摘 要

调查显示，我国的年轻人平均每天解锁屏幕35次，在解锁屏幕时背单词可以有效利用琐碎时间。本APP可以让用户通过对锁屏界面单词的拼写和发音来进行中文选词达到记忆单词的目的，当用户选择错误时，将单词加入错题本。同时支持切换壁纸、用户登录、错词和复习词上传、拉取数据功能、解锁需要的单词个数和选择词库难度等用户个性化设置，以此来实现多设备同步的功能，切合用户使用特点。本应用还提供了小讲堂功能，向用户推送最新的英语知识和英语技巧，给用户带来极大的便利。

**关键词：安卓应用开发；异步编程；网络编程；国际化；VCS**

Abstract

According to a survey, young Chinese unlock their screens an average of 35 times a day, and memorising words while doing so can help them make the most of their time. This APP allows the user to memorize words by spelling and pronunciation of words on the lock screen. When the user makes a wrong choice, he/she adds words to the wrong text. At the same time, it supports user's personalized Settings such as wallpaper switching, user login, wrong word and review word uploading, pull data function, number of words needed to unlock, difficulty of selecting thesaurus, etc., so as to realize the function of multi-device synchronization and meet users' usage characteristics. The app also provides a small lecture room to push the latest English knowledge and Skills to users, which brings great convenience to users.

**Keywords: Android application development; asynchronous programming; network programming; internationalization; VCS**

目录

[摘 要 I](#_Toc61288028)

[Abstract II](#_Toc61288029)

[1. 绪论 4](#_Toc61288030)

[1.1. 设计背景 4](#_Toc61288031)

[1.2. 设计目的与内容 4](#_Toc61288032)

[2. 系统设计 4](#_Toc61288033)

[2.1. 需求分析 4](#_Toc61288034)

[2.2. 功能设计 5](#_Toc61288035)

[2.3. 界面设计 6](#_Toc61288036)

[2.4. 数据库设计 7](#_Toc61288037)

[2.4.1. 配置数据库 7](#_Toc61288038)

[2.4.2. 单词数据库 7](#_Toc61288039)

[2.4.3. 格言数据库 8](#_Toc61288040)

[3. 算法逻辑 10](#_Toc61288041)

[3.1. 业务流程 10](#_Toc61288042)

[3.2. 子功能流程 11](#_Toc61288043)

[4. 系统实现 15](#_Toc61288044)

[4.1. API服务器实现 15](#_Toc61288045)

[4.2. 用户登录功能实现 15](#_Toc61288046)

[4.3. 上滑换壁纸功能实现 16](#_Toc61288047)

[4.4. 小讲堂功能实现 17](#_Toc61288048)

[4.5. SQLite数据库实现 18](#_Toc61288049)

[4.6. 复习本实现 19](#_Toc61288050)

[4.7. 国际化 19](#_Toc61288051)

[4.8. 手机解锁背单词实现 19](#_Toc61288052)

[5. 系统测试 20](#_Toc61288053)

[5.1. 锁屏界面 20](#_Toc61288054)

[5.2. 主界面 22](#_Toc61288055)

[5.3. 复习本界面 23](#_Toc61288056)

[5.4. 错词本界面 24](#_Toc61288057)

[5.5. 设置界面 25](#_Toc61288058)

[5.6. 小课堂界面 28](#_Toc61288059)

# 绪论

## 设计背景

随着经济全球化，我们与外国人的接触越来越多，英语已经成为了中国与外面交流的主要工具，同时我国的英语教育也开始普及到了小学，人们对学习英语越来越重视。而对于学习英语特别是英语单词记忆，记忆单词的方式也在不断的更新，而今移动互联网时代的到来，尤其智能手机的广泛应用，英语学习的平台已经逐渐向计算机或者手机等移动设备上转移，背单词的方式方法进入一个新的领域。

根据有关调查，中国的年轻人平均每天解锁手机屏幕35次，若在单纯的手机屏幕解锁中增加英语单词的学习，那么每天将能够学习至少30个单词。日复一日，用户的英语词汇量将会猛涨。“锁屏背单词”，锁屏结合单词，让手机屏幕解锁变得更加有意义。

## 设计目的与内容

英语作为国际性通用的语言，在当今全球化日益加深的趋势下，懂一门外语，流利说一门外语，十分重要。学好英语，走遍世界都不怕！英语的通用性，已经跨出了文化的界线，更是成为了衡量一个人的见识、国际化的程度。英语是全球通用的语言，我们需要这种语言去获取全球最新消息的知识和资讯。并且英语是各大公司都认可的能力，英语能力就是职场硬实力。

本应用将锁屏与英语单词相结合，充分利用琐碎时间，提高用户英语水平；让背单词的方式并不止局限于某个应用内部，在屏幕解锁的过程中增加英语的学习，新颖有趣，抓住用户的特点，提高用户的英语水平。

本应用旨在用户通过对锁屏界面单词的拼写和发音来进行中文选词，将回答正确的加入复习本，错误的加入错题本来帮助用户日后复习和纠正，复习和记忆完成之后还可以点击”我会了“按钮去掉单词，达到背单词的目的。在设备第一次安装应用时，设备会自动下载相应的词库。为防止用户背单词环境过于枯燥，本应用提供了上滑更换壁纸的功能去活跃锁屏环境。本应用还支持用户登录、错题上传与拉取功能，即使更换设备也可以跟踪用户学习情况，错题上传与拉取功能进一步利用用户的琐碎时间来复习错误单词，强化学习，二次记忆，变成英语达人。

# 系统设计

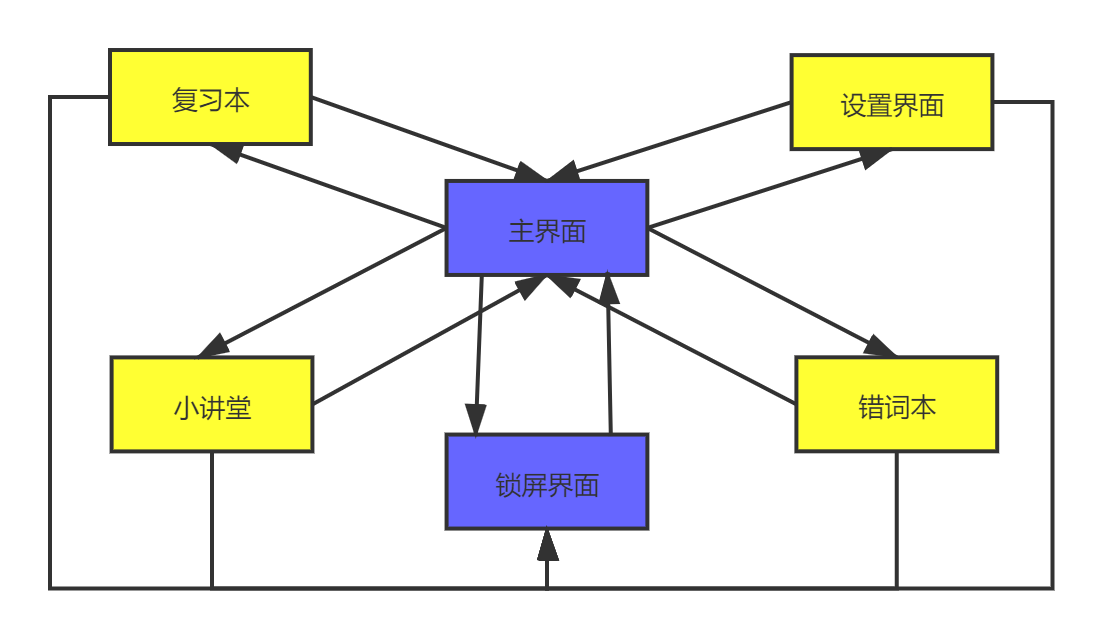
## 需求分析

背单词APP相比传统单词书来说，就是让形音义多管齐下，让每个词汇都能自动发音，迎合了懒人学英语的需求，也纠正了中国人的发音疑难。开发者需要引进词汇的自动发音程序以及使得单词反复出现加深人的记忆。

对于用户来说，用户可能在准备不同的考试，传统情况下，用户可以针对不同的考试需要买不同的单词书，背单词APP可以自己选择自己需要的书籍，这样节省了人力物力，还提高了背单词的效率。

中国的年轻人平均每天解锁手机35次，如果在手机解锁的过程中仅仅只增加一个单词，那么他们每天就能学习到35个单词，一周下来就是245个单词，日复一日，用户的英语词汇量就会猛增，所以开发背单词APP可以让手机锁屏变得更有意义。

## 功能设计



首先我们通过主界面通过按钮跳转到其他所有的界面，其他所有的界面也通过按钮可以跳转到主界面当中去，并且在任何界面直接锁屏可以跳转到锁屏界面，而在锁屏界面中解锁后可以跳转到主界面。

