

BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN

TRƯỜNG ĐẠI HỌC THỦY LỢI



# BÀI TẬP LỚN

HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG

ĐỀ TÀI: Xây dựng ứng dụng RSC (Ứng dụng đọc truyện)

Mã sinh viên	Họ và tên	Ngày sinh	Lớp
2151173761	Phạm Bá Đạt	04/04/2003	63KTPM1
2151173814	Khuất Minh Quang	29/08/2003	63KTPM1
2151173781	Trần Đức Huy	20/01/2003	63KTPM1
2151173823	Nguyễn Thị Thanh Thảo	23/03/2003	63KTPM1

Hà Nội, năm 2024

**BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN  
TRƯỜNG ĐẠI HỌC THỦY LỢI**



# **BÀI TẬP LỚN**

**HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG**

**ĐỀ TÀI: Xây dựng ứng dụng RSC (Ứng dụng đọc truyện)**

Mã sinh viên	Họ và Tên	Ngày Sinh	Điểm	
			Bảng Số	Bảng Chữ
2151173761	Phạm Bá Đạt	04/04/2003		
2151173814	Khuất Minh Quang	29/08/2003		
2151173781	Trần Đức Huy	20/01/2003		
2151173823	Nguyễn Thị Thanh Thảo	23/03/2003		

CÁN BỘ CHẤM THI 1

CÁN BỘ CHẤM THI 2

## LỜI NÓI ĐẦU

Trong bối cảnh phát triển nhanh chóng của công nghệ thông tin, việc nắm vững các kiến thức về phát triển ứng dụng cho các thiết bị di động đóng vai trò quan trọng trong quá trình học tập và ứng dụng vào thực tiễn. Bài tập lớn này được thực hiện nhằm mục tiêu củng cố kiến thức đã học, đồng thời giúp nhóm áp dụng các kiến thức đã học vào giải quyết các vấn đề thực tế.

Trong quá trình thực hiện, nhóm đã nghiên cứu kỹ lưỡng các tài liệu tham khảo và áp dụng các công nghệ, công cụ hiện đại nhất hiện nay như ngôn ngữ lập trình java, ngôn ngữ đánh dấu xml, IDE Android Studio, Firebase để phát triển và hoàn thiện bài tập. Báo cáo này sẽ trình bày chi tiết các bước thực hiện, kết quả đạt được, cũng như những khó khăn và giải pháp đã được áp dụng trong suốt quá trình làm việc.

Mặc dù đã cố gắng hết sức, báo cáo vẫn không tránh khỏi những thiếu sót. Nhóm mong nhận được những góp ý từ giáo viên và các bạn để có thể hoàn thiện hơn về kiến thức và kỹ năng.

Xin chân thành cảm ơn

Bảng phân công công việc

Họ và tên	Công việc
Phạm Bá Đạt	<ul style="list-style-type: none"> <li>• Chức năng đăng nhập</li> <li>• Chức năng đăng kí</li> <li>• Chức năng quên mật khẩu</li> <li>• Chức năng bình luận chương truyện</li> </ul>
Khuất Minh Quang	<ul style="list-style-type: none"> <li>• Chức năng hiển thị trang chủ</li> <li>• Chức năng quản lí thông tin tài khoản và bảo mật</li> </ul>
Trần Đức Huy	<ul style="list-style-type: none"> <li>• Chức năng hiển thị thông tin truyện</li> <li>• Chức năng đọc truyện</li> <li>• Chức năng hiển thị danh sách truyện yêu thích</li> </ul>
Nguyễn Thị Thanh Thảo	<ul style="list-style-type: none"> <li>• Chức năng tìm kiếm: <ul style="list-style-type: none"> <li>• Tìm kiếm theo tên truyện</li> <li>• Tìm kiếm theo tên tác giả</li> <li>• Tìm kiếm theo thể loại</li> </ul> </li> <li>• Chức năng hiển thị danh sách truyện yêu thích nhất</li> </ul>

## MỤC LỤC

CHƯƠNG 1. MÔ TẢ BÀI TOÁN .....	9
1.1. Mô tả bài toán .....	9
1.2. Chức năng chính .....	9
1.3. Yêu cầu phi chức năng .....	10
CHƯƠNG 2. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG .....	11
2.1. Phân tích yêu cầu .....	11
2.2. Thiết kế hệ thống .....	19
2.2.1 Thiết kế kiến trúc(MVC) .....	19
2.2.2. Biểu đồ lớp .....	24
2.3. Triển khai .....	43
2.4. Vận hành và bảo trì .....	43
CHƯƠNG 3. KẾT QUẢ THỰC HIỆN .....	44
3.1. Công nghệ đã sử dụng .....	44
3.2. Tiến độ thực hiện .....	44
<b>KẾT LUẬN .....</b>	<b>45</b>

## MỤC LỤC HÌNH ẢNH

Hình 2.1 Giao diện đăng kí .....	30
Hình 2.2 Giao diện đăng nhập .....	31
Hình 2.3 Giao diện quên mật khẩu .....	32
Hình 2.4 Giao diện trang chủ .....	33
Hình 2.5 Giao diện danh sách truyện mới .....	34
Hình 2.6 Giao diện tìm kiếm .....	35
Hình 2.7 Giao diện tìm kiếm theo bộ lọc .....	36
Hình 2.8 Giao diện hiển thị nội dung chi tiết truyện .....	37
Hình 2.9 Giao diện đọc truyện .....	38
Hình 2.10 Giao diện bình luận truyện theo mỗi chương .....	39
Hình 2.11 Giao diện danh sách truyện được yêu thích nhất .....	40
Hình 2.12 Giao diện lịch sử truyện đã đọc .....	41
Hình 2.13 Giao diện tài khoản người dùng .....	42
Hình 2.14 Giao diện đổi mật khẩu .....	43

## MỤC LỤC BẢNG

Bảng 2.1 Bảng “stories”: Chứa danh sách truyện với thông tin chi tiết về từng truyện...	27
Bảng 2.2 Bảng “users”: Chứa danh sách tài khoản người dùng .....	27
Bảng 2.3 Bảng “comments”: Chứa danh sách các bình luận cho từng chương của truyện .....	28
Bảng 2.4 Bảng “history”: Chứa lịch sử đọc của người dùng, lưu lại chương và trang hiện tại cho từng truyện .....	28
Bảng 2.5 Bảng “story_chapters”: Chứa danh sách các chương của từng truyện, với đường dẫn URL đến nội dung của từng chương .....	28
Bảng 2.6 Bảng “user_favorites”: Chứa danh sách các truyện mà người dùng đã yêu thích .....	29

## BẢNG CÁC TỪ VIẾT TẮT

STT	TỪ VIẾT TẮT	VIẾT ĐẦY ĐỦ
1	CSDL	Cơ sở dữ liệu
2	RSC	Ứng dụng đọc truyện
3	MVC	Model-View-Controller
4	UI	User Interface (Giao diện người dùng)
5	JSON	JavaScript Object Notation
6	IDE	Integrated Development Environment (Môi trường phát triển tích hợp)
7	API	Application Programming Interface (Giao diện lập trình ứng dụng)
8	UX	User Experience (Trải nghiệm người dùng)
9	XML	eXtensible Markup Language (Ngôn ngữ đánh dấu mở rộng)



# CHƯƠNG 1.      MÔ TẢ BÀI TOÁN

## 1.1. Mô tả bài toán

RCS là ứng dụng đọc truyện. App có giao diện thân thiện với người dùng và dễ sử dụng. Chức năng chính của app là cho phép người dùng đọc truyện ngay trên app thay vì phải lên web như thông thường.

## 1.2. Chức năng chính

Ứng dụng RSC cần có các chức năng sau:

- Đăng nhập: Cho phép người dùng nhập thông tin tài khoản để truy cập vào ứng dụng
- Đăng kí: Cho phép người dùng tạo tài khoản mới với các thông tin cần thiết.
- Quên mật khẩu: Hỗ trợ người dùng khôi phục mật khẩu trong trường hợp bị quên.
- Quản lí thông tin tài khoản và bảo mật: Cho phép người dùng cập nhật và chỉnh sửa thông tin cá nhân.
- Tìm kiếm truyện: Cung cấp tính năng tìm kiếm để người dùng có thể dễ dàng tìm truyện theo tên hoặc từ khóa.
- Hiển thị thông tin truyện: Cho phép hiển thị thông tin của các truyện mà người dùng chọn để đọc.
- Đọc truyện: Cho phép người dùng xem và đọc nội dung của truyện ngay trong ứng dụng với giao diện hiển thị rõ ràng, dễ theo dõi.
- Bình luận chương truyện: Cung cấp tính năng cho người dùng để lại nhận xét, ý kiến và tương tác với các độc giả khác thông qua phần bình luận của mỗi chương truyện.
- Thích truyện: Cho phép người dùng bày tỏ sự yêu thích của mình đối với các truyện bằng cách nhấn biểu tượng trái tim. Truyện được đánh dấu yêu thích có thể được lưu trữ trong danh sách riêng của người dùng.
- Hiển thị danh sách truyện yêu thích của người dùng
- Hiển thị danh sách truyện được yêu thích nhất: Chức năng này cho phép người dùng xem một danh sách các truyện được yêu thích nhất trong ứng dụng, dựa trên số lượng người dùng bày tỏ sự yêu thích (nhấn biểu tượng trái tim)

### **1.3. Yêu cầu phi chức năng**

Hiệu suất:

- Ứng dụng phải tải truyện và hình ảnh nhanh chóng (thời gian tải không quá 2-3 giây).
- Ứng dụng phải hoạt động mượt mà khi người dùng chuyển trang, lướt danh sách truyện.

