

**TÀI LIỆU DỰ ÁN**

**QUẢN LÝ THƯ VIỆN - LibMana**



|  |  |
| --- | --- |
| Giảng viên | : Nguyễn Văn Tèo |
| Học viên | : Phạm Văn Sơn |
| Lớp/Kỳ | : PT12345/FA20 |

MARCH 12, 2021

PHƯƠNG NAM LIBRARY

MỤC LỤC

[1](#_30j0zll) Giới thiệu dự án 3

[1.1](#_1fob9te) Giới thiệu PNLib 3

[1.2](#_3znysh7) Yêu cầu của PNLib 3

[1.3](#_2et92p0) Lập kế hoạch dự án 3

[2](#_tyjcwt) Phân tích yêu cầu khách hàng 3

[2.1](#_3dy6vkm) Sơ đồ Use Case 3

[2.2](#_1t3h5sf) Đặc tả yêu cầu hệ thống (SRS) 3

[2.2.1](#_4d34og8) Quản lý THÀNH VIÊN 4

[2.2.2](#_2s8eyo1) Quản lý LOẠI SÁCH 4

[2.2.3](#_17dp8vu) Quản lý SÁCH 4

[2.2.4](#_3rdcrjn) Quản lý PHIẾU MƯỢN 4

[2.2.5](#_26in1rg) thống kê 4

[2.2.6](#_lnxbz9) Đăng nhập 4

[2.2.7](#_35nkun2) Đổi mật khẩu 4

[2.3](#_1ksv4uv) Sơ đồ triển khai và yêu cầu hệ thống 4

[2.3.1](#_44sinio) Sơ đồ triển khai 4

[2.3.2](#_2jxsxqh) Yêu cầu hệ thống 5

[3](#_z337ya) Thiết kế ứng dụng 5

[3.1](#_3j2qqm3) Mô hình công nghệ ứng dụng 5

[3.2](#_1y810tw) Thực thể 5

[3.2.1](#_4i7ojhp) Sơ đồ quan hệ thực thể (ERD) 5

[3.2.2](#_2xcytpi) Chi tiết thực thể 6

[3.3](#_1ci93xb) Giao diện 7

[3.3.1](#_3whwml4) Sơ đồ tổ chức giao diện 7

[3.3.2](#_2bn6wsx) Giao diện MÀN HÌNH CHÍNH 7

[3.3.3](#_qsh70q) Giao diện quản lý 8

[3.3.4](#_3as4poj) Các giao diện hỗ trợ khác 10

[4](#_1pxezwc) Thực hiện dự án 11

[4.1](#_49x2ik5) Tạo giao diện 11

[4.1.1](#_2p2csry) Màn hình chính (MainActivity) 11

[4.1.2](#_147n2zr) Các màn hình quản lý 12

[4.1.3](#_3o7alnk) Màn hình thống kê top 10 (fragment\_top) 13

[4.1.4](#_23ckvvd) Màn hình thống kê doanh thu (fragment\_doanh\_thu) 13

[4.1.5](#_ihv636) Các giao diện hỗ trợ khác 13

[4.2](#_32hioqz) Tạo CSDL với SQLITE 14

[4.2.1](#_1hmsyys) Sơ đồ quan hệ 14

[4.2.2](#_41mghml) Chi tiết các bảng 15

[4.3](#_2grqrue) Lập trình CSDL (SQLiteOpenHelper+Model+DAO) 16

[4.3.1](#_vx1227) SQLiteOpenHelper 16

[4.3.2](#_3fwokq0) Model Class và DAO 16

[4.4](#_1v1yuxt) Lập trình chức năng 17

[4.4.1](#_4f1mdlm) Màn hình chính (MainActivity) 17

[4.4.2](#_2u6wntf) Các giao diện hỗ trợ 18

[4.4.3](#_19c6y18) Các màn hình quản lý 18

[4.4.4](#_3tbugp1) MÀN HÌNH THỐNG KÊ 18

[5](#_28h4qwu) Kiểm thử phần mềm và sửa lỗi 19

[5.1](#_nmf14n) LoginActivity 19

[5.2](#_37m2jsg) AddUserFragment 19

[5.3](#_1mrcu09) … 19

[6](#_46r0co2) Đóng gói và triển khai 19

[6.1](#_2lwamvv) Sản phẩm phần mềm 19

[6.2](#_111kx3o) Hướng dẫn cài đặt 19

[7](#_3l18frh) KẾT LUẬN 20

[7.1](#_206ipza) Khó khăn 20

[7.2](#_4k668n3) Thuận lợi 20

# Giới thiệu dự án

## Giới thiệu PNLib

Thư viện Phương Nam (PNLib) là một thư viện nhỏ, chuyên cho thuê các loại sách phục vụ cho các bạn đọc như “Kinh tế”, “Ngoại Ngữ”, “Công nghệ thông tin”, “Ẩm thực”, “Sức Khoẻ” … Việc quản lý các đầu sách, các phiếu mượn sách, thành viên hiện đang được thư viện quản lý trên sổ sách bằng tay. Việc này gây khó khăn cho thư viện, tốn thời gian ghi chép, và sai sót nhiều trong thống kê.

Vì vậy, PNLib mong muốn xây dựng một phần mềm chạy trên thiết bị di động cá nhân để hỗ trợ giải quyết các khó khăn trên.

