

2022年春季软件工程 PRJ4需求文档

华为-需求管理跟踪系统

负责助教与联系方式：

- 田倍闻 tbw18@mails.tsinghua.edu.cn
- 刘一凡 liu@yipe.ng

用户代表：

- 华为 骆晓徽
- 华为 周智勇

目录

目录

项目背景

项目简介

角色说明

客户

系统工程师

开发工程师

质量保证（QA）工程师

基础 workflow

原始需求定义

系统架构选型、方案设计

功能需求清单

迭代划分和开发

质量保证（QA）

原始需求列表

项目管理要求

需求规范

MR 信息规范

发布规范

业务流设计参考

注册账号

系统工程师

开发工程师	
质量保证工程师	
逻辑架构参考	
功能分解	
系统架构	
前端UI服务（UI）	
用户管理服务（User Management Service）	
需求管理服务（Requirement Management Service，RMS）	
需求/缺陷回溯服务（Requirement & Defection Trace Service，RDTS）	
数据表设计参考	
用户信息	
原始需求（IR）	
功能需求（SR）	
迭代	
系统服务	
功能需求变更日志（可选）	
服务接口设计参考	
/login	
/verify	
界面设计参考	
外部接口参考	

项目背景

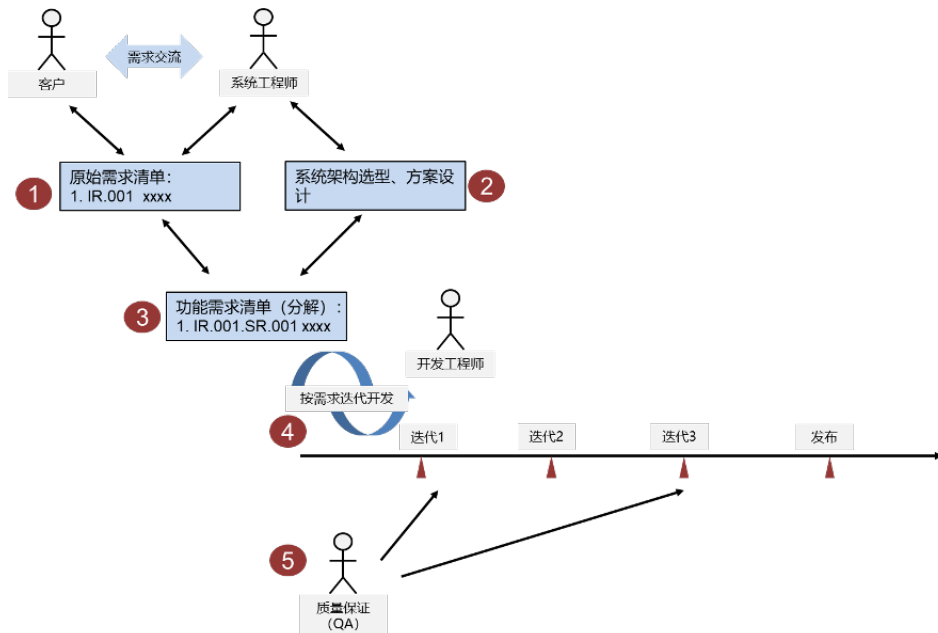
大规模复杂软件的开发需要很多不同分工的软件研发人员进行协同。虽然对不同类型的公司，软件系统架构、软件开发过程有较大的区别，但对于一个软件项目来说，需求的管理和缺陷的跟踪是比较基本的工作流的支撑。在历史上，华为公司曾经使用过很多不同支撑软件开发过程的软件，例如商用的IBM的产品，以及比较常用的开源的Jira，目前因为需要做大量的优化定制，目前基本上都切换成自研支撑软件。

本课题通过参考华为内的研发过程，进行高度简化，实现一个非常精简的软件研发的IT支撑系统，完成基本的**需求管理**和**缺陷跟踪功能、回溯**，对**研发过程进行管理**。