* 手机锁屏

软件的主打功能，用户可以在锁屏界面背单词，达到设置的个数自动解锁，或者向右滑动解锁；可以使得用户抓住碎片时间来提高英语水平，逐渐变成英语细节达人！

* 错词本

提供错题本功能来收集用户答错的单词，方便用户集中学习单词，掌握之前答错的单词；掌握之后还可以点击 “我会了”按钮来删除掌握单词。

* 复习本

提供复习本功能来强化用户的单词学习，二次记忆，长期不忘。

* 系统设置

用户在设置界面可以更改词库难度、解锁屏幕需要的单词个数和服务器地址；还有用户登录、拉取和上传选项，可以登录、拉取之前错词和上传本次错词。

* 用户登录

提供用户登录功能，登录之后可以拉取用户之前答错的单词还可以将本次答错的单词上传服务器。

* 上滑更换壁纸

上滑可以更换壁纸，风格多种多样，改善用户心情，给用户不一样的体验。

* 错题上传与拉取

用户可以根据账户密码从服务器拉取之前答错的单词，还可以将本次答错的单词可以上传到服务器。从此不受设备限制，随时随地背单词！

* 数据库下载

提供数据库下载功能，设备首次安装app会自动下载单词词库。

* 小讲堂

俗话说，授人以鱼不如授人以渔。本App提供小讲堂功能，包含专业英语知识和技巧，帮助用户以正确的学习方式去学习英语。

## 界面设计

* 数据库下载界面

整体是一个垂直方向上的线性布局，首先是一个TextView来提示用户正在下载数据库；接着是一个GifImageView控件来加载loading的动态图片显示正在下载数据库。

* 设置界面

整体是一个垂直方向上的线性布局，接着是一个水平方向上的线性布局来设置是否开启锁屏背单词功能的开关，包含一个TextView控件和一个Switch控件；接着是两个水平方向上的线性布局来设置单词难度和解锁需要的单词个数，其中每个线性布局里面有一个TextView控件和一个Spinner下拉框控件；接着是一个水平方向上的线性布局来提示和保存用户输入服务器地址，包含一个TextView和EditText控件；下面是一个线性布局内的按钮来保存用户输入的服务器地址；接着是两个水平线性布局，分别提示和保存用户输入的用户名和密码；最后是一个水平线性布局，包含四个按钮，分别完成保存用户信息、用户登录、错词拉取和错词上传的功能。

* 复习界面

整体是一个垂直方向上的线性布局，首先是三个垂直分布的TextView，分别显示名人名句，英文名句和中文名句；接下来是一个相对布局，首先是两个TextView来设置十字框，接着是八个TextView来显示答错个数、已掌握个数、学习难度和学习总数。

* 主界面

整体是一个垂直方向上的线性布局，首先是三个垂直分布的TextView，分别显示名人名句，英文名句和中文名句；接下来是一个相对布局，首先是两个TextView来设置十字框，接着是八个TextView来显示答错个数、已掌握个数、学习难度和学习总数。最下面是一个线性布局，里面包含三个按钮，分别是主页、小讲堂和设置。在其他界面点击主页会转换Fragment到主页。点击小讲堂会转换Fragment到小讲堂界面。点击设置会转换Fragment到设置界面。

* 锁屏界面

整体是一个垂直方向上的线性布局，前两个是分别水平居中的TextView，分别显示时间与日期；下面是一个相对布局，显示单词和音标喇叭图片；再下面是一个单选控件RadioGroup，包含单词的三个汉语选项。

* 复习本和错词本界面

复习本界面和错词界面较为相似简单。其功能主要依靠几个按钮的点击来实现。复习本的左滑事件为RightActivity中的nextRight方法。错词本中的左滑事件为WrongActivity中的nextWrong方法。

* 小讲堂界面

整体是一个相对布局，内含一个ListView控件。ListView控件内元素的格式：整体是一个线性布局，内部包含一个SmartImageView控件，接着是一个相对布局，内含三个TextView控件来分别显示小讲堂内帖子的标题、正文和文章数量。

## 数据库设计

### 配置数据库

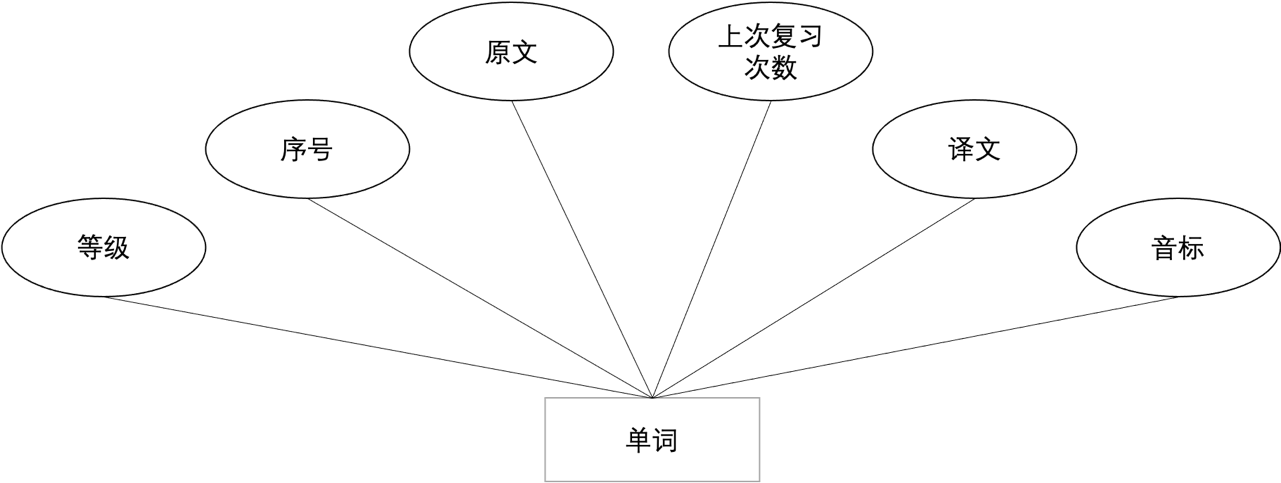
对于程序配置信息数据库，由于配置信息多为键值对（Key-Value）的形式，且较为短小，所以我们采用Android平台提供的轻量级数据库SharedPreferences对配置文件进行存储。

所有信息均存在于调用句柄名为config的SharedPreferences对象中，其中键名及其含义具体为：

* username ： 记录登录用户名，默认为空。
* password ： 记录登录密码，默认为空。
* server\_url ： 记录服务器API入口，默认为所配置测试服务器。
* level ： 记录单词数据库等级，默认为cet\_6。
* wrong ： 记录错误单词，默认为空。
* right ： 记录正确单词，默认为空。
* btnTf ： 记录是否开启锁屏背单词，默认为否。
* tf ： 记录当前是否为锁屏状态，默认为否。

### 单词数据库

对于一个单词，其具有序号，原文，译文等属性，通过分析绘制实体关系图（E-R图）如下：



由于其有明显的关系依赖特征，且在具体应用过程中我们需要根据数据关系进行选择，故这里采用关系型数据库SQLite对单词信息进行储存。

（**序号**，等级，原文，译文，音标，上次复习次数）

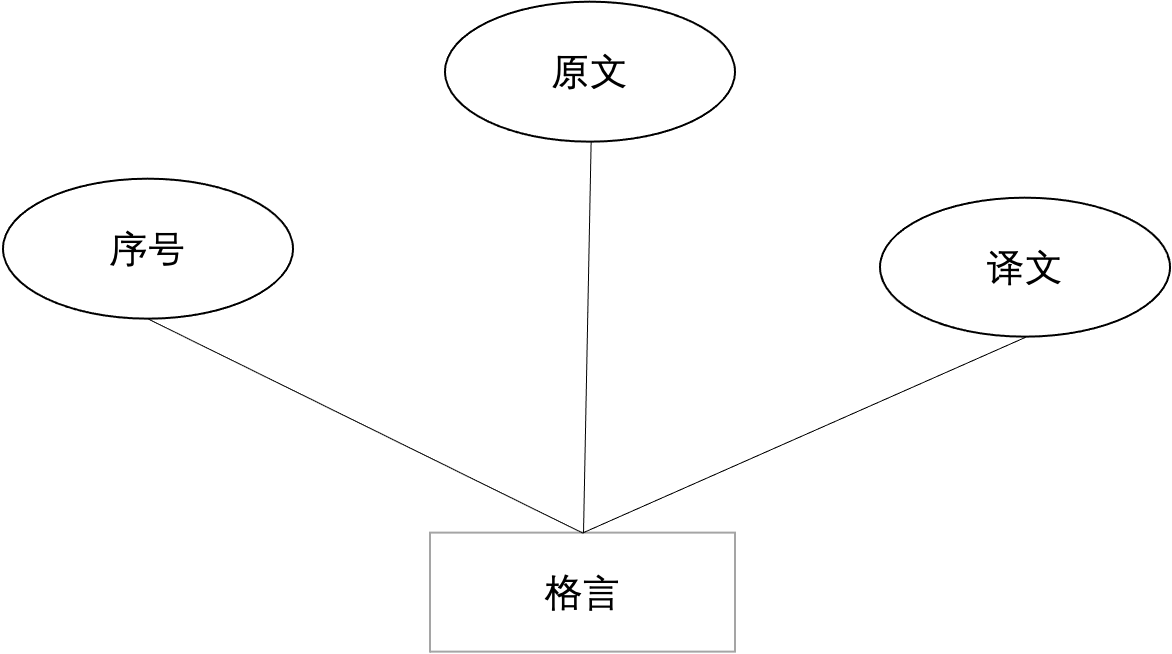
在此关系中，序号为主码，其余属性完全函数依赖于主码，且无传递函数依赖，故此关系为第三范式3NF。