## CHƯƠNG 2. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG

### 2.1. Phân tích yêu cầu

Xác định người dùng:

Bạn đọc:

- Đặc điểm: Người dùng cuối là những độc giả muốn truy cập vào kho truyện của ứng dụng. Người dùng có thể có các mức độ am hiểu về máy tính khác nhau, do đó ứng dụng cần dễ sử dụng và thân thiện với người dùng.
- Nhu cầu:
  - Giao diện dễ sử dụng.
  - Dễ dàng tìm kiếm và chọn truyện theo thể loại, tác giả, hoặc đánh giá.
  - Khả năng đánh dấu trang, tạo danh sách truyện yêu thích, hoặc lưu lại vị trí đã đọc.
  - Tùy chỉnh giao diện đọc (chế độ tối, thay đổi cỡ chữ, phong chữ, màu nền).

Thu thập yêu cầu:

- Mô tả theo yêu cầu dưới dạng User story:
  - Là một bạn đọc đã có tài khoản, tôi muốn đăng nhập bằng email và mật khẩu để truy cập vào tài khoản của mình và sử dụng các tính năng của ứng dụng.
  - Là một bạn đọc mới, tôi muốn tạo tài khoản bằng email và mật khẩu để có thể sử dụng đầy đủ tính năng của ứng dụng.
  - Là một bạn đọc, tôi muốn đặt lại mật khẩu nếu tôi quên mật khẩu hiện tại để có thể truy cập lại tài khoản của mình.
  - Là một bạn đọc, tôi muốn để lại nhận xét, ý kiến và tương tác với các độc giả khác thông qua phân bình luận của mỗi chương truyện.

Phân tích yêu cầu:

#### 1. Chức năng đăng nhập

##### 1.1. Nhập thông tin đăng nhập

- Mô tả:
- Yêu cầu:
  - Email phải đúng định dạng.
  - Mật khẩu phải được ẩn bằng ký tự ● để bảo vệ quyền riêng tư.

- Kiểm tra bạn đọc đã nhập đầy đủ thông tin hay chưa.

## 1.2. Xác thực thông tin đăng nhập

- Mô tả: Hệ thống cần kiểm tra tính hợp lệ của thông tin đăng nhập.
- Yêu cầu:
  - Xác thực email và mật khẩu với dữ liệu trên Firebase Authentication.
  - Nếu thông tin đăng nhập không đúng, hệ thống sẽ hiển thị thông báo lỗi.
  - Nếu thông tin đúng, người dùng sẽ được chuyển đến trang chủ của ứng dụng.

## 1.3. Xử lý lỗi khi đăng nhập thất bại

- Mô tả: Hệ thống sẽ thông báo đến bạn đọc khi đăng nhập thất bại.
- Yêu cầu:
  - Hiển thị thông báo “Đăng nhập thất bại” nếu thông tin tài khoản không đúng.

## 1.4. Ẩn / hiện mật khẩu

- Mô tả: Cho phép bạn đọc ẩn/hiện mật khẩu.

# 2. Chức năng đăng kí

## 2.1. Nhập thông tin đăng kí

- Mô tả: Bạn đọc cần nhập các thông tin tên tài khoản, ngày sinh, email, mật khẩu, xác nhận mật khẩu để tạo tài khoản.
- Yêu cầu:
  - Email phải đúng định dạng.
  - Mật khẩu phải được ẩn bằng kí tự ● để bảo vệ quyền riêng tư.
  - Kiểm tra bạn đọc đã nhập đầy đủ thông tin hay chưa.

## 2.2. Tạo tài khoản bạn đọc

- Mô tả: Tạo tài khoản khi bạn đọc nhấn nút “Đăng kí”.
- Yêu cầu:
  - Tạo tài khoản bạn đọc trên Firebase Authentication.
  - Nếu tạo tài khoản thành công thì chuyển bạn đọc đến màn đăng nhập.
  - Nếu tạo tài khoản thất bại thì thông báo đến bạn đọc.

## 2.3. Xử lý lỗi khi tạo tài khoản thất bại

- Mô tả: Ứng dụng phải thông báo đến bạn đọc nếu có lỗi xảy ra trong quá trình tạo tài khoản
- Yêu cầu:
  - Nếu không có kết nối Internet thì thông báo “Mất mạng. Không tạo được tài khoản!”.

### 3. Chức năng quên mật

#### 3.1. Nhập địa chỉ email

- Mô tả: Bạn đọc nhập địa chỉ email để ứng dụng gửi mail hướng dẫn đổi mật khẩu vào địa chỉ này.
- Yêu cầu:
  - Địa chỉ email bạn đọc nhập phải là email thực và là email mà bạn đọc đã dùng để tạo tài khoản.
  - Kiểm tra xem bạn đọc đã nhập địa chỉ email hay chưa.

#### 3.2. Gửi mail hướng dẫn đổi mật khẩu mới cho bạn đọc

- Mô tả: Ứng dụng gửi mail hướng dẫn đổi mật khẩu mới đến địa chỉ email bạn đọc đã nhập
- Yêu cầu:
  - Nếu gửi thành công thì thông báo đến bạn đọc và chuyển đến màn đăng nhập.
  - Nếu thất bại thì thông báo đến bạn đọc.

### 4. Chức năng hiển thị trang chủ

- Mô tả: Trang chủ là nơi người dùng truy cập đầu tiên khi vào hệ thống. Trang chủ hiển thị thông tin tổng quan, bao gồm các dịch vụ nổi bật, cập nhật gần đây,....
- Yêu cầu:
  - Các thông tin hiển thị phải được lấy từ cơ sở dữ liệu.
  - Giao diện đơn giản, dễ sử dụng với các trường dữ liệu được sắp xếp rõ ràng.
  - Có thể hiển thị ảnh, cho phép cuộn qua nhiều ảnh, hiển thị tối đa 5 bản tin mới nhất, có tiêu đề và mô tả ngắn.

### 5. Chức năng quản lý thông tin tài khoản và bảo mật: Người dùng có thể xem và cập nhật thông tin tài khoản của mình, thay đổi mật khẩu hoặc xóa tài khoản khi cần thiết

### 5.1. Xem thông tin tài khoản

- Mô tả: Hiện thị các thông tin cá nhân của người dùng bao gồm: ảnh đại diện, tên tài khoản, email..
- Yêu cầu:
  - Các thông tin hiển thị phải được lấy từ cơ sở dữ liệu và đúng với người dùng hiện tại.
  - Giao diện đơn giản, dễ sử dụng với các trường dữ liệu được sắp xếp rõ ràng.

### 5.2. Cập nhật thông tin tài khoản

- Mô tả: Cho phép người dùng chỉnh sửa và cập nhật thông tin cá nhân bao gồm: ảnh đại diện, tên tài khoản hoặc email..
- Yêu cầu:
  - Người dùng có thể chỉnh sửa các trường thông tin cá nhân.
  - Dữ liệu cập nhật phải được kiểm tra tính hợp lệ trước khi lưu, ví dụ như định dạng email.
  - Sau khi thay đổi thông tin, dữ liệu mới phải được lưu vào cơ sở dữ liệu.
  - Hiện thị thông báo thành công khi thông tin tài khoản được cập nhật: "Thông tin tài khoản của bạn đã được cập nhật thành công."
  - Nếu xảy ra lỗi trong quá trình cập nhật, hiện thị thông báo lỗi: "Cập nhật thất bại, vui lòng thử lại sau."

### 5.3. Thay đổi mật khẩu

- Mô tả: Cho phép người dùng thay đổi mật khẩu hiện tại của tài khoản
- Yêu cầu:
  - Người dùng nhập mật khẩu hiện tại.
  - Người dùng nhập mật khẩu mới, xác nhận lại mật khẩu để hoàn tất quá trình.
  - Mật khẩu phải được ẩn bằng ký tự ● để bảo vệ quyền riêng tư.
  - Mật khẩu hiện tại và mật khẩu mới không được trùng nhau, hiện thị thông báo lỗi "Mật khẩu mới và mật khẩu cũ không được trùng khớp".
  - Nếu mật khẩu mới và xác nhận mật khẩu không khớp, hiện thị thông báo lỗi "Mật khẩu mới và mật khẩu xác nhận không trùng khớp."

- Thông báo cho người dùng khi thay đổi mật khẩu thành công "Đổi mật khẩu thành công".

#### 5.4. Xóa tài khoản

- Mô tả: Người dùng có thể xóa tài khoản vĩnh viễn khỏi hệ thống
- Yêu cầu:
  - Hiện thị cảnh báo rõ ràng về việc xóa tài khoản không thể khôi phục "Bạn có chắc chắn muốn xóa tài khoản? Tất cả thông tin sẽ bị xóa vĩnh viễn và không thể khôi phục."
  - Người dùng phải xác nhận lại trước khi tài khoản bị xóa.
  - Nếu người dùng đồng ý xóa tài khoản, toàn bộ thông tin liên quan đến tài khoản sẽ bị xóa vĩnh viễn khỏi cơ sở dữ liệu.
  - Thông báo kết quả xóa tài khoản đến người dùng "Tài khoản của bạn đã được xóa thành công."

