**详细设计规约**

**——愉阅-个人阅读管理app**

目录

[1.引言 3](#_Toc29417477)

[1.1编写目的 3](#_Toc29417478)

[1.2背景与依据 3](#_Toc29417479)

[1.3参考资料 3](#_Toc29417480)

[2.系统软件结构 4](#_Toc29417481)

[3.模块设计 4](#_Toc29417482)

[3.1首页模块 4](#_Toc29417483)

[3.2想读模块 6](#_Toc29417484)

[3.3在读模块 8](#_Toc29417485)

[3.4已读模块 11](#_Toc29417486)

[3.5我的模块 13](#_Toc29417487)

[4.类图 16](#_Toc29417488)

[4.1总设计 16](#_Toc29417489)

[4.2子模块1—个性化推荐 17](#_Toc29417490)

[4.3子模块2—系统推荐书籍 17](#_Toc29417491)

[4.4子模块3—书摘记录管理 18](#_Toc29417492)

[4.5子模块4—阅读进度管理 18](#_Toc29417493)

[4.6子模块5—阅读数据统计 19](#_Toc29417494)

[4.7子模块6—用户信息管理 19](#_Toc29417495)

# 1.引言

## 1.1编写目的

编写详细设计规约的目的是为了使开发人员在完成概要设计的基础上，完成概要设计的各项功能规定，为程序开发者写到实际的程序代码提供依据，它是软件设计阶段所有任务和所有相关人员所需的参考资料。预期的读者为系统的开发者、使用者和指导老师。

