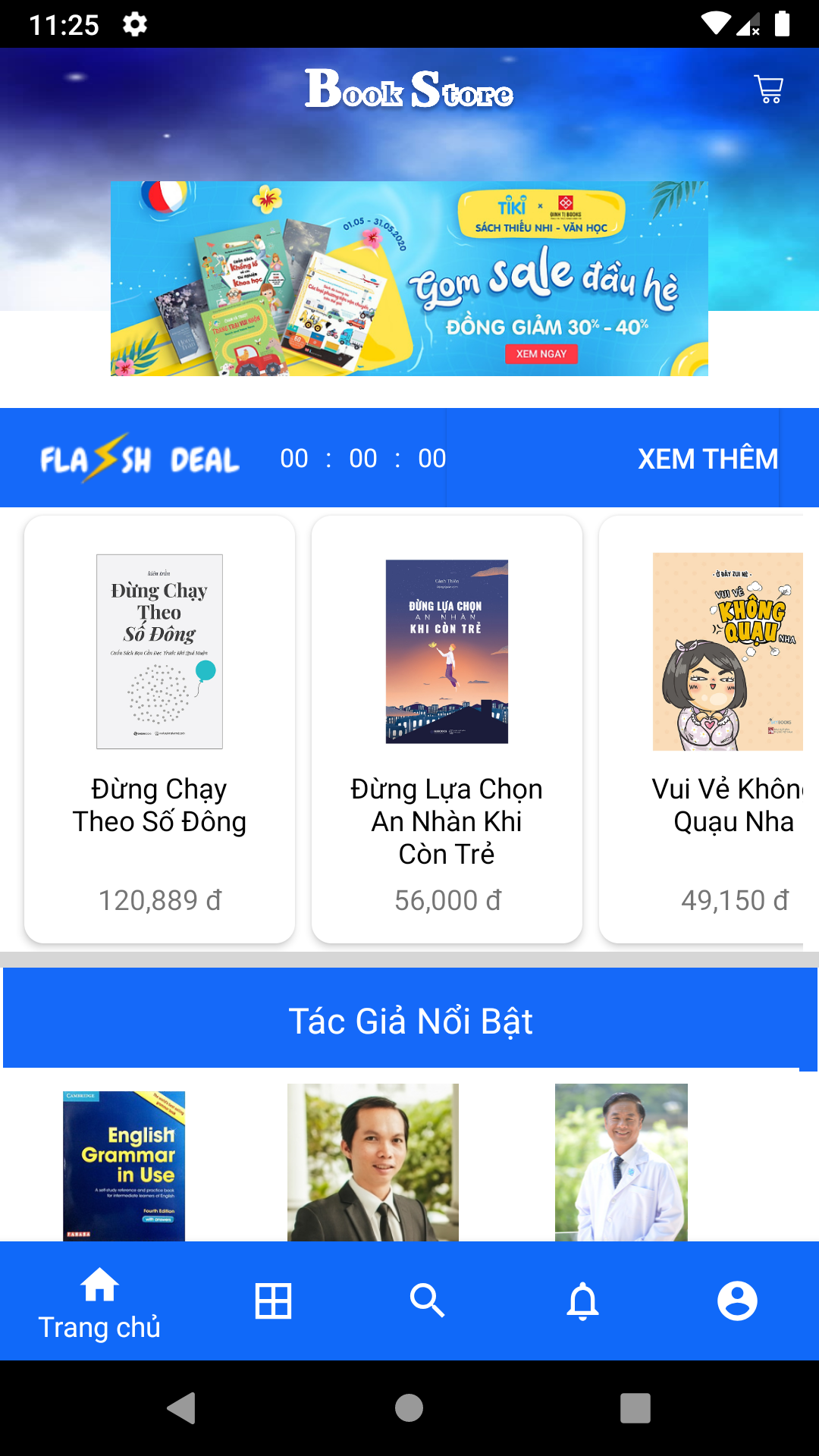
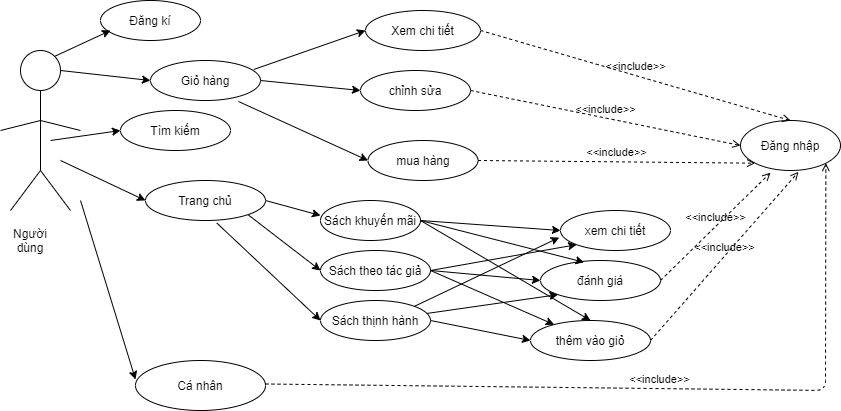
**ĐỀ TÀI :** ỨNG DỤNG GIỚI THIỆU VÀ MUA SÁCH.

