毕业设计说明书(论文)

作 者：

学 号：

院 系：

专 业： 计算机科学与技术

题 目： 基于Android的背单词系统的设计与开发 　　　　

**毕业设计说明书（论文）中文摘要**

|  |
| --- |
| Android是一种广泛流行的操作系统，在Android应用市场中每天都有上万个应用程序安装包投入使用。Android的问世极大地满足了人们追求方便的需要，不仅推动了当代智能手机多样性和多功能性发展，还将生活中的各种办公功能、娱乐功能和实用功能等结合起来，因此广受人们的喜爱。本课题的研究旨在通过词库的解析与导入，统计个人背诵信息以简化单词背诵流程并优化单词背诵方式，节约时间成本，追求学习工具便携化和单词学习效率最大化。  本应用是基于Android操作系统开发一个单词背诵手机客户端软件，能够结合不同用户的遗忘数据并反映为相应的遗忘曲线，从而达到更加有效地帮助用户记忆单词的目的。本应用具备学习任务量管理、学习内容设置、学习情况查询、遗忘曲线计算和统计以及单词查询（通过API实现）等功能。系统采用SQLite数据库来存用户信息，词汇本信息，词汇信息和用户学习数据信息等。  本文重点介绍了背单词应用的功能实现方法以及关键技术。  **关键字** 背单词应用 遗忘曲线 Android SQLite API |

**毕业设计说明书（论文）外文摘要**

|  |
| --- |
| **Title**  Design and development of word memorizing system  based on Android  **Abstract**  Android is a popular operating system. In the Android application market, tens of thousands of application packages are put into use every day. The advent of Android greatly meets the needs of people's pursuit of convenience, not only promotes the diversity and multi-functional development of contemporary smart phones, but also combines various office functions, entertainment functions and practical functions in life, so it is widely loved by people. The purpose of this research is to simplify the process of word recitation and optimize the way of word recitation by analyzing and importing the personal recitation information, saving time and cost, and pursuing the portable learning tools and maximizing the efficiency of word learning.  This application is based on Android operating system to develop a word recitation mobile phone client software, which can combine the forgetting data of different users and reflect it as the corresponding forgetting curve, so as to achieve the purpose of more effectively helping users memorize words. This application has the functions of learning task management, learning content setting, learning situation query, forgetting curve calculation and statistics, and word query (realized through API). The system uses SQLite database to store user information, vocabulary information, vocabulary information and user learning data information.  This paper focuses on the function realization method and key technology of memorizing word applicationThis paper focuses on the function realization method and key technology of memorizing word application.  **Keywords** Application of memorizing words, Forgetting curve, SQLite, Android, API |

目　　录

[第一章 绪论 1](#_Toc73311307)

[1.1 选题设计开发的应用背景及价值 1](#_Toc73311308)

[1.2 选题的研究现状 1](#_Toc73311309)

[1.3 关于本课题 2](#_Toc73311310)

[第二章 开发环境与主要技术 3](#_Toc73311311)

[2.1 开发平台 3](#_Toc73311312)

[2.2 主要技术 4](#_Toc73311313)

[2.3 术语及相关缩写解释 7](#_Toc73311314)

[第三章 系统分析 8](#_Toc73311315)

[3.1 系统构建需求分析 8](#_Toc73311316)

[3.2 系统构建目标分析 10](#_Toc73311317)

[3.3 系统构建功能分析 10](#_Toc73311318)

[第四章 系统设计 14](#_Toc73311319)

[4.1 前台背单词展示子系统详细设计 14](#_Toc73311320)

[4.2 后台数据管理子系统详细设计 22](#_Toc73311321)

[4.3 数据库设计 26](#_Toc73311322)

[第五章 应用编码、跨设备安装及测试 29](#_Toc73311323)

[5.1 文件结构 29](#_Toc73311324)

[5.2 典型编码 30](#_Toc73311325)

[5.3 应用跨设备安装 35](#_Toc73311326)

[5.4 应用测试及总结 35](#_Toc73311327)

[第六章 结束语 37](#_Toc73311328)

[6.1 毕业设计的难点与创新 37](#_Toc73311329)

[6.2 毕业设计的收获 37](#_Toc73311330)

[致   谢 38](#_Toc73311331)

[参考文献 39](#_Toc73311332)

# 第一章 绪论

本章主要针对选题的应用背景及价值、实际研究现状进行简单介绍。

## 1.1 选题设计开发的应用背景及价值

Android作为一种流行的移动操作系统越来越得到广泛应用，加上当前网络技术和数据库技术的广泛应用，基于Android的单词记忆APP也由此成为很多学生的学习工具。

随着越来越多的背单词软件进入市场，其智能化与互动性也越来越让其它背单词工具难以企及[1]。但是早期的单词背诵软件由于媒介主要是个人电脑，较差的便携性成为其缺陷，这也是早期背单词软件没有发展起来的主要原因。 而Android平台的迅速发展使得很多电商企业慢慢进入了一个开放手机联盟（OHA）时代[2]，由于Android系统的扩展性和开源性为众多Android开发者们提供了系统性平台,不但有利于集合众多种类的Android应用而且也让终端厂商可以根据自身的特殊需求“定制化开发”。这些无疑让Android手机成为当代移动互联网时代下催生的人们生活中必不可少的一个智能化产品，也完美地弥补了背单词软件便携式缺陷。

而且随着5G/WIFI技术的发展以及各种基于智能手机的应用和服务的增长，[3]用户不仅可以联网与好友进行实时交流，相互学习，共同进步，而且可以随时随地在没有网络的状态下自主学习，还可以在软件结合艾宾浩斯遗忘曲线规律[4]以帮助用户合理规划单词复习时间和复习内容。经过大量调查和实践验证，结合背单词应用程序记忆单词可以大大节省时间和提高记忆效率。

## 1.2 选题的研究现状

Android号称是首个为移动终端打造的真正开放和完整的移动操作系统，同时也是全世界占有率最大的移动操作系统。Android在SDK层面选择了Java作为开发语言。[5] Java语言是世界上最火的编程语言之一,而且Java是一款能够跨平台运行的语言，能够屏蔽掉底层硬件的差异，实现对于行发者来说是一个统一的平台。

## 1.3 关于本课题

本课题要实现一个集用户注册和登录、单词复习、单词本选择及查询、用户数据统计和软件设置等功能为一体的单词背诵应用。

本次设计主要是对大学时间所学的Android技术开发 ，数据库等课程的一次巩固复习及应用。本设计根据用户确定的单词本以及在设置中自定义的学习量，要求系统有计划地每天帮用户安排单词复习；用户可以自由选择更换单词本以及重置学习记录；可以通过各大翻译平台提供的API查找单词例句，可以了解自己的学习记录和记忆情况，可以自定义系统主题；

# 第二章 开发环境与主要技术

本章主要对以AndroidStudio为核心的IDE开发环境进行介绍，并对该课题开发过程中需要的集成API文档使用技术、SQLite应用技术、Json解析技术、Android开发技术进行详细介绍。

## 2.1 开发平台

本课题的开发平台为：

操作系统：Microsoft Windows 10

程序语言：Java，XML

IDE工具：AndroidStudio 4.1.2

IDE插件：Database browser

数据库服务器：SQLite

其他工具：Android 7.0以上系统的智能手机或者模拟器

关于本课题的开发环境安装配置过程比较简单，在此不再赘述。

### 2.1.1 AndroidStudio IDE

AndroidStudio是一个专门为Android平台开发程序的集成开发环境，其设计基于IntelliJ IDEA，可以运行在Linux、OSX和Windows等各大主流平台上。

AndroidStudio 会自动应用代码样式，用户可以在设置中指定样式和格式并且可以通过编程语言自定义代码样式进行设置。不仅如此，AndroidStudio还通过 [Android Plugin for Gradle](https://developer.android.com/studio/releases/gradle-plugin) 向用户提供更多面向 Android 的功能。该构建系统可以作为集成工具从 AndroidStudio 菜单运行，也可从命令行独立运行。可以说AndroidStudio是当前安卓开发市场上最受欢迎的、完全免费并且源代码开放的集成开发环境。

### 2.1.2 SQLite数据库管理系统

SQLite是一款轻型高效的并且遵守ACID的关系型数据库管理系统，其包含在一个相对小的C库中。SQLite的设计目标是嵌入式的，占用资源非常的低而且已经在Android系统的产品中投入使用。此外，它还支持Windows/Linux/Unix等等主流的操作系统，同时能够跟很多程序语言相结合，例如Java、C#、PHP、Tcl等，还提供了ODBC接口。SQLite第一个Alpha版本公布于2000年，至今SQLite已经发布到SQLite3版本。

与客户-服务器式的范例不同，SQLite[引擎](https://baike.baidu.com/item/%E5%BC%95%E6%93%8E)不是[程序](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F)与之通信的独立进程，而是连接并且成为程序中的一个重要要部分。[6]整个数据库文件都在宿主主机上存储在一个单一的文件中。SQLite在开启一个事务时会锁定一整个数据文件，从而由SQLite所建立的整个数据库都可以在宿主机上的对应的文件中被找到，这样使得对该数据库的分析和使用就变得十分方便。

### 2.1.3 Database Browser插件

DatabaseBrowser是一款连接SQLite数据库文件的开源（GPL许可）插件。该插件支持CLOB和BLOB，能够直接打开sqlite3数据库文件并且可以在AndroidStudio中直接使用。其作用主要是为方便开发者查看SQLite数据库的内容。DatabaseBrowser尽管提供了多种Filter和一个SQL窗口，但是开发者不必编写SQL语句来查看数据。实际上该插件虽然查看数据库文件内容很方便，但是并不能直接对Android设备上的数据库进行其他操作。

