**无道单词APP系统设计说明书**

**开发团队：**

姓名 班级 学号 自评分

孙子洪  计算机试验班71 2176112709 89.9

白思雨 计算机试验班81 2186123935 90.1

报告发布日期：2020年12月12日

1. **引言**
2. **标识**

文档标题：无道单词APP系统设计说明书

版本：1.0

1. **概述**

无道单词APP是一款运行在Android操作系统上，具备翻译单词，记录单词本和小游戏功能的单词学习软件，主要目标用户为外语学习者或日常使用外语的人员。

1. **概要设计**
2. **总体结构**

本系统根据功能可以分为三个子系统，即登陆、注册、找回密码子系统，翻译、推送单词、单词本子系统和小游戏子系统，在代码的实现上以MainActivity类为中心，各个子系统包含的功能也被定义为相应的类。

1. **接口设计**

* **用户界面**

**登陆界面**



**注册界面**



**找回密码界面**



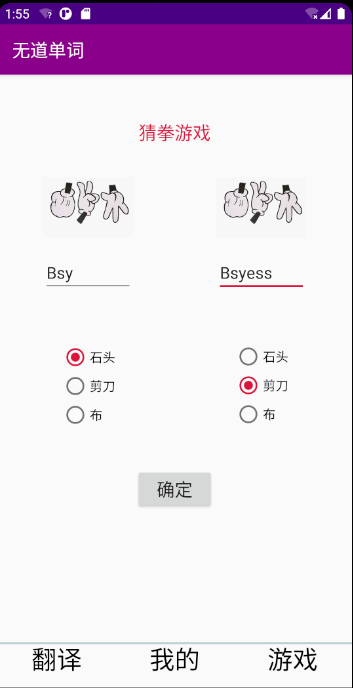
**翻译，推送单词界面**



**单词本界面**



**小游戏界面**



* **子系统接口**

**各个子系统是以Fragment的形式实现的，**

* **登陆，注册，找回密码子系统**

主体为Mine，同时Mine又调用RegisterFragment和ForgetPwdFragment，RegisterFragment实现了注册功能，ForgetPwdFragment实现了找回密码功能。

* **翻译，推送单词，单词本子系统**

主体为Mine，包括TranslateFragment和MyInfoFragment，TranslateFragment实现了翻译和推送单词功能，MyInfoFragment实现单词本功能。

* **小游戏子系统**

主体为Mine，包括GameFragment，实现了猜拳小游戏。

* **数据库接口**

数据库设计是指建立数据库及其应用系统，它是软件系统开发的核心部分。在嵌入式设备中，使用最为广泛的是SQLite数据库管理系统，因为是一款轻型数据库，而且占用非常低的系统资源，在一般的嵌入式设备中，只需要几百K的内存即可。使用SQLite数据库保存用户收藏的单词，包含DatabaseManager和DBHelper，DatabaseManager中包含了插入数据、删除数据等方法的详细定义，DBHelper则定义了数据库的相关信息和数据的类型。

SQLite数据库是嵌入式设备中应用最为广泛的数据库，它是一款轻型数据库，占用非常低的系统资源，为了实现轻量级，而放弃了高并发性、拓展复杂的功能等。总体来说，SQLite数据库非常可靠，稳定性高、数据库可移植性好、体积小、简单而且高效。Android操作系统提供SQLiteOpenHelper可以帮助用户重新创建一个数据库，用户只需要集成SQLiteOpenHelper类即可。

SQLiteOpenHelper类主要有三个方法：

1)构造函数该方法共有四个参数：Activity，数据库名称，游标工厂(通常为NULL)，数据库模型版本号；

2)onCreateO方法以SQLiteDatabase对象作为输入参数，主要用于数据初始化工作；

3)onUpgrage0方法方法共有三个参数：SQLiteDatabase对象，旧的数据库模型版本号和一个新的数据库模型版本号。

Android SQLite数据库来存储和管理数据，因此在本文设计研发的在线翻译软件中，需要使用数据库存储离线数据，因此主要的数据库表为生词本表、历史记录等等。生词本表用于保存并添加离线单词的各项属性信息，包括：新词汇ID、新词汇、新词汇释义。

生词本表：

|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 说明 |
| \_id Integer | Int | 新词汇id |
| Username text | String | 所属用户 |
| Word text | String | 单词 |
| Translation text | String | 译文 |

用户信息表：

|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 说明 |
| Username text | String | 用户名 |
| Password | String | 用户密码 |
| Email | String | 用户邮箱 |

* **通信接口**

本项目在设计在线翻译软件体系结构采用是C/S结构，即用户可以通过Android手机等客户端接入翻译接口，交给远程服务器来实现单词的在线翻译，在代码上是使用百度的翻译API来实现的。具体工作流程是直接将数据由客户端接入交给服务器进行工作，一旦服务器响应了客户端请求，那么服务器也就根据预定规则给客户端请求做出应答，返回结果。这样应用服务器运行数据负荷较轻。因此，本系统将翻译程序安装在查询网络服务器端，充分发挥服务器处理能力。当用户在手机应用程序客户端输入单词信息，客户端会进行初步数据处理，然后接入到服务器端，服务器端会根据请求进行实际的翻译工作。客户端和服务器端各司其职发挥效用，加快了客户端的响应速度，

