悠行旅游在线系统设计与开发

**软件开发计划**

**版本：1.0**

编写： 茆志

校对： 李超

审核： 丁嘉瑞

批准： 解智伟

**西北工业大学－12组**

**2019年7月**

目录

[1引言 4](#_Toc13040158)

[1.1文档标识 4](#_Toc13040159)

[1.2项目概述 4](#_Toc13040160)

[1.3 策划背景概述 4](#_Toc13040161)

[1.4 文档概述 5](#_Toc13040162)

[1.5 参考文档 5](#_Toc13040163)

[2 软件开发活动的总体实施计划 5](#_Toc13040164)

[2.1软件开发过程 5](#_Toc13040165)

[2.2软件开发总体计划 6](#_Toc13040166)

[2.2.1软件开发方法 6](#_Toc13040167)

[2.2.2软件产品标准 6](#_Toc13040168)

[2.2.3可重用软件产品 9](#_Toc13040169)

[2.2.4关键需求的处理 9](#_Toc13040170)

[2.2.5计算机硬件资源的利用 10](#_Toc13040171)

[2.2.6决策理由的记录 10](#_Toc13040172)

[2.2.7需方评审所需访问 11](#_Toc13040173)

[3. 软件开发活动实施计划 11](#_Toc13040174)

[3.1 项目策划和监控 11](#_Toc13040175)

[3.1.1任务进度监控 11](#_Toc13040176)

[3.1.2项目开支监控 12](#_Toc13040177)

[3.1.3人员表现监控 12](#_Toc13040178)

[3.2 系统设计 12](#_Toc13040179)

[3.3 软件需求分析 13](#_Toc13040180)

[3.4 软件设计 14](#_Toc13040181)

[3.5 软件实现 14](#_Toc13040182)

[3.6 软件测试计划 15](#_Toc13040183)

[3.7 软件移交准备 15](#_Toc13040184)

[3.8 软件质量保证 16](#_Toc13040185)

[3.9风险管理 17](#_Toc13040186)

[3.10安全性 18](#_Toc13040187)

[6.进度表和活动网络图 20](#_Toc13040188)

[6.1 进度表 20](#_Toc13040189)

[6.2 活动网络图 22](#_Toc13040190)

[7.项目组织和资源 23](#_Toc13040191)

[7.1 项目组织 23](#_Toc13040192)

[7.2 项目资源 24](#_Toc13040193)

[7.2.1 人力资源 24](#_Toc13040194)

[7.2.2 设施资源 25](#_Toc13040195)

[7.2.3 合同 25](#_Toc13040196)

[7.2.4其他资源 25](#_Toc13040197)

# 1引言

## 1.1文档标识

中文名称：《软件开发计划》。

英文名称：“Software Development Plan（SDP）”。

文档版本：“1.0”。

文档编号：“UTRIP-SDP-1.0”。

## 1.2项目概述

本文档适用于“悠行旅游在线系统”项目（以下简称“UTRIP项目”）的开发过程。UTRIP项目由西北工业大学软件与微电子学院实验室提出，由该学院学生组成的开发队伍12组承接，实现相应的功能，最终提交软件成品和必要的文件。该项目的标识号为“UTRIP”,初级的软件版本为“1.0”，同时会在后续的开发中推出“2.0”，“3.0”版本。

项目内容为：

开发基于web的在线旅游系统，目标是让用户足不出户就可以安排自己的旅游行程，将外出旅游的整个复杂过程简化，让您有着更好的出门体验。

## 1.3 策划背景概述

随着景区不断通过大数据、互联网打造智慧景区，使线上线下信息化体验不断融合，度假品质不断提高。在线旅游平台不断对景区进行融合打造、使在线度假更加便利。预计2019在线度假市场仍将保持高速增长。而机票交易份额则会不断下滑。住宿交易相对稳定。

