

同濟大學

生产实习报告

生产实习单位 校内实习

实习时间 2020 年 7 月 13 日至
2020 年 8 月 7 日止

指导人员姓名 黄杰

指导教师姓名 黄杰

学 号 1751022

学 生 姓 名 李翠琪

软件学院 院(系) 软件工程 专业 二 年级

学生电子签名:

李翠琪

说 明

- 1、生产实习结束之前，每个学生都必须认真撰写《生产实习报告》。
通过撰写生产实习报告，系统地回顾和总结实习的全过程，将实践性教学的感性认知升华到一定的理论高度，从而提高实习教学效果。
- 2、实习报告的芯页使用报告纸，首页为目录。实习报告要求字迹端正，条理清晰，内容详尽，数据准确。字数一般不少于 5000 字。
- 3、实习报告的撰写应符合实习大纲和实习指导书的要求。报告内容可包括：实习日程安排，实习单位情况，专题报告记录的整理，主要设备、工艺流程，技术参数的记录和分析，专题分析，实习收获和体会，合理化建议等。还应附上必要的图纸或表格。
- 4、生产实习报告的质量反映了生产实习的质量，它是实习成绩评定的主要依据之一。生产实习报告需经实习指导人员审阅，由实习指导教师负责评分。不交实习报告者不得参加实习成绩评定。

目录

- 1 项目简介.....5
- 2 用例模型.....6
 - 2.1 用例图6
 - 2.2 用例简介.....7
 - 2.2.1 用例总览.....7
 - 2.2.2 用例详细说明 10
- 3 术语列表..... 14
- 4 项目架构..... 15
- 5 数据库设计..... 18
 - 5.1 总体 E-R 图 18
 - 5.2 数据库表设计 18
 - 5.2.1 client..... 18
 - 5.2.2 *user*..... 19
 - 5.2.3 userInfo 19
 - 5.2.4 template 20
 - 5.2.5 indicator 20
 - 5.2.6 templateIndicator..... 21
 - 5.2.7 report 22
 - 5.2.8 record..... 22
 - 5.2.9 coefficient..... 23
 - 5.2.10 unit..... 23

6.测试	24
6.1 登录退出	24
6.2 用户管理	28
6.3 模板管理	32
6.4 账户管理	35
6.5 报表管理	37
7.接口规范文档	39
8.实习体会与感想	39
8.1 学到的知识	40
8.2 遇到的困难	40
8.3 反思与经验	41

