4/6/2015

王里奥 郭超凡 吉训煜 黄德华 姜栋煜

Instructor: hongyang zhang

BEIJING JIAOTONG UNIVERSITY

Project Plan

Intelligent Clinic System

Table of Contents

[Abstract 2](#_Toc416253815)

[1. User Story definition and Collection 4](#_Toc416253816)

[1.1 User Role 4](#_Toc416253817)

[1.2 & 1.3 & 1.4 (Functions & Purpose & Standards) 4](#_Toc416253818)

[2. Value Definition 12](#_Toc416253819)

[2.1 Area(s) of impact 12](#_Toc416253820)

[2.2 Potential Value of the IT solution 12](#_Toc416253821)

[2.3 Time Frame for achieving the MOV 13](#_Toc416253822)

[3. Feature Mining and Descriptions 13](#_Toc416253823)

[3.1 Feature Definition 13](#_Toc416253824)

[3.1.1 Feature Classification 14](#_Toc416253825)

[3.1.2 Feature Definition 14](#_Toc416253826)

[3.2 Detailed description (blueprint) 15](#_Toc416253827)

[3.2.1 信息维护与安全 15](#_Toc416253828)

[3.2.2 智能提示挂号 15](#_Toc416253829)

[3.2.3 网上在线支付 16](#_Toc416253830)

[3.2.4 预约挂号分流 17](#_Toc416253831)

[3.2.5 医生绩效评价 18](#_Toc416253832)

[3.2.6 管理员显示医生排名 19](#_Toc416253833)

[3.3 Identify the values for each Feature provided. 19](#_Toc416253834)

[3.4 Ways to verify or validate the feature in the future 20](#_Toc416253835)

[4. Features Priorities Identification 21](#_Toc416253836)

[4.1 Features value comparison (valued between 1, 6) 21](#_Toc416253837)

[4.2 Customer Satisfaction comparison 22](#_Toc416253838)

[4.3 Complex comparison 27](#_Toc416253839)

[4.4 Give the features priorities from high to low 29](#_Toc416253840)

[4.5 Summary and conclusion 29](#_Toc416253841)

[5. Road Map of Product 30](#_Toc416253842)

[5.1 Road Map – Project Life Cycle 30](#_Toc416253843)

[5.2 Envision/ Inception – Road Map 30](#_Toc416253844)

[6. Product Backlog 31](#_Toc416253845)

[6.1 Give the chart of your product backlog 31](#_Toc416253846)

[7. Sprints Definition & Sprint Backlog 37](#_Toc416253847)

[7.1 Give the chart of sprints for each of your features 37](#_Toc416253848)

[7. DOD & DOR Definition 39](#_Toc416253849)

[7.1 The Definition of DOR 39](#_Toc416253850)

[7.2 The Definition of DOD 39](#_Toc416253851)

[8. Test Plan (Checklist) 39](#_Toc416253852)

[8.1 The Checklist of your test plan for your product 41](#_Toc416253853)

# 

王里奥-----------------------------------------------------------20%

吉训煜-----------------------------------------------------------20%

黄德华-----------------------------------------------------------20%

姜栋煜-----------------------------------------------------------20%

郭超凡-----------------------------------------------------------20%

# Abstract

Note: a succinct one page summary of how you carry on the project and when the project can be released interactively.

为了开发一个智能医疗就诊系统，我们先对用户的需求进行收集，在采集用户故事之后，我们先对每一个故事进行了审核，确定出相应的验证审核标准，对现阶段无法确定验证标准的用户故事，我们将之放在第二次迭代开发的过程中，已确定用户是否真正需要该功能。接着我们根据上一阶段Proposal价值分析部分（MOV）重新定义和完善系统能给用户的价值，并最终确认出系统开发整个生命周期应遵循的Roadmap。在第三部分，我们对所有搜集到的用户故事进行分类，并挖掘出相应的feature。定义feature和进行的详细的说明是这一块的重要内容，最后我们给出了验证每一个feature的方法。第四部分我们把前面定义出来的feature根据能够用户提供价值的高低和使用拓扑排序确定怎么的顺序能够用户最大的满意度，进行优先级排序，最终给出排序后的结论和理由。第五部分和第六部分，我们分别根据前面的结论对backlog和sprints进行了编写和定义。第七部分，我们详细介绍了DOD和DOR的内容，在第八部分，我们确定了测试计划部分的checklist。

In order to develop an intelligent clinic system, we firstly collect all the user stories from customers, then we review each user story and determine the verification standards for them. For those we cannot verify right now, we will put them into the second interaction and determine whether they are needed or not. Next, we redefine and improve the values we can bring to the customer according to the MOV we gave in the last document “Proposal”, and we finally draw up the “Roadmap” our product lifecycle will follow. In the third part, we classify all the user story collected and mine out all the features accordingly. In the fourth part, we sort the features we determined and give the summary and reason at last. In the fifth and sixth part, we give the backlog and sprints.

In the end, we give the checklist of test plan.

# User Story definition and Collection

## User Role

根据前面的立项报告，我们已经确定下来我们的用户群体，分别为患者用户，普通医生，专家医生，管理员。然后我们根据开发产品的具体需求确定了如下的用户故事。每一个用户故事包含三个部分，分别是用户角色，用户需要的功能，和最终要达到的目的。最后我们附上了对每一个用户故事的验证标准。