#### 6. Chức năng tìm kiếm truyện

- Mô tả: Người dùng có thể sử dụng thanh tìm kiếm để tìm kiếm theo tên truyện, tên tác giả hoặc thể loại, và kết quả tìm kiếm phải được trả về nhanh chóng, chính xác.
- Yêu cầu:
  - Khi người dùng nhập vào tên truyện, tên tác giả hoặc thể loại hệ thống phải tìm kiếm trong cơ sở dữ liệu các truyện có tên, tác giả, thể loại tương ứng với từ khóa tìm kiếm
  - Tìm kiếm phải thực hiện nhanh chóng, không gây gián đoạn trải nghiệm người dùng.
  - Danh sách kết quả phải hiển thị tên truyện, hình ảnh đại diện, tác giả và thể loại của các truyện liên quan đến từ khóa được tìm kiếm.
  - . Kết quả tìm kiếm phải hiển thị dưới dạng danh sách, có phân trang (nếu số lượng kết quả lớn)
  - Người dùng có thể nhấn vào từng truyện trong danh sách để xem thông tin chi tiết về truyện đó.
  - Hiện thị thông báo "Không tìm thấy kết quả nào" nếu không có từ khóa nào khớp với từ khóa tìm kiếm.

#### 7. Chức năng hiển thị thông tin truyện.

- Mô tả: Chức năng này cho phép hiển thị chi tiết thông tin của một truyện bao gồm: tên truyện, hình ảnh, tác giả, số chương, thể loại, tóm tắt nội dung, lượt thích, và các bình luận.
- Yêu cầu:
  - Tên truyện, tác giả, và hình ảnh: Hiển thị rõ ràng tên truyện, tác giả và hình ảnh đại diện từ cơ sở dữ liệu, đảm bảo chất lượng hình ảnh và thông tin chính xác.
  - Số chương và thể loại: Hiển thị tổng số chương và thể loại, cập nhật tự động từ cơ sở dữ liệu.
  - Tóm tắt nội dung: Hiển thị phần tóm tắt ngắn gọn, giúp người dùng nhanh chóng nắm bắt nội dung chính.
  - Lượt thích và tương tác: Hiển thị số lượt thích và cho phép người dùng bấm thích, số lượt thích được cập nhật từ cơ sở dữ liệu.
  - Danh sách chương: Hiển thị danh sách chương truyện với hình ảnh và số chương, sắp xếp theo thứ tự mới nhất hoặc cũ nhất.
  - Bình luận: Hiển thị danh sách bình luận theo thứ tự thời gian, cho phép người dùng thêm bình luận mới để tương tác, chia sẻ suy nghĩ của mình về nội dung truyện.

8. Chức năng đọc truyện: Cho phép người dùng đọc nội dung các chương truyện một cách liên tục và dễ dàng, hỗ trợ các tính năng như lật trang, chuyển sang đọc chương trước/sau, chọn chương trong danh sách chương của truyện để đọc.và lưu lại trang đọc của người dùng

#### 8.1. Hiển thị nội dung chương truyện

- Mô tả: Hiển thị đầy đủ nội dung văn bản hoặc hình ảnh của chương truyện để người dùng có thể đọc.
- Yêu cầu:
  - Nội dung truyện phải được tải từ cơ sở dữ liệu và hiển thị rõ ràng, không thiếu sót.
  - Phải hiển thị được cả văn bản và hình ảnh
  - Giao diện hiển thị phải hỗ trợ lật trang hoặc cuộn liên tục để đọc.

#### 8.2. Ghi nhớ vị trí đọc



- Mô tả: Hệ thống phải ghi nhớ vị trí đọc cuối cùng của người dùng để khi quay lại, người dùng có thể tiếp tục đọc từ vị trí đã dừng.
- Yêu cầu: Vị trí đọc phải được tự động lưu sau mỗi lần thoát khỏi chương truyện và khi người dùng quay lại, nội dung sẽ được hiển thị từ đúng vị trí đã lưu.

### 8.3. Chuyển chương

- Mô tả: Người dùng có thể dễ dàng chuyển giữa các chương truyện bằng cách lướt sang chương tiếp theo hoặc quay lại chương trước.
- Yêu cầu:
  - Cung cấp nút để lật trang, chuyển chương một cách dễ dàng.
  - Đảm bảo nội dung chương tiếp theo/ trước được tải nhanh chóng và mượt mà.

## 9. Chức năng bình luận chương truyện: Người dùng có thể để lại bình luận và xem các bình luận khác về một chương truyện cụ thể trong ứng dụng.

### 9.1. Xem bình luận

- Mô tả: Hiển thị danh sách bình luận của các người dùng về chương truyện.
- Yêu cầu:
  - Mỗi bình luận hiển thị avatar và tên người dùng, nội dung bình luận, số lượt thích.
  - Giao diện hiển thị bình luận phải dễ dàng theo dõi, trực quan và hỗ trợ cuộn dài để xem nhiều bình luận.

### 9.2. Thêm mới bình luận

- Mô tả: Người dùng có thể nhập bình luận mới về chương truyện và gửi lên hệ thống, cho phép người đọc tương tác và chia sẻ suy nghĩ của mình về nội dung chương truyện.
- Yêu cầu:
  - Khi người dùng gửi bình luận, nó phải được lưu trữ trong cơ sở dữ liệu và hiển thị ngay trên giao diện.
  - Thông báo lỗi nếu bình luận không gửi được do lỗi kết nối hoặc máy chủ "Không thể đăng bình luận. Vui lòng kiểm tra kết nối mạng và thử lại."

### 9.3. Thích bình luận

- Mô tả: Người dùng có thể nhấn thích trên bình luận của mình hoặc của người khác..
- Yêu cầu:

10. Chức năng hiển thị danh sách truyện yêu thích: Cho phép người dùng xem danh sách các truyện mà họ đã đánh dấu yêu thích. Từ đó, người dùng có thể nhanh chóng truy cập các truyện này để tiếp tục đọc hoặc quản lý danh sách (xóa truyện, sắp xếp lại truyện yêu thích)..

#### 10.1. Hiển thị danh sách truyện yêu thích của từng người dùng

- Mô tả: Hiển thị danh sách các truyện mà người dùng đã đánh dấu yêu thích, bao gồm thông tin về tên truyện, hình ảnh đại diện, số chương, tác giả, và thể loại
- Yêu cầu:
  - Hiển thị dưới dạng danh sách, có phân trang (nếu số lượng kết quả lớn)
  - Thông tin hiển thị bao gồm: tên truyện, ảnh bìa, số chương, tác giả và thể loại.
  - Nếu người dùng chưa thêm truyện nào vào danh sách yêu thích, hiển thị thông báo: "Bạn chưa có truyện yêu thích nào."

#### 10.2. Xóa truyện khỏi danh sách yêu thích

- Mô tả: Người dùng có thể xóa truyện ra khỏi danh sách yêu thích nếu không muốn theo dõi nữa.
- Yêu cầu:
  - Người dùng ấn đúp vào biểu tượng icon trái tim ở mỗi truyện trong danh sách (khi đó icon từ màu đỏ sẽ chuyển sang trắng) để người dùng có thể dễ dàng loại bỏ truyện khỏi danh sách yêu thích.
  - Hiển thị thông báo xác nhận trước khi xóa: "Bạn có chắc chắn muốn xóa truyện này khỏi danh sách yêu thích không?"
  - Sau khi xóa, danh sách phải được cập nhật ngay lập tức mà không cần tải lại trang.

11. Chức năng hiển thị danh sách truyện được yêu thích nhất trên ứng dụng:

- Mô tả: Hiển thị danh sách các truyện có lượt yêu thích cao nhất từ người dùng trên toàn ứng dụng, thông tin truyện bao gồm: tên truyện, ảnh bìa, và số lượt yêu thích.

- Yêu cầu:
  - Dữ liệu truyện được yêu thích phải lấy từ cơ sở dữ liệu, dựa trên số lượt yêu thích của người dùng.
  - Sắp xếp danh sách từ truyện có lượt yêu thích cao nhất đến thấp nhất.
  - Người dùng có thể nhấp vào từng truyện để xem chi tiết hoặc đọc truyện.
  - Có giới hạn hiển thị danh sách truyện được yêu thích nhiều nhất: tối đa 50 truyện.

## 2.2. Thiết kế hệ thống

### 2.2.1 Thiết kế kiến trúc(MVC)

- Model
  - UserProfile: Class ứng với collection users trong Firebase firestore
  - Comment: Class ứng với cấu trúc json comments trong Firebase realtime database
- View
  - ActivitySignInBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file activity\_sign\_in.xml.
  - ActivitySignUpBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file activity\_sign\_up.xml.
  - ActivityForgotPasswordBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file activity\_forget\_password.xml.
  - ActivitySplashBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file activity\_splash.xml.
  - ActivityCommentBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file activity\_comment.xml.
  - CommentItemBinding: Bao gồm các thuộc tính ánh xạ đến các thành phần UI trong file comment\_item.xml
- Controller
  - SplashActivity

- SignInActivity
- SignUpActivity
- ForgetPasswordActivity
- CommentActivity

**Dựa trên kiến trúc MVC đã học, ta có thể xác định các lớp sau**

- Lớp SplashActivity:
  - Thuộc tính:
    - FirebaseAuth auth: Cung cấp các phương thức và API để xác thực người dùng.
  - Phương thức:
    - onCreate(): Phương thức được gọi khi Activity được tạo ra. Dùng để gắn view cho Activity.
    - onStart(): Phương thức được gọi ngay sau khi phương thức onCreate() chạy xong. Gọi đến phương thức checkUser().
    - checkUser(): Kiểm tra bạn đọc đã đăng nhập vào ứng dụng hay chưa.
- Lớp SignInActivity:
  - Thuộc tính:
    - boolean passwordVissible: Trạng thái ẩn/hiển của mật khẩu.
    - FirebaseAuth auth: Cung cấp các phương thức và API để xác thực người dùng.
    - ActivitySignInBinding binding: Thuộc tính có kiểu của lớp được tự động tạo ra khi sử dụng View Binding. Cung cấp một cách an toàn để truy cập các View mà không cần sử dụng findViewById()
  - Phương thức:
    - onCreate(): Phương thức để gắn view cho activity và đăng kí call back cho các button.