在SQLite数据库具体创建过程中，我们建立两张表——cet*4与cet*6，其中每列具体定义为：

* index ： 序号，整型，主键，非空，自动增长。
* word ： 原文，文本型。
* soundmark ： 音标，文本型。
* chinese ： 译文，文本型。
* last\_review ： 上次复习次数。

### 格言数据库

对于一条格言，其具有需要，原文，译文三个属性，其实体关系图如下：



（**序号**，原文，译文）

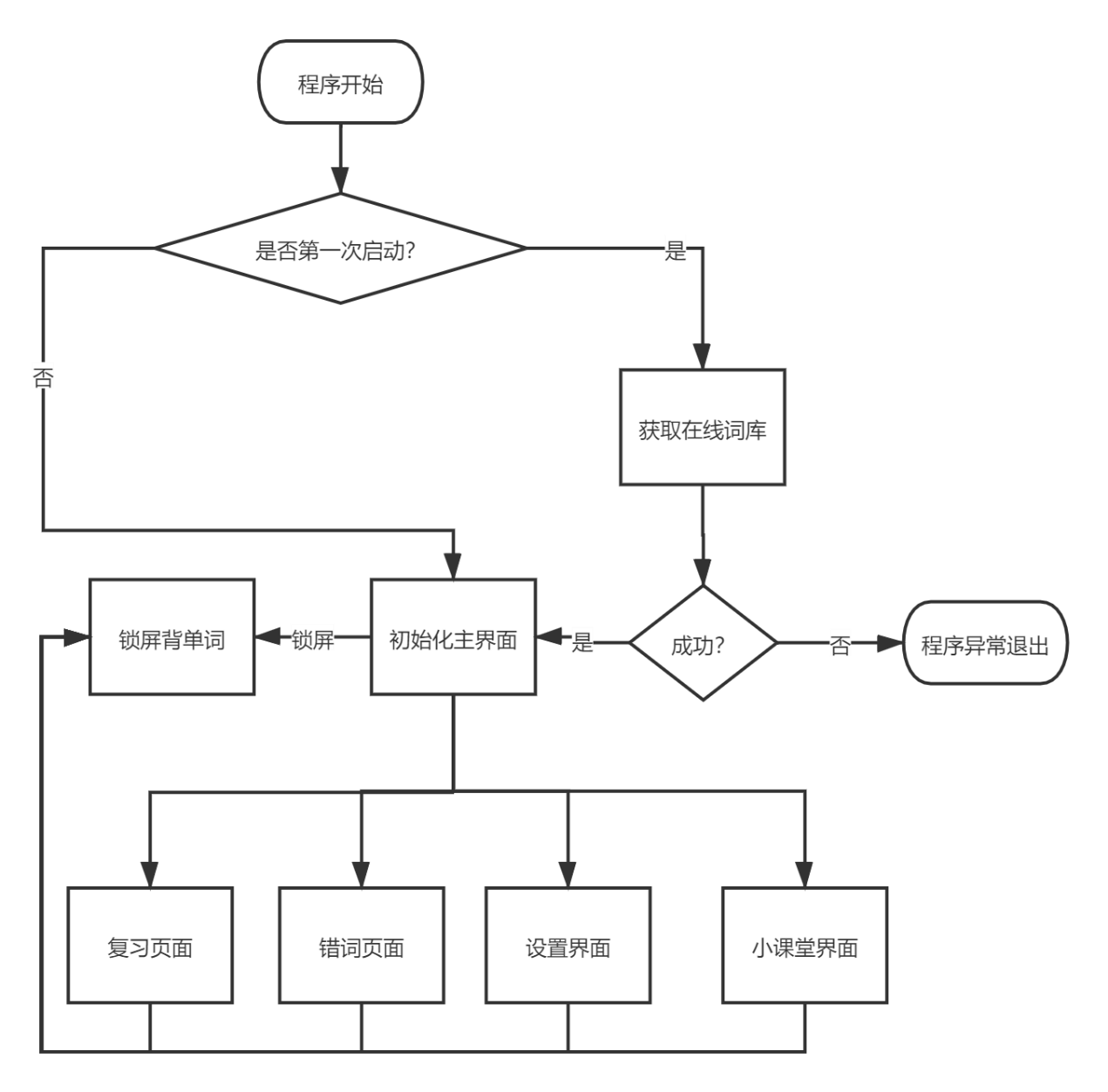
在此关系中，序号为主码，其余属性完全函数依赖于主码，且无传递函数依赖，故格言关系亦为第三范式3NF。