## & 1.3 & 1.4 (Functions & Purpose & Standards)

这部分我们使用Excel 表格进行详细的说明。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | 用户角色 | 功能 | 目的 | 验证标准 |
| 1 | 病人 | 注册系统 | 注册后能够登录系统，确定用户的身份 | 病人注册使用手机进行身份验证，并输入身份证号，作为初次注册的凭证，如果信息不正确，无法完成注册。 |
| 2 | 病人 | 登录系统 | 保护自己的账户信息或者显示用户自定义的信息 | 系统随机产生随机码，用户必须输入正确的信息，才能完成登录验证 |
| 3 | 病人 | 查询医生信息 | 了解医生具体的信息，如医生的科室，学历等 | 系统能够显示医生的科室，学历，临床经验等病人关注的信息。 |
| 4 | 病人 | 找回密码 | 忘记密码或丢失密码能够重新找回 | 系统要求用户输入身份证号，并输入系统发送到绑定手机的验证号，信息正确无误才能完成密码找回操作 |
| 5 | 病人 | 编辑自己的信息 | 完善自己的资料信息，可以供别人查看 | 病人可以输入一些自己的信息，如年龄，性别，所患的病症等等。 |
| 6 | 病人 | 修改自己的信息 | 资料信息提交后，可以更改 | 病人可以修改一些自己的信息，如年龄，性别，所患的病症等等。 |
| 7 | 病人 | 删除自己的信息 | 可以根据自己的需要，删除不需要的信息 | 病人可以删除一些自己的信息，如年龄，性别，所患的病症等等，这些信息将从数据库中永久删除。 |
| 8 | 病人 | 系统能够智能提示用户挂号的科室 | 方便挂号 | 系统能够根据用户的测试问题情况，自动推荐病人应该挂号的科室 |
| 9 | 病人 | 系统能够智能推荐用户可选的医生 | 方便挂号 | 系统能够根据用户测试问题情况，自动推荐病人可预约的医生 |
| 10 | 病人 | 系统提示所选医生可挂号的时间 | 方便挂号 | 每一个医生的坐诊时间不同，系统显示当前医生可接诊的时间。 |
| 11 | 病人 | 系统显示在线的医生 | 可以咨询在线医生 | 医生登录系统后，能够查看登录系统在线的医生列表，并能够点击咨询，发起咨询会话 |
| 12 | 病人 | 系统可以接受离线的医生回复信息 | 可以查看未查看的咨询回复 | 用户登录系统可以查看医生回复未查看的咨询回复。并用动态的图标提示用户查看信息 |
| 13 | 病人 | 系统能够保障用户的信息 | 防止用户的信息被不法分子窃取 | 当用户的账号密码输入5次以上未成功登录，系统自动锁定用户帐户，并通知用户账号错误登录的信息。 |
| 14 | 病人 | 系统能够在在线用户高峰时期，在10s内反馈用户信息 | 减少用户等待时间 | 在并发用户数量过多时，系统能够智能处理用户请求数据，保证用户正常使用不同模块的功能。 |
| 15 | 病人 | 提交预约 | 网上在线预约 | 用户确定预约信息无误后，提交预约信息，系统自动生成相应的订单 |
| 16 | 病人 | 在线支付预约订单 | 网上在线支付 | 用户点击支付，系统跳转到相应的支付界面。如果用户在提交订单之后15分钟未支付订单，系统自动取消掉用户的预约。 |
| 17 | 病人 | 查看用户的历史订单信息和待办订单和已办订单信息 | 查看订单信息 | 用户可以查看历史订单，未支付订单和已支付订单。并能够点击查看订单的具体信息。 |
| 18 | 病人 | 用户可以取消未支付待确定的订单 | 取消预约订单 | 用户可以在未支付订单选项卡里，取消掉已经提交的预约订单。系统一天只能受理最多3次订单取消业务，超过3次，用户当日无法再次进行预约。 |
| 19 | 病人 | 系统能够链接多种支付方式，如支付宝，微信支付，网银支付 | 支付方式多样 | 系统可以提供给用户不同的支付方式，选择相应的支付方式，跳转到相关的支付界面。 |
| 20 | 病人 | 用户可以从不同的角度评价接诊自己的医生 | 评价医生 | 用户可以对接诊的医生进行评价，并提交最终的评价。系统会动态提示用户未完成的评价信息。 |
| 21 | 病人 | 用户可以选择删除自己的评价 | 删除评价 | 用户可以选择删除对接诊医生的评价。删除后，数据库的相关信息被抹掉。 |
| 22 | 病人 | 用户可以选择修改自己的评价 | 修改评价 | 用户可以选择重新修改提交对接诊医生的评价。 |
| 23 | 病人 | 系统能够提示已经预约满员，无法预约的信息 | 防止挂无效预约 | 当用户预约人员已满的医生时，系统自动提示无法完成预约并提供相应的解决方案。 |
| 24 | 病人 | 系统能够显示当前医生的评价分数 | 了解医生的好评率 | 系统可以提供用户医生当前的评价分数，也可以根据用户的喜好按不同的方式进行排序。（如接诊人数最多，好评最多，收费最少等） |
| 25 | 普通医生 | 登录系统 | 保护自己的账户信息或者显示用户自定义的信息 | 系统随机产生随机码，用户必须输入正确的信息，才能完成登录验证 |
| 26 | 普通医生 | 查看自己的个人信息 | 了解自己网上的资料信息 | 医生可以查看系统中存储的自己的个人信息，并可以对信息的正确性进行申请审核操作，提醒管理员及时更新相关信息。 |
| 27 | 普通医生 | 查看自己的评价信息 | 了解自己的评价情况 | 医生可查看当前一个月用户对自己的评价信息，可以对个别用户的评价提出申述，提示管理员进行受理。 |
| 28 | 普通医生 | 查看系统自己分配的接诊病人列表 | 查看待接诊列表 | 普通医生的接诊列表由系统根据分流算法自动生成，并不断更新。 |
| 29 | 普通医生 | 系统能够识别病人的身份 | 识别病人身份 | 病人接诊时，使用就诊卡进行身份确认，系统对比挂号单与身份信息，一致，医生才可以进行接诊操作。 |
| 30 | 普通医生 | 系统能够在医生点击结束就诊后，重新生成新的待接诊列表 | 动态生成接诊名单 | 普通医生的接诊列表处理完毕后，系统根据分流算法自动给该医生分配新的待接诊病人。 |
| 31 | 普通医生 | 系统显示在线用户的咨询信息 | 查看咨询信息 | 在线医生可以随时接收病人在线咨询的信息，并以动态的方式提醒医生。 |
| 32 | 普通医生 | 医生可以回复咨询信息 | 回复咨询 | 医生回复病人的咨询问题。 |
| 33 | 普通医生 | 举报不良的用户评价信息 | 举报评价 | 医生查看评价后，可以对不正确不公平的评价进行举报申述，系统将信息反馈到管理员进行审核，同时通知用户是否更改相关评价信息。 |
| 34 | 普通医生 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾。 | 有效处理叫号未到的病人。 | 当医生叫号三次，病人还未到，系统自动将该用户的预约放到排列队尾 |
| 35 | 普通医生 | 系统根据医生在线的时间，确定当日医生工作时长 | 记录工作时间 | 系统可以计算医生的在线工作时间，医生工作结束后，注销掉账号，否则强制关闭系统时，系统保存信息之后，自动用户注销。 |
| 36 | 普通医生 | 系统实行评价一月制，及每周系统自动将以前的用户评价抹掉，供下周用户重新进行评价 | 避免以为个别用户给分偏低导致医生整体评价偏差 | 系统在每个月底将当月的用户评价自动转移到历史评价记录里，并将当前的评价信息自动清零。 |
| 37 | 专家医生 | 登录系统 | 保护自己的账户信息或者显示用户自定义的信息 | 同普通医生 |
| 38 | 专家医生 | 查看自己的个人信息 | 了解自己网上的资料信息 | 同普通医生 |
| 39 | 专家医生 | 查看自己的评价信息 | 了解自己的评价情况 | 同普通医生 |
| 40 | 专家医生 | 查看待接诊的病人列表 | 查看待接诊列表 | 专家医生系统按照预约激活的先后顺序加入到等待队列中，不断更新待接诊列表。 |
| 41 | 专家医生 | 查看已接诊的病人列表 | 了解历史接诊记录 | 专家医生可以查询所有接诊的历史记录。 |
| 42 | 专家医生 | 查看接诊的具体信息 | 了解病人接诊后的具体信息 | 专家医生可以在历史记录中查询病人的具体信息。 |
| 43 | 专家医生 | 系统识别病人的身份 | 识别病人身份 | 同普通医生 |
| 44 | 专家医生 | 举报不良的用户评价信息 | 举报评价 | 同普通医生 |
| 45 | 专家医生 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾。 | 有效处理叫号未到的病人。 | 同普通医生 |
| 46 | 管理员 | 录入病人的信息 | 录入病人 | 管理员录入病人的相关信息，并与就诊卡进行绑定。存储在医院的信息管理数据库中。 |
| 47 | 管理员 | 录入医生的信息 | 录入医生 | 管理员录入医生的相关个人信息。 |
| 48 | 管理员 | 修改医生的信息 | 更新医生信息 | 管理员可以审核医生提交的信息更新申请，并对医生个人资料信息进行更新。 |
| 49 | 管理员 | 删除医生的信息 | 删除医生信息 | 管理员可以删除医生的相关个人信息。 |
| 50 | 管理员 | 查看医生接诊信息 | 查看医生接诊 | 管理员可以浏览医生的所有的就诊信息 |
| 51 | 管理员 | 查看综合的医生排名 | 医生排名 | 管理员有权限查看所有医生的综合排名信息。 |

# Value Definition

在这一部分，我们对系统的价值进行了分析，使用MOV进行具体的量度。主要从以下四个方面，资金，客户，内部流程和学习与成长进行说明。

## Area(s) of impact

Financial(资金)

Customer(客户)

Internal Process(内部流程)

Learning and growth(学习与成长)