## 1.2背景与依据

开发软件系统的名称：愉阅—个人阅读管理app。

开发者：修习软件工程课程的大三学生。

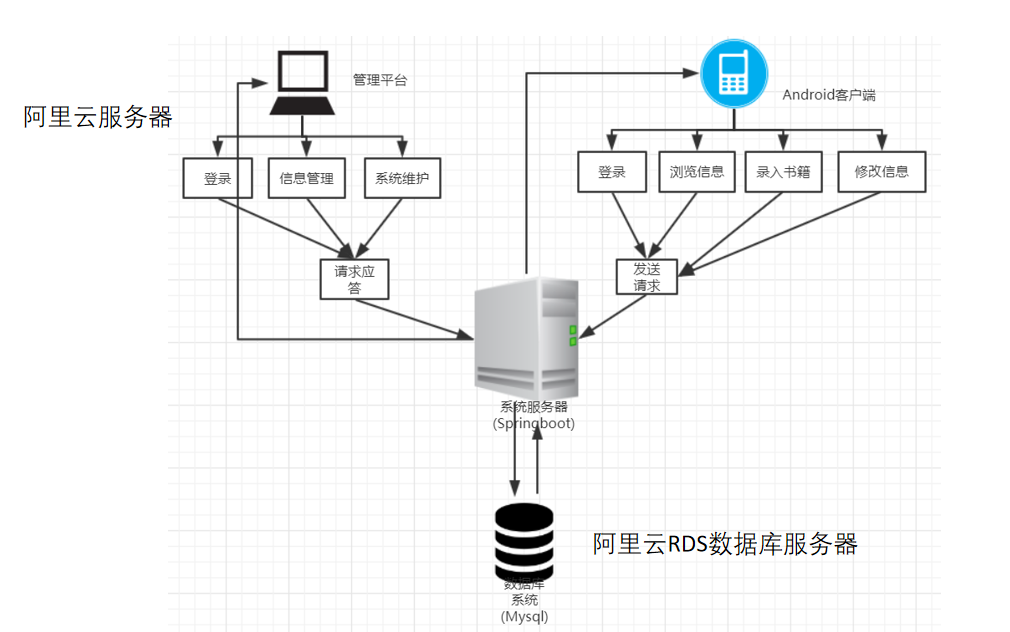
本系统将使用MySQL作为数据库存储系统，阿里云为服务器。

## 1.3参考资料

《Component-Based Software Development》

《A Guide to Programming Logic and Design》

# 2.系统软件结构



# 3.模块设计

## 3.1首页模块

3.1.1 程序描述

用户在此查看推荐书单和个性化推荐，同时可以搜索书籍并查看。

3.1.2 功能

1.用户点击查看相应书单时，返回书单信息的列表给用户。

2.用户输入书名的关键字时，在数据库中查找，若找到则返回给用户书籍信息。

3.用户扫描书籍的ISBN码进行查找书籍时，在数据库中查找，若找到则返回给用户书籍信息。

3.1.3 输入项

1.用户点击的书单选项button的id。

2.用户输入的书名关键字keyword。

3.用户扫描的书籍ISBN码。

3.1.4 输出项

1.相应的推荐书单列表

2.符合搜索keyword的结果书籍的列表

3.符合搜索ISBN的结果书籍的列表

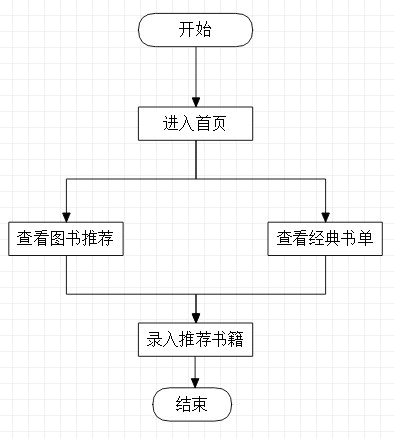
3.1.5 算法

1.获取用户点击的书单选项button的id，调用该button的onClick函数。onClick函数中将使用api（http://139.196.36.97:8080/sbDemo/v2/booklist-management/booklists）调取数据库中相应书单中的书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

2.接收用户输入的书名关键字keyword，确认搜索后使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/titles）调取在数据库中完成搜索后返回的结果书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

3.接收用户扫描的ISBN码，使用ZXING库将ISBN码转换成String类型的ISBN。使用豆瓣图书api，调取符合按ISBN搜索的结果的书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

3.1.6 流程图



## 3.2想读模块

3.2.1 程序描述

用户在此查看个人想读书籍，并可新增想读书籍，或将某本想读书籍标记为在读。

3.2.2 功能

1. 查看想读书籍
2. 手动录入想读书籍
3. 将某本想读书籍标记为在读

3.2.3 输入项

1. 用户信息与书籍状态
2. 录入书籍的书名、ISBN、作者、出版社
3. 被标记为在读的书籍信息与当前用户信息

3.2.4 输出项

1. 想读书籍列表
2. 更新后的想读书籍列表
3. 更新后的想读书籍列表

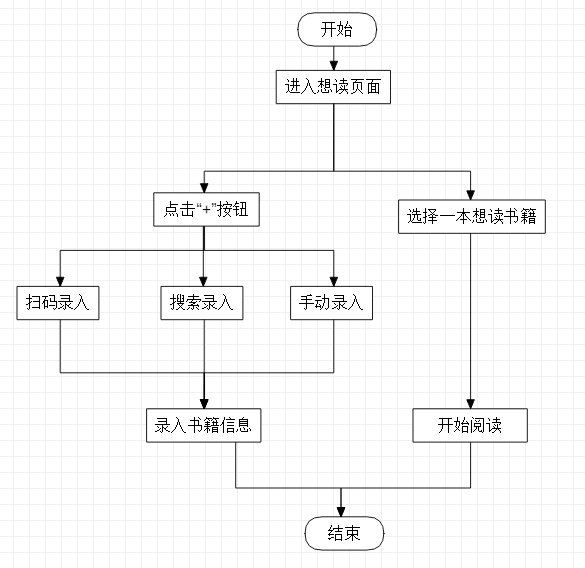
3.2.5 算法

1.获取当前用户的手机号userID与个人书单的状态state，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states）调取在数据库中完成搜索后返回的符合的结果书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

2.获取用户输入的书籍的书名、ISBN、作者、出版社与当前用户的userID，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/books）对数据库进行新增操作。在完成操作后，重新加载view，即使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states） 调取在数据库中当前状态的所有书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

3.获取当前用户的userID与用户点击的想要改变状态的书籍的bookID，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states）给数据库发送修改请求，将该书籍状态改为在读。在完成操作后，重新加载view，即使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states） 调取在数据库中当前状态的所有书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

3.2.6 流程图



## 3.3在读模块

3.3.1 程序描述

用户在此查看个人在读书籍，并可新增想读书籍，或将某本在读书籍标记为已读。用户还可以查看、新增与更改某本书籍的阅读记录。

3.3.2 功能

1. 查看在读书籍
2. 将某本在读书籍标记为已读
3. 查看阅读记录
4. 新增阅读记录
5. 修改阅读记录
6. 删除阅读记录

3.3.3 输入项

1. 用户信息与书籍状态
2. 被标记为已读的书籍信息与当前用户信息
3. 当前用户信息userID和书摘所属的书籍的bookID。
4. 当前用户信息userID和新增记录所属书籍的bookID与记录的信息：开始页码、起始页码。
5. 当前用户信息userID和被修改记录的信息：noteID、感想。
6. 当前用户信息userID和被删除记录的noteID（PK）。