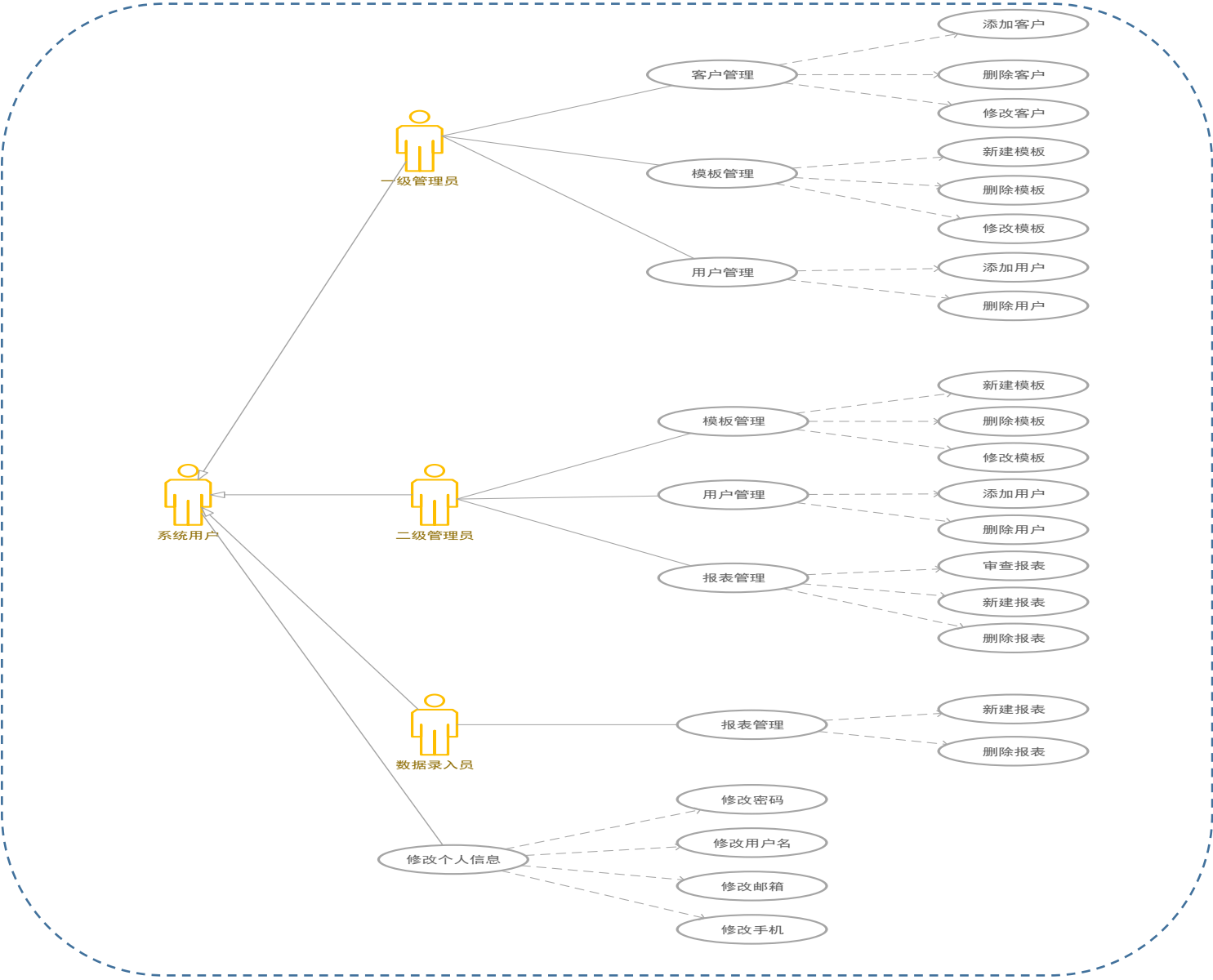
1 项目简介

ESG(Environmental 环境、Social 社会、Governance 治理)是当今国际社会衡量企业可持续发展能力的通用标准之一，弥补了单纯财务评价的不足。2016 年，中国人民银行等七部委印发的《关于构建绿色金融体系的指导意见》提出，要逐步建立和完善上市公司强制性环境信息披露制度，鼓励第三方专业机构参与采集、研究和发布企业环境信息分析报告。

本项目旨在开发一个可用的 ESG 管理系统，以满足用户 ESG 需求。

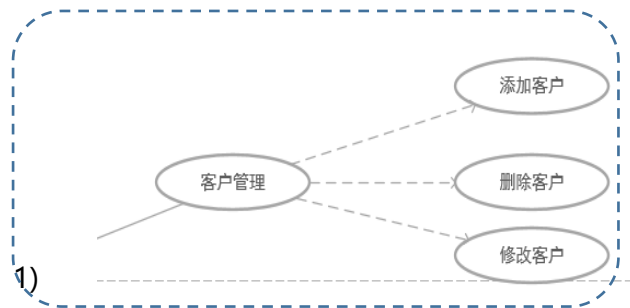
2 用例模型

2.1 用例图



2.2 用例简介

2.2.1 用例总览

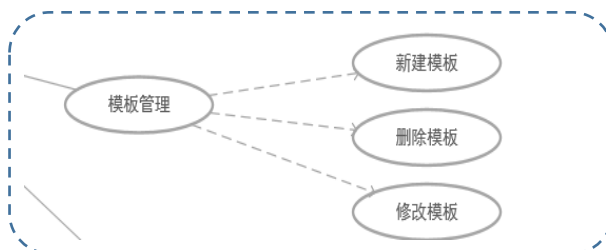


对象：一级管理员

用例：客户管理(添加客户，删除客户，修改客户)

描述：对于使用 esg 的客户(公司)进行管理。

2)

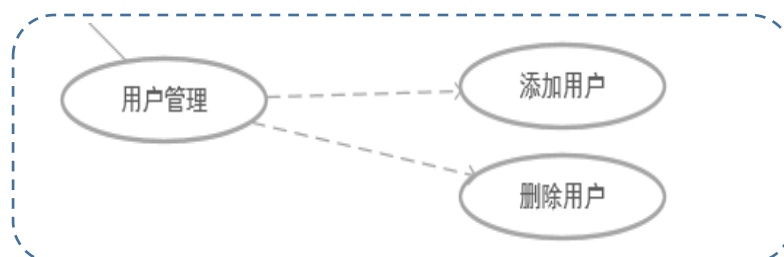


对象：一级管理员

用例：模板管理(新建模板，删除模板，修改模板)

描述：对公司录入 esg 信息的进行管理，可以为用户定制模板，对应于其特点信息。

3)

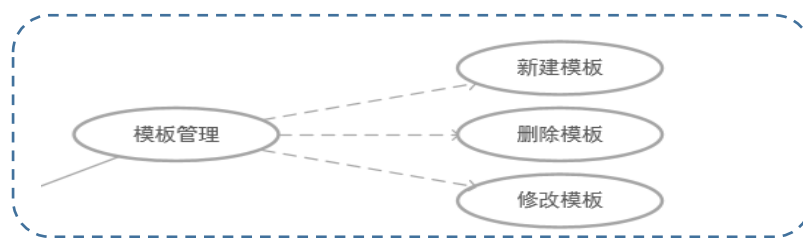


对象：一级管理员

用例：用户管理(添加用户，删除用户)

