

# High-level Design

## Prototype Design

The UI design of the project

### User Section

#### View estimate tasks

This part is for viewing estimate tasks. Here you can see the basic information of all the estimated projects for that user.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin





用户--查看估算任务

admin

这里可以查看你的估算记录

请选择

请输入搜索内容

创建估算 +

#	项目名称	项目简介	创建时间	状态	估算方法	操作	备注
1	admin的项目	admin的项目	2018-01-15	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	
2	校园信息管理系统	同济大学	2018-01-15	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	
3	众包	tongji409	2018-01-16	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	
4	新项目	新项目	2018-01-16	待审核	IFPUG	<a href="#">移除</a>	
5	全新项目	全新项目	2018-01-16	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	

You can search for projects based on status, time, etc.



## 用户--查看估算任务



admin

这里可以查看你的估算记录

状态



待审核



创建估算 +

#	项目名称	项目简介	创建时间	状态	估算方法	操作	备注
1	新项目	新项目	2018-01-16	待审核	IFPUG	<a href="#">移除</a>	
2	众包	项目综合管理平台	2018-01-17	待审核	IFPUG	<a href="#">移除</a>	



## 用户--查看估算任务



admin

这里可以查看你的估算记录

所有



项目|

创建估算 +

#	项目名称	项目简介	创建时间	状态	估算方法	操作	备注
1	admin的项目	admin的项目	2018-01-15	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	
2	新项目	新项目	2018-01-16	待审核	IFPUG	<a href="#">移除</a>	
3	全新项目	全新项目	2018-01-16	完成	IFPUG	<a href="#">移除</a> <a href="#">查看报告</a>	
4	众包	项目综合管理平台	2018-01-17	待审核	IFPUG	<a href="#">移除</a>	

## Create Estimation-Project description

Fill in the project name, Introduction and other basic project information.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

用户创建估算--项目说明

admin

填写估算项目的详细信息

\* 项目名称

测试项目

\* 项目简介

暂无简介

\* 项目负责人

张三

\* 联系方式

18634969595

\* 估算方式

IFPUG

返回

下一步

## Create Estimation-Functional Decomposition

Describe the basic structure of a project.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

用户创建估算--功能分解

admin

展开

隐藏

► 用户管理

► 财务管理

使用说明

1.查看右键菜单

在左侧的 新建文件夹 上右键，查看右键菜单

使用说

新建文件夹

新建文件夹 (副本)

新建文件夹 (2)

新建文件夹 (3)

新建文件夹 (4)

新建文件夹 (5)

新建文件夹 (6)

新建文件夹 (7)

新建文件夹 (8)

新建文件夹 (9)

新建文件夹 (10)

新建文件夹 (11)

新建文件夹 (12)

新建文件夹 (13)

新建文件夹 (14)

新建文件夹 (15)

新建文件夹 (16)

新建文件夹 (17)

新建文件夹 (18)

新建文件夹 (19)

新建文件夹 (20)

新建文件夹 (21)

新建文件夹 (22)

新建文件夹 (23)

新建文件夹 (24)

新建文件夹 (25)

新建文件夹 (26)

新建文件夹 (27)

新建文件夹 (28)

新建文件夹 (29)

新建文件夹 (30)

新建文件夹 (31)

新建文件夹 (32)

新建文件夹 (33)

新建文件夹 (34)

新建文件夹 (35)

新建文件夹 (36)

新建文件夹 (37)

新建文件夹 (38)

新建文件夹 (39)

新建文件夹 (40)

新建文件夹 (41)

新建文件夹 (42)

新建文件夹 (43)

新建文件夹 (44)

新建文件夹 (45)

新建文件夹 (46)

新建文件夹 (47)

新建文件夹 (48)

新建文件夹 (49)

新建文件夹 (50)

新建文件夹 (51)

新建文件夹 (52)

新建文件夹 (53)

新建文件夹 (54)

新建文件夹 (55)

新建文件夹 (56)

新建文件夹 (57)

新建文件夹 (58)

新建文件夹 (59)

新建文件夹 (60)

新建文件夹 (61)

新建文件夹 (62)

新建文件夹 (63)

新建文件夹 (64)

新建文件夹 (65)

新建文件夹 (66)

新建文件夹 (67)

新建文件夹 (68)

新建文件夹 (69)

新建文件夹 (70)

新建文件夹 (71)

新建文件夹 (72)

新建文件夹 (73)

新建文件夹 (74)

新建文件夹 (75)

新建文件夹 (76)

新建文件夹 (77)

新建文件夹 (78)

新建文件夹 (79)

新建文件夹 (80)

新建文件夹 (81)

新建文件夹 (82)

新建文件夹 (83)

新建文件夹 (84)

新建文件夹 (85)

新建文件夹 (86)

新建文件夹 (87)

新建文件夹 (88)

新建文件夹 (89)

新建文件夹 (90)

新建文件夹 (91)

新建文件夹 (92)

新建文件夹 (93)

新建文件夹 (94)

新建文件夹 (95)

新建文件夹 (96)

新建文件夹 (97)

新建文件夹 (98)

新建文件夹 (99)

新建文件夹 (100)

2.构建架构模块

Fill the ILF and EIF tables. Then write the steps of "login".

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

admin

用户创建估算--功能分解

展开

隐藏

+

▼ 用户管理

登陆

注册

查看个人信息

▼ 财务管理

Step0:

输入登录信息

—

>

Step1:

验证登录信息

—

>

Step2:

返回结果

—

>

内部逻辑文件

外部接口文件

文件名	DET	
用户信息	用户ID 用户名 密码	删除

+

Detail each step, choose the ILFs and EIFs, then choose the data fields for each file.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

▼ 用户管理

登陆

注册

查看个人信息

▼ 财务管理

Step0:

输入登录信息

—

▼

+

内部逻辑文件

用户信息

▼

数据字段

+

用户名

—

密码

—

—

+

外部接口文件

验证码

▼

数据字段

+

验证图片

—

### Create Estimation-Adjustment Factor

Select adjustment factors for the project.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

