**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

****

**BÁO CÁO ĐỒ ÁN 1**

**ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG U-SCHOOL TRÊN THIẾT BỊ DI ĐỘNG**

**Sinh viên thực hiện: - Bùi Thành Long – 16110139**

**- Vũ Tấn Đạt– 16110055**

**Giáo viên hướng dẫn: Trương Thị Ngọc Phượng**

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

Ngày.........Tháng..........Năm.........

**Giáo Viên Hướng Dẫn**

*(Ký, ghi rõ họ và tên)*

MỤC LỤC

[CHƯƠNG 1: MỞ ĐẦU 4](#_Toc528901815)

[1.1. Giới thiệu đề tài 4](#_Toc528901816)

[1.2. Mục tiêu đề tài 4](#_Toc528901817)

[1.3. Công cụ triển khai đề tài 4](#_Toc528901818)

[CHƯƠNG 2: XÁC ĐỊNH YÊU CẦU VÀ MÔ TẢ CHỨC NĂNG 5](#_Toc528901819)

[1. Xác định yêu cầu 5](#_Toc528901825)

[2. Mô tả chức năng 5](#_Toc528901826)

[CHƯƠNG 3: THIẾT KẾ CƠ SỞ DỮ LIỆU 14](#_Toc528901827)

[1. Liệt kê các quan hệ 14](#_Toc528901828)

[2. Biểu diễn quan hệ dưới dạng bảng (diagrams) 16](#_Toc528901829)

[3. Thiết kế code 17](#_Toc528901830)

[3.1. Môi trường lập trình 17](#_Toc528901831)

[3.2. Chi tiết 18](#_Toc528901832)

[CHƯƠNG 4: KẾT LUẬN 34](#_Toc528901833)

[1. Kết luận 34](#_Toc528901834)

[2. Một số khó khăn gặp phải 34](#_Toc528901835)

[TÀI LIỆU THAM KHẢO 35](#_Toc528901836)

DANH MỤC HÌNH ẢNH

[Hình 1: Giao diện đăng kí tài khoản 5](#_Toc528899515)

[Hình 2: Giao diện nhập thông tin và đăng nhập 6](#_Toc528899516)

[Hình 3: Giao diện bài đăng 7](#_Toc528899517)

[Hình 4: Giao diện Photo Albums 8](#_Toc528899518)

[Hình 5: Giao diện đăng bài 9](#_Toc528899519)

[Hình 6: Giao diện tin nhắn 10](#_Toc528899520)

[Hình 7: Camera từ ứng dụng 11](#_Toc528899521)

[Hình 8: Giao diện hoạt động từng khung giờ 12](#_Toc528899522)

[Hình 9: Giao diện cập nhật thông tin muốn thay đổi 13](#_Toc528899523)

[Hình 10: Giao diện chọn chức năng đăng xuất 14](#_Toc528899524)

[Hình 11: Hình ảnh kết nối các quan hệ của cơ sở dữ liệu 17](#_Toc528899525)

DANH MỤC BẢNG

[Bảng 1: Các lớp sử dụng trong chương trình 18](#_Toc528898022)

[Bảng 2: Các phương thức của lớp Chat 19](#_Toc528898023)

[Bảng 3: Các phương thức của lớp Comment 20](#_Toc528898024)

[Bảng 4: Các phương thức của lớp Post 21](#_Toc528898025)

[Bảng 5: Các phương thức trong lớp PostId 23](#_Toc528898026)

[Bảng 6: Các phương thức trong lớp Schedual 24](#_Toc528898027)

[Bảng 7: Các phương thức trong lớp User 25](#_Toc528898028)

[Bảng 8: Các phương thức của lớp KiemTra 26](#_Toc528898029)

[Bảng 9: Các phương thức của lớp SupportImage 27](#_Toc528898030)

[Bảng 10: Các phương thức của lớp UserAdapter 27](#_Toc528898031)

[Bảng 11: Các phương thức của lớp SchedualAdapter 28](#_Toc528898032)

[Bảng 12: Các phương thức của lớp PostAdapter 29](#_Toc528898033)

[Bảng 13: Các phương thức của lớp CommentAdapter 31](#_Toc528898034)

# CHƯƠNG 1: MỞ ĐẦU

## Giới thiệu đề tài

* Hiện nay, với sự phát triển của kinh tế xã hội thì mỗi người đều phải bỏ nhiều thời gian để lao động kiếm thu nhập lo toan cho cuộc sống đầy đủ, chính vì vậy mà thời gian dành cho gia đình sẽ ít hơn. Riêng đối với trẻ em, các em rất cần sự quan tâm của người lớn, cần sự quan tâm của cha mẹ nhiều hơn nhưng họ lại quá tất bật nên khả năng nắm được những hoạt động của con cái trong ngày là rất ít. Chính vì vậy, theo dõi con trẻ qua ứng dụng di động mọi lúc mọi nơi, không bên cạnh nhưng vẫn biết khi con đi nhà trẻ làm những gì, hoạt động những gì đều có thể nắm được.
* Ứng dụng này giúp cho nhà trường và gia đình hỗ trợ lẫn nhau trong việc chăm sóc trẻ, chỉ cần 1 chiếc smartphone chạy hệ điều hành android thì các phụ huynh hoàn toàn có thể biết con mình đang làm gì trên nhà trẻ.
* Với ưu điểm là dễ dàng sử dụng, không quá yêu cầu cao về cấu hình của thiết bị di động, tiện lợi cho người sử dụng nên nhóm quyết định xây dựng ứng dụng mang tên U-School để giúp cha mẹ làm được những điều đó một cách dễ dàng.

