项目技术选型与技术原型指南

技术选型与技术原型开发的评估是一项需要丰富行业经验的工作，该工作质量将直接影响后续开发的进程。

1. 技术选型表

项目	互联网电影售票系统	备注
1.终端支持	Mobile app	
1.1 开发语言框架	Java	
1.3 传感器	GPS,距离	
2 服务器支持		
2.1 语言	Node.js	
2.2Web 框架	无	
2.3ORM 框架	无	
2.4 关系数据库	MySQL	
2.5 数据缓存（非关系）	Redis	
2.6 负载均衡机制	Nginx	
2.7 消息中间件	无	
2.8 其他第三方组件	百度地图 API	
3 开发平台与工具		

3.1IDE	Android Studio	
3.3 源代码管理	Github	

2. 技术原型开发内容

本部分对项目涉及的关键技术做一些前期研究，Java 是最好的开发语言。

2.1 项目技术风险元素

1.获取电影院及豆瓣评分相关信息
2.用户信息及订、退票信息存储
3.支付跳转
4.电影院周边推荐及路线选择

2.2 验证性的程序开发方案或技术原理

在影院购票系统中，每个功能模块的 Activity 类继承自 Activity 基类，实例化为一个独立的类，类中定义了各层视图控件，作为用户接口来响应用户触发的事件，实现影院购票系统的各个功能。影院购票系统中所有的 Activity 都必在 AndroidMainfest.xml 中注册。

（一）注册功能实现

新用户注册部分的应用层需通过如下方法进行注册：

#define Activity_Regist “Regist”

注册模块的业务层要为应用层提供一个与用户交互的图形界面服务，该服务注册如下：

#define Business_RegistCustomer “RC”

该模块主要是实现提取用户输入的注册信息并将该信息发送到服务器端进行注册的功能，期间涉及注册信息的录入，为此在业务层中设置了 7 个服务项目：新用户名输入框，密码输入框，密码确认框，手机号码输入框，电子邮件输入框以及注册确认按钮和退出系统按钮控件。

应用 findViewById()方法完成确认按钮服务的调用，并将此服务添加到系统程序的按钮变量中。在业务层还需要为按钮设置一个监听器，这个监听器的作业是监听用户的点击事件，当用户按下该按钮后就会发起向服务器发送用户注册信息的操作。

由于注册模块需要向服务器发送用户注册信息，应用程序与协议栈的交互是必不可少的，因此适配器必须要与协议栈进行绑定，如此，用户启动注册功能时，担负与协议栈交互任务的适配器即被创建和绑定至注册业务。

（二）选择观看电影功能实现

该模块存在一个问题，即在跳转到该页面时如何才能知道用户选择的是哪一家影院，因为只有确定用户选择的电影院名字才能在数据库中找到该影院当日放映电影信息。要解决这个问题就必须在签约影院模块的业务层中添加一个能够记录用户选择电影院信息并将信息传递到下一个模块服务。该服务使用 onListItemClick()方法实现了对用户点击电影院时间的响应。其中该方法的第一个参数是 ListView 类型的影院列表。通过 getSelectedItem()方法从电影院列表中获取用户选中的电影院名字，将它的值赋给 String 类型的变量 str。然后定义一个 Intent 变量，使用 setData()方法将该电影院名字保存到 Intent 变量中，再调用 setClass()方法将电影院名字传递到下一个界面。

获得用户选择的电影院名字后，下一步工作就是从客户端数据库中获得该影院当日放映电影的信息，因此在业务层还需要再添加一个获取放映电影信息的服务。该服务通过生成一个 DatabaseHelper 助手类的实例获得客户端数据库的使用权，并以上个模块传递来的电影院名称为查询条件，调用 query()方法遍历数据库，从而查找到用户选择电影院放映电影的信息，把电影信息存放在变量中。

（三）观影座位选择功能实现

该模块的功能是为用户显示选择的放映厅座位预定情况，方便用户根据个人喜好选择座位，为此需在业务层注册一个显示服务的界面方便应用层的调用。服务注册如下：

#define Business_FilmHouseSeatShow “FHSS”

在 filmhouseseat.xml 文件中定义了每个控件的具体位置，应用层调用 setContentView()方法加载 filmhouseseat.xml 后即可实现放映厅座位显示界面服务。

获得用户选择的电影院名字，电影名字和影厅信息后，接下来的工作就是根据这些信息从客户端数据库中获得放映厅座位分布情况，在业务层我们还需要注册一个获取放映厅座位信息的服务。通过生成一个 DatabaseHelper 助手类的实例获得查询客户端数据库的权限，以获得放映厅信息为查询条件，调用 query()方法查找数据库中存储的数据，查找到放映厅的座位的信息后，将这些信息保存到变量 filemhouseseat 中。在业务层还必须在注册一个显示放映厅座位信息的服务，该服务在业务层的注册方法是：

#define Business_DiapiayFilmHouseSeat “DFilmHouseSeat”

该服务在构建放映厅座位视图时，采用了表格布局方式，按照行列来组织座位的布局。行采用 TableRow 对象来定义，每行可以包含数个单元格，每个单元格又可以设置一个 View 对象，View 对象可以通过调用 setColer()方法为自己设置颜色，通过显示不同颜色用户就可以了解座位的售卖情况。

（四）支付功能实现：

用户登入支付账号后，进入影票支付。该模块主要功能是将用户前期选择电影票的相关信息通过终端界面反馈给用户，用户确认无误，即可点击确定按钮完

成购票操作；若用户要修改信息或放弃此次购票操作，点击返回按钮，退至主界面。

（五）已购票务信息查询功能实现：

该模块主要实现用户对已购电影票信息的查询界面，因此要注册一个为用户提供查看已购电影票信息的界面服务，这个服务注册如下：

#define Business_CustomerTicketInfoShow　“CTIS”

我们在 customerticketinfo.xml 文件中为每一个控件定义了具体位置，这样应用层调用 setContentView()方法即可加载该文件，实现用户购票信息查询界面服务。

用户已购电影票信息主要是从客户端数据库中获取的，这就需要与本地的 SQlite 数据库进行交互，因此在业务层我们还需要注册一个获取用户已购电影票信息的服务，该服务注册如下：

#define Business_CustomeTicketInfoTable　“CTicketInfoTable”

同样，我们是通过生成一个 DatabaseHelper 助手类的实例获得查询客户端数据库的权限，以用户登录时的用户名为查询条件，调用 query()方法对本地数据库进行查询，当查找到用户已购电影票信息后，将其保存到变量 custometicketinfo 中。查找到用户已购电影票信息后，我们还必须在业务层再注册一个显示用户已购电影票信息的服务，定义一个 Arraylist 类型的实例 CustomerTicketList，将信息保存到 CustomerTicketList 中，这样应用层只需调用 show()方法就可以把用户已购影票信息显示在客户端屏幕上，完成业务层的交互。