## Yêu cầu của PNLib

QUẢN LÝ

Quản lý các thể loại sách

Quản lý sách

Quản lý thành viên

Quản lý phiếu mượn (mỗi phiếu chỉ cho mượn 01 quyển sách)

THỐNG KÊ

Thống kê 10 sách mượn nhiều nhất.

Thống kê doanh thu theo thời gian chọn (từ ngày -> đến ngày)

BẢO MẬT

Muốn truy cập vào ứng dụng, admin và thủ thư phải đăng nhập.

Admin có quyền tạo tài khoản thủ thư.

Admin & thủ thư có quyền đổi mật khẩu.

CÔNG NGHỆ

Ứng dụng cài đặt trên điện thoại chạy hệ điều hành Android.

JDK: 1.8+

Hệ quản trị CSDL: SQLite

## Lập kế hoạch dự án

* *Lập kế hoạch kế hoạch dự án theo mẫu sau*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Hạng mục** | **Bắt đầu** | **Kết thúc** | **Kết quả** |
| 1 | Phân tích yêu cầu khách hàng | 13/09/2023 | 21/09/2023 | Hoàn thành |
| 1.1 | Vẽ sơ đồ use cases | 13/09/2023 | 21/09/2023 | Hoàn thành |
|  | … |  |  |  |
| 2 | Thiết kế ứng dụng |  |  |  |
| 2.1 | Thiết kế mô hình công nghệ |  |  |  |
|  | … |  |  |  |
| 3 | Thực hiện dự án |  |  |  |
| 3.1 | Tạo giao diện |  |  |  |
|  | … |  |  |  |
| 4 | Kiểm thử |  |  |  |
| 4.1 | Xây dựng kịch bảng kiểm thử |  |  |  |
|  | … |  |  |  |
| 5 | Đóng gói & triển khai |  |  |  |
| 5.1 | Đóng gói sản phẩm |  |  |  |
|  | … |  |  |  |

# Phân tích yêu cầu khách hàng

* *Tiến hành phân tích yêu cầu khách hàng theo các bước sau đây*

## Sơ đồ Use Case

A diagram of a diagram

Description automatically generated

## Đặc tả yêu cầu hệ thống (SRS)

Quản lý sách

* Làm công việc liệt kê sách, thêm, sửa, xóa. Mục đích quản lý sách
* Điều kiện hoạt động: đăng nhập hệ thống
* Đối tượng sử dụng: quản trị, thủ thư

Quản lý loại sách

* Làm công việc liệt kê loại sách, thêm, sửa, xóa. Mục đích quản lý phiếu mượn
* Điều kiện hoạt động: đăng nhập hệ thống
* Đối tượng sử dụng: quản trị, thủ thư

Quản lý phiếu mượn

* Làm công việc liệt kê phiếu mượn, thêm, sửa, xóa. Mục đích quản lý loại sách
* Điều kiện hoạt động: đăng nhập hệ thống
* Đối tượng sử dụng: quản trị, thủ thư

Quản lý thành viên

* Làm công việc liệt kê thành viên, thêm, sửa, xóa. Mục đích quản lý thành viên
* Điều kiện hoạt động: đăng nhập hệ thống
* Đối tượng sử dụng: quản trị, thủ thư

### Quản lý THÀNH VIÊN

* **Mô tả chức năng:**

Chức năng quản lý thành viên được sử dụng để quản lý thông tin thành viên. ức năng này là liệt kê danh sách thành viên, xem thông tin chi tiết củYêu cầu của cha mỗi thành viên, thêm thàn h viên mới, cập nhật thông tin hoặc xóa thành viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi thành viên gồm: mã loại sách, họ tên thành viên, năm sinh thành viên.

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì tất cả người dùng có thể sử dụng chức năng này.

### Quản lý LOẠI SÁCH

* **Mô tả chức năng:**
  + Chức năng quản lý thành viên được sử dụng để quản lý thông tin thành viên. ức năng này là liệt kê loại sách, xem thông tin chi tiết mỗi loại sách, thêm loại sách mới, cập nhật thông tin hoặc xóa loại sách đã tồn tài.
* **Dữ liệu liên quan:**
  + Thông tin của mỗi loại sách gồm: mã loại sách, tên loại sách
* **Đối tượng sử dụng:**
  + Sau khi đăng nhập thì tất cả người dùng có thể sử dụng chức năng này.

### Quản lý SÁCH

* **Mô tả chức năng:**
  + Chức năng quản lý thành viên được sử dụng để quản lý thông tin thành viên. ức năng này là liệt kê sách, xem thông tin chi tiết, thêm sách mới, cập nhật thông tin hoặc xóa sách đã tồn tài.
* **Dữ liệu liên quan:**
  + Thông tin của mỗi sách gồm: mã sách, tên sách
* **Đối tượng sử dụng:**
  + Sau khi đăng nhập thì tất cả người dùng có thể sử dụng chức năng này.

### Quản lý PHIẾU MƯỢN

* **Mô tả chức năng:**
  + Chức năng quản lý Phiếu mượn được sử dụng để quản lý thông tin phiếu mượn. Chức năng này là liệt kê phiếu mượn, xem thông tin chi tiết, thêm phiếu mượn mới, cập nhật thông tin hoặc xóa phiếu mượn đã tồn tài.
* **Dữ liệu liên quan:**
  + Thông tin của mỗi phiếu mượn gồm: mã phiếu mượn, tên phiếu mượn
* **Đối tượng sử dụng:**
  + Sau khi đăng nhập thì tất cả người dùng có thể sử dụng chức năng này.

