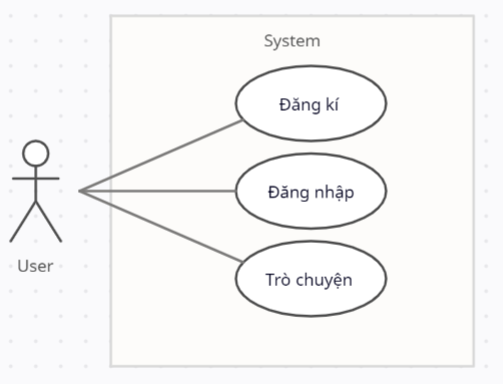
Client hay chính là máy khách, máy trạm – là nơi gửi yêu cầu đến server. Nó tổ chức giao tiếp với người dùng, server và môi trường bên ngoài tại trạm làm việc. Client tiếp nhận yêu cầu của người dùng sau đó thành lập các query string để gửi cho server. Khi nhận được kết quả từ server, client sẽ tổ chức và trình diễn những kết quả đó.

* Server (Máy chủ)

Server xử lý yêu cầu gửi đến từ client. Sau khi xử lý xong, server sẽ gửi trả lại kết quả, client có thể tiếp tục xử lý các kết quả này để phục vụ người dùng. Server giao tiếp với môi trường bên ngoài và client tại server, tiếp nhận yêu cầu dưới dạng query string (xâu ký tự). Khi phân tích xong các xâu ký tự, server sẽ xử lý dữ liệu và gửi kết quả về cho client.

# CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### Sơ đồ Usercase



Hình 2: Sơ đồ usercase

## Các chức năng chỉnh của ứng dụng

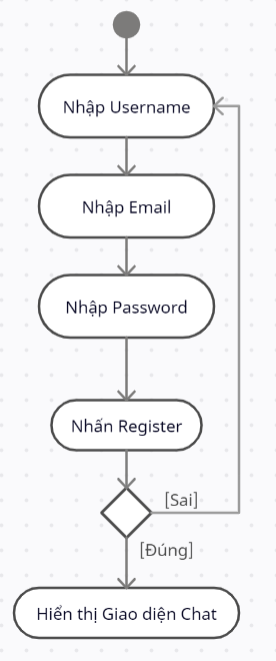
### Đăng kí

Graphical user interface, application, Teams

Description automatically generated

Hình 3: Giao diện Đăng kí

* Sơ đồ Activity



Hình 4: Sơ đồ Activity-Đăng kí

* Sơ đồ tuần tự

Diagram

Description automatically generated

Hình 5: Sơ đồ tuần tự-Đăng kí

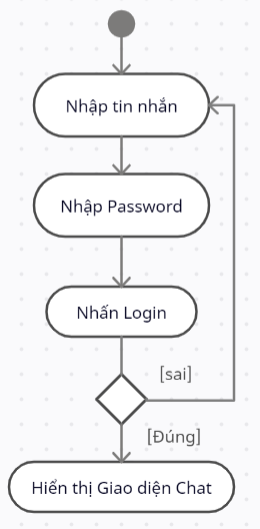
### Đăng nhập

A screenshot of a phone

Description automatically generated with medium confidence

Hình 6: Giao diện Đăng nhập

* Sơ đồ Activity



Hình 7: Sơ đồ Activity-Đăng nhập

* Sơ đồ tuần tự

Diagram

Description automatically generated

Hình 8: Sơ đồ tuần tự-Đăng nhập

### Trò chuyện

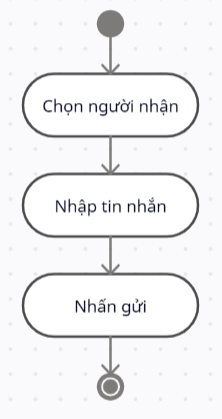
A screenshot of a phone

Description automatically generated with low confidenceA screenshot of a phone

Description automatically generated with medium confidence

Hình 9: Giao diện Chat

* Sơ đồ Activity



Hình 10: Sơ đồ Activity-Chat

* Sơ đồ tuần tự

Diagram

Description automatically generated

Hình 11: Sơ đồ tuần tự-Chat

## Sơ đồ lớp

### Cơ sở dữ liệu

* Sử dụngdịch vụ Authentication và Realtime Database của Firebase
* Authentication

A screenshot of a computer

Description automatically generated

Hình 12: Authentication

* Realtime Database
* Chats
* ressage
* receiver
* sender
* Users
* id
* imageURL
* username

Graphical user interface, text, application, email

Description automatically generated

Hình 13: Realtime Database

# CHƯƠNG 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

## Yêu cầu hệ thống

* Hệ thống chạy được trên các nền tảng Android.
* Các IDE, Text Editor phát triển: Android Studio
* Ngôn ngữ lập trình: Java
* Công cụ quản lý: Github
* Cơ sở dữ liệu : sử dụng Realtime Database Firebase (cơ sở dữ liệu NoQuery) để lưu trữ dữ liệu, dữ liệu được lưu trữ trên đám mây cho phép lưu trữ và đồng bộ hóa giữa những người dùng trong thời gian thực.
* Firebase cung cấp dịch vụ backend, bao gồm các API đơn giản và mạnh mẽ mà không cần backend hay server.
* Người lập trình chỉ lập trình phía Client

## Đánh giá kết quả

* Ứng dụng có giao diện thân thiện dễ sử dụng
* Các chức năng đăng kí, đăng nhập, trò chuyện hoạt động tốt và dễ dàng
* Tốc độ xử lí tưởng đối nhanh
* Tuy nhiên ứng dụng vẫn còn hạn chế và ít chức năng

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

* **Kết luận**
* Ứng dụng cho phép người dùng đăng kí tài khoản chat và trò chuyện với nhau
* Tuy nhiên ứng dụng vẫn còn đơn giản , chưa đầy đủ các chức năng cần thiết và còn hạn chế nhiều mặt
* **Hướng phát triển:**
* Hoàn hiện các chức năng khác như: gửi tin nhắn nhóm, gửi ảnh, gửi file dữ liệu, xác thực email, số điện thoại, …
* Ứng dụng rộng rãi vào thực tế.

TÀI LIỆU THAM KHẢO

[1] Client Server Architecture

<https://cio-wiki.org/wiki/Client_Server_Architecture>

[2] Cách tạo một Ứng dụng Chat trong Android bằng Firebase, <https://code.tutsplus.com/vi/tutorials/how-to-create-an-android-chat-app-using-firebase--cms-27397>

[3] Android

<https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)>

[4] Tổng quan về Firebase trong Android

<https://vncoder.vn/bai-viet/android-tong-quan-ve-firebase-trong-android>