

BỘ THÔNG TIN VÀ TRUYỀN THÔNG  
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

---



**BÁO CÁO KẾT THÚC MÔN HỌC**  
**Lập trình di động**  
**Xây dựng ứng dụng “Độc sách”**

Giảng viên hướng dẫn: Trương Bá Thái

Sinh viên thực hiện: nhóm 6.2

Đặng Thị Thiên Lý - N14DCAT082

Trần Gia Bảo Ngọc - N14DCAT006

Nguyễn Ngọc Hân - N14DCAT131

Tôn Thị Thanh Vân - N14DCAT005

Ngành: An Toàn Thông Tin

Lớp D14CQAT01

Ngày 5/5/2018

**NHẬT KÝ HOẠT ĐỘNG NHÓM 6.2**

STT	HỌ TÊN	CÔNG VIỆC	TỰ ĐÁNH GIÁ	NHÓM ĐÁNH GIÁ	KÝ TÊN
1	Đặng Thị Thiên Lý	<ul style="list-style-type: none"><li>- FlashCreen</li><li>- Đăng ký</li><li>- Đăng nhập, quên mật khẩu</li><li>- Trang chủ</li><li>- Postbai</li><li>- Comment bài post</li><li>- Cài đặt giao diện, liên hệ, giới thiệu.</li><li>- Sửa thông tin cá nhân</li><li>- Firebase</li><li>- Làm báo cáo</li></ul>	%	%	
2	Trần Gia Bảo Ngọc	<ul style="list-style-type: none"><li>- Trang tìm kiếm</li><li>- Trang chi tiết sách</li><li>-Trang đọc sách</li><li>- SQLite</li><li>- Làm báo cáo</li></ul>	%	%	
3	Nguyễn Ngọc Hân	<ul style="list-style-type: none"><li>- Trang tử sách</li><li>-Trang đọc sách</li><li>- Làm báo cáo</li></ul>	%	%	
4	Tôn Thị Thanh Vân	<ul style="list-style-type: none"><li>-Trang danh mục sách</li><li>-Trang sách theo danh mục</li><li>-Trang cá nhân</li><li>- Làm báo cáo</li></ul>	%	%	



## MỤC LỤC

<b>1. MỞ ĐẦU .....</b>	<b>4</b>
1.1. <i>Giới thiệu môn học và nhóm thực hiện.....</i>	<i>4</i>
1.2. <i>Mô tả ứng dụng .....</i>	<i>4</i>
1.3. <i>Ứng dụng trong cuộc sống.....</i>	<i>4</i>
<b>2. PHÂN TÍCH CẤU TRÚC HỆ THỐNG .....</b>	<b>5</b>
2.1. <i>Màn hình “Flash Screen” .....</i>	<i>5</i>
2.2. <i>Màn hình “Đăng nhập” .....</i>	<i>7</i>
2.3. <i>Màn hình “Quên mật khẩu” .....</i>	<i>9</i>
2.4. <i>Màn hình “Đăng ký” .....</i>	<i>11</i>
2.5. <i>Màn hình “Trang chủ” .....</i>	<i>14</i>
2.6. <i>Màn hình “Sửa thông tin cá nhân” .....</i>	<i>21</i>
2.7. <i>Màn hình “Cài đặt giao diện” .....</i>	<i>25</i>
2.8. <i>Màn hình “Hẹn giờ đọc sách” .....</i>	<i>26</i>
2.9. <i>Màn hình “Hướng dẫn và hỗ trợ” .....</i>	<i>28</i>
2.10. <i>Màn hình “Giới thiệu và liên hệ ” .....</i>	<i>29</i>
2.11. <i>Màn hình Post bài .....</i>	<i>31</i>
2.12. <i>Màn hình “Post bài mới” .....</i>	<i>33</i>
2.13. <i>Màn hình “Bình luận bài post” .....</i>	<i>34</i>
2.14. <i>Màn hình “Danh mục sách” .....</i>	<i>35</i>
2.15. <i>Màn hình “Sách theo danh mục” .....</i>	<i>37</i>
2.16. <i>Màn hình “Tìm kiếm sách” .....</i>	<i>39</i>
2.17. <i>Màn hình “Chi tiết sách” .....</i>	<i>40</i>
2.18. <i>Màn hình “Tủ sách” .....</i>	<i>42</i>
2.19. <i>Màn hình “Đọc sách” .....</i>	<i>44</i>
2.20. <i>Màn hình “Cá nhân” .....</i>	<i>46</i>
<b>3. CÀI ĐẶT .....</b>	<b>48</b>
3.1. <i>Trang đăng kí.....</i>	<i>48</i>
3.2. <i>Trang đăng nhập .....</i>	<i>48</i>
3.3. <i>Trang quên mật khẩu .....</i>	<i>49</i>
3.4. <i>Trang chủ.....</i>	<i>49</i>
3.5. <i>Trang sửa thông tin .....</i>	<i>51</i>
3.6. <i>Trang hướng dẫn hỗ trợ.....</i>	<i>53</i>
3.7. <i>Trang post bài .....</i>	<i>53</i>
3.8. <i>Trang bài post mới.....</i>	<i>54</i>
3.9. <i>Danh mục sách .....</i>	<i>55</i>
3.10. <i>Sách theo danh mục .....</i>	<i>58</i>
3.11. <i>Trang tìm kiếm.....</i>	<i>63</i>

3.12.	<i>Trang chi tiết sách</i> .....	64
3.13.	<i>Trang tử sách</i> .....	66
3.14.	<i>Trang đọc sách</i> .....	67
4.	<b>KẾT QUẢ ĐẠT ĐƯỢC</b> .....	69
4.1.	<i>Kết quả đạt được</i> .....	69
4.2.	<i>Kết luận</i> .....	69

## CHƯƠNG 1: MỞ ĐẦU

### ***1.1 Giới thiệu môn học và nhóm thực hiện***

Môn lập trình di động này giúp sinh viên phát triển kỹ năng lập trình, tự tìm kiếm, cập nhật kiến thức mới giúp áp dụng cho các dự án thực tế có lợi sau khi ra trường.

Kiến thức mới: API, JSON, splash screen, navigation drawer, tìm hiểu và sử dụng thư viện Volley, thư viện Picasso, FolioReader....

Thành viên thực hiện:

- Đặng Thị Thiên Lý
- Trần Gia Bảo Ngọc
- Nguyễn Ngọc Hân
- Tôn Thị Thanh Vân

### ***1.2 Mô tả ứng dụng***

- Tên ứng dụng: “LVHN Readbook”
- Cần kết nối mạng
- Android 5.0+
- Thông tin sách lấy từ firebase
- Đọc sách mọi nơi và miễn phí
- Người dùng lưu sách để đọc sau này
- Lưu lịch sử xem

### ***1.3 Ứng dụng trong cuộc sống***

Với việc điện thoại thông minh, thiết bị mạng ngày càng phát triển vì vậy các ứng dụng có kết nối mạng được sử dụng liên tục. Phục vụ nhu cầu đọc sách càng ngày càng nhiều.

Ứng dụng đọc sách mọi lúc mọi nơi chỉ cần 1 điện thoại và không yêu cầu điện thoại có bộ nhớ lớn hơn nữa chức năng lưu trữ giúp người đọc có thể đánh dấu lại sách yêu thích

## CHƯƠNG 2: PHÂN TÍCH CẤU TRÚC HỆ THỐNG

### *Phân tích thiết kế hệ thống*

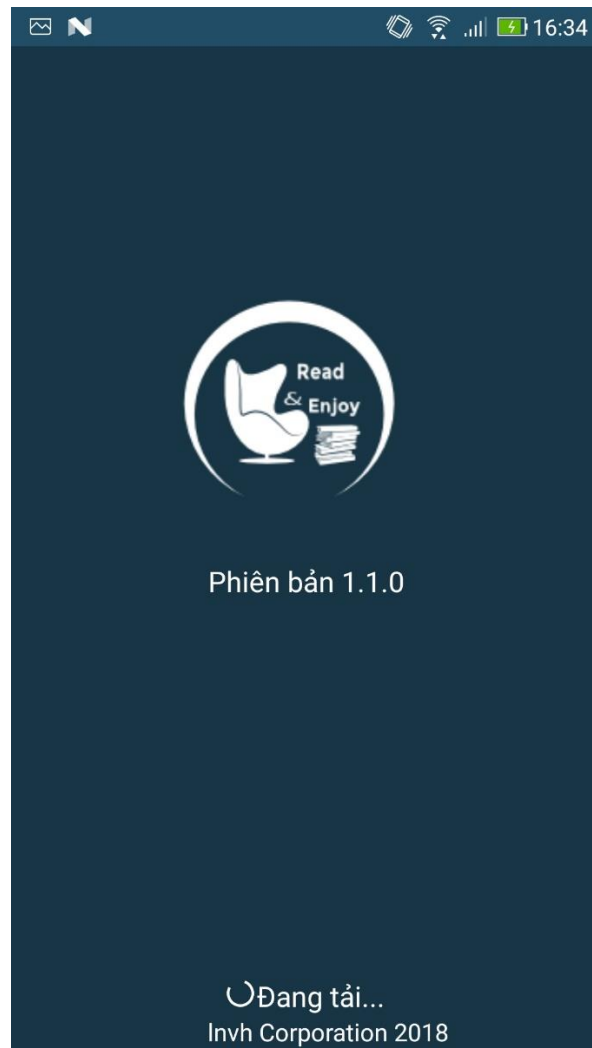
**Tổng quan:** Đây là ứng dụng dùng để đọc sách online, người dùng có thể tải sách về lưu trữ đọc lại khi không có internet, đánh dấu sách yêu thích.

#### **Phạm vi đề tài:**

- Tìm sách và đọc sách, lưu trữ sách
- Lập trình trên android studio v3.1.2
- Hệ điều hành android: 5.0+
- Ứng dụng lấy API từ Website: <http://openweathermap.org>
- Thiết bị thử nghiệm: Pixel XL, Asus zenfone, Samsung Galaxy
- Độ phân giải màn hình 1440x2560xp 5.5inch

#### **1. Màn hình “Flash Screen” (Đặng Thị Thiên Lý)**

- Giao diện người dùng



**Hình 1 Màn hình splash**

**o Yêu cầu chức năng**

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình splash	Hiển thị logo ,version app	Sau 3s chuyển qua màn hình login	Hoàn thành

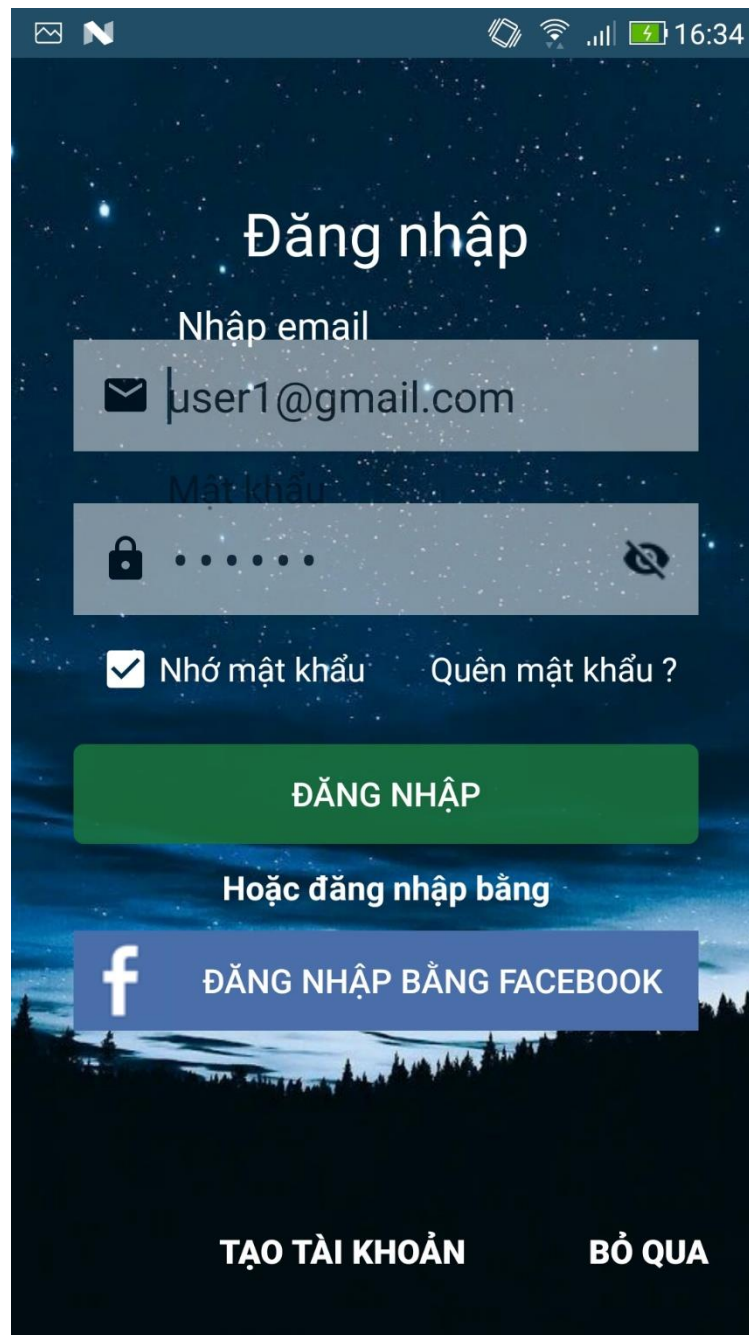
**o Làm thế nào ?**

- Chạy ứng dụng
- Chương trình sẽ chạy màn hình Splash
- Sau 3s sẽ tự động chuyển qua màn hình Login