用户创建估算--调整因子

admin

开发类型

☒ New Development

☐ Enhancement

☐ Re-development

开发平台

☒ Personal computer

☐ Mainframe

☐ Mid-range

开发语言

☐ 3GL

☒ 4GL

☐ ApG

是否使用数据库

☒ Yes

☐ No

软件可靠性要求

Very High

产品复杂度

Very Low

执行时间约束

Extra High

要求的开发进度

请选择

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,admin

开发类型

☒ New Development

☐ Enhancement

☐ Re-development

开发平台

☒ Personal computer

☐ Mainframe

☐ Mid-range

开发语言

☐ 3GL

☒ 4GL

☐ ApG

是否使用数据库

☒ Yes

☐ No

软件可靠性要求

Very High

产品复杂度

Very Low

执行时间约束

Extra High

要求的开发进度

Low

生产率

80

小时 / FP

人力成本

3000

元 / 人月

返回

提交

# Administrator Section

## View estimate tasks

This part is for viewing estimate tasks. Here you can see the basic information of all the estimated projects for that user.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,sixaps

管理员--查看估算任务

sixaps

请选择

请输入搜索内容

#	用户名	项目名称	项目简介	创建时间	状态	估算方法	操作	备注
1	admin	admin的项目	admin的项目	2018-01-15	完成	IFPUG	<a href="#">查看报告</a>	
2	admin	校园信息管理系统	同济大学	2018-01-15	完成	IFPUG	<a href="#">查看报告</a>	
3	admin	众包	tongji409	2018-01-16	完成	IFPUG	<a href="#">查看报告</a>	
4	admin	新项目	新项目	2018-01-16	待审核	IFPUG	<a href="#">估算</a>	
5	admin	全新项目	全新项目	2018-01-16	完成	IFPUG	<a href="#">查看报告</a>	

校园管理系统

# Functional Decomposition

Describe the basic structure of a project.

管理员估算--功能分解

sixaps

展开

收起

▶ 用户管理

▶ 财务管理

使用说明

1.查看右键菜单

在左侧的 新建文件夹 上右键，查看右键菜单

展开

隐藏

▶新建文件夹

+ 新建子功能

+ 新建子模块

✕ 删除子功能 / 子模块

✕ 删除功能 / 模块

+ 新建功能功能

+ 新建功能模块

— 剪切

✕ 粘贴

✎ 重命名

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,sixaps

展开

收起

▼ 用户管理

登陆

注册

查看个人信息

▼ 财务管理

管理员估算--功能分解

sixaps

Step0: 输入登录信息

Step1: 验证登录信息

Step2: 返回结果

事务功能

数据功能

事务名称	事务类型	逻辑文件	FTR数量	DET	DET数量	复杂度	UFP	
输入登录信息	EI	用户信息	1	用户名 密码	2	低	3	删除
返回结果	EO	用户信息	1	用户名	1	低	4	删除

+

## View Adjustment Factor

View adjustment factors for the project.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,sixaps

管理员--查看调整因子

sixaps

编辑

开发类型	New Development	软件可靠性要求	Very High
开发平台	Personal computer	产品复杂度	Very Low
开发语言	4GL	执行时间约束	Extra High
是否使用数据库	Yes	要求的开发进度	Very High
生产率	80 小时 / FP	人力成本	3000 元 / 人月

返回

提交

## Edit Adjustment Factor

Edit adjustment factors for the project.