## Mục tiêu đề tài

* Sử dụng hệ điều hành Android để viết ứng dụng trên di động.
* Phân tích, thiết kế và xây dựng ứng dụng hỗ trợ phụ huynh theo dõi hoạt động của trẻ trên trường
* Tìm hiểu kiến thức môn Lập trình di động.

## Công cụ triển khai đề tài

* Dịch vụ server Firebase và một số thư viện khác.
* Ngôn ngữ lập trình Java.
* Hệ điều hành Android.

# CHƯƠNG 2: XÁC ĐỊNH YÊU CẦU VÀ MÔ TẢ CHỨC NĂNG

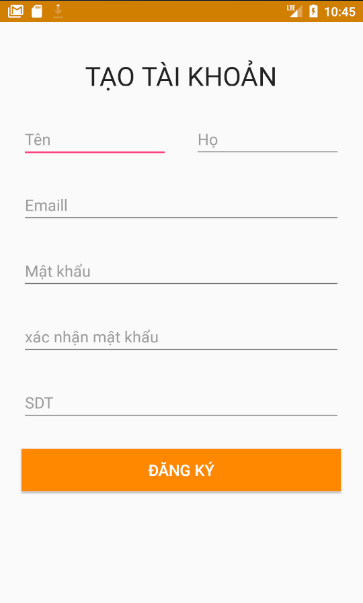


## Xác định yêu cầu

* Ứng dụng cung cấp các công cụ để người dùng có thể sử dụng một cách đơn giản, nhanh chóng.
* Ứng dụng cho phép người dùng đăng ký – đăng nhập tài khoản, xem các bài đăng của giáo viên phụ trách, liên hệ với giáo viên bằng cách nhắn tin trực tiếp từ ứng dụng và một số tính năng hữu ích khác.
* Thành viên sử dụng là phụ huynh và giáo viên: yêu cầu bắt buộc phải đăng kí tài khoản mới sử dụng được các chức năng khác.

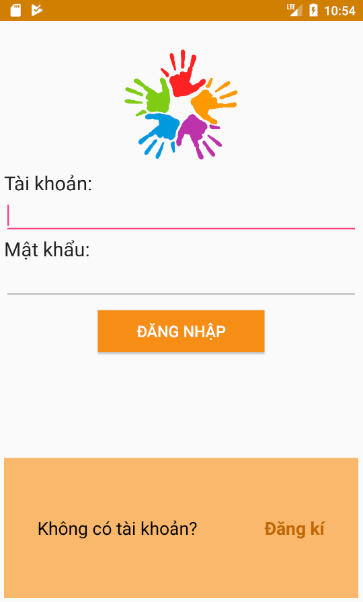
1. ***Mô tả chức năng***

* Đăng kí tài khoản: Để sử dụng ứng dụng thì phụ huynh và giáo viên phải đăng ký tài khoản thành viên để sử dụng các chức năng của ứng dụng. (hình 1)



Hình 1: Giao diện đăng kí tài khoản

* Đăng nhập: Sau khi đăng ký tài khoản xong thì sử dụng account vừa đăng ký để đăng nhập (sign in) và sử dụng ứng dụng. (hình 2)