## 2. Màn hình “Login”

### o Giao diện người dùng



Hình 2 Màn hình Login

o **Yêu cầu chức năng**

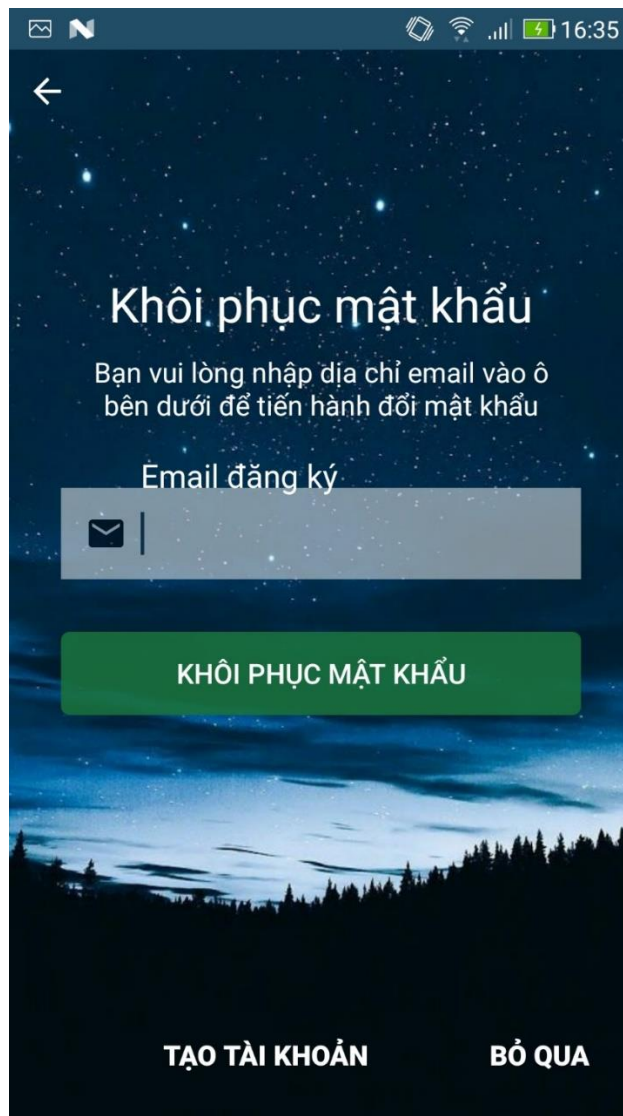
Tiêu đề	Mô tả	Quy trình	Kết quả
Màn hình home	Màn hình chính có: - 1 image background - 4 Button - 2 EditText - 1 Check box	Thiết kế	Hoàn thành
Edit text “Email”	Nhập email đăng ký để login	Thiết kế	Bắt lỗi sai định dạng email
Edit text “ Password”	Nhập pass để login	Thiết kế	Ẩn và hiện password
Checkbox “ Nhớ tài khoản”	Lưu tài khoản sau khi login thành công	Click	Nhớ tài khoản
TextView “Quên mật khẩu”	Chuyển sang giao diện “Reset password qua email”	Click	Chuyển sang giao diện “Quên mật khẩu”
Button “Đăng nhập”	Login	Click	Chuyển sang giao diện “Trang chủ” nếu login thành công, ngược lại báo sai
Button “Đăng nhập bằng facebook”	Login bằng facebook	Click	Chuyển sang giao diện “Trang chủ” sau khi login facebook
Button “Tạo tài khoản”	Chuyển qua trang “Đăng ký tài khoản”	Click	Chuyển qua trang “Đăng ký tài khoản”
Button “Bỏ qua”	Bỏ qua đăng nhập để vào trang chủ	Click	Chuyển qua trang “Trang chủ”

#### o Làm thế nào ?

- Làm thế nào click button đăng nhập sẽ xét dữ liệu edittext nếu đúng định dạng ,khác rỗng thì sẽ gửi yêu cầu tạo tài khoản firebase.
- Làm thế nào click button đăng kí thì chuyển qua trang đăng kí
- Làm thế nào click button bỏ qua thì không cần đăng nhập mà chuyển qua trang chủ
- Làm thế nào click textview “Quên mật khẩu” sẽ chuyển qua giao diện quên mật khẩu
- Làm thế nào click button đăng nhập facebook ứng dụng sẽ gửi yêu cầu tạo tài khoản bằng facebook lên firebase.

### 3. Màn hình “Quên mật khẩu”

#### o Giao diện người dùng



Hình 3 Màn hình Quên mật khẩu

#### o Yêu cầu chức năng

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình Reset Password	Màn hình chính có: - 1 image background - 2 Button - 1 EditText	Thiết kế	Hoàn thành
Edit text “Email”	Nhập email đăng ký để reset password	Thiết kế	Bắt lỗi sai định dạng email
Button “Khôi phục mật khẩu”	Gửi yêu cầu đổi password đến email trên	Click	Hoàn thành
Button “Tạo tài khoản”	Chuyển qua trang “Đăng ký tài khoản”	Click	Chuyển qua trang “Đăng ký tài khoản”
Button “Bỏ qua”	Bỏ qua đăng nhập để vào trang chủ	Click	Chuyển qua trang “Trang chủ”

#### o Làm thế nào ?

- Làm thế nào khi click button “ Khôi phục mật khẩu” thì gửi reset password đến email
- Làm thế nào check email đúng định dạng và có đăng kí hay chưa.

#### 4. Màn hình “Đăng ký”

##### o Giao diện người dùng

16:35

# Đăng ký

Tên tài khoản

nguyen hai

Email

user1@gmail.com

Mật khẩu

•

Mật khẩu ít nhất 6 kí tự và có in hoa

Nhập lại mật khẩu

•

ĐĂNG KÝ

Hình 4 Màn hình Đăng ký

### o Yêu cầu chức năng

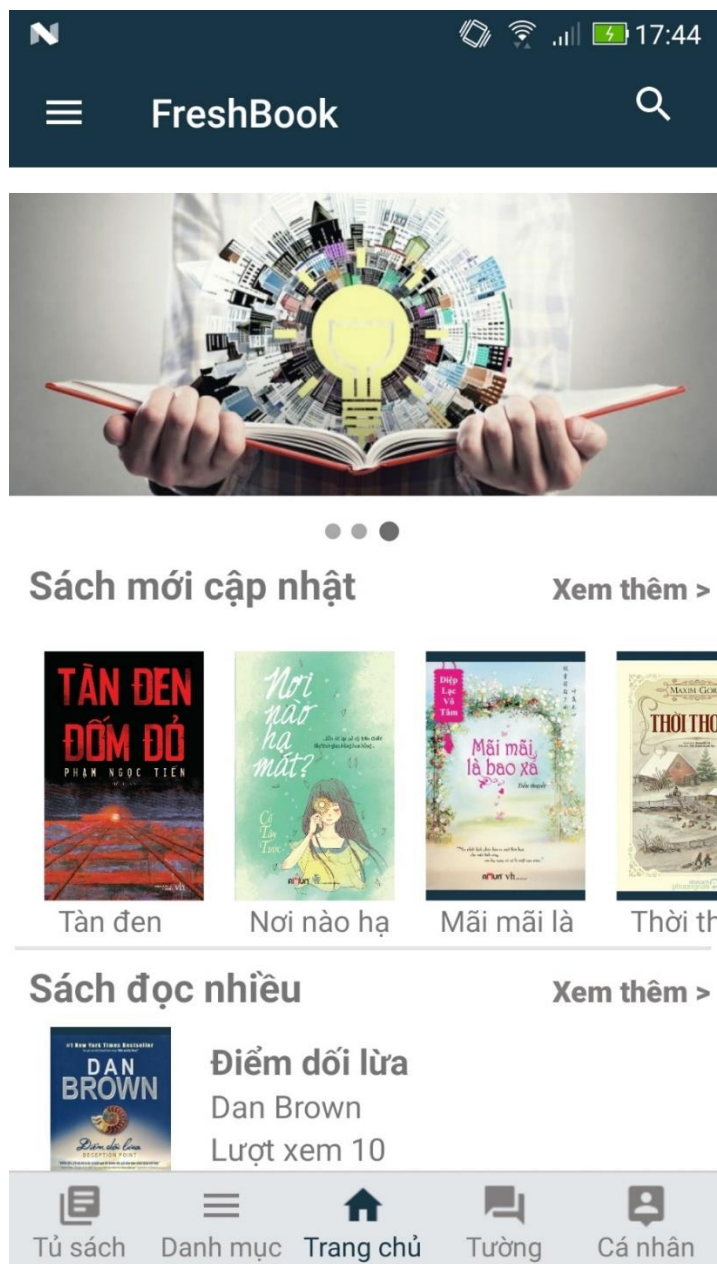
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình Đăng ký	Màn hình chính có: - 1 image background - 1 Button - 4 EditText	Thiết kế	Hoàn thành
Edit text “username”	Nhập user hiển thị	Thiết kế	Hoàn thành
Edit text “Email”	Nhập email đăng ký	Thiết kế	Bắt lỗi sai định dạng email
Edit text “Password”	Nhập password	Thiết kế	Bắt lỗi nhập password yếu
Edit text “Nhập lại password”	Nhập lại password, so sánh password ở trên	Thiết kế	So khớp password
Button “Đăng ký”	Đăng ký thành công chuyển qua trang chủ	Click	Chuyển qua trang “Trang chủ” nếu email, pass đúng định dạng

### o Làm thế nào ?

- Làm thế nào check định dạng email hợp lệ
- Làm thế nào check định dạng mật khẩu mạnh
- Làm thế nào so khớp mật khẩu
- Làm thế nào khi click button đăng kí, yêu cầu tạo tài khoản sẽ gửi lên firebase và thông báo đăng kí thành công, rồi chuyển qua trang chủ.

## 5. Màn hình “Trang chủ”

### o Giao diện người dùng







## Harry Potter

J.K Rowling

Lượt xem 4

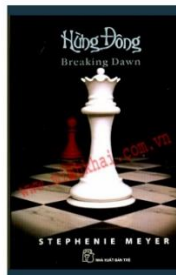
24-08-2017 22:45:15

### Văn học nước ngoài

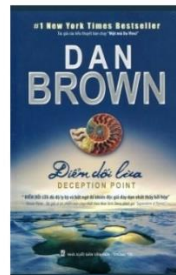
[Xem thêm >](#)



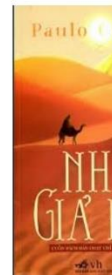
Harry Potter



Hừng đông



Điểm đổi



Nhà giả kim

### Văn học trong nước

[Xem thêm >](#)



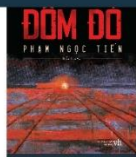
## Ai rồi cũng khác

Hamlet Trương

Lượt xem 9

22-08-2016 22:54:45





Phạm Ngọc Tiến  
Lượt xem 0  
25-5-2018 22:45:12



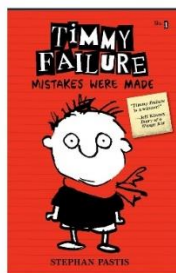
**Nắng trong vườn**  
Thạch Lam  
Lượt xem 0  
22-05-2014 21:52:56



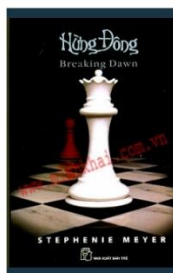
**Nếu như không thể nói**  
Join Phạm  
Lượt xem 0  
25-08-2014 20:55:23

## Sách đề cử

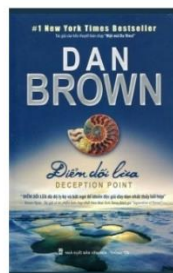
[Xem thêm >](#)



Timmy



Hùng đông



Điểm đối



Thời tr



Người dùng  
user@gmail.com



Cài đặt giao diện



Hẹn giờ đọc sách



Ngôn ngữ



Hướng dẫn và hỗ trợ



Giới thiệu và liên hệ



Theo dõi chúng tôi trên



Đăng nhập

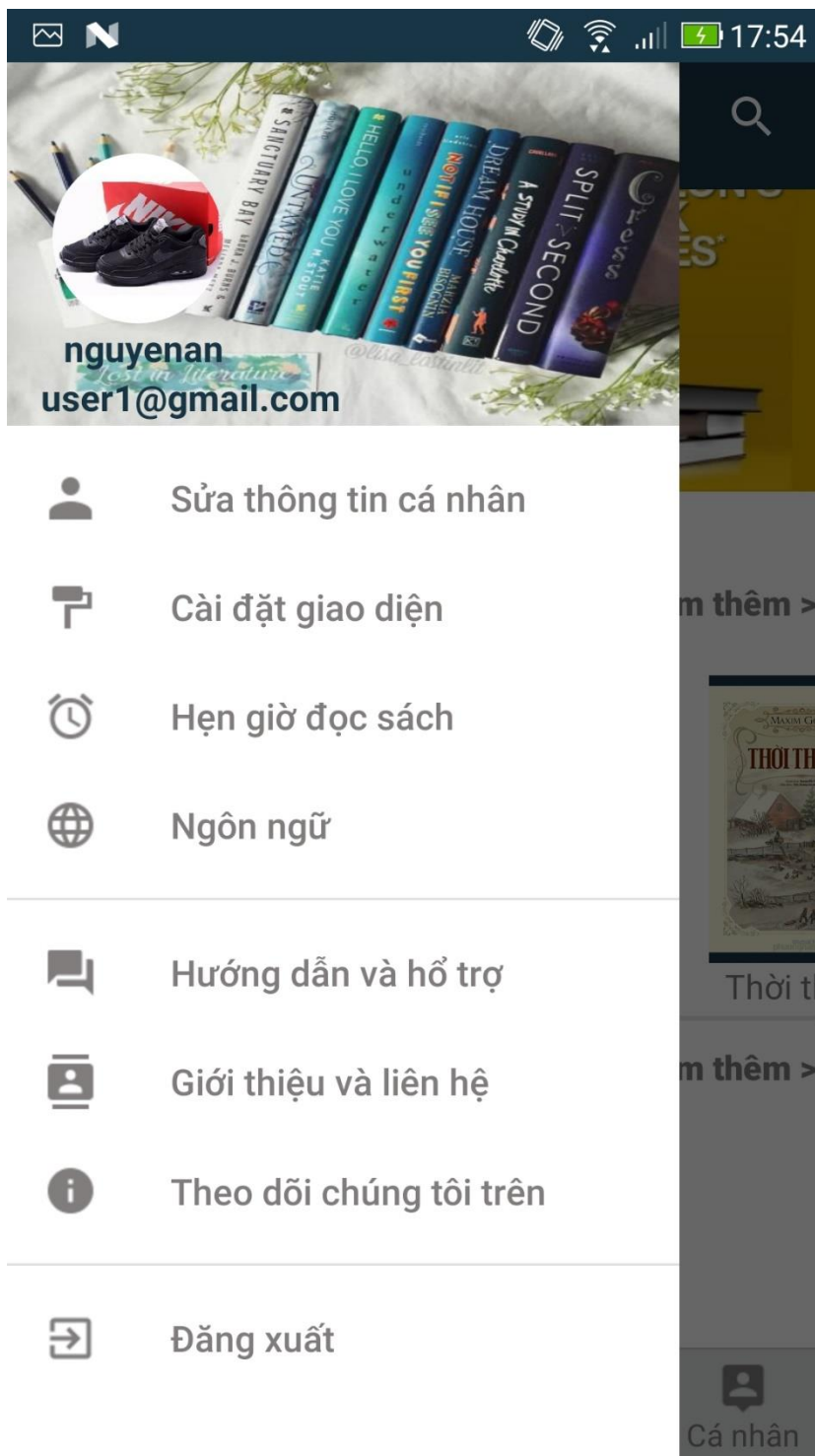
m thêm >



Thời th



Cá nhân



Hình 5,6,7,8,9 Màn hình Trang chủ

o **Yêu cầu chức năng**

Tiêu đề	Mô tả	Quy trình	Kết quả
Màn hình Đăng ký	Màn hình chính có: - 1 toolbar - 1 ViewPager - 1 Recyclerview - 1 BottomNavigation - 1 Navigation menu - 1 Button tìm kiếm	Thiết kế	Hoàn thành
Toolbar	Hiện thị Navigation menu	Click	Show Navigation menu thành công
Button tìm kiếm	Chuyển qua trang tìm kiếm	Click	Chuyển qua trang tìm kiếm
BottomNavigation	Chuyển qua lại các layout “Trang chủ”, “Danh mục”, “Cá nhân”, “Tủ sách”...	Click	Hoàn thành
Navigation drawer menu “Chỉnh sửa cá nhân”	Chuyển sang giao diện chỉnh sửa thông tin cá nhân nếu người dùng đã login, nếu chưa login mục này bị ẩn	Click	Chuyển sang sửa thông tin cá nhân khi đã login, nếu không bị ẩn