## 2.2 主要技术

### 2.2.1 Android开发技术

Android主要由谷歌与[开放手机联盟](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%89%8B%E6%9C%BA%E8%81%94%E7%9B%9F)共同开发，目前开放手机联盟是由包括高通、[T-Mobile](https://baike.baidu.com/item/T-Mobile)和[中国移动](https://baike.baidu.com/item/%E4%B8%AD%E5%9B%BD%E7%A7%BB%E5%8A%A8/237216)在内的多达三十余家无线和技术应用领衔企业组成[7]。借助日趋完善的开放式、标准化的移动应用软件平台，Android在移动产业内已经形成一个开放式的生态系统，并向着更快、更好的创新目标不断推进。未来Android将持续为移动用户提供更加人性化、智能化、全面化的应用和服务。第一部Android智能手机发布于2008年10月，如今Android已经扩展到平板电脑及其他领域上，如[数码相机](https://baike.baidu.com/item/%E6%95%B0%E7%A0%81%E7%9B%B8%E6%9C%BA/277472)、[电视](https://baike.baidu.com/item/%E7%94%B5%E8%A7%86/228945)、汽车、[智能手表](https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E6%89%8B%E8%A1%A8/71070)等等。

Android的开发语言是[Java](https://baike.baidu.com/item/Java/85979)，因此也继承了Java跨平台的特点。借助JVM的应用，使得Java技术成为第一个真正独立于平台的语言[8]，基于Java语言开发的应用软件可以真正实现“一次设计，到处运行”， 即任何计算机结构和操作系统只要可以运行Java虚拟机，Java程序就可不需改变地运行于其上。如今除了手机、平板等传统Android移动设备，汽车，智能手表，电视和其他各种智能家居也可使用 Android 系统，未来的Android必将成为万物互联化与智能化的重要桥梁。

**2.2.2.1 什么是Android**

Android是一种基于[Linux](https://baike.baidu.com/item/Linux)内核的操作系统，主要用于各种[移动设备](https://baike.baidu.com/item/%E7%A7%BB%E5%8A%A8%E8%AE%BE%E5%A4%87/9157757)。如[智能手机](https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E6%89%8B%E6%9C%BA/94396)和[平板电脑](https://baike.baidu.com/item/%E5%B9%B3%E6%9D%BF%E7%94%B5%E8%84%91/1348389)。Android系统包括操作系统、Android应用程序和用户界面，而且不存在任何以往阻碍移动产业创新的专有权障碍。[9]Android应用程序有四大组件，分别是活动（Activity），服务（Service），广播（Broadcast）和内容提供器（ContentProvider）；由四大组件构成整个应用程序的基本运行逻辑。

Android四大组件：

* + 活动，活动与用户直接相关，是Android应用程序和用户进行交互的窗口。活动要保持各界面的状态，包括生命周期的良好管理和一些逻辑跳转[10]，活动有四种启动模式，分别是standard，singleTask，singleTop和singleInstance。每个活动都是相对独立的，各个活动之间采取进栈出栈的形式呈现给用户。
  + 服务，服务是在Android中实现后台运行程序的解决方案。服务在后台为活动服务并通过活动的指令执行一些封装好的完整的逻辑功能，然后完成相关的事务处理。
  + 内容提供器，Android提供的用来访问第三方应用数据的方案，同时还能保证被访问数据的安全性[11]。数据提供端所在的应用程序可以派生出一个ContentProvider类，该类通过匹配uri从而使访问者安全地对所提供的数据进行CURD操作。而且内容提供器是一个统一接口模型，访问者不必知道其内部具体的实现细节，这极大地简化了上层应用。
  + 广播，广播由Android系统或者应用程序发出。每个Android应用程序都可以对这些广播进行针对性的注册和监听。在接收到广播后，Android应用程序可以通过继承BroadcastReceiver类实现相应的简单逻辑，但是在其中不允许开启子线程。广播通常被用来开启一个服务和活动。

**2.2.2.2 Android多媒体层**

Android系统提供了丰富的多媒体服务，例如如视频，音乐，录音，录像和拍照等操作；本次应用程序设计将会利用Android多媒体中的音频与API结合，使应用功能更加丰富；Android使用音频设置数据源的方式有三种，分别是播放应用程序中预备的音频文件、设置播放网络音频文件和设置播放SD卡中的音频文件。

**2.2.2.3 系统控件层**

Android系统为开发者们提供了丰富的控件，开发者可以很轻松地利用这些控件编写出漂亮的界面；在Android中，View包含了控件显示和用户交互，例如Button，TextView，ImageView和ViewGroup等。其中ViewGroup也具有View的特性，但它的主要作用是充当View的容器，例如FrameLayout，LinearLayout和RelativeLayout等。将其中的View视作自己的子控件，对它的子View进行布局管理，当然其子控件也可以是ViewGroup类型。

### 2.2.2 XML和JSON解析技术

XML是一种为了使文件具有结构性的[标记语言](https://baike.baidu.com/item/%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/5964436)。XML可以被用来定义数据类型、标记数据。本应用使用SAX解析XML数据；SAX不会保存任何数据，SAX在每读取一行XML文档数据后就会发给用户一个通知，用户可以在解析过程中利用通知来实现自己的逻辑 [12]。

JSON是一种轻量级的数据交换格式，是存储和交换文本信息的语法。JSON采用完全独立于语言的文本格式，保留了C语言家族的习惯而且语法简洁，易于开发者读写。JSON 类似 XML而且比 XML 更快， 更易解析[13]。本应用使用GSON解析JSON数据； Gson是一个Java库，能够将Java对象转换为其JSON表示，还可以将JSON字符串转换为相同的Java数据类型。Gson可以处理所有的Java对象，当然也包括没有源代码的预先存在的对象。[14]

### 2.2.3集成API文档使用技术

API是一些被预先定义的接口或[软件系统](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E7%B3%BB%E7%BB%9F/224122)中不同组成部分衔接的约定。API提供[应用程序](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F)与开发者们基于某[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6)或硬件得以访问的一组[例程](https://baike.baidu.com/item/%E4%BE%8B%E7%A8%8B/2390628)。开发者们使用API时只需要理解调用方法并直接调用即可，并不需要关心或者是理解其内部具体的[工作机制](https://baike.baidu.com/item/%E5%B7%A5%E4%BD%9C%E6%9C%BA%E5%88%B6/9905789)。

本次项目涉及了HelloChart绘图API，爱词霸英汉字典API，幽灵数据背单词API，返回的数据格式均为各种JSON数据，配合JSON解析技术完成各种数据的使用。另外SQLite封装好的CURD操作API，也让本次项目中存储和读取数据变得非常方便。

### 2.2.4 ADB shell数据库连接技术

**ADB**是一个命令行窗口，用于通过电脑端与模拟器或者真实设备交互。虽然Database Browser查询sqlite内嵌数据库内容非常方便，但是无法使用SQL语句对应用程序中所使用到的数据库进行CURD操作。而在开发过程中对数据库的CURD操作是十分必要的，此ADB shell就变得十分方便；ADB shell可以直接用于AndroidStudio中自带的Terminal中，且可以使用任何SQLite所支持的全部SQL语句进行相应的操作。

## 2.3 术语及相关缩写解释

* + IDE：全称Integrated Development Environment，集成开发环境；
  + API：全称Application Programming Interface，应用程序接口；
  + CURD：全称Create、Update、Retrieve和Delete，本文中专指在关系型数据库系统中对其数据所进行的创建，更新，检索和删除操作；
  + SQL：全称 Structured Query Language，结构化查询语言，它是一种数据库查询和程序设计语言，用于存取数据以及查询和管理关系数据库系统[15]。用户在使用结构化查询语言时不需要具体了解数据存放的实现方式。如果使用相同的结构化查询语言，还可以作为数据输入与数据管理的接口[16]。
  + XML：全称Extensible Markup Language,即可扩展标记语言；
  + JSON：全称JavaScript Object Notation，JavaScript对象标记；
  + ADB：全称Android Debug Bridge，安卓调试桥，通过ADB就可以在Android Studio中调试Android程序。

# 第三章 系统分析

本章主要对系统构建进行较为详细的需求分析、目标分析和功能分析。

## 3.1 系统构建需求分析

### 3.1.1 国外安卓应用发展的现状

据调查，2019年美国的互联网用户规模已不低于2.9亿人，互联网使用率达到89.4%，每日手机使用的平均时长超过三个小时。在免费应用中比较热门类型为社交类，游戏类，办公类及网络视频类；在付费应用中比较热门类型为工具类及网络游戏类。在Android平台，GooglePlay中拥有庞大的应用程序；其数量和下载量均远超国内任意一家应用商店，由此可见国外使用移动应用程序的大潮早已来临,刺激了更多移动应用程序的开发需求。[17]不同于国内多数开发者在应用程序开发语言的选择上仍然使用Java作为开发语言，国外多数开发者普遍选用Kotlin作为开发语言。[18]

### 3.1.2 国内安卓应用发展的现状

国内的 Android开发还是主要以应用开发为主, 主要分成三类: 为企业开发应用、游戏开发和开发通用应用。

* 企业开发应用，此类应用一般被大规模公司所需求,这些公司主要为自有品牌或者其他品牌设计手机或者平板电脑的总体方案。
* 开发通用应用，这类应用的盈利方式主要有两种:为国外公司进行外包开发,通过内嵌广告获取广告分成以及通过付费购买的形式来盈利 [19]。
* 游戏开发，目前和第二类开发者类似。此类应用通过正确利用大多数用户的各种心理，不仅能使用户对保持应用的新鲜感而且更容易使通过付费购买的方式来盈利。