本管理系统的**第一个客户就是其开发项目**。本系统的开发项目需要能够把自己项目的需求管理和缺陷跟踪端到端的支撑起来，也鼓励各开发项目组之间能够相互使用对方的系统，提升项目质量。

用户代表及联系方式：待定

项目简介



角色说明

客户

需要获得软件产品的客户（此项目不需要关注）

系统工程师

负责和客户交流需求、设计总体架构方案，并完成需求/功能的分解分配

开发工程师

根据系统工程师的需求和设计，实现软件开发运行环境的搭建，实现需求和功能的迭代开发

质量保证（QA）工程师

对项目需求开发进展、缺陷解决进展进行跟踪，发现问题和风险

基础 workflow

原始需求定义

由系统工程师完成和客户的交流和定义。

在此项目中，原始需求会直接以原始清单的方式直接提供。需求会包含基本需求和扩展需求。

系统架构选型、方案设计

系统工程师实现编程语言、数据库等选型，并对软件架构方案进行设计。

后面的章节会包含架构参考设计，不同的组可以根据自己的设计，选择合理的实现方案。

功能需求清单

需要将原始需求、根据架构设计，进行分解、分配；将原始需求分解为各个子系统能够独立开发、交付的功能需求点。

注意功能需求清单不仅仅包含外部特性需求，也包含非功能需求，比如数据库的安装等，这样便于识别依赖、进行合理的迭代划分。

注意：需要保证需求清单的完整性，所有的原始需求都需要有分解。

后面的章节会提供功能需求清单的参考样例。

迭代划分和开发

基于功能需求清单，按照合理的节奏进行迭代划分。需要注意的是，尽量每个迭代交付一个完整的需求，所以需要识别清晰功能需求之间的依赖，相互依赖的需求，能够划分到一个迭代统一交付。

所以系统工程师需要和开发工程师交流后，达成一致，然后将对应需求分配给开发人员。

开发人员根据需求进行开发，并在开发/自验完成后，将需求置成完成状态。

一个原始需求所分解的功能需求都完成后，原始需求进入到发布状态。

质量保证（QA）

QA的职责就是负责跟踪项目的进展和质量，通过相关统计报表评估项目进度和质量风险，并且随着安全可信的要求，需要对每次代码的合入都需要可追溯，所以需要：

每个迭代的状态报表

每次代码提交可回溯（需要从git提取信息和需求/缺陷匹配）

原始需求列表

需求编号	需求说明	基础✔ / 扩展★
IR.001	系统工程师 支持 录入原始需求	✔
IR.002	系统工程师 支持 定义软件系统服务	✔
IR.003	系统工程师 支持 分解原始需求到功能需求	✔
IR.004	系统工程师 支持 关联功能需求到各服务	✔
IR.005	系统工程师 支持 划分功能需求到不同开发迭代	✔
IR.006	系统工程师 支持 指定迭代开发责任人	✔
IR.007	开发工程师 支持 查看所有需求、负责的迭代、迭代关联的需求	✔
IR.008	开发工程师 支持 根据交付情况修改功能需求状态：初始、开发中、已交付	✔