当前移动旅游的主要形式为线上线下一体化解决方案。包括了以携程为代表的在线旅游；以马蜂窝为代表的用户生产内容社区；以腾讯的“一部手机游云南”为代表的头部企业，以及众多类似于定制游、个性游产品的创新型服务。过去一年中，携程、美团点评通过资本层面并购等方式，分别确立了各自在旅游预订与生活服务市场的优势地位；用户生产内容攻略社区尽管在2018年出现了诸如评论造假等负面事件，但仍通过内容变现商业模式的回归，重新成为在线旅游资本寒冬中为数不多的亮色；而以腾讯“一机游”模式为代表的目的地一体化平台，虽具争议但也对以目的地整体进行信息化建设和运营提供了新的模式。线上旅游项目具有很大的市场前景。

## 1.4 文档概述

本文档依据国家标准[《GB/T 8567-2006计算机软件文档编制规范》](../资料/GBT%208567-2006%20计算机软件文档编制规范.pdf)制定，属于技术文档，仅限于联合实验室和中兴通讯的项目相关人员阅读。

## 1.5 参考文档

* [《GB/T 8567-2006计算机软件文档编制规范》](../资料/GBT%208567-2006%20计算机软件文档编制规范.pdf)，国家标准
* [《UTRIP-1.0 可行性分析报告》](file:///G:\1－项目计划阶段\SSM-ZTE-AndroidUI-SDP-1.0(E)%20软件开发计划.docx)，12组

# 2 软件开发活动的总体实施计划

## 2.1软件开发过程

模型选择：本软件采用瀑布模型进行软件开发

原因：本软件的需求明确，理解充分，并且较为稳定

阶段定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | 阶段 | 入口标准 | 任务 | 出口标准 |
| 1 | 需求分析 | 项目启动 | 进行软件需求分析 | 软件需求规格说明书形成基线 |
| 2 | 概要设计 | 软件需求规格说明书已完成 | 进行各模块的概要设计 | 概要设计完成并形成基线 |
| 3 | 详细设计 | 数据库设计，概要设计已完成 | 进行详细设计编写 | 详细设计完成并形成基线 |
| 4 | 编码 | 详细设计编写完成 | 进行编码 | 编码完成并形成基线 |
| 5 | 测试 | 编码完成，各测试用例编写完成 | 进行各级测试 | 测试报告 |
| 6 | 验收 | 测试已完成 | 验收测试，项目总结 | 产品发布，项目总结 |

## 2.2软件开发总体计划

### 2.2.1软件开发方法

本软件的开发采用GitHub去进行版本控制，成员每天会将自己今天的工作文档上传，负责管理的人将会为每天的进程创建一个文件夹，让组员上传自己的工作文件。

一个共享的代码版本控制系统是必须的，团队可建立自己单独的持续集成环境，但需要保持系统环境的一致。

每个迭代结束时，应尽量安排一个两地同步的演示会议。让所有人都在电话会议上看到这个迭代的成果。迭代后的总结与回顾也应当两地一起进行，如果人数和条件不允许，可以分别进行，并互相通报回顾结果和改进方法。