### 3.1.3 背单词App面临困境

国内背单词App的目的主要有：利用应用内嵌广告盈利，诱使用户每月开通各种会员赚钱，要求用户对单词量的直接购买，紧跟安卓应用发展潮流等，其中，前三种占大多数。然而，实际情况是即使是完全免费的背单词App的用户量和优评率都未达到预期效果，有的根本没有什么用户量，有的虽然用户量众多，但是用户使用期限很短，差评率很高，同时安卓App应用商店对其所列出的优点也并未被广大用户认同。于是，这些背单词App不得不面临生存危机，究其原因，主要有：

（1）背单词App内嵌广告出现过于频繁：即使是再优质的应用，用户也会对这些不必出现的广告表现出极大的反感。有些App甚至在用户使用期间一直弹出各种广告，并且用户一旦误触即会提示在后台下载，这将导致该应用的用户量迅速流失以及差评率飙升。

（2）背单词App功能过于机械，不满足用户学习需求；很多应用只是简单地分配罗列每天的单词量以供用户复习，却并没有根据用户遗忘程度安排复习特定的单词，也没有避免背诵很多简单的，已学会的单词。这导致用户做了很多无用功且遗忘的单词没有得到及时的复习，使用户的学习效率很差。

（3）背单词App词汇量固定，词汇内容无法修改；很多应用只提供词汇收藏却不提供词汇内容修改功能，这造成用户在学习熟词偏义时无法自定义对应词汇内容，对学习造成阻碍。

### 3.1.4 背单词App的生存环境

* 背单词App的用户量不断增大：

在中国，有很多电商企业都步入了一个开放手机联盟（OHA）， 由于Android系统高度的开源性和扩展性，为众多移动应用开发者提供了良好的系统性平台, 也无疑让Android手机成为当代移动互联网时代下催生的人们生活中不可或缺的一个智能产品，也完美地弥补了背单词软件便携式缺陷。因此在中国基于Android使用背单词App的用户数量增长趋势早已是势如破竹；

* 背单词App的应用质量不断提高：

由于安卓应用市场庞大的应用量和用户量，整个应用市场一片繁荣。各种用户要求和软件市场的竞争使得各种开发者和开发团队对应用质量的要求越来越高。而且目前政府和谷歌公司对于国内应用市场的干预程度越来越高。政府出台更多的是安卓应用市场的调控，资质、隐私、控制相关，谷歌管控更多的是安全、规范、性能相关。在此大环境下，相信背单词App的应用质量一定会更加精益求精。

## 3.2 系统构建目标分析

背单词App的开发，需要用户能够随时地操作用户数据以及对每日任务和各单词出现的词频率自动做出调整，以满足用户高效率背诵单词的需要；因而在安排用户复习时对应单词本中的词频计算尤为重要，本系统目标是按艾宾浩斯遗忘曲线和用户熟识度相结合安排复习的方法来实现用户单词背诵效率最大化[20]。此外，应用还需要实时给出直观的用户熟识度统计和完成情况图：

1. 实现熟识度计算词频

对各个单词的熟识度应由用户在复习单词时来自主选择；熟识度包括熟记，认识，模糊和忘记四个选项，每个熟识度将由系统判断在保证今日任务的情况下再次安排复习。保证用户在当天背诵记忆能够更加巩固，牢靠。

1. 实现艾宾浩斯遗忘曲线计算词频

本应用充分利用艾宾浩斯遗忘曲线，使用户对非今日任务做一个合理的复习；艾宾浩斯遗忘曲线计算方法只依赖与系统时间戳；被由于艾宾浩斯遗忘曲线而安排用户复习的单词将刷新该单词的遗忘率（这与用户熟识度无关）；从而保证用户对过去已学单词的背诵记忆够更加巩固，牢靠。

1. 实现学习情况统计

通过数据库统计数据并且利用API以图的形式直观地展现给用户。统计图包括当前词本学习进度统计图，艾宾浩斯遗忘曲线图和学习情况统计图；其中学习情况统计图提供用户明确的用户熟识度情况，剩余计划和单词收藏情况。

1. 实现用户数据操作

通过用户界面设置达到用户自定义每日任务量，用户学习任务管理和系统主题等目的，进一步提高用户的使用体验和学习效率。

## 3.3 系统构建功能分析

### 3.3.1 系统的总体架构

该系统由两个子系统构成：前台展示背单词子系统和后台数据管理子系统。同时主要包含两种Json数据解析和一种XML数据解析过程：幽灵背单词API词汇本和词汇数据解析，爱词霸词典API数据解析。其中幽灵背单词API数据解析是关键部分，是全部相关词汇的数据来源。

### 3.3.2 系统功能模块

**3.3.2.1前台展示背单词子系统**

前台展示背单词子系统如图3-1所示：

复习模块

统计模块

词本模块

前台展示背单词子系统

设置模块

导航栏模块

图3-1前台展示背单词子系统图

* 前台展示背单词子系统简述

1. 复习模块：背单词App的主要功能，安排用户背诵相应单词，更新用户信息和词汇信息数据库；
2. 统计模块：显示用户学习进度扇形统计图，用户学习情况柱状统计图和艾宾浩斯遗忘数据折线统计图；
3. 词本模块：变更当前所学习的英语词本，查看收藏词汇和搜索单词；
4. 设置模块：设置背单词App的主题，每日背诵的单词任务量和操作用户数据；
5. 导航栏模块：帮助用户快速切换到对应的功能模块，展示模块切换风格；

**3.3.2.2后台数据管理子系统**

后台数据管理子系统如图3-2所示：

CURD用户数据表

CURD词汇本数据表

后台数据管理子系统

词汇例句初始化

CURD词汇数据表

图3-2后台数据管理子系统

* 后台数据管理子系统简述

1．用户数据初始化：初始化当前用户信息数据库，包括用户ID，用户所用词本ID，用户任务量，用户今日任务进度，时间戳等信息；

2. 词汇本数据初始化：初始化全部词汇本信息数据库，包括词汇本数据请求信息，词汇本ID，词汇本词汇数量，是否存有词汇数据等信息；

3. 词汇数据初始化：初始化全部词汇信息数据库，包括词汇ID，词汇数据请求信息，词汇本ID，词汇音标，词汇发音地址，词汇名，词汇释义，词汇是否被收藏，词汇时间戳，词汇记忆程度，词汇熟识度和词汇例句等信息；

4. 词汇例句初始化：借助爱词霸API查询对应词汇的单词例句并更新词汇信息数据库；

### 3.3.3 应用功能规划图

该系统的主要功能规划如图3-3所示，这里展现的只是系统的主要流程的大致环节，详细说明会在后面功能模块的详细设计中涉及。

JSON解析

CURD词汇数据表

前台背单词展示子系统

后台数据管理子系统

学习情况图示

设置每日任务量

设置用户学习数据

设置

设置应用主题

词汇搜索

查看收藏单词

词本

更换用户所用词本

用户熟识度选择

词汇收藏

复习

词汇释义发音

下个单词背诵

Internet访问幽灵数据API，爱词霸词典API

CURD词汇本数据表

XML解析

CURD用户数据表

词汇例句数据初始化

图3-3 主要功能流程图

# 第四章 系统设计

本章内容包括两个子系统各功能模块的详细设计和数据库设计。

## 4.1 前台背单词展示子系统详细设计

前台背单词展示子系统与用户直接相关，所以该子系统在设计时除了需要关注功能和性能的实现，更需要额外花些精力去关注美工和用户的体验度。这将在很大程度上影响用户对背单词App的第一印象和评价，从而直接能和用户使用量体现出正相关的联系。根据分析，前台背单词展示子系统的页面中的导航栏模块，系统导航栏以及系统状态栏信息是每个页面都要有的，并且这部分的数据和位置都是固定的，因而可以采用Android中Activity的fragment，RadioButton和ViewPager的搭配技术来避免这部分内容的重复加载。此外由于可能需要大量从API往各种数据表中初始化数据且数据量很庞大，因此项目中使用分批访问数据和多线程对相关数据库使用CURD操作的方案，从而加速页面的显示和流畅度，极大地提高用户体验度。

前台背单词展示子系统上分为四大功能模块，分别是复习功能模块，统计功能模块，词本功能模块和设置功能模块。在实现时，每个功能模块对应一个fragment，本项目采用在一个主活动上通过ViewPager和RadioButton同时加载这四个fragment的方式，这样用户在使用过程中每个fragment就只需要初始化一次即可，并且开发者还可以通过ViewPager在主活动中定义fragment切换风格，在本项目中每个模块切换时就不会有画面突变现象，用户感觉就是一串连贯的页面，体验度更高。使用ViewPager和RadioButton后， 部分fragment间的跳转就不能通过各种生命周期的方法来更新fragment上的数据了。最终的解决方案是重写setUserVisibleHint(boolean isVisibleToUser)方法来更新碎片上的数据信息。

### 4.1.1 复习功能模块

复习功能模块是展示给用户的第一个fragment，其工作原理和用户界面是背单词App最关键的部分，对是否能激发用户继续使用本应用起着非常巨大的作用。因此，复习功能模块的设计需要重视用户的体验度和美工设计，同时还需要兼具使用方便，信息全面等特点。在复习功能模块中，fragment主体上从上到下依次分为三个框架，第一个框架显示该用户的今日进度和单词任务量信息；第二个框架是用户对当前单词的操作框架（例如单词收藏，单词发音和熟识度选择等）；第三个框架是单词释义，音标，例句等信息。

为了方便叙述，先从第二框架和第三框架开始介绍。

对用户来说，如果不进行任何操作，第二框架和第三框架的可见部分只有单词，单词音标，单词发音和单词收藏。单词的词性，释义，和例句部分以及单词的熟识度选择部分会被完全隐藏起来。本项目将单词例句和单词词性控件上设置点击事件并由提示性点击文字。用户执行相应操作后则会显示第二框架和第三框架的用户熟识度选择，单词词性和单词例句。其中用户熟识度一共有四种选择，分别是熟记，认识，模糊和忘记。单词操作以及熟识度选择的作用如下：