- `boolean isValidInput(String email, String password)`: Phương thức kiểm tra dữ liệu bạn đọc nhập đã hợp lệ hay chưa.
- **Lớp SignUpActivity:**
  - Thuộc tính:
    - `ActivitySignUpBinding binding`: Thuộc tính có kiểu của lớp được tự động tạo ra khi sử dụng View Binding. Cung cấp một cách an toàn để truy cập các View mà không cần sử dụng `findViewById()`
    - `FirebaseAuth auth`: Cung cấp các phương thức và API để xác thực người dùng.
  - Phương thức:
    - `void onCreate()`: Gắn view cho activity. Đăng kí call back cho các button.
    - `boolean isValidInput(String tenDangNhap, String email, String matKhau, String ngaySinh, String xacNhanMatKhau)`: Kiểm tra dữ liệu bạn đọc nhập đã hợp lệ hay chưa.
    - `void onDateSet(DatePicker view, int year, int month, int dayOfMonth)`: Phương thức cập nhật dữ liệu ngày tháng cho trường ngày sinh khi người dùng chọn trên hộp thoại.
    - `void showDatePicker()`: Phương thức hiển thị hộp thoại chọn ngày tháng lên.
- **Lớp ForgetPasswordActivity:**
  - Thuộc tính:
    - `ActivityForgetPasswordBinding binding`: Thuộc tính có kiểu của lớp được tự động tạo ra khi sử dụng View Binding. Cung cấp một cách an toàn để truy cập các View mà không cần sử dụng `findViewById()`
    - `FirebaseAuth auth`: Cung cấp các phương thức và API để xác thực người dùng.
  - Phương thức:

- void onCreate(): Gắn view cho Activity và đăng kí call back cho các button.
- Lớp CommentActivity:
  - Thuộc tính:
    - ActivityCommentBinding binding:
    - ArrayList<Comment> commentList: Danh sách bình luận của chương truyện
    - String storyID: id của truyện
    - String chapID: id của chương truyện
    - CommentAdapter adapter: Là cầu nối liên kết dữ liệu với recycler view và quản lí view holder
    - DatabaseReference commentsRef: Tham chiếu đến Firebase realtime database
  - Phương thức:
    - void onCreate(): gắn view cho activity, set adapter cho recycler view, gọi hàm getComments() và đăng kí call back cho button.
    - void getComments(): Lấy danh sách bình luận của chương truyện từ Firebase realtime database, cập nhật lại danh sách nếu có sự thay đổi trên database.
    - void getID(): Lấy id chương truyện và id truyện từ intent.
- Lớp CommentAdapter:
  - Thuộc tính:
    - List<Comment> comments: Danh sách bình luận của chương truyện
  - Phương thức:
    - CommentViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType): Tạo view holder cho recycler view.

- void onBindViewHolder(@NonNull CommentViewHolder holder, int position): Đổ dữ liệu vào view holder.
- int getItemCount(): Trả về số lượng phần tử trong danh sách bình luận.
- void updateData(ArrayList<Comment> commentList): cập nhật lại danh sách bình luận nếu có sự thay đổi từ database
- Lớp CommentViewHolder:
  - Thuộc tính: CommentItemBinding binding
  - Phương thức:
    - void bind(Comment comment): Lấy các thông tin của bình luận để đổ vào các thành phần UI.

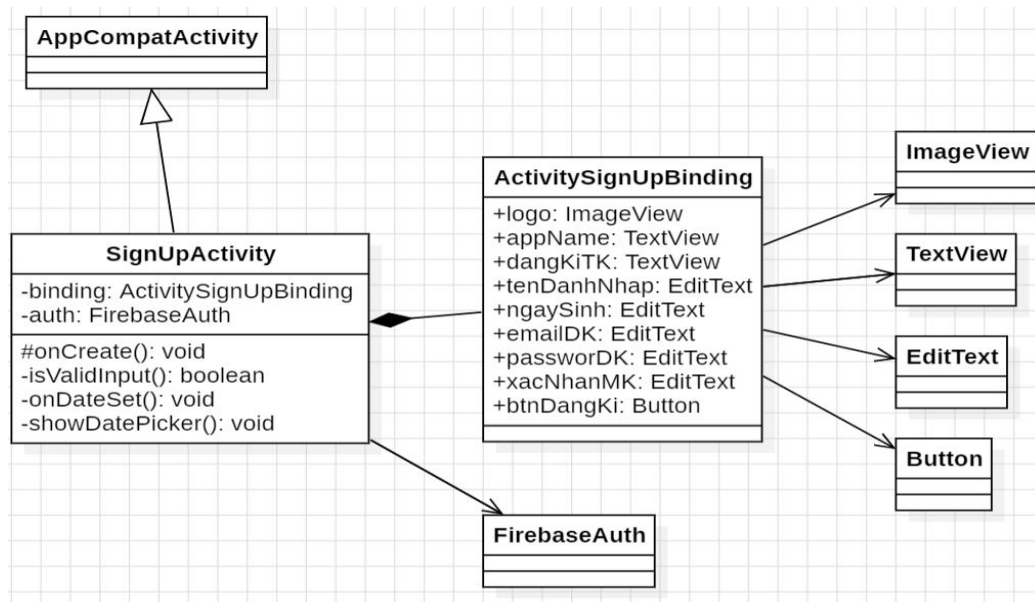
### ***Mối quan hệ giữa các lớp:***

- SignInActivity truy cập vào các thành phần UI để lấy dữ liệu và điều khiển chúng thông qua ActivitySignInBinding.
- SignUpActivity truy cập vào các thành phần UI để lấy dữ liệu và điều khiển chúng thông qua ActivitySignUpBinding.
- ForgetPasswordActivity truy cập vào các thành phần UI để lấy dữ liệu và điều khiển chúng thông qua ActivityForgetPasswordBinding.
- SplashActivity truy cập vào các thành phần UI để lấy dữ liệu và điều khiển chúng thông qua ActivitySplashBinding.
- SignInActivity, SignUpActivity, ForgetPasswordActivity, SplashActivity gọi các API của FirebaseAuth qua lớp FirebaseAuth
- CommentActivity chịu trách nhiệm lấy và cập nhật lại danh sách bình luận cho CommentAdapter, lấy danh sách bình luận, thêm bình luận trên Firebase realtime qua lớp DatabaseReference

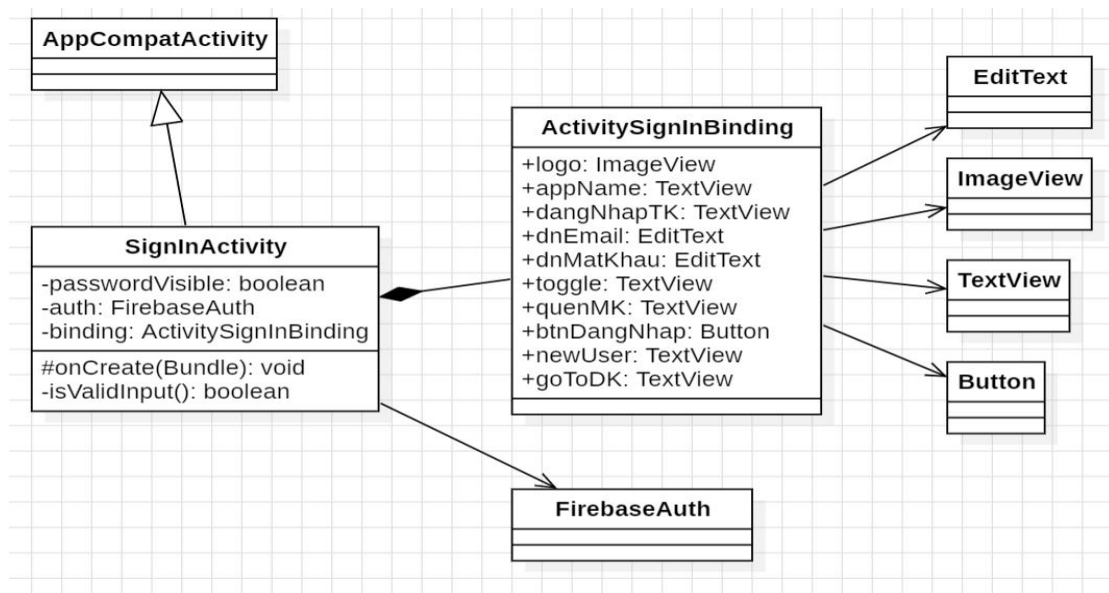
- CommentViewHolder lấy ảnh và tên người dùng của bình luận trên Firestore qua lớp FirebaseFirestore, cập nhật lượt thích của bình luận qua lớp DatabaseReference, lấy id của người dùng hiện tại qua lớp FirebaseAuth