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,sixaps

管理员--编辑调整因子

sixaps

编辑

开发类型

☒ New Development

☐ Enhancement

☐ Re-development

开发平台

☒ Personal computer

☐ Mainframe

☐ Mid-range

开发语言

☐ 3GL

☒ 4GL

☐ ApG

是否使用数据库

☒ Yes

☐ No

软件可靠性要求

Very High

产品复杂度

Very Low

执行时间约束

Extra High

要求的开发进度

基于人工智能的软件外包服务平台

项目

估算

帮助

你好,sixaps

管理员--查看调整因子

编辑

提示

修改状态

待修改

备注

登陆部分分解不合理

取消

确定

开发语言

4GL

执行时间约束

Extra High

是否使用数据库

Yes

要求的开发进度

Very High

生产率

80 小时 / FP

人力成本

3000 元 / 人月

返回

提交

View The Report



# IFPUG报告

数据功能表格

逻辑文件名称	逻辑文件类型	RET	RET数量	DET	DET数量	复杂度	DFP
用户信息	ILF	用户信息	1	学号 密码	2	低	5

事务功能表格

事务名称	事务类型	逻辑文件	逻辑文件数量	DET	DET数量	复杂度	TFP
输入用户名	EI	用户信息	1	用户名 密码	2	低	3

调整因子表格

..				..			
----	--	--	--	----	--	--	--

调整因子表格

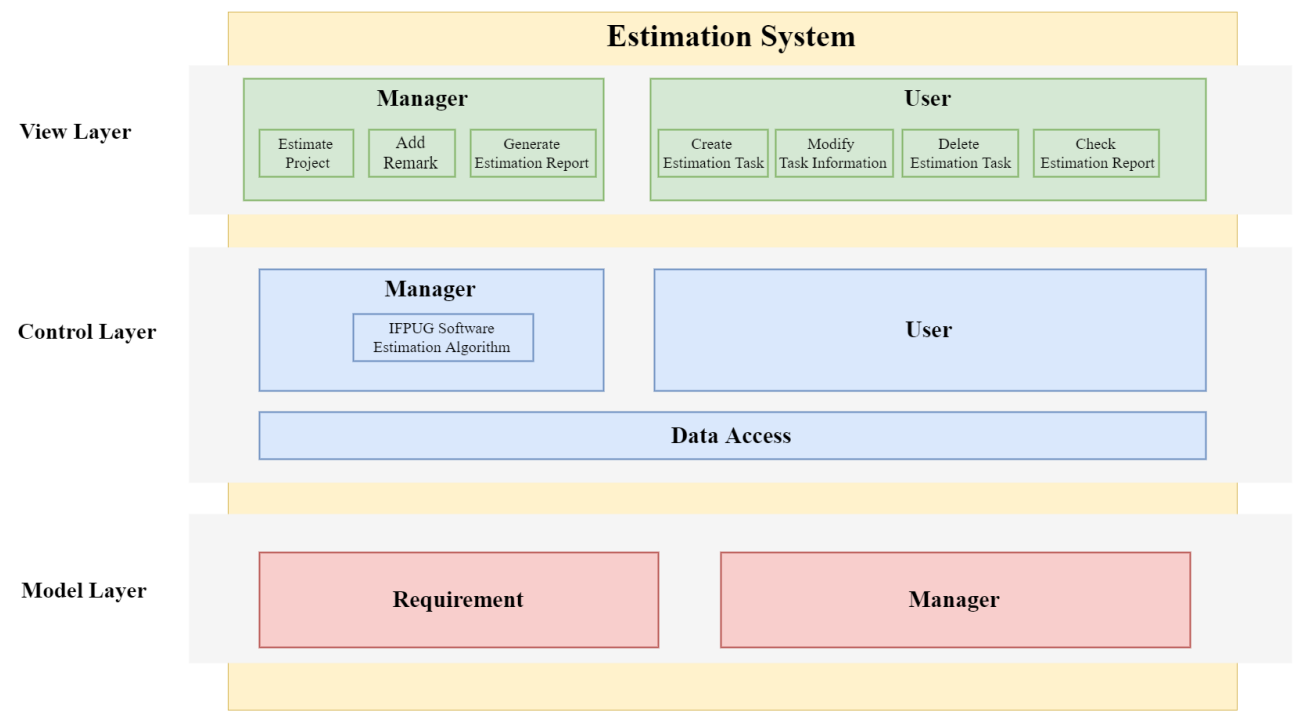
估算人员调整后的估算因子，用蓝色标注

开发类型	开发平台	开发语言	是否使用数据库	软件可靠性
Enhancement	Mid-range	4GL	No	Low

估算结果表格

UFP	AFP	项目工期	项目成本
8	11.25	123.78天	123782.82元

The business architecture outline the key components in each layers of the project systems. Here is our Business Architecture:



In the view layer, we have two roles in the system, which can see different pages and have different functions. Because the whole estimation system is a subsystem of the Crowdsourcing System, and the login/logout function is realized by them. What we need to do is just the process of estimation.

The user subsystem is mainly responsible for estimating task creation and modification. The users can first create the estimation task and then finish the details of it. And he can modify the specific information of the estimated task if the error is filled in before. If he thinks that the project is totally wrong, he can also delete his project. Once the task completes estimation, the user could check the estimation Report of it.

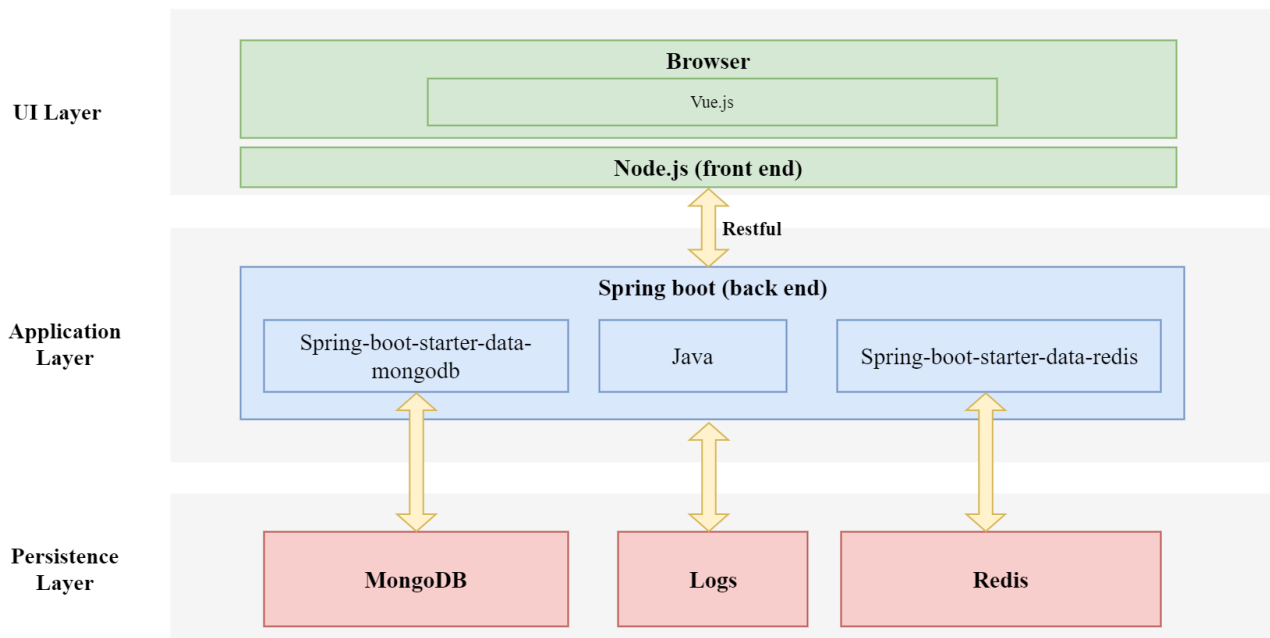
And the manager subsystem is primarily responsible for estimating and generating estimate reports. He can also add remark of the estimation task to inform the user to modify task information.

In the control layer, we need to have different business logic for the user and the administrator, and they will both interact with the database through the same data access class. And we can see, there is an estimation method named IFPUG in the manager subsystem to help the managers with the software estimate.

In the Model Layer, we need to store the data of all the requirements and the managers. The information of the users is stored by the Crowdsourcing System.

## Technology Architecture

The architecture of the technology stacks (coding language, frameworks, libraries, styles, etc) used in the project. Here is our Technology Architecture:



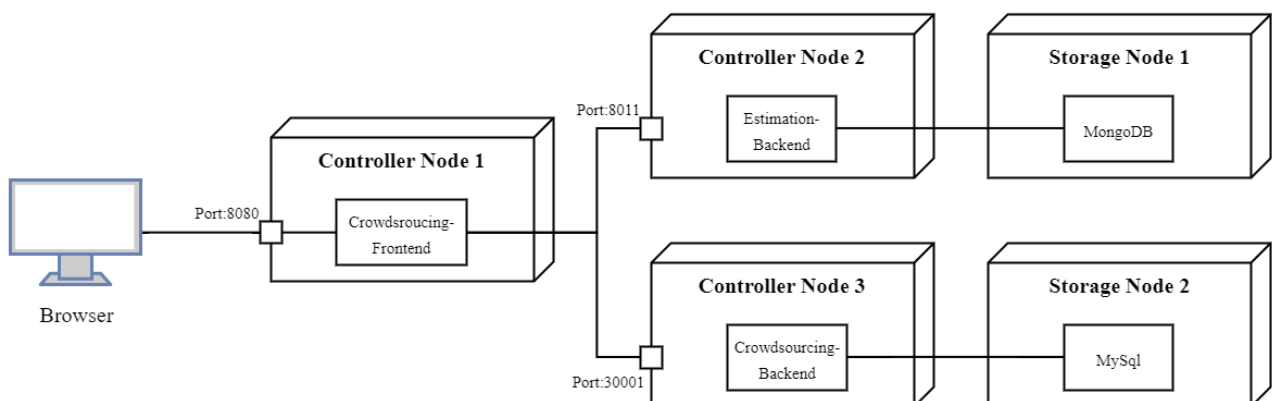
In the UI Layer, we use Vuejs as our front end frame and build the front end on the Nodejs.