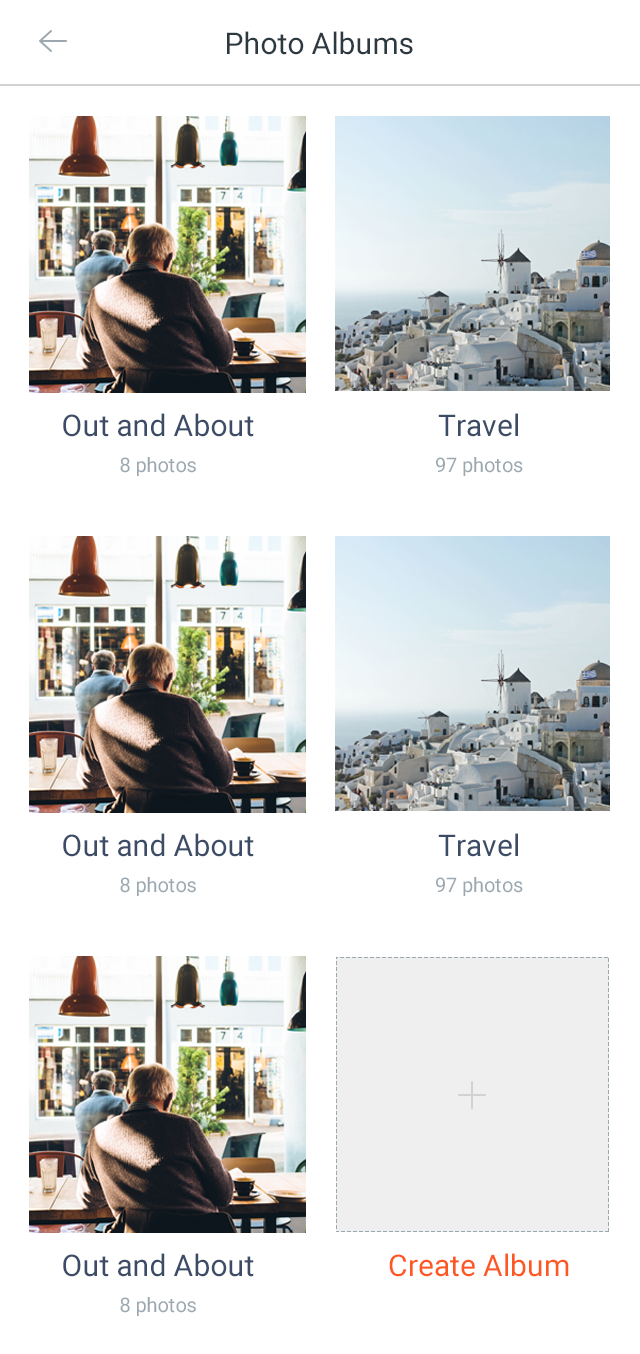
Hình 2: Giao diện nhập thông tin và đăng nhập

* Bài đăng hoạt động của trẻ: Khi giáo viên đăng bài thì sẽ hiển thị lên giao diện như hình 3 kèm theo ngày đăng. Phụ huynh có thể theo dõi hoạt động của các bé trên lớp, có thể “thích” bài viêt hoặc bình luận bài đăng này (hình 3)



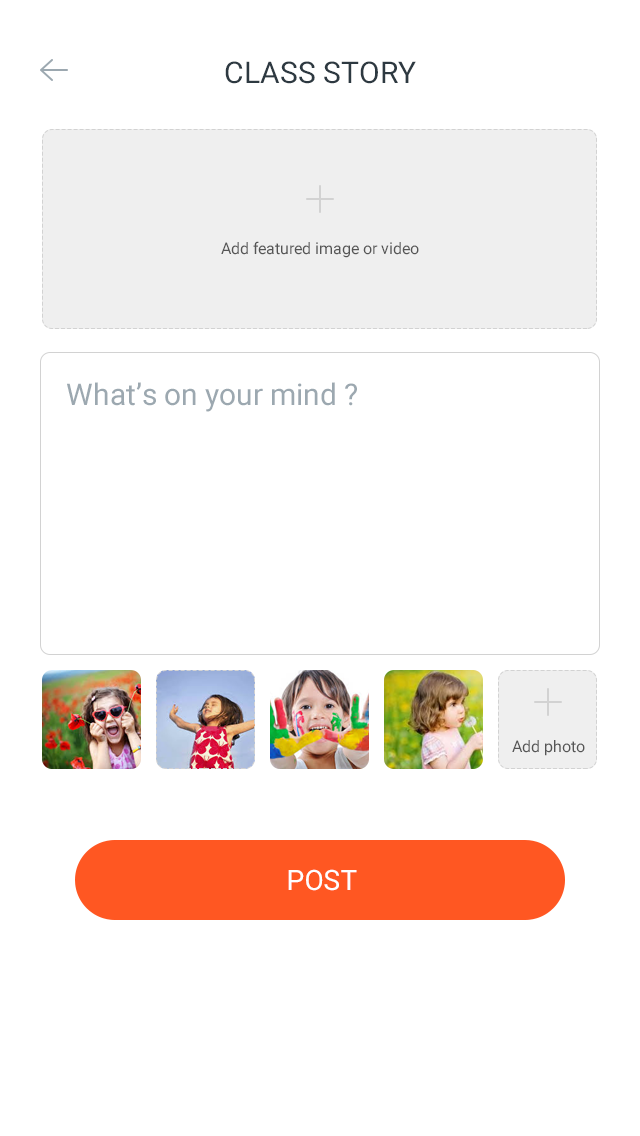
Hình 3: Giao diện bài đăng

* Photo Albums: Kết nối với Albums ảnh từ thiết bị đăng nhập với ứng dụng. Khi đã kết nối thành công thì ứng dụng có thể sử dụng tất cả những hình ảnh của thiết bị. (hình 4)