IR.009	开发工程师 支持 基于 commit 修改功能需求状态	★
IR.010	质量保证工程师 支持 查看迭代状态、需求状态、需求交付与迭代计划的偏差	✓
IR.011	质量保证工程师 支持 查看定时获取的 Merge Request 信息	✓
IR.012	质量保证工程师 支持 将 MR 与功能需求关联	✓
IR.013	质量保证工程师 支持 查看 自动完成的 MR 与功能需求的关联	★
IR.014	质量保证工程师 支持 查看 功能需求在哪些 MR 交付	✓
IR.015	质量保证工程师 支持 查看 需求交付后的缺陷（issue）统计	✓
IR.016	质量保证工程师 支持 查看 缺陷在哪些 MR 修复	✓
IR.017	质量保证工程师 支持 查看 基于 issue 的需求交付质量分析	★
IR.018	质量保证工程师 支持 查看基于 MR 数量、代码行数的开发工程师活跃度	★
IR.019	质量保证工程师 支持 查看基于 issue 解决时间的开发工程师能力评价	★
IR.020	质量保证工程师 支持 查看代码自动分析结果（代码片段对应的功能需求）	★
IR.021	所有用户 支持 注册，注册信息包含用户名，密码，角色	✓
IR.022	所有用户 支持 修改密码、经负责人同意后修改角色	✓
IR.023	所有用户 认证后才可操作	✓
IR.024	安全性 用户和系统访问使用安全加密通道	✓
IR.025	安全性 不使用明文存储用户密码，不需要还原口令时必须不可逆加密	✓
IR.026	系统规格 1. 支持100个用户同时在线 2. 支持至少1k原始需求、10k功能需求	✓
IR.027	系统设计 一个功能需求只由一个系统服务实现	✓
IR.028	其它 支持 角色更改导致的需求回退	★
IR.029	其它 支持 开发过程需求更改	★
IR.030	其它 数据可视化	★
IR.031	使用自己的项目管理项目开发过程	★

项目管理要求

需求规范

需求定义：分两级，IR 为原始需求，SR 为功能需求。

IR 是用户提出的需求，在粒度上不严格定义大小；在需求开发过程中识别的新需求，可以以IR方式加入； IR 要有编号，且编号不能重复。

SR 为功能需求，在本项目中，为简化操作， SR 只能在一个系统服务实现。SR要具备可测试性，因此注意 SR 的粒度不要过细。对于内部非功能需求（例如数据库创建）也需要分解 SR 。总之，所有实现的功能都要有 SR 对应。