## Potential Value of the IT solution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Financial | Customer | Internal Process | Learning and growth |
| 政府 | 政府在医疗行业中的投入可以得到更大的回报。 | 对就诊的运行状态更满意。 |  | 能够和政府保持良好的关系。 |
| 医生 | 因为有更多的医生就诊，能够在一定程度上增加收入。 | 能够与病人关系更加和睦。 |  | 可以帮助更多需要帮助的人。 |
| 病人 | 能够在一定程度上减少就诊过程中无用的时间和节省金钱。 | 对医疗机构提供的服务更加满意。 | 病人能够更加方便的寻求医疗协助。 | 将会有更少的人觉得就医困难。 |
| 管理员 | 减少冗余的就诊流程使医院减少支出，能够更多的改进医疗设备和相应的服务。 | 减少额外无用的人力资源开销，客户更加高兴和灵活。 | 能够更加简单的管理有形或者无形的人力资源或其他资产。 | 医疗流程更有组织化 |

## Time Frame for achieving the MOV

接着我们定义了Time Frame 如下

|  |  |  |
| --- | --- | --- |
| years | Specific operations | targets |
| Year 1 | Spend half of a year to modify and update the platform. Spend the last half to pre-run the platform and record the number of successful medical cases. | Eliminate duplicate process by 10%.  Save cost by 10%  Receive more patients by 5% |
| Year 2 | Complete the platform and find more requirements. | Save cost by 15%  Get profit by the second year  Receive more patients by 10% |
| Year 3 | Finish the platform so that improve reliability and usability of the platform. | Get more profit by 10% |

# 3. Feature Mining and Descriptions

## 3.1 Feature Definition

通过对前面user story的定义和整理，我们对story进行了分类，然后得到如下的feature

划分和定义。每一个feature可能包含有多个user story，接着我们对每一个归纳出来的feature进行了具体的定义和流程图的说明。

### 3.1.1 Feature Classification

|  |  |
| --- | --- |
| feature name | feature content（user stories） |
| 信息维护 | 2,3,4,5,6,7,8,12,13,14,26,27,28,30,32,33,38,39,42,43,  44,47,48,49,50 |
| 智能提示挂号 | 9,10,11,16,41 |
| 网上在线支付 | 17,18,19,20, |
| 预约挂号分流 | 15,24,29,31,35,46 |
| 医生绩效评价 | 21,22,23,25,28,34,36,37,40,45 |
| 医生排名智能排序 | 51,52 |

### 3.1.2 Feature Definition

|  |  |
| --- | --- |
| 信息安全与维护 | 在这一部分，我们将录入医院所有医生和病人的信息，完成系统数据库与医院信息数据库的对接工作。管理员可以通过系统更新和维护医生和病人的信息。 |
| 智能提示挂号 | 让患者用户通过做测试的方式，系统智能根据测试的结果推荐挂号的科室和相关科室的医生。防止用户因为挂错科室，延误病情，损失钱财。 |
| 网上在线支付 | 患者用户可以直接在线支付挂号订单，并能够查询历史挂号订单信息和取消已经预约的挂号单。免去了病人排长队挂号的窘境。 |
| 预约挂号分流 | 系统使用独特的排队分流算法，对已激活预约状态的病人进行医生的接诊分配。避免出现一个医生很忙，其他医生被闲置的情况。 |
| 医生绩效评价 | 为了保证数据的有效性，在这一部分，需要进行实时的调研，确保系统评价医生的绩效水平考虑因素的准确性和合理性，最终产生总的绩效排名，可作为医院表彰，提升优秀医生的依据 |
| 管理员显示医生排名 | 医生排名最终只呈现给管理员，系统设计人员通过调研最终确认医生排名的标准和算法，系统最终将产生的排名信息显示给管理员，作为管理员监督医生工作的信息来源之一。 |

## 3.2 Detailed description (blueprint)

### 3.2.1 信息维护与安全

信息维护与安全特点描述：息维护功能是对系统正常运作的一种保障，这里信息维护功能不仅要保证病人和医生信息的正确性和安全性，还要保证系统中病人和医生各种交互信息的实时性。在系统中不同身份的用户拥有不同的功能和权限，因而不同身份的人只能选择相应的身份登入系统进行操作。在系统的运作中，信息的更新及维护有着至关重要的作用，很有必要对此进行功能分析和处理。

具体特点的功能流程图如下：

****图 3.2.1 信息维护与安全特点流程图

### 3.2.2 智能提示挂号

智能提示挂号特点具体描述：智能提示挂号功能是在用户进行智能测试后系统帮助病人匹配到最合适的挂号科室，帮助病人解决了“三长一短”问题中挂号时间长的问题。病人在进行相应的题目回答后，即可或者系统推荐的挂号科室，用户也可自己进行重新筛选，然后确定合适的科室挂号。在选择科室完毕后，病人可让系统匹配相应的医生，也可自己选择特定的医生进行挂号。然后进行时间的预约，当医生的预约人数截止后，病人需更改预约日期或者重新选择医生进行挂号。

具体特点的功能流程图如下：

****图 3.2.2 智能提示挂号特点流程图

### 3.2.3 网上在线支付

网上支付特点具体描述：网上支付功能使病人能够在线进行挂号预约的支付，解决了医疗系统“三长一短”中缴费时间长的问题，大大节省了病人等候的时间，并提高了医疗系统运作的效率。病人在提交预约订单后，确认进行支付后，可直接在网上选择支付宝，网银或者微信支付进行支付。支付的高效，是系统高效的重要一部分。

具体特点的功能流程图如下：



图 3.2.3 网上在线支付特点流程图

### 3.2.4 预约挂号分流

预约挂号分流特点具体描述：预约挂号分流是系统为了平衡所有医生工作量而进行病人挂号的智能分配。对病人挂号的分流，不仅仅缩短了病人实现挂号的时间，同时也平衡了所有医生的工作量，有效地利用了医院的所有的医疗资源，大大地提高了医院各部门的运行效率。在完成预约的条件下病人进行挂号激活，如果此时在该医生已激活等待的人数超过设置的标准，系统将提示病人进行重新分配医生的提议，进入等待人数较少的医生队列中。智能分流有效地解决了“三长一短”中等待时间长的问题，不仅给病人带来方便，也给医生减轻了压力。

具体特点的功能流程图如下：

****

图 3.2.4 预约挂号分流特点流程图

### 3.2.5 医生绩效评价

医生绩效评价特点描述：对医生的绩效评价是通过该周期对医生工作量和病人评价指数进行的综合统计评比，该评比具体内容不对外公开，但结果在医院进行优秀医生评比及奖励有着重要的地位。通过对医生的综合评价，不仅是对医生工作的一种监督，更是对医生工作的一种激励。在病人结束就诊后，可对医生进行公开的评价，如果医生对评价存在异议，可向管理员提起申述，否者评价将录入系统。在周期结束后，系统会重新进入下一轮新的评价。

具体特点的功能流程图如下：

****

图 3.2.5 医生绩效评价特点流程图

### 3.2.6 管理员显示医生排名

管理员督查医生排名的描述：管理员对医院排名的督查是通过管理员对医生评价的监督，避免出现虚假评价。这不仅关系到医生间的公平公正评比，也关系到病人对医院及医生的信任。所以通过对医生评比排名的正确监督管理有着不可或缺的地位。

## 3.3 Identify the values for each Feature provided.

在这一模块，我们为系统能够提供的特点逐一定义它们的价值。具体如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Features | 病人 | 医生 | 系统管理员 |
| 信息安全与维护 |  |  | 方便管理员对系统的信息进行验证和管理 |
| 智能提示挂号 | 指导病人正确挂号，方便病人就诊 |  | 减少不必要的纠纷和医疗延误事故 |
| 网上在线支付 | 方便病人缴费，缩短病人排队就诊的时间 |  |  |
| 预约挂号分流 | 在一定程度上增加病人的就诊时间，同时可以体改就诊效率 | 避免出现普通医生工作负荷不平衡的情况 |  |
| 医生绩效评价 | 让病人参与到就诊医生的评价中，缓解医患关系 | 激励医生，同时评分的透明化确保了公平公正的原则 | 可作为管理则表彰和提升优秀医生的依据 |
| 管理员显示医生排名 |  | 确保医生的工作能够受到监督和管理 | 方便管理者监督医生的工作状态 |