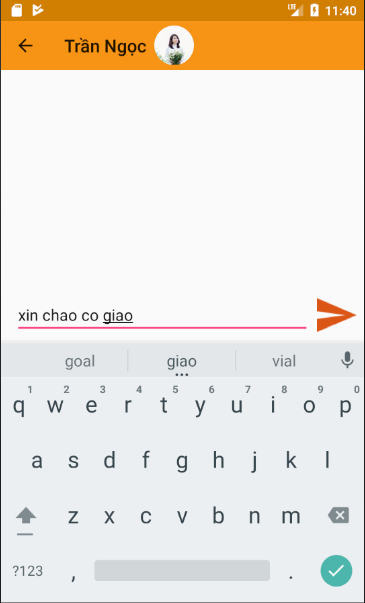
Hình 4: Giao diện Photo Albums

* Đăng bài: Tại giao diện này người dùng có thể nhấn vào (Add featured image or video) đăng tải hình ảnh hoặc video ngắn của trẻ và đi kèm là những nội dung (What’s on your mind) hoạt động trong ngày của trẻ giúp cho phụ huynh có thể nắm được những hoạt động trong ngày của trẻ. Và những nội dung được đăng tải lên thì những người dùng khác có thể bày tỏ cảm xúc và bình luận về nội dung được đăng tải. (hình 5)



Hình 5: Giao diện đăng bài

* Nhắn tin: Công cụ hỗ trợ giao tiếp giữa phụ huynh và giáo viên trên ứng dụng. Tại giao diện này người dùng có thể gửi tin nhắn, xóa tin nhắn, thu hồi tin nhắn khi lỡ gửi tin nhắn nhưng nội dung không phù hợp hoặc sai, xem thời gian của tin nhắn đã gửi và có thể gửi tin nhắn dưới dạng hình ảnh. (hình 6)



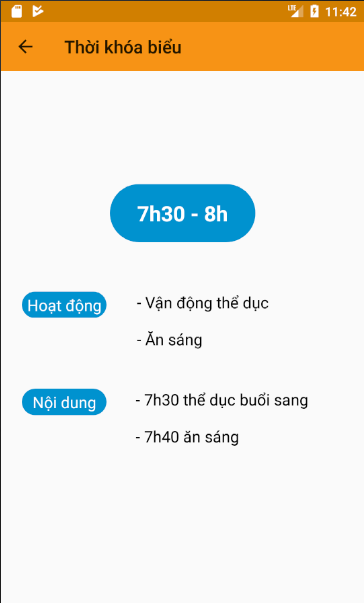
Hình 6: Giao diện tin nhắn

* Camera: Sử dụng camera của thiết bị để chụp ảnh và ảnh sẽ được lưu vào album ảnh, ảnh sẽ được lưu ở albums của thiết bị và trong thư mục của ứng dụng. (hình 7)



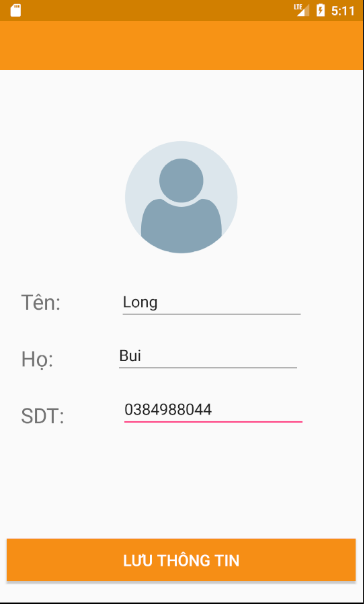
Hình : Camera từ ứng dụng

* Schedual: Thời khóa biểu của trẻ sẽ được hiển thị ở đây để phụ huynh có thể nắm được thời gian cụ thể của từng việc trẻ sẽ thực hiện trên lớp. Muốn xem khung giờ tiếp theo chỉ cần vuốt màn hình từ phải sang trái. (hình 8)



Hình 8: Giao diện hoạt động từng khung giờ

* Settings: Nơi người dùng cài đặt thay đổi liên quan đến thông tin của tài khoản như tên, họ và số điện thoại…ngoài ra có thể cập nhật ảnh đại diện cho tài khoản, ảnh đại diện sẽ hiển thị lên cho mọi tài khoản khác thấy. (hình 9)



Hình : Giao diện cập nhật thông tin muốn thay đổi

* Đăng xuất: khi người dùng muốn kết thúc phiên làm việc hoặc đổi tài khoản thì sẽ dùng chức năng này. (hình 10)