**I. MÔ TẢ NGHIỆP VỤ:**

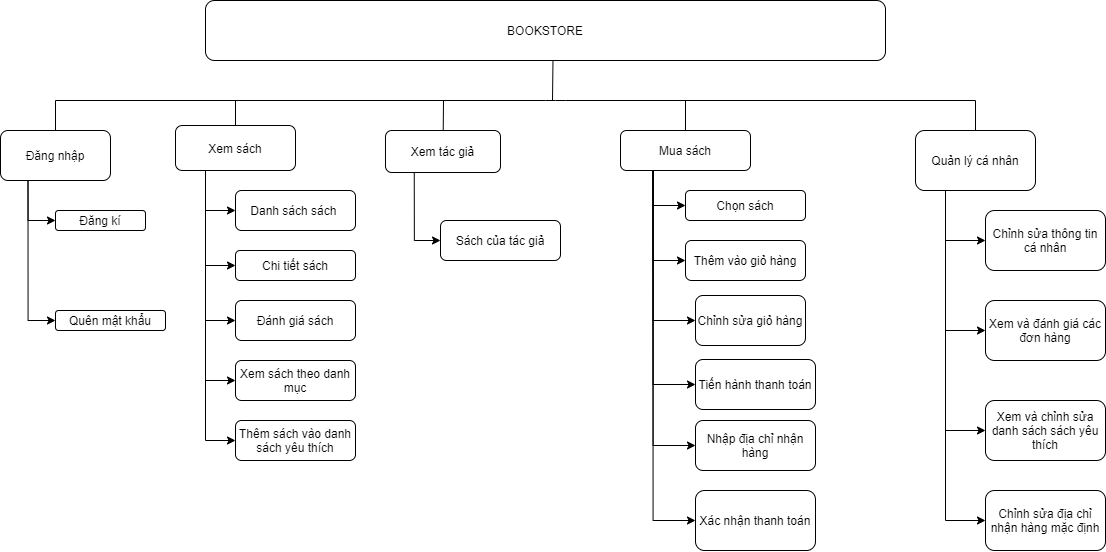
* Ứng dụng hiển thị các loại sách. Chọn xem thông tin sách vào cho sách vào giỏ hàng sau đó thanh toán.