Navigation drawer menu “Hẹn giờ đọc sách”	Sau khi hẹn giờ, đồng hồ đếm lùi, đến khi xong thì thông báo	Click	Đếm lùi thành công
Navigation drawer menu “Ngôn ngữ”	Thay đổi ngôn ngữ app	Click	Chưa xử lý sự kiện
Navigation drawer menu “Hướng dẫn hỗ trợ”	Các câu hỏi thường gặp	Thiết kế	Hoàn thành
Navigation drawer menu “Giới thiệu và liên hệ”	Giới thiệu và cách thức liên hệ app	Thiết kế	Hoàn thành
Navigation drawer menu “Theo dõi chúng tôi”	Chuyển trực tiếp đến facebook fanpage của app	Click	Chưa xử lý sự kiện
Navigation drawer menu “Đăng xuất/Đăng nhập”	Đăng xuất khi người dùng đã login. Đăng nhập nếu người dùng chưa login	Click	Hoàn thành
RecyclerView	Show “Sách mới cập nhật” sắp xếp theo thời gian cập nhật Show “Sách đọc nhiều” sắp xếp theo lượt đọc Show “Văn học trong nước” Show “Văn học nước ngoài” Show “Sách đề cử” sắp xếp theo lượt đánh giá Khi click vào sách sẽ chuyển sang trang chi tiết sách	Click	Hoàn thành

	Khi click button “Xem thêm” chuyển sang xem nhiều sách hơn.		
--	---	--	--

#### **o Làm thế nào ?**

- Làm thế nào khi click button tìm kiếm chuyển qua layout tìm kiếm sách
- Làm thế nào khi click button hom sẽ hiện navigation drawer menu
- Làm thế nào hiển thị danh sách sách mới (xếp theo ngày cập nhật), sách đọc nhiều (xếp theo lượt xem), sách đề cử( xếp theo đánh giá), và một số thể loại sách tiêu biểu
- Làm thế nào khi click các button trên bottom navigation sẽ chuyển qua layout tương ứng
- Làm thế nào khi click button xem thêm thì sẽ chuyển qua danh mục sách theo loại sách đã thể hiện trên trang chủ ( do trang chủ chỉ hiển thị một số sách trong từng loại sách”
- Làm thế nào khi click các item trong navigation drawer menu thì sẽ chuyển qua các layout tương ứng
- Làm thế nào khi click các item sách trên trang chủ thì sẽ chuyển qua trang chi tiết của sách đó.

### **6. Màn hình “Sửa thông tin cá nhân”**

#### **o Giao diện người dùng**



## Thay đổi tài khoản



Tên tài khoản



nguyenan

Email



user1@gmail.com

Mật khẩu



• • • • •



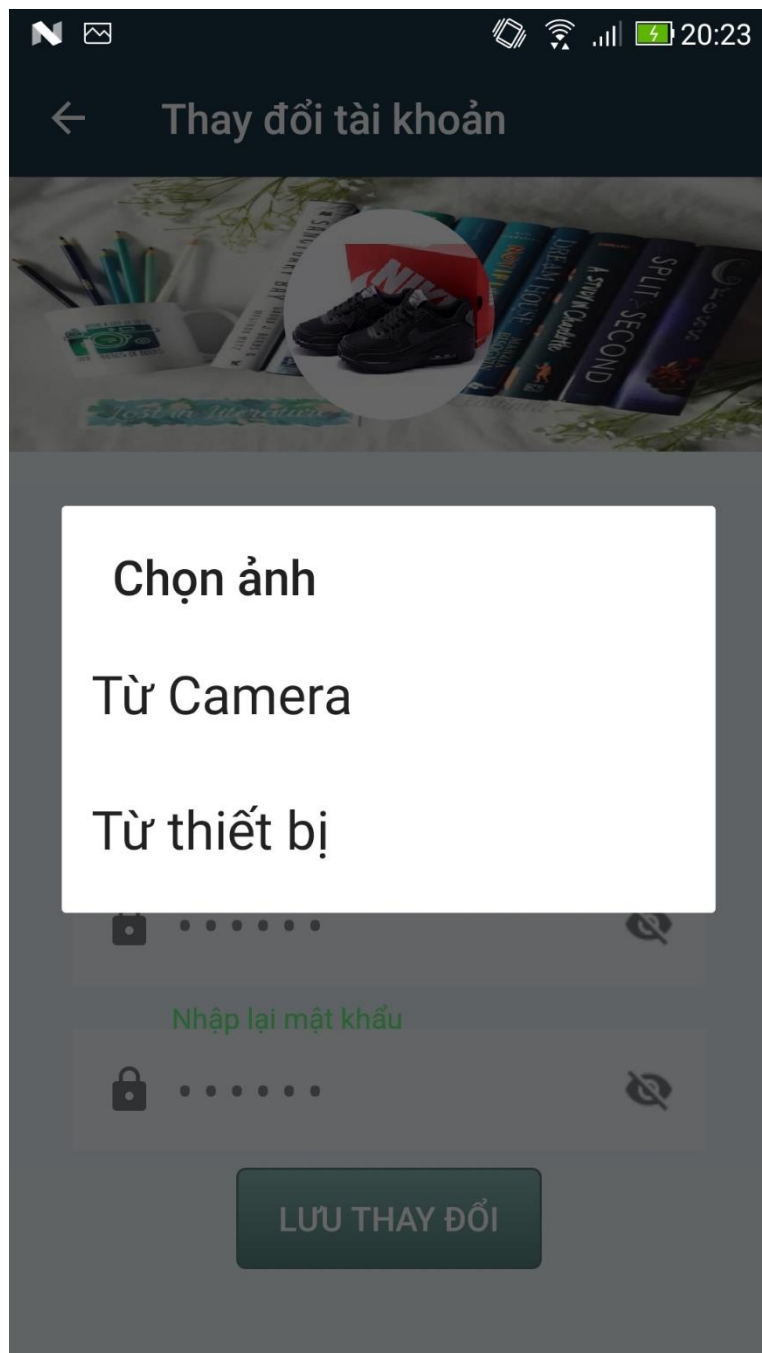
Nhập lại mật khẩu



• • • • •



LƯU THAY ĐỔI



Hình 10,11 Màn hình “Sửa thông tin tài khoản”



o **Yêu cầu chức năng**

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Sửa thông tin tài khoản”	Màn hình chính có: - 1 toolbar - 1 background image trên applayoutbar - 1 Imageview - 4 EditText - 1 Dialog chọn ảnh	Thiết kế	Hoàn thành
Toolbar	Quay về trang chủ	Click	Chuyển về trang chủ
ImageView	Hiển thị ảnh đại diện của user, user có thể thay đổi ảnh từ thư viện hay chụp đều được	Click	Hoàn thành
Dialog “chọn ảnh”	Khi click vào Imageview, dialog xuất hiện cho phép chọn ảnh thay đổi avatar	Click	Hoàn thành
Edit text “username”	Sửa user hiển thị	Thiết kế	Hoàn thành
Edit text “Email”	Nhập email đăng ký	Thiết kế	Bắt lỗi sai định dạng email
Edit text “Password”	Nhập password	Thiết kế	Bắt lỗi nhập password yếu

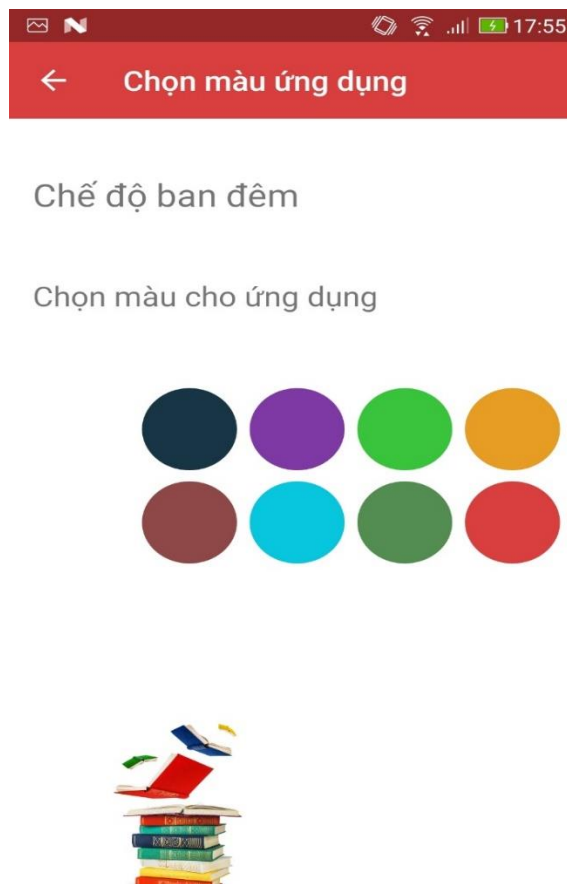
Edit text “Nhập lại password”	Nhập lại password, so sánh password ở trên	Thiết kế	So khớp password
Button “Gửi”	Cập nhật lại thông tin user	Click	Cập nhật thành công

#### o Làm thế nào ?

- Làm thế nào load thông tin user đã đăng nhập
- Làm thế nào khi click vào hình ảnh người dùng có thể load ảnh để thay đổi ảnh đại diện
- Làm thế nào check email hợp lệ và khớp mật khẩu
- Làm thế nào khi click button Lưu thì thông tin tài khoản được sửa trên firebase và thông báo thành công

### 7. Màn hình “Cài đặt giao diện”

#### o Giao diện người dùng



Hình 12 Màn hình “Cài đặt giao diện”

### o Yêu cầu chức năng

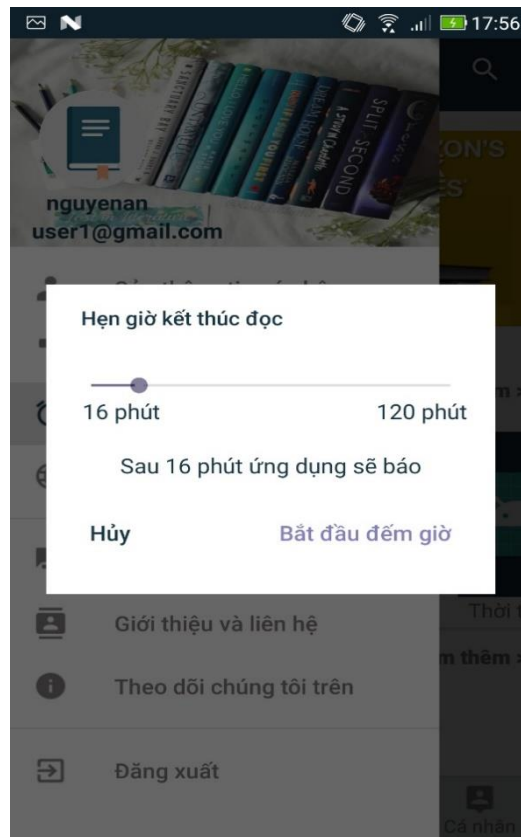
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Cài đặt giao diện”	Màn hình chính có: - 1 toolbar - 8 button - 1imageview	Thiết kế	Hoàn thành
Toolbar	Quay về trang chủ	Click	Chuyển về trang chủ
8 Button màu sắc	Khi click vào màu nào app sẽ thay đổi giao diện màu tương ứng	Click	Hoàn thành

### o Làm thế nào ?

- Làm thế nào đổi màu app khi click button
- Làm thế nào khi click button trên tool bar sẽ quay về trang chủ

## 8. Màn hình “Hẹn giờ đọc sách”

### o Giao diện người dùng



**Hình 13 Màn hình “Hẹn giờ đọc sách”**

**o Yêu cầu chức năng**

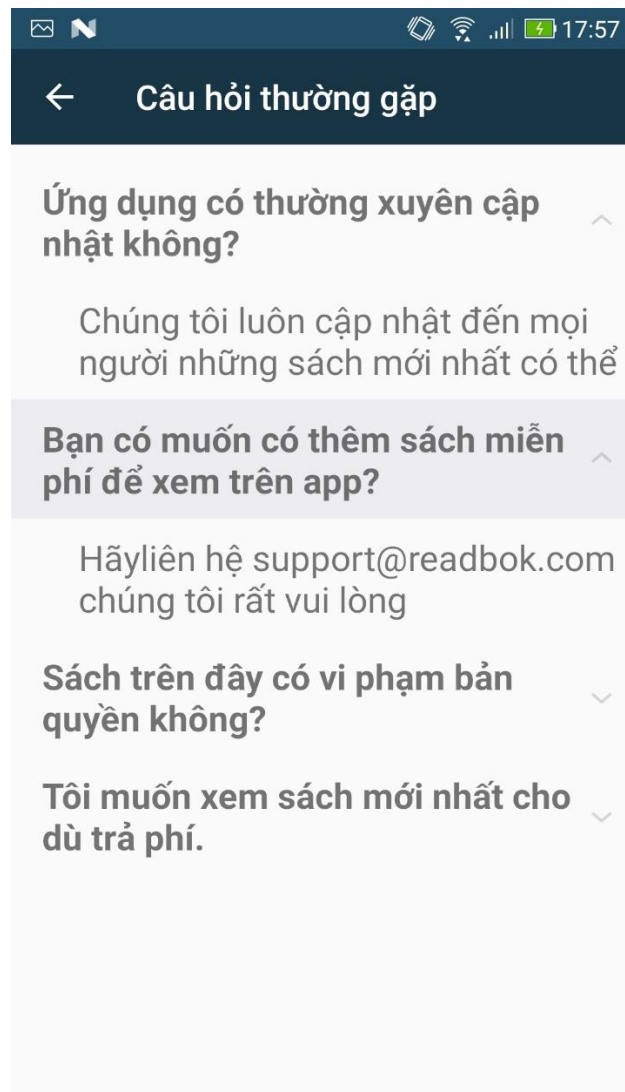
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Hẹn giờ đọc sách”	Màn hình chính có: - 1 dialog , có 2 button, 1 seekbar	Thiết kế	Hoàn thành
Seekbar	Đếm lùi thời gian khi hẹn giờ đọc	Click	Hoàn thành
Button	Khi click button hẹn giờ sẽ bắt đầu đếm giờ, hết giờ thì báo cho người đang đọc	Click	Đếm lùi nhưng chưa báo được

**o Làm thế nào ?**

- Làm thế nào seek bar chạy
- Làm thế nào đếm lùi
- Làm thế nào báo cho người đọc