3.3.4 输出项

1.在读书籍列表

2.更新后的在读书籍列表

3.阅读详情页面

4.阅读详情页面

5.阅读详情页面

6.阅读详情页面

3.3.5 算法

1. 获取当前用户的手机号userID与个人书单的状态state，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states）调取在数据库中完成搜索后返回的符合的结果书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

2. 获取当前用户的userID与用户点击的想要改变状态的书籍的bookID，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states）给数据库发送修改请求，将该书籍状态改为已读。在完成操作后，重新加载view，即使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states） 调取在数据库中当前状态的所有书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

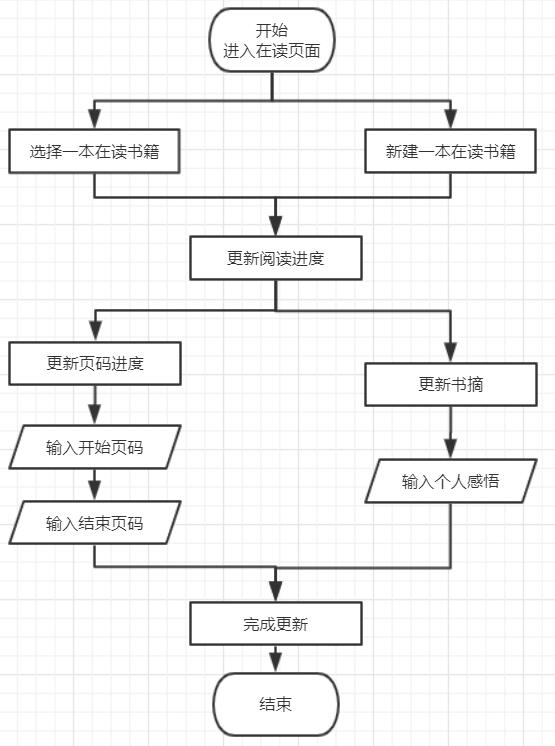
3. 获取当前用户的userID与用户想要查看书摘的所属书籍的bookID，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送Get请求，调取在数据库中完成搜索后返回的符合的结果书摘信息。在完成操作后，重新加载view。

4.获取当前用户的userID与用户想要新增记录所属书籍的bookID与该记录的信息：开始页码、起始页码。使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送POST请求，完成记录的增加操作。在完成操作后，重新加载view。

5. 获取当前用户的userID与用户想要修改的记录的noteID与感想，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送PUT请求，修改相应记录。在完成操作后，重新加载view。

6. 获取用户所选择删除的记录的noteID，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送Delete请求，删除相应记录。在完成操作后，重新加载view。

3.3.6 流程图



## 3.4已读模块

3.4.1 程序描述

用户可以在此查看已读书籍，查看与更改某本书籍的阅读记录。

3.4.2 功能

1. 查看已读书籍
2. 查看阅读记录
3. 修改阅读记录
4. 删除阅读记录

3.4.3 输入项

1. 用户信息与书籍状态
2. 当前用户信息userID和书摘所属的书籍的bookID。
3. 当前用户信息userID和被修改记录的信息：noteID、感想。
4. 当前用户信息userID和被删除记录的noteID（PK）。