### 2.2.2软件产品标准

需求标准：

1. 完整性：每一项都必须完整的描述即将交付的使用的功能，必须包含开发人员设计和实现这项功能需要的所有信息。
2. 正确性：每一项需求都必须准确地描述将要开发的功能。
3. 可行性：需求必须能够在系统上以及运行环境的已知能力和约束条件内实现。为避免不可实现的需求，在需求获取阶段，应安排开发人员进行可行性检查，判断技术上能够实现哪些需求，或者什么功能需要额外的成本才能实现。
4. 必要性：每一项需求记录的功能都必须是用户的真正需求，或者是为符合外部系统需求或某一标准而必须具备的功能。
5. 有优先次序：为每一项功能需求、特性或用例指定一个实现的优先级，以表明它在产品的某一版本中的重要程度。
6. 无歧义：一项需求声明对所有读者应该只有一种一致的解释，应在词汇表中列出所有专用和可能让用户感到迷惑的术语。
7. 设计标准：
   1. 可靠性：该软件应具备在测试运行过程中避免可能发生故障的能力，且一旦发生故障后，具有解脱和排除故障的能力。
   2. 健壮性：软件对于规范要求以外的输入能够判断出这个输入不符合规范要求，并能有合理的处理方式。
   3. 可修改性：软件有良好的结构和完备的文档，系统性能易于调整。
   4. 容易理解：软件的可理解性是其可靠性和可修改性的前提。它并不仅仅是文档清晰可读的问题，更要求软件本身具有简单明了的结构。
   5. 效率性：在达到原理要求功能指标的前提下，程序运行所需时间愈短和占用存储容量愈小，则效率愈高。
   6. 标准化原则：在结构上实现开放，基于业界开放式标准，符合国家和信息产业部的规范。
   7. 可扩展性：软件设计完要留有升级接口和升级空间。对扩展开放，对修改关闭。
   8. 安全性：系统能够保持用户信息、操作等多方面的安全要求，同时系统本身也要能够及时修复、处理各种安全漏洞，以提升安全性能。
8. 编码标准：
   1. 程序语句组成容易识别，每个程序块只有一个入口和出口。
   2. 复杂的结构应该用基本控制结构进行组合嵌套来实现。
   3. 语句中没有的控制结构，可用一段等价的程序模拟，但要求该程序段在整个系统应前后严格一致。
   4. 应尽量避免在系统初始化时运行过多的代码。
      1. 测试用例标准：
9. 应尽可能采用成熟的测试用例设计方法来知道设计。
10. 测试用例能够代表并覆盖各种合理的不合理的，合法的不合法的，边界的和越界的以及极限的输入数据、操作等。
11. 测试用例能对程序中可能存在的错误有针对地测试。
12. 测试结果的正确性是可判定的，每一测试用例都应有相应的期望结果。
13. 对同样的测试用例，系统的执行结果应是相同的。
    * 1. 测试过程标准：软件测试过程应遵循PDCA循环，即计划（Plan）,执行（Do）,检查（Check）,处理（Action）。具体的流程可以概括为：首先在分析清楚需求的前提下对测试活动进行计划和设计，然后按既定的策划执行测试和记录测试，对测试的结果进行检查分析，形成测试报告，这些测试结果和分析报告又能指导下一步的测试设计。进而由此形成了一个质量改进的闭环。

### 2.2.3可重用软件产品

采用可重用软件产品

* 1. 开发方应标识和评价为满足合同要求而使用的可重用软件产品。
  2. 只要切实可行，就应该采用满足准则的可重用软件产品。

开发可重用软件产品

1. 合同期间，开发方应评估开发可重用软件产品的可行性、成本以及可能产生的效益，并向需方说明费效比。
2. 合同中也可以按要求专门开发可重用软件产品。

### 2.2.4关键需求的处理

安全保密性处理方法：

1. 透明加密技术：提供对涉密或敏感文档的加密保护，达到机密数据资产防盗窃、防丢失的效果，同时不影响用户正常使用。
2. 泄密保护：通过对文档进行读写控制、打印控制、剪切板控制、拖拽、拷屏/截屏控制、和内存窃取控制等技术，防止泄漏机密数据。
3. 强制访问控制：根据用户的身份和权限以及文档的密级，可对机密文档实施多种访问权限控制，如共享交流、带出或解密等。
4. 双因子认证：系统中所有的用户都使用USB-KEY进行身份认证，保证了业务域内用户身份的安全性和可信性，完全符合国家保密局的要求。
5. 文档审计：能够有效地审计出，用户对加密文档的常规操作事件。
6. 三权分立：系统借鉴了企业和机关的实际工作流程，采用了分权的管理策略，系统管理采用审批，执行和监督了职权分离的模式。
7. 安全协议：确保密钥操作和存储的安全，密钥存放和主机分离。
8. 对称加密算法：系统支持常用的AES、RC4、3DES等多种算法，支持随机密钥和统一密钥两种方式，更安全可靠。