### thống kê

* **Mô tả chức năng:**
  + Chức năng thống kê được sử dụng để thống kê, tóm tắt dữ liệu
* **Dữ liệu liên quan:**
  + Thông tin thống kê gồm: tên loại sách, số lượng
* **Đối tượng sử dụng:**
  + Sau khi đăng nhập thì tất cả người dùng có thể sử dụng chức năng này.

### Đăng nhập

* **Mô tả chức năng:**
  + Chức năng đăng nhập được sử dụng để đăng nhập, truy cập vào hệ thống
* **Dữ liệu liên quan:**
  + Thông tin đăng nhập gồm: username, password
* **Đối tượng sử dụng:**
  + Tất cả người dùng có thể sử dụng chức năng này

### Đổi mật khẩu

* **Mô tả chức năng:**
  + Chức năng đổi mật khẩu được sử dụng để đổi mật khẩu tài khoản người dùng
* **Dữ liệu liên quan:**
  + Thông tin đổi mật khẩu gồm: mật khẩu cũ, mật khẩu mới, nhập lại mật khẩu mới
* **Đối tượng sử dụng:**
  + Tất cả người dùng có thể sử dụng chức năng này

## Sơ đồ triển khai và yêu cầu hệ thống

### Sơ đồ triển khai

A diagram of a business process

Description automatically generated

### Yêu cầu hệ thống

A diagram of a business process

Description automatically generated

# Thiết kế ứng dụng

* *Dựa vào tài liệu đã phân tích ở mục 2 để phác thảo:*
  + *Mô hình công nghệ*
  + *Giao diện*
  + *Thực thể*

## Mô hình công nghệ ứng dụng

* *Phác thảo mô hình công nghệ ứng dụng của hệ thống phần mềm. Giải thích rõ các thành phần trong mô hình*

## Thực thể

* *Căn cứ vào SRS để thiết kế các thực thể. Cần phải hoàn thiện 2 phần theo yêu cầu sau đây.*

### Sơ đồ quan hệ thực thể (ERD)

A diagram of a diagram

Description automatically generated

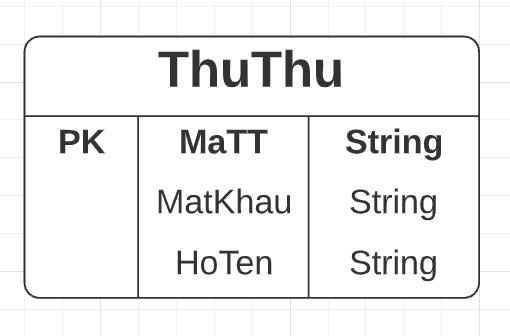
A diagram of a computer

Description automatically generated5

### Chi tiết thực thể

* *Thiết kế chi tiết thực thể có nghĩa là thiết kế các thuộc tính cần thiết cho mỗi thực thể (căn cứ vào phần dữ liệu liên quan của SRS).*

#### Thực thể Thủ Thư



|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| MaTT | String | Mã thủ thư |
| MatKhau | String | Mật khẩu |
| HoTen | String | Họ và tên thủ thư |

#### Thực thể Phiếu Mượn

A screenshot of a computer

Description automatically generated

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| MaPM | String | Mã phiếu mượn |
| MaTV | String | Mã thành viên |
| MaSach | String | Mã sách |
| MaTT | String | Mã thủ thư |

#### Thực thể Thành viên

A screenshot of a computer screen

Description automatically generated

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| MaTV | String | Mã thành viên |
| MatKhau | String | Mật khẩu |
| HoTen | String | Họ và tên thành viên |

#### Thực thể Loại Sách

A graph with black text

Description automatically generated

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| MaLoai | String | Mã loại sách |
| TenLoai | String | Tên loại sách |

#### Thực thể Sách

A screenshot of a computer screen

Description automatically generated

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| MaSach | String | Mã sách |
| MaLoai | String | Mã loại sách |
| TenSach | String | Tên sách |