* 熟记：该用户词汇数据表记录对当前单词的熟记标志位，并有0%的可能性使该单词在不计今日任务量的情况下再次被安排复习。
* 认识：该用户词汇数据表记录对当前单词的认识标志位，并有20%的可能性使该单词在不计今日任务量的情况下再次被安排复习。
* 模糊：该用户词汇数据表记录对当前单词的模糊标志位，并有40%的可能性使该单词在不计今日任务量的情况下再次被安排复习。
* 忘记：该用户词汇数据表记录对当前单词的忘记标志位，并有60%的可能性使该单词在不计今日任务量的情况下再次被安排复习。
* 单词收藏：该用户词汇数据表记录对当前单词的收藏或取消收藏标志位，可在词本功能模块中的“我的收藏中”再次查看已收藏和取消收藏操作。
* 单词发音：根据词汇数据表中的地址播放真人发音；

在初始化数据时，应用程序将准备wordList和hitList。其中wordList中装有今日用户需要完成的所有单词而hitList装有所有被击中词汇（在这里将因熟识度而再次被安排复习的所有单词成为被击中词汇，下同）。用户必须在这四种熟识度中选择一个才能进行下一个单词的复习。在准备安排下一个单词复习时将由50%的可能性使该单词是来自hitList；

第一框架中的今日进度和每日任务量数据都来自于用户数据表，当用户在进行熟识度选择并且下一个单词是来自于wordList时，今日进度加一。今日进度的最大值不大于每日任务量否则提示任务已完成信息。当用户未完成昨日任务量时，今日进度会重置为1，但在安排复习时仍然从昨日未进行熟识度选择的那个单词开始计算。当用户在设置功能模块中重新设置任务量时分为以下几种情况：

* 用户新设定每日任务量大于原每日任务量：复习功能模块将会在wordList中追加新扩容的词汇并且根据新设定的值更新每日任务量的值；
* 用户新设定每日任务量小于原每日任务量：复习功能模块将会在wordList中按顺序从后往前删减多余词汇并且根据新设定的值更新每日任务量的值。如果新值已经小于今日进度则将今日进度重置为每日任务量最大值，并且提示今日任务已完成。但是明日依然从未进行熟识度选择的那个单词开始计算。

复习功能模块图例如图4-1，图4-2所示：

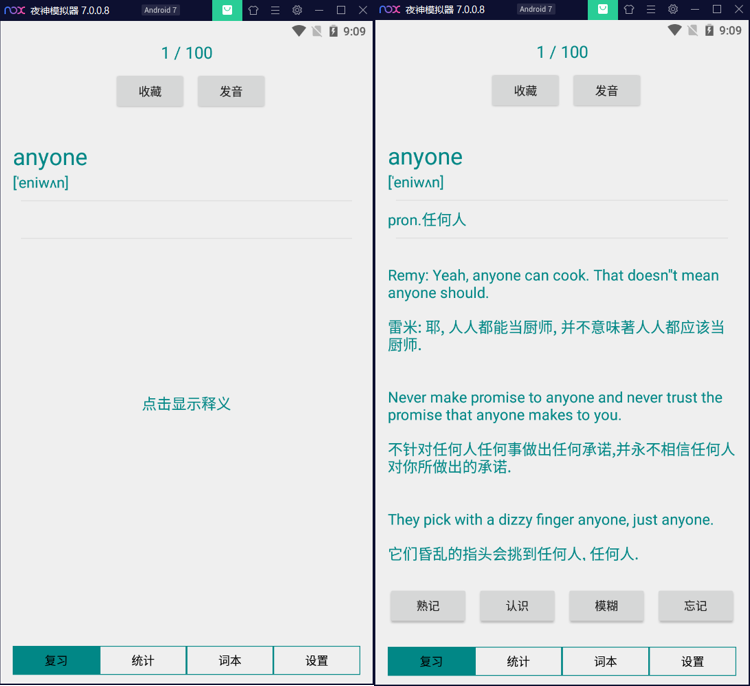


图4-1 复习功能模块演示（白日主题）

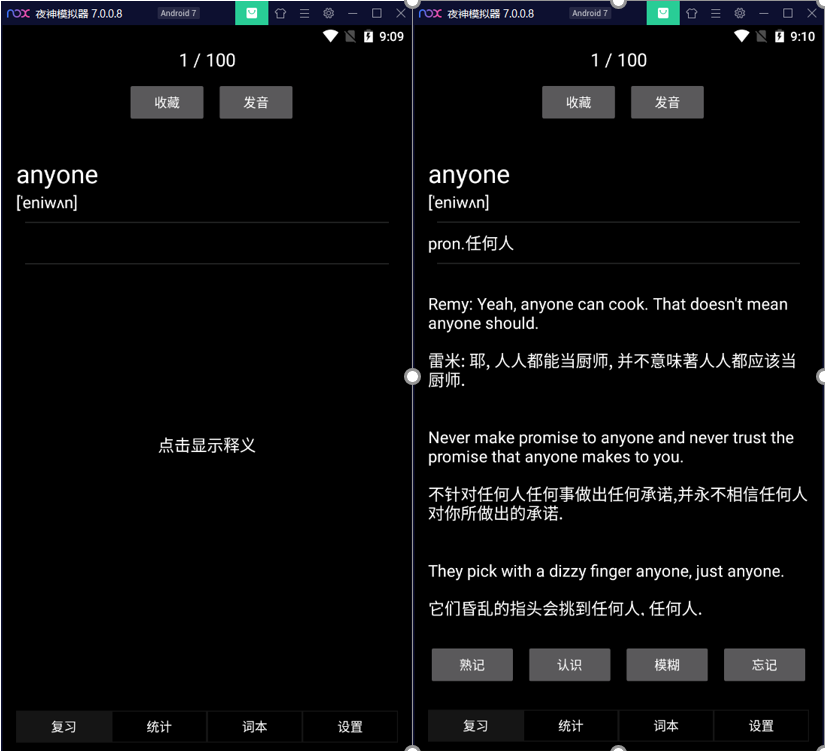


图4-2 复习功能模块演示（黑夜主题）

### 4.1.2 统计功能模块

统计功能模块是被设计用来直观地反映用户学习情况的fragment，其统计数据全部来源于数据库并且被实时更新。因此除一定的点击响应事件外（如扇形统计图可以随用户手势进行原地旋转），用户不可以直接对统计图进行编辑或修改操作。其运用了HelloChart图形绘制框架，可以描绘除各种漂亮的统计图。通过HelloChart，本模块包含了用户学习进度扇形统计图，艾宾浩斯遗忘数据折线图和用户学习情况柱形统计图：

* 学习进度扇形统计图：针对当前词汇本而做出的统计，包含剩余计划词汇量，今日未完成词汇量，已完成词汇量，当前单词本的简要信息和学习进度；
* 艾宾浩斯遗忘数据折线图：德国心理学家艾宾浩斯对遗忘现象做了系统的研究，他用无意义的音节作为记忆的材料，把实验数据绘制成一条曲线，称为艾宾浩斯遗忘曲线。曲线的纵坐标代表记忆保持量，表明了遗忘发展的一条规律：遗忘进程是不均衡的，在识记的最初遗忘很快，以后逐渐缓慢，到了相当的时间，几乎就不再遗忘了，也就是遗忘的发展是“先快后慢”。本项目中每一个单词都有与其对应的记忆值和复习时的时间戳，用来表示艾宾浩斯遗忘曲线的纵坐标。需要说明的是，纵坐标的值只与其对应的时间戳有关。
* 学习情况柱形统计图：针对的是用户所有已学的单词做出的统计。其纵轴是单词数量，横轴包含用户熟识度统计量，单词收藏量和今日剩余未完成量。

统计功能模块图例如图4-3所示：

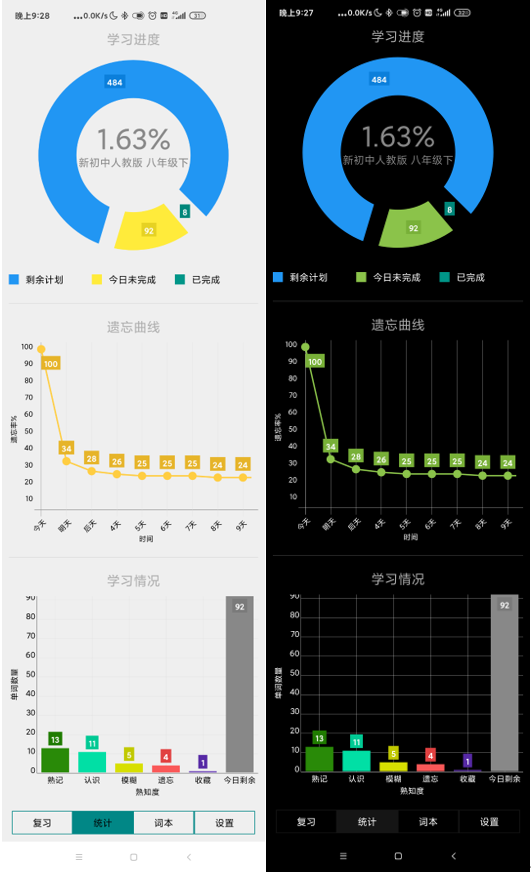


图4-3统计功能模块统计图（长截图）

### 4.1.3 词本功能模块

词本功能模块也是背诵单词App的核心模块之一，其主要功能是用来帮助用户更换当前所学词本，执行这个操作后再回到复习功能模块中会发现今日进度会被重置，但是用户每日任务量保持不变，在统计功能模块中针对当前所学词本的学习进度扇形统计图数据也发生改变。此外词本功能还有“我的收藏”选项以查看所有已收藏词汇并且用户可以任意取消收藏某个单词，还有搜索框以查询数据库中相关词汇和网络查询，这里只介绍最为主要的切换词本功能。词本中的所有数据是预加载的并直接通过AsyncTask同步到数据库和应用界面中，但是每个词本未必有单词数据，因为如果将全部数据预加载会出现API频繁访问错误。最终解决方案是用户必须要选择一个词汇本，再由AsyncTask将对应词汇本中的词汇数据同步到数据库和复习功能模块fragment中。当用户是初次使用本应用时，复习功能模块中所有控件都将不可用并且会给出选择词本提示，如图4-4所示；