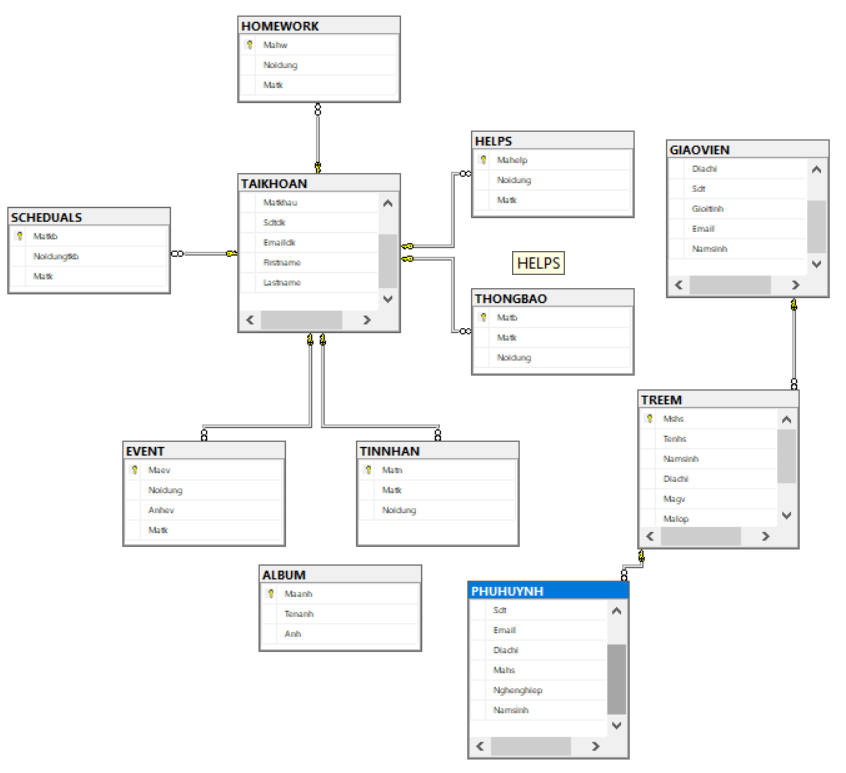
Hình : Giao diện chọn chức năng đăng xuất

# CHƯƠNG 3: THIẾT KẾ CƠ SỞ DỮ LIỆU

1. ***Liệt kê các quan hệ***

* TAIKHOAN:
  + Matk: mã số của từng tài khoản khi đã được đăng kí
  + Tentk: tên tài khoản đăng nhập
  + Matkhau: mật khẩu của tài khoản đăng nhập
  + Emaildk: email dùng để đăng kí tài khoản
  + Sdtdk: số điện thoại dùng để đăng kí tài khoản
  + Name: tên hiển thị trong ứng dụng
* EVENT:
  + Maev: mã số của bài đăng sau khi được đăng tải
  + Noidung: nội dung (phần chữ) khi đăng bài
  + Anhev: ảnh đi kèm (nếu có) của bài đăng
  + Matk: mã tài khoản đăng bài
* SCHEDUAL:
  + Matkb: mã số của thời khóa biểu được đăng lên bởi giáo viên
  + Noidungtkb: nội dung thời khóa biểu
  + Matk: mã tài khoản đưa thời khóa biểu lên ứng dụng
* HOMEWORK:
  + Mahw: mã số của phần bài tập được giáo viên đăng tải
  + Noidung: nội dung phần bài tập
  + Matk: mã tài khoản đăng bài tập
* ALBUM:
  + Maanh: mã số của hình ảnh trong album
  + Tenanh: tên của hình ảnh
  + Anh: hình ảnh
* TINNHAN:
  + Matn: mã số tin nhắn
  + Matk: mã số tài khoản gửi tin nhắn
  + Noidung: nội dung tin nhắn gửi đi
* THONGBAO:
  + Matb: mã số của thông báo
  + Matk: mã tài khoản gửi thông báo
  + Noidung: nội dung thông báo
* HELPS:
  + Mahelp: mã số của phần trợ giúp
  + Noidung: nội dung của phần trợ giúp để người dùng làm theo
  + Matk: mã tài khoản người cập nhật hướng dẫn
* GIAOVIEN:
  + Magv: mã số giáo viên
  + Tengv: họ tên của giáo viên
  + Diachi: địa chỉ nơi ở hiện tại của giáo viên
  + Sdt: số điện thoại của giáo viên
  + Gioitinh: giới tính ghi trong chứng minh nhân dân của giáo viên
  + Email: email liên lạc của giáo viên
  + Namsinh: ngày tháng năm sinh của giáo viên
* TREEM:
  + Mahs: mã số học sinh
  + Tenhs: họ tên của học sinh
  + Namsinh: ngày tháng năm sinh của học sinh
  + Diachi: nơi ở hiện tại của học sinh
  + Magv: mã giáo viên phụ trách quản lí học sinh
  + Gioitinh: giới tính của học sinh
* PHUHUYNH:
  + Maph: mã số phụ huynh
  + Tenph: họ và tên phụ huynh
  + Sdt: số điện thoại liên lạc của phụ huynh
  + Email: email liên lạc của phụ huynh
  + Diachi: địa chỉ nơi ở hiện tại của phụ huynh
  + Mahs: mã số học sinh mà phụ huynh phụ trách nuôi dạy
  + Nghenghiep: nghề nghiệp của phụ huynh
  + Namsinh: ngày tháng năm sinh của phụ huynh