其他关键需求的保证：

制定策略，用来保证使所标识软件的需求、设计、实现和操作出错时，使其损失尽可能达到最小。

### 2.2.5计算机硬件资源的利用

* 1. 如果作业的等待时间相同，则要求服务的时间越短，其优先权越高，采用SJF（短作业优先）算法。
  2. 当要求服务的时间相同时，作业的优先权又决定于其等待时间，因而采用FIFO（先进先出）算法。
  3. 对于长作业的优先级，可以随等待时间的增加而提高，当其等待时间足够长时，获得计算机的硬件资源。

### 2.2.6决策理由的记录

1. 开发方应记录在软件需求分析、设计、实现和测试中作出重要决策的理由，这些记录对保障机构有用。
2. 决策理由应包括所考虑的折中情况、分析方法和决策所用的准则。
3. 这些理由应记录在文档、代码注释或其他将移交给保障机构的媒介中。
4. “重要决策”的含义应在软件开发计划中加以描述，做出这些决策的理由应在软件开发计划中指出。

### 2.2.7需方评审所需访问

开发方应为需方或其授权代表访问有关设施提供方便。

# 3. 软件开发活动实施计划

## 3.1 项目策划和监控

在线考试平台，是一款面向中学生的和学校的在线考试、试题管理平台。本平台为使用者提供了在线考试和在线评阅等功能。实现了教学互联网化，为学生和老师带来了便利。

项目市场分析：

当前市场环境下，缺乏功能全面的考试系统。

项目监控内容：

项目负责人应定期的对项目进行进展报告，将各项监控的结果记录在项目进展报告里，以便使项目主管部门及项目组成员及时了解项目的真是进展状况。

### 3.1.1任务进度监控

主要任务是记录下任务对的实际开发时间与实际结束时间，实际的工作量及工作成果等信息以判断该任务是否正常执行。

项目负责人应该和任务的责任人沟通，找出眼目的原因，适当的修改对于进度延误的任务，原有的计划或者要求责任人加紧完成进度。

### 3.1.2项目开支监控

主要的目的是将项目的实际开支控制在预算范围之内。记录下所有的项目开支，与计划中的开支进行比对，看是否超出了原预算，若有较大的赤字，则要找出具体的费用超出项，分析原因，并采取相应的措施。

### 3.1.3人员表现监控

项目的负责人应在平时记录下项目每个成员的表现，对表现突出的成员进行表扬和肯定；对表现不好的成员提出批评，并要求其立即改正态度。

## 3.2 系统设计

系统开发一共分为以下几个模块：

1. 登录
2. 评分
3. 答题计时
4. 试卷展示

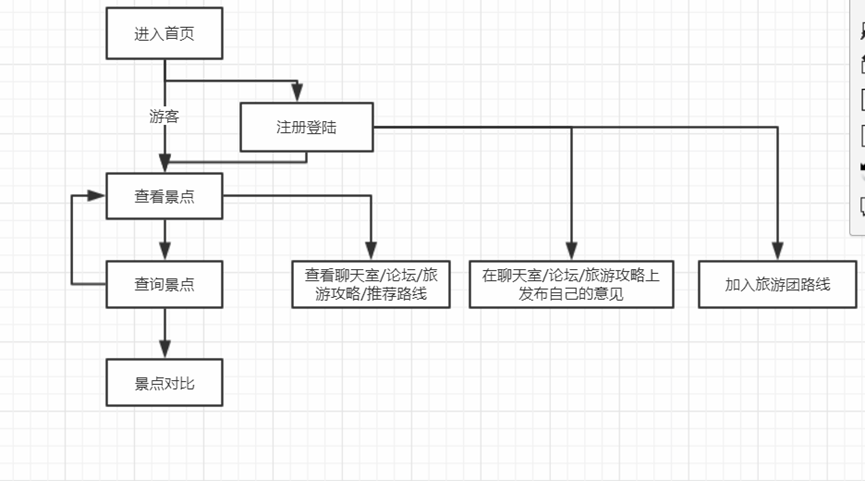
由于本系统使用了MVC设计模式，所以，在每个开发模块中可分为前端模板、数据库表结构设计和后台业务逻辑处理等步骤。其中每个模块所使用的具体技术和相应的设计描述如下：