1. **全局数据说明**

* **数据**
* Data：当前登陆的用户，包括用户名
* User：注册的用户，包括用户名，密码和邮箱
* Word：收藏的单词，包括序号，用户名，单词和翻译
* TranslateResult：翻译的参数和结果，包括源、目标语言，单词和翻译
* ResultPair：翻译结果，包括单词和翻译
* MailInfo：采用邮箱注册时的配置信息
* **方法**
* MailSender：根据MailInfo中配置的信息发送邮件
* md5Password：使用md5算法对密码加密

1. **运行行为**

打开APP后首先出现的是翻译界面，默认是以游客身份登陆的，如果想要使用单词本服务，就需要进行登陆，如果没有账号，需要输入用户名，邮箱和密码进行注册。在忘记密码时，还可以输入自己的用户名和邮箱重新设置密码。

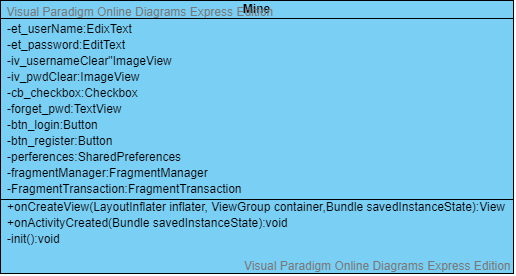
在翻译界面，可以选择源语言和目标语言（目前只支持中文和英文互译），输入待翻译文本进行翻译。如果已经登陆，还可以点击收藏按钮将单词进行收藏，点击推送单词按钮查看系统推送的单词和翻译。

在小游戏界面，可以进行猜拳游戏，输入自己的名称并在复选框内选择石头、剪刀或布来进行PK。

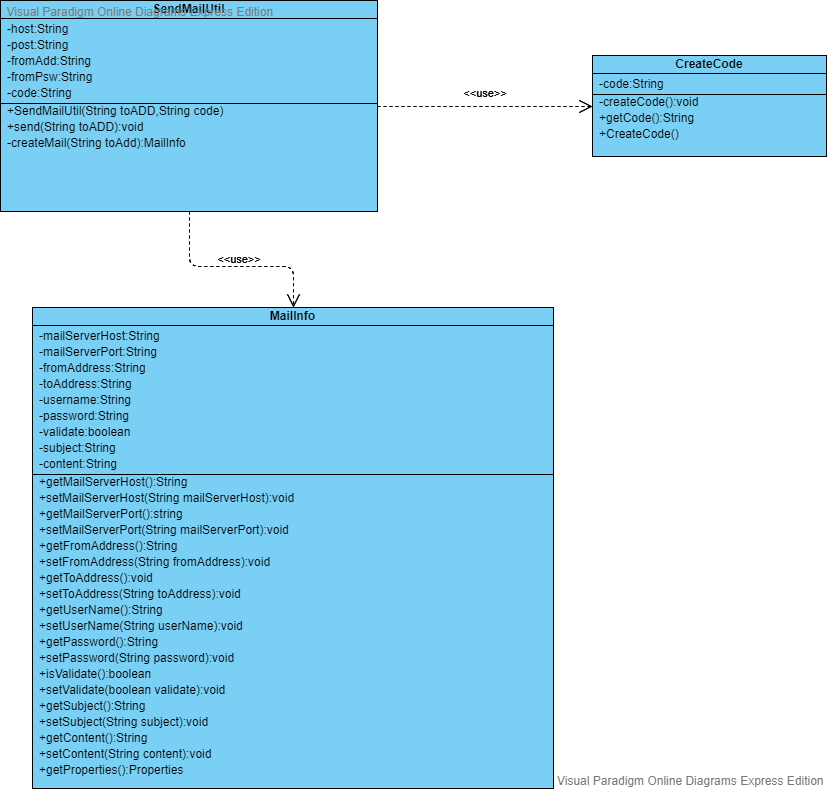
1. **详细设计**

* **登陆，注册，找回密码子系统**
* **设计类图**

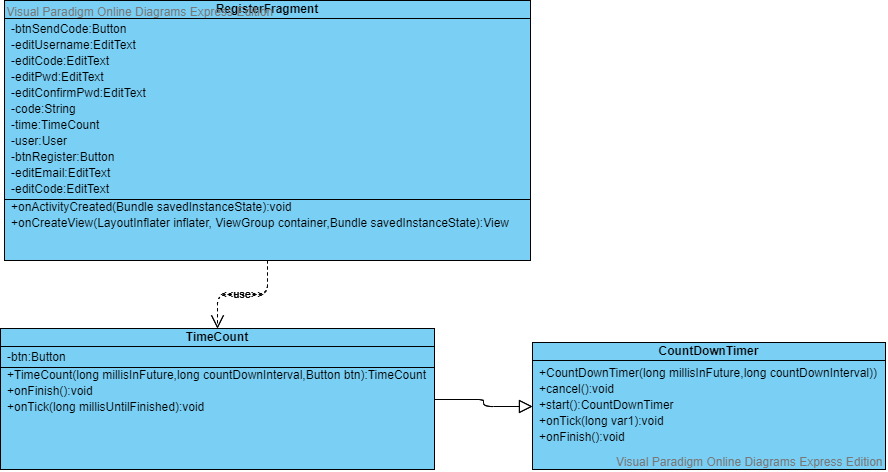
下图中是设计的是用户类，对用户的前端输入进行合法检测以及对系统的返回结果进行输出。用户属性中主要有用户名，用户密码，用户邮箱以及用户的单词本这一数据库文件。其中用户对应的方法有注册按钮，修改密码按钮，收藏单词按钮，进入单词本按钮，移除单词按钮，以及发送验证码按钮等。