3.4.4 输出项

1.已读书籍列表

2.阅读详情页面

3.阅读详情页面

4.阅读详情页面

3.4.5 算法

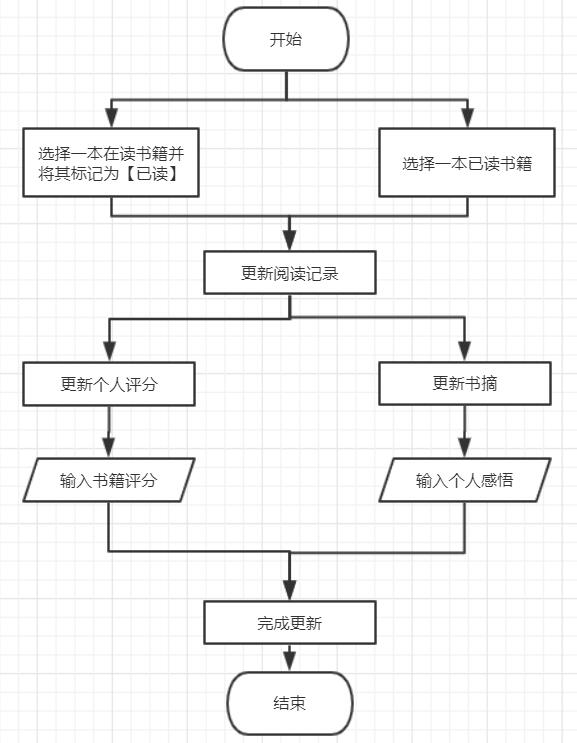
1. 获取当前用户的手机号userID与个人书单的状态state，使用api（http://139.196.36.97:8080/sbDemo/v2/read-management/states）调取在数据库中完成搜索后返回的符合的结果书籍信息。将api返回的数据进行处理后，使用Adapter将数据放置于view上，返回给用户相应的view。

2. 获取当前用户的userID与用户想要查看书摘的所属书籍的bookID，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送Get请求，调取在数据库中完成搜索后返回的符合的结果书摘信息。在完成操作后，重新加载view。

3. 获取当前用户的userID与用户想要修改的记录的noteID与感想，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送PUT请求，修改相应记录。在完成操作后，重新加载view。

4. 获取用户所选择删除的记录的noteID，使用api（http://139.196.36.97:8080/sbDemo/v2/note-management/notes）给数据库发送Delete请求，删除相应记录。在完成操作后，重新加载view。

3.4.6 流程图



## 3.5我的模块

3.5.1 程序描述

用户查看数据统计信息，点击昵称进行修改昵称。

3.5.2 功能

1. 获取用户资料
2. 查看数据统计
3. 修改昵称

3.5.3 输入项

1. 当前用户信息userID
2. 当前用户信息userID
3. 当前用户信息userID和新昵称name

3.5.4 输出项

1.用户的头像和昵称信息

2.数据统计信息，包括总阅读量，已读量，阅读天数和书摘条数

2.昵称更新后的我的页面

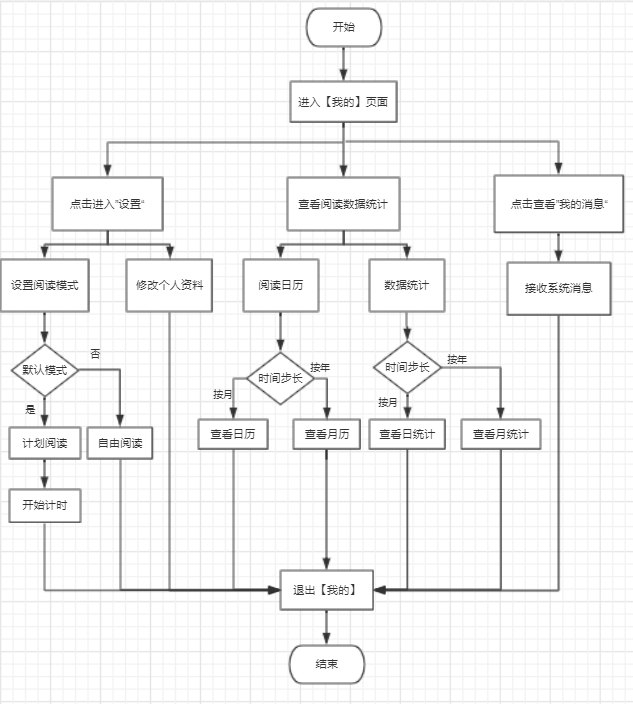
3.5.5 算法

1.获取当前用户的userID，使用api（http://139.196.36.97:8080/sbDemo/v2/user-management/ users）调取在数据库中的用户信息。将api返回的数据赋值给用户相应的view。

2.获取当前用户的userID，使用api（http://139.196.36.97:8080/sbDemo/v2/statistic-management/statistics）调取在数据库中该用户的数据统计信息。将api返回的数据赋值给数据统计相应的view。