In the Application Layer, we use Java as the primary development language for our backend. At the same time, we use the Spring-boot-starter-data-mongodb to Exchange data with the MongoDB database, and through the Spring-boot-starter-data-redis to obtain the crowdsourcing system cache cookies on the Redis, which records the user's login status.

In the Persistence Layer, we use MongoDB for basic data storage, while using Redis to save cookies that record user login status. Spring-boot automatically caches logs and checks for errors or business records.

## Deployment Topology

Deployment Topology shows how the whole project is deployed in the physical environment, including servers, VMs/containers, networks, key components, etc. An overview picture is required to show the topology. Here is our Deployment Topology:



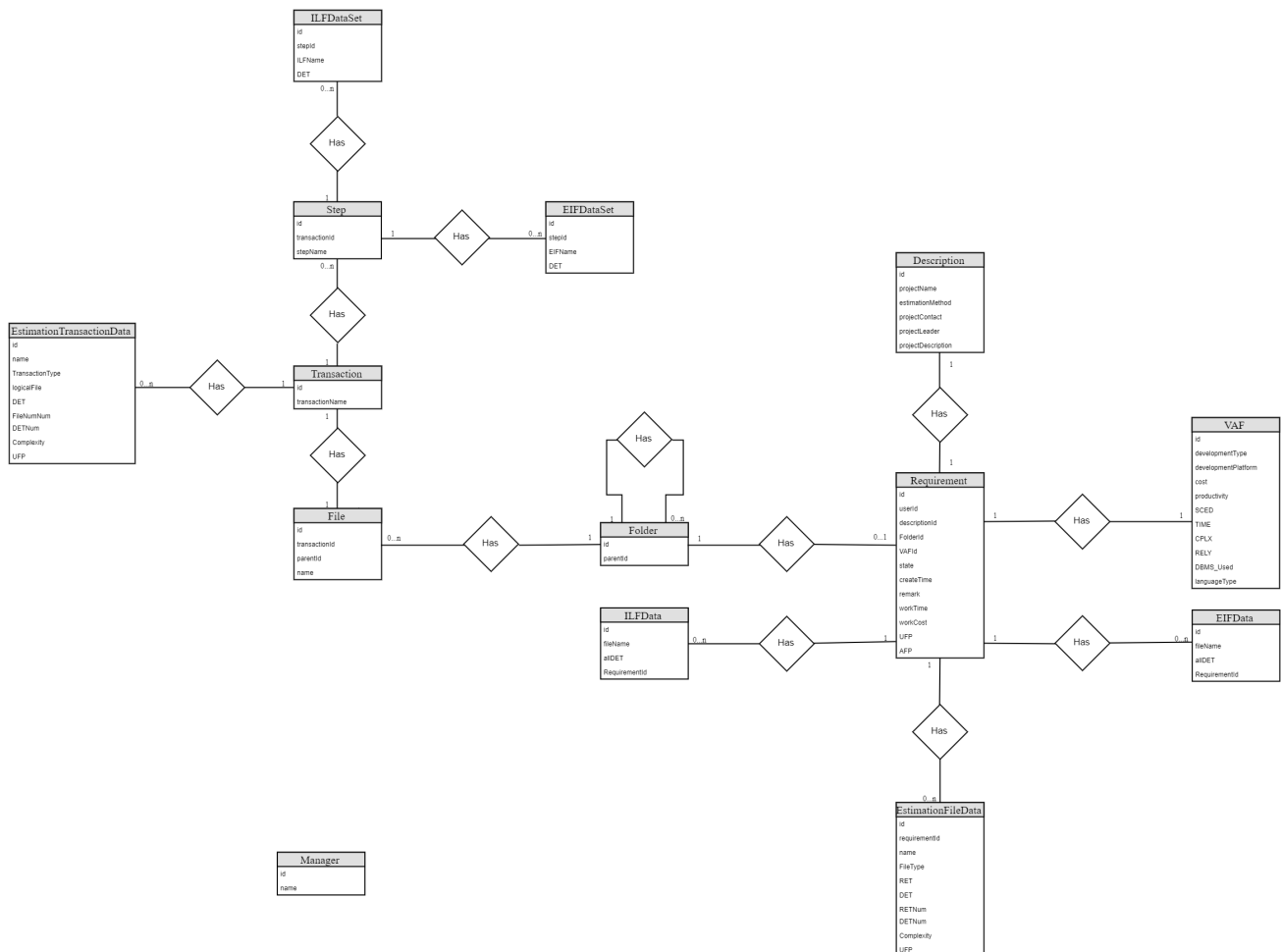
We use the method of front and rear separation to deploy, at the same time, the back end of our estimation system and the back end of the crowdsourcing system are respectively deployed on two computers, respectively interacting with different databases, but together for the same front-end service. This depends on spring's support for cross-domain access, and the low coupling between the estimate system and the crowdsourcing system.

# Data Models

## Entity-Relation Design

An Entity-Relation Diagram is required to show the concepts in the project and their relations.

Here is our E-R diagram:



## Relationship

- *Requirement-Folder*

For each estimate task, it needs to have a function decomposition Tree, so each estimate task must correspond to a top-level folder. But not at the top of the folder does not correspond to an estimate task.

- *Requirement-Description*

Each estimate task has a description, and each description must correspond to an estimate task.

- *Requirement-VAF*

Each estimate task has a set of adjustment factors, and each group of adjustment factors necessarily corresponds to an estimate task.

- *Requirement-ILFData*

Each estimate task has a list of all ILF and corresponding DET information, and each item in the list corresponds to only one estimate task.

- *Requirement-EIFData*

Each estimate task has a list of all EIF and corresponding DET information, and each item in the list corresponds to only one estimate task.