MR 信息规范

为了能从 git log 获取有效信息，需要定义MR合入规范，例如， git 描述信息格式要求如下：

Requirement: SR.001.001 表示本次合入的是 SR.001.001 的代码

Bugfix: Issue 001 表示本次合入的是 Issue 001 的修复代码

Git 描述信息后续会用于需求/缺陷回溯分析使用，该规则是必须执行的。

发布规范

迭代开发、测试完成，进行发布时，需要在 git 仓打上对应 tag ， tag 要携带发布版本信息，例如：

Release-0.0.1 表示 0.0.1 版本发布

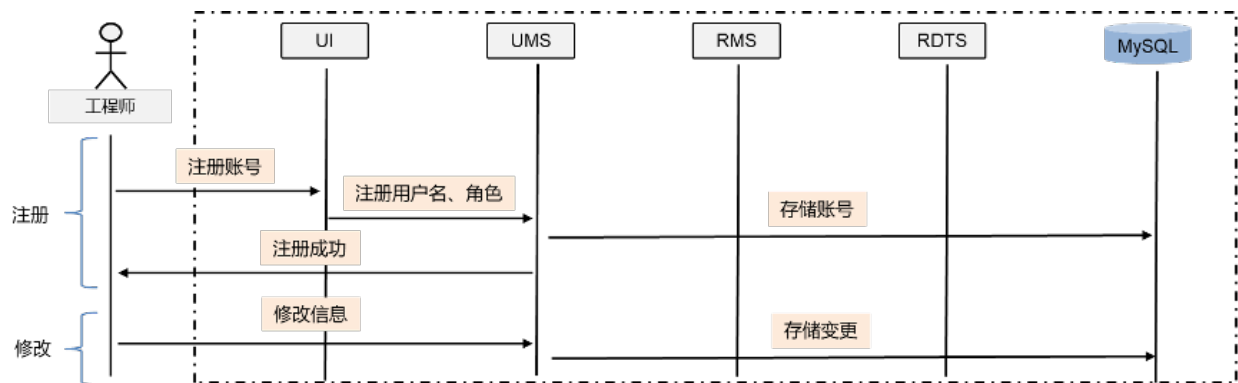
在实际项目中，tag 用于缺陷修复时找到对应代码，用于观测发布后缺陷情况等。

版本发布时，还应该提供 Release 报告，说明本次发布交付的 IR 、 SR 。

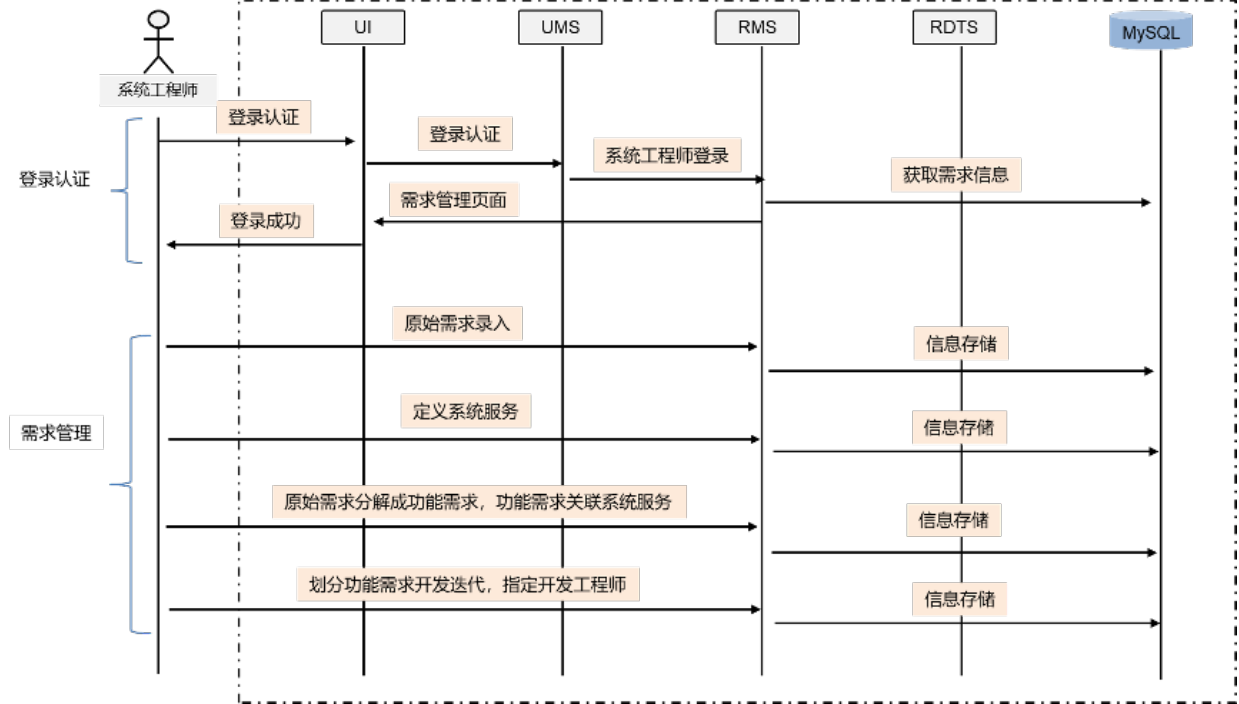
以下部分由客户代表提供，以供实现参考，不作为评分依据

业务流设计参考

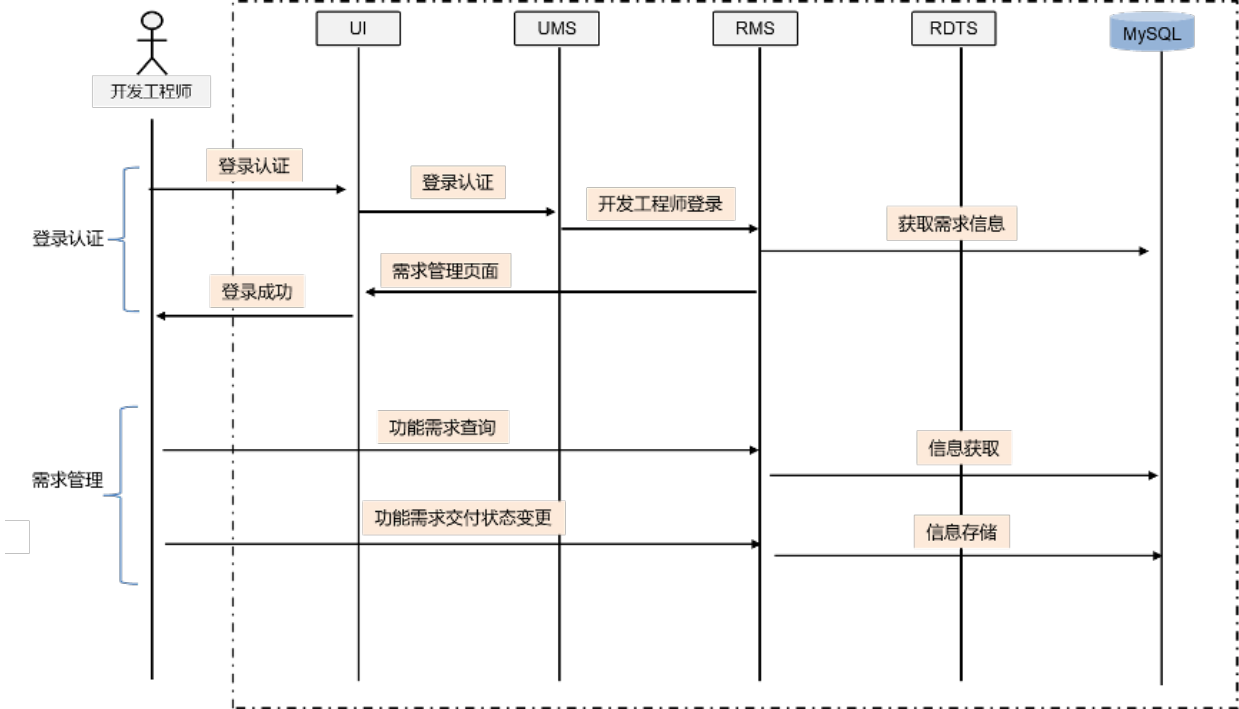
注册账号



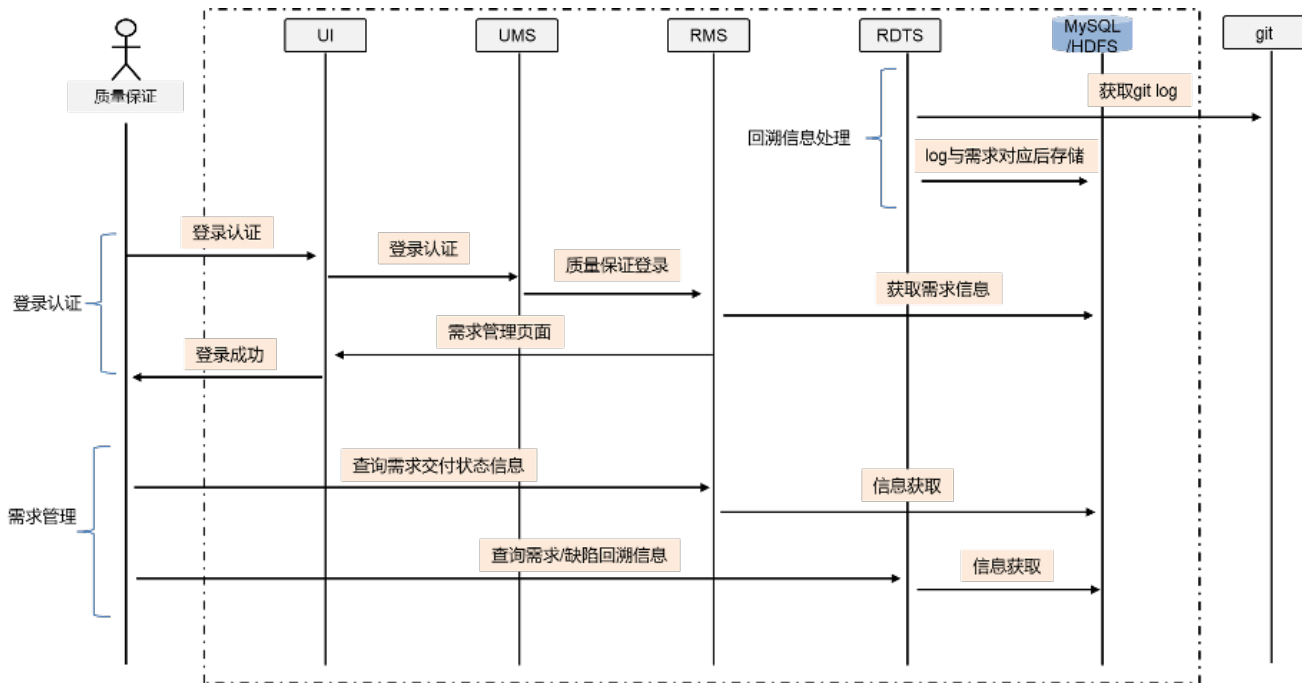