### 2.2.2. Biểu đồ lớp

Chức năng đăng kí

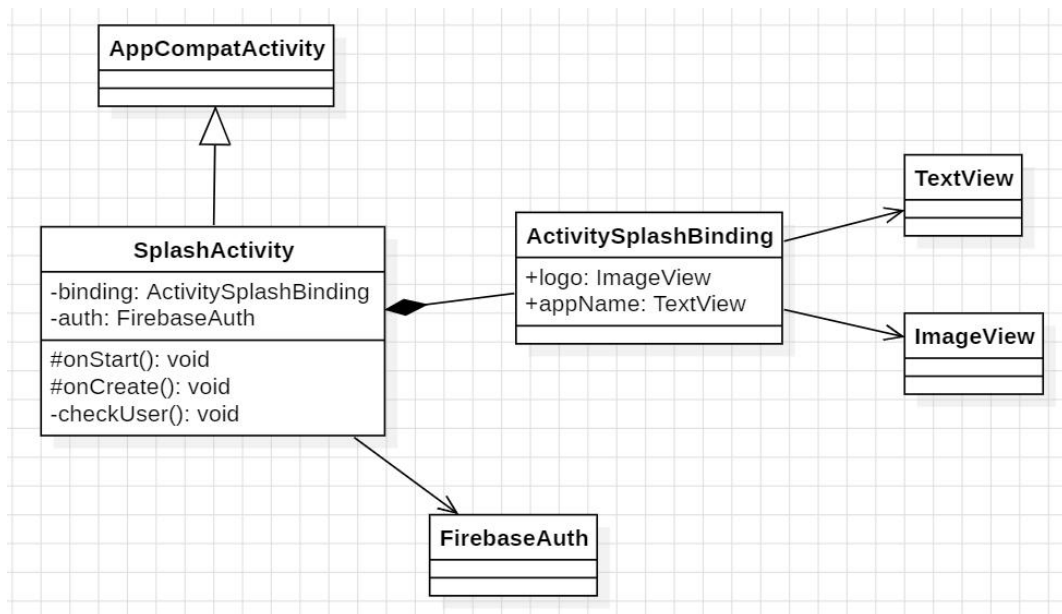


Chức năng đăng nhập

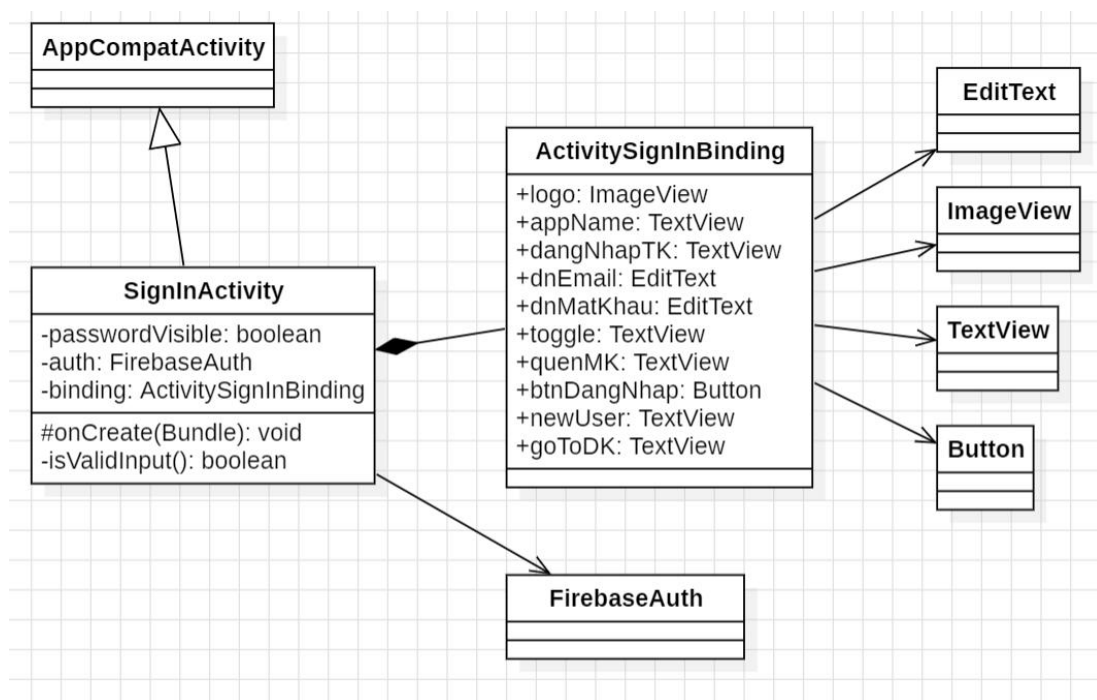


Chức năng hiển thị logo ứng dụng

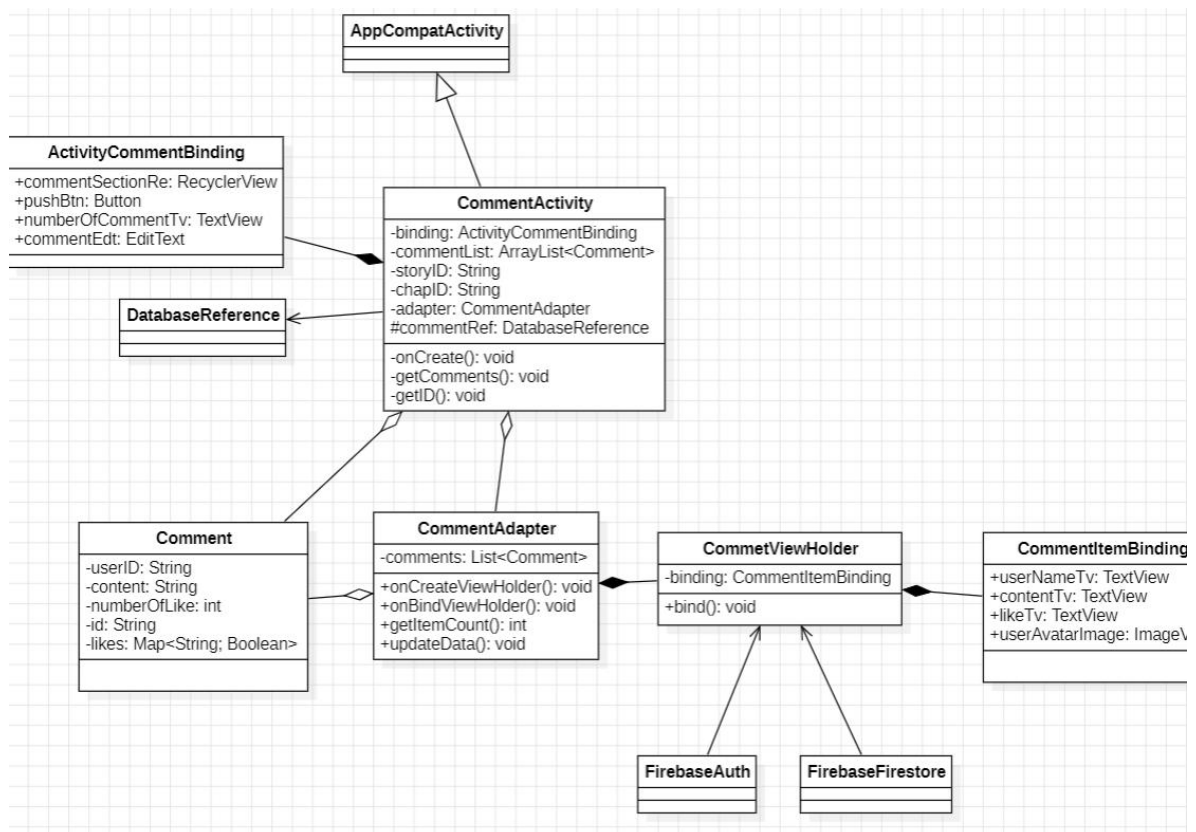




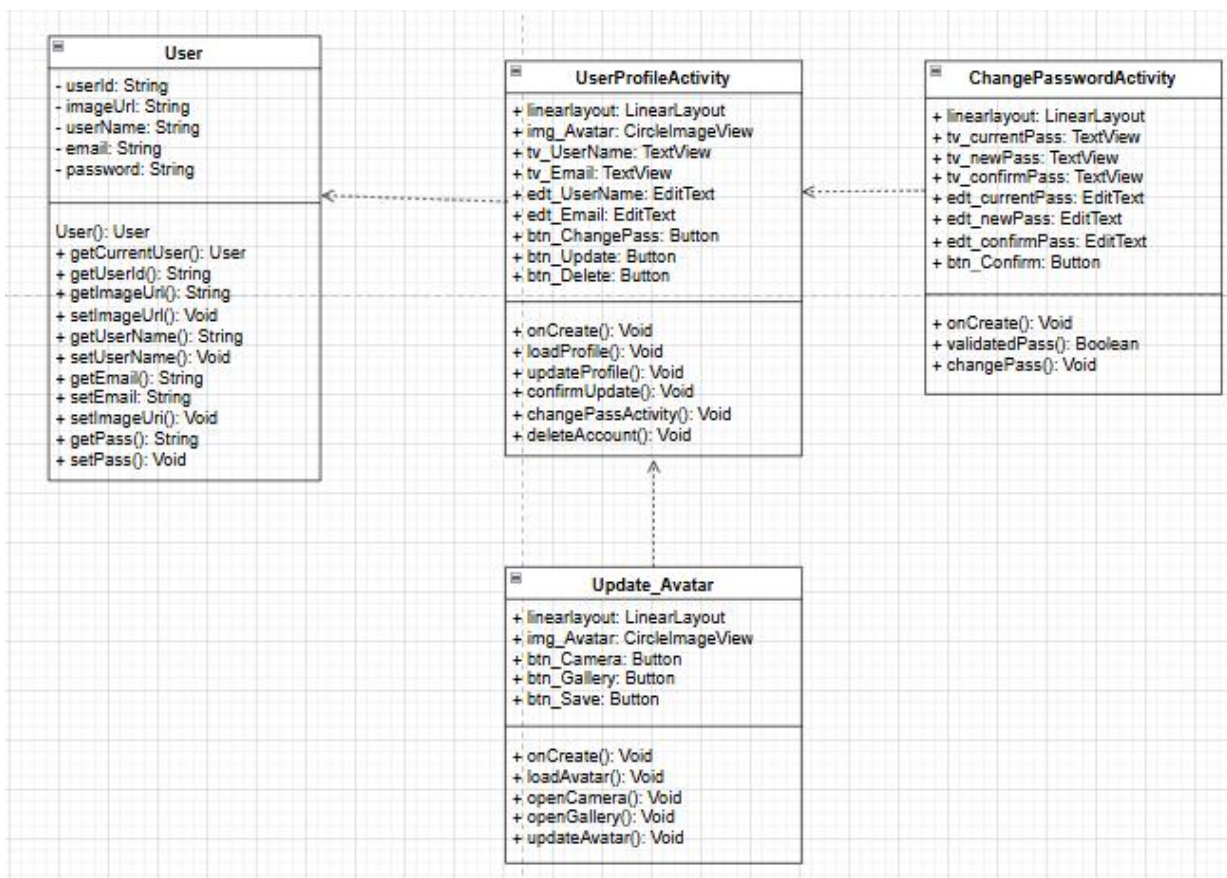
Chức năng quên mật khẩu



Chức năng bình luận chương



## Chức năng quản lý tài khoản người dùng



### 2.2.3. Thiết kế cơ sở dữ liệu:

Bảng 2.1 Bảng “stories”: Chứa danh sách truyện với thông tin chi tiết về từng truyện

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
id	String	Chính	Mã định danh của truyện
author_key	Array[String]		Danh sách các từ khóa liên quan đến tên tác giả
categories	Array[String]		Danh sách các thể loại truyện
content	String		Nội dung hoặc mô tả của truyện
created_at	Timestamp		Thời gian tạo truyện
favorites	integer		Số lượt yêu thích của truyện
imageUrl	String		Đường dẫn URL của ảnh bìa truyện
name	String		Tên truyện
views	integer		Số lượt xem của truyện

Bảng 2.2 Bảng “users”: Chứa danh sách tài khoản người dùng

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
id	String	Chính	Mã định danh của người dùng (được tạo tự động)
name	String		Tên của người dùng
imageUrl	String		Đường dẫn URL tới ảnh đại diện của người dùng

Bảng 2.3 Bảng “comments”: Chứa danh sách các bình luận cho từng chương của truyện

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
id	String	Chính	Mã thể loại
title	String		Tên thể loại
content	String		Nội dung của bình luận
userID	String		Mã định danh của người dùng đã đăng bình luận
numberOfLike	Integer		Số lượt thích của bình luận
likes	Map[String, Boolean]		Bản đồ (map) chứa trạng thái thích của từng người dùng với bình luận

Bảng 2.4 Bảng “history”: Chứa lịch sử đọc của người dùng, lưu lại chương và trang hiện tại cho từng truyện