1. ***Biểu diễn quan hệ dưới dạng bảng (diagrams)***

******

Hình 11: Hình ảnh kết nối các quan hệ của cơ sở dữ liệu

1. ***Thiết kế code***
   1. **Môi trường lập trình**
   * Hệ điều hành: Window 10 version 1809
   * Java: JDK 8
   * Nền tảng lập trình: tập trung Android API 26
   1. **Chi tiết**

Bảng 1: Các lớp sử dụng trong chương trình

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Mục đích chính |
| 1 | ChatAdapter | Tạo chức năng nhắn tin trong ứng dụng |
| 2 | CommentAdapter | Tạo chức năng bình luận bài đăng |
| 3 | PostAdapter | Tạo chức năng đăng bài |
| 4 | SchedualAdapter | Tạo chức năng hiển thị thời khoá biểu |
| 5 | UserAdapter | Tạo chức năng tài khoản để dùng ứng dụng |
| 6 | Chat | Khai báo và định dạng các thuộc tính dùng trong nhắn tin |
| 7 | Comment | Khai báo và định dạng các thuộc tính dùng trong bình luận |
| 8 | PostId | Tạo Id bài đăng |
| 9 | Post | Là lớp thừa kế từ lớp cha( PostId) khai báo và định dạng các thuộc tính dùng để đăng bài |
| 10 | Schedual | Khai báo và định dạng các thuộc tính dùng để đăng thời khoá biểu |
| 11 | User | Là lớp thừa kế từ lớp cha( PostId) khai báo và định dạng các thuộc tính dùng để tạo ra một tài khoản hoàn chỉnh |
| 12 | KiemTra | Kiểm tra dữ liệu đầu vào khi đăng kí một tài khoản có hợp lệ hay không |
| 13 | SupportImage | Hỗ trợ định dạng hiển thị hình ảnh của toàn bộ ứng dụng |

Bảng 2: Các phương thức của lớp Chat

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public Chat(String message, String userID, Date timestamp)  Input: message, userID, timestamp  Output: message, userID, timestamp  Pseudo code:  this.message = message;  this.userID = userID;  this.timestamp = timestamp; | Phương thức mặc định để hoàn thành kiểu khai báo lớp Chat | Chat.java (13) |
| 2 | public String getMessage() và  public void setMessage(String message)  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói dữ liệu thuộc tính message của lớp Chat | Chat.java (19 – 23) |
| 3 | public String getUserID() và  public void setUserID(String userID)  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói dữ liệu thuộc tính userID của lớp Chat | Chat.java (27 – 31) |

Bảng 3: Các phương thức của lớp Comment

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public Comment(String message, String userId, Date timestamp, String username, String imgUser)  Input: message, userId, username, imgUser  Output: message, userId, username, imgUser  Pseudo code: không có vì đơn giản | Phương thức mặc định để hoàn thành khai báo lớp Comment | Comment.java (15) |
| 2 | public String getMessage()  public String getUserId()  public Date getTimestamp()  public String getUsername()  public String getImgUser()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính Message, UserId, Timestamp, Username,  ImgUser trong lớp Comment | Comment.java (23 – 31 – 39 – 47 – 55) |
| 3 | public void setMessage(String message)  public void setUserId(String userId)  public void setTimestamp(Date timestamp)  public void setUsername(String username)  public void setImgUser(String imgUser)  Input: Message, UserId, Timestamp, Username,  ImgUser  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính Message, UserId, Timestamp, Username,  ImgUser trong lớp Comment | Comment.java (27 – 35 – 43 – 51 – 59) |

Bảng 4: Các phương thức của lớp Post

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public Post(String image, String imageCompress, String description, Date timestamp, String userID, String username, String imgUser)  Input: image, imageCompress, description, timestamp, userID, username, imgUser  Output: image, imageCompress, description, timestamp, userID, username, imgUser  Pseudo code: không có vì đơn giản | Phương thức mặc định để hoàn thành khai báo lớp Post | Post.java (18) |
| 2 | public String getImage()  public String getImageCompress()  public String getDescription()  public Date getTimestamp()  public String getUserID()  public String getUsername()  public String getImgUser()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính Image, ImageCompress, Description, Timdestamp, UserID, Username, ImgUser của lớp Post | Post.java (28 – 36 – 44 – 52 – 60 – 68 – 76) |
| 3 | public void setImage(String image)  public void setImageCompress(String imageCompress)  public void setDescription(String description)  public void setTimestamp(Date timestamp)  public void setUserID(String userID)  public void setUsername(String username)  public void setImgUser(String imgUser)  Input: Image, ImageCompress, Description, Timdestamp, UserID, Username, ImgUser  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính Image, ImageCompress, Description, Timdestamp, UserID, Username, ImgUser của lớp Post | Post.java (32 – 40 – 48 – 56 – 64 – 72 – 80) |