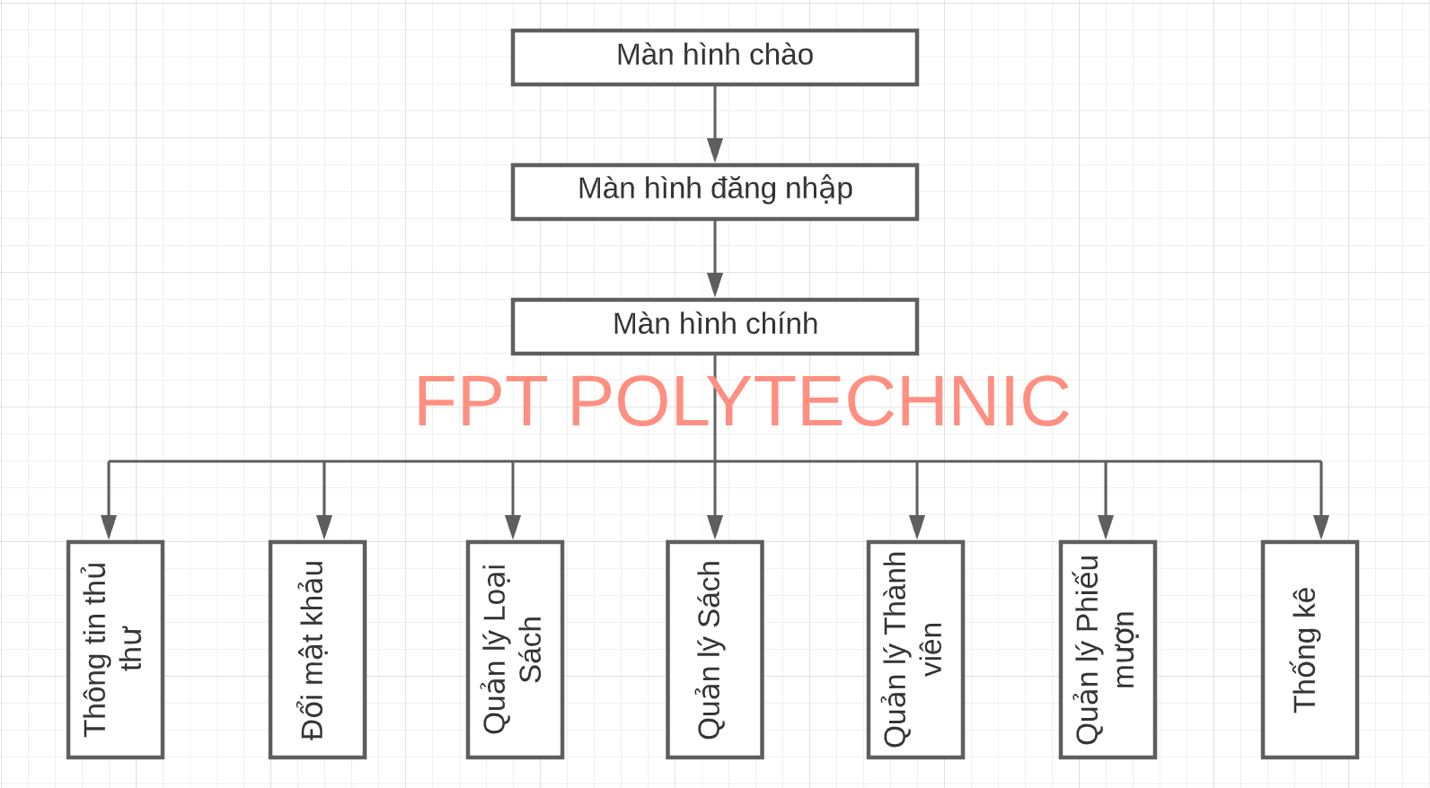
…

## Giao diện

* *Chú ý: phần này chỉ phác thảo giao diện chứ không phải xây dựng nên bạn có thể sử dụng bất ký công cụ gì thậm chí vẽ trên giấy rồi chụp hình đưa vào đây.*

### Sơ đồ tổ chức giao diện

* *Phác thảo sơ đồ liên kết giữa các giao diện chức năng trong ứng dụng*

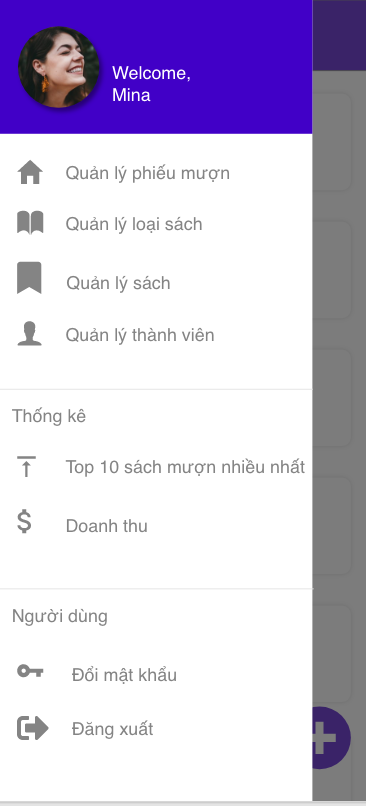


* *Mô tả sơ đồ*

### Giao diện MÀN HÌNH CHÍNH

* *Phác thảo giao diện cho màn hình chính và mô tả các hoạt động trên giao diện*

**Giao diện:**



**Mô tả hoạt động**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | Navigation Drawer | Vuốt từ trái/ Click Menu icon | Show Navigation Drawer |
| 2 | [nav\_PhieuMuon] | Click | Chuyển Fragment\_PhieuMuon |
| 3 | [nav\_LoaiSach] | Click | Chuyển Fragment\_LoaiSach |
| 4 | [nav\_Sach] | Click | Chuyển Fragment\_Sach |
|  | … |  |  |

### Giao diện quản lý

* *Phác thảo giao diện cho tất cả các cửa sổ chức năng và các cửa sổ hỗ trợ tổ chức*

#### Màn hình quản lý Thành Viên

**Giao diện**

**Mô tả hoạt động**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | … |  |  |
| 2 |  |  |  |

#### Màn hình quản lý Loại Sách

**Giao diện:**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình quản lý Sách

**Giao diện:**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình quản lý Phiếu Mượn

**Giao diện:**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình thống kê

**Giao diện:**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

### Các giao diện hỗ trợ khác

#### Màn hình chào

**Giao diện**

**Mô tả hoạt động**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình đăng nhập

**Giao diện**

**Mô tả hoạt động**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình thêm người Dùng (chỉ dành cho admin)

**Giao diện:**

**<Hình>**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

#### Màn hình đổi mật khẩu

**Giao diện:**

**<Hình>**

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **@ID** | **Sự kiện** | **Mô tả hoạt động** |
| **1** | **….** |  |  |
| **2** |  |  |  |