同上，采用SQLite数据库对格言进行储存，具体创建一张表——quote，其中每列具体定义为：

* index ： 序号，整型，主键，非空，自动增长。
* english ： 原文，文本型。
* chinese ： 译文，文本型。

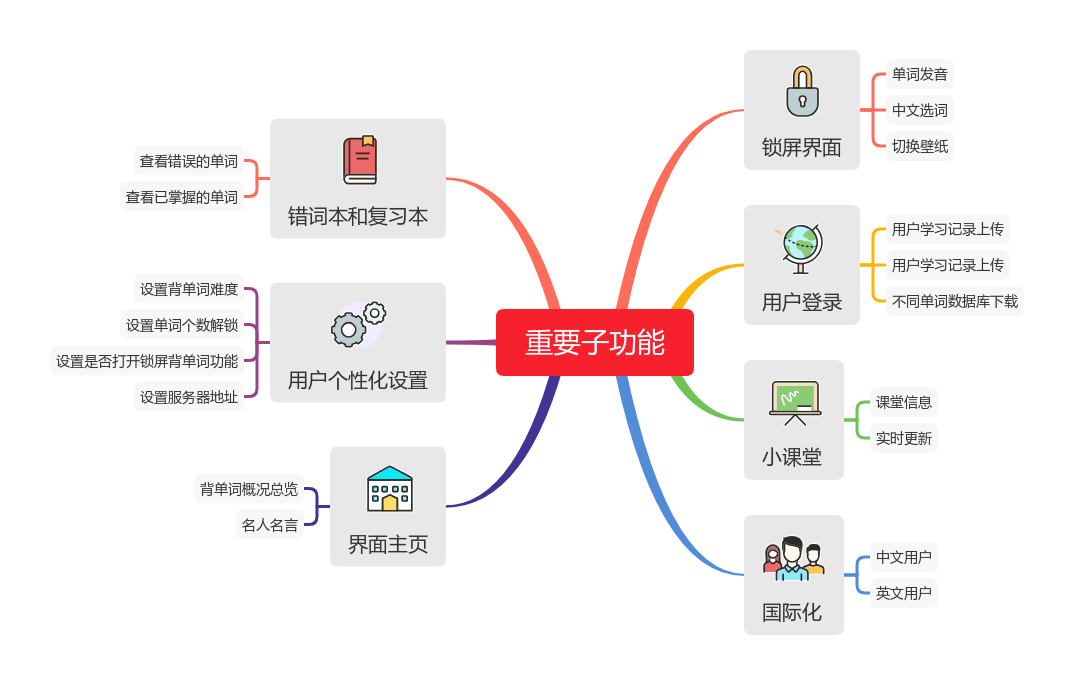
# 算法逻辑

## 业务流程

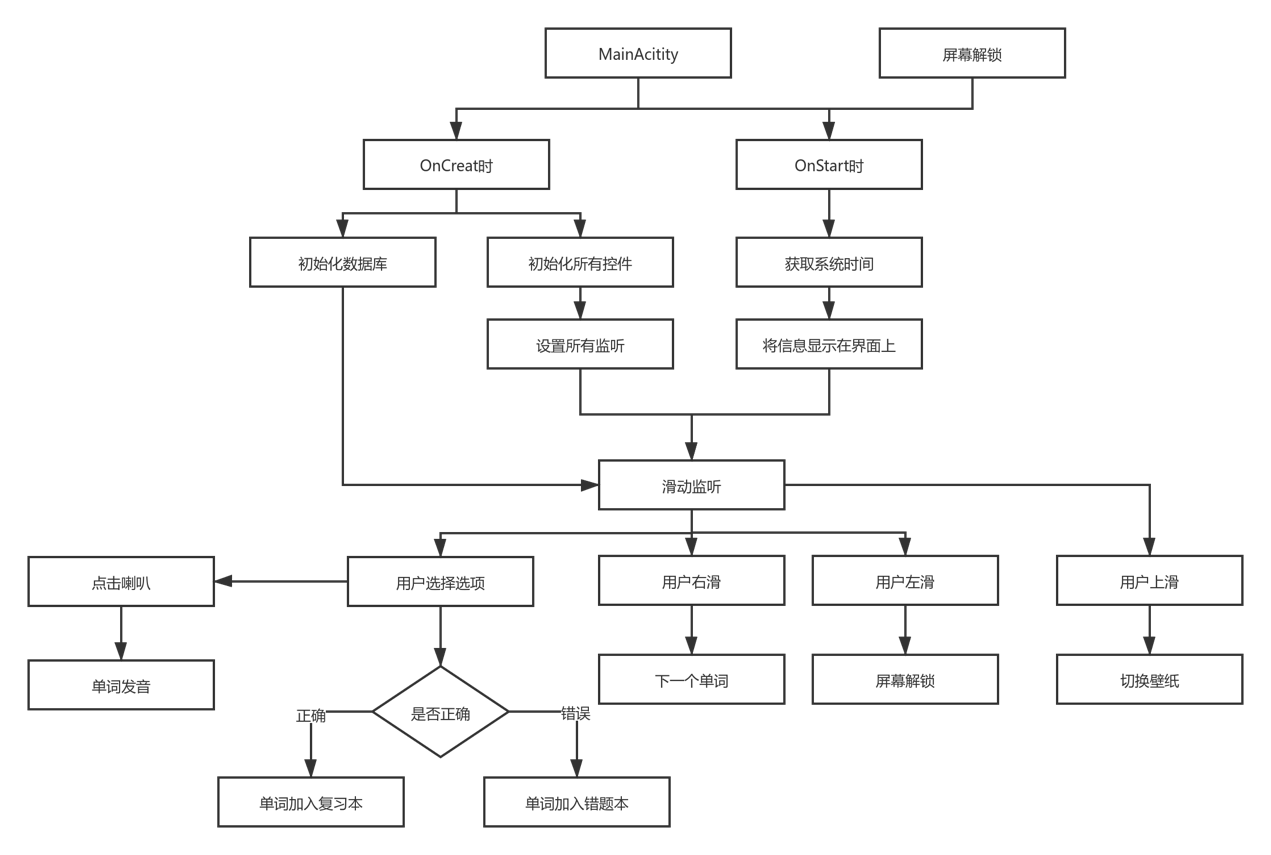


进入程序，首先判断是否是第一次启动，是的话就从服务器上下载词库，下载成功就会跳转到主界面，下载失败就会退出程序。如果不是第一次启动的话就直接跳转到主界面。在主界面锁屏的话会进入锁屏背单词界面，同时也可以进入复习页面、错词页面、设置页面和小课堂界面。同样的，在复习页面、错词页面、设置页面和小课堂界面也可以锁屏进入锁屏背单词界面。

## 子功能流程



* 锁屏界面

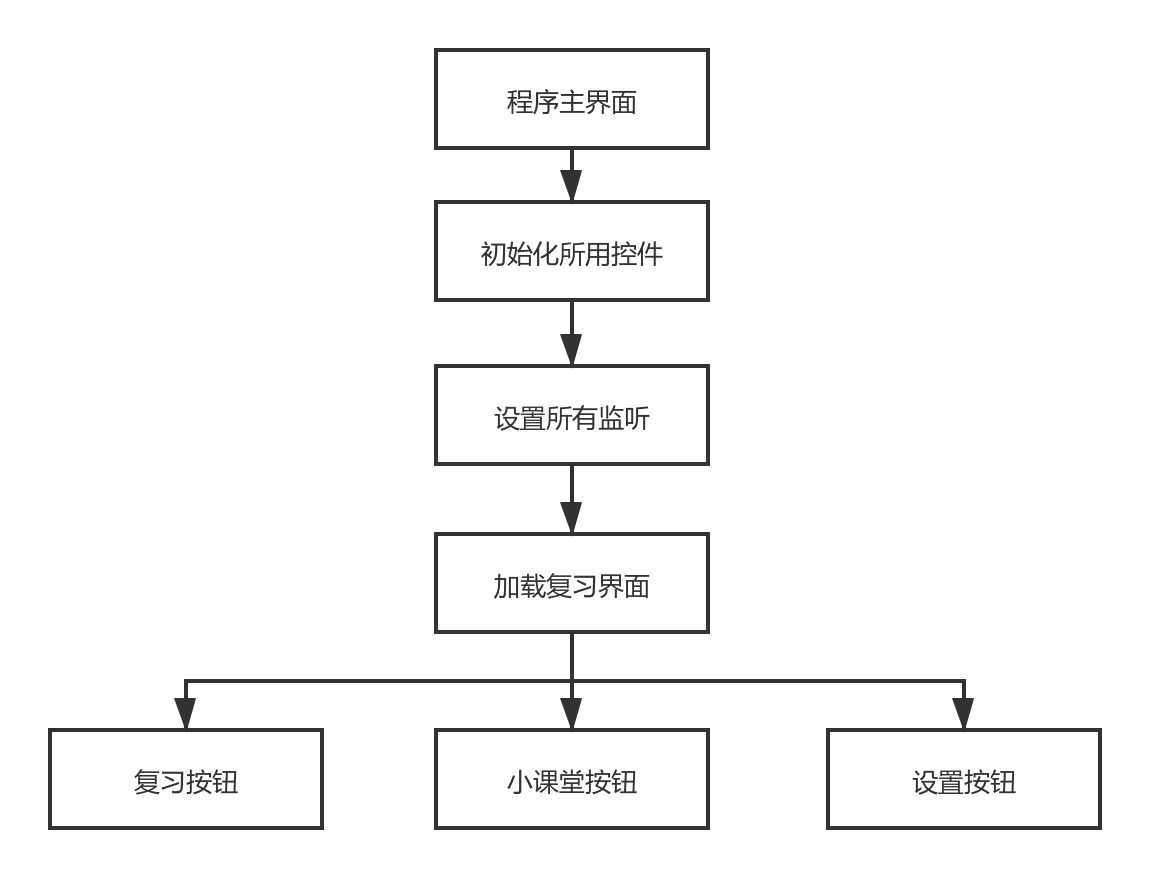


我们可以通过打开应用程序或者通过屏幕解锁来条跳转到MainActivity界面中。启动界面时，我们会初始化所有的控件和监听，当界面可见我们更新我们的系统时间。

初始化完毕后，当用户点击喇叭时，TTS会进行发音，当用户选择的单词是正确，将单词加入复习本中去，当用户选择的单词是错误的，将单词加入到错题本中去，最终根据用户的选择将选项显示不同的文本。

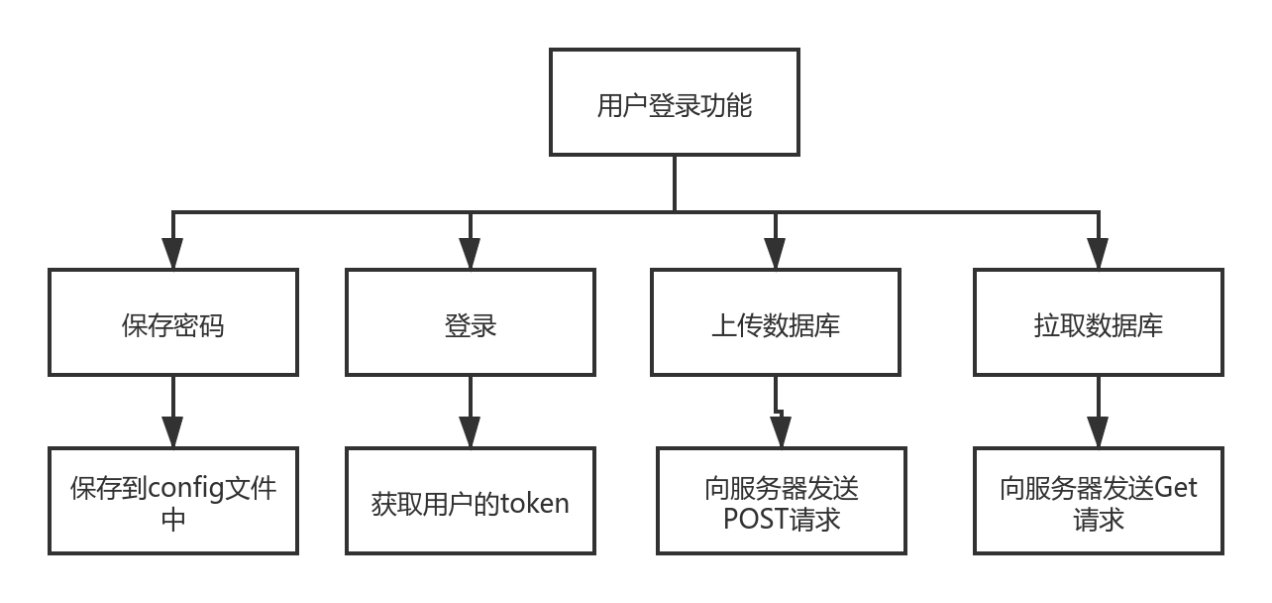
当用户向右滑动屏幕时，将切换下一个单词，当用户向左滑动屏幕时，将解锁屏幕，当用户向上滑动屏幕时，将切换锁屏的背景。