描述：用户包括低级别管理员以及数据录入员，可以为其添加删除用户，维护账户信息。

4)

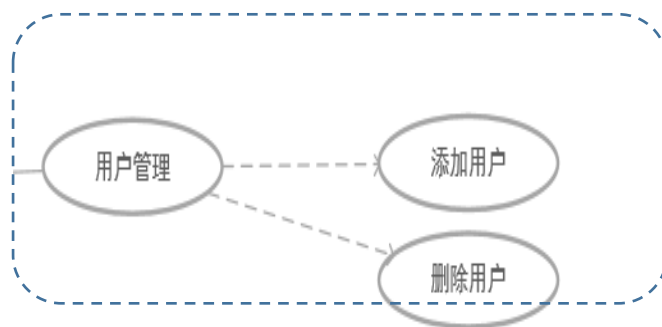


对象：二级管理员

用例：模板管理(新建模板，删除模板，修改模板)

描述：可以为所在公司进行模板的管理以及定制。

5)

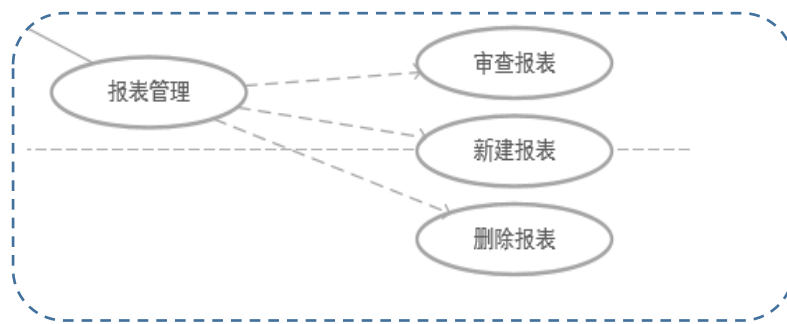


对象：二级管理员

用例：用户管理(添加用户，删除用户)

描述：用户包括低级别管理员以及数据录入员，可以为其添加删除用户，维护账户信息。

6)

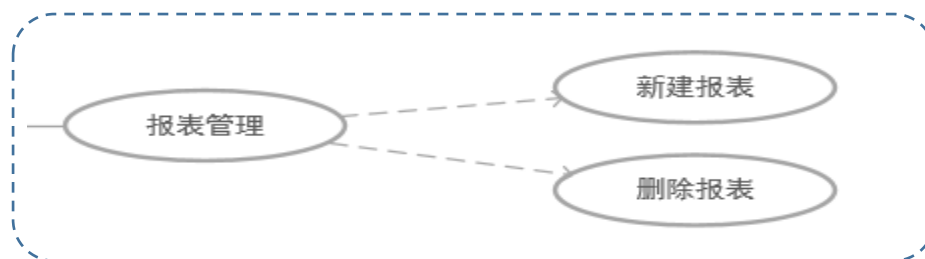


对象：二级管理员

用例：报表管理(审查报表，新建报表，删除报表)

描述：对于本公司数据录入员以及低级别公司上传的报表进行管理，对报表进行审查，同时也可以对报表进行创建，删除等等。

7)

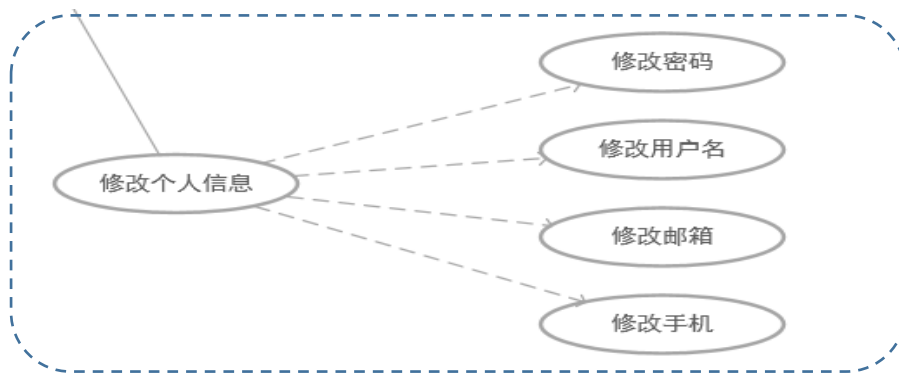


对象：数据录入员

用例：报表管理(新建报表，删除报表)

描述：数据录入员主要完成数据录入，报表填写工作，仅支持新建，删除报表的功能。

8)



对象：所有系统用户

用例：修改个人信息(修改密码，修改用户名，修改邮箱，修改手机)