**9. Màn hình “ Hướng dẫn và hỗ trợ”**

**o Giao diện người dùng**



**Hình 14 Màn hình “Hướng dẫn và hỗ trợ”**

### o Yêu cầu chức năng

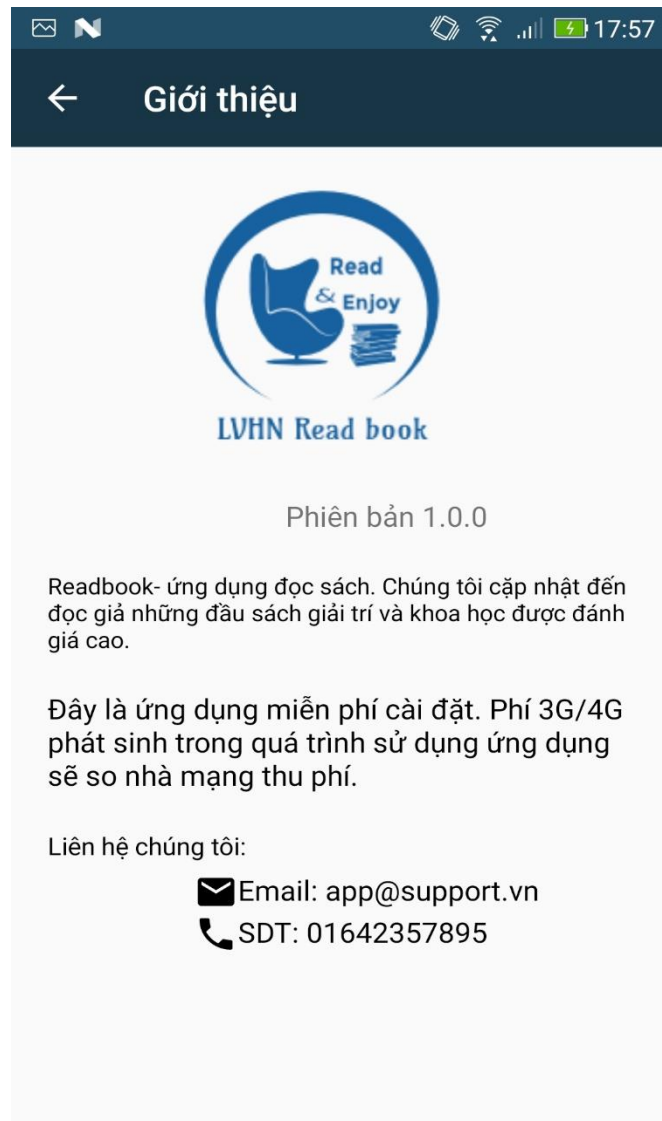
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Hướng dẫn và hỗ trợ”	Màn hình chính có: - 1 toolbar - listview	Thiết kế	Hoàn thành
Toolbar	Quay về trang chủ	Click	Chuyển về trang chủ
listview	Khi click vào câu hỏi sẽ hiện câu trả lời	Click	Hoàn thành

### o Làm thế nào ?

- Làm thế nào lấy câu hỏi và trả lời từ firebase về
- Làm thế nào khi click button trên tool bar sẽ quay về trang chủ

## 10. Màn hình “Giới thiệu và liên hệ ”

### o Giao diện người dùng



**Hình 15 Màn hình “Giới thiệu và liên hệ**

**o Yêu cầu chức năng**

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Giới thiệu và liên hệ”	Màn hình chính có: - 1 toolbar - 1 Imageview - textview	Thiết kế	Hoàn thành

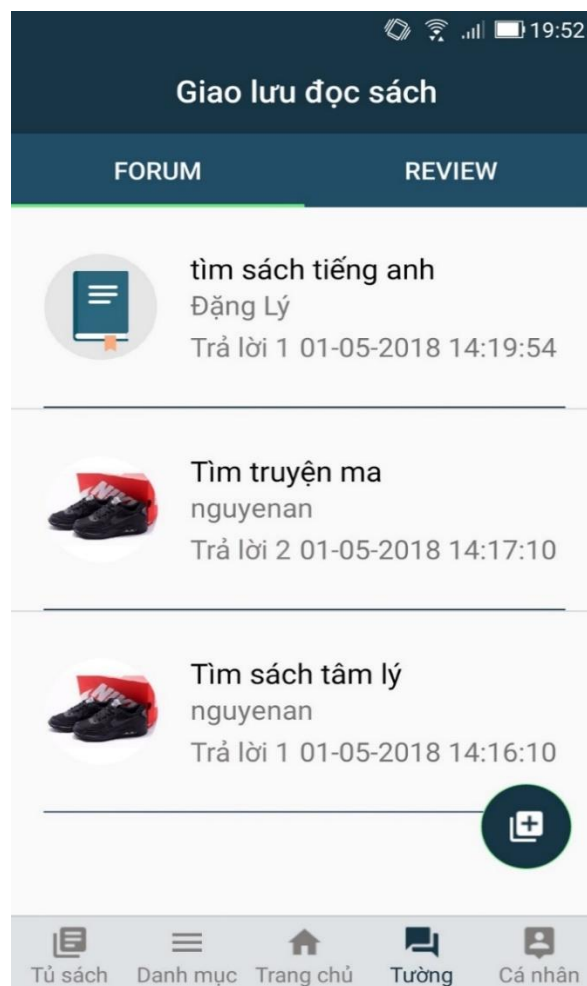
Toolbar	Quay về trang chủ	Click	Chuyển về trang chủ
Imageview và giới thiệu	Icon app và giới thiệu app	Thiết kế	Hoàn thành

o **Làm thế nào ?**

- Làm thế nào khi click button trên tool bar sẽ quay về trang chủ

## 11. Màn hình “Post bài”

o **Giao diện người dùng**



**Hình 16 Màn hình “Post bài”**



### o Yêu cầu chức năng

Tiêu đề	Mô tả	Quy trình	Kết quả
Màn hình “Post bài”	Màn hình chính có: - 1 toolbar - Tablayout Viewpager - 1 FloatButton Action - ListView	Thiết kế	Hoàn thành
Toolbar	Hiển thị tiêu đề	Thiết kế	Hoàn thành
Viewpager	Tab forum cho phép user sau khi login post bài tìm sách, giao lưu, Tab review hiển thị các bình luận mới nhất, click vào nhảy đến chi tiết sách đã bình luận	Click	Tab forum hoàn thành
Listview	Hiển thị các bình luận sắp xếp theo ngày đăng mới nhất	Click	Hoàn thành
FloatButtonAction	Cho phép user đăng bài, nếu user đã đăng nhập thì chuyển sang “Đăng bài” nếu chưa chuyển sang trang “Đăng nhập”	Click	Hoàn thành

### o Làm thế nào ?

- Làm thế nào lấy danh sách bài post và user đăng bài hiển thị listview
- Làm thế nào khi click button floatactionbutton “+” nếu người dùng chưa login thì yêu cầu login, nếu người dùng đã login thì chuyển sang trang đăng bài
- Làm thế nào khi click vào bài post thì sẽ chuyển qua layout danh sách bình luận của bài post

12. Màn hình “Post bài mới”

o Giao diện người dùng



Hình 17 Màn hình “Post bài”

o Yêu cầu chức năng

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Post bài mới”	Màn hình chính có: - 1 toolbar - 2 Edittext - 1 Button	Thiết kế	Hoàn thành
Toolbar	Quay lại trang “Post bài”	Thiết kế	Hoàn thành
Edittext “Tiêu đề”	Tiêu đề bài post	Click	Tab forum hoàn thành

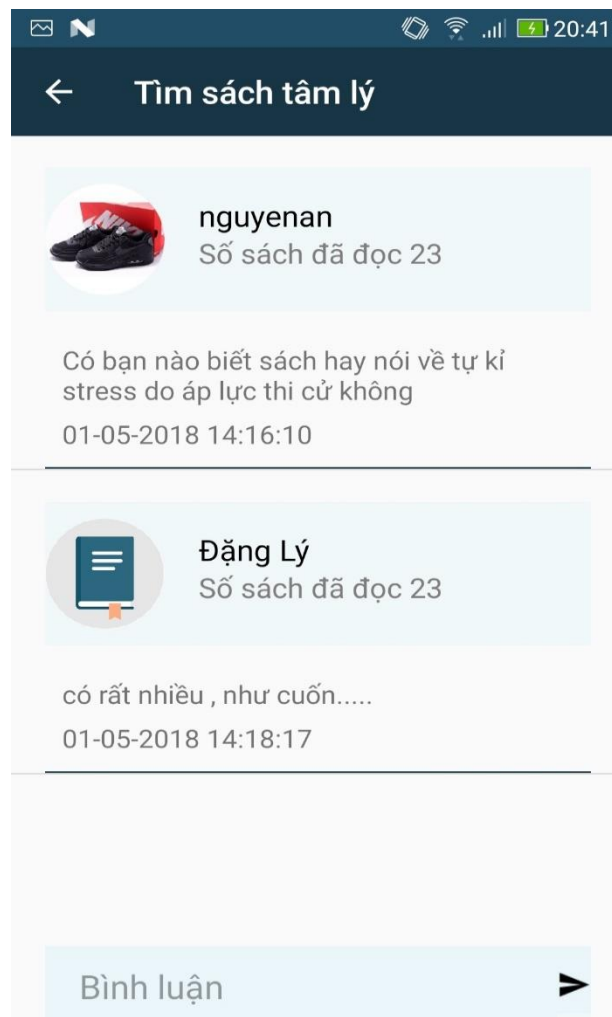
Edittext “Nội dung”	Nội dung bài post	Click	Hoàn thành
Button “Gửi”	Nếu tiêu đề và nội dung khác rỗng thì thêm bài post	Click	Hoàn thành

**o Làm thế nào ?**

- Làm sao click button quay về sẽ chuyển sang layout danh sách bài post
- Làm thế nào khi click button đăng thì thêm bài post vào firebase với userid người đăng

### 13. Màn hình “Bình luận bài post”

**o Giao diện người dùng**



**Hình 18 Màn hình “Post bài”**

o **Yêu cầu chức năng**

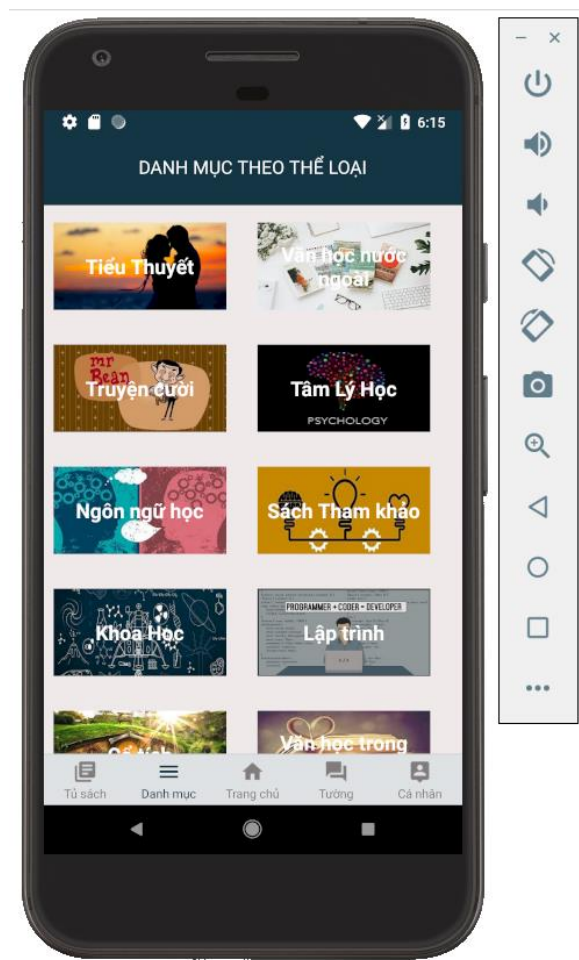
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình “Post bài mới”	Màn hình chính có: - 1 toolbar - 1 Listview - 1 Edittext -1 ImageButton	Thiết kế	Hoàn thành
Toolbar	Hiển thị tiêu đề bài post và uay lại trang “Post bài”	Thiết kế	Hoàn thành
Listview	Hiển thị các bình luận trong bài post	Click	Hoàn thành
Edittext “Nội dung”	Nội dung bình luận mới cho bài post hiện tại	Click	Hoàn thành
Button “Gửi” icon là send	Nếu user chưa login yêu cầu login, ngược lại thêm bình luận	Click	Hoàn thành

o **Làm thế nào ?**

- Làm thế nào lấy danh sách bình luận của bài post đã click vào và hiển thị
- Làm thế nào khi click nút gửi bình luận, nếu người dùng chưa login thì bắt login, nếu người dùng đã login thì thêm vào danh sách bình luận của bài post nội dung bình luận và userid

**14. Màn hình “Danh mục sách” ( Tôn Thị Thanh Vân)**

o **Giao diện người dùng**



**Hình 19** Màn hình “danh mục thể loại sách”

o **Yêu cầu chức năng**

Tiêu đề	Mô tả	Qui trình	Kết quả
Danh mục theo thể loại	Màn hình có: ImageView, TextView, GridView	Nhấn vào tên thể loại để hiển thị sách	Hoàn thành
ImageView	Hiển thị ảnh của thể loại	Click	Hoàn thành
Text View	Hiển thị tên thể loại	Hiển thị	Hoàn thành
GridView	Sắp xếp các thể loại theo dạng lưới có 2 cột	Hiển thị	Hoàn thành

### o Làm thế nào ?

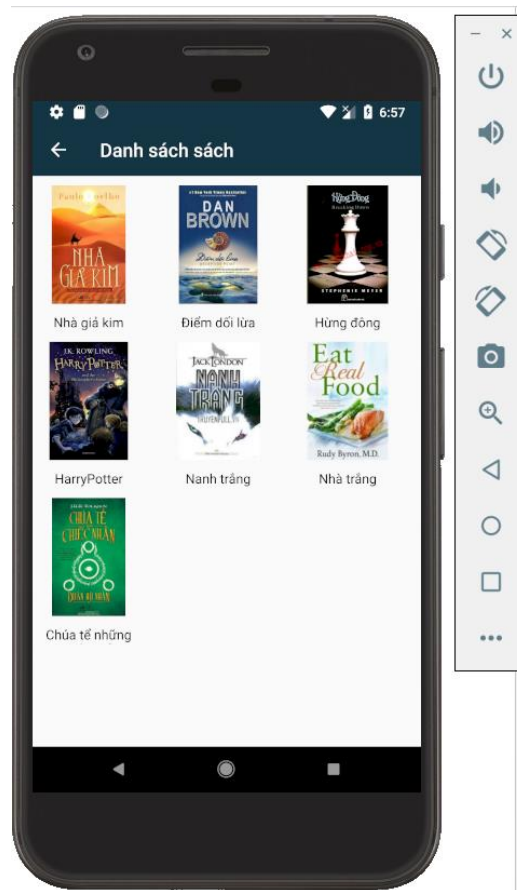
- Từ màn hình chính của trang chủ, bấm vào Danh mục ở phía Bottom Navigation View để vào được danh mục
- Lấy dữ liệu trên Firebase như hình ảnh để hiển thị về ImageView, lấy tên thể loại để hiển thị về TextView.