* 界面主页



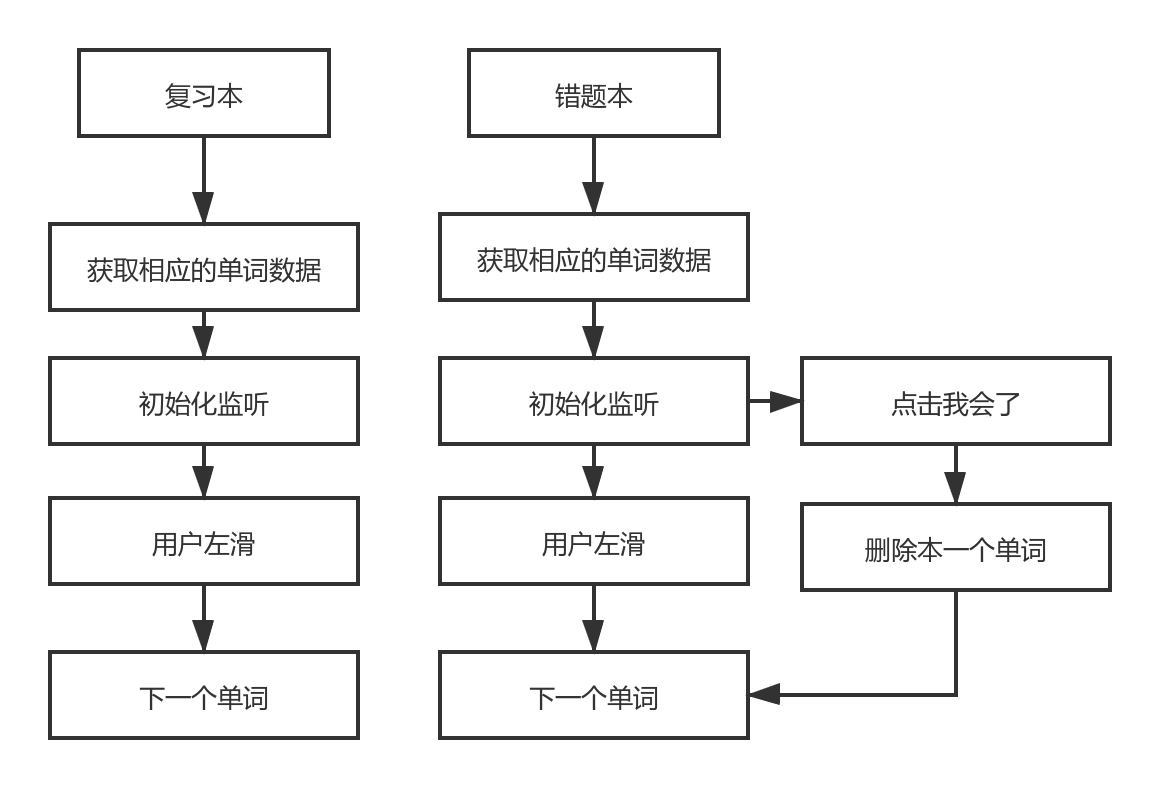
进入主界面之后初始化所有控件然后设置所有的监听，接着是加载复习界面里面的复习按钮、小课堂按钮和设置按钮。

* 用户登录



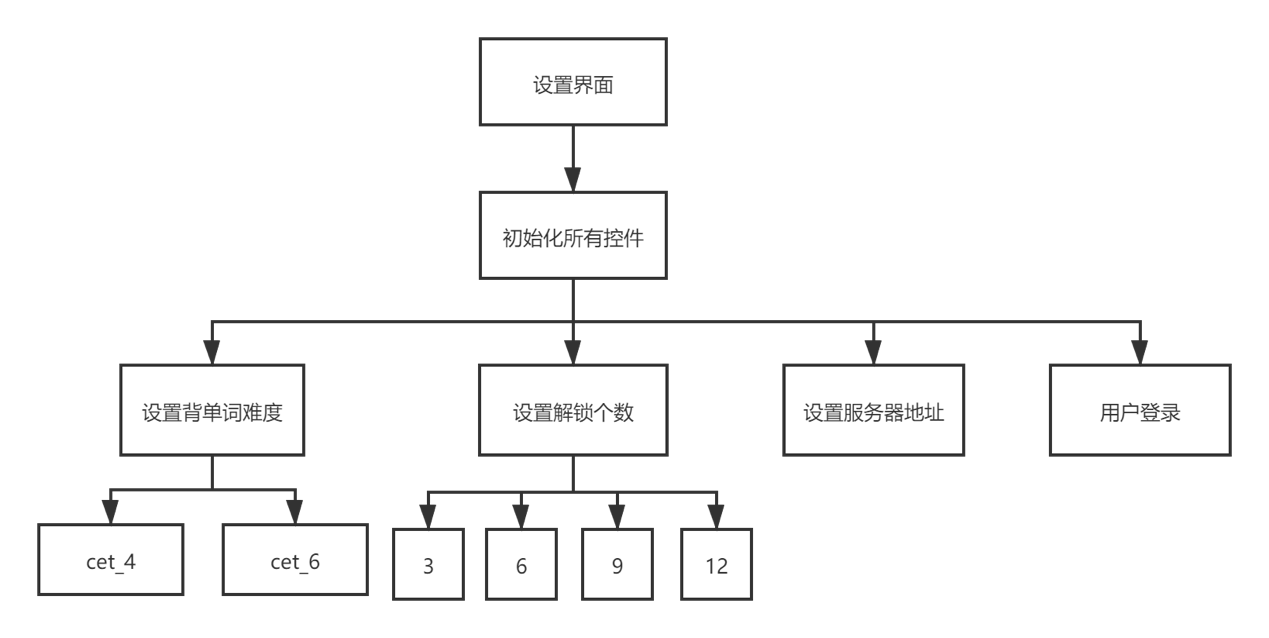
用户登录下面有四个分支，分别是保存、登录、上传和拉取。其中保存到密码是将账号和密码信息保存到config文件中；登录功能是获取用户的token；上传功能是向服务器发送POST请求；拉取数据库是向数据库发送GET请求。

* 错词本和复习本



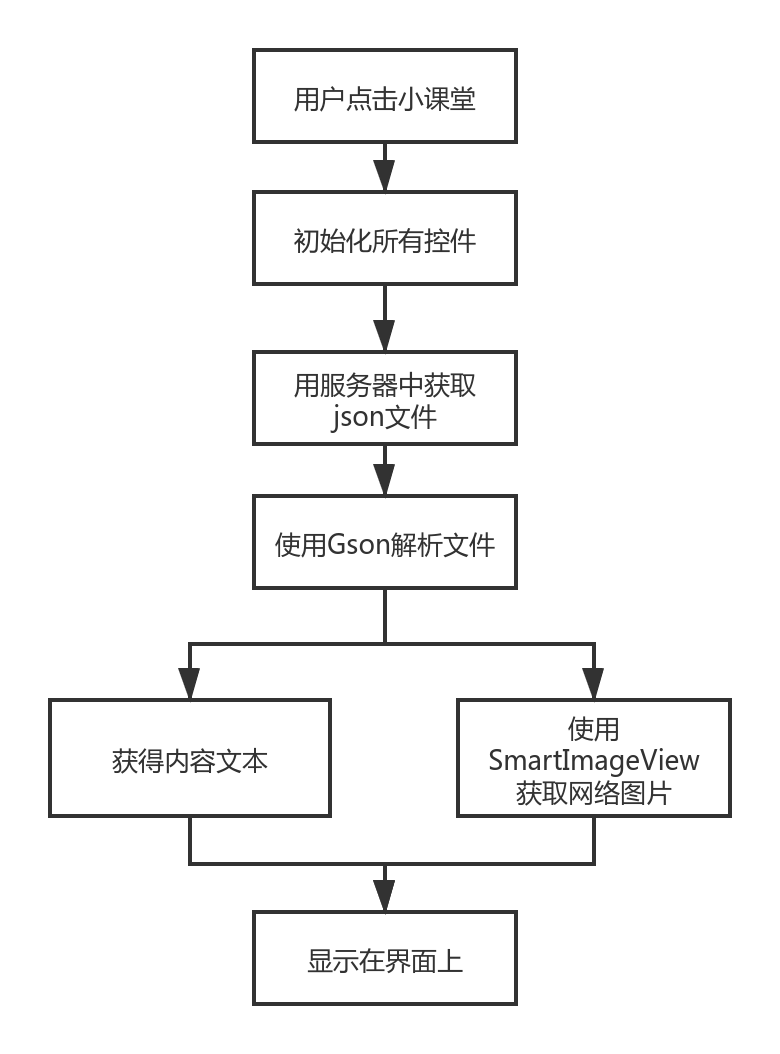
复习本逻辑较为简单：首先获取单词数据，之后初始化监听：用户左滑则是下一个单词。而错词本逻辑则是初始化监听左滑和按钮，点击按钮’我会了‘则会删除目前单词并且显示下一个单词，用户左滑则会显示下一个单词。

