**迭代计划（简化版）**

　　　　　　　　　　　　　制定日期：2020.9.14

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 11 | 项目名称 | WowTime时间管理app |
| 迭代名称 | 迭代一 | 计划起止日期 | 2020.9.14 - 2020.10.4 |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 产品调研，统计得到用户需求的调研结果 | 9.14 | 所有人 | | 2 | 分析和定义需求，统计梳理前后端所有需求，前端所需页面，书写初步的《Vision文档》、use-case模型、《软件需求规约》 | 9.15 | 所有人 | | 3 | 确定技术栈，基本代码规范，熟悉安卓教程、组件 | 9.16-9.17 | 所有人 | | 4 | 分析与识别风险，提出解决方案，确定本阶段开发方案 | 9.18 | 所有人 | | 5 | 熟悉基本页面绘制技术，按照教程书写代码进行练习，更加熟练 | 9.19-9.20 | 所有人 | | 6 | 基本的页面绘制：注册、登录、个人信息、好友、添加好友、积分的原型页面及其他可能需要的页面 | 9.21-9.22 | 刘遇时 | | 7 | 基本的页面绘制：确定配色，侧边栏、更换主题、年月日远离手机时间统计的原型页面及其他可能需要的页面 | 9.21-9.22 | 周洹羽 | | 8 | 基本的页面绘制：闹钟列表、设定、选择铃声、选择任务、响起的原型页面及其他可能需要的页面 | 9.21-9.22 | 杨弈骋 | | 9 | 基本的页面绘制：远离手机的任务列表、选择白名单、任务完成、屏保的原型页面及其他可能需要的页面 | 9.21-9.22 | 崔绍杰 | | 10 | 以上所有页面的设计审核；  进一步改进需求文档 | 9.23 | 所有人 | | 11 | 页面风格统一，改进和美化 | 9.24-9.25 | 周洹羽 | | 12 | 页面本身逻辑和跳转逻辑 | 9.24-9.25 | 崔绍杰 杨弈骋 | | 13 | 后端基本框架设计与搭建 | 9.24-9.25 | 刘遇时 | | 14 | 闹钟任务页面的绘制（即游戏页面） | 9.26-9.28 | 所有人 | | 15 | 页面和后端框架的进一步审核与改进 | 9.29-10.1 | 所有人 | | 16 | 编写《迭代评估报告》，进行评审记录和开发总结  软件打包，其余审核与改进修复 | 10.2-10.4 | 所有人 | | | | |
| 预期成果：  《Vision文档》、use-case模型、《软件需求规约》、《迭代评估报告》  一份安卓java代码和一个安装包，除了无执行逻辑之外的所有页面绘制（即除了不会真实设定闹钟、无数据之外，其他所有跳转和逻辑均完善），界面美观 | | | |
| 主要的风险和应对方案：  当前项目是完成原型UI设计，从高到低的4个风险如下：   1. **要求制定三套及以上不同的主题皮肤**   ①评估：  一般app都有更换主题的设置，并且至少有dark和light两种主题。但在此项目中，后期还可能将主题作为积分的奖励，因此可能产生需要多个不同主题的需求。由于该需求出现可能较大，而且实现较为麻烦，故是我们需要解决的风险中优先级最高的。  ②解决：  主要用到values/styles.xml来实现，具体可有如下两种情况。我们在实现时注意规范，留下修改的余地，方便需求风险出现时及时进行修改，即使需求紧急，也可以快速修改完成。  对于在切换主题时不改变的属性： 在styles.xml中完成一个某个组件的样式设计后，在layout.xml中相对引用，例如：  <style name="test\_but"><item name="android:shape">@drawable/shape\_cart\_but</item></style>  <Button android:id="@+id/but\_cart\_reduce" style="@style/cart\_but" android:text="-"/>  对于在切换主题时改变的属性: 在manifest.xml里设置theme，然后在改变theme时使用setTheme的方法。   1. **要求界面风格一致且做到简洁美观**   ①评估：  该风险主要为人员配合容易出现的问题。由于工作进度原因，安卓APP前端需要由小组所有成员共同设计，由于小组成员对Android Studio工具掌握程度不一，小组4人对APP设计理解不尽相同，可能会出现界面设计风格不一致的问题，最终会造成用户觉得上手的APP风格混乱，使用逻辑迥异，从而使用户失去使用APP的兴趣。由于该需求有很大可能出现，而且处理起来较为复杂，有一定可能会耗费很多时间才能达到预期效果，故风险级别高  ②解决：  根本解决方案：在熟悉Android Studio开发，各组件的运用，view，viewgroup，fragment，activity等层次架构的基础上，形成共同的前端开发流程和思路，商讨形成一套完整的APP风格体系。但是由于时间紧迫，且开发人员都是第一次进行app开发，思路和体系难以做到如此清晰，该方案过于理想，难以实现。  缓解应急方案：先商讨出一定的规范，所有人的界面设计尽量简约，尽量清爽干净，将一些常用的组件抽离出来共同使用。如果在后期审核过程中依然发现风格不同需要调整，那么在小组中派一个开发人员对所有人画的APP前端进行最后的审核，微调，进而达成风格的一致。   1. **要求可以自定义屏保页面上的激励语句**   ①评估：  经过调研我们发现，用户对于可以自定义屏保页面上的激励语句的功能有着极大的喜爱。虽然大作业选题文档中没有此项需求，但考虑到迎合用户的需求和喜好才是app抢占市场和获取用户粘性的必要手段，我们大概率会在后期完善阶段加入此需求。由于该需求虽然大概率出现，但是实现简单，耗时少，故风险级别较低。  ②解决：  缓解方案：此项需求较为简单，只需要在本次迭代中为自定义激励语句预留位置（如设置为默认激励语句）和接口，并在需求产生时新添加一个设置页面，用于用户输入自定义语句，就能在较短时间内顺利保证该功能的简单实现。  完整方案：如果需要常用自定义语句的存储功能，则需要进一步地考虑对语句在前端或后端进行存储。   1. **要求多种语言适配**：   ①评估：  此APP的主要使用区域是中国境内，系统语言包括汉语。同时考虑到英语为当前世界上最通行的语言，而且部分中国人也习惯使用英文界面，所以后期可能会出现提供中英文切换乃至于繁体中文、俄文、日文适配的需求。由于该需求解决方案简单，而且也无需在当前开发中预留接口或规范，故风险级别较低。  ②解决：  当需求出现时，开发者为中英文分别新建value包value-en和value-zh，对每个出现文字的地方都提供中英文的string。 APP在启动时获取手机系统语言，初始化语言。同时系统界面也提供相关按钮来更新语言配置，从而实现语言切换的功能。 | | | |