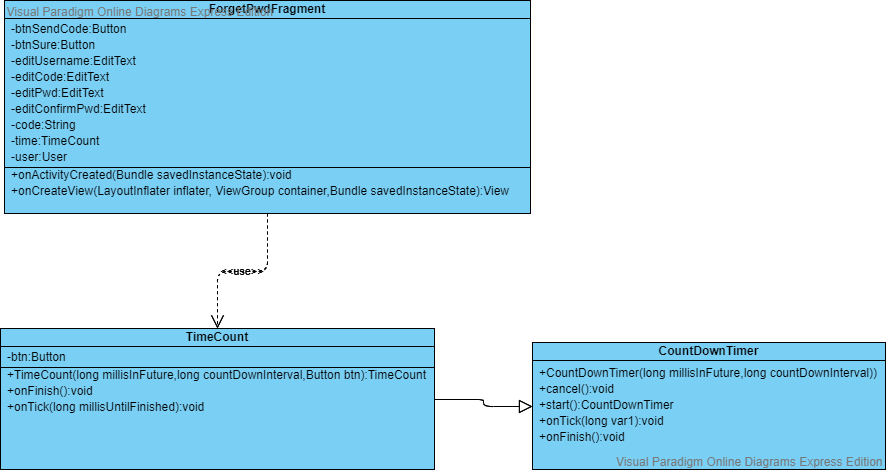
下图是用户注册，修改密码时的邮箱验证功能。该类主要使用两个自定义类：CreateCode为用户当前操作创建一个验证码，MailInfo则向App返回对应的邮件信息，便于进行验证码的比对。邮箱验证功能用户无法直接访问，只能通过Mine类进行调度，主要有对用户邮箱发送验证码，对用户输入验证码的核查，创建邮件，返回邮件信息等功能。



下图为我们自定义的注册类，实现用户的注册功能，主要包含验证码，用户名，密码，倒计时，密码确认等属性，与之对应的操作有发送验证码按钮，注册按钮等。该类主要使用了TimeCount类，TimeCount是对java自带CountDownTimer抽象类的实现，负责用户发送验证码后的倒计时功能。

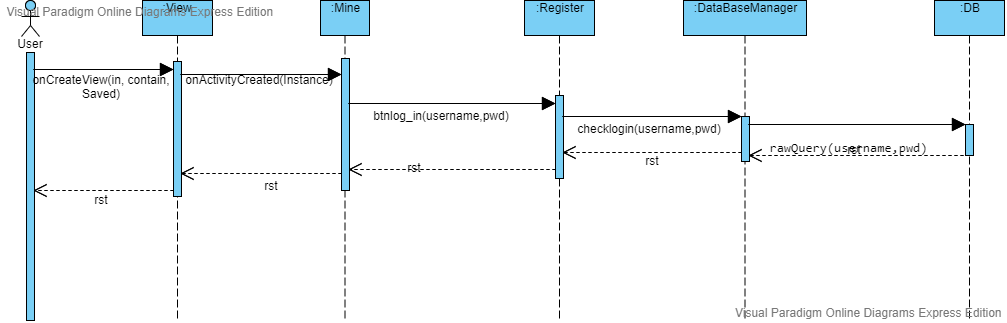


下图为我们自定义的修改密码类，实现用户的忘记密码时修改密码的功能，主要包含验证码，用户名，密码，倒计时，密码确认等属性，与之对应的操作有发送验证码按钮，确认修改按钮等。该类主要使用了TimeCount类，TimeCount是对java自带CountDownTimer抽象类的实现，负责用户发送验证码后的倒计时功能。