## 3.4 Ways to verify or validate the feature in the future

如下表格列出了为了能够检验每一个feature实现情况的相关TC，保证后期检验工作的顺利进行。该表格是对检验工作的初步定义，会在产品生产周期中不断更新和改进。

|  |  |  |
| --- | --- | --- |
| Test Case ID | Test Case | Coverage |
| TC1 | 登入未注册用户 | 覆盖特点1 |
| TC2 | 输入已存在用户，登入错误密码 | 覆盖特点1 |
| TC3 | 输入正确的用户及密码 | 覆盖特点1 |
| TC4 | 查询不存在的医生信息 | 覆盖特点1 |
| TC5 | 查询系统中的医生信息 | 覆盖特点1,5,6 |
| TC6 | 病人点击编辑自己的附加信息 | 覆盖特点1 |
| TC7 | 病人点击更改自己的真实信息 | 覆盖特点1 |
| TC8 | 一个病人点击提交一个挂号预约订单 | 覆盖特点1,4 |
| TC9 | 删除已预约挂号预约订单 | 覆盖特点1,2 |
| TC10 | 一个病人提交多个挂号预约订单 | 覆盖特点1,4 |
| TC11 | 管理员点击删除已存在的病人信息 | 覆盖特点1 |
| TC12 | 管理员点击更改医生的真实信息 | 覆盖特点1 |
| TC13 | 删除已存在的医生信息 | 覆盖特点1 |
| TC14 | 病人发送消息至在线医生 | 覆盖特点1,5 |
| TC15 | 病人发送消息至离线医生 | 覆盖特点1,5 |
| TC16 | 病人点击评价已完成订单 | 覆盖特点1,5 |
| TC17 | 医生点击对评价提交申诉 | 覆盖特点1,5 |
| TC18 | 医生点击确认就诊结束 | 覆盖特点1,5 |
| TC19 | 病人点击选择智能挂号测试 | 覆盖特点2 |
| TC20 | 病人未完成全部智能挂号测试点击提交 | 覆盖特点2 |
| TC21 | 病人点击取消智能挂号测试 | 覆盖特点2 |
| TC22 | 病人点击选择智能挂号结果进行挂号 | 覆盖特点2 |
| TC23 | 病人选择跳过智能挂号结果进行挂号 | 覆盖特点2 |
| TC24 | 对已预约挂号点击付款 | 覆盖特点3 |
| TC25 | 在选择一种付款方式后点击取消重新选择付款方式 | 覆盖特点3 |
| TC26 | 付款账号账户余额不足 | 覆盖特点3 |
| TC27 | 病人未支付完成点击支付完成 | 覆盖特点3 |
| TC28 | 多人预约在同个医生选择同日挂号 | 覆盖特点4 |
| TC29 | 医生点击查看预约列表 | 覆盖特点1 |
| TC30 | 医生在在线咨询栏输入回复内容 | 覆盖特点1 |
| TC31 | 管理员点击查看医生绩效评价 | 覆盖特点1,5 |
| TC32 | 管理员点击查看医生综合排名 | 覆盖特点1,6 |
| TC33 | 病人点击选择智能匹配医生 | 覆盖特点4 |
| TC34 | 病人取消智能匹配医生，点击重新选择 | 覆盖特点4 |
| TC35 | 编辑信息输入不符合格式要求的内容 | 覆盖特点1 |
| TC36 | 病人选择不符合预约条件的日期 | 覆盖特点1,2 |
| TC37 | 病人预约后至预约日未到医院签到激活 | 覆盖特点1,4 |
| TC38 | 病人确认就诊后不对医生进行评价 | 覆盖特点1，5 |
| TC39 | 病人就诊评价后，选择修改评论 | 覆盖特点1，5 |
| TC40 | 注册已存在用户 | 覆盖特点1 |

# 4. Features Priorities Identification

## 4.1 Features value comparison (valued between 1, 6)

由于每一个feature能够给客户提供的价值不同，根据前面部分的价值分析，我们对每一个feature的价值进行了如下的定义。

|  |  |
| --- | --- |
| Features | value |
| 医生绩效评价 | 5 |
| 预约挂号分流 | 5 |
| 网上在线支付 | 4 |
| 智能提示挂号 | 6 |
| 管理员显示医生排名 | 2 |
| 信息维护与安全 | 3 |

在所有的特点中，智能提示挂号能够给客户提供的价值最大，其次是医生绩效评价和预约挂号分流，接着是网上在线支付，最后是信息安全与维护和管理员显示医生排名。

## 4.2 Customer Satisfaction comparison

在产品迭代开发的过程中，不同特点的开发先后顺序也会导致用户满意度的高低不一，我们对不同特点开发先后顺序的客户满意度进行比较，综合得到能够让客户满意度最高的feature 优先级顺序，如下是具体的比较过程。

图 4.2.1 产品所有的特点

Step 1：对这6个Features先两两排序，但是所有的这些都要基于信息维护这个feature,所以先来两两排序：

2

5

1

3

4

得到第一个排序:

Step 2：3个Features进行排序:

4

1

2

3

得到第二个排序:

Step 3：4个Features进行排序:

2

3

1

得到的排序:

Step 4： 剩下的继续排序

2

1

之后得到:

最后得到了一个满意度最高的排序:

## 4.3 Complex comparison

对前面两个比较的结果进行综合比较，得到如下的表格

|  |  |  |  |
| --- | --- | --- | --- |
| Features |  | | Summary |
| 医生绩效评价 | value(0.3) | 5 | 2.9 |
| Satisfaction(0.7) | 2 |
| 预约挂号分流 | value(0.3) | 5 | 5 |
| satisfaction(0.7) | 5 |
| 在线支付 | value(0.3) | 4 | 3.3 |
| satisfaction(0.7) | 3 |
| 智能提示挂号 | value(0.3) | 6 | 4.6 |
| satisfaction(0.7) | 4 |
| 管理员显示医生的排名 | value(0.3) | 2 | 1.3 |
| satisfaction(0.7) | 1 |
| 信息维护与安全 | value(0.3) | 3 | 5.1 |
| satisfaction(0.7) | 6 |

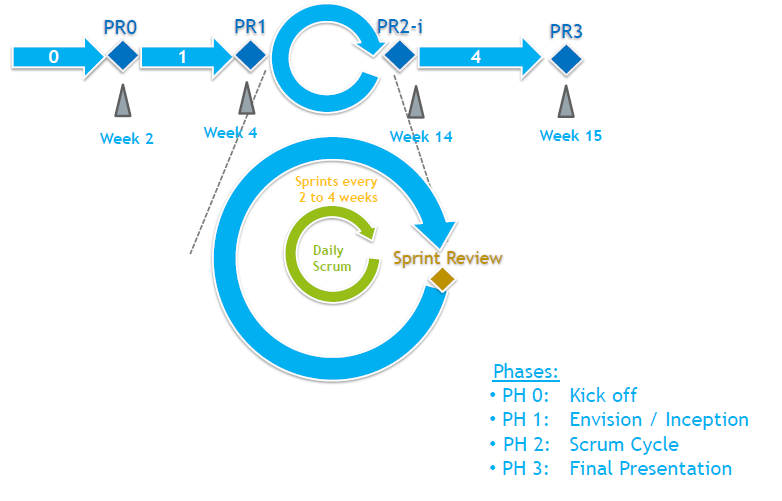
## 4.4 Give the features priorities from high to low

## 4.5 Summary and conclusion

通过features们的价值和客户的满意度,再经过它们之间的权重组合我们最后得到了这个feature的优先级排序,由于我们团队认为客户满意度的权重更加的高,所以在组合中它占了0.7,而价值只占了0.3.这个优先级排序能帮助我们确定开放的优先级和次序.我们在发现features价值时是通过我们团队成员站在用户角度时价值比较，而我们在进行客户满意度调查时使用的排序方式是：我们先进行两两比较,得出最好组合,再在最好的两两组合中比较三三组合的最优,依次往下最终得到我们的满意度排序.