1）登录模块：

2）数据库：为了保证数据的安全和保密性，答案在数据库中使用其内容的散列进行储存。

1. 前端模板：使用jsp作为模板。采用servlet作为controller层控制模块。

## 3.3 软件需求分析



业务流程图

业务流程描述：

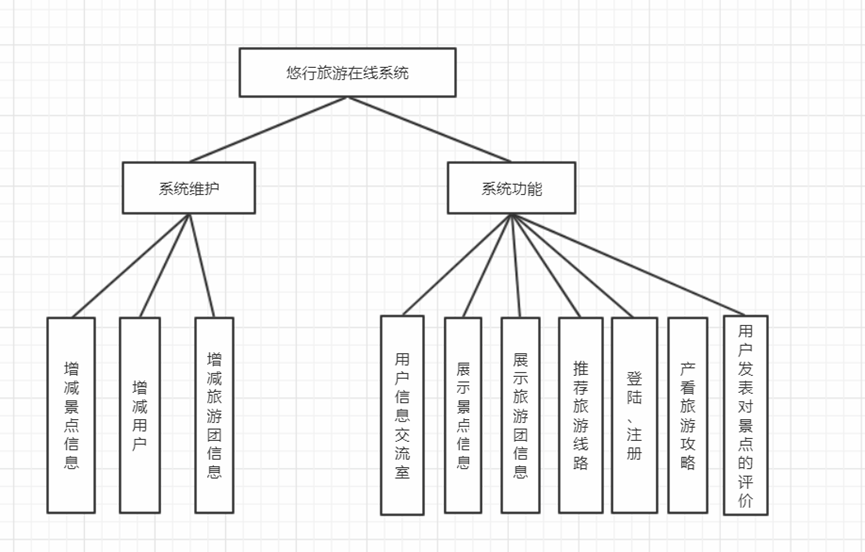
1. 游客和客户都可以对网页进行查看
2. 游客和客户都可以进行景点对比
3. 所有用户和系统的互动（点赞，评论，论坛发表意见）都需要登陆

跟据上述的业务流程描述，系统需要的功能有如下几点：

* 1. 景点信息编辑
  2. 论坛消息管理
  3. 景点评论管理
  4. 点赞实时显示

由于系统功能较为复杂，并且业务在短时间内扩展的 可能性很大。系统架构必须能够支持功能的水平扩展并且尽可能的实现团队开发友好型。所以，技术选型上我们采用了MVC的设计模式，并且使用了SpringMVC这种优秀的开源框架。对于在线计时器这一需求，我们使用了WevSocket协议实现了前后端的全双工通讯，在保证安全性的同时，使协作变得更容易。

## 3.4 软件设计



系统结构图

本系统各个功能模块均围绕系统维护、系统功能这两个方面进行设计。根据以上提供的系统结构图可以看出，

## 3.5 软件实现

软件实现使用java语言，MVC实现使用的是springmvc框架。具体实现内容请参阅软件设计实现说明书。

## 3.6 软件测试计划

本项目中，测试可以分为如下几种类型：

1. 代码静态审查
2. 单元测试
3. 集成测试
4. 系统测试

为保证程序质量，开发人员需要对进行结对编程。完成单元测试并达到相应标准（代码覆盖率100%，分支覆盖率85%），运行正确。测试人员根据软件需求分析报告对软件进行软件集成测试和系统测试。对编写完成的代码进行评审，同时质量保证人员对评审过程和工作产品进行检测。