- *Requirement-EstimationFileData*

Each estimate task has a list that contains all the data features, and each item in the list corresponds to only one estimate task.

- *Folder-Folder*

A directory tree is made up of files and folders, and a folder can contain several folders, but there is only one parent node for each folder.

- *Folder-File*

A directory tree is made up of files and folders, and a folder can contain several files, but there is only one parent node for each file.

- *File-Transaction*

Each file node of the tree corresponds to a function of the project, and each function is relative to a file node in the directory tree.

- *Transaction-EstimationTransactionData*

Each feature in the project has a list of its transactional function points, and each item in the list corresponds to only one function.

- *Transaction-Step*

Each feature in the project contains several steps, each of which corresponds to only one function.

- *Step-EIFDataSet*

Each step in a feature corresponds to a ILF and a list of DETs used, and each item in the list will only correspond to one step.

- *Step-ILFDataSet*

Each step in a feature corresponds to a ILF and a list of DETs used, and each item in the list will only correspond to one step.

# Database Design

According to the Entity-Relation Design, the detailed database design for entities and relations should be listed as tables. Here is our tables:

Table: Requirement

Column	Type	Description
id	varchar[100]	key, identifier
userId	varchar[100]	id of the user
descriptionId	varchar[100]	id of the description
FolderId	varchar[100]	id of the Folder
VAFId	varchar[100]	id of the VAF
state	varchar[100]	the state of Requirement
createTime	varchar[100]	the createTime of Requirement
remark	varchar[100]	the remark of Requirement
workTime	integer	the workTime of Requirement
workCost	integer	the workCost of Requirement
UFP	integer	the UFP of Requirement
AFP	integer	the AFP of Requirement

Table: Description

Column	Type	Description
id	varchar[100]	key, identifier
projectName	varchar[100]	name of the project
estimationMethod	varchar[100]	the estimation method of the project
projectContact	varchar[100]	the contact of the project
projectLeader	varchar[100]	the leader of the project
projectDescription	varchar[100]	the description of the project

Table: VAF

Column	Type	Description
id	varchar[100]	key, identifier
developmentType	varchar[100]	the development type of the project
developmentPlatform	varchar[100]	the development platform of the project
cost	varchar[100]	monthly salary per person
productivity	varchar[100]	average efficiency
SCED	varchar[100]	Development Progress of project requirements
TIME	varchar[100]	Project Execution time constraints
CPLX	varchar[100]	the complexity of the project
RELY	varchar[100]	the reliability of the project
DBMS_Used	varchar[100]	whether the project needs to use database or not
languageType	varchar[100]	which language will be used by the project

**Table: EIFData**

Column	Type	Description
id	varchar[100]	key, identifier
fileName	varchar[100]	the name of the EIF
allDET	varchar[100]	all DETs of the EIF
requirementId	varchar[100]	id of the corresponding requirement

**Table: ILFData**

Column	Type	Description
id	varchar[100]	key, identifier
fileName	varchar[100]	the name of the ILF
allDET	varchar[100]	all DETs of the ILF
requirementId	varchar[100]	id of the corresponding requirement

**Table: EstimationFileData**

Column	Type	Description
id	varchar[100]	key, identifier
requirementId	varchar[100]	id of the corresponding requirement
name	varchar[100]	the name of the file
FileType	varchar[100]	the type of the file
RET	varchar[100]	all RETs of the file
DET	varchar[100]	all DETs of the file
RETNum	varchar[100]	the number of RET in the file
DETRNum	varchar[100]	the number of DET in the file
complexity	integer	the complexity of the file
UFP	integer	the UFP of the file

**Table: EstimationTransactionData**

Column	Type	Description
id	varchar[100]	key, identifier
requirementId	varchar[100]	id of the corresponding requirement
name	varchar[100]	the name of the transaction
TransactionType	varchar[100]	the type of the transaction
File	varchar[100]	all files of the transaction
DET	varchar[100]	all DETs of the transaction
FileNum	varchar[100]	the number of file in the transaction
DETRNum	varchar[100]	the number of DET in the transaction
complexity	integer	the complexity of the transaction
UFP	integer	the UFP of the transaction

**Table: Folder**

Column	Type	Description
id	varchar[100]	key, identifier
parentId	varchar[100]	parent id of the folder



**Table: Folder**

Column	Type	Description
id	varchar[100]	key, identifier
parentId	varchar[100]	parent id of the file
transactionId	varchar[100]	the id of the corresponding transaction
name	varchar[100]	the name of the file

**Table: Transaction**

Column	Type	Description
id	varchar[100]	key, identifier
transactionName	varchar[100]	the name of the transaction

**Table: Step**

Column	Type	Description
id	varchar[100]	key, identifier
transactionId	varchar[100]	id of the corresponding transaction
stepName	varchar[100]	the name of the step

**Table: ILFDataSer**

Column	Type	Description
id	varchar[100]	key, identifier
stepId	varchar[100]	the id of the corresponding step
ILFName	varchar[100]	the name of the ILF
DET	varchar[100]	all DETs of the ILF

**Table: EIFDataSer**

Column	Type	Description
id	varchar[100]	key, identifier
stepId	varchar[100]	the id of the corresponding step
EIFName	varchar[100]	the name of the EIF
DET	varchar[100]	all DETs of the EIF

## Interface Specifications

The specifications of interfaces used in interactions among subsystems.

### *estimation/addRequirement*

#### Description

Add a new requirement(estimation project).

Request Method	Get
Authorization	Required

#### Parameters

Name	Located in	Description	Required	Schema
null	null	null	null	null

#### Headers

Key	value
Authorization	String

#### Body

Key	value
void	void

## Responses

Code	Description	Schema
200	Successful response	<i>id</i> : string

## Request Sample

```
estimation/addRequirement
```

## Response Sample

```
{
  "_id" : "1515998053438"
}
```

## *estimation/getRequirement*

### Description

Get a requirement(estimation project) by id.