# 5. Road Map of Product

### 5.1 Road Map – Project Life Cycle



### 5.2 Envision/ Inception – Road Map

在这一部分，我们对产品的release Plan进行计划。我们根据文档前面定义的feature数量和开发的实际情况，最后决定产品分6次进行发布，每一次发布，产品都在前面的基础上加入新的功能内容。具体的内容如下。

* Final release
* Database connection
* All the functions in the feature of information maintenance and safety
* All the functions in the feature of doctor’s performance appraisal.
* All the functions in the feature of online payment
* All the functions in the feature of intelligent hint reservation
* All the functions in the feature of reservation and flow cut.

# 6. Product Backlog

## 6.1 Give the chart of your product backlog

根据前面分析的user story和features，对product backlog进行了制定，因为版面有限，所以这里只是粘贴了大概的内容，详细内容参见excel表格：[backlog and sprints.xlsx](file:///C:\Users\Administrator.PC-20150209NFHA\Desktop\backlog%20and%20sprints.xlsx)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **特性** | **故事ID** | **用户故事** | **任务编号** | **任务** | **Sprint** | **耗时（小时）** |
| 信息维护 | 1 | 用户注册 | 1 | 注册界面设计 | 1 | 4 |
| 2 | 连接数据库获取数据 | 1 | 4 |
| 3 | 服务端处理注册信息并返回注册结果 | 1 | 2 |
| 2 | 用户登录系统 | 4 | 登录界面设计 | 1 | 2 |
| 5 | 客户端对登录信息进行判断过滤 | 1 | 2 |
| 6 | 服务端处理登录信息并返回登录结果 | 1 | 4 |
| 3 | 病人查询医生信息 | 7 | 查询界面设计 | 1 | 4 |
| 8 | 服务器获取查询信息，读数据库返回查询结果 | 1 | 6 |
| 9 | 客户端获取查询结果信息并在屏幕上显示 | 1 | 4 |
| 4 | 病人找回密码 | 10 | 找回密码界面设计 | 1 | 2 |
| 11 | 服务端验证账户信息 | 1 | 6 |
| 5 | 病人编辑自己的信息 | 12 | 界面设计 | 1 | 4 |
| 13 | 客户端更新信息发送到服务端更新保存 | 1 | 4 |
| 6 | 病人修改自己的信息 | 14 | 界面设计 | 1 | 4 |
| 15 | 服务端发送病人目前的信息到客户端显示 | 1 | 6 |
| 16 | 客户端更新信息发送到服务端更新保存 | 1 | 6 |
| 7 | 病人删除自己的信息 | 17 | 界面设计 | 1 | 8 |
| 18 | 客户端分类显示病人的信息 | 1 | 4 |
| 19 | 服务端响应病人删除操作，更新数据库 | 1 | 8 |
| 11 | 系统显示在线的医生 | 20 | 界面设计 | 1 | 6 |
| 21 | 服务器读取在线医生信息返回给客户端 | 1 | 4 |
| 22 | 客户端获取数据分类处理显示 | 1 | 4 |
| 12 | 系统可以接受离线的医生回复信息 | 23 | 界面设计 | 1 | 4 |
| 24 | 服务器读取未读的消息返回给客户端 | 1 | 6 |
| 25 | 客户端接收消息并显示 | 1 | 8 |
| 13 | 系统能够保障用户的信息 | 26 | 登录时，当用户密码输入超过5次，则锁定账户，并通知用户 | 1 | 6 |
| 25 | 普通医生登录系统 | 27 | 登录节界面设计 | 1 | 4 |
| 28 | 客户端对登录信息进行判断过滤 | 1 | 6 |
| 29 | 服务端处理登录信息并返回登录结果 | 1 | 6 |
| 26 | 普通医生查看自己的个人信息 | 30 | 界面设计 | 1 | 4 |
| 31 | 从服务器获取数据显示 | 1 | 4 |
| 29 | 普通医生接诊时系统能够识别病人的身份 | 32 | 就诊卡刷卡器进行连接，获取病人编号 | 1 | 5 |
| 33 | 根据病人编号获取病人信息显示 | 1 | 4 |
| 31 | 系统能为普通医生显示用户的在线咨询信息 | 34 | 界面设计 | 1 | 4 |
| 35 | 从数据库读取咨询信息显示 | 1 | 3 |
| 32 | 普通医生可以回复咨询信息 | 36 | 回复界面设计 | 1 | 4 |
| 37 | 将回复内容保存到数据库，返回结果 | 1 | 4 |
| 37 | 专家医生登录系统 | 38 | 登录界面设计 | 1 | 4 |
| 39 | 客户端对登录信息进行判断过滤 | 1 | 4 |
| 40 | 服务端处理登录信息并返回登录结果 | 1 | 4 |
| 38 | 专家医生查看自己的个人信息 | 41 | 界面设计 | 1 | 3 |
| 42 | 从服务器获取数据显示 | 1 | 4 |
| 41 | 专家医生查看已接诊的病人列表 | 43 | 界面设计 | 1 | 3 |
| 44 | 从服务器获取数据显示 | 1 | 4 |
| 42 | 专家医生查看接诊的具体信息 | 45 | 界面设计 | 1 | 3 |
| 46 | 从服务器获取数据显示 | 1 | 4 |
| 43 | 系统为专家医生识别病人的身份 | 47 | 同普通医生 | 1 | 4 |
| 46 | 管理员录入病人的信息 | 48 | 界面设计 | 1 | 4 |
| 49 | 服务器对导入数据进行处理存数据库 | 1 | 4 |
| 47 | 管理员录入医生的信息 | 50 | 界面设计 | 1 | 3 |
| 51 | 服务器对导入数据进行处理存数据库 | 1 | 3 |
| 48 | 管理员修改医生的信息 | 52 | 界面设计 | 1 | 4 |
| 53 | 客户端发送更新信息请求 | 1 | 3 |
| 54 | 服务器保存新的信息并返回结果 | 1 | 3 |
| 49 | 管理员删除医生信息 | 55 | 界面设计 | 1 | 5 |
| 56 | 客户端发送删除信息请求 | 1 | 5 |
| 57 | 服务器删除信息并返回结果 | 1 | 4 |
| 预约挂号分流 | 14 | 系统能够在在线用户高峰时期，在10s内反馈用户信息 | 58 | 系统能够智能处理用户请求数据，保证用户正常使用不同模块的功能 | 2 | 20 |
| 23 | 系统能够提示已经预约满员，无法预约的信息 | 59 | 系统读取预约信息，若预约人数达到上限，则提示且禁止预约 | 2 | 2 |
| 28 | 普通医生查看系统自己分配的接诊病人列表 | 60 | 界面设计 | 2 | 4 |
| 61 | 叫号系统分配病人算法设计 | 2 | 10 |
| 62 | 获取就诊病人的信息并显示 | 2 | 8 |
| 30 | 系统能够在普通医生点击结束就诊后，重新生成新的待接诊列表 | 63 | 界面设计 | 2 | 4 |
| 64 | 刷新待接诊列表 | 2 | 4 |
| 34 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾 | 65 | 算法和实现设计 | 2 | 15 |
| 45 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾 | 66 | 同普通医生 | 2 | 4 |
| 智能提示挂号 | 8 | 系统能够智能提示用户挂号的科室 | 67 | 自主测试界面设计 | 3 | 10 |
| 68 | 自主测试算法及分析规则设计 | 3 | 20 |
| 69 | 测试结果与科室匹配推荐算法设计 | 3 | 20 |
| 9 | 系统能够智能推荐用户可选的医生 | 70 | 推荐医生显示界面设计 | 3 | 6 |
| 71 | 获得推荐医生算法设计 | 3 | 20 |
| 10 | 系统提示所选医生可挂号的时间 | 72 | 客户端将所选医生编号发送到服务器 | 3 | 10 |
| 73 | 服务端从数据库获取医生排班信息返回给客户端 | 3 | 5 |
| 15 | 病人提交预约 | 74 | 预约界面设计 | 3 | 4 |
| 75 | 客户端向服务器提交预约请求 | 3 | 4 |
| 76 | 服务端处理预约请求，返回预约结果 | 3 | 4 |
| 40 | 专家医生查看待接诊的病人列表 | 77 | 界面设计 | 3 | 4 |
| 78 | 客户端从服务端获取数据显示 | 3 | 4 |
| 网上在线支付 | 16 | 在线支付预约订单 | 79 | 连接支付宝和银行网上支付的接口 | 4 | 10 |
| 80 | 支付结果处理 | 4 | 4 |
| 17 | 查看用户的历史订单信息和待办订单和已办订单信息 | 81 | 界面设计 | 4 | 4 |
| 82 | 服务器读取数据库信息返回给客户端 | 4 | 2 |
| 83 | 客户端接收信息处理显示 | 4 | 2 |
| 18 | 用户可以取消未支付待确定的订单 | 84 | 客户端发送取消订单请求 | 4 | 2 |
| 85 | 服务器处理请求返回结果 | 4 | 2 |
| 医生绩效评价 | 20 | 用户可以从不同的角度评价接诊自己的医生 | 86 | 界面设计，列出各个方面供用户打分，并填写评价内容 | 5 | 6 |
| 87 | 系统获取保存评价内容，汇总分析 | 5 | 6 |
| 21 | 用户可以选择删除自己的评价 | 88 | 界面设计 | 5 | 4 |
| 89 | 客户端发送删除评价请求 | 5 | 4 |
| 90 | 服务端处理请求返回结果 | 5 | 4 |
| 22 | 用户可以选择修改自己的评价 | 91 | 服务端读取原评价信息显示在客户端 | 5 | 3 |
| 92 | 客户端更新评价信息，将信息的信息发送给服务端 | 5 | 3 |
| 93 | 服务端接收更新信息保存并返回更新结果 | 5 | 4 |
| 24 | 系统能够显示当前医生的评价分数 | 94 | 在医生信息界面，读取医生评价分数显示 | 5 | 2 |
| 27 | 普通医生查看自己的评价信息 | 95 | 界面设计 | 5 | 2 |
| 96 | 获取评价详细信息进行分类显示 | 5 | 2 |
| 33 | 普通医生可以举报不良的用户评价信息 | 97 | 举报界面设计 | 5 | 4 |
| 98 | 将举报内容发送到服务器保存 | 5 | 2 |
| 35 | 系统根据普通医生在线的时间，确定当天工作时长 | 99 | 算法和实现设计 | 5 | 20 |
| 36 | 系统实行评价一月制，及每周系统自动将以前的用户评价抹掉，供下周用户重新进行评价 | 100 | 算法和实现设计 | 5 | 20 |
| 39 | 专家医生查看自己的评价信息 | 101 | 界面设计 | 5 | 4 |
| 102 | 获取评价详细信息进行分类显示 | 5 | 6 |
| 44 | 专家医生举报不良的用户评价信息 | 103 | 同普通医生 | 5 | 4 |
| 医生排名智能排序 | 50 | 管理员查看医生接诊信息 | 104 | 界面设计 | 6 | 4 |
| 105 | 从服务器获取数据显示 | 6 | 4 |
| 51 | 管理员查看综合医生排名 | 106 | 界面设计 | 6 | 4 |
| 107 | 排名公式算法设计 | 6 | 20 |
| 108 | 获取排名显示 | 6 | 6 |

