**项目总结报告**

日期：2018.09.09

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 19 | 项目名称 | 校园巴士APP - SJTUBUS |
| 编程语言 | Java、Javascript | 开发平台和框架 | Intellij IDEA、Android Studio、Spring Framework、React |

|  |  |
| --- | --- |
| **软件需求特性** | |
| 1.是否实现了项目立项时的所有需求？列出实现的新增需求和未实现的需求。  基本实现所有立项时需求。  新增需求：   * 增加统计功能：预约信息 & 已发车次 * 已预约用户设置系统提醒发车时间 * 用户预约时设置特殊需求 * 用户收藏班次，快速预约收藏班次 * 生成二维码 扫码以确认是否预约 * 管理员在手机端可录入发车信息 * 实时检测手机网络状态变化，给用户做出提示   未实现需求：   * 虚拟支付功能  1. 采用哪种架构风格？哪些设计模式？   架构风格:   * 采用客户／服务器风格实现管理端、客户端与后端的分离； * 通信API采用RESTful架构风格； * 后端、客户端编码中采用面向对象风格； * 后端编码中采用层次系统风格进行系统设计；   设计模式:   * 使用单例模式管理Android客户端Application对象、UserManager对象与RetrofitClient对象 * 使用适配器模式完成Android客户端视图与数据的绑定 * 使用了观察者模式来完成用户状态与网络状态的通信Android客户端 * 使用了代理模式来动态完成网络请求   3.技术方案有哪些亮点？   * 客户端使用Android原生SDK进行开发，与用户的交互更加流畅，更加可操控Android系统如随时监测网络状态变化、设置日历提醒发车等。 * 使用了Cookies持久化解决方案、推出应用后下次自动登陆 * 提供了APP端的Jaccount认证服务功能，较大用户无需注册即可使用 * 使用SpringSecurity保护后台管理系统，禁止跨域请求，提高安全性 * 使用了BaiduMap SDK来完成功能   4.是否做了单元测试？是否做了系统功能测试？是否做了性能测试？是否做了兼容性等其他非功能测试？  完成单元测试、完成系统功能测试、完成性能测试  易用性测试：寻找目标用户安装产品搜集改进意见，通过增加提示、用户指南等功能增加易用性  兼容性测试：通过在不同版本的Android手机上安装，查看有无异常 | |
| **项目组成员对项目的贡献度（%）** | |
| 姚子航：25%，王鑫伟：25%，励颖：25%， 王一林：25% | |
| **软件度量** | |
| 软件代码行数（不包括注解行、空行和复用代码）： | 19454 |
| 复用他人代码行数： | 612 |
| 类的个数： | 228 |

|  |
| --- |
| **经验、教训和建议** |
| 经验、教训和建议：  应当在充分地、深入地学习了开发所要使用的技术之后，再进入开发工作，一步一个脚印，不可急于求成；  代码应做到可重用性强，以免浪费不必要的时间在写重复的代码上。  适当选用优秀的框架、类库和工具辅助开发工作，能有效地减少开发的工作量，并提升代码的可维护性；  在协作开发时应选取并严格遵循良好的git workflow，以防版本控制混乱。  开发过程中对同一个问题可以有多种解决方案，应该尝试之后再进行决定，避免将来改变方案带来的返工；  对于项目中需要较长时间审核的账号或Key等开发需要的凭证，应当尽早着手准备，避免影响开发进度如jaccount账号；  应将系统的测试任务安排在整个项目周期中，完成部分功能后便进行相应测试，做到测试驱动的开发。 |