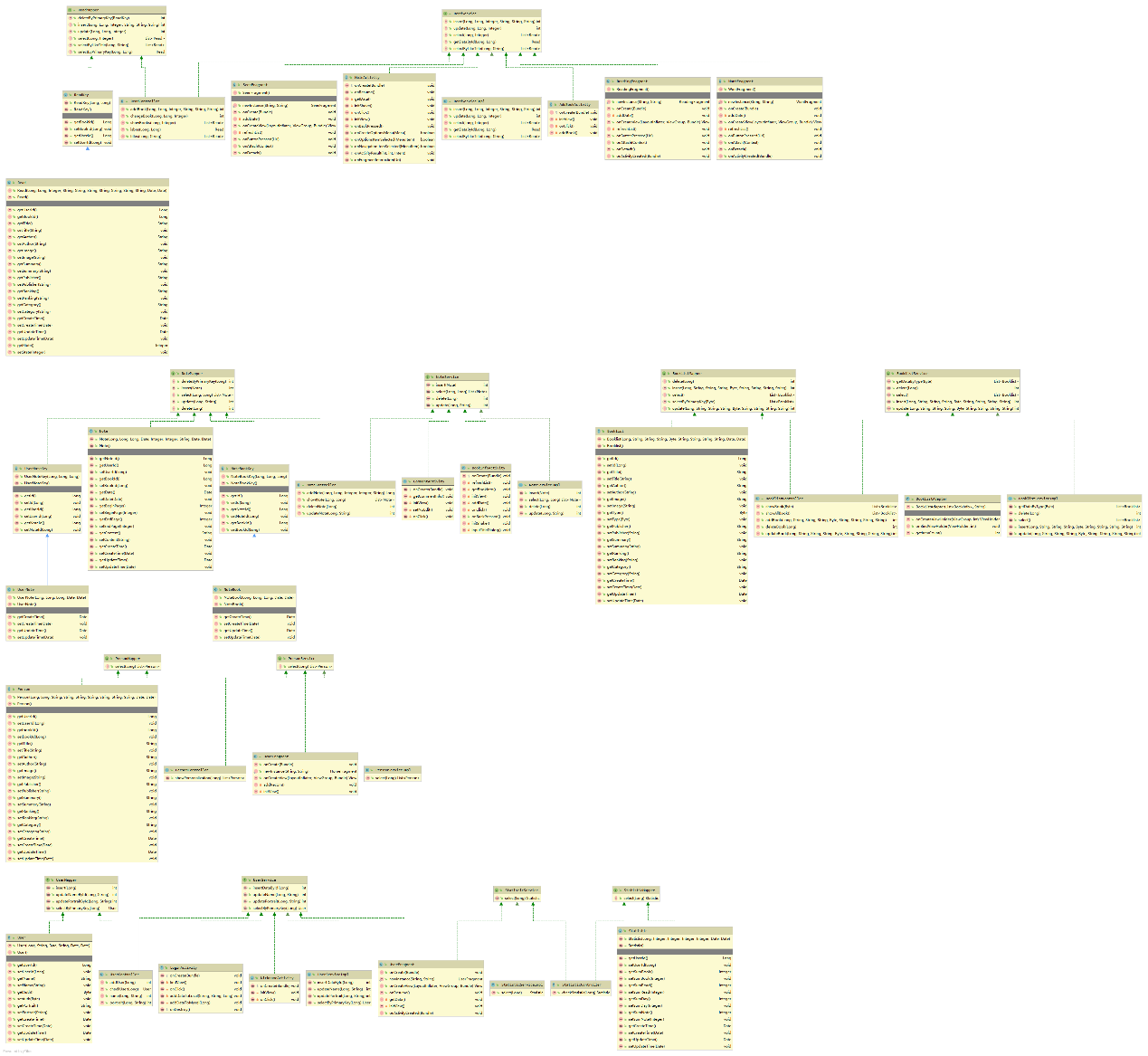
3. 获取当前用户的userID与用户新修改的昵称，使用api（http://139.196.36.97:8080/sbDemo/v2/user-management/ names）给数据库发送Put请求修改用户昵称。在完成操作后，重新加载view。