系统工程师



开发工程师



质量保证工程师



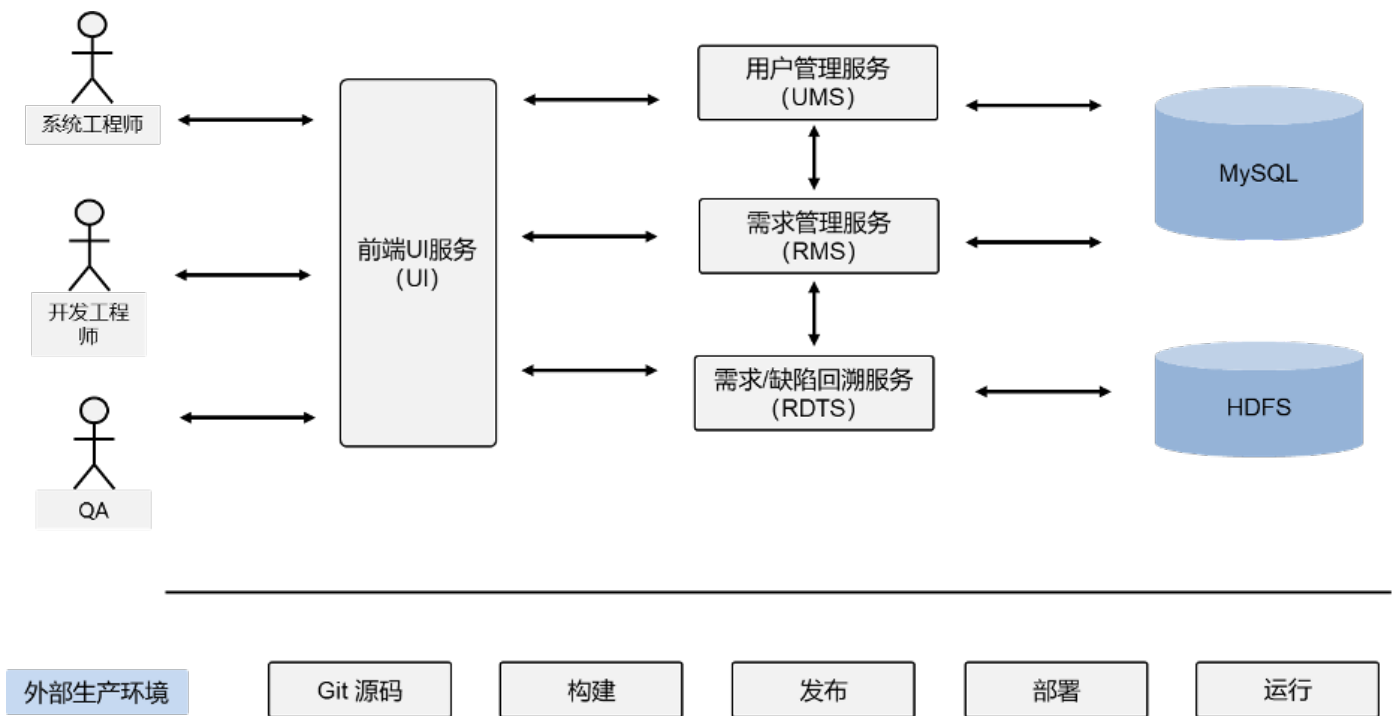
逻辑架构参考

功能分解

建议研发工作都由SR跟踪，包括功能开发、数据库安装调试等。

原始需求	功能需求	需求描述	开发工程师	系统服务	迭代计划	状态
IR.001						
	SR.001.001	系统工程师登录后，首先进入需求管理页面。该页面需求可编辑。	张xx	RMS	迭代1	
	SR.001.002	支持新增原始需求	张xx	RMS	迭代1	
	SR.001.003	支持修改原始需求	张xx	RMS	迭代2	
	SR.001.004	支持删除原始需求	张xx	RMS	迭代2	
	SR.001.005	系统工程师需求管理前端页面	王xx	UI	迭代1	
IR.009						
	SR.009.001	支持用户注册账号	李xx	UMS	迭代1	
	SR.009.002	支持用户修改账号信息	李xx	UMS	迭代2	
	SR.009.003	支持用户登录	李xx	UMS	迭代1	
	SR.009.004	登录页面	刘xx	UI	迭代1	

系统架构



前端UI服务（UI）

提供不同角色的登录、输出/输出的界面

用户管理服务（User Management Service）