## 3.7 软件移交准备

软件移交的软件准备：

为保证软件能够达到预期，并能够更流畅的运行。推荐使用如下版本的依赖或者软件系统：

系统运行依赖环境：

1. Java SE Development Kit version8
2. Web服务器：Tomcat 8.x及以上
3. 数据库：MySQL 5.6数据库

系统依赖框架：

1. Spring MVC 3.0及以上
2. Maven 3构建工具

推荐使用环境：

1. 浏览器：Chrome、firefox、safari、opera均可。最佳为chrome浏览器
2. 分辨率：最佳显示效果为1920\*1080

软件移交的硬件准备：

处于对成本和需求的考虑，本系统并未进行跨平台的优化。所以，推荐将该系统巡行在x86架构的计算机或者能够良好运行jdk1.9及以上版本的计算机平台上。

软件移交的文档准备：

为能够更平稳的部署和使用该软件，我们提供了如下文档分别对应相应的使用时期：

1）软件部署说明书。该文档记录了详细的系统部署步骤、依赖软件的版本等内容。在系统的部署和前期维护提供相关的参考和辅助。

2)软件使用人员培训说明书。用于系统正式运行后，上线前相关涉众的培训时期

3)软件维护说明书。该说明书详细记录了软件可能发生的问题，并提供了相应的解决方法。并详细的记录了关键模块的实现方式和扩展方法，为后续软件功能的扩展提供指导。

## 3.8 软件质量保证

软件质量保证有如下几个基本目标：

1. 软件质量保证工作是有计划进行的。
2. 客观地验证软件项目产品和工作是否遵循恰当的标准、步骤和需求。
3. 将软件质量保证工作及结果通知给相关组别和个人。
4. 高级管理层接触到在项目内部不能解决的不符合类问题。
5. 软件质量需要全面的测试工作来保证。

为达到相应的目标，软件质量保证小组的具体职责划分如下：

1. 组长全面负责有关质量保证个各项工作
2. 总体组代表负责有关阶段评审、项目进展报表检查以及软件验收准备等三方面工作中的质量保证工作。
3. 项目的专职配置管理人员负责有关软件配置变动、软件媒体控制以及对供货单位的控制等三仿麦呢的质量保证活动
4. 各子系统的软件质量保证人员负责测试复查和文档的规范化检查工作
5. 用户代表负责反应用户的质量要求，并协助检查各类人员对软件质量保证计划的执行情况
6. 项目的专职质量保证人员协助组长开展各项软件质量保证活动，负责审查所采用的质量保证工具、技术和方法，并负责汇总、维护和保存有关软件质量或许哦那个的各项记录

质量保证措施如下：

通过质量管理责任分配，通过如下几个方面质量保证实施过程。

## 3.9风险管理

根据项目的内容，所涉及到的风险可分为如下几种：

1. 性能风险：产品能够满足需求和符合使用目的的不确定程度。
2. 成本风险：项目预算能够被维持的不确定的程度。
3. 支持风险：软件易于纠错、适应及增强的不确定的程度。
4. 进度风险：项目进度能够被维持且产品能按时交付的不确定的程度。

考虑到当前市场环境和其他环境因素以后，可以得到以下风险评估表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 风险 | 类别 | 概率 |  | 影响 |
| 资金将会流失 | 商业风险 | 40% |  | 1 |
| 技术达不到预期效果 | 技术风险 | 30% |  | 1 |
| 人员流动频繁 | 人员风险 | 60% |  | 3 |

为控制以上风险，在控制过程中需要遵守以下几个基本原则：

* 1. 实施对重要风险的跟踪；
  2. 每月对风险进行一次跟踪；
  3. 风险跟踪应与项目管理中的整体跟踪管理相一致；
  4. 风险项目应随着时间的不同而相应地变化。