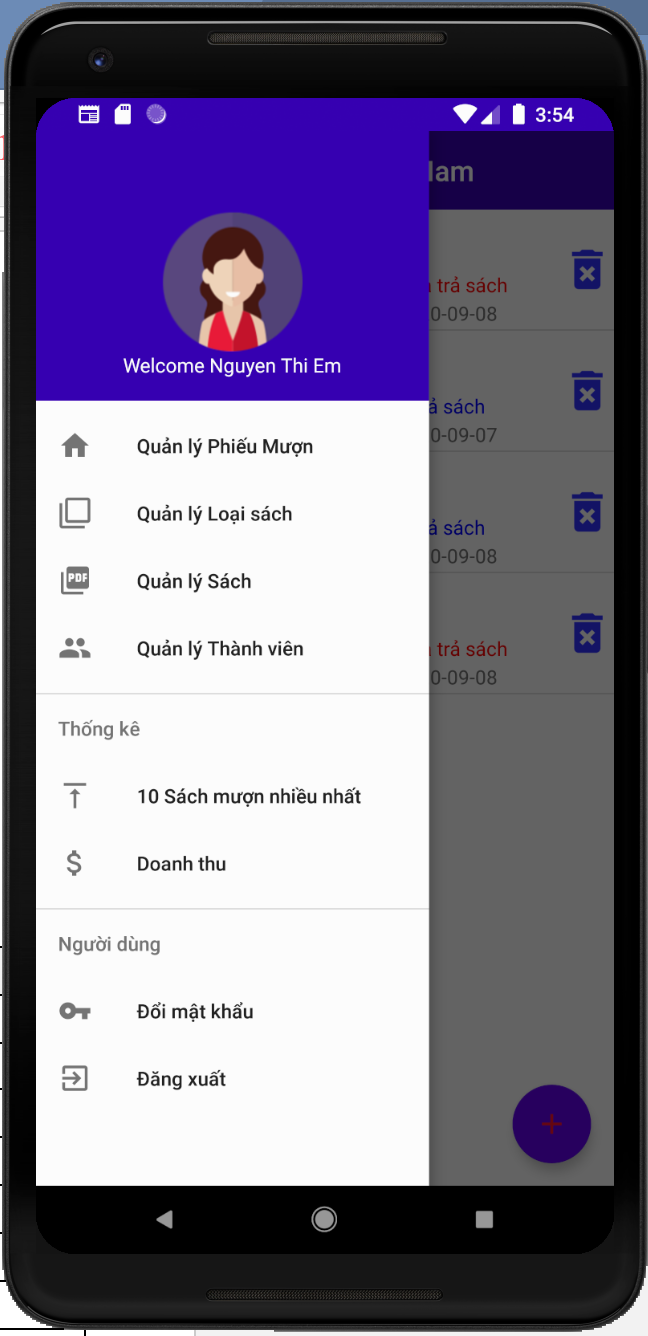
# Thực hiện dự án

## Tạo giao diện

* *Mỗi của sổ chức năng cần làm rõ các mục sau đây:*
  + *Hình giao diện của cửa sổ chức năng*
  + *Bảng mô tả các điều khiển (control) và tên của chúng*

### Màn hình chính (MainActivity)

**Giao diện**

****

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **DrawerLayout** | **drawer\_layout** | **layout\_width="match\_parent" layout\_height="match\_parent"** |
| **include** |  | **layout="@layout/toolbar** |
| **FrameLayout** | **flContent** | **layout\_width="match\_parent" layout\_height="match\_parent"** |
| **NavigationView** | **nvView** | **menu="@menu/drawer\_view" headerLayout="@layout/nav\_header"** |

### Các màn hình quản lý

#### Màn hình quản lý Phiếu mượn (fragment\_phieu\_muon)

**Giao diện**

**Đặt tên các điều khiển**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình quản lý Loại sách (fragment\_loai\_sach)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình quản lý Sách (fragment\_sach)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình quản lý Thành viên (fragment\_thanh\_vien)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

### Màn hình thống kê top 10 (fragment\_top)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

### Màn hình thống kê doanh thu (fragment\_doanh\_thu)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

### Các giao diện hỗ trợ khác

#### Màn hình chào (activity\_splash\_screen)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình đăng nhập (activity\_login)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình Thêm thủ thư (fragment\_add\_user)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

#### Màn hình đổi mật khẩu (fragment\_change\_pass)

**Giao diện**

**Đặt id các view**

|  |  |  |
| --- | --- | --- |
| **VIEW** | **ID** | **Các thuộc tính** |
| **…..** |  |  |

## Tạo CSDL với SQLITE

### Sơ đồ quan hệ

* *Hình: Sơ đồ quan hệ*
* *Ràng buộc: Mô tả ràng buộc của các quan hệ*

### Chi tiết các bảng

* *Mỗi bảng cần viết theo hướng dẫn sau đây:*

#### Bảng ThuThu

**Cấu trúc bảng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| MaTT | TEXT | PK | Mã thủ thư |
| MatKhau | TEXT | NOT NULL | Mật khẩu |
| HoTen | TEXT | NOT NULL | Họ và tên |