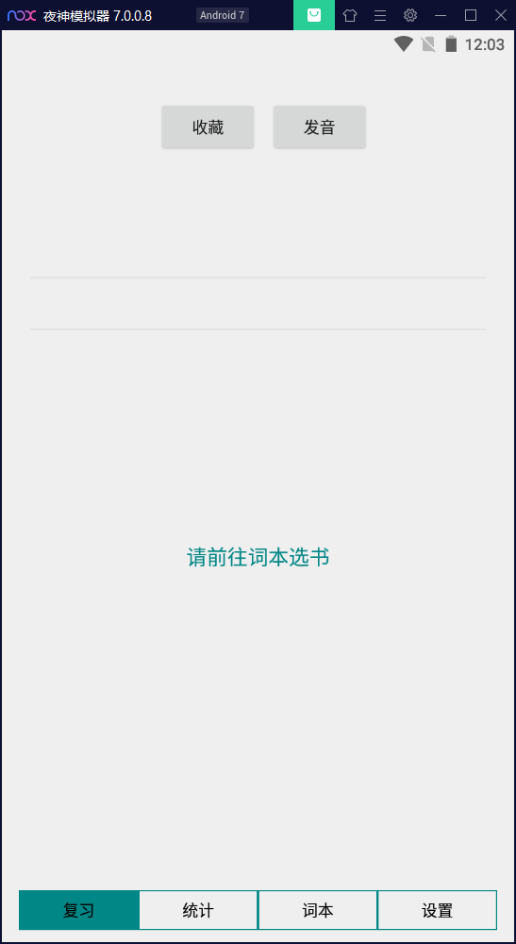


图4-4 未选词本时复习功能模块界面

实现用户选择词本操作也很简单，全部书本列是可以滑动的，并设有点击事件；因此用户只需要直接点击所列词汇本名称即可，此时该功能模块界面会有切换成功提示。当然，如上所述相应的数据也会自动添加并更新到数据库和各大功能模块中。词本功能模块界面如图4-5所示：



图4-5 词本功能模块界面

### 4.1.4 设置功能模块

设置功能模块的功能主要有用户每日任务量设置，用户数据重置和应用主题切换，此外还有用户切换和当前单词本更新功能；其作用分别是：

* 用户每日任务量设置：用户任务量设置由用户自定义，是复习功能模块中对用户词汇数据表操作的重要依据；其作用已经在复习功能模块中说明，故此处不在赘述。
* 用户数据重置：该操作将会重置用户数据表和用户词汇数据表中的所有数据，重置后用户将犹如新用户一般不再有任何学习记录，但是相关数据库中的内容不用重新下载和解析。相关的数据变动也会实时同步到统计数据模块中，因此查看统计功能模块内容也是判断用户数据是否已经重置的依据。
* 应用主题切换：有黑夜和白日两种主题可选，引入黑夜主题迎合了Android开发的发展潮流和越来越高的用户需求。应用主题切换也是用户数据表中的字段，可根据不同用户需求自动切换到相应主题。
* 用户切换：由于所有的词汇数据来源都来自幽灵数据API，而API提供的用户ID和用户密钥已经非常具有唯一性和私密性。故将用户账号和密码直接配置为API提供的用户ID和密钥。如在词汇本模块功能中所述，可以利用词汇本数据的预加载，初次使用时可在主活动使用加载fragment之前判断是否已有登陆状态，若已有登陆状态则直接加载各类fragment；若无则根据返回数据判断是否可登录成功。若成功则记入用户数据表并加载fragment否则予以拒绝登录，不加载fragment只停留在主活动的登录页面（图4-7）。故不再提供用户注册功能。
* 当前单词本更新：此操作将废除当前用户词汇本所对应的用户词汇数据表，然后重新申请API数据，重新初始化用户词汇数据表。

设置功能模块图例如图4-6，图4-7所示：

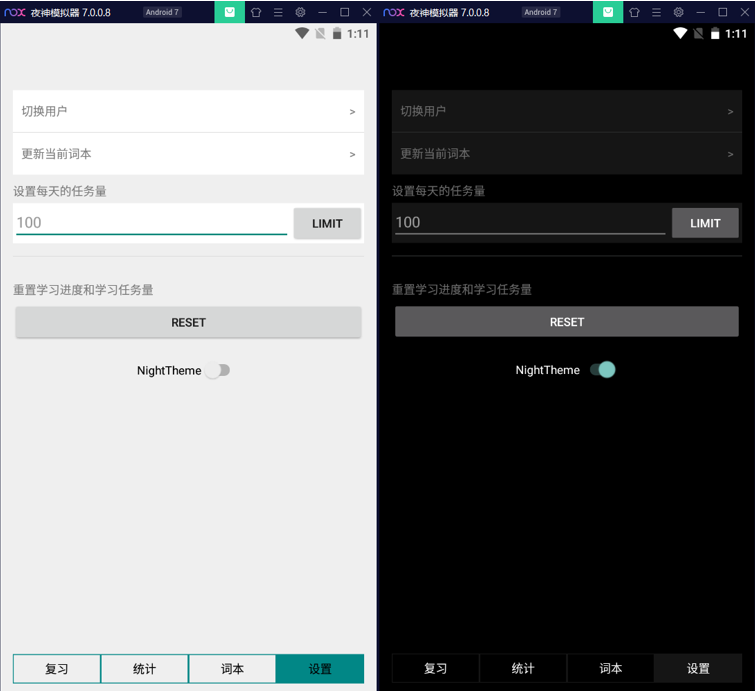


图4-6 设置功能模块界面

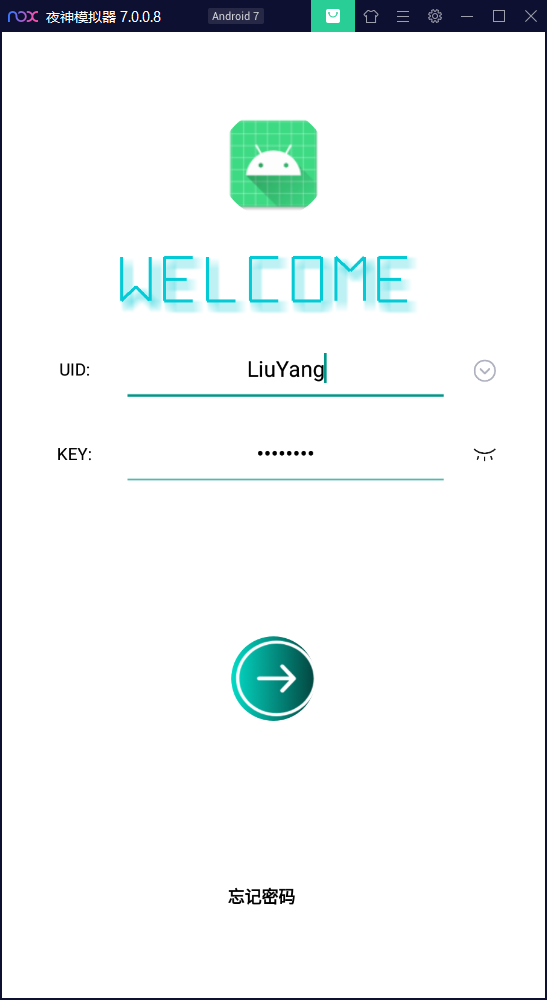


图4-7 用户登录界面

## 4.2 后台数据管理子系统详细设计

后台数据管理子系统的工作由系统根据用户操作和系统时间戳在子线程和服务中自行完成，因此用户在使用过程中不需要关心这些实现过程。

后台数据管理子系统的工作内容主要是对每个数据表进行CURD操作，此外还有从各个数据表中获取统计数据、使用并解析幽灵API和爱词霸API数据、为每个单词初始化例句以及利用艾宾浩斯遗忘曲线安排用户复习已学单词等等。其作用分别是：

* 对每个数据表进行CURD操作：包含用户数据表，用户词汇数据表，用户词汇本数据表和单词例句初始化；
* 各个数据表中获取统计数据：是统计功能模块的所有数据来源；
* 使用并解析幽灵API和爱词霸API数据：当判断出数据库缺少相应数据时将通过Internet访问API并进行解析其中幽灵API使用JSON解析，爱词霸API使用XML解析。
* 为每个单词初始化例句：由于API访问不能过于频繁，需要用户在复习到对应单词时才进行初始化判断；
* 艾宾浩斯遗忘曲线安排用户复习已学单词：按艾宾浩斯遗忘曲线进行概率复习。

### 4.2.1词汇本数据的使用

词汇本数据表中的数据是预加载的。当词汇本功能模块对应的fragment被建立时，系统会检查数据库中是否已含有对应数据。若有则直接加载数据库中的数据，否则自行向幽灵背单词API申请相关数据并更新用户界面，然后存入数据库中。

词汇本数据表字段包含数据请求结果，词汇本书名，词汇本id，词汇本单词数量和是否存在词汇数据；其中是否存在词汇数据的字段作用是标记该词汇本中是否已经含有词汇数据，0表示没有，1表示有，默认值是0。用户在选择相应的词汇本后将下载该词汇本中相应的数据，词汇返回结果成功后重置是否存在词汇数据的字段为1，更新用户数据表中书本id字段为当前书本id以及相应用户界面。