Bảng 5: Các phương thức trong lớp PostId

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public String getId()  public boolean isLike()  public String getFinish()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính id like, finish trong lớp PostId | PostId.java (8 - 17 – 25) |
| 2 | public <T extends PostId> T setId(String id)  Input: id  Output: id  Pseudo code: không có vì đơn giản | Phương thức generic khai báo hoàn thành cho lớp PostId | PostId.java (12) |
| 3 | public void setLike(boolean like)  public void setFinish(String finish)  Input: id, like, finish  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính id like, finish trong lớp PostId | PostId.java (21 – 29) |

Bảng : Các phương thức trong lớp Schedual

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public Schedule(String time, String action, String description)  Input: time, action, description  Output: time, action, description  Pseudo code: không có vì đơn giản | Phương thức mặc định để khai báo lớp Schedual | Schedual.java (11) |
| 2 | public String getTime()  public String getAction()  public String getDescription()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính time, action, description trong lớp Schedual | Schedual.java (17 – 25 – 33) |
| 3 | public void setTime(String time)  public void setAction(String action)  public void setDescription(String description)  Input: time, action, description  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính time, action, description trong lớp Schedual | Schedual.java (21 – 29 – 37) |

Bảng : Các phương thức trong lớp User

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public User(String firstName, String lastName, String phone, String image)  Input: firstName, lastName, phone, image  Output: firstName, lastName, phone, image  Pseudo code: không có vì đơn giản | Phương thức mặc định để hoàn thành khai báo lớp User | User.java (12) |
| 2 | public String getFirstName()  public String getLastName()  public String getPhone()  public String getImage()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Đóng gói các thuộc tính firstName, lastName, phone, image trong lớp User | User.java (19 – 27 – 35 – 43) |
| 3 | public void setFirstName(String firstName)  public void setLastName(String lastName)  public void setPhone(String phone)  public void setImage(String image) | Đóng gói các thuộc tính firstName, lastName, phone, image trong lớp User | User.java (23 – 31 – 39 – 47) |

Bảng : Các phương thức của lớp KiemTra

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public static boolean kiemTraEmail(String email)  Input: email  Output: True or False  Pseudo code: không có vì đơn giản | Kiểm tra email khi tạo tài khoản có hợp lệ không | KiemTra.java (7) |
| 2 | public static boolean KiemTraPhone(String phone)  Input: phone  Output: True or False  Pseudo code: không có vì đơn giản | Kiểm tra số điện thoại nhập vào khi đăng kí tài khoản có hợp lệ không | KiemTra.java (16) |

Bảng : Các phương thức của lớp SupportImage

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public static byte[] nenAnh(Uri imageUri, Context context)  Input: imageUri, context  Output: không  Pseudo code: không có vì đơn giản | Nén ảnh khi đưa lên ứng dụng đi nhiều lần giúp giảm dung lượng cơ sở dữ liệu | SupportImage.java (17) |

Bảng : Các phương thức của lớp UserAdapter

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) Input: parent, viewType  Output: không  Pseudo code: không có vì đơn giản |  | UserAdapter.java (34) |
| 2 | public void onBindViewHolder(@NonNull final ViewHolder holder, final int position)  Input: holder, position  Output: không  Pseudo code: không có vì đơn giản |  | UserAdapter.java (41) |
| 3 | public int getItemCount()  Input: không  Output: không  Pseudo code: không có vì đơn giản |  | UserAdapter.java (61) |

Bảng : Các phương thức của lớp SchedualAdapter

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public ScheduleAdapter(List<Schedule> lsSchedule, Context context)  Input: lsSchedual, context  Output: không  Pseudo code: không có vì đơn giản | Phương thức generic mặc định để hoàn thành khai báo lớp SchedualApdapter | SchedualAdapter.java (20) |
| 2 | public int getCount()  Input: không  Output: không  Pseudo code: không có vì đơn giản | Lấy giá trị count | SchedualAdapter.java (27) |
| 3 | public boolean isViewFromObject(@NonNull View view, @NonNull Object object)  Input: view, object  Output: True or False  Pseudo code: không có vì đơn giản |  | SchedualAdapter.java (32) |
| 4 | public Object instantiateItem(@NonNull ViewGroup container, int position)  Input: container, position  Output: view  Pseudo code: không có vì đơn giản |  | SchedualAdapter.java (39) |
| 5 | private String convertString(String s)  Input: s  Output: chuỗi sau khi đã convert  Pseudo code: không có vì đơn giản | Chuyển hoá s thành dạng chuỗi khác để hiển thị | SchedualAdapter.java (57) |