**Mã lệnh tạo bảng**

|  |
| --- |
| *//Tao bang Thu Thu* String createTableThuThu=  **"create table ThuThu ("** +  **"maTT TEXT PRIMARY KEY, "** +  **"hoTen TEXT NOT NULL, "** +  **"matKhau TEXT NOT NULL)"**; db.execSQL(createTableThuThu); |

**Câu lệnh SQL cơ bản**

|  |
| --- |
| **db**.insert(**"ThuThu"**, **null**, values);  **db**.update(**"ThuThu"**, values, **"maTT=?"**, **new** String[]{obj.**maTT**});  **db**.delete(**"ThuThu"**, **"maTT=?"**, **new** String[]{id});  **db**.execSQL(“SELECT \* FROM **ThuThu** WHERE **maTT=?"**, **new** String[]{obj.**maTT**});  **db**.execSQL(“SELECT \* FROM **ThuThu”**); |

#### Bảng ThanhVien

**Cấu trúc bảng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
|  |  |  |  |

**Mã lệnh tạo bảng**

|  |
| --- |
|  |

**Câu lệnh SQL cơ bản**

|  |
| --- |
|  |

#### Bảng ….

## Lập trình CSDL (SQLiteOpenHelper+Model+DAO)

### SQLiteOpenHelper

* *Mô tả lớp SQLiteOpenHelper*
* *Hướng dẫn sử dụng các phương thức tiện ích của SQLiteOpenHelper*
* *Code DbHelper kế thừa SQLiteOpenHelper*

**Mô tả:**

SQLiteOpenHelper là lớp tiện ích hỗ trợ lập trình Android với SQLite. Khi kế thừ Lớp này cần ghi đè 2 phương thức được mô tả trong phần hướng dẫn sử dụng sau đây.

**Hướng dẫn sử dụng:**

* onCreate(SQLiteDatabase db): chứa các lệnh tạo table (có thể chèn dữ liệu test)
* onUpgrade(SQLiteDatabase db, int i, int i1): chứa các lệnh xoá table và gọi lại onCreate khi nâng version

**Code: ….**

**public class dbhelper extends SQLiteOpenHelper {**

**public dbhelper(@Nullable Context context) {**

**super(context, "DANGKYMONHOC", null, 4);**

**}**

**@Override**

**public void onCreate(SQLiteDatabase sqLiteDatabase) {**

**String dbThuThu = "Create table thuthu(matt text primary key, hoten text, matkhau text)";**

**sqLiteDatabase.execSQL(dbThuThu);**

**String dbThanhvien = "Create table thanhvien(matv integer primary key autoincrement,hoten text, namsinh text)";**

**sqLiteDatabase.execSQL(dbThanhvien);**

**String dbLoaiSach = "Create table loaisach(maloai integer primary key autoincrement,tenloai text)";**

**sqLiteDatabase.execSQL(dbLoaiSach);**

**String dbSach = "Create table sach(masach integer primary key autoincrement, tensach text, giathue float,maloai integer references loaisach(maloai))";**

**sqLiteDatabase.execSQL(dbSach);**

**String dbPhieuMuon = "Create table phieumuon(mapm integer primary key autoincrement,matv integer references thanhvien(matv),matt text references thuthu(matt),masach integer references sach(masach),ngay text, trasach integer, tienthue float )";**

**sqLiteDatabase.execSQL(dbPhieuMuon);**

**//data mẫu**

**sqLiteDatabase.execSQL("insert into loaisach values(1,'Sách thiếu nhi'),(2,'Sách tình cảm'),(3,'Sách giáo khoa')");**

**sqLiteDatabase.execSQL("insert into sach values(1,'Hãy đợi đấy',2500,1),(2,'Thằng cuội',1000,1),(3,'Lập trình Android',2000,3)");**

**sqLiteDatabase.execSQL("insert into thuthu values('thuthu01','Nguyễn Văn An','abc123'),('thuthu02','Phạm Văn B','abc124')");**

**sqLiteDatabase.execSQL("insert into thanhvien values(1,'Phạm Thị Linh','2004'),(2,'Nguyễn Văn An','2003')");**

**sqLiteDatabase.execSQL("insert into phieumuon values(1,1,'thuthu01',1,'2023/09/29',1,30000.0)");**

**}**

**@Override**

**public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {**

**if(i!= i1){**

**sqLiteDatabase.execSQL("DROP TABLE IF EXISTS thuthu");**

**sqLiteDatabase.execSQL("DROP TABLE IF EXISTS thanhvien");**

**sqLiteDatabase.execSQL("DROP TABLE IF EXISTS loaisach");**

**sqLiteDatabase.execSQL("DROP TABLE IF EXISTS sach");**

**sqLiteDatabase.execSQL("DROP TABLE IF EXISTS phieumuon");**

**onCreate(sqLiteDatabase);**

**}**

**}**

**}**

### Model Class và DAO

* *Mỗi mục cần làm rõ 2 vấn đề sau*
  + *Mô hình và mô tả lớp thực thể*
  + *Mô hình và mô tả lớp DAO*

1. Model Phiếu Mượn code:

public class PhieuMuon {

private int mapm,matv;

private String matt;

private int masach;

private String ngay;

private int trasach;

private float tienthue;}

1. Model Sách code:

public class Sach {

private int masach;

private String tensach;

private float giathue;

private int maloai;