Cài đặt: Danh mục sách

- Sử dụng GridView + ImageView + TextView để thiết kế giao diện

## 15. Màn hình “Sách theo danh mục” (Tôn Thị Thanh Vân)

### o Giao diện người dùng



**Hình 20 Màn hình “Sách theo danh mục”**

### o Yêu cầu chức năng

Tiêu đề	Mô tả	Qui trình	Kết quả
Sách theo danh mục	Màn hình có: ImageView, TextView, GridView	Hiển thị	Hoàn thành

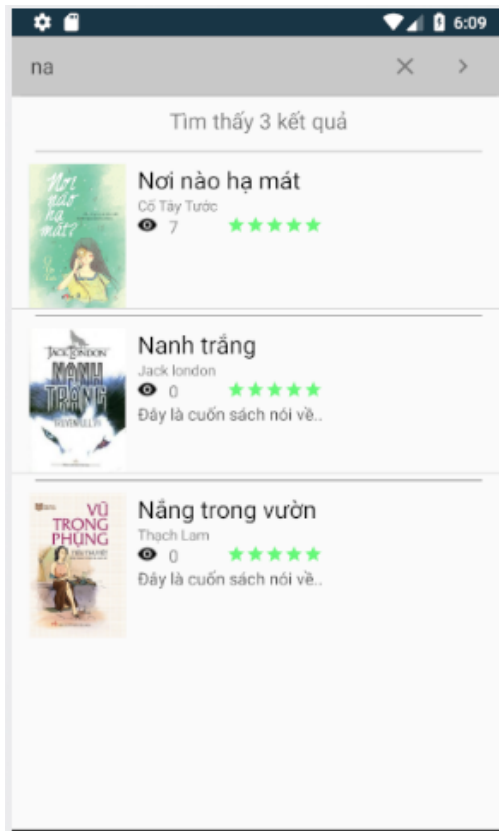
ImageView	Hiển thị ảnh của thể loại	Click	Hoàn thành
Text View	Hiển thị tên thể loại	Hiển thị	Hoàn thành
GridView	Sắp xếp các thể loại theo dạng lưới có 2 cột	Hiển thị	Hoàn thành
Toolbar	Nhấn để trở về trang Danh mục theo thể loại	Hiển thị	Hoàn thành

#### o Làm thế nào ?

- Từ trang Danh Mục Theo Thể Loại, ta nhấp vào thể loại bất kỳ để hiện ra sách trong thể loại đó
- ImageView hiển thị hình ảnh của sách
- TextView hiển thị tên sách

## 16. Màn hình “Tìm kiếm sách” (Trần Gia Bảo Ngọc)

### o Giao diện người dùng



**Hình 21 Màn hình “Tìm kiếm sách”**

**o Yêu cầu chức năng**

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình tìm kiếm	Màn hình chính có: 1 SearchView 1 ListView	Thiết kế	Hoàn thành
SearchView “Tìm kiếm”	Nhập tên sách cần tìm	Input	Lấy tên sách cần tìm
ListView	Hiển thị danh sách các sách có chứa từ tìm kiếm	View	Hiển thị danh sách các sách có chứa từ tìm kiếm

**o Làm thế nào ?**

SearchView:



- Nhập tên sách cần tìm
- Giá trị vừa nhập thành dạng không dấu
- Duyệt tất cả sách trong firebase, so sánh giá trị nhập với thuộc tính tên sách trong firebase
- Các sách thỏa điều kiện được thêm vào ArrayList

ListView:

- Load tất cả đối tượng trong arraylist lên ListView thông qua custom adapter
- Click vào đối tượng trong ListView, ứng dụng chuyển sang màn hình trang chi tiết của đối tượng đó, truyền tham số mã sách và mã thể loại.

## 17. Màn hình “Chi tiết sách” (Trần Gia Bảo Ngọc)

### o Giao diện người dùng



**Hình 22 Màn hình “Chi tiết sách”**

### o Yêu cầu chức năng

Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình chi tiết sách	Màn hình chính có: 7 TextView 1 RatingBar 1 ImageView 1 ImageButton	Thiết kế	Hoàn thành
ImageView “Bìa sách”	Hiển thị hình ảnh bìa sách	View	Hiển thị hình ảnh bìa sách
TextView “Tên sách”	Hiển thị tên sách	View	Hiển thị tên sách
TextView “Tác giả”	Hiển thị tên tác giả	View	Hiển thị tên tác giả
RatingBar “đánh giá”	Hiển thị và thay đổi điểm đánh giá	Click	Thay đổi điểm đánh giá
TextView “Lượt xem”	Hiển thị lượt xem	View	Hiển thị lượt xem
ImageButton “đọc sách”	Download sách về điện thoại và chuyển sang giao diện “đọc sách”	Click	Sách được download và chuyển sang giao diện “đọc sách”
TextView “Ngày đăng”	Hiển thị thông tin ngày đăng	View	Hiển thị thông tin ngày đăng
TextView “Lượt thích”	Hiển thị thông tin lượt thích	View	Hiển thị thông tin lượt thích
TextView “Năm xuất bản”	Hiển thị thông tin năm xuất bản	View	Hiển thị thông tin năm xuất bản
TextView “Mô tả”	Hiển thị thông tin mô tả sách	View	Hiển thị thông tin mô tả sách

### o Làm thế nào ?

Lấy mã sách, mã thể loại sách trong getIntent()

Lấy thông tin sách trên firebase từ mã sách và mã thể loại sách nhận được

Từ thông tin đường dẫn hình ảnh, lưu hình ảnh từ firebase storage về RAM

Đưa thông tin lấy được lên Layout

RattingBar:

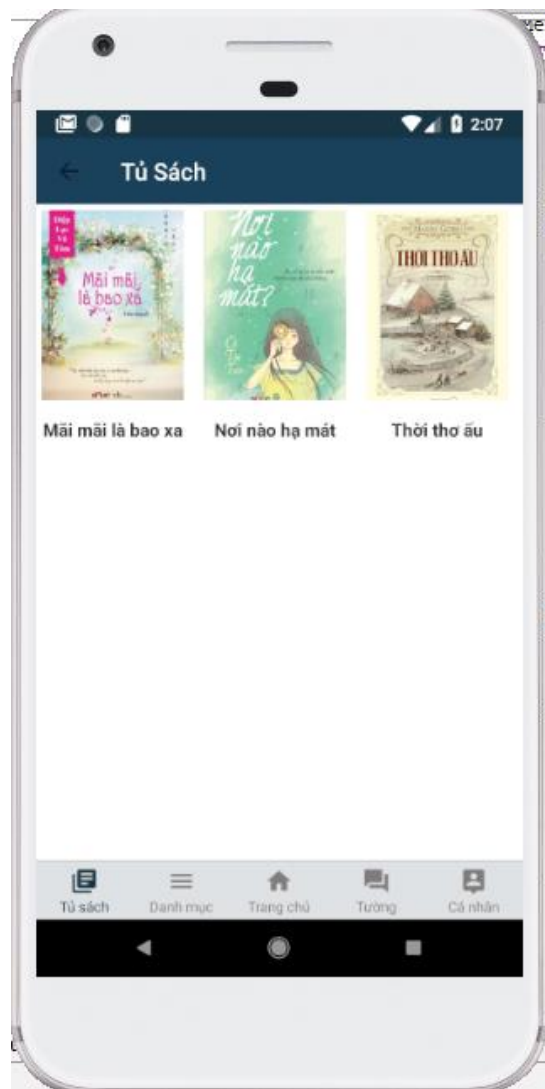
- Lấy giá trị từ RattingBar lưu lên child “ratting” trên firebase

Button “Đọc sách”:

- Kiểm tra mã sách có trong SQLite không?
- Nếu có, lấy đường dẫn file lưu trong SQLite của mã sách truyền intent cho trang đọc sách.
- Nếu không, lấy đường dẫn hình ảnh và file sách trên firebase database, download hình ảnh và file về máy, lấy thông tin đường dẫn của hình ảnh và file trong điện thoại cùng với tên sách và tác giả lưu vào SQLite
- Truyền đường dẫn file sang trang đọc sách

## 18. Màn hình “Tủ sách” (Nguyễn Thị Ngọc Hân)

o Giao diện người dùng



### Hình 23 Màn hình “Tủ sách”

#### o Yêu cầu chức năng

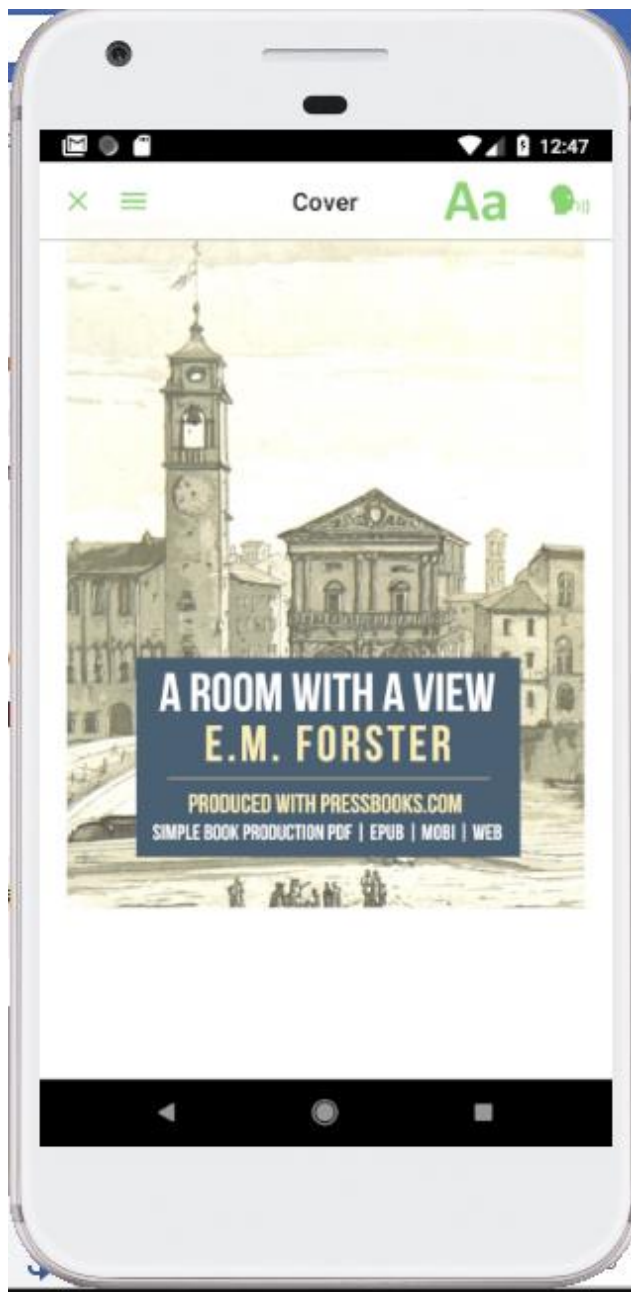
Tiêu đề	Mô tả	Qui trình	Kết quả
Màn hình Tủ sách	Màn hình chính gồm: 1 RecyclerView 1 ToolBar 1 Button	Thiết kế	Hoàn thành
Button “back”	Trở về trang chủ	Click để quay lại	Hoàn thành
RecyclerView “Tủ sách”	Hiển thị danh sách các sách trong SQLite	View	Hiển thị danh sách các sách trong SQLite
ToolBar “Tủ sách”	Chuyển đến các trang chính của app	Click	Chuyển đến các trang chính của app

#### o Làm thế nào ?

- Lấy tất cả dữ liệu trong SQLite đưa vào ArrayList
- CardView, RecyclerView lấy dữ liệu từ ArrayList thông qua adapter
- Khi Click vào item trong RecyclerView, truyền đường dẫn lưu file sách đến trang đọc sách

### 19. Màn hình “Đọc sách” (Nguyễn Thị Ngọc Hân & Trần Gia Bảo Ngọc)


#### o Giao diện người dùng



**Hình 24 Màn hình “Đọc sách”**

**o Yêu cầu chức năng**

Tiêu đề	Mô tả	Quy trình	Kết quả
Màn hình đọc sách	Màn hình chính có:	Thiết kế	Hoàn thành

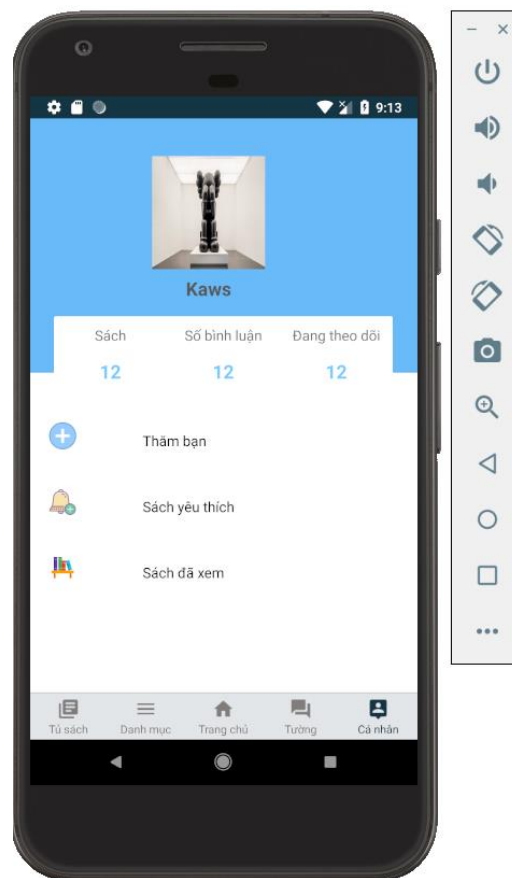
Font	Người dùng tự điều chỉnh cỡ chữ	Nhấn vào “Aa” hiện ra thanh điều chỉnh cỡ chữ	Hoàn thành
Highlights	Copy, đánh dấu, chia sẻ dòng hoặc đoạn văn trong sách	Bôi đen dòng hoặc đoạn văn yêu thích, nhấn thêm một lần rồi tùy chọn theo ý muốn	Hoàn thành
Chọn chương	Lựa chọn chương để đọc	Nhấn biểu tượng chọn chương theo ý muốn 	Hoàn thành

### o Làm thế nào ?

- Lấy đường dẫn file sách từ getIntent
- Khai báo biến thư viện FolioReader
- Các sự kiện lắng nghe cho biến vừa khai báo
- Dùng hàm openBook() với tham số là đường dẫn file
- Thư viện FolioReader sẽ gọi 1 activity khác để hiển thị sách

## 20. Màn hình “Cá nhân” (Tôn Thị Thanh Vân)

### o Giao diện người dùng



**Hình 25 Màn hình “Trang cá nhân”**

### o Yêu cầu chức năng

Tiêu đề	Mô tả	Quy trình	Kết quả
Trang cá nhân	Màn hình có: ImageView, TextView, CardView	Hiển thị	Chưa hoàn thành
ImageView	Hiển thị ảnh đại diện của user	Click	Chưa hoàn thành
Text View	Hiển thị tên user	Hiển thị	Chưa hoàn thành
CardView	Hiển thị sách trong tủ, số bình luận, và người theo dõi của user	Hiển thị	Chưa hoàn thành