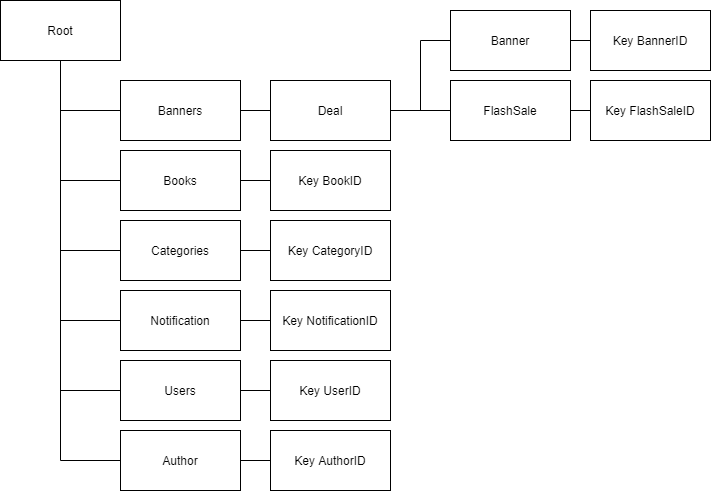
**USE CASE:**

****

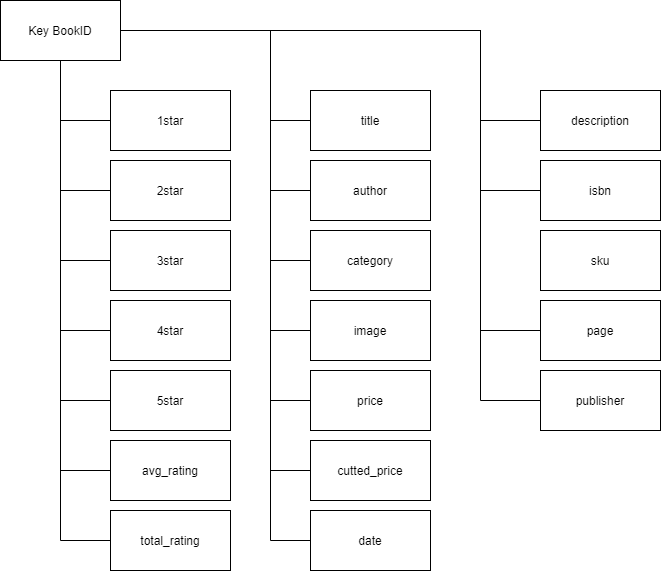
**PHÂN RÃ CHỨC NĂNG:**

****

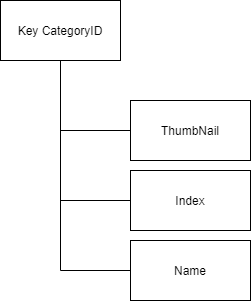
**CẤU TRÚC CÂY DỮ LIỆU**

**1/Root:**

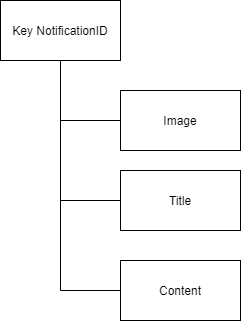
**2/ Book:**

****

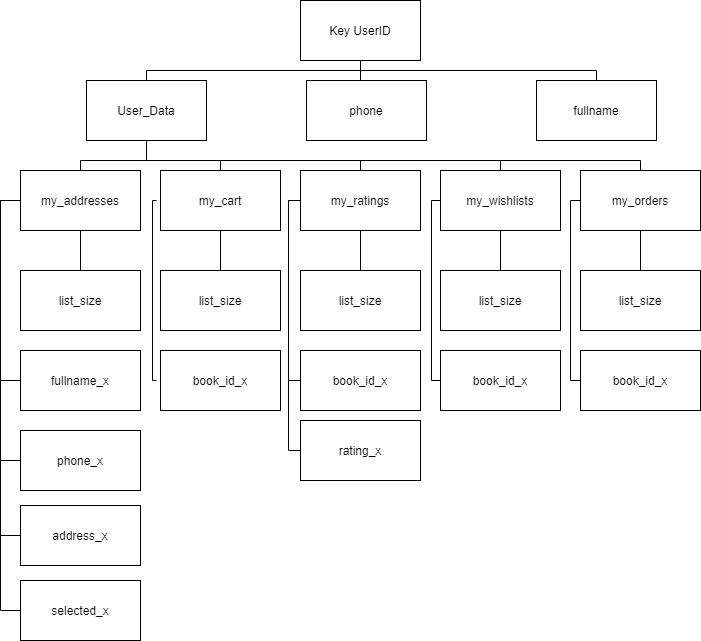
**3/Categories**

****

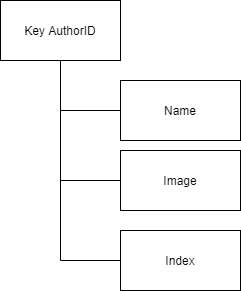
**4/Notification:**

****

**5/User**

****

**6/Author**

****

**II. MÔ TẢ KỸ THUẬT**

**1./Các quan hệ trong hệ thống**

**a/ Các lớp Activity:**

* Activity chính của ứng dụng là MainActivity ( activity\_main.xml). MainActivity chứa các Fragment: HomeFragment ( fragment\_home.xml), CategoryFragment ( fragment\_category.xml), SearchFragment ( fragment\_search.xml), NotificationFragment ( fragment\_notification.xml), UserFragment (fragment\_user.xml).
* Khi ứng dụng được chạy SplashActivity (activity\_splash.xml) sẽ được hiển thị, sau đó kiểm tra xem ứng dụng đã được đăng nhập hay chưa nếu chưa thì gọi đến RegisterActivity (activity\_register.xml) để người dùng đăng nhập hoặc đăng ký.
* AddAddressActivity ( activity\_add\_address.xml)sẽ được gọi khi người dùng thanh toán nếu chưa có địa chỉ giao hàng hoặc được UserFragment gọi khi người dùng muốn thêm địa chỉ giao hàng.
* BookDetailActivity (activity\_book\_detail.xml)được gọi khi người dùng click chọn vào sách muốn xem chi tiết.
* CategoryActivity (activity\_category.xml)được gọi khi người dùng muốn xem sách theo thể loại.
* ViewAllActivity (activity\_view\_all.xml)được gọi khi người dùng click vào button xem thêm ở HomeFragment.
* AuthorActivity (activity\_author.xml)được gọi khi người dùng click vào 1 trong những tác giả trong danh sách tác giả nổi bật.
* MyAddressActivity (activity\_my\_address.xml)được gọi khi người dùng click vào sổ địa chỉ để xem danh sách Địa Chỉ của mình.
* DeliveryActivity (activity\_delivery.xml)được gọi khi người dùng xác nhận địa chỉ giao hàng.

**b/ Các lớp Adapter và Recycler view:**

* AddressAdapter (kế thừa từ lớp RecyclerView.Adapter ) dùng để làm adapter cho RecyclerView Address trong MyAddressActivity và AddAddressActivity.
