# 论文第三版

### 数据库设计

本app采用了SQLite数据库，SQLite不依赖第三方软件，也无需安装，他所有的数据库信息都存储在一个文件内，但为了开发方便，使用了python中继承的sqlite库，同样也无需安装，

为什么使用SQLite：

1. SQLite无需配置复杂的连接驱动，连接数据库只需要sqlite3.connect(“数据库地址”)，就像是打开了一个文件
2. SQLite 支持常规sql语句，只需要sqlite3.connect(“数据库地址”).execute(“sql语句”)，就能实现对表的增删改查

为了保持数据的持久性，是用户登陆之后就能查询到自己之前最近一次的检测记录，故使用了一张data表来保存用户的测试数据，为了保证数据的唯一性，故又采用id与eye作为联合主键的方式来保证每个人的每只眼睛都是唯一存在的，data中的字段包括id,eye,sightingLoseRatio（固视丢失率），falseNegativeRatio(假阴性率)，falsePositiveRatio（假阳性率），location1，，，location72(72个位点的视野阈值情况)

为了数据的准确性，确保每个人获得的测试数据都是自己的，故使用了一个user表来进行id与passWord的校验，id作为唯一主键，user中的字段包括id,passWord

每个id在user表中对应一条数据，在data表中对应两条数据

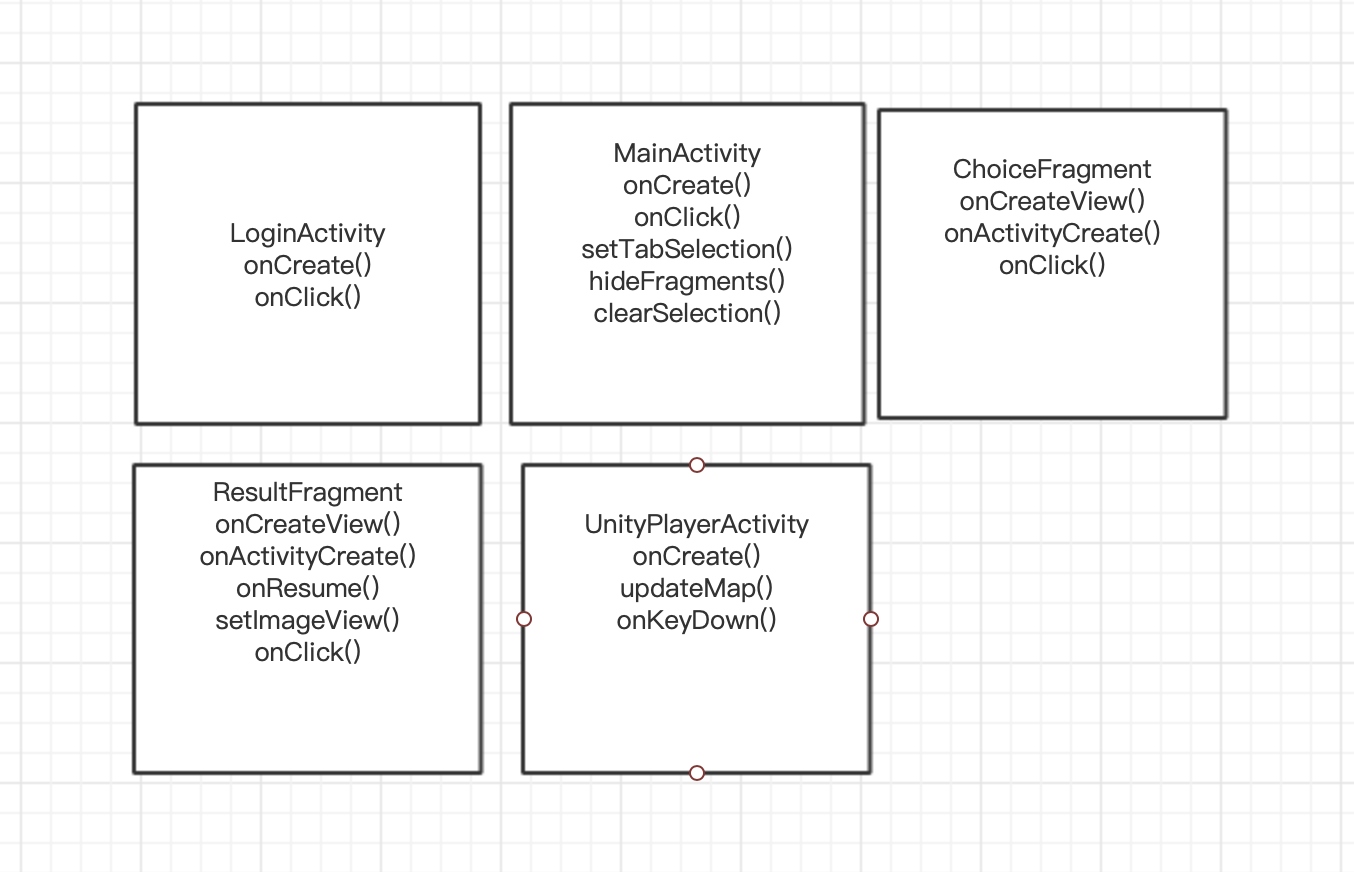
# 详细设计

## 视图模块

### 功能描述

该模块主要用于将所有处理好的结果进行展示，以及帮助用户完成测试

### 类图设计



### 关键流程设计

打开app首先会进入到LoginActivity中的onCreate(),调用setContentView()方法加载activity\_login.xml,当点击按钮时会进入到onClick中，如果校验不通过，则会显示相应提示