### o Làm thế nào ?

- Từ Trang chủ, ta nhấp vào Cá nhân ở Bottom Navigation View để chuyển sang giao diện của Cá nhân
- ImageView hiển thị hình ảnh của user, TextView hiển thị tên user
  - Sách yêu thích và Sách đã xem sẽ hiển thị sách mà user đánh dấu là yêu thích, còn những sách user click vào xem qua thì sẽ được lưu và Sách đã xem

# CHƯƠNG 3: CÀI ĐẶT

## 1. Trang đăng kí

### Sự kiện click button đăng ký

```
public void onClick(View v) {
    progressDialog.setMessage("Đang xử lý");
    progressDialog.setIndeterminate(true);
    progressDialog.setIndeterminateDrawable(getDrawable(R.drawable.progressbar_dialog));

    String mail=edemail.getText().toString();
    String mk=edmatkhau.getText().toString();
    String nlmk=ednhaplaiMK.getText().toString();
    final String ten= edtenTk.getText().toString();

    if(ten.trim().length()==0){
        Toast.makeText(DangkyActivity.this,"Vui lòng nhập tên tài khoản",Toast.LENGTH_SHORT).show();
    }else if(mail.trim().length()==0) {
        Toast.makeText(DangkyActivity.this, "Vui lòng nhập email", Toast.LENGTH_SHORT).show();
    }else {
        if (!Kiemtraemail(mail)) {
            Toast.makeText(DangkyActivity.this, "Email không hợp lệ", Toast.LENGTH_SHORT).show();
        }
        if (mk.trim().length() == 0) {
            Toast.makeText(DangkyActivity.this, "Vui lòng nhập mật khẩu", Toast.LENGTH_SHORT).show();
        } else if (!nlmk.equals(mk)) {
            Toast.makeText(DangkyActivity.this, "Mật khẩu không khớp", Toast.LENGTH_SHORT).show();
        } else {
            progressDialog.show();
            firebaseAuth.createUserWithEmailAndPassword(mail, mk).addOnCompleteListener((task) -> {
                if (task.isSuccessful()) {
                    progressDialog.dismiss();
                    user=firebaseAuth.getCurrentUser();
                    UserProfileChangeRequest profileUpdate = new UserProfileChangeRequest.Builder()
                        .setDisplayName(ten)
                        .build();

                    user.updateProfile(profileUpdate).addOnCompleteListener((task) -> {
                        if(task.isSuccessful()){
                            Toast.makeText( context: DangkyActivity.this, user.getDisplayName() , Toast.LENGTH_SHORT).show();
                        }
                    });
                    String id = user.getId();
                    Toast.makeText( context: DangkyActivity.this, id , Toast.LENGTH_SHORT).show();
                    //For adding into database
                    databaseReference = databaseReference.child("thanhviens").child(id);
                    databaseReference.child("ten").setValue(ten);
                    databaseReference.child("anhdaidien").setValue("thanhvien/default_user.jpg");
                    // Toast.makeText(DangkyActivity.this, R.string.dangkythanhcong, Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(DangkyActivity.this, "Email này đã đăng ký rồi", Toast.LENGTH_SHORT).show();
                }
            });
        }
    }
}
```

## 2. Trang đăng nhập

### Xử lý sự kiện trang login



```

@Override
public void onClick(View v) {
    int id=v.getId();
    switch (id){
        case R.id.button_dangky:
            Intent dangky=new Intent(LoginActivity.this,DangkyActivity.class);
            startActivity(dangky);
            break;
        case R.id.button_boquadangnhap:
            Intent boquadangnhap=new Intent(LoginActivity.this,MainActivity.class);
            startActivity(boquadangnhap);
            break;
        case R.id.text_quenmk:
            Intent guimail=new Intent(LoginActivity.this,ForgetPassActivity.class);
            startActivity(guimail);
            break;
        case R.id.btn_dangnhap:
            Dangnhap();
            break;
        case R.id.btn_dangnhapface:
            dangnhapface=1;
            dangnhapFacebook();
            break;
    }
}
}

```

Sau đó viết 2 hàm đăng nhập facebook và đăng nhập bằng email

### 3. Trang quên mật khẩu

Xử lý sự kiện click button gửi mail

```

guimail.setOnClickListener((v) → {
    String email=editemail.getText().toString();
    boolean kiemtraemail= Kiemtraemail(email);
    if(kiemtraemail){
        firebaseAuth.sendPasswordResetEmail(email).addOnCompleteListener((task) → {
            if(task.isSuccessful()){
                Toast.makeText(ForgetPassActivity.this,"Vui lòng kiểm tra mail của bạn",Toast.LENGTH_SHORT).show();

                alertDialog.setTitle("Thông báo");
                alertDialog.setIcon(R.mipmap.ic_launcher);
                alertDialog.setMessage("Vui lòng kiểm tra email");
                alertDialog.setPositiveButton(text: "OK", (dialog, which) → {
                    Toast.makeText(getBaseContext(), text: "OK",
                        Toast.LENGTH_SHORT).show();
                });
            }else{
                Toast.makeText(ForgetPassActivity.this,"Thông tin sai",Toast.LENGTH_SHORT).show();
            }
        });
    }else{
        Toast.makeText(ForgetPassActivity.this,"Email không hợp lệ",Toast.LENGTH_SHORT).show();
    }
});
boqua.setOnClickListener((v) → {

```

### 4. Trang chủ

```

public void initView(){
    toolbar= (Toolbar) findViewById(R.id.toolbar_trangchu);
    viewPager= (ViewPager) findViewById(R.id.viewpager);
    recyclerView=(RecyclerView) findViewById(R.id.recyclerview);
    progressBar=findViewById(R.id.progressbar_trangchu);
    navigationView=(NavigationView) findViewById(R.id.navigationview_trangchu);
    drawerLayout=(DrawerLayout) findViewById(R.id.drawerlayout_trangchu);
    sliderDots=(LinearLayout) findViewById(R.id.SliderDots);
    timkiem=findViewById(R.id.button_timkiem);
    fragmentManager=getSupportFragmentManager();
    progressDialog=new ProgressDialog( context: MainActivity.this);
    View headerNavi=navigationView.getHeaderView( index: 0);

    username=headerNavi.findViewById(R.id.text_user);
    email=headerNavi.findViewById(R.id.emailuser);

    anhuser=headerNavi.findViewById(R.id.anhuser);
    firebaseAuth=FirebaseAuth.getInstance();
    user=firebaseAuth.getCurrentUser();
    Menu menu = navigationView.getMenu();
    MenuItem navTime = menu.findItem(R.id.thaydoi_canhan);
    MenuItem dangxuat=menu.findItem(R.id.btn_dangxuat);
}

```

Xử lý Navigation drawer menu trên trang chủ:

```

navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        selectItemDrawer(item);
        return true;
    }
});

```

```

public void selectItemDrawer(MenuItem menuItem) {
    Intent intent;
    switch(menuItem.getItemId()){
        case R.id.thaydoi_canhhan:
            if(user!=null) {
                intent = new Intent( packageContext: MainActivity.this, SuathongtinCanhanNav.class);
                startActivity(intent);
            }else{
                intent = new Intent( packageContext: MainActivity.this, LoginActivity.class);
                startActivity(intent);
            }
            break;
        case R.id.caidat_giaodien:
            intent =new Intent( packageContext: MainActivity.this,CaidatGiaodienNav.class);
            startActivity(intent);
            break;
        case R.id.ngonngu:
            Toast.makeText( context: MainActivity.this, text: "chua hỗ trợ",Toast.LENGTH_SHORT).show();
            break;
        case R.id.caidat_time:
            activityTimer();
            break;
        case R.id.hotro_ungdung:
            intent =new Intent( packageContext: MainActivity.this,HuongdanHotroNav.class);
            startActivity(intent);

            break;
        case R.id.gioithieu_lienhe:
            intent =new Intent( packageContext: MainActivity.this,LienheGioithieuNav.class);
            startActivity(intent);

            break;
        case R.id.btn_dangxuat:
            if(user!=null){
                firebaseAuth.signOut();
                intent =new Intent( packageContext: this,MainActivity.class);
                startActivity(intent);
                Toast.makeText( context: MainActivity.this, text: "Bạn đã đăng xuất",Toast.LENGTH_SHORT).show();
            }else{
                intent =new Intent( packageContext: MainActivity.this,LoginActivity.class);
                startActivity(intent);
            }

            break;
        case R.id.face_theodoi:

            //Toast.makeText(MainActivity.this,"chua hỗ trợ",Toast.LENGTH_SHORT).show();
            Intent viewIntent =
                new Intent( action: "android.intent.action.VIEW",
                    Uri.parse("https://www.facebook.com/READbooks.vn/"));
            startActivity(viewIntent);
            break;
        default:
            Toast.makeText( context: MainActivity.this, text: "Sai",Toast.LENGTH_SHORT).show();
            break;
    }
    drawerLayout.closeDrawers();
    menuItem.setChecked(false);
    if(menuItem.getItemId()==R.id.caidat_time) drawerLayout.openDrawer(Gravity.START);
}

```

## 5. Trang sửa thông tin

Load thông tin user đã login

```

public void loaddata(){
    if(user!=null) {
        String tentk = user.getDisplayName();
        String emailu=user.getEmail();
        tendanhnhap.setText(tentk);
        email.setText(emailu);
        final DatabaseReference database=FirebaseDatabase.getInstance().getReference().child("thanhviens").child(user.getUid()).ch

        database.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(DataSnapshot dataSnapshot) {
                String anh= dataSnapshot.getValue().toString();
                Toast.makeText(SuathongtinCanhanNav.this,anh,Toast.LENGTH_SHORT).show();
                StorageReference storagehinhanh = FirebaseStorage.getInstance().getReference().child(anh);
                long ONE_MEGABYTE = 1024 * 1024;
                storagehinhanh.getBytes(ONE_MEGABYTE).addOnSuccessListener(onSuccess(bytes) -> {
                    Bitmap bitmap = BitmapFactory.decodeByteArray(bytes, 0, bytes.length);
                    anhdaidien.setImageBitmap(bitmap);
                });
            }
            @Override
            public void onCancelled(DatabaseError databaseError) {

            }
        });
    }
}
}

```

## Update thông tin user

```

public void updatedata(){
    DatabaseReference databaseuptenanh=FirebaseDatabase.getInstance().getReference().child("thanhviens").child(user.getUid());
    if(chonanh==true){
        String path="thanhvien/";
        Calendar calendar=Calendar.getInstance();
        path=path+tendanhnhap.getText().toString()+calendar.getTimeInMillis()+".jpg";
        StorageReference mountainsRef=storageRef.child(path);
        anhdaidien.setDrawingCacheEnabled(false);
        Bitmap bitmap = anhdaidien.getDrawingCache();
        ByteArrayOutputStream baos = new ByteArrayOutputStream();
        bitmap.compress(Bitmap.CompressFormat.JPEG, 100, baos);
        byte[] data = baos.toByteArray();

        UploadTask uploadTask = mountainsRef.putBytes(data);
        uploadTask.addOnFailureListener((exception) -> {
            Toast.makeText(context: SuathongtinCanhanNav.this, text: "Lỗi up file",Toast.LENGTH_SHORT).show();
        }).addOnSuccessListener((OnSuccessListener) (taskSnapshot) -> {
            // taskSnapshot.getMetadata() contains file metadata such as size, content-type, and download URL.
            Uri downloadUrl = taskSnapshot.getDownloadUrl();
            Toast.makeText(context: SuathongtinCanhanNav.this, text: "Sửa thành công",Toast.LENGTH_SHORT).show();
            // Log.d( tag: "AAA", msg: downloadUrl+"");
        });

        databaseuptenanh.child("anhdaidien").setValue(path);
    }else{
        // Toast.makeText(context: SuathongtinCanhanNav.this, text: "Chưa chọn ảnh",Toast.LENGTH_SHORT).show();
    }
}

```

```

final String tendn=tendanhnhap.getText().toString();
String pass=mk.getText().toString().trim();
String email=email.getText().toString().trim();
String pass2=nhaplaimk.getText().toString().trim();
if(tendn.trim().length()==0){
    Toast.makeText(SuathongtinCanhanNav.this, "Vui lòng nhập tên tài khoản", Toast.LENGTH_SHORT).show();
}else {if(email.length()==0){
    Toast.makeText(SuathongtinCanhanNav.this, "Vui lòng nhập email", Toast.LENGTH_SHORT).show();
}else{ if (pass.trim().length() == 0) {
    Toast.makeText(SuathongtinCanhanNav.this, "Vui lòng nhập mật khẩu", Toast.LENGTH_SHORT).show();
    } else {
        if (!pass2.trim().equals(pass.trim())) {
            Toast.makeText(SuathongtinCanhanNav.this, "Mật khẩu không khớp", Toast.LENGTH_SHORT).show();
        }
        else {

            UserProfileChangeRequest profileUpdate = new UserProfileChangeRequest.Builder()
                .setDisplayName(tendn)
                .setPhotoUri(Uri.EMPTY)
                .build();
            user.updateEmail(email);
            user.updateProfile(profileUpdate).addOnCompleteListener((task) -> {
                if (task.isSuccessful())
                    databaseReference = FirebaseDatabase.getInstance().getReference().child("thanhviens").child(user.getId());
                    databaseReference.child("ten").setValue(tendn);
                    Toast.makeText(SuathongtinCanhanNav.this, "Đã cập nhật thông tin", Toast.LENGTH_SHORT).show();
                });
            user.updatePassword(pass);
        }
    }
}
}

```

## 6. Trang hướng dẫn hỗ trợ

### Lấy danh sách câu hỏi từ firebase

```

databaseReference= FirebaseDatabase.getInstance().getReference().child("cauhoi");
//exListView.setIndicatorBounds(width - GetPixelFromDips(50), width - GetPixelFromDips(10));

databaseReference.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        int i=0;
        for(DataSnapshot data:dataSnapshot.getChildren()){
            listdataQuestion.add(data.child("hoi").getValue().toString());
            listdata.put(listdataQuestion.get(i),data.child("traloi").getValue().toString());
            i++;
            huongdanHotroAdapter=new HuongdanHotroAdapter (HuongdanHotroNav.this,listdataQuestion,listdata);
            exListView.setAdapter(huongdanHotroAdapter);
        }
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }

});

```

```

blic void initView(){

```

## 7. Trang post bài

### Hiện thị danh sách bài post từ firebase

```

public void addListView() { //tieude,anhuser,tenuser,ngaytao,sotl
    baiPostModel=new BaiPostModel();
    baiPostList=new ArrayList<>();

    baiPostAdapter=new BaiPostAdapter(getActivity(),R.layout.item_postbai_listview,baiPostList);
    listView.setAdapter(baiPostAdapter);

    final BaiPostInterface baiPostInterface=(baiPostModel) -> {
        baiPostList.add(baiPostModel);
        baiPostAdapter.notifyDataSetChanged();
        Collections.sort(baiPostList, new Comparator<BaiPostModel>() {
            DateFormat f=new SimpleDateFormat( pattern: "dd-MM-yyyy HH:mm:ss");
            @Override
            public int compare(BaiPostModel o1, BaiPostModel o2) {
                try {
                    return f.parse(o2.getNgaytao()).compareTo(f.parse(o1.getNgaytao()));
                } catch (ParseException e) {
                    e.printStackTrace();
                }
                return 0;
            }
        });
    };
    baiPostModel.getDataRoot().addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            baiPostList.clear();
            baiPostModel.getdataPost(baiPostInterface);
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {
    }
    });
}