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
userID	String	Chính	Mã định danh của người dùng
storyID	String		Mã định danh của truyện mà người dùng đang đọc
currentChap	Integer		Chương hiện tại mà người dùng đã đọc
currentPage	Integer		Trang hiện tại trong chương mà người dùng đã đọc

Bảng 2.5 Bảng “story\_chapters”: Chứa danh sách các chương của từng truyện, với đường dẫn URL đến nội dung của từng chương

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
storyID	String	Chính	Mã định danh của truyện (ví dụ: story1)
chap1	String		Đường dẫn URL tới nội dung của chương 1 (ví dụ: URL Firebase Storage của chương 1)

Bảng 2.6 Bảng “user\_favorites”: Chứa danh sách các truyện mà người dùng đã yêu thích

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
userID	String	Chính	Mã định danh của người dùng
storyID	Map[String, Boolean]		Bản đồ chứa mã định danh của truyện (storyID) và trạng thái yêu thích (true)

Tên trường	Kiểu dữ liệu	Khóa	Mô tả
id	String	Chính	Mã chương
comic_id	String	Liên kết với id bảng comics	Mã truyện
title	String		Tên chương
order	integer		Số thứ của chương trong truyện

img_url	String		Đường link ảnh tiêu đề chương
pdf_url	String		Đường link pdf chứa ảnh đọc của chương
created_at	long		Thời gian tạo

## Thiết kế giao diện

- Giao diện đăng kí





**ĐĂNG KÝ TÀI KHOẢN**

Tên tài khoản \*

Email \*

Mật khẩu \*

Xác nhận mật khẩu \*

☐ Tôi đồng ý với [Điều khoản Dịch vụ](#) và [Chính sách Bảo mật](#).

**Đăng kí**

*Hình 2.1 Giao diện đăng kí*

- Giao diện đăng nhập



## ĐĂNG NHẬP TÀI KHOẢN

E-mail

---

Mật khẩu

---

☐

Ghi nhớ đăng nhập

[Quên mật khẩu ?](#)

**Đăng nhập**

Bạn chưa có tài khoản ? [Đăng ký](#)

*Hình 2.2 Giao diện đăng nhập*

- Giao diện quên mật khẩu



## Đặt lại mật khẩu

Nhập địa chỉ E-mail liên kết với tài khoản của bạn để nhận hướng dẫn cách đặt lại mật khẩu.