}

1. Model Thành Viên code:

public class ThanhVien {

private int matv;

private String hoTen;

private String namSinh;

}

1. Dao Phiếu Mượn code:

public class PhieuMuonDao {

dbhelper dbhelper;

public PhieuMuonDao(Context context){

dbhelper = new dbhelper(context);

}

//Lấy toàn bộ đầu sách có ở trong thư viên

public ArrayList<PhieuMuon> getDSPM(){

ArrayList<PhieuMuon> list = new ArrayList<>();

SQLiteDatabase sqLiteDatabase = dbhelper.getReadableDatabase();

try {

Cursor cursor = sqLiteDatabase.rawQuery("Select\*from phieumuon",null);

if(cursor.getCount()!=0){

cursor.moveToFirst();

do{

list.add(new PhieuMuon(cursor.getInt(0),cursor.getInt(1),cursor.getString(2),cursor.getInt(3),cursor.getString(4),cursor.getInt(5),cursor.getFloat(6)));

}while(cursor.moveToNext());

}

}catch (Exception ex){

Log.*i*(*TAG*,"Lỗi",ex);

}

return list;

}

public boolean insert(PhieuMuon pm){

SQLiteDatabase db = dbhelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put("matv",pm.getMatv());

values.put("matt",pm.getMatt());

values.put("masach",pm.getMasach());

values.put("ngay",pm.getNgay());

values.put("trasach",pm.getTrasach());

values.put("tienthue",pm.getTienthue());

long row = db.insert("phieumuon",null,values);

return (row>0);

}

public boolean update (PhieuMuon pm){

SQLiteDatabase db = dbhelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put("matv",pm.getMatv());

values.put("matt",pm.getMatt());

values.put("masach",pm.getMasach());

values.put("ngay",pm.getNgay());

values.put("trasach",pm.getTrasach());

values.put("tienthue",pm.getTienthue());

long row = db.update("phieumuon",values,"mapm=?",new String[]{String.*valueOf*(pm.getMapm())});

return (row>0);

}

public boolean delete (int mapm){

SQLiteDatabase db = dbhelper.getWritableDatabase();

long row = db.delete("phieumuon","mapm=?",new String[]{String.*valueOf*(mapm)});

return (row>0);

}

}

1. Dao Sách code:

public class SachDAO {

dbhelper dbhelper;

public SachDAO(Context context){

dbhelper = new dbhelper(context);

}

//Lấy toàn bộ đầu sách có ở trong thư viên

public ArrayList<Sach> getDSDauSach(){

ArrayList<Sach> list = new ArrayList<>();

SQLiteDatabase sqLiteDatabase = dbhelper.getReadableDatabase();

try {

Cursor cursor = sqLiteDatabase.rawQuery("Select\*from sach",null);

if(cursor.getCount()!=0){

cursor.moveToFirst();

do{

list.add(new Sach(cursor.getInt(0),cursor.getString(1),cursor.getFloat(2),cursor.getInt(3)));

}while(cursor.moveToNext());

}

}catch (Exception ex){

Log.*i*(*TAG*,"Lỗi",ex);

}

return list;

}

public boolean insert(Sach s){

SQLiteDatabase db = dbhelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put("masach",s.getMasach());

values.put("tensach",s.getTensach());

values.put("giathue",s.getGiathue());

values.put("maloai",s.getMaloai());

long row = db.insert("sach",null,values);

return (row>0);

}

public boolean update (Sach s){

SQLiteDatabase db = dbhelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put("masach",s.getMasach());

values.put("tensach",s.getTensach());

values.put("giathue",s.getGiathue());

values.put("maloai",s.getMaloai());

long row = db.update("sach",values,"masach=?",new String[]{String.*valueOf*(s.getMasach())});

return (row>0);

}

public boolean delete (int masach){

SQLiteDatabase db = dbhelper.getWritableDatabase();

long row = db.delete("sach","masach=?",new String[]{String.*valueOf*(masach)});

return (row>0);

}

}

1. Dao Thành Viên code:

public class ThanhVienDAO {

dbhelper dbhelper;

public ThanhVienDAO(Context context){

dbhelper = new dbhelper(context);

}

public ArrayList<ThanhVien> getDSTV(){

ArrayList<ThanhVien> list = new ArrayList<>();

SQLiteDatabase sqLiteDatabase = dbhelper.getReadableDatabase();

Cursor cursor = sqLiteDatabase.rawQuery("select\*from thanhvien",null);

if(cursor.getCount()!= 0){

cursor.moveToFirst();

do {

list.add(new ThanhVien(cursor.getInt(0),cursor.getString(1),cursor.getString(2)));

}while (cursor.moveToNext());

}

return list;

}

}

1. Dao Thủ Thư code:

public class ThuThuDao {

dbhelper dbhelper;

public ThuThuDao(Context context){

dbhelper = new dbhelper(context);

}

//đăng nhập

public boolean checkDN(String matt,String matkhau){

SQLiteDatabase sqLiteDatabase = dbhelper.getReadableDatabase();

Cursor cursor = sqLiteDatabase.rawQuery("select\*from thuthu where matt=? and matkhau = ?",new String[]{matt,matkhau});

if(cursor.getCount()!=0){

return true;