```

## Xử lý sự kiện click float button

```

View view=inflater.inflate(R.layout.fragment_forum_postbai,container, attachToRoot: false);
listView=view.findViewById(R.id.listbaipost);

add=view.findViewById(R.id.add_post);
add.setOnClickListener((v) -> {
    FirebaseAuth firebaseAuth=FirebaseAuth.getInstance();
    FirebaseUser firebaseUser=firebaseAuth.getCurrentUser();
    if(firebaseUser!=null){
        Intent intent = new Intent(getContext(), AddPostActivity.class);
        startActivity(intent);
    }else {
        Intent intent = new Intent(getContext(), LoginActivity.class);
        startActivity(intent);
    }
});
listView.setOnItemClickListener((parent, view, position, id) -> {
    //Toast.makeText(getContext(),baiPostList.get(position).getMabaipost()+" ",Toast.LENGTH_SHORT).show();
    Log.d( tag: "kiemtraintent",baiPostList.get(position).getMabaipost());
    Intent intent=new Intent(getContext(),BinhluanPostActivity.class);
    BaiPostModel exa=baiPostList.get(position);
    BinhluanPostModel first=new BinhluanPostModel(exa.getThanhvienModel(),exa.getNoidungpost(),exa.getNgaytao(),exa.getMabaipost());
    intent.putExtra( name: "modelfirst",first);
    Log.d( tag: "check5", msg: exa.getSotraloi()+"");
    intent.putExtra( name: "sotl",String.valueOf(exa.getSotraloi()));
    intent.putExtra( name: "title",exa.getTieude());
    intent.putParcelableArrayListExtra( name: "postModel", (ArrayList<? extends Parcelable>) baiPostList.get(position).getListBL());
    startActivity(intent);
});

```

## 8. Trang bài post mới

### Xử lý sự kiện đăng bài

```

public void actionPostadd() {
    user= FirebaseAuth.getInstance().getCurrentUser();
    postbai.setOnClickListeners(onClick(v) → {
        Calendar calendar=Calendar.getInstance();
        DateFormat dateFormat=new SimpleDateFormat("dd-MM-yyyy HH:mm:ss");
        String strDate=dateFormat.format(calendar.getTime()).toString();
        Toast.makeText(AddPostActivity.this,strDate,Toast.LENGTH_SHORT).show();
        String tieudePost=tieude.getText().toString();
        String noidungPost=noidung.getText().toString();
        databaseReference=FirebaseDatabase.getInstance().getReference().child("baiposts").push();
        databaseReference.child("tieude").setValue(tieudePost);
        databaseReference.child("mauser").setValue(user.getId());
        databaseReference.child("noidungpost").setValue(noidungPost);
        databaseReference.child("sotraloi").setValue(0);
        databaseReference.child("ngaytao").setValue(strDate);
        Toast.makeText(AddPostActivity.this,"Đăng thành công",Toast.LENGTH_SHORT).show();

    });
}
public void actionBar(){
    toolbar.setTitle("Đăng bài");
    toolbar.setBackgroundColor(Constant.color);
    toolbar.setNavigationIcon(R.drawable.ic_arrow_back_black_24dp);
    toolbar.setNavigationOnClickListener(onClick(view) → {
        Intent intent =new Intent(AddPostActivity.this,PostbaiActivity.class);
        startActivity(intent);
    });
}

```

## 9. Danh mục sách

Tạo class Danh muc noibat Activity để set sự kiện hiển thị các hình ảnh lấy từ Firebase về và hiển thị lên GridView

```

public class Danh muc noibat Activity extends AppCompatActivity {
    private Context mContext = Danh muc noibat Activity.this;
    private static final int ACTIVITY_NUM=1;
    Button btnTruyencuoi;
    DanhMuc danhMuc;
    GridView gridView;
    ArrayList<DanhMuc> danhMucList;
    Danh muc noibat Adapter adapter;
    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setTheme(Constant.theme);
        setContentView(R.layout.activity_danhmuc);
        gridView=findViewById(R.id.gridViewDanhmuc);
        setupBottomNavigationView();

        gridView.setOnItemClickListener((parent, view, position, id) → {
            Intent intent= new Intent(Danh muc noibat Activity.this,Sach theodanh muc Activity.class);
            intent.putExtra( name: "keydanhmuc", danhMucList.get(position).getMatheloi().toString());
            //Toast.makeText(Danh muc noibat Activity.this,danhMucList.get(position).getMatheloi().toString(),Toast.LENGTH_SHORT).show();
            startActivity(intent);
        });
    }
    addGridView();
}

```

Tạo lớp addGridView để thêm các hình ảnh ta lấy về được trên Firebase vào GridView

```

public void setupBottomNavigationView(){
    BottomNavigationViewEx bottomNavigationViewEx=findViewById(R.id.bottom_Na);
    BottomNavigationViewHelper.setupBottomNavigationView(bottomNavigationViewEx);
    BottomNavigationViewHelper.enableNavigation( context: Danh muc noibat Activity.this, callingActivity: this, bottomNavigationViewEx);
    Menu menu=bottomNavigationViewEx.getMenu();
    MenuItem menuItem=menu.getItem(ACTIVITY_NUM);
    menuItem.setChecked(true);
}

public void addGridView() { // tieude, anh user, ten user, ngay tao, sot l
    danhMuc=new DanhMuc();
    danhMucList=new ArrayList<>();

    adapter=new Danh muc noibat Adapter( context: this, R.layout.item_danh muc gridView, danhMucList);
    gridView.setAdapter(adapter);

    final Danh muc Interface danh muc Interface= (danh muc) -> {
        danhMucList.add(danh muc);
        adapter.notifyDataSetChanged();
    };

    danhMuc.getDataTheLoai().addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            danhMucList.clear();
            danhMuc.getdataTheLoai(danh muc Interface);
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {

        }
    });
}

```

danhMuc.getDataTheLoai().addValueEventListener giúp chúng ta lấy tên thể loại và ảnh theo key mà ta tạo ra

```

public void getdataTheLoai(final Danh muc Interface danh muc Interface){
    ValueEventListener valueEventListener=new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            DataSnapshot dataSnapshotTheLoai=dataSnapshot.child("theLoais");

            for (DataSnapshot valueTL:dataSnapshotTheLoai.getChildren()){
                DanhMuc danhMuc=new DanhMuc();
                danhMuc=valueTL.getValue(DanhMuc.class);
                danhMuc.matheloai=valueTL.getKey();
                //Log.d("kiemtratheloai", danhMuc.getTen()+"");
                Log.d( tag: "matheloai", msg: danhMuc.getMatheloai()+" ");
                Log.d( tag: "kiemtratheloai", msg: danhMuc.getAnhtheloai()+"");
                danh muc Interface.getdanh muc (danhMuc);
            }

            @Override
            public void onCancelled(DatabaseError databaseError) {

            }
        }
    };
    dataTheLoai.addListenerForSingleValueEvent(valueEventListener);
}

```



Và DanhMucInterface được tạo ra:

```
package com.example.user.lvhreadbookapp.Models;

public interface DanhMucInterface {
    void getDanhMuc(DanhMuc danhMucModel);
}
```

Trước đó, ta tạo adapter tên DanhMucnoibatAdapter để lấy ảnh và hình ảnh từ Firebase

```
public class DanhMucnoibatAdapter extends ArrayAdapter<DanhMuc> {
    private List<DanhMuc> danhMucList;
    private int resource;
    private Activity context;

    public DanhMucnoibatAdapter(@NonNull Activity context, @LayoutRes int resource, @NonNull List<DanhMuc> objects) {
        super(context, resource, objects);
        this.context = context;
        this.resource = resource;
        danhMucList = objects;
    }
}
```

```
@NonNull
@Override
public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
    LayoutInflater inflater = context.getLayoutInflater();

    View v = inflater.inflate(resource, root null);
    TextView txtTen = (TextView) v.findViewById(R.id.tenDanhMuc);
    final ImageView imgHinh = (ImageView) v.findViewById(R.id.hinhDanhMuc);

    txtTen.setText(danhMucList.get(position).getTen());
    StorageReference storagehinhanh= FirebaseStorage.getInstance().getReference().child(danhMucList.get(position).getAnhtheloi().toString());
    long ONE_MEGABYTE=1024*1024;
    storagehinhanh.getBytes(ONE_MEGABYTE).addOnSuccessListener((OnSuccessListener) (bytes) -> {
        Bitmap bitmap= BitmapFactory.decodeByteArray(bytes, offset 0,bytes.length) ;
        imgHinh.setImageBitmap(bitmap);
    });
    return v;
}
```

DanhMucnoibatAdapter được extends từ ArrayAdapter<DanhMuc>: <DanhMuc> để khai báo và tạo getter, setter và constructor cho các biến được khai báo

```

public class DanhMuc {
    String matheloai;
    String ten;
    String anhhtheloai;
    private DatabaseReference dataTheloi;

    public String getMatheloai() { return matheloai; }

    public void setMatheloai(String matheloai) { this.matheloai = matheloai; }

    public String getTen() { return ten; }

    public void setTen(String ten) { this.ten = ten; }

    public String getAnhhtheloai() { return anhhtheloai; }

    public void setAnhhtheloai(String anhhtheloai) { this.anhhtheloai = anhhtheloai; }

    public DatabaseReference getDataTheloi() { return dataTheloi; }

    public void setDataTheloi(DatabaseReference dataTheloi) { this.dataTheloi = dataTheloi; }

    public DanhMuc() { dataTheloi= FirebaseDatabase.getInstance().getReference(); }
}

```

## 10. Sách theo danh mục

```

public class SachtheodanhmucActivity extends AppCompatActivity {
    private Context mContext = SachtheodanhmucActivity.this;
    SachtheodanhmucAdapter adapter;
    ArrayList<SachModel> listsach;
    String key;
    SachModel sachModel;
    GridView gridView;
    Toolbar toolbar;
    DanhMuc danhMuc;

    @Override
    protected void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setTheme(Constant.theme);
        setContentView(R.layout.activity_sachtheodanhmuc);
        gridView=findViewById(R.id.grid_sachtheodm);
        key= getIntent().getStringExtra( name: "keydanhmuc");//khóa gửi qua
        Toast.makeText( context: this,key,Toast.LENGTH_SHORT).show();
        gridView.setOnItemClickListener((parent, view, position, id) → {
            Intent intent= new Intent(SachtheodanhmucActivity.this,SachtheodanhmucActivity.class);
            intent.putExtra( name: "keysach",listsach.get(position).getMasach());
            intent.putExtra( name: "keytheloai",listsach.get(position).getMatheloai());
            startActivity(intent);
        });
        actionBar();
        addGridView();
    }
}

```

Hàm actionBar() dùng để nhấn nút trở về, từ màn hình sách của thể loại quay về Danh mục theo thể loại

```

public void actionBar(){
    toolbar=findViewById(R.id.toolbar_nav);
    toolbar.setTitle("Danh sách sách");
    toolbar.setBackgroundColor(Constant.color);
    toolbar.setNavigationIcon(R.drawable.ic_arrow_back_black_24dp);
    toolbar.setNavigationOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent =new Intent(SachtheodanhmucActivity.this,DanhmucnoibatActivity.class);
            startActivity(intent);
        }
    });
}
}

```

Hàm gettheloai() dùng để lấy sách trên Firebase theo thể loại mà chúng ta lưu và trước đó cũng phải tạo Interface

```

package com.example.user.lvhnreadbookapp.Models;

public interface SachInterface {
    void getSachModel(SachModel sachModel);
}

```

```

public void gettheloai(){
    danhMuc=new DanhMuc();
    listsach=new ArrayList<>();

    adapter=new SachtheodanhmucAdapter( context: this,R.layout.gridview_row,listsach);
    gridView.setAdapter(adapter);

    final SachInterface sachInterface= (sachModel) -> {
        listsach.add(sachModel) ;
        adapter.notifyDataSetChanged();
        sortDay(listsach);
    };
    sachModel.getDataSach().addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            listsach.clear();
            sachModel.getdataMatheloai(sachInterface,key);
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {

        }
    });
}
}

```

Trong gettheloai() ta có sử dụng sachModel.getDataSach() dùng để lấy dữ liệu sách và ảnh theo đúng tên thể loại click vào

```

public void getdataSach(final SachInterface sachInterface) {
    ValueEventListener valueEventListener=new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            DataSnapshot dataSnapshotSach=dataSnapshot.child("sachs");
            for (DataSnapshot valueSach:dataSnapshotSach.getChildren()){//cac the loai
                String theloai=valueSach.getKey();//matheloai
                // Log.d("checksach",theloai);
                for(DataSnapshot value:valueSach.getChildren()){
                    SachModel sachModel=value.getValue(SachModel.class);
                    sachModel.setMatheloai(theloai);
                    sachModel.setMasach(value.getKey());
                    //Log.d("checksach",value.getKey());
                    sachInterface.getSachModel(sachModel);
                }
            }
        }
    };

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }

};
dataSach.addListenerForSingleValueEvent(valueEventListener);

```

Và hàm getdataMatheloai() dùng để duyệt sách theo key

```

public void getdataMatheloai(final SachInterface sachInterface, final String key){
    final String keyTL=key;
    ValueEventListener valueEventListener=new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            DataSnapshot dataSnapshotSach=dataSnapshot.child("sachs").child(keyTL);
            for (DataSnapshot valueSach:dataSnapshotSach.getChildren()){//Các cuốn sách.
                SachModel sachModel=valueSach.getValue(SachModel.class);
                sachModel.setMatheloai(keyTL);
                sachModel.setMasach(valueSach.getKey());
                Log.d( tag: "masach", valueSach.getKey());
                Log.d( tag: "matheloaisach", sachModel.getMatheloai());
                sachInterface.getSachModel(sachModel);
            }
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {

        }
    };

    dataSach.addListenerForSingleValueEvent(valueEventListener);
}