Request Method	Get
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement(estimation project) id	Yes	String

### Headers

Key	value
Authorization	String

## Body

Key	value
void	void

## Responses

Code	Description	Schema
200	Successful response	<i>requirement</i> : Object

## Request Sample

```
estimation/getRequirement?id=1515998053438
```

## Response Sample

```
{
  requirement:{
    "_id" : "1515998053438",
    "_class" : "estimation.bean.Requirement",
    "userId" : "admin",
    "description" : {
      "projectName" : "admin的项目",
      "projectDescription" : "admin的项目",
      "projectLeader" : "admin",
      "projectContact" : "192837192837",
      "estimationMethod" : "IFPUG"
    },
    "transactions" : [
      {
        "_id" : "1515998086763",
        "transactionName" : "lskdjf",
        "steps" : [
          {
            "stepName" : "sad ",
            "eifDataSets" : [
              {
                "ExternalInterfaceFileName" : "qwe ",
                "DET" : [
                  "qw"
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```

    ],
    "ilfDataSets" : [
        {
            "InnerlogicalFileName" : "sd",
            "DET" : [
                "d"
            ]
        }
    ]
},
],
"estimationTransactionDatas" : [
    {
        "name" : "asd ",
        "TransactionType" : "EI",
        "LogicalFile" : " as a as as a ",
        "DET" : "asd ad asdasd ",
        "FileNum" : 5,
        "DETNum" : 3,
        "Complexity" : "低",
        "UFP" : 3
    }
]
}
],
"treeOfTransactions" : {
    "_id" : "1515998074311",
    "name" : "tree",
    "childFolders" : [
        {
            "_id" : "00001",
            "name" : "功能模块1",
            "childFolders" : [],
            "childFiles" : [
                {
                    "_id" : "1515998086763",
                    "name" : "lskdjff"
                }
            ]
        }
    ]
},
"childFiles" : []
},
"state" : "完成",
"createTime" : "2018-01-15T14:34:13.516",
"remark" : "",
"UFP" : 0,
"AFP" : 0.0,
"workTime" : 0.0,
"workCost" : 0.0,
"allEIFData" : [
    {
        "fileName" : "qwe ",
    }
]

```

```

        "allDET" : "qw qwe"
    }
],
"allILFData" : [
    {
        "fileName" : "sd",
        "allDET" : "sd d"
    }
],
"estimationFileDatas" : [
    {
        "name" : "dfs dfs",
        "FileType" : "ILF",
        "RET" : "dfds",
        "DET" : "df df dfs",
        "RETRNum" : 1,
        "DETNum" : 3,
        "Complexity" : "低",
        "UFP" : 5
    }
],
"originalVAF" : {
    "developmentType" : "New Development",
    "developmentPlatform" : "Mainframe",
    "languageType" : "4GL",
    "DBMS_Used" : "No",
    "RELY" : "Low",
    "CPLX" : "Very High",
    "TIME" : "Very High",
    "SCED" : "Low",
    "productivity" : "12",
    "cost" : "123"
},
"newVAF" : {
    "developmentType" : "New Development",
    "developmentPlatform" : "Mainframe",
    "languageType" : "4GL",
    "DBMS_Used" : "No",
    "RELY" : "Low",
    "CPLX" : "Very High",
    "TIME" : "Very High",
    "SCED" : "Low",
    "productivity" : "12",
    "cost" : "123"
}
}

```

## ***estimation/getAllRequirementsByUser***

### **Description**

Get basic information about all the requirements of a user by user's id.

Request Method	Get
Authorization	Required

Parameters

Name	Located in	Description	Required	Schema
null	null	null	null	null

Headers

Key	value
Authorization	String

Body

Key	value
void	void

Responses

Code	Description	Schema
200	Successful response	<i>requiemnts</i> : List

Request Sample

estimation/getAllRequirementsByUser
-------------------------------------

Response Sample

--

```

{
  requirements: [{
    "_id" : "1515998053438",
    "_class" : "estimation.bean.Requirement",
    "userId" : "admin",
    "description" : {
      "projectName" : "admin的项目",
      "projectDescription" : "admin的项目",
      "projectLeader" : "admin",
      "projectContact" : "192837192837",
      "estimationMethod" : "IFPUG"
    }
  }],{
    "_id" : "1516007668493",
    "_class" : "estimation.bean.Requirement",
    "userId" : "admin",
    "description" : {
      "projectName" : "校园信息管理系统",
      "projectDescription" : "同济大学",
      "projectLeader" : "刘家琪",
      "projectContact" : "156666666666",
      "estimationMethod" : "IFPUG"
    }
  }]
}

```

## ***estimation/getAllRequirements***

### **Description**

Get basic information about all the requirements of all users.

Note: Only administrators can call this interface.

Request Method	Get
Authorization	Required

### **Parameters**

Name	Located in	Description	Required	Schema
null	null	null	null	null

### **Headers**



Key	value
Authorization	String

## Body

Key	value
void	void

## Responses

Code	Description	Schema
200	Successful response	<i>requiements</i> : List

## Request Sample

```
estimation/getAllRequirements
```

## Response Sample

```
{
  requirements: [{
    "_id" : "1515998053438",
    "_class" : "estimation.bean.Requirement",
    "userId" : "admin",
    "description" : {
      "projectName" : "admin的项目",
      "projectDescription" : "admin的项目",
      "projectLeader" : "admin",
      "projectContact" : "192837192837",
      "estimationMethod" : "IFPUG"
    }
  },{
    "_id" : "1516007668493",
    "_class" : "estimation.bean.Requirement",
    "userId" : "admin",
    "description" : {
      "projectName" : "校园信息管理系统",
      "projectDescription" : "同济大学",
      "projectLeader" : "刘家琪",
      "projectContact" : "15666666666",
      "estimationMethod" : "IFPUG"
    }
  }
}]
```

```
}
```

## ***estimation/deleteRequirement***

### **Description**

Delete a requirement by id.

Note: Only owner of the requirement could delete his requirement.

Request Method	Post
Authorization	Required

### **Parameters**

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### **Headers**

Key	value
Authorization	String

### **Body**

Key	value
void	void

### **Responses**

Code	Description	Schema
200	Successful response	void

### **Request Sample**

```
estimation/deleteRequirement?id=1516007668493
```

## Response Sample

```
{}
```

## *estimation/changeState*

### Description

change the state and remark of a requirement.

Note: Only administrators can call this interface.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

### Body

Key	value
void	void

### Responses

Code	Description	Schema
200	Successful response	void

### Request Sample

estimation/changeState?id=1516007668493

## Response Sample

```
{}
```

## *estimation/addDescription*

### Description

Add or change the description of a requirement.

Note: Only owner of the requirement could add the description of his requirement.