### 4.2.2击中函数

首先阐述本应用中概率事件发生方法的实现原理：设计一个返回值为布尔类型的方法，方法中传入的参数在[0,100]之间；通过Java中的Math.random()方法产生一个在[0,100]之间的一个随机数，若传入的参数小于该随机数则返回true表示事件发生，否则表示事件不发生。从而保证传入参数越大则事件发生的概率越大。

概率发生方法的实现过程虽然很简单却贯穿后台数据管理子系统的始终，是实现按照用户所选熟识度安排用户复习单词、按照艾宾浩斯遗忘曲线安排用户复习单词以及用户在复习单词时，判断下一个单词是否来自hitList的重要依据。本文将上述概率事件发生的方法定义为击中函数（下同）。

### 4.2.3词汇数据的使用

词汇数据表是所有数据表中最复杂的数据表。其不仅包含了英文单词，单词释义，单词发音地址，单词音标，单词含义和单词例句等基本属性字段，还包含了单词id，单词数据请求结果，单词所属词汇本id，单词所属词汇本名称，单词收藏，单词熟识度选择，时间戳和记忆值等重要属性字段。

**4.2.3.1词汇数据表**

在词汇数据表中，基本属性字段中的单词例句数据来自于爱词霸字典API，其余数据来自幽灵背单词API。在选词功能模块中，当用户在选择词本时，系统将判断所选词本是否有对应词汇数据。若有则直接加载，若则访问幽灵背单词API对应的数据再通过JSON解析并被加载到词汇数据表中。在复习功能模块中，当用户完成熟识度选择后，系统将判断所下一个词汇是否有对应单词例句数据。若有则直接加载，若则访问爱词霸词典API对应的数据再通过XML解析并被加载到词汇数据表中。

在每次应用启动后，加载fragment前，都会通过用户词汇数据表中每个单词的时间戳在后台服务的子线程中利用艾宾浩斯遗忘曲线（详见艾宾浩斯遗忘曲线的实现方式）更新用户词汇表中的每个词汇记忆值，与此同时加载各类fragment。

下面对词汇数据表的部分重要字段做详解：

* 单词收藏：在复习功能模块中由用户根据单词id设置，0表示单词没有被收藏，1表示单词已被收藏；在单词初始化后，其默认值为0；
* 单词熟识度选择：在复习功能模块中由用户根据单词id设置，0表示忘记，1表示模糊，2表示认识，3表示熟记；在单词初始化后，其默认值为0；
* 单词时间戳：在复习功能模块中对单词进行熟识度选择后，根据单词id更新该单词的时间戳为当前时间戳。在单词初始化后，其默认值为一个单词的时间戳默认值为1618674655870即2021年04月18日 00时00分00秒；
* 单词记忆值：用户在复习功能模块中完成单词熟识度选择后被设定为100（其与用户熟识度无关）或者应用在每次被启用时由系统设定，其范围在[0-100]且与遗忘值的关系为：遗忘值=100-记忆值。因此单词记忆值是击中函数的重要参数，记忆值越高被击中的概率越小。在单词初始化后，其默认值为0。根据艾宾浩斯遗忘曲线定义，一个单词的记忆值只要大于零就不可能再等于0，因此也可以作为该单词是被已被用户复习过的重要依据。

**4.2.3.2艾宾浩斯遗忘曲线的实现方式**

艾宾浩斯遗忘曲线是有推荐公式的。但是为了系统效率更高，采用取点法使用艾宾浩斯遗忘曲线。在后台数据管理子系统中更新时间戳时，会利用当前时间戳与原时间戳作比较如果差值达到下表数值则更新为当前时间戳并且更新对应的记忆值。时间戳差值更新处理办法如表4-1所示：

表4-1 时间戳差值更新处理办法

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 差值 | 刚刚 | 15min | 20min | 1h | 8h | 1天 | 2天 | 6天 | 30天 |
| 记忆值 | 100 | 58 | 44 | 36 | 34 | 28 | 25 | 21 | 15 |

在复习功能模块中，首先选择与包含在当前书本内的，记忆值小于40且大于0（即已学过且遗忘值大于60）的单词，并将遗忘值传入击中函数。若该单词被击中则放入wordList之中；这样就实现了利用艾宾浩斯遗忘曲线实现对当前词汇本所有已学单词复习的过程。对未学单词的学习详见4.2.4。

### 4.2.4 用户数据的使用

用户数据表中不仅包含用户id和API密钥等用户信息，还记录着词汇本id、单词背诵位置、今日任务进度、任务量、用户应用主题和日期号等信息。

下面对用户数据表的部分字段做详解：

* 词汇本id：表示当前该用户正在使用的书本id，可利用其在词汇本数据表中查询该书本的详细信息；
* 单词背诵位置：默认值为0，当fragment被销毁时将更新用户数据表中该字段信息，单词背诵位置依赖于词汇数据表中的单词id；
* 日期号：默认值为19971010，每天更新一次，当日期号被更新时，用户数据表中的今日任务进度字段会被重置为1并更新相应的用户界面；
* 用户应用主题：默认值为0，在设置功能模块中由用户设置。在主活动启动后fragment建立前，将根据该字段设置当前应用主题；其中0表示白日主题，1表示黑夜主题；

今日任务进度和任务量的字段作用在4.1中已详述，故此处不在赘述。

在复习功能模块中定义“今日剩余量=任务量-今日进度”（任务量内且未进行熟识度选择的单词），在利用艾宾浩斯遗忘曲线初始化wordList后，将在根据书本id从单词背诵位置开始选择今日剩余量大小的单词数追加到wordList中。这样就实现了对当前词汇本所有未学部分单词复习的过程。

## 4.3 数据库设计

### 4.3.1 表间关系图

表间关系图如图4-8所示：

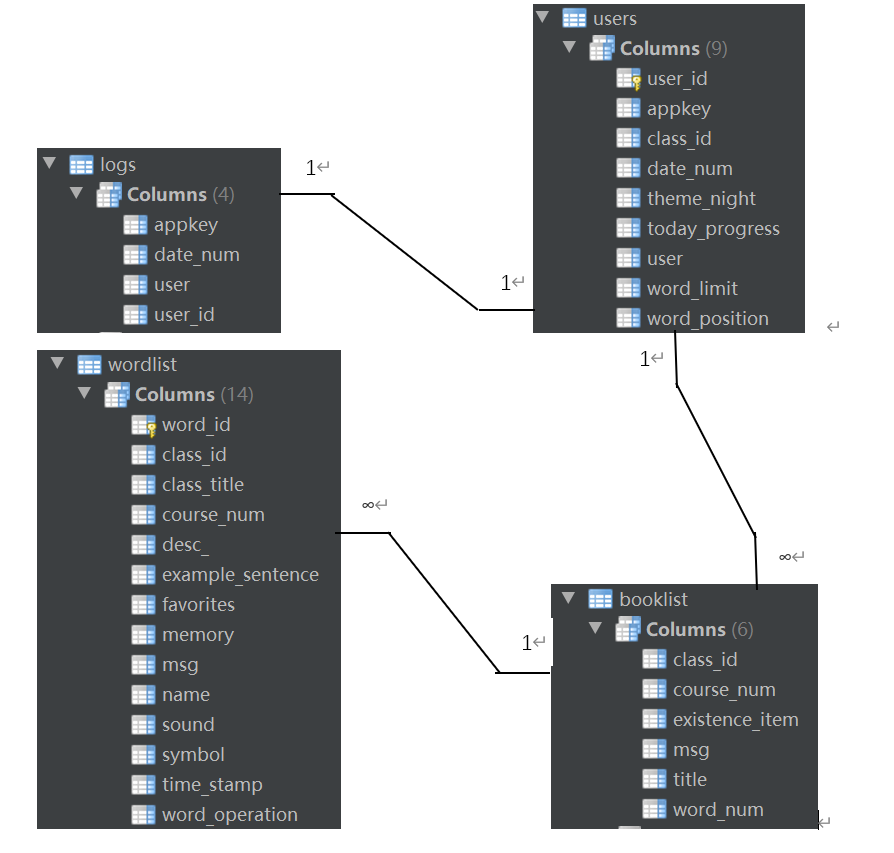


图4-8 表间关系图

### 4.3.2 表汇总

表4-2 数据库表汇总表

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| logs | 登录日志表，用户登录成功记录 |
| users | 用户信息表，记录具体的用户信息 |
| wordlist | 用户词汇数据表，记录用户词汇使用信息和词汇信息 |
| booklist | 用户词汇本数据表，记录用户词汇本使用信息和词汇本信息 |

### 4.3.3 表明细

表4-3 登录日志表

|  |  |  |
| --- | --- | --- |
| 表名 | logs | |
| 字段 | 数据类型 | 说明 |
| user\_id | integer | 非空 |
| user | text | 非空 |
| appkey | text | 非空 |
| date\_num | integer | 日期 |

表4-4 用户信息表

|  |  |  |
| --- | --- | --- |
| 表名 | users | |
| 字段 | 数据类型 | 说明 |
| user\_id | integer | PK |
| user | text | 非空 |
| appkey | text | 非空 |
| class\_id | text | 书本id |
| word\_position | integer | 默认100 |
| today\_progress | integer | 默认1 |
| date\_num | integer | 日期 |
| theme\_night | integer | 1—黑夜主题；  0—白日主题；  默认0 |

表4-5 用户[词汇本数据表](#_出版社（合作伙伴信息）表)

|  |  |  |
| --- | --- | --- |
| 表名 | booklist | |
| 字段 | 数据类型 | 说明 |
| msg | text | 默认failure |
| title | text | 书本名 |
| class\_id | text | 书本id |
| word\_num | integer | 默认0 |
| course\_num | integer | 默认0 |
| existence\_item | integer | 1—词汇数据存在；  0—词汇数据不存在；  默认0 |