3.5.6 流程图

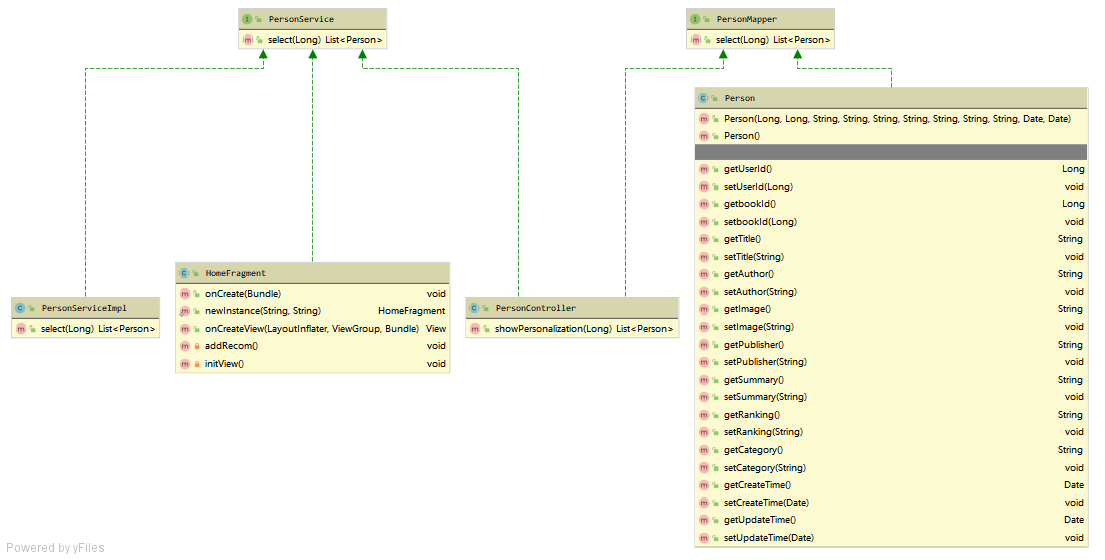


# 4.类图

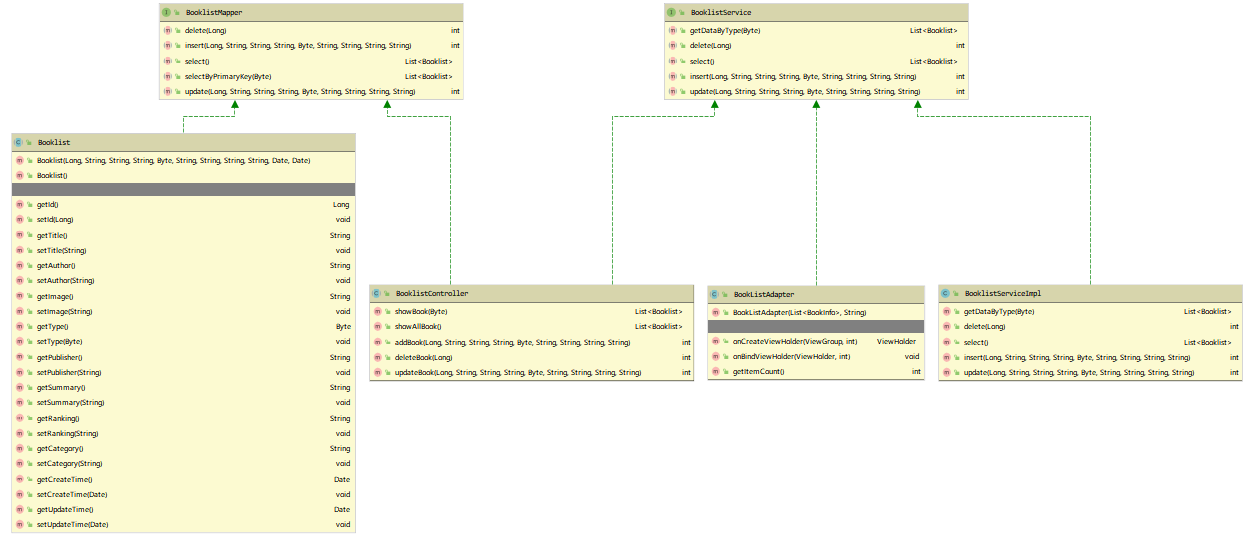
## 4.1总设计



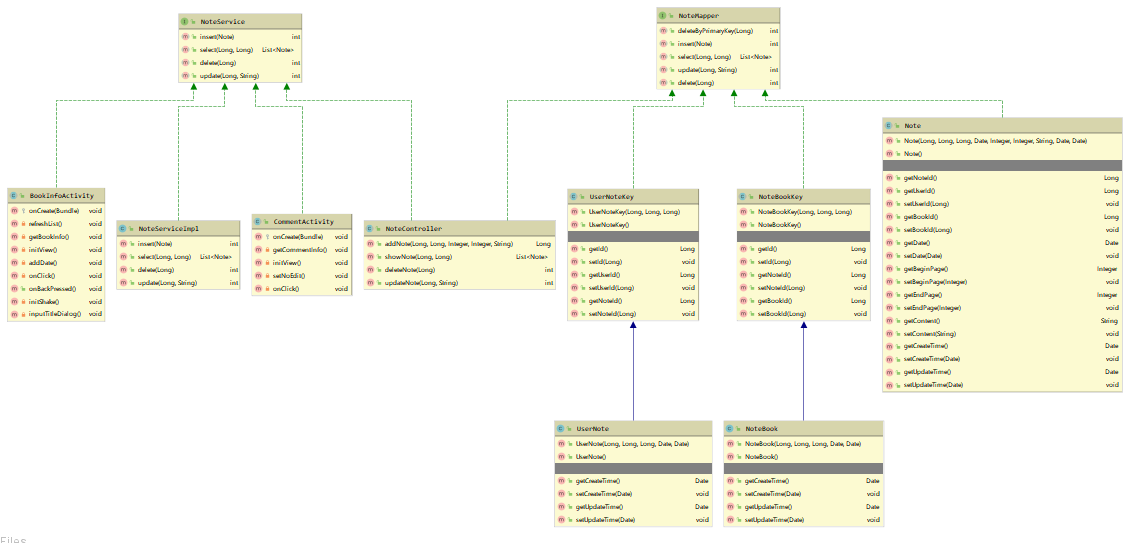