* 用户个性化设置



进入设置界面之后首先初始化所有控件，其下有四大主功能：设置单词难度、设置解锁个数、设置服务器地址和用户登录设置。单词难度可以设置为cet\_4或者cet\_6。解锁个数可以设置为3、6、9和12四种选项。设置服务器地址时用户可以填写自己的服务器地址。用户登录则是包含一些用户账号同步功能，例如登录、拉取和上传。

* 小课堂



进入小课堂之后首先初始化所有控件，接着向服务器中获取JSON文件，使用GSON解析文件获得文本内容，同时使用SmartImageView获取网络控件。最后将文本和图片显示在界面上。

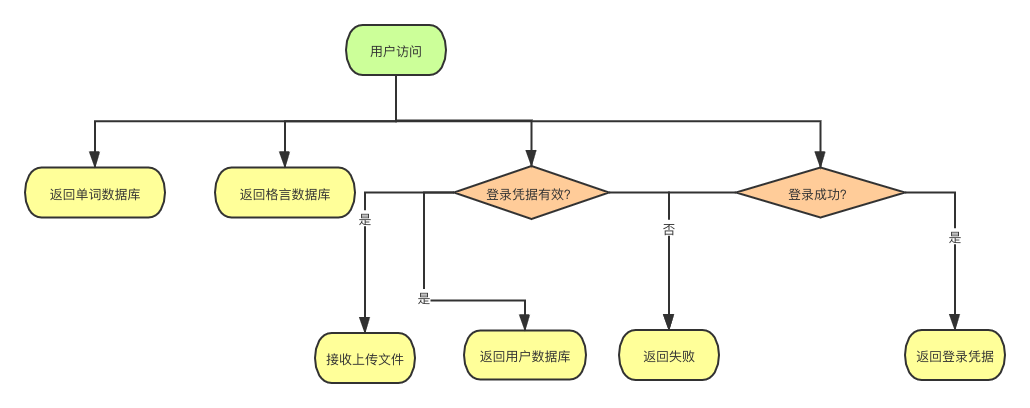
# 系统实现

## API服务器实现

本设计服务端采用Python Flask框架实现，具体实现了以下路由：

* /word ： 获取最新单词数据库
* /quote ： 获取最新格言数据库
* /login ： 获取登录凭据
* /api/upload ： 上传用户单词数据库
* /api/fetch ： 获取用户单词数据库

其业务逻辑具体如下：



在单词与格言路由中，服务器端返回对应初始数据库，不要求验证用户凭据。

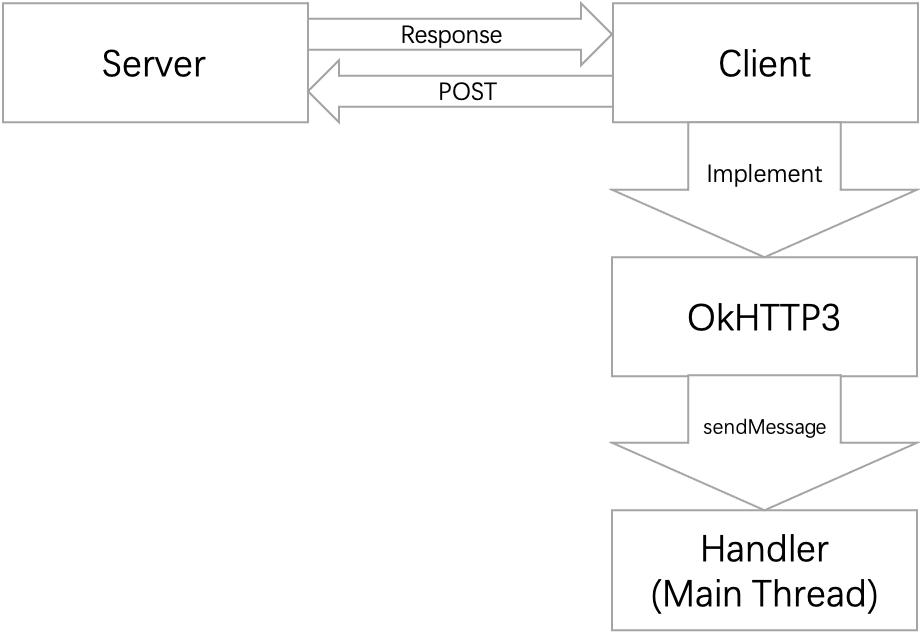
在登录路由中，我们使用了JSON对用户信息进行储存。对用户信息验证后，服务器端使用TimedJSONWebSignatureSerializer生成并返回用户凭据，其有效期为30天。

在上传与下载路由中，用户在用户凭据有效期内，可通过凭据上传或下载用户数据库以同步单词进度。

## 用户登录功能实现

为了实现用户登录并同步单词词库，我们设计了LoginHelper类，其具体实现了下面三种方法：

* login ： 登录并获取用户token
* upload ： 上传用户词库
* fetch ： 下载用户词库



用户在SettingFragment中输入用户名与密码并保存于SharedPreferences后，点击登录即实例化一个LoginHelper对象，该对象具有三个handler，即用于处理异步登录、上传与下载用户数据库功能。在login方法内，通过使用OkHttp3网络库构造一个HTTP POST请求到API服务器 /login 路由下，若成功，服务器将返回一个标示用户身份的token，客户端通过Message对象的形式将处理结果返回主线程处理相应动作的handler，若失败，则返回对应错误信息。同样，upload与fetch方法内实现了异步上传与下载用户数据库的功能。

## 上滑换壁纸功能实现

此功能在MainActivity.java中的MainActivity类的changeBackground实现

在这个方法中首先实例化Random类通过Random的nextInt(5)方法可以生成0-5的随机数

创建一个含有所有背景图片ID的Arraylist，完成随机获取图片id的功能

我们调用linearLayout的setBackgroundResource方法来将新背景替换原来的背景

最终用户上滑时，我们调用这个方法，所以我们在onTouchEvent方法加上一个新的判断条件，判断条件为y1 - y2 > 200.核心代码如下所示：

**private** void changeBackground() {  
 Random random = **new** Random();  
 int index = random.nextInt(5);  
 List<Integer> backgrounds = **new** ArrayList<>();  
 backgrounds.add(R.mipmap.background\_1);  
 backgrounds.add(R.mipmap.background\_2);  
 backgrounds.add(R.mipmap.background\_3);  
 backgrounds.add(R.mipmap.background\_4);  
 backgrounds.add(R.mipmap.background\_5);  
 linearLayout.setBackgroundResource(backgrounds.get(index));  
}