# 7. Sprints Definition & Sprint Backlog

## 7.1 Give the chart of sprints for each of your features

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Description** | **Sprint #** | 1 | 2 | 3 | 4 | 5 | 6 |
|  |  | | 245 | 77 | 111 | 26 | 88 | 38 |
| 1 | 用户注册 |  | 10 | 0 | 0 | 0 | 0 | 0 |
| 2 | 用户登录系统 |  | 8 | 0 | 0 | 0 | 0 | 0 |
| 3 | 病人查询医生信息 |  | 14 | 0 | 0 | 0 | 0 | 0 |
| 4 | 病人找回密码 |  | 8 | 0 | 0 | 0 | 0 | 0 |
| 5 | 病人编辑自己的信息 |  | 8 |  |  |  |  |  |
| 6 | 病人修改自己的信息 |  | 16 | 0 | 0 | 0 | 0 | 0 |
| 7 | 病人删除自己的信息 |  | 20 | 0 | 0 | 0 | 0 | 0 |
| 11 | 系统显示在线的医生 |  | 14 | 0 | 0 | 0 | 0 | 0 |
| 12 | 系统可以接受离线的医生回复信息 |  | 18 | 0 | 0 | 0 | 0 | 0 |
| 13 | 系统能够保障用户的信息 |  | 6 | 6 | 0 | 0 | 0 | 0 |
| 25 | 普通医生登录系统 |  | 16 |  |  |  |  |  |
| 26 | 普通医生查看自己的个人信息 |  | 8 | 0 | 0 | 0 | 0 | 0 |
| 29 | 普通医生接诊时系统能够识别病人的身份 |  | 9 | 0 | 0 | 0 | 0 | 0 |
| 31 | 系统能为普通医生显示用户的在线咨询信息 |  | 7 | 0 | 0 | 0 | 0 | 0 |
| 32 | 普通医生可以回复咨询信息 |  | 8 | 0 | 0 | 0 | 0 | 0 |
| 37 | 专家医生登录系统 |  | 12 | 0 | 0 | 0 | 0 | 0 |
| 38 | 专家医生查看自己的个人信息 |  | 7 | 0 | 0 | 0 | 0 | 0 |
| 41 | 专家医生查看已接诊的病人列表 |  | 7 | 0 | 0 | 0 | 0 | 0 |
| 42 | 专家医生查看接诊的具体信息 |  | 7 | 0 | 0 | 0 | 0 | 0 |
| 43 | 系统为专家医生识别病人的身份 |  | 4 | 0 | 0 | 0 | 0 | 0 |
| 46 | 管理员录入病人的信息 |  | 8 | 0 | 0 | 0 | 0 | 0 |
| 47 | 管理员录入医生的信息 |  | 6 | 0 | 0 | 0 | 0 | 0 |
| 48 | 管理员修改医生的信息 |  | 10 | 0 | 0 | 0 | 0 | 0 |
| 49 | 管理员删除医生信息 |  | 14 | 0 | 0 | 0 | 0 | 0 |
| Sprint1 | 信息安全与维护 |  |  |  |  |  |  |  |
| 14 | 系统能够在在线用户高峰时期，在10s内反馈用户信息 |  | 0 | 20 | 0 | 0 | 0 | 0 |
| 23 | 系统能够提示已经预约满员，无法预约的信息 |  | 0 | 2 | 0 | 0 | 0 | 0 |
| 28 | 普通医生查看系统自己分配的接诊病人列表 |  | 0 | 22 | 0 | 0 | 0 | 0 |
| 30 | 系统能够在普通医生点击结束就诊后，重新生成新的待接诊列表 |  | 0 | 8 | 0 | 0 | 0 | 0 |
| 34 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾 |  | 0 | 15 | 0 | 0 | 0 | 0 |
| 45 | 系统出现叫号三次病人未到的情况下，自动将该用户的预约放置队列尾 |  | 0 | 4 | 0 | 0 | 0 | 0 |
| Sprint2 | 预约挂号分流 |  |  |  |  |  |  |  |
| 8 | 系统能够智能提示用户挂号的科室 |  | 0 | 0 | 50 | 0 | 0 | 0 |
| 9 | 系统能够智能推荐用户可选的医生 |  | 0 | 0 | 26 | 0 | 0 | 0 |
| 10 | 系统提示所选医生可挂号的时间 |  | 0 | 0 | 15 | 0 | 0 | 0 |
| 15 | 病人提交预约 |  | 0 | 0 | 12 | 0 | 0 | 0 |
| 40 | 专家医生查看待接诊的病人列表 |  | 0 | 0 | 8 | 0 | 0 | 0 |
| Sprint3 | 智能提示挂号 |  |  |  |  |  |  |  |
| 16 | 在线支付预约订单 |  | 0 | 0 | 0 | 14 | 0 | 0 |
| 17 | 查看用户的历史订单信息和待办订单和已办订单信息 |  | 0 | 0 | 0 | 8 | 0 | 0 |
| 18 | 用户可以取消未支付待确定的订单 |  | 0 | 0 | 0 | 4 | 0 | 0 |
| Sprint4 | 网上在线支付 |  |  |  |  |  |  |  |
| 20 | 用户可以从不同的角度评价接诊自己的医生 |  | 0 | 0 | 0 | 0 | 12 | 0 |
| 21 | 用户可以选择删除自己的评价 |  | 0 | 0 | 0 | 0 | 12 | 0 |
| 22 | 用户可以选择修改自己的评价 |  | 0 | 0 | 0 | 0 | 10 | 0 |
| 24 | 系统能够显示当前医生的评价分数 |  | 0 | 0 | 0 | 0 | 2 | 0 |
| 27 | 普通医生查看自己的评价信息 |  | 0 | 0 | 0 | 0 | 4 | 0 |
| 33 | 普通医生可以举报不良的用户评价信息 |  | 0 | 0 | 0 | 0 | 6 | 0 |
| 35 | 系统根据普通医生在线的时间，确定当天工作时长 |  | 0 | 0 | 0 | 0 | 20 | 0 |
| 36 | 系统实行评价一月制，及每周系统自动将以前的用户评价抹掉，供下周用户重新进行评价 | | 0 | 0 | 0 | 0 | 10 | 0 |
| 39 | 专家医生查看自己的评价信息 |  | 0 | 0 | 0 | 0 | 4 | 0 |
| 44 | 专家医生举报不良的用户评价信息 |  | 0 | 0 | 0 | 0 | 8 | 0 |
| Sprint5 | 医生绩效评价 |  |  |  |  |  |  |  |
| 50 | 管理员查看医生接诊信息 |  | 0 | 0 | 0 | 0 | 0 | 8 |
| 51 | 管理员查看综合医生排名 |  | 0 | 0 | 0 | 0 | 0 | 30 |
| Sprint6 | 医生排名智能排序 |  |  |  |  |  |  |  |