描述：用户可以对自已的信息进行维护，信息包括密码，用户，邮箱，手机信息

2.2.2 用例详细说明

用例：添加客户
用例编号：001
对象：
一级管理员
事件流：
a.一级管理员进入客户管理界面，点击创建客户
b.管理员录入客户的基本信息，例如客户名称，客户所在地，客户简介等等
c.完成客户创建
前提条件：
无

用例：新建模板
用例编号：002
对象：
一级管理员
事件流：
<p>a.一级管理员进入模板管理界面，点击新建模板</p> <p>b.管理员进入模板创建页面，根据具体需求选择需要的指标。</p> <p>C.完成模板创建</p>
前提条件：
无

用例：添加用户
用例编号：003
对象：
一级管理员

事件流：

- a.一级管理员进入用户管理界面
- b.选择公司，进入相对应公司的用户查看界面
- c.在相对应的职位处点击添加用户，设置初始账户以及密码
- d.完成用户添加

前提条件：

必须有相对应的公司存在

用例：审查报表

用例编号：004

对象：

二级管理员

事件流：

- a.二级管理员进入报表管理界面，点击审查报表
- b.在报表列表中选择对应的报表进入报表查看页面
- c.完成报表审查后，可以添加审查评语，选择退回或者通过
- d.完成审查报表

前提条件：

报表已经完成创建，填写并且由低级别用户提交。

用例：新建报表
用例编号：005
对象：
二级管理员，数据录入员
事件流：
a.用户进入报表管理界面，点击新建报表
b.输入报表信息，例如报表名称，报表简介等
c.选择报表模板，点击确认
d.进入报表填写页面，在相对应的表项处填写数据
e.填写完毕后，点击提交，完成新建报表流程
前提条件：
无

用例：修改个人信息
用例编号：006
对象：
所有用户
事件流：

a.用户进入信息维护界面

b.点击相对应的要更新的数据项，对数据进行修改

c.点击保存完成个人信息修改

前提条件：

无

3 术语列表

项目	说明
客户	参加 esg 系统的各个以公司为单位的实体，系统完整服务的最小单位。
用户	实际登录进入系统并进行操作的实体，其包括各级管理员，数据录入员
指标	系统维护的各个公司的数据的项或者分类，其呈现多级结构，最后一级指标为具体的信息，而前面的多级指标都是信息的分类信息。
模板	客户录入系统的信息的集合，由于各公司属性不同，为便于各类公司的正常录入，系统提供的指标是各公司可录入信息的并集，

	实际并不需要全部填写，可由用户进行定制。
报表	各个客户一次录入的信息的集合，它在模板的基础上，由数据录入员填入实际信息。
数据录入员	系统用户，其主要工作是创建报表并把确切的数据录入系统。
管理员	系统用户，其主要工作是进行报表的管理(具体涉及到报表创建删除以及报表的审核)、用户管理(主要是对于级别低于该级别用户进行管理，包括同级别数据录入员以及附属低级别管理员)

4 项目架构

前端： vue+html5+css+jquery+ajax

后端： mybatis + springboot + mysql + redis

项目架构设计：

该项目采用的是前后端分离的设计模式，即前端页面与后端服务分离开来，通过 ajax 进行数据的传递以及服务的调用。

相对于传统的 MVC 模式，前后端分离的设计模式更加符合模块化开发的设计思想，对于传统的 layer 模式以及 MVC 模式，前端的代码与后端代码将会共同进行开发，前端作为后端 MVC 中 V (即 view) 的组成部分，这样会使得系统具有较强的耦合性，是的开发不能具有一定的独立性，且这样使得所有的页面返回都需要从后端进行，增加后端服务器负荷。所以前后端分离的设计模式是目前的大趋势，也是我们项目中我们经过考虑最终采取的方法。

设计过程:

开发过程中,我们将项目设计成三个板块,前端,后端,中间层,前端即是又 vue 编写的展示页面,后端是 Springboot 开发的负责从数据库中取出数据加以处理,返回前端的服务端,而中间层主要是以 ajax 服务为基础的交互层,这一层的基础主要是通过约定好的接口方式进行的,前端以及后端成员并不需要了解对方的具体设计与实现,只需要提供以及调用获取一定数据的接口即可。

项目伊始,小组成员先总体估计了前端所需的数据,完成后设计出了接口供前端同学直接调用,通过 SwaggerUI,将后端接口名称以及相关信息以文档形式展现给前端同学,前端同学只需通过文档信息,即可调用接口进行交互。

在完成了接口设计的基础上,前后端同学进行并行开发。

前端:

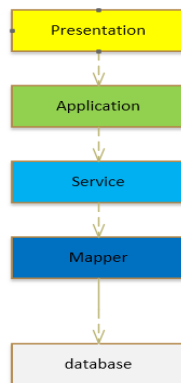
前端同学通过 vue.js 进行开发, vue 是一种渐进式,响应式开发的前端框架。小组成员充分利用了 vue 开发的组件和路由思想,按照不同的分类以及不同的功能,前端同学将前端不同的页面,分成了不同的组件,例如登录,例如导航等等,之后将组件进行分工,最终通过路由的方式,使得页面可以通过不同的事件展示不同的组件,实现了页面跳转,并且相较于传统的 html5+css+js,页面跳转,以及动态变化等等的数据传递,数据绑定在 vue 中能够以非常方便的方式进行。

后端:

后端同学采用 Springboot 作为后端框架, Springboot 可以通过标注的方式,完成请

求映射，数据库语句映射，以及注入，同时通过 mybatis 作为中间件，在 mapper 层将数据库语言映射到 java 函数中，再通过 service 层对外的 controller 提供访问数据的接口，实现了数据库访问以及映射的控制器有效隔离，同时简单的配置方法也使得开发效率提升。

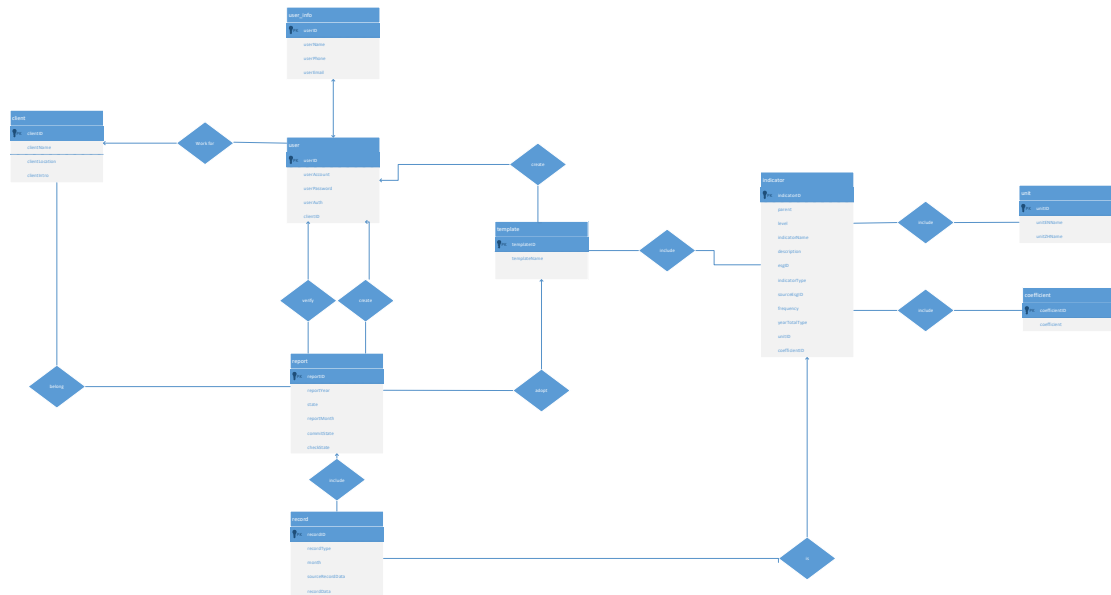
整体的层次如图所示



其中 presentation，也就是前端界面，想用户展示，application 相当于传统 mvc 模式中的 controller，负责接受前端用户的访问请求并且根据不同的 url 提供不同的服务，service 即是由 mybatis 工具向控制器提供的数据库访问接口，mapper 则是将具体的 mysql 语言映射到 service 接口的具体实现函数上，最终由 mapper 与数据库交互。

5 数据库设计

5.1 总体 E-R 图



5.2 数据库表设计

5.2.1 client

create table client

(

clientID char(20) primary key

clientName char(40) not null unique

clientLocation char(20)

```
        clientIntro varchar(500)
    )
```

5.2.2 *user*

```
create table user
(
    userID char(20) primary key
    userAccount char(20) not null
    userPassword char(20) not null
    userAuth int
    clientID char(20)
)
```

5.2.3 *userInfo*

```
create table userInfo
(
    userID char(20) primary key
    userName char(20)
    userPhone char(20)
    userEmail char(30)
)
```

5.2.4 template

```
create table template
(
    templateID char(20) primary key
    clientID char(20)
    templateInfo char(20)
    templateName char(20) unique
)
```

5.2.5 indicator

```
create table indicator
(
    //common property of indicator

    indicatorID char(20) primary key

    parent char(20)

    level int

    indicatorName char(40)

    description char(200)
```

```

//specific property of leaf indicator

ESGIId char(20)

indicatorType int//0-定量 1-定性 2-系数计算

//property for coefficient calculation

sourceESGIId char(20)

coeffientID char(20) foreign key

unitID char(20) foreign key

frequency int //0-年 1-月

yearTotalType int //0-mean 1-sum

)