Bảng : Các phương thức của lớp PostAdapter

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public PostAdapter(List<Post> lsPost)  Input: lsPost  Output: không  Pseudo code: không có vì đơn giản | Phương thức generic mặc định để hoàn thành khai báo lớp PostAdapter | PostAdapter.java (44) |
| 2 | public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)  Input: parent, viewType  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (50) |
| 3 | public void onBindViewHolder(@NonNull ViewHolder holder, final int position)  Input: holder, position  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (59) |
| 4 | private void setCountComment(final ViewHolder holder, String postID)  Input: holder, postID  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (110) |
| 5 | private void setCountLike(final ViewHolder holder, String postID)  Input: holder, PostID  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (125) |
| 6 | private void likeReadTime(final ViewHolder holder, String postID, String userIdCurrent)  Input: holder, postID, userIdCurrent  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (139) |
| 7 | private void setLikeorDislike(final ViewHolder holder, final String postID, final String userIdCurrent)  Input: holder, postID, userIdCurrent  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (152) |
| 8 | public int getItemCount()  Input: không  Output: không  Pseudo code: không có vì đơn giản |  | PostAdapter.java (176) |

Bảng : Các phương thức của lớp CommentAdapter

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public CommentAdapter(List<Comment> lsComment)  Input: lsComment  Output: không  Pseudo code: không có vì đơn giản |  | CommentAdapter.java (25) |
| 2 | public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)  Input: parent, viewType  Output: không  Pseudo code: không có vì đơn giản |  | CommentAdapter.java (31) |
| 3 | public void onBindViewHolder(@NonNull ViewHolder holder, int position)  Input: holder, position  Output: không  Pseudo code: không có vì đơn giản |  | CommentAdapter.java (38) |
| 4 | public int getItemCount()  Input: không  Output: không  Pseudo code: không có vì đơn giản |  | CommentAdapter.java (50 ) |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | public ChatAdapter(List<Chat> lsChat)  Input: lsChat  Output: không  Pseudo code: không có vì đơn giản | Phương thức mặc định để hoàn thành khai báo lớp ChatAdapter | ChatAdapter.java (21) |
| 2 | public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)  Input: parent, viewType  Output: không  Pseudo code: không có vì đơn giản |  | ChatAdapter.java (27) |
| 3 | public void onBindViewHolder(@NonNull ViewHolder holder, int position)  Input: holder, position  Output: không  Pseudo code: không có vì đơn giản |  | ChatAdapter.java (35) |
| 4 | public int getItemCount()  Input: không  Output: không  Pseudo code: không có vì đơn giản |  | ChatAdapter.java (35) |

# CHƯƠNG 4: KẾT LUẬN

## Kết luận

* Sau thời gian tìm hiểu và thực hiện, đề tài “Tìm hiểu Firebase và phát triển ứng dụng trên di động” đã cơ bản được hoàn thành.
* Mục tiêu khi thực hiện đề tài là bước đầu làm quen với môi trường lập trình di động và tạo ra sản phẩm có sử dụng những kiến thức cơ bản về lập trình di động trên nền tảng hệ điều hành android.
* Cụ thể nhóm đã tạo ra được một ứng dụng tên là U-School trên nền tảng Android với một số chức năng cơ bản.
* Song song với việc tìm hiểu và lập trình ứng dụng đó thì nhóm đã tìm hiểu thêm được một phần kiến thức về firebase hỗ trợ cho việc tạo và đưa cơ sở dữ liệu lên server một cách dễ dàng hơn một số cách khác.
* Theo nhóm tự đánh giá thì mức độ hoàn thành được mục tiêu khoảng 70%, dù không hài lòng lắm về kết quả nhưng có thể nhóm sẽ cố gắng phát triển tiếp sản phẩm này và sẽ làm hoàn thiện sản phẩm một cách hoàn chỉnh nhất.

1. ***Một số khó khăn gặp phải***

* Chưa có kiến thức về lập trình di động nên trong thời gian ngắn nhóm vừa tìm hiểu cấp tốc và vừa tạo ra sản phẩm nên sản phẩm chưa thực sự ổn định và còn nhiều mặt hạn chế:
  + Màu sắc, icon của ứng dụng chưa được thiết kế đồng nhất, hài hòa.
  + Chưa xử lý tốt 100% các lỗi xảy ra.
  + Chưa hiện thực tốt các chức năng đề ra như trong phần mở đầu, vẫn còn nhiều thiếu sót về mặt logic cũng như nghiệp vụ.
  + Một số chức năng theo dự kiến ban đầu vẫn chưa hoạt động tốt cũng như là một số chức năng chưa đi vào hoạt động.

# TÀI LIỆU THAM KHẢO

1. https://firebase.google.com/products/
2. https://vi.wikipedia.org/wiki/Android
3. https://yellowcodebooks.com/2016/08/26/lap-trinh-android-bai-1-gioi-thieu-ve-android-moi-truong-phat-trien-pham-mem/#Dung\_Ngon\_Ngu\_Lap\_Trinh\_Java
4. <https://techmaster.vn/posts/33851/lap-trinh-android-bang-ngon-ngu-lap-trinh-nao>
5. https://feedy.vn
6. https://developer.android.com/about/dashboards/index.html
7. https://github.com/firebase/FirebaseUI-Android
8. http://square.github.io/picasso/
9. https://firebase.google.com/docs/storage/android/start