在具体的实施过程中采用CRM风险控制模型。由于CRM模型要求在项目生命期的所有阶段都关注风险识别和管理，它将风险管理划分为5个步骤：风险识别、分析、计划、跟踪、控制。强调了这是一个在项目开发过程中反复持续进行的活动序列。每个风险因素一般都需要按顺序经过这些活动，但是对不同风险因素开展的不同活动可以是并发的或者交替的。更符合本项目的风险管理模式。

## 3.10安全性

本系统针对多种工具方式都制定了一套较为完善的防御方案。可以抵制恶意用户对系统进行的攻击，有力的提高系统的安全性，保证系统能够相对安全的部署、运营、维护、升级、发展。

密码保护：

1)用户密码使用32位MD5散列保存

2)利用密码强度保护机制，排除过于薄弱的密码

输入验证：

1)利用验证控件，对用户输入的数据进行类型、大小、范围的验证

访问控制：

1. 后天管理目录可以通过更改目录名来防御攻击
2. 后台管理文件对访客的用户身份的统一验证
3. 从整体上限制直接输入地址访问后台文件

注入攻击防范：

1. 查询参数使用严格的过滤函数进行过滤
2. 限制URL的传参类型、数量、范围等来防止通过构造URL进行恶意攻击

其他措施：

1. 跟踪用户操作和系统异常，避免恶意用户通过反射程序集利用代码漏洞攻击
2. 建议项目运行方配备有关放病毒及木马软件来处理系统病毒及木马问题
3. 建议项目运行方配备具有快照功能的存储设备对用户文件进行实时备份

4）建议项目运行放配备分级存储管理软件系统，以提高存储控件利用率

# 6.进度表和活动网络图

## 6.1 进度表

|  |  |  |  |
| --- | --- | --- | --- |
| 任务名称 | 工期 | 开始时间 | 完成时间 |
| **1.旅游在线系统** | **12天** | **2019/7/1** | **2019/7/12** |
| **1.1 项目设计** | **2天** | **2019/7/1** | **2019/7/2** |
| **1.1.1 开发计划** | **1天** | **2019/7/1** | **2019/7/1** |
| 1.1.1.1 撰写项目计划文档 | 1 天 | **2019/7/1** | **2019/7/1** |
| **1.1.2 项目分析** | **1 天** | **2019/7/1** | **2019/7/1** |
| 1.1.2.1 撰写概要设计说明书 | 1 天 | **2019/7/1** | **2019/7/1** |
| 1.1.2.2 撰写需求规格说明书 | 2 天 | **2019/7/1** | **2019/7/1** |
| **1.2 第一次迭代** | **4 天** | **2019/7/1** | **2019/7/4** |
| **1.2.1 开发阶段1** | **2 天** | **2019/7/1** | **2019/7/3** |
| 1.2.1.1 明确项目实体，确定实体类 | 1 天 | **2019/7/1** | **2019/7/1** |
| 1.2.1.2 明确各实体之间的关系，对项目进行整体建模 | 1 天 | **2019/7/2** | **2019/7/2** |
| **1.2.2 开发阶段2** | **2 天** | **2019/7/1** | **2019/7/2** |
| 1.2.2.1 明确数据库，编写数据库导入语言 | 1 天 | **2019/7/1** | **2019/7/1** |
| 1.2.2.2 编写相关测试数据,创建数据库 | 1 天 | **2019/7/2** | **2019/7/2** |
| **1.2.3 完成第一次迭代 ，对数据库进行单元测试** | 2天 | **2019/7/3** | **2019/7/4** |
| **1.3 第二次迭代** | **4 天** | **2019/7/5** | **2019/7/8** |
| **1.3.1 开发阶段3** | **2 天** | **2019/7/1** | **2019/7/6** |
| 1.3.1.1 实现后台对于景点的创建 | 1 天 | **2019/7/7** | **2019/7/7** |
| 1.3.1.2 实现后台对于景点的更改 | 1 天 | **2019/7/8** | **2019/7/8** |
| **1.3.2 开发阶段4** | **2 天** | **2019/7/5** | **2019/7/6** |
| 1.3.2.1 实现用户对于景点的操作 | 1 天 | **2019/7/7** | **2019/7/7** |
| 1.3.2.2 实现用户的互动效果 | 1 天 | **2019/7/8** | **2019/7/8** |
| **1.3.3 完成第二次迭代，对代码进行单元测试和整合测试** | 1 天 | **2019/7/8** | **2019/7/8** |
| **1.4 第三次迭代** | **4天** | **2019/7/9** | **2019/7/12** |
| **1.4.1 开发阶段5** | **2 天** | **2019/7/9** | **2019/7/10** |
| 1.4.1.1 完成后台对于线路推荐算法 | 1 天 | **2019/7/9** | **2019/7/9** |
| 1.4.1.2 完成后台的对比 | 1 天 | **2019/7/10** | **2019/7/10** |
| **1.4.2 开发阶段6** | **2 天** | **2019/7/11** | **2019/7/10** |
| 1.4.2.1 完成用户对于不同城市景点查询 | 1 天 | **2019/7/11** | **2019/7/11** |
| 1.4.2.2 完成对于热点景点查看 | 1 天 | **2019/7/12** | **2019/7/12** |
| **1.4.3 完成第三次迭代，对代码进行单元测试和整合测试** | 1 天 | **2019/7/12** | **2019/7/12** |
| **1.5 监督 & 控制** | **12 天** | **2019/7/1** | **2019/7/12** |
| 1.5.1 每天召开开发会议 | 12 天 | **2019/7/1** | **2019/7/12** |
| 1.5.2 开发回顾 | 12 天 | **2019/7/1** | **2019/7/12** |
| **1.6 项目结束** | **1 天** | **2019/7/12** | **2019/7/12** |
| 1.6.1 进行项目展示 | 1 天 | **2019/7/12** | **2019/7/1** |