```

5.2.6 templateIndicator

```

create table templateIndicator

(

    indicatorID char(20) primary key

    templateID char(20) primary key

)

```

5.2.7 report

```
create table report
(
    reportID char(20) primary key
    templateID char(20)
    reporterID char(20) foreign key
    reportYear int
    reportMonth int
    commitState int
    checkState int
)
```

5.2.8 record

```
create table record
(
    recordID char(20) primary key
    reportID char(20) foreign key referencing report
    indicatorID char(20) foreign key referencing template
    recordType int //text or numeric
    sourceRecordData char(50)
    recordData char(50)
)
```

5.2.9 coefficient

```
create table coefficient  
(  
    coefficientID char(20) primary key  
    coefficient numeric(20,8)  
)
```

5.2.10 unit

```
create table unit  
(  
    unitID char(20) primary key  
    unitZHName char(50)  
    unitENName char(50)  
)
```

6.测试

6.1 登录退出

DLTC-001：登录退出-ESG 管理员登陆系统	
测试目的：	<ul style="list-style-type: none">验证 ESG 管理员能否登录系统
测试条件：	<ol style="list-style-type: none">已成为 ESG 管理员；
测试过程：	<ol style="list-style-type: none">输入 ESG 管理系统网站；在管理员以及密码处输入相应信息；点击登录按钮
期望结果：	<ol style="list-style-type: none">能够进入系统管理界面
测试结果：	通过

说明：	日期：2020.9.3
DLTC-002：登录退出-集团公司管理员登录	
<p>测试目的：</p> <ul style="list-style-type: none"> 验证集团公司管理员能否登录系统 	
<p>测试条件：</p> <ol style="list-style-type: none"> 管理员建立公司 管理员已分配账号 	
<p>测试过程：</p> <ol style="list-style-type: none"> 输入 ESG 管理系统网站； 在账号以及密码处输入相应信息； 点击登录按钮 	
<p>期望结果：</p> <ol style="list-style-type: none"> 能够进入系统管理界面 	
测试结果： 通过	
说明：	日期：2020.9.3

DLTC-003：登录退出-公司录入员登陆系统	
测试目的： <ul style="list-style-type: none">• 验证公司录入员能否登录系统	
测试条件： <ol style="list-style-type: none">1. ESG 已创建公司2. 上一级分配账号	
测试过程： <ol style="list-style-type: none">1. 输入 ESG 管理系统网站；2. 在账号以及密码处输入相应信息；3. 点击登录按钮	
期望结果： <ol style="list-style-type: none">1. 能够进入系统管理界面	
测试结果： 通过	
说明：	日期： 2020.9.3

DLTC-004: 登录退出-退出系统	
测试目的: <ul style="list-style-type: none">• 验证集团公司能否正常退出系统	
测试条件: <ol style="list-style-type: none">1. 已登入账号	
测试过程 : <ol style="list-style-type: none">1. 登入系统状态下, 点击右上角注销退出登录;	
期望结果: <ol style="list-style-type: none">1.能够回到 ESG 信息管理系统登录界面	
测试结果: 通过	
说明:	日期 : 2020.9.3

6.2 用户管理

YHGL-001：用户管理-ESG 管理员创建用户	
测试目的：	<ul style="list-style-type: none">验证 ESG 管理员是否能够创建用户
测试条件：	<ol style="list-style-type: none">账号已成为 ESG 管理员
测试过程：	<ol style="list-style-type: none">输入账号密码登入管理系统；点击公司管理按钮；点击添加公司按钮；选择用户名称，密码，公司名称，位置以及简介；输入完毕点击确定按钮；
期望结果：	<ol style="list-style-type: none">在用户管理界面可以看到新创建的用户；创建用户模板正确；
测试结果：	通过

说明：	日期：2020.9.3
YHGL-002：用户管理-ESG 管理员查看用户信息	
<p>测试目的：</p> <ul style="list-style-type: none"> 验证集团公司管理员是否能够查看用户信息 	
<p>测试条件：</p> <ol style="list-style-type: none"> 账号已成为 ESG 管理员 已有用户被创建 	
<p>测试过程：</p> <ol style="list-style-type: none"> 输入账号密码登入管理系统； 点击公司管理按钮； 在公司管理界面查看某公司的名称，位置，简介等 点击用户名称或查看详情按钮进入模板信息界面查看模板信息； 	
<p>期望结果：</p> <ol style="list-style-type: none"> 可以在用户管理界面可以看到公司的名称、模板等信息； 可以通过点击用户名称或者查看报表按钮进入模板信息界面查看模板信息； 	
测试结果： 通过	