提供用户管理服务，不同角色的账号，登录后会跳转到对应服务界面

需求管理服务（Requirement Management Service，RMS）

提供需求管理、跟踪的服务

需求/缺陷回溯服务（Requirement & Defection Trace Service，RDTS）

提供Git中代码变更的回溯服务

要求实现多于 1 个系统服务，将功能分解到多个子项目中开发

参考架构使用MySQL、HDFS，实现时可自由选择对应功能数据库

数据表设计参考

用户信息

数据	类型	说明
账号	字符串	如 admin
密码	字符串	需要有一定安全要求
名称	字符串	用户姓名，如张三
角色	字符串	系统工程师、开发工程师、质量保证
组织（可选）	字符串	不同组之间信息不可见

原始需求（IR）

数据	类型	说明
编号	字符串	如IR.001
描述	字符串	需求描述
状态	字符串	初始（未被分解）、开发中、已交付
交付迭代【fk】	Id	迭代id

功能需求（SR）

数据	类型	说明
编号	字符串	如SR.001.001
描述	字符串	需求描述
状态	字符串	初始、开发中、已交付
原始需求【fk】	Id	归属的原始需求
分发者【fk】	Id	系统工程师用户id
开发者【fk】	Id	开发工程师用户id
系统服务【fk】	Id	关联的系统服务id
交付迭代【fk】	Id	迭代id

迭代

数据	类型	说明
名称	字符串	如 Release-0.0.1
截止时间	Date	如 2022-2-22

系统服务

数据	类型	说明
名称	字符串	如 RDTs
描述	字符串	如 需求/缺陷回溯服务

功能需求变更日志（可选）

数据	类型	说明
功能需求	Id	功能需求id
变更人	Id	用户id
描述	字符串	变更描述
初始状态	字符串	初始、开发中、已交付
更新状态	字符串	初始、开发中、已交付
时间	Date	如 2022-2-22

服务接口设计参考

/login

场景: 用于用户登陆

请求参数

参数名称	类型	参数说明	参数示例
userName	string	用户名	"zhangsan@xxx.com"
password	string	用户密码	"**"

响应参数

参数名称	类型	参数说明	参数示例
errmsg	string	若失败，返回错误信息	"The userName or password is incorrect."
token	string	若成功，返回用户token	"x-abcdefghijklmnopqrstuvwxyz"
expiredTime	date	若成功，返回token过期时间	"2022-2-22 22:22:22"