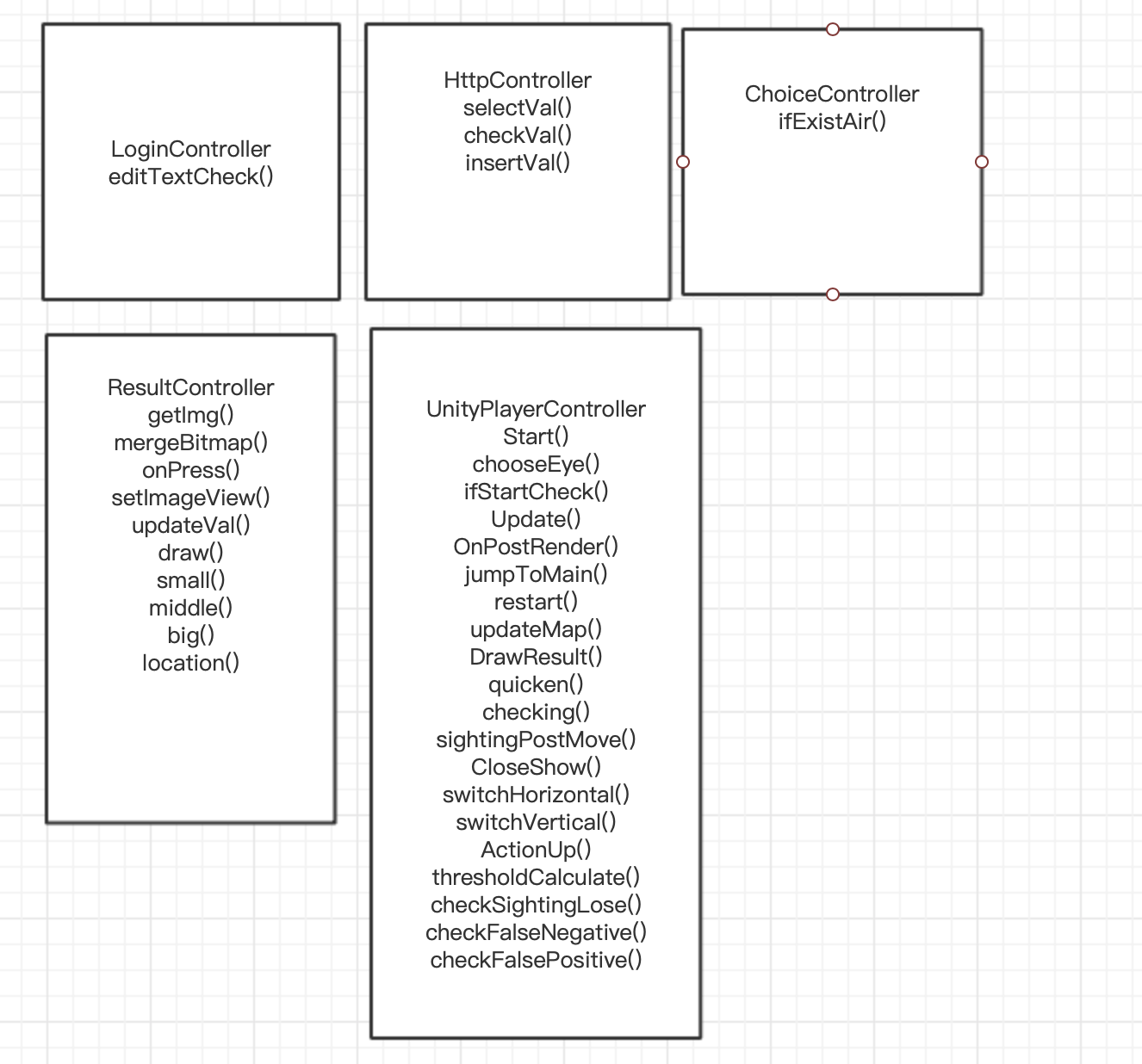
，通过校验后，会跳转到MainActivity,之后进入到MainActiviy中的onCreate(),调用setContentView()方法加载activity\_main.xml,当选中tab时会进入到onClick（）中，这时会根据选中的tab调用setTabSelection进入到不同的Fragment中，当进入到ChoiceFragment中时，会调用onCreateView()方法加载fragment\_chice.xml,当点击按钮时会进入到onClick（）中，通过校验后，会跳转到UnityPlayerActivity,之后进入到UnityPlayerActivity中的onCreate()方法加载UnityPlayer,当进入到ResultFragment中时，会调用onCreateView()方法加载fragment\_result.xml,之后每次重新进入到ResultFragment中时，都会调用onResume(),当点进按钮时会进入到onClick（）中，选择好跳转的app后，会跳转到其他app的当前Activity中

## 业务逻辑模块

### 功能描述

该模块主要用于处理数据以及响应监听，发送接收服务器请求

### 类图设计



### 关键流程设计

打开app,当填写完用户id和密码后点击登陆按钮时，会进入到onClick,首先调用LoginController中的editTextCheck（）方法校验用户id时候合规，如果合规，则会调用httpController中的checkVal()方法，携带用户id和密码向服务其发送一条校验请求，如果服务器返回true，则校验通过，跳转到下一个Activity，如果服务器返回false，则校验不通过，直接return，如果不合规，则直接return

当在choiceFragment选择测试眼睛后，会首先调用choiceController中的ifExistAir()方法来判断控制器是否连接，如果没有连接，直接return，如果连接，跳转到下一个Activity

当每次选择resultFragment后，会首先进入到onResume()方法，在onResume中调用httpController中的selectVal()方法，携带用户id和eye向服务器发送一条查询请求，如果获取到的数据不为空，则更新全局变量中的值，如果获取到的数据为空，会判断全局变量中有没有值，如果没有，则附一个默认值，////////////////没写完////////////