E-mail

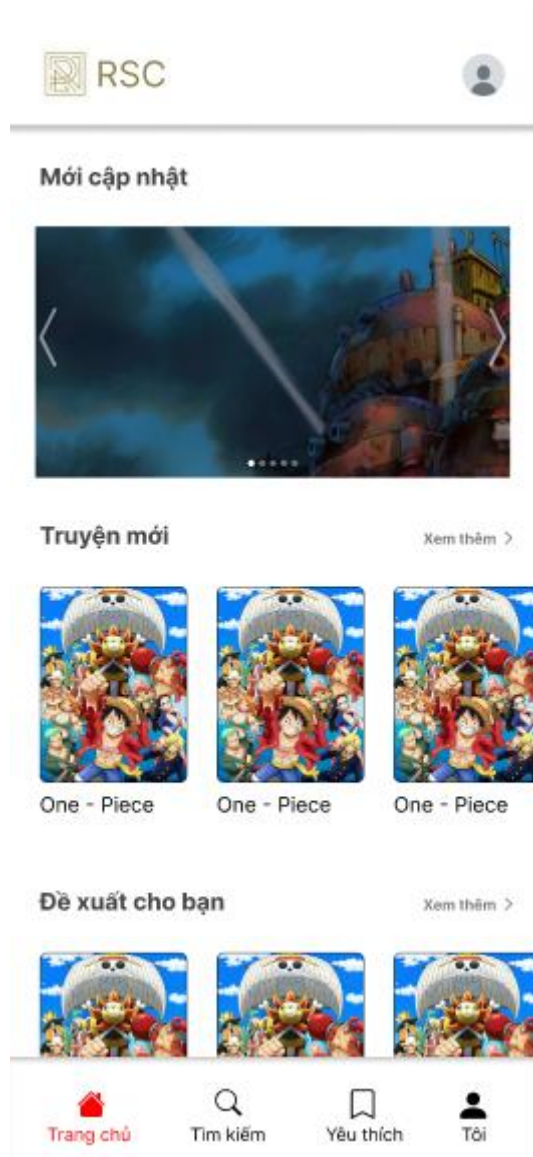
---

Xác nhận

*Hình 2.3 Giao diện quên mật khẩu*

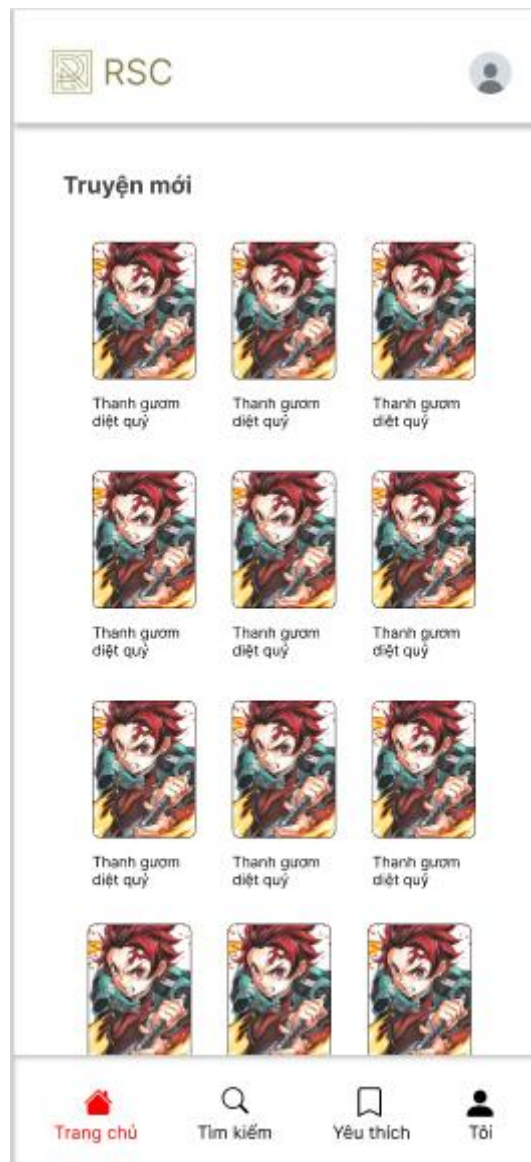
- Giao diện trang chủ





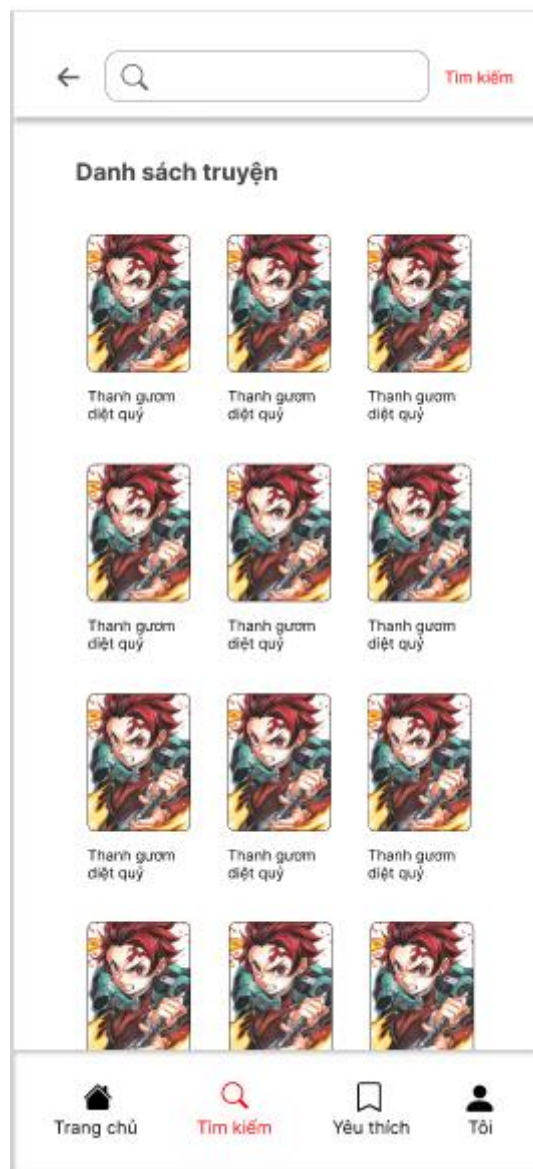
Hình 2.4 Giao diện trang chủ

- Giao diện danh sách truyện mới



Hình 2.5 Giao diện danh sách truyện mới

- Giao diện tìm kiếm



Hình 2.6 Giao diện tìm kiếm

- Giao diện tìm kiếm theo bộ lọc



Hình 2.7 Giao diện tìm kiếm theo bộ lọc

- Giao diện hiển thị nội dung chi tiết truyện



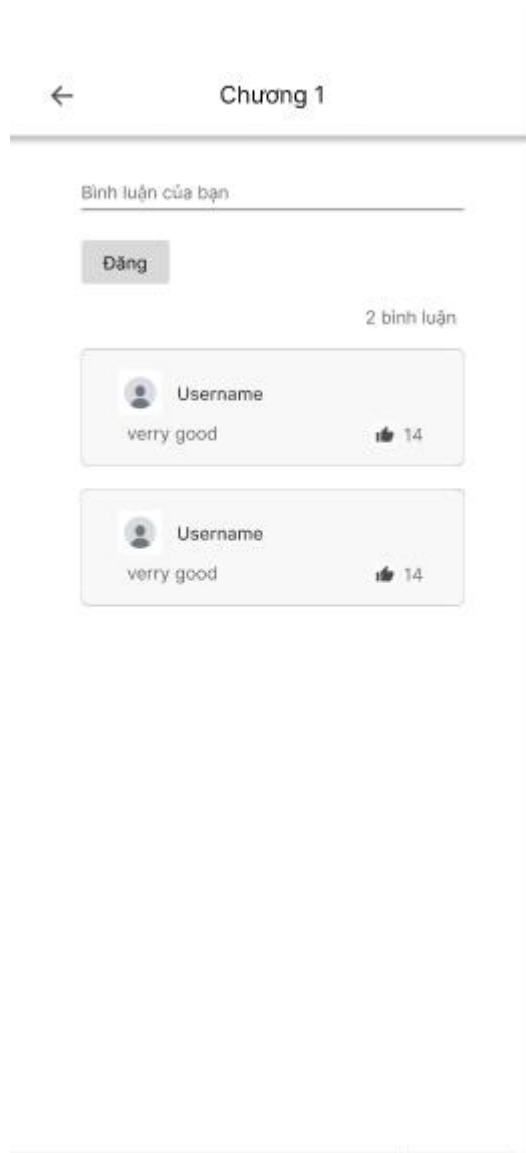
Hình 2.8 Giao diện hiển thị nội dung chi tiết truyện