说明：	日期：2020.9.3
YHGL-003：用户管理-ESG 管理员删除用户	
测试目的： <ul style="list-style-type: none"> 验证 ESG 管理员能够删除用户 	
测试条件： <ol style="list-style-type: none"> 账号已成为 ESG 管理员 已有用户被创建 	
测试过程： <ol style="list-style-type: none"> 输入账号密码登入管理系统； 点击公司管理管理按钮； 找到要删除的公司，点击删除按钮； 	
期望结果： <ol style="list-style-type: none"> 用户被成功删除，在用户管理界面不再出现该用户； 	
测试结果： 通过	
说明：	日期：2020.9.3

YHGL-004：用户管理-ESG 管理员修改用户绑定模板

测试目的：

- 验证 ESG 管理员能够修改用户绑定模板

测试条件：

1. 账号已成为 ESG 管理员
2. 已有用户以及子用户被创建

测试过程：

1. 输入账号密码登入管理系统；
2. 点击公司管理按钮；
3. 找到待操作公司，点击查看详情；
4. 点击右侧弹出框的模板管理界面
5. 可以修改用户目前所有的模板

期望结果:

1. 操作公司的模板修改成功且正确；

测试结果： 通过

说明：	日期：2020.9.3
-----	-------------

6.3 模板管理

MBGL-001：模板管理-ESG 管理员创建模板	
测试目的：	2. 验证 ESG 管理员能否创建模板。
测试条件：	1. 已成为 ESG 管理员
测试过程：	<ol style="list-style-type: none"> 1. 输入 ESG 管理员用户名密码，登录 ESG 管理员账号； 2. 点击模板管理按钮； 3. 点击新建模板按钮； 4. 输入模板名称、所属公司以及信息； 5. 点击编辑模板 6. 进入模板指标选择界面，选择模板中所需指标； 7. 点击确定按钮；

期望结果: 1. 能成功创建模板。	
测试结果： 通过	
说明:	日期： 2020.9.3

MBGL-002：模板管理-ESG 管理员修改模板

测试目的：

1. 验证 ESG 管理员能否创建模板。

测试条件：

1. 已成为 ESG 管理员
2. 已有存在的模板

测试过程：

1. 输入 ESG 管理员用户名密码，登录 ESG 管理员账号；
2. 点击模板管理按钮；
3. 点击待修改模板名称；
4. 点击编辑模板；

<p>5. 进入模板指标选择界面，修改模板中指标；</p> <p>6. 点击确定按钮；</p>	
<p>期望结果:</p> <p>1. 能成功修改模板。</p>	
<p>测试结果： 通过</p>	
<p>说明:</p>	<p>日期：2020.9.3</p>
<p>MBGL-003：模板管理-ESG 管理员删除模板</p>	
<p>测试目的:</p> <p>3. 验证 ESG 管理员能否删除模板。</p>	
<p>测试条件:</p> <p>2. 已成功创建了模板。</p>	
<p>测试过程：</p> <p>2. 输入 ESG 管理员用户名密码，登录 ESG 管理员账号；</p> <p>3. 点击模板管理按钮；</p> <p>4. 选中待删除的模板</p>	

<p>5. 点击删除模板按钮;</p> <p>6. 测试能否删除模板。</p>	
<p>期望结果:</p> <p>3. 能成功删除模板。</p>	
<p>测试结果: 通过</p>	
<p>说明:</p>	<p>日期 : 2020.9.3</p>

6.4 账户管理

ZHGL-001: 账户管理-ESG 管理员创建账户	
<p>测试目的:</p> <p>1. 验证 ESG 管理员能否创建账户。</p>	
<p>测试条件:</p> <p>1. 账户所属的公司已存在。</p>	
<p>测试过程 :</p> <p>1. 输入 ESG 管理员用户名密码, 登录 ESG 管理员账号;</p>	

2. 点击账户管理按钮进入平台所有公司的管理页面； 3. 点击新建公司 4. 在对应的新建界面输入账户账号 5. 点击确定按钮，测试能否创建正常账户； 6. 第 4 步输入重名账户，测试能够创建重名账户。	
期望结果： 1. 成功创建账户； 2. 不能成功创建重名账户。	
测试结果： 通过	
说明：	日期：2020.9.3
ZHGL-002：账户管理-ESG 管理员删除账户	
测试目的： 2. 验证 ESG 管理员能否删除账户。	
测试条件： 1. 账户已存在。	

测试过程： <ol style="list-style-type: none"> 1. 输入 ESG 管理员用户名密码，登录 ESG 管理员账号； 2. 点击公司管理按钮进入平台所有公司的管理页面； 3. 或点击相应的公司按钮； 4. 在对应的公司管理界面点击删除公司按钮； 5. 测试能否删除账户。 	
期望结果： <ol style="list-style-type: none"> 1. 成功删除账户。 	
测试结果： 通过	
说明：	日期：2020.9.3