# 7. DOD & DOR Definition

## 7.1 The Definition of DOR

|  |  |  |
| --- | --- | --- |
| 用户故事 | 迭代开发 | 版本发布 |
| 1. 确定了用户故事的依赖关系 2. 用户故事的验收标准确定了 3. 接收用户故事的人是确定的 4. 用户故事的最终结果很明确 | 1. 计算上一次迭代的速度 2. 上一次迭代的故事是完整的或者仍在待办事件表中搁置 | 1. 确定了发布人员 2. 确定了发布的主题 3. 确定了发布日期 |

## 7.2 The Definition of DOD

用户故事:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 第一部分 | 用户故事清晰 | 任务被确认 | 建立设置改变 | 产品拥有者通过 | 更新产品待办事件列表 |
|  | 为Sprint设计的用户故事尊重产品主题，被团队理解并且通过具体的验收标准的验证。 | 指定用户故事的任务被确认，并且被团队估量。 | 构建和包的变化传达到构建负责人。这些变化已经实现,测试和记录,以确保它们覆盖sprint的特性。 | 每个完成的用户故事都通过了用户验收测试，并署名为会议的需求。 | 1. 所有没有在sprint中实现的功能，添加到产品待办事项列表中。 2. 所有没有在sprint中处理的事件/缺陷添加到产品待办事项列表中。 |
| 第二部分 | 环境准备好 | 设计完成 | 单元测试样例完成 | 文件写完 | 构建预发行版本 |
|  | 1. 开发环境配置完成。 2. 阶段环境配置完成。 3. 完成持续集成框架搭建。构建引擎配置安排每小时,每晚构建和发布。 4. 需要的构建自动化完成。 5. 创建了指定功能的测试数据。 | 完成了每个用户故事的设计分析，开发时每个功能的UML图都创建了。 | 待开发的每个功能的单元测试样例已经完成。 | 完成支持Sprint演示的文件。 | 着手准备预发行版本，并且构建报告定期发布。 |
| 第三部分 | 完成源码 | 单元测试完成 | 代码重构 | 代码检验 | 代码合并与标签 |
|  | 记录下在待办事件表里的所有功能的源码改变，源码适当添加了注释。 | 单元测试样例运行成功。 | 源代码已经被重构,使其全面、可维护的,易于改变。 | 在代码库中添加适量注释检查源代码。 | 完成的源代码已经与主干合并并适当地标记。 |
| 第四部分 | 自动代码审查 | 同行审查 | 代码覆盖率 | 项目度量 | 构建发行 |
|  | 通过使用工具/技术支持，自动代码审查已经完成。代码差异已经与团队共享，团队解决了所有差异以坚持编码标准。 | 同行评审完成。 | 每个包的代码覆盖率记录可用，无论团队如何决策，已经达到最低标准。 | 燃尽图一直定期更新,并且是最新的。 | 1. 构建和包装：使用持续集成框架构建成功。源码库生成更改日志报告和发行说明，可交付成果已移交到发行区域。 2. 在阶段环境建立部署：构建可提交物在阶段环境部署交付。 |
| 第五部分 | 完成功能性测试 | 完成回归测试 | 完成性能测试 | 完成验收测试 | 关闭 |
|  | 1. 自动化测试：所有类型的自动化测试用例执行和测试报告已经生成。所有事件/缺陷写成报告。 2. 手工测试：质量保证团队回顾了自动化测试生成的报告并进行了必要的手动测试案例,以确保测试通过。所有事件/缺陷写成报告。 | 为了确保缺陷没有被引入软件的不变区域的回归测试完成。 | 完成性能测试。 | 每个完成的用户故事都通过了用户验收测试，并署名为会议的需求。 | 所有完成的用户故事/标记完成的任务解决。在任务关闭前，任务的剩余时间设置为零。 |