Request Method	Get
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

### Body

Key	value
projectName	String
projectDescription	String
projectLeader	String
projectContact	String
estimationMethod	String

## Responses

Code	Description	Schema
200	Successful response	void

## Request Sample

```
estimation/addDescription?id=1516007668493
```

## Response Sample

```
{}
```

# *estimation/identity*

## Description

Check if the logged in user is an administrator.

Request Method	Get
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
null	null	null	null	null

## Headers

Key	value
Authorization	String

## Body

Key	value
null	null

## Responses

Code	Description	Schema
200	Successful response	<i>code</i> : int

## Request Sample

estimation/identity

## Response Sample

```
{
  "code" : 0    //0 represents the administrator, 1 represents the normal user, 200 is not
logged in, and 500 represents database error
}
```

# *estimation/add*

## Description

Add the manager authority of a user.

Note: Only administrators can call this interface.

Request Method	Post
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
null	null	null	null	null

## Headers

Key	value
Authorization	String

## Body

Key	value
username	String

### Responses

Code	Description	Schema
200	Successful response	<i>code</i> : Int

### Request Sample

estimation/add
----------------

### Response Sample

<pre>{   "code" : 0    //0 represents success, 200 represents failure, and 500 represents database error }</pre>
--

## ***estimation/delete***

### Description

Remove administrator authority for a user.

Note: Only administrators can call this interface.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
null	null	null	null	null

### Headers

Key	value
Authorization	String

## Body

Key	value
username	String

## Responses

Code	Description	Schema
200	Successful response	<i>code</i> : Int

## Request Sample

```
estimation/delete
```

## Response Sample

```
{
  "code" : 0      //0 represents successful, 200 represents failure, and 500 represents database
  error
}
```

## *estimation/delete*

### Description

get the usernames of all administrators.

Note: Only administrators can call this interface.

Request Method	Post
Authorization	Required

## Parameters



Name	Located in	Description	Required	Schema
null	null	null	null	null

## Headers

Key	value
Authorization	String

## Body

Key	value
username	String

## Responses

Code	Description	Schema
200	Successful response	<i>list</i> : List

## Request Sample

```
estimation/getAll
```

## Response Sample

```
{
  "list" : ["sixaps","admin"]
}
```

## ***estimation/updateResult***

### Description

Update estimated transaction information and data information, and calculate their complexity and UFP .

Note: Only administrators can call this interface.

Request Method	Get
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

## Headers

Key	value
Authorization	String

## Body

Key	value
eTDs	JSONArray
eFDs	JSONArray
tId	String

## Responses

Code	Description	Schema
200	Successful response	<i>result</i> : Map

## uest Sample

```

url:"http://ip:8011/estimation/updateResult?id=1515998053438"
body:{
  "eTDs" : [ {
    "name" : "asd ",
    "transactionType" : "EI",
    "logicalFile" : " as a as as a ",
    "DET" : "asd ad asdasd ",
    "FileNum" : 5,
    "detNum" : 3
  }
}

```

```

],
  "eFDs" : [
    {
      "name" : "dfs dfs",
      "FileType" : "ILF",
      "RET" : "dfds",
      "DET" : "df df dfs",
      "RETRNum" : 1,
      "DETNum" : 3
    }
  ],
  "tId" : "1515998086763"
}

```

## Response Sample

```

{
  "estimationTransactionDatas" : [
    {
      "name" : "asd ",
      "TransactionType" : "EI",
      "LogicalFile" : " as a as as a ",
      "DET" : "asd ad asdasd ",
      "FileNum" : 5,
      "DETNum" : 3,
      "Complexity" : "低",
      "UFP" : 3
    }
  ],
  "estimationFileDatas" : [
    {
      "name" : "dfs dfs",
      "FileType" : "ILF",
      "RET" : "dfds",
      "DET" : "df df dfs",
      "RETRNum" : 1,
      "DETNum" : 3,
      "Complexity" : "低",
      "UFP" : 5
    }
  ]
}

```

## *estimation/getReport*

### Description

Get the estimation report of a requirement.

Request Method	Get
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
id	query	requirement	yes	String

## Headers

Key	value
Authorization	String

## Body

Key	value
void	void

## Responses

Code	Description	Schema
200	Successful response	<i>requiement</i> : Object

## Request Sample

```
estimation/getReport?id=1515998053438
```

## Response Sample

```
{
  "_id" : "1515998053438",
  "_class" : "estimation.bean.Requirement",
  "userId" : "admin",
  "description" : {
    "projectName" : "admin的项目",
    "projectDescription" : "admin的项目",
    "projectLeader" : "admin",
    "projectContact" : "192837192837",
```

```

    "estimationMethod" : "IFPUG"
  },
  "transactions" : [
    {
      "_id" : "1515998086763",
      "transactionName" : "lskdjf",
      "steps" : [
        {
          "stepName" : "sad ",
          "eifDataSets" : [
            {
              "ExternalInterfaceFileName" : "qwe ",
              "DET" : [
                "qw"
              ]
            }
          ],
          "ilfDataSets" : [
            {
              "InnerlogicalFileName" : "sd",
              "DET" : [
                "d"
              ]
            }
          ]
        }
      ],
      "estimationTransactionDatas" : [
        {
          "name" : "asd ",
          "TransactionType" : "EI",
          "LogicalFile" : " as a as as a ",
          "DET" : "asd ad asdasd ",
          "FileNum" : 5,
          "DETNum" : 3,
          "Complexity" : "低",
          "UFP" : 3
        }
      ]
    }
  ],
  "treeOfTransactions" : {
    "_id" : "1515998074311",
    "name" : "tree",
    "childFolders" : [
      {
        "_id" : "00001",
        "name" : "功能模块1",
        "childFolders" : [],
        "childFiles" : [
          {
            "_id" : "1515998086763",
            "name" : "lskdjf"
          }
        ]
      }
    ]
  }
}