表4-6 用户词汇数据表

|  |  |  |
| --- | --- | --- |
| 表名 | wordlist | |
| 字段 | 数据类型 | 说明 |
| word\_id | integer | PK，自增 |
| msg | integer | 默认failure |
| class\_id | integer | 书本id |
| course\_num | integer | 默认0 |
| symbol | text | 音标 |
| sound | text | 发音地址 |
| name | text | 英文单词 |
| descrimination | text | 释义 |
| favorites | integer | 1—收藏单词；  0—不收藏单词；  默认0 |
| time\_stamp | integer | 时间戳  默认1618674655870 |
| memory | integer | 记忆值，默认0 |
| word\_operation | integer | 3—熟记；  2—认识；  1—模糊；  0—忘记；  默认0 |
| example\_sentence | text | 例句 |

# 第五章 应用编码、跨设备安装及测试

## 5.1 文件结构

本系统的文件层次结构如图5-1所示。

在Java中, httpHelper中存放所有完成API数据访问和解析文件，db中存放数据库统计和对相关数据表的CURD文件；book\_list和word\_list中是临时存放从数据表读出的词汇表数据和词汇数据的文件；fragment中存放的是各大功能模块对应的fragment，MainActivity就是创建并管理这些fragment的主活动；当数据库中需要CURD大量数据时，由DBService后台服务完成相关工作。ui文件中放有和用户界面相关的文件，需要配合res中文件使用。

res中主要含有布局相关文件，图片和图标资源文件，字体配置文件，颜色配置文件以及控件风格配置文件。其作用包括定义fragment切换风格，定义导航栏模块风格，定义主活动和各fragment的用户界面以及定义白日主题和黑夜主题等。

AndroidManifest文件针对的是整个应用的属性配置，包括系统权限，系统主题以及对Android四大组件的管理等。

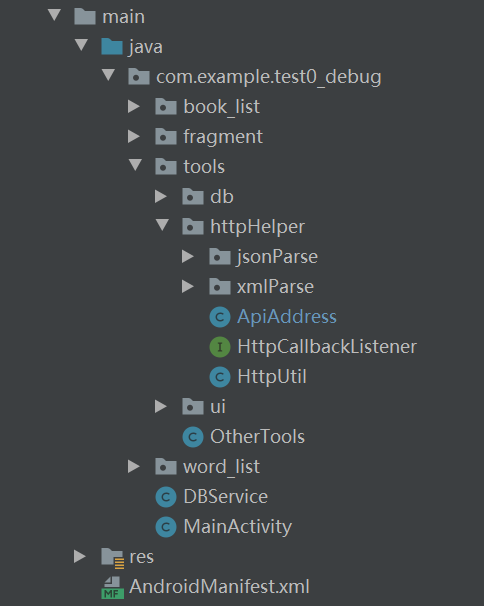


图5-1 背单词app系统文件层次结构图

## 5.2 典型编码

### 5.2.1 数据库操作封装类DBHelper和DBStatistics

DBHelper类封装了数据库的相关操作（比如数据库的建立与升级等），重点用于配合各功能模块完成CURD操作。DBStatistics类用于从数据库中获取统计，该类主要服务于统计功能模块。

DBHelper类关键代码如下：

public class DBHelper extends SQLiteOpenHelper {

……

public DBHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {

super(context, name, factory, version);

}

/\*建表，若已有表则不执行该方法\*/

public void onCreate(SQLiteDatabase db) {

db.execSQL("create table logs("+……+");");

db.execSQL("create table users(" +……+ ");");

db.execSQL("insert into users(……) values (……) ");

db.execSQL("create table booklist(" +"……+ ") ;");

db.execSQL("create table wordlist(" + ……+ ") ;");

}

/\*设置黑夜主题\*/

public static void setThemeNight(Context context, int theme\_night, int user\_id) {……

String sql = "update users set theme\_night=" + theme\_night + " where user\_id=" + user\_id + ";";

db.execSQL(sql); ……

}

…………//省略设置各表字段函数，方法与上相似

/\*sqlite字符转义\*/

public static String sqliteEscape(String keyWord) {keyWord = keyWord.replace("/", "//");

keyWord = keyWord.replace("'", "''");

……//置换转义字符

keyWord = keyWord.replace(")", "/)");

return keyWord;

}

DBStatistics类关键方法如下：

public static int getFavoritesWordNum(Context context) {……}//获取收藏数

public static int getTotalWordNum(Context context) {……}//今日任务量

public static int getRestWordNum(Context context) {……}//今日剩余

public static int getLearnedStatistic(Context context) {……}//已学

public static String getProgressStatistics(Context context) {……}//今日进度

public static int getMemorizingStatistic(Context context) {……}//熟识数

public static int getKnowingStatistic(Context context) {……}//认识数

public static int getVaguenessStatistic(Context context) {……}//模糊数

public static int getForgettingStatistic(Context context) {……}//遗忘数

### 5.2.2 主题选择代码

黑夜主题和白日主题在主活动中完成初始化，其颜色对比定义在values和values-night中。系统加载主题时关键代码如下：

if (DBHelper.getThemeNight(this,1)==DBHelper.THEME\_NIGHT){

StatusBarUtil.setStatusBarMode(this, false, R.color.status\_bar\_color);

AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE\_NIGHT\_YES);

}else{

StatusBarUtil.setStatusBarMode(this, true, R.color.status\_bar\_color);

AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE\_NIGHT\_NO);}

### 5.2.3 击中函数实现代码

击中函数参数为记忆值，击中率为(100-memory)/100，击中函数实现的关键代码：

public static boolean random\_hit(int memory) {

if (memory < 0)

memory = 0;

int num = (int) (Math.random() \* 101);//产生[0-100]随机数

return num <= 100 - memory;

}

### 5.2.4艾宾浩斯遗忘曲线的实现方法

前一章提到为了系统效率更高，采用取点法使用艾宾浩斯遗忘曲线。在后台数据管理子系统中更新时间戳时，选择与包含在当前书本内的已学过且遗忘值大于60的单词，并将遗忘值传入击中函数。若该单词被击中则放入wordList之中；

艾宾浩斯取点法关键代码：

public static int ebbinghaus(long relativeTimeStamp) {

if (relativeTimeStamp > THIRTY\_DAY\_MILLIS)//时间戳差值超过30天

return 15; ……

else if (relativeTimeStamp > EIGHT\_HOUR\_MILLIS)

//时间戳差值超过8小时

return 34;……

else

return 100;

}

DBService实现按艾宾浩斯遗忘曲线更新记忆值的方法：

public int onStartCommand(Intent intent, int flags, int startId) {……

switch (intent.getIntExtra(DBHelper.SERVICE\_SIGN, 0)) {

case DBHelper.SET\_EBBINGHAUS\_MEMORY:

new Thread(() -> {

//记忆值更新条件：1.已学 ；2.本书相关

String sql1 = "select memory,word\_id from wordlist where class\_id='" +intent.getStringExtra("class\_id") + "' and memory!=0;";

Cursor cursor1 = db.rawQuery(sql1, null);

if (cursor1.moveToFirst()) {

do {

int word\_id = cursor1.getInt(cursor1.getColumnIndex("word\_id"));

DBHelper.setMemoryByEbbinghaus(getBaseContext(), word\_id);

} while (cursor1.moveToNext());

}cursor1.close();stopSelf(startId); }).start();

break;

实现复习功能模块按艾宾浩斯遗忘曲线安排复习已学单词关键代码：

new Thread(() -> {……

//被初始化入wordList条件：1.记忆值小于40；2.被击中函数击中；

String sql = "select \* from wordlist where class\_id='" + DBHelper.getBookClassId(context,1)+"' and memory<40 and word\_operation!="+DBHelper.MEMORIZING +" limit "+wordPosition + " ;";

Cursor cursor = db.rawQuery(sql, null);

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

do {

int memory = DBHelper.getMemory(context, cursor.getInt(cursor.getColumnIndex("word\_id")));

if (OtherTools.random\_hit(memory)) {

addWordDataList(cursor);// 被击中则初始化入wordList

}

} while (cursor.moveToNext());

}……}).start();

### 5.2.5安排用户学习未学词汇的代码

在使用艾宾浩斯遗忘曲线安排用户复习后，还需要安排用户学习任务量内且未进行熟识度选择的单词，并将在根据书本id从单词背诵位置开始选择今日剩余量大小的单词数追加到wordList中。其关键代码如下：

wordPosition = DBHelper.getWordPosition(context, 1);//背诵的单词位置

wordProgress = DBHelper.getTodayProgress(context, 1);//今日进度

wordLimit = DBHelper.getWordLimit(context, 1);//任务量

……

//利用sql语句中的limit关键字，使得今日剩余量=任务量-今日进度,且未学过

String sql = "select \* from wordlist where memory=0 and class\_id='" + DBHelper.getBookClassId(context, 1)+"' limit "+wordPosition + "," + wordLimit- wordProgress + " ;";

Cursor cursor = db.rawQuery(sql, null);

if (cursor.moveToFirst()) {

//任务量发生改变时不符合实际时，重置

wordLimit = Math.min(wordLimit, cursor.getCount());

wordId = cursor.getInt(cursor.getColumnIndex("word\_id"));

……

new Thread(() -> {

do {

addWordDataList(cursor);//符合条件的词汇安排入wordDataList

} while (cursor.moveToNext());……

}).start();

}

### 5.2.6 实现用户熟识度安排复习

对于在今日进度中已学过的单词，系统将使用按照用户熟识度安排复习的方法。用户在学习未学单词时必须要为每一个单词进行用户熟识度选择。用户在选择熟识度同时，该单词会根据对应的熟识度被击中函数概率安排到hitList；系统在准备下一个单词时又有50%的概率由击中函数安排是hitList中的单词（除非hitList或者wordList是空的）。