## 4.2子模块1—个性化推荐



## 4.3子模块2—系统推荐书籍



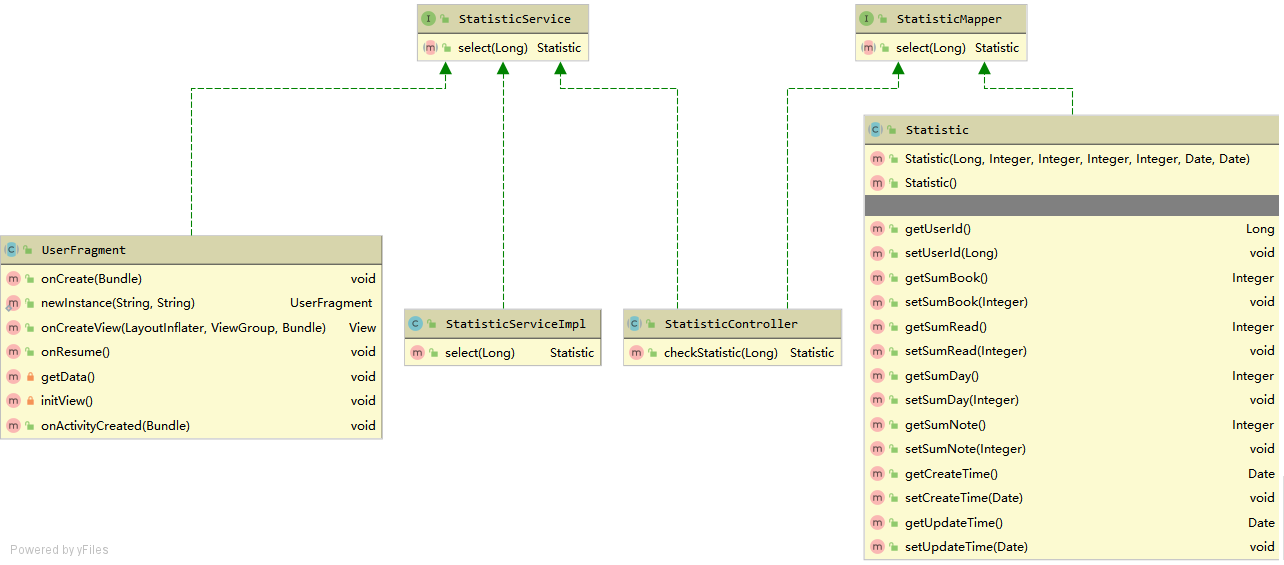
## 4.4子模块3—书摘记录管理



## 4.5子模块4—阅读进度管理



## 4.6子模块5—阅读数据统计



## 4.7子模块6—用户信息管理