/verify

场景: 用户登陆获取token后，用于后续操作的验证

请求参数

参数名称	类型	参数说明	参数示例
token	string	用户token	"x-abcdefghijklmnopqrstuvwxyz"

响应参数

参数名称	类型	参数说明	参数示例
status	bool	token是否合法	True/False

界面设计参考

登录页面

用户名：明文

密码：****（隐藏）

系统工程师界面 – 基础信息配置

架构要素配置

xxx1
xxx2
xxx3

创建修改删除

迭代划分配置

xxx1
xxx2
xxx3

创建修改删除

开发人员配置

xxx1
xxx2
xxx3

创建修改删除

其他需要的配置

xxx1
xxx2
xxx3

创建修改删除

【说明】

1. 如果信息比较多，可以按功能在拆分成不同页面
2. 每个列表的明细信息，可以自行设计（应该列表就能支持）
3. 删除的时候，如果需求的分解分配已经存在关联，需要弹出确认窗口，确认是否删除；如果确认删除，相关需求分解悬空，等待重新分解分配

系统工程师界面 – 需求管理和分解分配页面

可选择需求树

选中后

- IR.001.xxxx
 - SR.001.001.xxxx
 - SR.001.002.xxxx
 - SR.001.003.xxxx
 - ...
- IR.002.xxxx
 - SR.002.001.xxxx
 - SR.002.002.xxxx
 - SR.002.003.xxxx
 - ...
- IR.00n.xxxx

IR 需求管理

选中后

1. 选中或创建
2. 通过编辑框修改

IR.001.xxxx	<button>创建</button> <button>删除</button>
-------------	---

需求分解和分配

选中后

1. 选中或创建，通过编辑框修改SR内容分解
2. 通过列表选择基础配置信息，实现分配

通过列表选择基础配置

SR.001.001.xxxx	架构要素	开发责任人	迭代
	UI		
	UMS		
	...		
	<button>创建</button>	<button>删除</button>	

保存配置

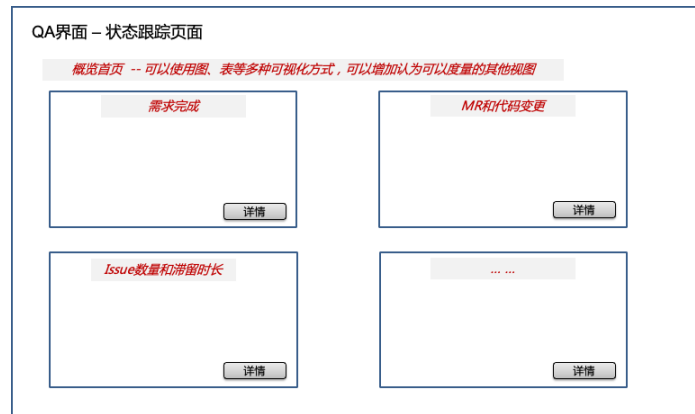
开发工程师界面 – 需求交付状态修改

看到自己负责开发需求列表

需求	当前状态
SR.001.001.xxxx	完成
SR.001.002.xxxx	完成
SR.001.003.xxxx	开发中
...	...
SR.002.001.xxxx	初始化
SR.002.002.xxxx	初始化
SR.002.003.xxxx	初始化
...	初始化

选中后，通过列表选择状态变化

保存修改



外部接口参考

个人 git 仓挂钩子参考：<https://git-scm.com/book/en/v2/Customizing-Git-Git-Hooks>

主 git 仓挂钩子参考：

- github: <https://docs.github.com/en/developers/webhooks-and-events/webhooks/creating-webhooks#setting-up-a-webhook>
- gitlab: https://docs.gitlab.com/ee/administration/server_hooks.html

git api 获取 MR 信息：

- github: <https://docs.github.com/en/rest/reference/pulls#list-pull-requests>
- gitlab: https://docs.gitlab.com/ee/api/merge_requests.html#list-project-merge-requests