```

```

    }
  ]
}
],
"childFiles" : [],
},
"state" : "完成",
"createTime" : "2018-01-15T14:34:13.516",
"remark" : "",
"UFP" : 0,
"AFP" : 0.0,
"workTime" : 0.0,
"workCost" : 0.0,
"allEIFData" : [
  {
    "fileName" : "qwe ",
    "allDET" : "qw qwe"
  }
],
"allILFData" : [
  {
    "fileName" : "sd",
    "allDET" : "sd d"
  }
],
"estimationFileDatas" : [
  {
    "name" : "dfs dfs",
    "FileType" : "ILF",
    "RET" : "dfds",
    "DET" : "df df dfs",
    "RETRNum" : 1,
    "DETNum" : 3,
    "Complexity" : "低",
    "UFP" : 5
  }
],
"originalVAF" : {
  "developmentType" : "New Development",
  "developmentPlatform" : "Mainframe",
  "languageType" : "4GL",
  "DBMS_Used" : "No",
  "RELY" : "Low",
  "CPLX" : "Very High",
  "TIME" : "Very High",
  "SCED" : "Low",
  "productivity" : "12",
  "cost" : "123"
},
"newVAF" : {
  "developmentType" : "New Development",
  "developmentPlatform" : "Mainframe",

  "languageType" : "4GL",

```

```
    "DBMS_Used" : "No",
    "RELY" : "Low",
    "CPLX" : "Very High",
    "TIME" : "Very High",
    "SCED" : "Low",
    "productivity" : "12",
    "cost" : "123"
  }
}
```

## *estimation/addVAF*

### Description

Add the adjustment factors for an estimate project.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

### Body

Key	value
developmentType	String
developmentPlatform	String
languageType	String
DBMS_Used	String
RELY	String
CPLX	String
TIME	String
SCED	String
productivity	String
cost	String

## Responses

Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
estimation/addVAF?id=1515998053438
```

## Response Sample

```
{}
```

## *estimation/changeVAF*

### Description

Change the adjustment factor for an estimation project.

Note: Only administrators can call this interface.



Request Method	Get
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

## Headers

Key	value
Authorization	String

## Body

Key	value
developmentType	String
developmentPlatform	String
languageType	String
DBMS_Used	String
RELY	String
CPLX	String
TIME	String
SCED	String
productivity	String
cost	String

## Responses

Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
estimation/changeVAF?id=1515998053438
```

## Response Sample

```
{}
```

# *estimation/addTree*

## Description

Add or update the project function decomposition tree for a requirement.

Note: Only users can call this interface.

Request Method	Post
Authorization	Required

## Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

## Headers

Key	value
Authorization	String

## Body

Key	value
tree	Object

## Responses

Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
url:"http://ip:8011/estimation/addTree?id=1515998053438"
body:{
  "tree" : {
    "_id" : "1515998074311",
    "name" : "tree",
    "childFolders" : [
      {
        "_id" : "00001",
        "name" : "功能模块1",
        "childFolders" : [],
        "childFiles" : [
          {
            "_id" : "1515998086763",
            "name" : "lskdjf"
          }
        ]
      }
    ]
  },
  "childFiles" : []
}
```

## Response Sample

```
{}
```

## *estimation/getTree*

### Description

Get the project function decomposition tree for a requirement..

Request Method	Get
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

## Headers

Key	value
Authorization	String

## Body

Key	value
void	void

## Responses

Code	Description	Schema
200	Successful response	<i>treeOfTransaction</i> :Object

## Request Sample

```
estimation/getTree?id=1515998053438
```

## Response Sample

```
{
  "treeOfTransactions" : {
    "_id" : "1515998074311",
    "name" : "tree",
    "childFolders" : [
      {
        "_id" : "00001",
        "name" : "功能模块1",
        "childFolders" : [],
        "childFiles" : [
          {
            "_id" : "1515998086763",
            "name" : "lskdjf"
          }
        ]
      }
    ]
  }
},
```

```
    "childFiles" : []
  },
  "allEIFData" : [
    {
      "fileName" : "qwe ",
      "allDET" : "qw  qwe"
    }
  ],
  "allILFData" : [
    {
      "fileName" : "sd",
      "allDET" : "sd d"
    }
  ]
}
```

## ***estimation/addFile***

### **Description**

Add a new file node(child function) of the project function decomposition tree.

Note: Only administrators can call this interface.

Request Method	Post
Authorization	Required

### **Parameters**

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### **Headers**

Key	value
Authorization	String

### **Body**

Key	value
name	String
id	String

## Responses

Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
estimation/addFile?id=1515998053438
```

## Response Sample

```
{}
```

## *estimation/getTransaction*

### Description

Get the details about child function of a requirement.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

## Body

Key	value
tId	String

## Responses

Code	Description	Schema
200	Successful response	<i>Transaction</i> : Object

## Request Sample

```
estimation/getTransaction?id=1515998053438
```

## Response Sample

```
{
  "transaction" : {
    "_id" : "1515998086763",
    "transactionName" : "lskdjf",
    "steps" : [
      {
        "stepName" : "sad ",
        "eifDataSets" : [
          {
            "ExternalInterfaceFileName" : "qwe ",
            "DET" : [
              "qw"
            ]
          }
        ],
        "ilfDataSets" : [
          {
            "InnerlogicalFileName" : "sd",
            "DET" : [
              "d"
            ]
          }
        ]
      }
    ],
    "estimationTransactionDatas" : [
```

```
{
  {
    {
      "name" : "asd ",
      "TransactionType" : "EI",
      "LogicalFile" : " as a as as a ",
      "DET" : "asd ad asdasd ",
      "FileNum" : 5,
      "DETNum" : 3,
      "Complexity" : "低",
      "UFP" : 3
    }
  }
}
```

## *estimation/deleteTransaction*

### Description

Get a child function from the tree of a requirement.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

### Body

Key	value
tId	String

### Responses



Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
estimation/deleteTransaction?id=1515998053438
```

## Response Sample

```
{}
```

## *estimation/updateTransaction*

### Description

Update the details about child function of a requirement.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

### Headers

Key	value
Authorization	String

### Body

Key	value
tId	String
transactionName	String
steps	List
ILFTable	List
EIFTable	List

## Responses

Code	Description	Schema
200	Successful response	<i>void</i>

## Request Sample

```
estimation/updateTransaction?id=1515998053438
```

## Response Sample

```
{}
```

## *estimation/TransactionReName*

### Description

Change the name of a child function.

Request Method	Post
Authorization	Required

### Parameters

Name	Located in	Description	Required	Schema
id	query	requirement id	yes	String

Headers

Key	value
Authorization	String

Body

Key	value
tId	String
tName	String

Responses

Code	Description	Schema
200	Successful response	<i>void</i>

Request Sample

estimation/TransactionReName?id=1515998053438

Response Sample

{}