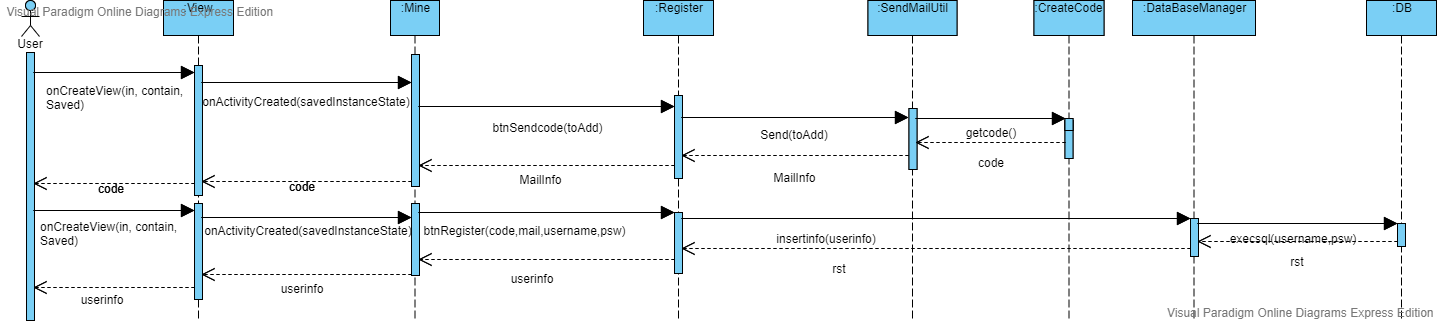


* **顺序图**

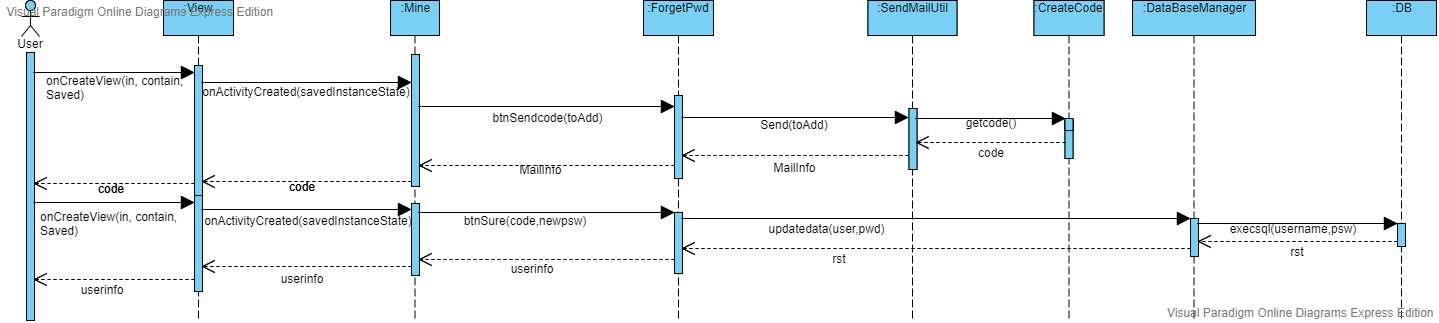
登陆：

****

注册：

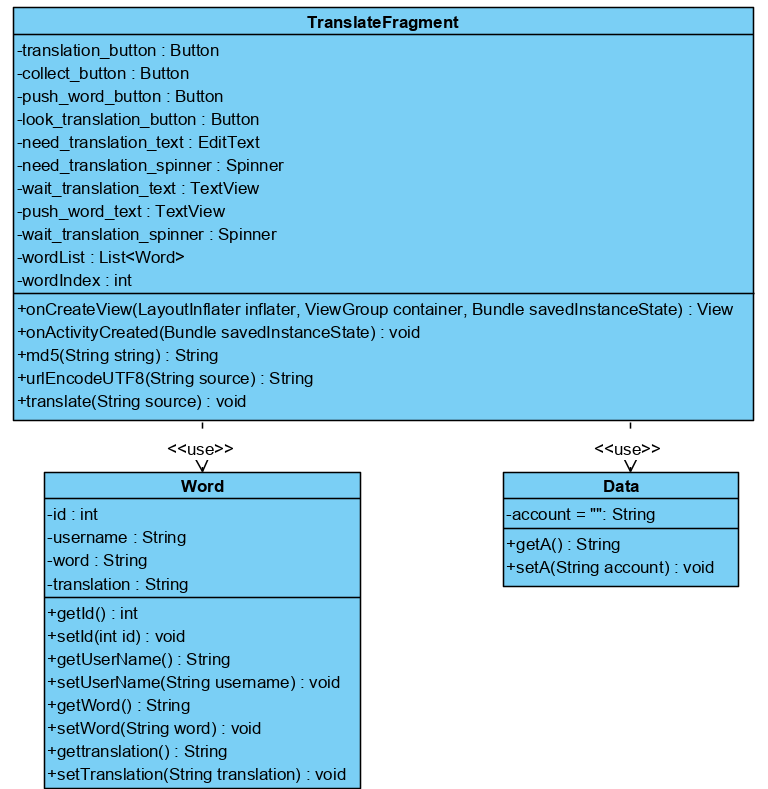
****

找回密码：

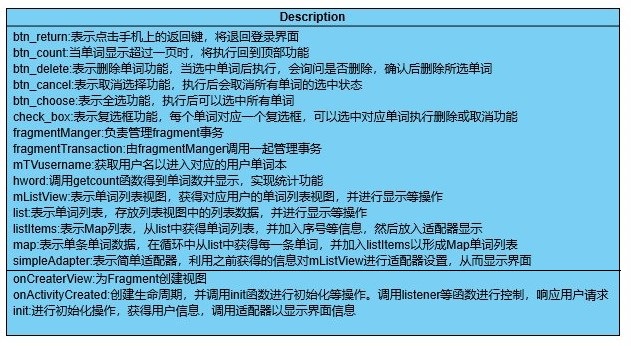
****

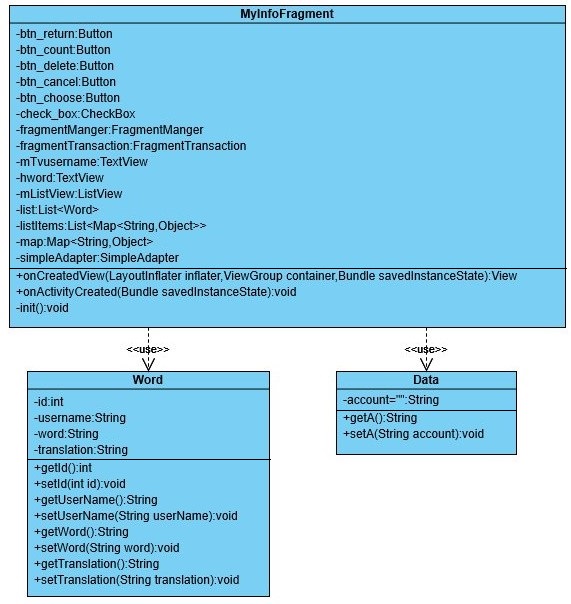
* **翻译，推送单词，单词本子系统**