## 小讲堂功能实现

首先我们实现了讲堂类，讲堂类是一个Java Bean，讲堂类的所有属性有:标题，内容，文章数量，讲堂图片地址，每个属性具有Get和set方法。

我们实现了JsonParse类，这个类的功能是用于解析出JSON里面的内容并设置到相应的实体类中,该类核心采用Gson来编写。

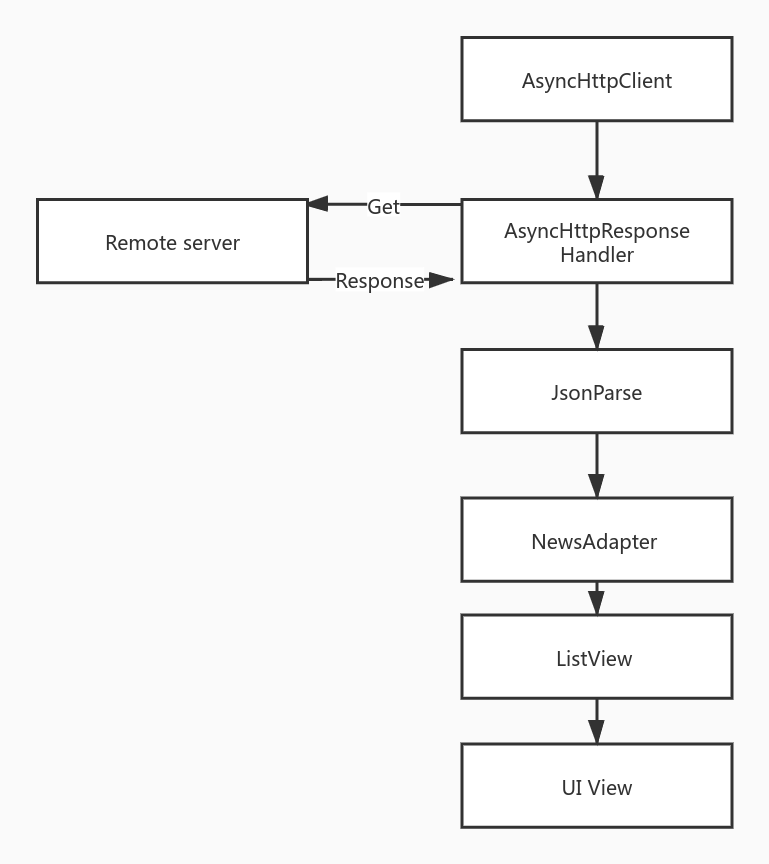
最终是NewsActivity的实现，这个类的核心是使用fileData方法,创建AsyncHttpClient对象,通过这个对象的get方法来完成json文件的获取，这个方法的参数是AsyncHeepResponseHandler类，通过这个类的onSuccess方法，我们才能真正的获取Json文件,然后交给JsonParse对象类解析文件最终交给NewsAdapter数据适配器来展示对象,核心代码如下所示：.

String json = **new** String(responseBody,"utf-8");  
newsInfos = JsonParse.getNewsInfo(json);  
lv1.setAdapter(**new** NewsAdapter());

数据适配器这个类是NewsActivitity的内部类，在这个内部类的中，首先通过getCount方法实现了获取newsInfos的数量。然后通过getView方法来将数据展示到UI界面中去，核心代码如下：

@Override  
**public** View getView(int i, View view, ViewGroup viewGroup) {  
 View v = View.inflate(NewsActivity.this,R.layout.news\_layout,**null**);  
 lTitle = v.findViewById(R.id.ltitle);  
 lContext = v.findViewById(R.id.lcontext);  
 lCount = v.findViewById(R.id.lcount);  
 smartImageView = v.findViewById(R.id.icon);  
  
 newsInfo = newsInfos.get(i);  
 lTitle.setText(newsInfo.getTitle());  
 lContext.setText(newsInfo.getContext());  
 lCount.setText(newsInfo.getCount());  
 smartImageView.setImageUrl(getString(R.string.server) + newsInfo.getIcon());  
 **return** v;  
}

数据流图如下所示：



## SQLite数据库实现

我们采用SQLite替换原来的数据库，首先我们实现了Quote和Word两个Java Bean.

其中Quote实体对应的是QuoteHelper类，在这个类中核心的功能就是getRandQuote方法.在getRandQuote方法中，我们首先打开数据库，然后实例化Random类，生成一个随机的索引，最终通过一个SQL语句来完成查找该单词.核心代码如下所示：

Random random = **new** Random();  
Cursor cursor = db.query("quote", **new** String[]{"COUNT(\*)"}, **null**, **null**, **null**, **null**, **null**);  
cursor.moveToFirst();  
int index = random.nextInt(cursor.getInt(0));  
  
cursor = db.query("quote", **null**, "`index`=?", **new** String[]{Integer.toString(index)}, **null**, **null**, **null**);  
cursor.moveToFirst();  
Quote quote = **new** Quote(cursor.getInt(0), cursor.getString(1), cursor.getString(2));

Word实体对应的是WordHelper类，在这个类中核心功能就是getRandWord方法，这个方法实现起来和getRandQuote是类似的.

## 复习本实现

复习本与错题本的实现非常相似,以复习本为例，为了实现与其他设备的学习的记录同步，并且为了复习少的单词多复习几遍,核心方法是把复习过的学习次数+1,以后复习的实现优先显示复习次数少的单词。核心代码如下所示：

word.setLast\_review(word.getLast\_review() + 1);

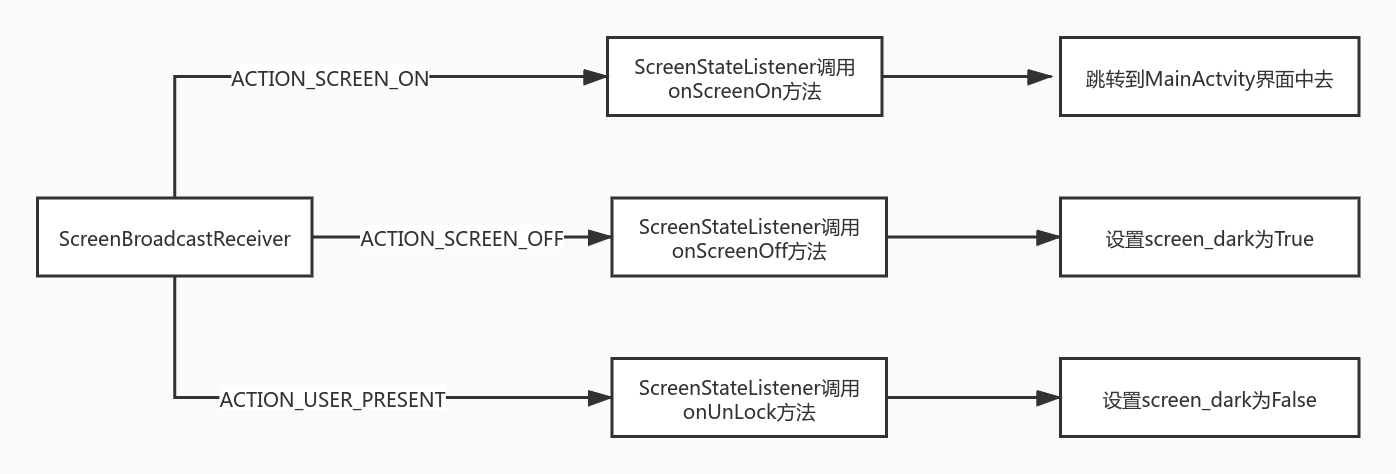
## 国际化

我们将一些固定的中文语句放在了res/values文件下，并实现了对应的英文语句，在java文件我们通过getString(R.string.server)这样的方式来获取.

## 手机解锁背单词实现

首先通过ScreenListener对象完成屏幕状态的监听，如果用户唤醒手机则会被该对象的ScreenBroadcastReceiver发现，并由ScreenStateListener对象进行相应的操作，当前屏幕是熄屏状态，那么我们就要使用intent跳转到MainActivity中去。核心代码如下所示：

screenListener = **new** ScreenListener(**this**);  
screenListener.begin(**new** ScreenStateListener() {  
 @Override  
 **public** void onScreenOn() {  
 **if** (sharedPreferences.getBoolean("btnTf", **false**)) {  
 **if** (sharedPreferences.getBoolean("screen\_dark", **false**)) {  
 Intent intent = **new** Intent(HomeActivity.this, MainActivity.class);  
 startActivity(intent);  
 }  
 }  
 }  
});  
  
**public** void onReceive(Context context, Intent intent) {  
 action = intent.getAction();  
 **if** (Intent.ACTION\_SCREEN\_ON.equals(action)) {  
 mScreenStateListener.onScreenOn();  
 } **else** **if** (Intent.ACTION\_SCREEN\_OFF.equals(action)) {  
 mScreenStateListener.onScreenOff();  
 } **else** **if** (Intent.ACTION\_USER\_PRESENT.equals(action)) {  
 mScreenStateListener.onUnLock();  
 }  
}