* 熟识度选择时，点击事件实现关键代码：

switch (v.getId()) {

……//wordOperation中第一个参数表示记忆值

case R.id.memorizing:

if (wordDataList.size() != 0)

wordOperation(100, DBHelper.MEMORIZING);

break;

case R.id.knowing:

if (wordDataList.size() != 0)

wordOperation(80, DBHelper.KNOWING);

break;

case R.id.vagueness:

if (wordDataList.size() != 0)

wordOperation(60, DBHelper.VAGUENESS);

break;

case R.id.forgetting:

if (wordDataList.size() != 0)

wordOperation(40, DBHelper.FORGETTING);

break;

……

}

* 击中函数根据用户熟识度安排单词到hitList的关键代码：

private void wordOperation(int possibility, int word\_operation) {

……//击中函数安排被击中的单词到hitDataList中

if (OtherTools.random\_hit(possibility)) {

hitDataList.add(wordDataList.get(0));

prepareNextData();……

}

}

* 击中函数准备下一个单词来源的关键代码：

private void prepareNextData() {

if (hitDataList.size() != 0)

isFromHitDataList = OtherTools.random\_hit(50);//50%来自hitList

if (isFromHitDataList && hitDataList.size() != 0) {

wordId = hitDataList.get(0).getWordId();…}//安排hitList中单词

else {//isFromWordDataList

wordId=wordDataList.get(0).getWordId();…}//否则安排wordList中单词

}

## 5.3 应用跨设备安装

AndroidStudio开发环境提供了installDebug功能。在AndroidStudio右侧栏选择Gradle，在install选项中选择installDebug，点击运行，等待应用安装完成。

选择installDebug页面如图5-2所示：

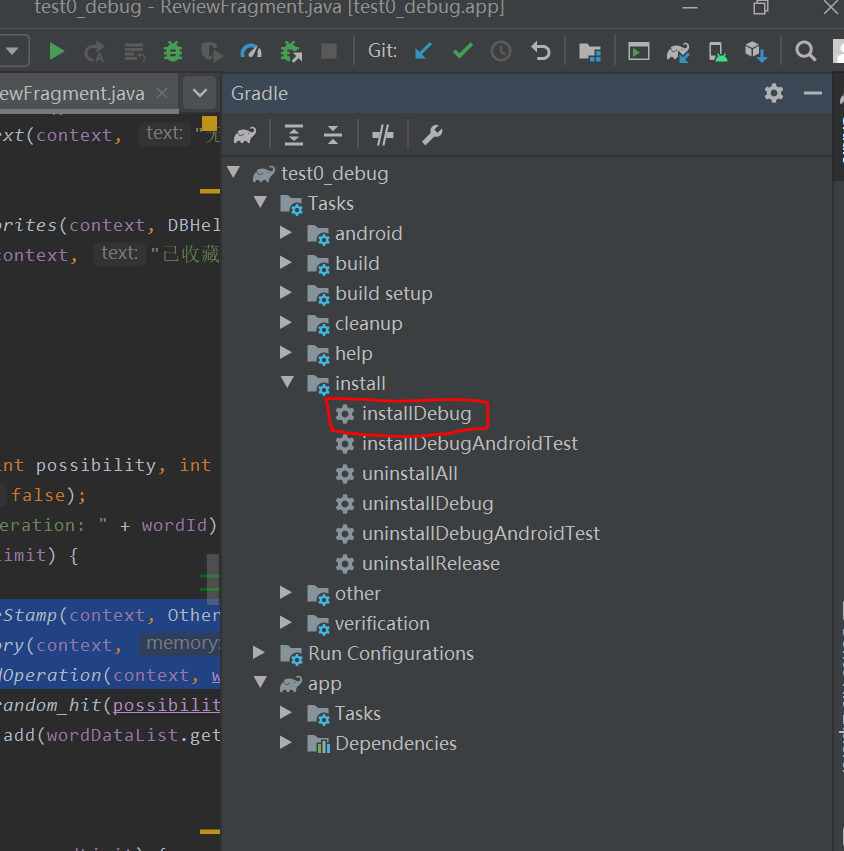


图5-2 使用installDebug调试项目

此时，经过installDebug的apk，在任意一款Android7.0以上系统的手机或模拟器中都可以直接安装运行。可以通过系统自带的应用分享接口，也可以通过qq或者微信将apk分享到不同的安卓设备上。这样就实现了应用的跨设备安装。

## 5.4 应用测试及总结

本应用的测试主要以模拟器（Android 7.0）测试和用户手机测试（Android 10.0）直接安装并使用背单词App。

使用安卓模拟器测试时，系统运行较为流畅。但是会出现因“主线程任务过多而掉帧”的问题。针对本问题已经对多个数据库的CURD放入子线程中进行实现，大大减轻了掉帧个数，但有时仍然会提示上述问题。

使用安卓手机测试时，“主线程任务过多而掉帧”的情况很少见。但是一些系统默认字体的颜色与模拟器上有很大的不同，造成在黑夜主题和白日主题中提示文字与背景色相融的问题。针对本问题已经为两种主题分别在控件上指定了不同字体颜色，现已得到完全地修复。

当然，仍有少部分功能暂时尚未能实现，但亦为其留下了接口，方便日后完善（如用户自定义名称，单词查询界面优化和历史记录生成等）。

# 第六章 结束语

通过近一个学期的努力，开发完成的背单词App界面友好，功能丰富，用户体验度相对较高。

## 6.1 毕业设计的难点与创新

本次设计的最大难点就是使用API时访问次数不能过于频繁的问题，这样会造成在Android在解析Json数据时突然遭受异常而终止。由于是多线程加载数据。，开发者往往难以确定数据库数据加载进度，同时也对用户的使用体验造成极差的影响。经过综合考虑，发现使用词汇本数据预加载，词汇数据用户选择时加载，具体单词的例句数据逐个加载的方式是最好的解决方案。

本次设计的特色主要有以下几点：

* ViewPager和RadioButton的使用避免了fragment内容的重复加载；
* 多线程技术的应用有效地提高了应用运行时的流畅度；
* 精心设计的美工。

## 6.2 毕业设计的收获

通过这次毕业设计以及撰写本毕业论文，我学习了SQLite技术，虽不能说对其已非常了解，但它的一些经典应用一比较熟悉。同时我对使用fragment技术的建立和使用方式有了进一步的认识。我将大学四年学到的Android开发和性能优化等知识充分应用于实践，提高了自己的动手能力。

毕业设计能够很好地提高我们的综合能力。除了学习到了相关的技术之外，还培养了自学、独立的解决问题的能力。

# 致   谢

在毕业设计即将结束之际，我要真诚地感谢我的指导老师——邱晓华老师。在整个毕业设计过程中，邱老师给了我悉心的指导和热情的帮助。在毕业设计刚开始时，老师耐心地给我分析背单词App系统的开发研究思路，使我一开始就能够很好地把握开发设计的方向。在毕业设计开发过程中，邱老师始终在百忙之中为我在毕业设计中遇到的问题给予解答，使我的毕业设计能够顺利地进行下去，在整个系统开发即将结束的时候，老师又很负责地给我提出了很多改进意见，并帮助我一起解决遇到的难题，使我能够很好地完成本次毕业设计，所以我要再一次地感谢邱老师。

另外在本次毕业设计过程中，我也得到了朋友的帮助，特别是在代码设计和程序调试过程中，他们都很热心地给出了自己的意见或者帮助我解决了一些小问题，在此我也要向他们表示衷心的感谢。

# 参考文献

[01] 李娜. 中小学电教：上[J]. 教育, 2015（2）1：63

[02] 刘宇轩. 基于安卓系统的 APP开发技术研究[J] ．电子科技,2019（12）14：90

[03] 黄宏程. 胡敏，陈如松．Andriod移动应用设计与开发[M]．北京：人民邮电出版社，2018

[04] 边玉芳. 艾宾浩斯的记忆遗忘曲线实验[J]. 教育，2013（03）1：24

[05] 郭霖. 第一行代码 第3版[M]． 北京：人民邮电出版社，2019

[06]陈彦恒. 企业库存定位盘点系统和质检数据采集系统的研究与设计[D].西南交通大学,2014.

[07]顺顺.Android高占有率,监管难[J].软件工程师,2013(04):10-11.

[08]李世波. 中山市公安局网上党校系统的设计与实现[D].电子科技大学,2013.

[09]陈熙. 泛在学习下移动协同编辑技术的研究与应用[D].西安电子科技大学,2013.

[10]蔡蔚. 基于隐私保护的Android应用的设计与实现[D].北京交通大学,2018.

[11]杨昌庆. 基于Android的云购软件的设计与实现[D].北京交通大学,2017.

[12]陈飞飞.基于DOM4J的XML文档解析技术研究与应用[J].软件导刊,2016,15(03):36-37.

[13]保冲兵. 基于移动智能终端的家庭多屏互动系统设计与实现[D].电子科技大学,2018.

[14] 王承君．Web数据库技术应用教程[M]．北京：中国水利水电出版社，2016

[15] 朱威. Android开发环境中的SQLite性能优化[J]. 湖南邮电职业技术学院学报,2015-01-11(1)

[16] 薛宾. 嵌入式数据库SQLite在Java中的应用[J].天津职业院校联合学报, 2008-04-14(4)

[17] 丰君子. 谷歌的安卓攻略（二）[J]. 世界经济，2016（08）11：20

[18] 欧阳燊. Kotlin从零到精通Android开发[M]. 北京：清华大学出版社，2018

[19] 马歇尔·普拉德．Google之路[J]．产业经济，2003（01）2：8

[20] 卢建红. 重视课文背诵，有效提高英语学习效率[J]. 教学,2009（7）21：26