- Giao diện đọc truyện



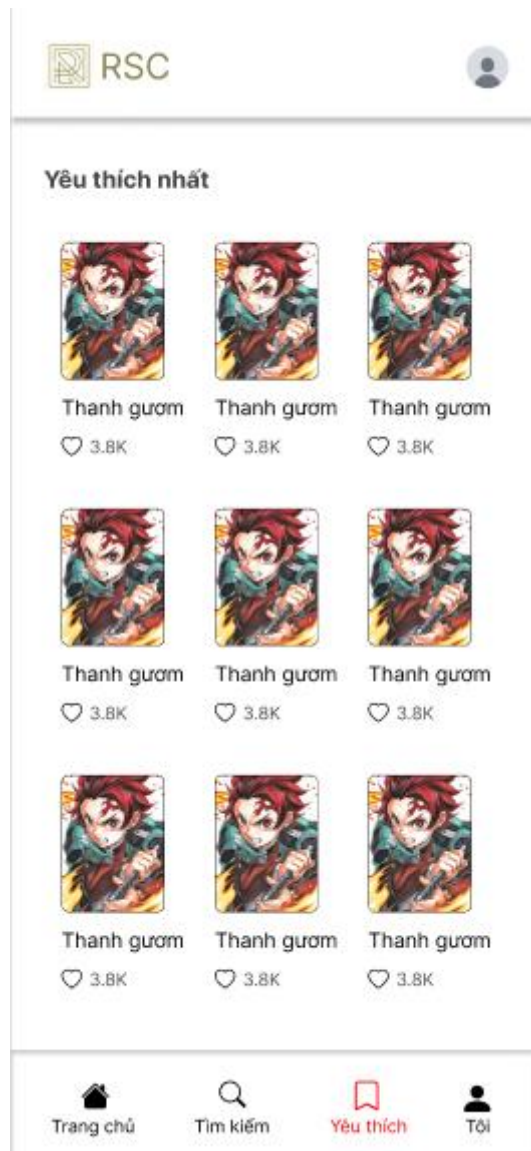
Hình 2.9 Giao diện đọc truyện

- Giao diện bình luận



*Hình 2.10 Giao diện bình luận truyện theo mỗi chương*

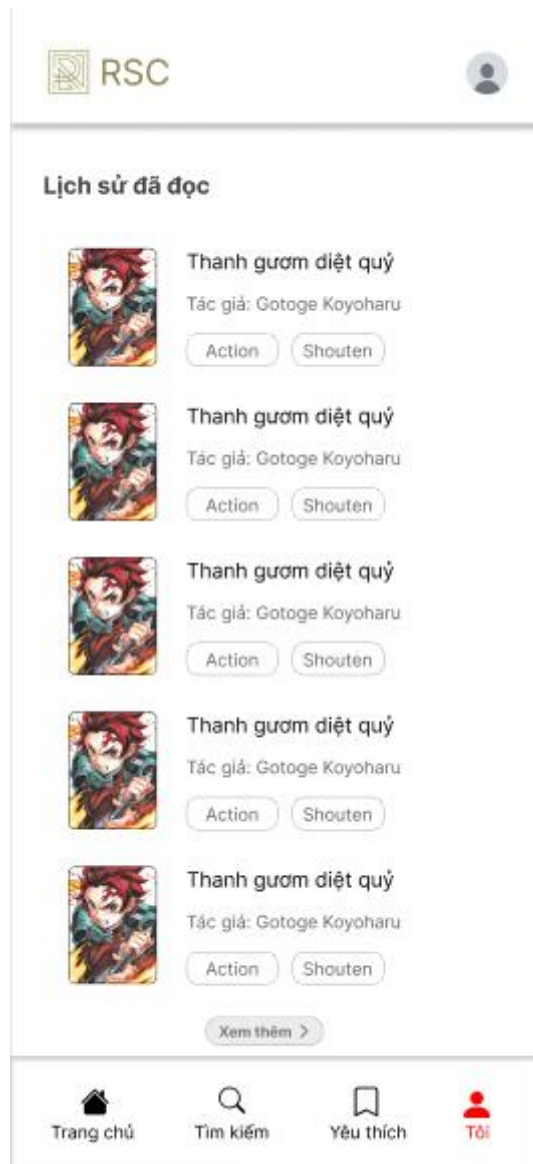
- Giao diện truyện được yêu thích nhất



Hình 2.11 Giao diện danh sách sách truyện được yêu thích nhất

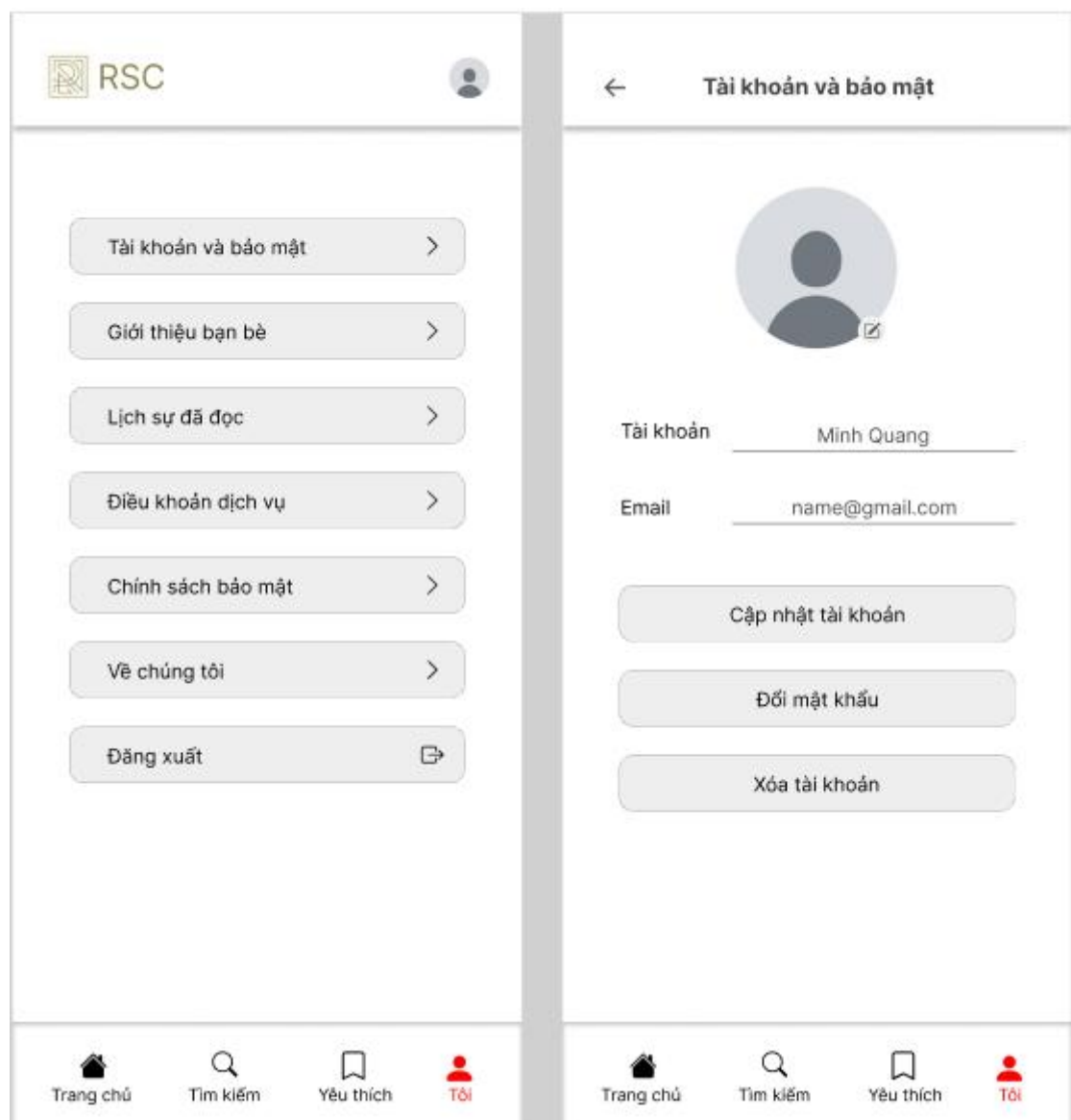
- Giao diện lịch sử đã đọc





Hình 2.12 Giao diện lịch sử truyện đã đọc

- Giao diện tài khoản người dùng



Hình 2.13 Giao diện tài khoản người dùng

- Giao diện đổi mật khẩu

Hình 2.14 Giao diện đổi mật khẩu

### 2.3. Triển khai

- Viết code
- Kiểm thử: chưa thực hiện

### 2.4. Vận hành và bảo trì

- Cài đặt và triển khai:
- Bảo trì: chưa thực hiện

## CHƯƠNG 3. KẾT QUẢ THỰC HIỆN

### 3.1. Công nghệ đã sử dụng

- Ngôn ngữ lập trình: Java
- Công cụ: Android Studio
- Thư viện:
  - `libs.appcompat`, `libs.material`, `libs.activity`, `libs.constraintlayout`:
    - cung cấp các thành phần giao diện và chức năng cơ bản như Activity, Material Design, Constraint Layout
  - `libs.junit`, `libs.ext.junit`, `libs.espresso.core`:
    - Các thư viện hỗ trợ cho việc kiểm thử (testing)
  - `com.github.barteksc:AndroidPdfViewer`:
    - Hiển thị tệp PDF trong ứng dụng Android, cho phép người dùng xem các tài liệu PDF trực tiếp trên ứng dụng
  - `com.google.firebase:firebase-bom`:
    - Firebase BOM (Bill of Materials) giúp quản lý các phiên bản của tất cả các thư viện Firebase.
  - `com.google.firebase:firebase-auth`:
    - Thư viện hỗ trợ chức năng xác thực người dùng của Firebase, cho phép đăng nhập qua email, Google, Facebook, và các nhà cung cấp dịch vụ xác thực khác.
  - `com.google.firebase:firebase-database`:
    - Firebase Realtime Database, cho phép lưu trữ và đồng bộ hóa dữ liệu trong thời gian thực.

### 3.2. Tiến độ thực hiện

Link github tới dự án: [https://github.com/mkq48/CSE441\\_PROJECT](https://github.com/mkq48/CSE441_PROJECT)

## KẾT LUẬN

- Ưu điểm:
  - Ứng dụng cung cấp các chức năng cần thiết cho người dùng như đăng ký, đăng nhập, quên mật khẩu, tìm kiếm, đọc truyện, bình luận, và quản lý tài khoản cá nhân.
  - Tìm kiếm hiệu quả: Chức năng tìm kiếm cho phép tìm theo tên truyện, tác giả, và thể loại giúp người dùng nhanh chóng tìm được nội dung yêu thích mà không mất nhiều thời gian.
  - Quản lý yêu thích và lịch sử đọc: Tính năng này giúp người dùng dễ dàng truy cập lại truyện yêu thích hoặc tiếp tục từ vị trí đã đọc, mang lại trải nghiệm liền mạch và cá nhân hóa.
  - Giao diện thân thiện: Giao diện được thiết kế đơn giản, dễ dùng và thân thiện sẽ giúp người dùng có trải nghiệm tốt, giảm thiểu tỷ lệ bỏ ứng dụng.
- Nhược điểm:
  - Hiệu năng chưa tối ưu: Khi ứng dụng xử lý dữ liệu lớn hoặc hình ảnh độ phân giải cao, có thể gặp tình trạng lag hoặc giảm hiệu suất, khiến người dùng khó chịu.
  - Thiếu tương thích trên nhiều thiết bị Android: Ứng dụng chưa được tối ưu hóa cho các thiết bị Android khác nhau, dẫn đến khả năng gặp lỗi hoặc hiệu suất không đồng đều trên các thiết bị khác nhau.
  - Xử lý dữ liệu Firebase phức tạp: Khi có nhiều người dùng thao tác đồng thời, dễ xảy ra xung đột hoặc giảm hiệu suất. Đặc biệt, với truy vấn phức tạp, Firebase có thể làm ứng dụng bị chậm.
- Hướng phát triển chủ đề:
  - Tối Ưu Hóa Trải Nghiệm Người Dùng (User Experience - UX): Chế độ đọc linh hoạt như cung cấp nhiều chế độ đọc khác nhau: chế độ ban đêm, chế độ đọc offline, và chế độ đọc ngang (cho phép cuộn ngang).
  - Tăng Cường Tương Tác Cộng Đồng: Cho phép người dùng thả biểu tượng cảm xúc (emoji) trên các đoạn văn hoặc chương truyện yêu thích.
  - Tính năng Audio Story: Phát triển tính năng audio cho phép người dùng nghe truyện thay vì đọc, tương tự như các audiobook. Điều này giúp tăng trải nghiệm cho những người thích nghe truyện khi di chuyển hoặc thư giãn.