```

Cũng như Danh Mục Theo Thê Loại, phải tạo 1 adapter để lấy thông tin như ảnh sách và tên của sách đó hiển thị lên custom gridview, và sau đó là load lên hết lên GridView

```
public class SachtheodanhmucAdapter extends ArrayAdapter<SachModel> {
    private List<SachModel> sachModelList;
    private int resource;
    private Activity context;

    public SachtheodanhmucAdapter(@NonNull Activity context, @LayoutRes int resource, @NonNull List<SachModel> objects) {
        super(context, resource, objects);
        this.context = context;
        this.resource = resource;
        sachModelList = objects;
    }

    @NonNull
    @Override
    public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
        LayoutInflater inflater = context.getLayoutInflater();

        View v = inflater.inflate(resource, root null);
        TextView txtTen = (TextView) v.findViewById(R.id.textTensach);
        final ImageView imgHinh = (ImageView) v.findViewById(R.id.imageHinh);

        txtTen.setText(sachModelList.get(position).getTensach());
        StorageReference storagehinh = FirebaseStorage.getInstance().getReference().child(sachModelList.get(position).getHinh().toString());
        long ONE_MEGABYTE = 1024 * 1024;
        storagehinh.getBytes(ONE_MEGABYTE).addOnSuccessListener(new OnSuccessListener<Bytes>() {
            @Override
            public void onSuccess(Bytes bytes) {
                Bitmap bitmap = BitmapFactory.decodeByteArray(bytes.toByteArray(), 0, bytes.length);
                imgHinh.setImageBitmap(bitmap);
            }
        });
        return v;
    }
}
```

List trong Sách theo thể loại này sử dụng trong class SachModel như bên dưới

```
public class SachModel {
    private String masach, matheloai, file, tensach, ngaydang, tacgia, gioithieu, hinh;
    long danhgia, luotxem, luotthich, namxb;
    private DatabaseReference dataSach;

    public SachModel() { dataSach = FirebaseDatabase.getInstance().getReference(); }

    public String getMasach() { return masach; }

    public void setMasach(String masach) { this.masach = masach; }

    public String getMatheloai() { return matheloai; }

    public void setMatheloai(String matheloai) { this.matheloai = matheloai; }

    public String getFile() { return file; }

    public void setFile(String file) { this.file = file; }

    public String getTensach() { return tensach; }

    public void setTensach(String tensach) { this.tensach = tensach; }

    public String getNgaydang() { return ngaydang; }

    public void setNgaydang(String ngaydang) { this.ngaydang = ngaydang; }
}
```

```

public String getTacgia() { return tacgia; }

public void setTacgia(String tacgia) { this.tacgia = tacgia; }

public String getGioithieu() { return gioithieu; }

public void setGioithieu(String gioithieu) { this.gioithieu = gioithieu; }

public String getHinhanh() { return hinhanh; }

public void setHinhanh(String hinhanh) { this.hinhanh = hinhanh; }

public long getDanhgia() { return danhgia; }

public void setDanhgia(long danhgia) { this.danhgia = danhgia; }

public long getLuotxem() { return luotxem; }

public void setLuotxem(long luotxem) { this.luotxem = luotxem; }

public long getLuotthich() { return luotthich; }

public void setLuotthich(long luotthich) { this.luotthich = luotthich; }

public long getNamxb() { return namxb; }

public void setNamxb(long namxb) { this.namxb = namxb; }

public DatabaseReference getDataSach() {
    return dataSach;
}

public void setDataSach(DatabaseReference dataSach) { this.dataSach = dataSach; }

```

Class SachModel này cũng là lớp mà ta đã tạo các hàm getdataSach(), getdatatheloai() và getdataMatheloai()

## 12. Trang tìm kiếm

### Sự kiện submit SearchView

```
svTimKiem.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
    @Override
    public boolean onQueryTextSubmit(final String query) {
        data.clear();
        adapter.notifyDataSetChanged();
        count=0;
        if (query.trim().length() !=0){
            database.child("sachs").addListenerForSingleValueEvent(new ValueEventListener() {
                @Override
                public void onDataChange(DataSnapshot dataSnapshot) {
                    for (DataSnapshot singleSnapshot: dataSnapshot.getChildren()){
                        String name=(String) singleSnapshot.child("TenSach").getValue();
                        String theloai=singleSnapshot.getKey();
                        for(DataSnapshot theloaiSnapshot: singleSnapshot.getChildren()){
                            String s1=toASCII(theloaiSnapshot.child("tensach").getValue().toString().toLowerCase());
                            String s2=toASCII(query.toLowerCase());
                            if(s1.contains(s2)==true){
                                for(int i=0;i<data.size();i++){
                                    if(data.get(i).getMasach().equalsIgnoreCase(theloaiSnapshot.getKey().toString())){
                                        flag=true;
                                    }
                                }
                                if(flag==true){
                                    flag=false;
                                    continue;
                                }
                                count++;
                                SachModel sachModel=theloaiSnapshot.getValue(SachModel.class);
                                sachModel.setMatheloai(theloai);
                                sachModel.setMasach(theloaiSnapshot.getKey());
                                data.add(sachModel);
                                adapter.notifyDataSetChanged();
                                String s="Tim thấy "+String.valueOf(count)+" kết quả";
                            }
                        }
                    }
                }
            });
        }
    }
});
```

SearchActivity > onCreate() > new OnQueryTextListener > onQueryTextSubmit() > new ValueEventListener > onDataChange()

### Sự kiện chuyển sang trang chi tiết sách

```
String search;
public void chuyen(View v){
    SachModel sachModel=new SachModel();
    int pos=lvTimKiem.getPositionForView(v);
    sachModel=(SachModel) lvTimKiem.getItemAtPosition(pos);
    Intent intent=new Intent( packageContext: SearchActivity.this,ChitietsachActivity.class);
    intent.putExtra( name: "keysach",sachModel.getMasach());
    intent.putExtra( name: "keytheloai",sachModel.getMatheloai());
    startActivity(intent);
}
```

### 13. Trang chi tiết sách

Cơ sở dữ liệu SQLite thông tin sách trong máy

```
-
public void onCreate(SQLiteDatabase db) {
    db.execSQL("CREATE TABLE " + TB_NAME + "( " +
        TB_COL_ID + " TEXT PRIMARY KEY, " +
        TB_COL_NAME + " TEXT, " +
        TB_COL_TACGIA + " TEXT, " +
        TB_COL_LINK_ANH + " TEXT, " +
        TB_COL_LINK_FILE + " TEXT )"
    );
}

public void insert(SachModel data){
    db = this.getWritableDatabase();
    ContentValues value=new ContentValues();
    value.put(TB_COL_ID,data.getMasach());
    Log.d( tag: "id",data.getMasach());
    value.put(TB_COL_NAME,data.getTensach());
    value.put(TB_COL_TACGIA,data.getTacgia());
    value.put(TB_COL_LINK_ANH,data.getHinhanh());
    value.put(TB_COL_LINK_FILE,data.getFile());
    db.insert(TB_NAME, nullColumnHack: null,value);
}
```

Lấy intent mã sách, mã thể loại và lấy dữ liệu trên firebase

```
masach=getIntent().getStringExtra( name: "keysach");
matheloai=getIntent().getStringExtra( name: "keytheloai");
myRef= FirebaseDatabase.getInstance().getReference();
myRef=myRef.child("sachs").child(matheloai).child(masach);
myRef.addListenerForSingleValueEvent(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        sachModel=dataSnapshot.getValue(SachModel.class);
        mas=dataSnapshot.getKey();
        tens=dataSnapshot.child("tensach").getValue().toString();
        tacgia=dataSnapshot.child("tacgia").getValue().toString();
        hinh=dataSnapshot.child("hinhanh").getValue().toString();
        file=dataSnapshot.child("file").getValue().toString();
        luotxem=Integer.parseInt(dataSnapshot.child("luotxem").getValue().toString());
        Log.d( tag: "test",mas);
        Log.d( tag: "file",file);
        FirebaseStorage storage = FirebaseStorage.getInstance();
        StorageReference storageRef = storage.getReference();
        StorageReference islandRef =storageRef.child(sachModel.getHinhanh());
        final long ONE_MEGABYTE = 1024 * 1024;
        islandRef.getBytes(ONE_MEGABYTE).addOnSuccessListener((OnSuccessListener) (bytes) -> {
            Bitmap bitmap = BitmapFactory.decodeByteArray(bytes, offset: 0, bytes.length);
            ivAnhBia.setImageBitmap(bitmap);
        }).addOnFailureListener((exception) -> {
            // Handle any errors
        });
    }
});
```

Download hình ảnh và file về máy, lấy đường dẫn và lưu thông tin vào SQLite



```

myRef.child("luotxem").setValue(luotxem);
FirebaseStorage storage = FirebaseStorage.getInstance();
StorageReference storageRef = storage.getReference();
StorageReference pathReference = storageRef.child(hinh);
String[] split=hinh.split( regex: "/" );
final File localHinh= new File(getCacheDir(),split[1]);
linkhinh=localHinh.toString();
pathReference.getFile(localHinh).addOnSuccessListener((OnSuccessListener) (taskSnapshot) -> {
    linkhinh=localHinh.toString();
}).addOnFailureListener((exception) -> {
    Log.e( tag: "firebase ", msg: ";local tem file not created created " +exception.toString()+localHinh.toSt
});
pathReference=storageRef.child(file);
String[] split2=file.split( regex: "/" );
final File localFile= new File(getCacheDir(),split2[1]);
linkfile=localFile.toString();
pathReference.getFile(localFile).addOnSuccessListener((OnSuccessListener) (taskSnapshot) -> {
    linkfile=localFile.toString();
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception exception) {
        Log.e( tag: "firebase ", msg: ";local tem file not created created " +exception.toString()+localFile.toSt
    }
});
SachModel sachModel1=new SachModel();
sachModel1.setMasach(mas);
sachModel1.setTensach(tens);
sachModel1.setTacgia(tacgia);
sachModel1.setHinhanh(linkhinh);
Log.d( tag: "linkhinh", linkhinh);
sachModel1.setFile(linkfile);
Log.d( tag: "file", linkfile);
db.insert(sachModel1);

```

ctivity > onCreate() > new OnClickListner > onClick() > new OnSuccessListener > onSuccess()

## 14. Trang tử sách

- Sử dụng cardview + recyclerview để thiết kế giao diện.

Tạo MyViewHolder làm nơi lưu giữ các view, tránh việc khởi tạo lại view gây mất thời gian

```

public static class MyViewHolder extends RecyclerView.ViewHolder {

    private ImageView imgsach;
    private TextView txtten;
    CardView cardView;

    public MyViewHolder(View itemView) {
        super(itemView);
        imgsach = (ImageView) itemView.findViewById(R.id.imagesach);
        txtten = (TextView) itemView.findViewById(R.id.textviewten);
        cardView = itemView.findViewById(R.id.cardview_id);
    }
}

```

Tạo phương thức onBindViewHolder dùng để gán dữ liệu từ list vào viewHolder.

Set sự kiện click vào item cardview (sách) để chuyển qua trang đọc sách.

```
public void onBindViewHolder(@NonNull MyViewHolder holder, final int position) {
    MyViewHolder viewHolder = (MyViewHolder) holder;
    SachModel sach = list.get(position);
private void addControls() {
    RecyclerView myrv = (RecyclerView) findViewById(R.id.viewtusach);
    toolbar = findViewById(R.id.toolbar_tusach);
    listsach = new ArrayList<>();
    //
    database=SQLiteHelper.getHeler(this);

    recyclerViewAdapter = new RecyclerViewAdapter( context: this,listsach);
    myrv.setLayoutManager(new GridLayoutManager( context: this, spanCount: 3));
    myrv.setAdapter(recyclerViewAdapter);
    database.loadData(listsach);
    recyclerViewAdapter.notifyDataSetChanged();
}
```

Ảnh xạ, lấy dữ liệu trong sqlite hiển thị lên Recyclerview, tùy chỉnh giao diện hiển thị của Recyclerview bằng GridlayoutManager

Kết nối với layout, gọi các phương thức đã tạo.

```
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setTheme(Constant.theme);
    setContentView(R.layout.activity_tusach);
    setupBottomNavigationView();
    addControls();
    ReadData();
    ActionBar();
}
```

## 15. Trang đọc sách

- Sử dụng thư viện FolioReader-Android để cài đặt trang đọc sách

```
private void ActionBar() {
    setSupportActionBar(toolbar);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    toolbar.setNavigationOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
        }
    });
}
```

Khai báo thư viện, set các sự kiện lắng nghe và dùng hàm mở sách

```

folioReader=FolioReader.getInstance(getApplicationContext());
folioReader.setOnHighlightListener(this);
folioReader.setReadPositionListener(this);
String path;
path=getIntent().getStringExtra( name: "path");
folioReader.openBook(path);
intent=path.split( regex: "/" );
getHighlightsAndSave();
getLastReadPosition();

```

Khởi tạo phương thức HighlightsAndSave để cài đặt cho chức năng Highlight.

```

private void getHighlightsAndSave() {
    new Thread((Runnable) () -> {
        ArrayList<HighLight> highlightList = null;
        ObjectMapper objectMapper = new ObjectMapper();
        try {
            highlightList = objectMapper.readValue(loadAssetTextAsString( name: "highlights/highlights_data.json"),
                new TypeReference<List<HighlightData>>() {});
        } catch (IOException e) {
            e.printStackTrace();
        }

        if (highlightList == null) {
            folioReader.saveReceivedHighLights(highlightList, () -> {
                //You can do anything on successful saving highlight list
            });
        }
    }).start();
}

@Override
// Làm điều gì đó với dòng highlight nhận được
public void onHighlight(HighLight highlight, HighLight.HighLightAction type) {
}

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_docsach);
    json=SQLiteJson.getHelper(this);
    json.loadData(arr);
    Log.d( tag: "sizejson",String.valueOf(arr.size()));
    folioReader=FolioReader.getInstance(getApplicationContext()); // tùy chỉnh cấu hình cho folioReader
    folioReader.setOnHighlightListener(this);
    folioReader.setReadPositionListener(this); // tạo triển khai ẩn danh của ReadPositionListener
    String path;
    path=getIntent().getStringExtra( name: "path");
    folioReader.openBook(path);
    intent=path.split( regex: "/" );
    getHighlightsAndSave();
    getLastReadPosition();
}

```

Tiến hành gọi các phương thức đã khởi tạo.

## CHƯƠNG 4. KẾT QUẢ ĐẠT ĐƯỢC

### 4.1 Kết quả đạt được

- Hệ thống cho phép đăng ký bằng email, password, facebook và sửa thông tin tài khoản
- Hệ thống hiển thị danh sách sách mới và đọc nhiều, sách theo thể loại
- Hệ thống cho phép user post bài giao lưu, tìm sách, bình luận bài post.
- Hệ thống cho phép hiển thị danh sách thể loại và sách theo thể loại
- Hệ thống cho phép xem chi tiết sách, tải về và đọc sách, chỉnh sửa font chữ
- Hệ thống có chức năng đổi màu giao diện, hẹn giờ đọc sách, cung cấp thông tin trả lời các câu hỏi thường gặp, giới thiệu và liên hệ

### 4.2 Kết luận

#### Những việc đã làm được:

- Đăng ký, Đăng nhập được
- Hiển thị danh sách sách mới, đọc nhiều, sách theo thể loại
- Hiển thị các danh mục sách, sách theo danh mục
- Xem chi tiết sách, tải sách về xem
- Xem và chỉnh sửa cỡ chữ khi đọc
- Chỉnh sửa giao diện, cung cấp thông tin hỗ trợ, câu hỏi thường gặp

#### Những việc chưa làm được

- Chưa xử lý khi không có internet
- Trang cá nhân chưa hoàn thành
- Chưa lưu vị trí đọc