迭代：

1. 最新的 CI编译成功，CI和回归测试通过。

2. 在客户交付时，除非PO同意，否则不存在关键的错误开放。

3. 在客户交付时，由开源检测的问题关闭。

4. Sprint测试计划执行;在适当的时候应该包括回归、压力异常条件和系统测试。

5. 为下一阶段的sprint准备了足够的用户故事。

6. 完成Sprint回顾/样例。

7. 开完Sprint回顾会议，行动点记录下来。

版本发布：

1. 出版发布说明。

2. 记录TRS，API，指导。

3. 软件按照基本方法发布。

4. 发布在安全的平台（GIT，CC，备份的空间等）。

5. 发布在CTS运行后的测试计划，如果相关，考虑客户场景。

6. 零缺陷管理和关键错误开放。

7. 完成安全检验，管理和关闭安全风险。

# 8. Test Plan (Checklist)

## 8.1 The Checklist of your test plan for your product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **活动分类** | **活动内容** | **重要性** | **达到** |
| 1 | 需求 | 项目范围是否清晰明确？ | 高 |  |
| 2 | 需求 | 是否定义了需求开发的过程和方法？ | 高 |  |
| 3 | 需求 | 是否对每类用户进行了描述，包括使用频率、特性、优先级等？ | 高 |  |
| 4 | 需求 | 是否清晰描述了用户业务流程？ | 高 |  |
| 5 | 需求 | 是否对系统进行划分（如果是大型系统，划分为子系统，再划分为模块；如果小型系统，进行模块划分）？ | 高 |  |
| 6 | 需求 | 是否创建了系统原型？ | 低 |  |
| 7 | 需求 | 是否进行风险分析？ | 低 |  |
| 8 | 需求 | 是否对需求进行分类？ | 中 |  |
| 9 | 需求 | 是否记录了目前尚未明确的需求？ | 中 |  |
| 10 | 需求 | 需求文档的上下文有矛盾吗？ | 高 |  |
| 11 | 需求 | 是否根据需求文档撰写测试用例？ | 低 |  |
| 12 | 需求 | 是否定义了需求变更流程？ | 低 |  |
| 13 | 需求 | 需求变更是否进行了记录？ | 低 |  |
| 14 | 需求 | 需求是否足够清晰，以至可以转交给一个独立小组来实现，并能够被理解？ | 高 |  |
| 15 | 需求 | 每个需求是可测试的吗？是否可以通过独立的测试来决定需求是否被满足？ | 高 |  |
| 16 | 设计 | 是否描述系统的所有输入，包括输入源、准确性、取值范围和出现频率？ | 低 |  |
| 17 | 设计 | 是否描述系统的所有输出，包括输出的目标、准确性、取值范围、出现频率和格式？ | 低 |  |
| 18 | 设计 | 是否描述所有硬件和软件的外部接口？ | 低 |  |
| 19 | 效率 | 从用户的角度来看，是否描述了对所有必要操作的预计响应时间？ | 高 |  |
| 20 | 效率 | 是否对时间方面问题进行考虑，如处理时间、数据传输和系统的吞吐量？ | 中 |  |
| 21 | 可靠性 | 是否描述了系统的可靠性，包括软件产生故障的后果、故障后重要数据的保护、错误检测和恢复？ | 高 |  |
| 22 | 可维护性 | 是否详细说明了系统的可维护性，包括适应操作环境变化的能力、与其它软件的接口、精确性、性能和附加的可以预知的性能？ | 高 |  |
| 23 | 通用 | 是否描述了可接受的折中原则，如健壮性和正确性之间的选择？ | 高 |  |
| 24 | 通用 | 有些信息只有到开发时才能获得，是否对这些信息不完全的领域进行描述？ | 中 |  |
| 25 | 功能性 | 病人通过输入身份证注册系统，身份匹配失败，是否有处理？ | 高 |  |
| 26 | 功能性 | 病人通过身份证注册系统，是否只能为自己挂号？ | 高 |  |
| 27 | 功能性 | 登录系统，多次密码、随机码输入错误，是否有处理？ | 高 |  |
| 28 | 功能性 | 病人账户锁定之后，是否有方法再次激活账户？ | 高 |  |
| 29 | 功能性 | 系统并发用户过多时，是否有处理？ | 高 |  |
| 30 | 功能性 | 病人编辑自己的症状，如果病症已经治愈，原来的症状是由病人手动删除，还是系统自动删除？ | 中 |  |
| 31 | 功能性 | 病人删除自己不必要的信息时，是否级联删除？ | 高 |  |
| 32 | 功能性 | 智能挂号提出的测试问题是否层层嵌套？ | 高 |  |
| 33 | 功能性 | 当智能提示预约系统提示可挂不止一个科室的号时，是否有处理？ | 高 |  |
| 34 | 功能性 | 在线医生列表是否有具体分类，比如耳鼻喉科、外科等？ | 高 |  |
| 35 | 功能性 | 咨询在线医生可能会出现排队状况，是否有处理？ | 高 |  |
| 36 | 功能性 | 当病人预约的医生预约名额满时，是否会有提示？ | 高 |  |
| 37 | 功能性 | 在线支付时，病人是否可以选择不同的支付方式？ | 中 |  |
| 38 | 功能性 | 用户在线支付提交订单后，长时间不支付，是否会有处理？ | 高 |  |
| 39 | 功能性 | 用户是否可以取消已支付的订单？ | 高 |  |
| 40 | 功能性 | 用户支付订单后，未去医院就诊，是否对这种情况有处理？ | 高 |  |
| 41 | 功能性 | 用户是否可以无限次取消未支付待确定的订单？ | 中 |  |
| 42 | 功能性 | 用户可以取消未支付待确定的订单三次后，再次取消，是否会有提醒？ | 中 |  |
| 43 | 功能性 | 病人评价医生是完全通过文字评价还是兼备星级评价？ | 中 |  |
| 44 | 功能性 | 病人评价医生时，过于主观，言辞激烈，后台是否会有处理？ | 高 |  |
| 45 | 功能性 | 病人删除对接诊医生的评价时，是否级联删除？ | 高 |  |
| 46 | 功能性 | 普通医生申请管理员更新信息时，管理员受理是否有期限？ | 中 |  |
| 47 | 功能性 | 普通医生是否可以查看自己的排名信息？ | 低 |  |
| 48 | 功能性 | 普通医生申诉病人的评价不合理时，出现作假现象，是否有处理？ | 中 |  |
| 49 | 功能性 | 医生是否有方法确定就诊病人身份？ | 高 |  |
| 50 | 功能性 | 系统识别病人身份，出现不一致时，是否有处理？ | 高 |  |
| 51 | 功能性 | 普通医生动态分配的接诊名单是否随着时间的变化而有改变，比如下班时间、休息时间等？ | 中 |  |
| 52 | 功能性 | 提供在线咨询的医生是固定的一批医生，还是医院的所有医生均提供此服务？ | 中 |  |
| 53 | 功能性 | 根据医生在线时间确定工作时长，万一出现其他状况，如断电，是否有处理？ | 低 |  |
| 54 | 功能性 | 病人评价医生存在几个很不好的评价，是否会一直保留，导致医生排名一直处在下游？ | 中 |  |
| 55 | 功能性 | 如果病人都只预约非专家号，导致专家医生空闲，是否会有处理？ | 高 |  |
| 56 | 功能性 | 管理员录入病人信息时，是否存在再次审核，防止出错？ | 低 |  |
| 57 | 功能性 | 管理员在录入病人信息时，就诊卡的绑定，是之前存在的，还是全新的？ | 低 |  |
| 58 | 功能性 | 管理员删除医生信息时是否级联删除？ | 高 |  |
| 59 | 功能性 | 管理员浏览医生的就诊信息时，是否可以按科室检索？ | 低 |  |
| 60 | 功能性 | 管理员查看的医生的综合排名是全部医生混合科室的排名吗？ | 低 |  |

# 9. Summary

本文档对我们的项目计划进行了详细的定义和整理。在本文档中，我们包含了user story（用户故事），然后定义了feature，给出了feature 优先级的确定方法。然后对确定好的feature规划出了相应的roadmap。接着我们根据feature 和 user story 确定了我们项目的product backlog。在确定好product backlog之后，我们对sprints 进行定义，最后得到 sprints backlog。

文档的最后部分，我们给出了DOD和DOR的定义和详细内容，其中DOD是对最后产品完成情况的一个标准量度，而DOR是在项目计划开发前期的准备工作的定义规范，防止因为前期准备工作不够，或详细，影响后续的开发进程和活动。在Test plan 部分，我们使用了张茜老师给的模板，在确定好的feature的基础上，定义了初步测试要考虑的问题，为以后软件项目测试工程奠定基础。