具体流程如下： 

# 系统测试

## 锁屏界面

点击小喇叭图标会发音。点击选项A、B、C会根据是否匹配英文单词来变换颜色。错误是红色，正确是青色；上滑屏幕会切换壁纸。

* 未选择状态



* 选择错误



* 选择正确



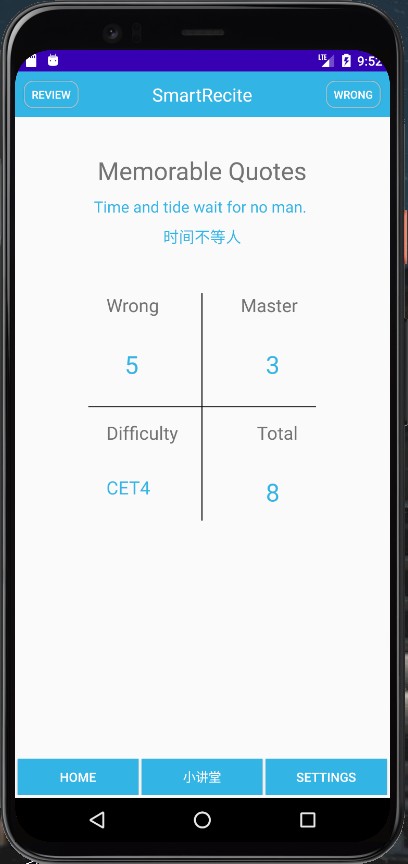
* 切换壁纸



## 主界面

左上角为复习本/REVIEW按钮，里面是回答正确的单词。右上角为错词本/WRONG按钮，里面为回答错误的单词。接着下面是名言的英语和汉语。再下面的十字框是统计框，左上角为回答错误的单词数，右上角为回答正确的单词数，左下角为选择词库的难度，右下角为总共选择的单词数目。最下面一栏分为三部分，第一个块为主界面，第二块为小讲堂。第三块为设置界面。

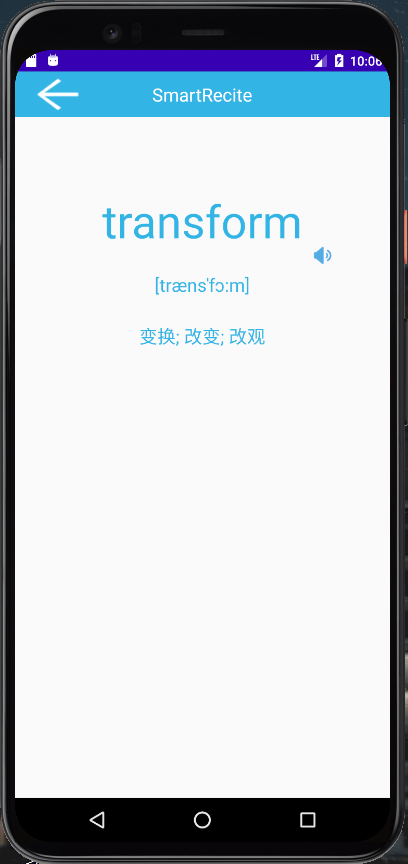
* 主界面展示



## 复习本界面

在里面可以听单词发音，单词翻译；左滑可以切换单词，达到复习单词的目的。点击左上角的返回按钮可以回到主界面。

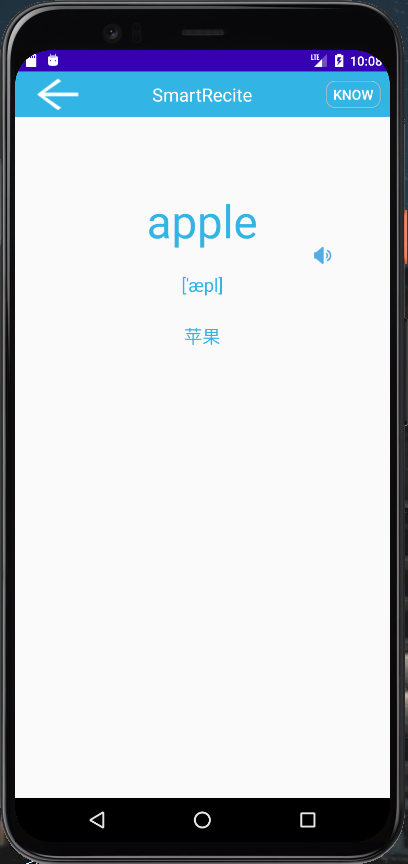
* 复习本界面展示



## 错词本界面

在里面可以听单词发音，单词翻译；左滑可以切换单词，学习之前选择错误的单词。点击右上角的 ’我会了‘/'KNOW' 可以将该单词从错词本中删除。点击左上角的返回按钮可以回到主界面。

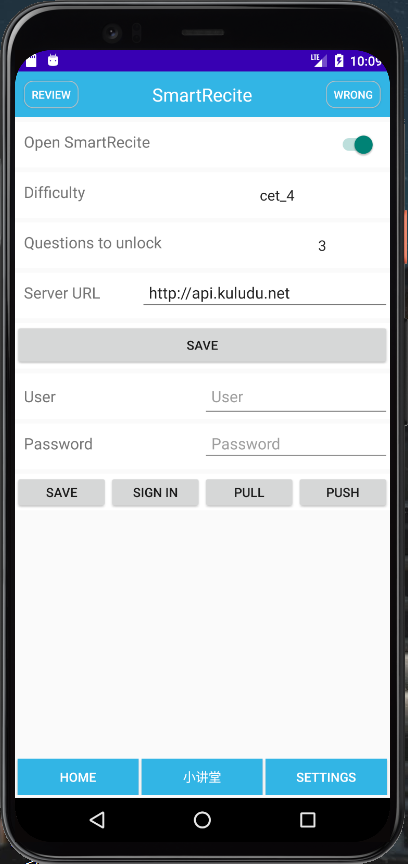
* 错词本界面展示



## 设置界面

点击'Open SmartRecite'/'开启单词锁屏'可以开或者关闭单词锁屏功能。点击’开锁单词难度‘/’Difficulty‘后的下拉框可以切换锁屏背单词的难度，有cet4和cet6两种选择。点击’需要解锁题的个数‘/’Questions to unlock‘后的下拉框可以切换锁屏背单词的难度，有3、5、9、12四种选择。再下面是服务器的地址，可以填入自己的服务器地址。接着是保存按钮。再下面是用户登录部分，可以输入用户和密码来进行登录，还可以自己之前上传的错词记录下载下来，或者将本次的错词上传到服务器。

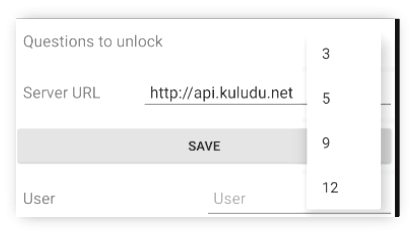
* 设置界面展示



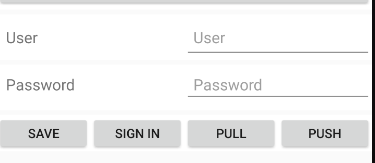
* 选择词库难度



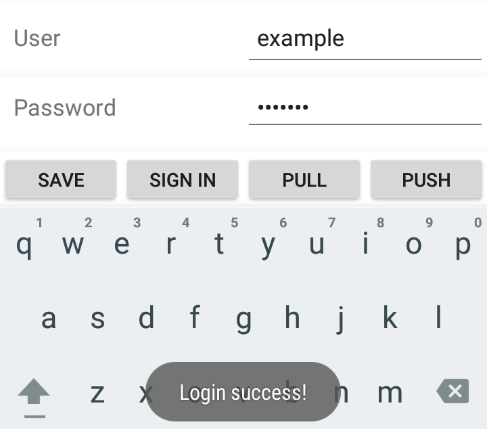
* 选择解锁题的个数



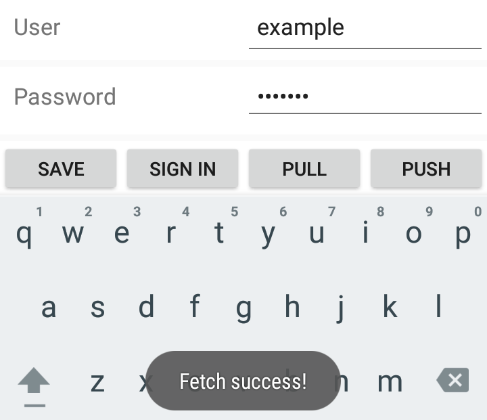
* 用户登录



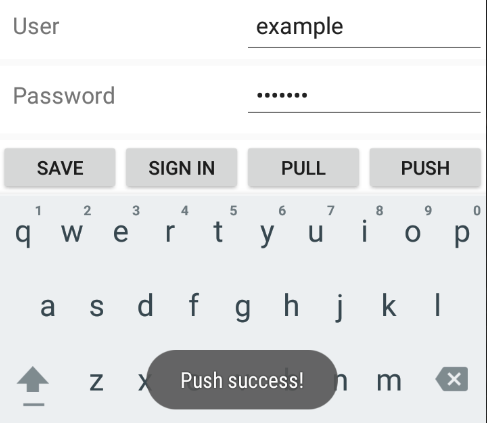
* 登录成功



* 拉取成功



* 上传成功



## 小课堂界面

用户可以在其中学习专业的英语知识和技巧来提高自身英语水平。

* 小课堂界面：