## 6.2 活动网络图

开发计划

项目分析

创建项目时间表

迭代1

监管&控制

迭代2

迭代3

# 7.项目组织和资源

## 7.1 项目组织

|  |  |  |
| --- | --- | --- |
| 人员 | 职务 | 职责 |
| Teacher | 项目发起人 | 项目需求的提供者和项目的验收者，决定项目是否成功，并对于参与者进行评价。 |
| 李超 | 开发经理 | 与项目发起人进行沟通，对于内部的开发和验收负责，对于项目组内部的人力、资源进行协调。 |
| 李超 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 解智伟 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 陈卓明 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 茆志 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 丁嘉瑞 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 刘欣禹 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 马驰 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 马鑫 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |
| 李子奇 | 开发人员 | 对于分担的模块进行编写，并完成单元测试，对于编写的模块负全责。 |

## 7.2 项目资源

### 7.2.1 人力资源

* + - 1. 项目的开发人员主要有9人，其中项目经理由一位开发人员兼任。
      2. 估计项目开发阶段总共需要450工时，每人50小时工作量，分为7天完成。
      3. 9人位于同一地理位置，无需远程协调工作，便于交流。
      4. 项目计划阶段需要9人共同完成，达成统一认识，约需要25工时。第一次迭代任务较为简单，主要是数据库的搭建，越需要10工时。第二次迭代任务较重，需要完成项目的基本功能，越需要50工时的工作量。第三次迭代对系统进行完善，越需要25工时的工作量。
      5. 项目的开发者均为未毕业大学生，编程技术有限，项目开发经验不足，技术级别较低

### 7.2.2 设施资源

1）项目几乎没有设施支持，主要为上课时间机房的计算机。其他需要的硬件资源由开发人员自行解决。

### 7.2.3 合同

1）合同中未就资源提出介绍。

### 7.2.4其他资源

1）开发人员没有任何除精神支持以外的任何资源支持并乐在其中。