* AuthorAdapter (kế thừa từ lớp BaseAdapter ) dùng để làm adapter cho gridView trong HomeFragment.
* BookDetailAdapter (Kế thừa từ lớp FragmentPagerAdapter ) dùng để làm adapter cho ViewPager trong BookDetailActivity.
* BookImageAdapter (Kế thừa từ lớp PagerAdapter ) dùng để làm adapter cho ViewPager trong BookDetailActivity.
* BookSpecificationAdapter (Kế thừa từ lớp RecyclerView.Adapter) dùng để làm adapter cho RecyclerView chứa thông tin sách trong BookDetailActivity.
* CartAdapter ( kế thừa từ lớp RecyclerView.Adapter) dùng để làm adapter cho recyclerview trong MyCartFragment.
* CategoryAdapter ( kế thừa từ lớp RecyclerView.Adapter) dùng để làm adapter cho RecyclerView trong CategoryActivity.
* HomePageAdapter (kế thừa từ lớp RecyclerView.Adapter) dùng để tập hợp các adapter như sliderAdapter, horizontalBookAdapter và GridBookLayoutAdapter cho RecyclerView ở HomeFragment.
* MyOrderAdapter (kế thừa từ lớp RecyclerView.Adapter) dùng để làm adapter cho RecyclerView trong MyOrderFragment.
* NotificationAdapter (kế thừa từ lớp RecyclerView.Adapter) dùng để làm adapter cho RecyclerView trong NotificationFragment.
* RewardAdapter (Kế thừa từ lớp RecyclerView.Adapter ) dùng để làm adapter cho RecyclerView trong MyRewardFragment.

**c/ Các lớp còn lại:**

* DataBaseQueries dùng để chứa các câu truy vấn đến CSDL FireBase và truyền lại cho adapter.
* Các lớp Model có các trường để khởi tạo đối tượng khi đổ dữ liệu về từ firebase.

**2./Các kỹ thuật, công nghệ sử dụng**

**a/ Lưu trữ:**

* Sử dụng FireBase FireStore là một CSDL đám mây được phát triển bởi Google.

**b/ Hiển thị:**

* Dùng RecyclerView để hiển thị các Sách trên Activity chính. Các bước cài đặt:

+ Tạo một view RecyclerView trên layout của Activity.

+ Tạo lớp Adapter (kế thừa RecyclerView.Adater) để làm adapter cho Activity đó.

+ Gán RecyclerView trên layout với Activity và nạp adapter vào RecyclerView được gán.

* Sử dụng thư viện Glide (<https://github.com/bumptech/glide>) để render ảnh từ url được lưu trên cơ sở dữ liệu. Khi ứng dụng được chạy lần đầu tiên các ảnh sẽ được cache lại lần sau chạy sẽ nhanh hơn lần đầu.