6.5 报表管理

ZHGL-001：公司录入员新建普通报表	
测试目的： <ol style="list-style-type: none"> 1. 验证公司录入员能否创建普通报表。 	
测试条件：	

1. 模板已存在。	
测试过程： <ol style="list-style-type: none"> 1. 输入公司录入员用户名密码，登录公司录入员账号； 2. 点击新建报表按钮； 3. 选择模板 4. 点击确定按钮。 	
期望结果： <ol style="list-style-type: none"> 1. 成功创建报表。 	
测试结果： 通过	
说明：	日期：2020.9.3
ZHGL-002：公司录入员删除报表	
测试目的： <ol style="list-style-type: none"> 3. 验证公司录入员能否删除报表。 	
测试条件： <ol style="list-style-type: none"> 1. 报表已存在。 	

测试过程： 1. 输入公司录入员用户名密码，登录公司录入员账号； 2. 点击报表的删除按钮；	
期望结果： 1. 成功删除按钮。	
测试结果： 通过	
说明：	日期： 2020.9.3

7.接口规范文档

本次项目的接口设计由后端同学设计，利用 SwaggerUI 与 Springboot 整合，将接口文档以可视化形式进行展示，接口已经部署在
<http://47.103.203.188:4396/swagger-ui.html#/>

8.实习体会与感想

这一个月暑假实习收获满满,我们根据老师的指导和所给资料，通过模仿 ESG 成品项目，小组自己搭建了一个简易的前后端分离 ESG 项目，并且部署到了云服务器上。通过从无到有完整的走了一遍项目实现流程，并动手实现了后端，学习到了很多知识，明白了团

队协作的重要性。也遇到了很多困难和问题，通过进一步的反思，为将来的项目积累了很多的经验。

8.1 学到的知识

1. 学会了如何使用 GitHub 进行代码版本控制和团队协作。例如如何在本地 push、pull 代码到远程仓库，当两个人代码冲突时如何解决，如何优雅的使用合并分支等。
2. 学会了如何操作云服务器和 linux 系统。熟悉了 linux 命令行的常用命令。如何把自己的项目部署到云服务器上。例如常用的：netstat -tunlp |grep 4396，查看某个端口是否已经被占用；nohup java -jar ESG.jar 实际在云服务器上部署项目等等。
3. 明白了如何搭建 Spring Boot 应用，如何用 Mybatis 整合连接数据库 MySQL，如何用三层架构(表示层，业务逻辑层，数据访问层)的思想实现后端应用。
4. 明白了什么是前后端分离，如何与前端同学进行 API 接口交互。明白如何用 Swagger 展示后端接口并测试后端接口逻辑的正确性。明白了如何传递符合阅读规范的 json 给前端。
5. 从实战中明白了一个数据库如何设计更减少冗余，以及在后端中如何写出优秀的 DML 语句进行增删改查，更加提高性能等等。

8.2 遇到的困难

在做项目时碰到了很多 bug，虽然报错那瞬间很难受。之后进行调试，找资料，解决 bug 所花的时间与精力也很久。但是解决问题，成功运行后还是很感动。

1. 遇到了服务器跨域访问问题，因为是前后端分离，不是在同一个 IP 地址上，出现了跨域问题，最后通过搜索资料，在 Spring Boot 中添加了 `@CrossOrigin` 注解以支持跨域访问。
2. 阿里云服务器上没开放相应端口导致 Swagger 网址不能正常访问。最后通过在阿里云

服务器上添加防火墙规则解决问题。

3. MySQL 中文乱码问题，最后统一使用了 UTF-8 编码避免出现乱码。
4. 指标的目录树不知道如何构建问题。最后对应每个五级都创建一个集合，然后通过表中对应的父指标字段不断递归寻找父指标并加入集合，从而构造出某个五级指标的目录树。

8.3 反思与经验

在项目中也走了许多的弯路，导致进度有些延缓，在做完项目后，我总结出以下几点经验，在以后的项目中可以更快更好地完成项目：

1. 小组需要先头脑风暴，充分交流意见。把需求先完全敲定，把用例图,用例规约等规定下来，建立数据库。
2. 分工要明确。且不要立刻开始写代码，要先不断讨论，完善业务逻辑，使得业务逻辑各种细节经得住推敲。接下来根据分工前后端对接进行讨论。
3. 前后端组长要先把框架搭建好,并且定下小组内的代码规范,接口规范等。
4. 前后端交接的人要充分交流需求，把接口按规范约定好,再讨论细节问题。最后小组再汇总，完成需求分析，之后再开始具体实现。
5. 每周定下一个 ddl,定时汇报,及时进行讨论修改。一方面能防止组内成员懒惰，也能更好地把握整体的项目进度等。
6. 写代码时要实时完善文档，并且写的时间不宜推太长，以免忘记细节。
7. 合理使用 git 进行代码管理，并且符合 git 分支/提交规范等。