}else{

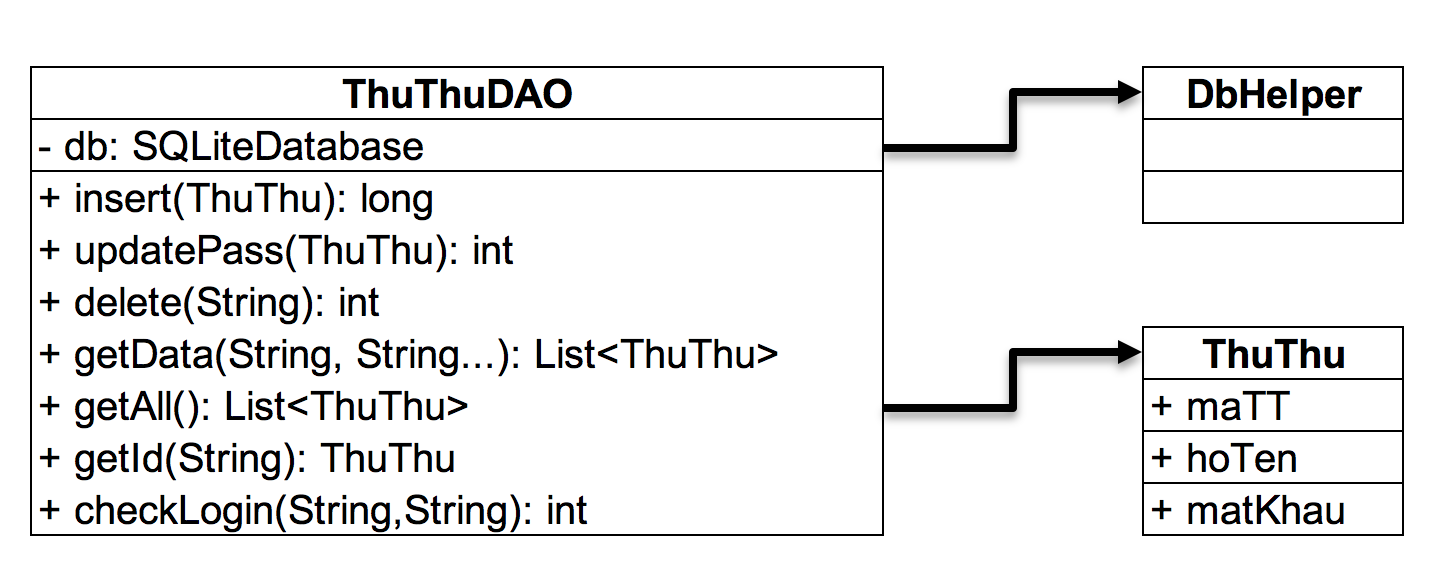
return false;

}

}

}

#### ThuThu và ThuThuDAO



**Diễn giải**

|  |  |  |
| --- | --- | --- |
| **Thành phần** |  | **Mô tả** |
| ThuThu |  | Là lớp thực thể mô tả dữ liệu thủ thư |
| ThuThuDAO |  | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng ThuThu |

#### ThanhVien & ThanhVienDAO

…

#### LoaiSach & LoaiSachDAO

…

#### …

## Lập trình chức năng

* *Mỗi chức năng sau đây cần liệt kê và mô tả tất cả các hành vi của người sử dụng tương tác lên các chức năng của các giao diện (có thể sử dụng lưu đồ thuật toán hoặc các loại sơ đồ Activity Diagram, Sequence Diagram - các loại sơ đồ này được học trong môn Nhập môn kỹ thuật phần mềm)*

### Màn hình chính (MainActivity)

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức/ sự kiện** | **Mô tả** |
| 1 | setNavigationItemSelectedListener() | Điều hướng Navigation |
| 2 | setSupportActionBar() | Sử dụng toolbar |
|  | … |  |

### Các giao diện hỗ trợ

#### Màn hình chào (SplashScreenActivity)

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 |  |  |
| 2 | … |  |

#### Màn hình đăng nhập (LoginActivity)

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 |  |  |
| 2 | … |  |

#### Màn hình đổi mật khẩu (ChangePassFragment)

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 |  |  |
| 2 | … |  |

#### Màn hình thêm người dùng (AddUserFragment)

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 |  |  |
| 2 | … |  |

### Các màn hình quản lý

#### ThanhVienFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | capNhatLV() | GetAll() và cập nhật dữ liệu lên ListView |
| 2 | … |  |

#### LoaiSachFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | capNhatLV() |  |
| 2 | … |  |

#### SachFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | capNhatLV() |  |
| 2 | … |  |

#### PhieuMuonFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | capNhatLV() |  |
| 2 | … |  |

### MÀN HÌNH THỐNG KÊ

#### TopFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | … |  |
|  | … |  |

#### DoanhThuFragment

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | … |  |
|  | … |  |

# Kiểm thử phần mềm và sửa lỗi

## LoginActivity

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | Teo, admin | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | admin, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | admin, admin | Longin thành công |  |

## AddUserFragment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |

## …

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |

# Đóng gói và triển khai

## Sản phẩm phần mềm

|  |  |  |
| --- | --- | --- |
| **TT** | **Thành phần** | **Mô tả** |
| 1 | LibMana.apk | File cài đặt |
| 2 | Readme.txt | Hướng dẫn cài đặt |
| 3 | Guide.docx | Hướng dẫn sử dụng |

## Hướng dẫn cài đặt

* Bước 1:
* Bước 2:
* Bước 3:
* …

# KẾT LUẬN

## Khó khăn

## Thuận lợi