* **设计类图**

翻译，收藏，推送单词：



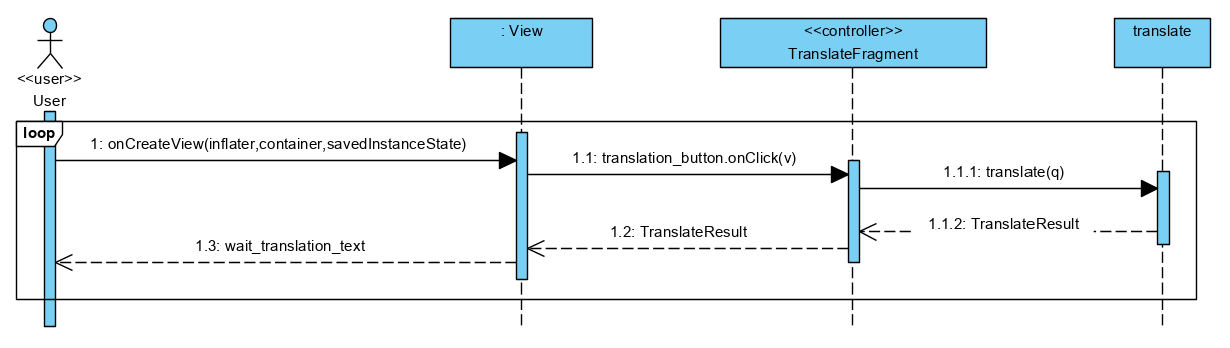
单词本：



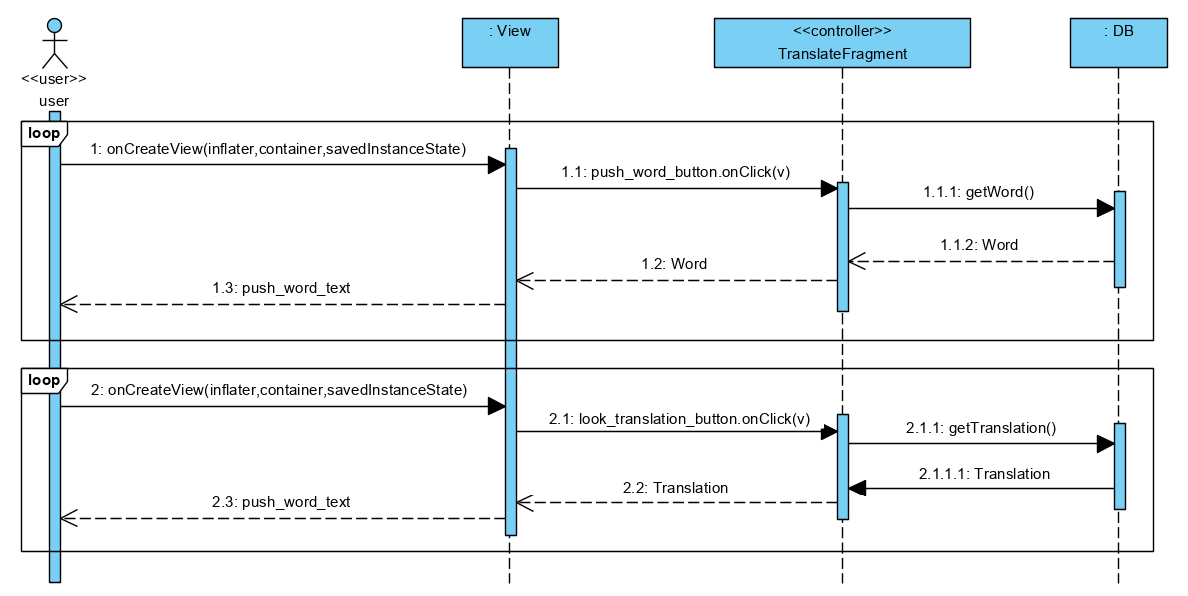


* **顺序图**

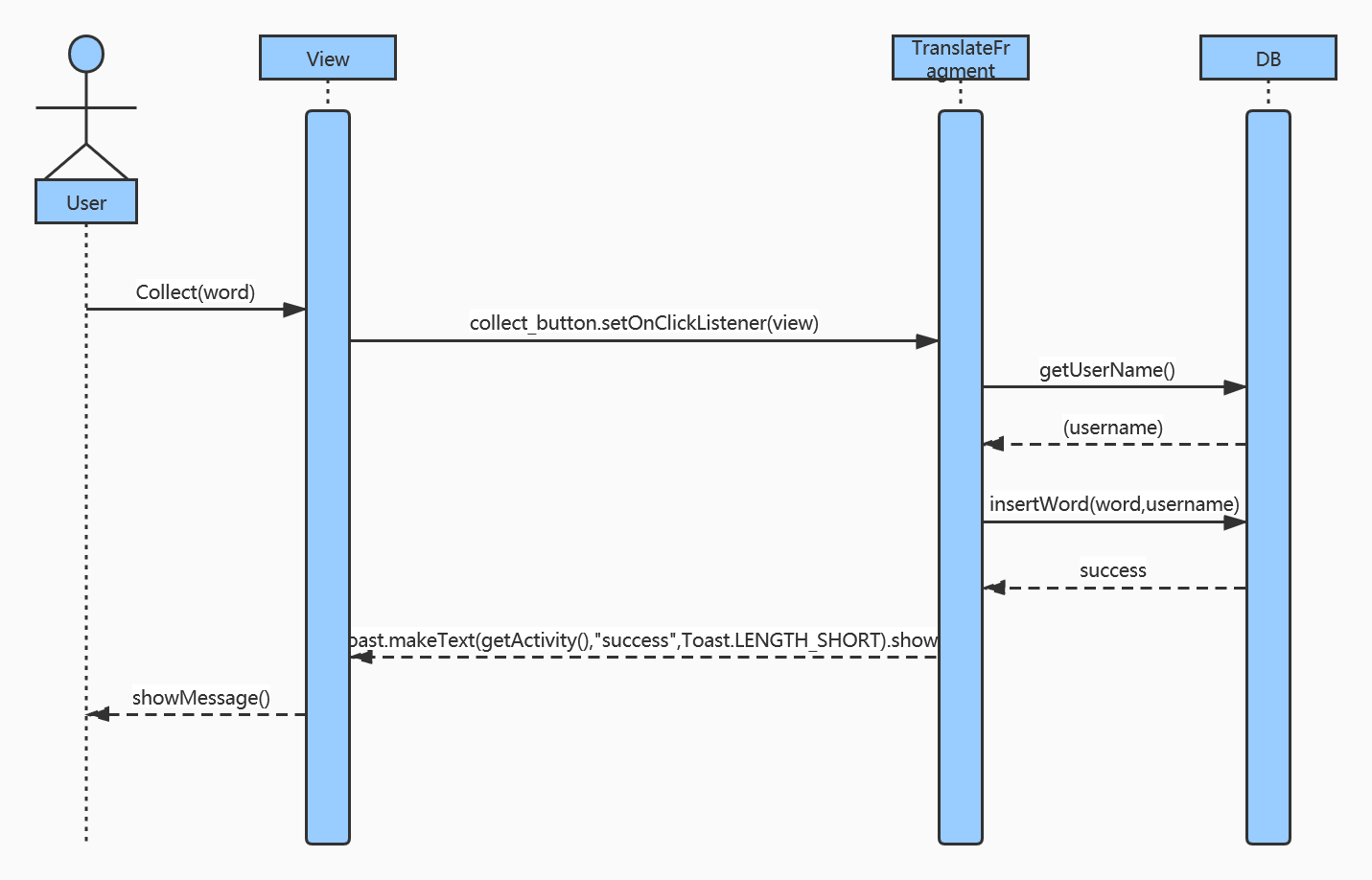
翻译：



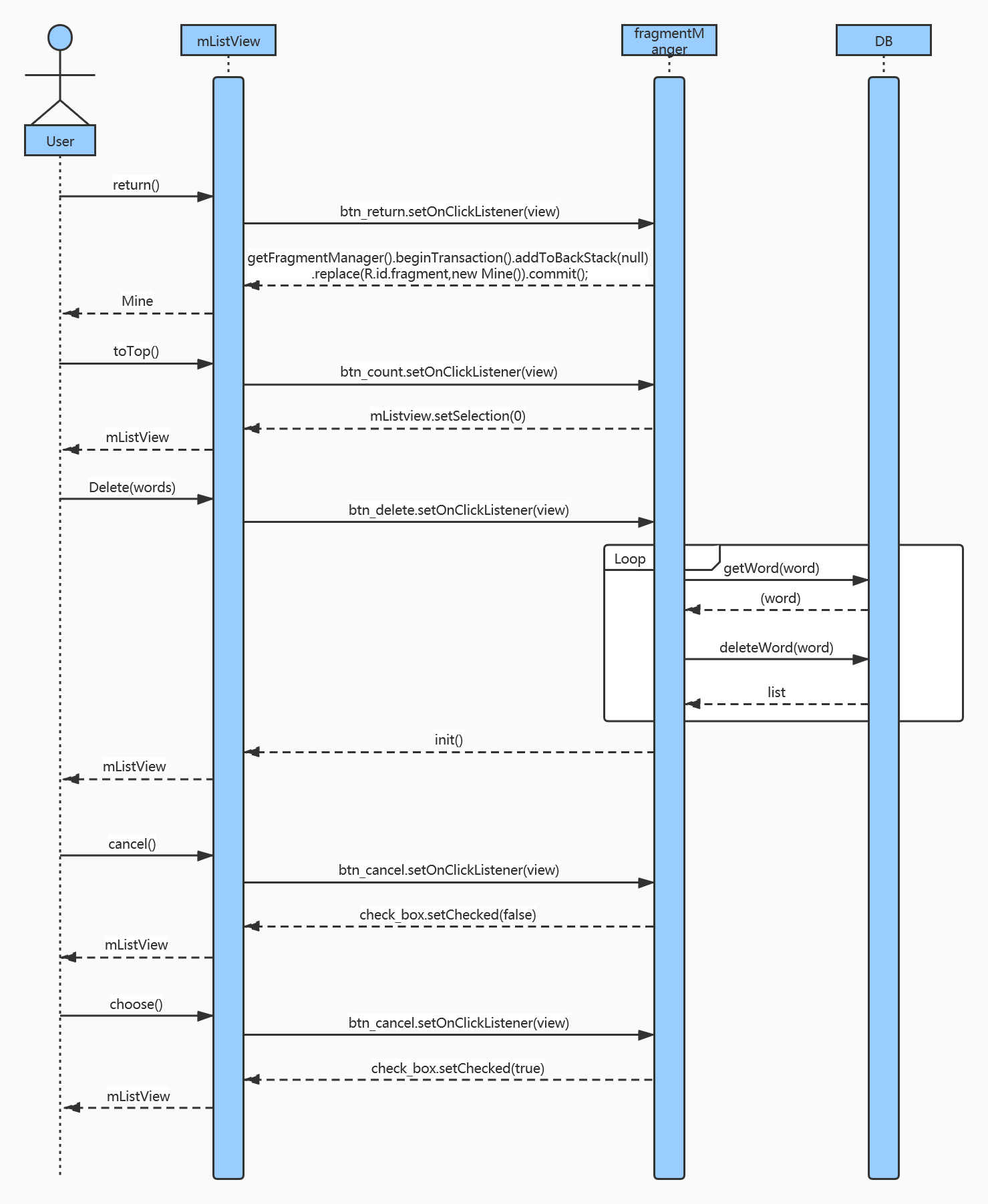
推送单词：



收藏单词：



单词本：



* **小游戏子系统**
* **模块与类的详细设计**

这一部分实现的是小游戏功能。小游戏为猜拳游戏。在APP页面上，游戏功能位于底部栏的第三栏，点击后直接进入游戏。设计上，主要包括MainActivity.java、GameFragment.java和Caiquan\_game.xml，并引用放在资源区的图片。MainActivity.java用来在用户选择游戏栏时开启游戏功能。GameFragment.java是游戏实现的主要代码。Caiquan\_game.xml是游戏页面的主要UI设计，并引用资源区的资源。

* **类属性与功能详细说明**

猜拳游戏只需要用户输入用户名，并选择所要出的选项（石头，剪刀，布），点击确认，通过与电脑随机产生的石头剪刀布的其中一种进行比较，产生游戏结果（胜负平）。

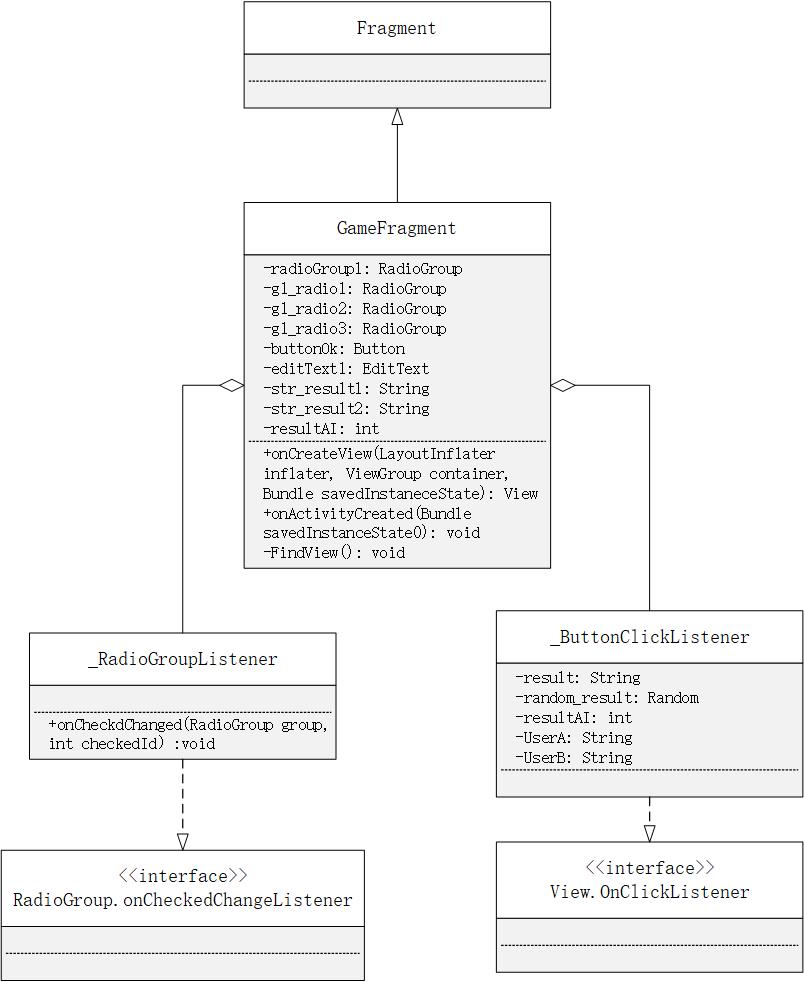
在代码中，需要变量来储存用户名UserA: string，用户选择的结果str\_result1: string，产生的随机数random\_result: int，由随机数产生电脑的出拳结果str\_string2: string，游戏结果result: string以及弹出的语句winner: string。

用户结果的产生通过监听（onCheckedChanged()）用户选择了哪个出拳所对应的按钮得出；对于电脑结果，是通过产生1到3的随机数，再一一对应到石头剪刀布产生的，并且由于每局结束是以用户按下确认键为标志，因此需要在产生默认游戏视图时与监听到用户按下确认键后均要产生随机的电脑猜拳结果。

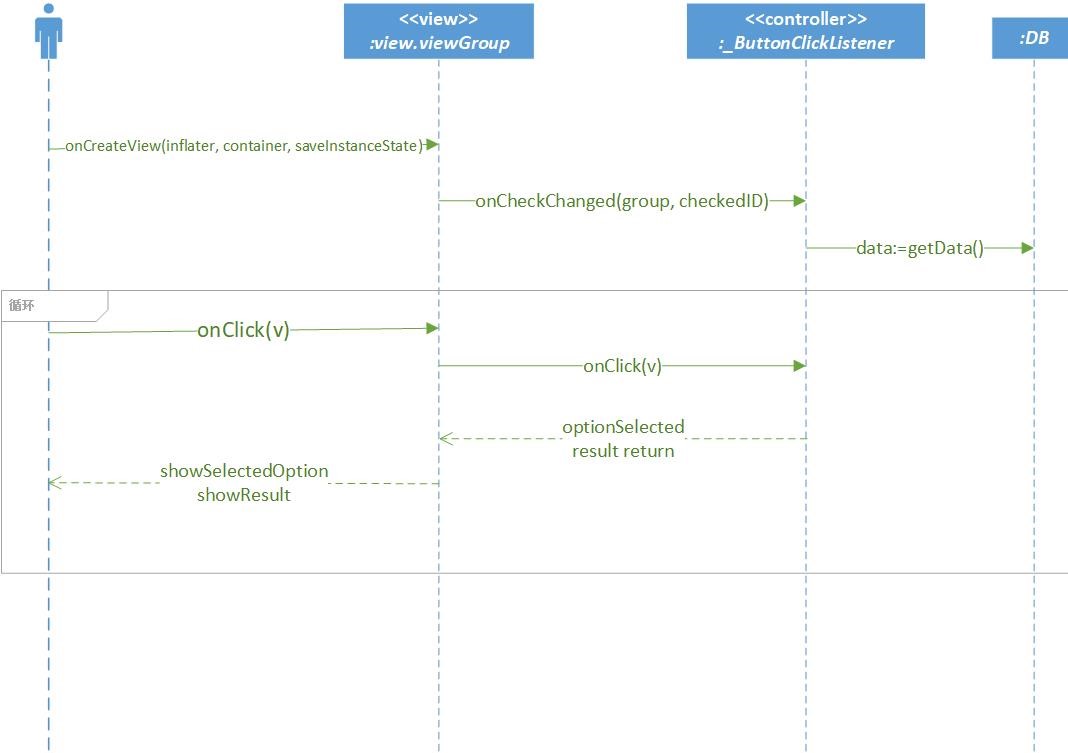
对于结果，需要通过逐个比较确认是九种情况中的哪一种，产生结果（即赢家）存入result。

通过result中是否含A和B（用户是UserA，电脑是UserB）来判定结果，含A，那么result就是UserA（玩家）；含B，那就是UserB（电脑）；都没有（此时存的结果是字符串平局），就是平局。产生提示用户语句winner，作为输出的对象。

* **设计类图**



* **顺序图**

****

1. **需求的可追踪性**

在APP的运行过程中，最主要升级维护的就是项目维护，修复Bug与追踪用户需求。在项目管理与维护方面，在设计应用程序框架时将采用MVC(Model,View-Controller)模式，MVC是模型、视图和控制器的缩写，MVC模式是一种强制性的使应用程序的输入、处理和输出分开的设计模式。最初是于二十世纪八十年代产生的设计模式，至今已经被广泛应用于软件程序设计中使用模式，它是用来分离视图层和业务层，不用重新编译模型代码及控制器代码就可以更改视图层代码。在项目管理方面，我们选择Devcloud这一开源项目仓库，可以很方便的进行版本的升级与回退，例如针对用户需求发放升级补丁，对某一个不理想的升级进行回滚操作。同时，对于我们的用户，我们会定期发送调查问卷及升级预览等信息，以便用户对我们的设计思路有一个清晰的认识，使我们可以及时收到反馈，从中筛选出有价值的信息进行下一步开发。同时拟在后续版本更新中增加反馈界面，同时对有意义的反馈进行主题解锁，单词上限增加，解锁词典等奖励，激发用户反馈的积极性，强化对需求的可追踪性。