

# 数据库 Lab 03 实验报告

银行业务管理系统系统设计与实现报告

PB19071405 王昊元

2022 年 06 月 10 日

# 目录

<b>1</b>	<b>概述</b>	<b>1</b>
1.1	系统目标 . . . . .	1
1.2	需求说明 . . . . .	1
1.3	本报告的主要贡献 . . . . .	1
<b>2</b>	<b>总体设计</b>	<b>2</b>
2.1	系统模块结构 . . . . .	2
2.2	系统工作流程 . . . . .	2
2.3	数据库设计 . . . . .	3
<b>3</b>	<b>详细设计（部分）</b>	<b>7</b>
<b>4</b>	<b>实现与测试</b>	<b>8</b>
4.1	实现结果 . . . . .	8
4.2	测试结果 . . . . .	12
4.3	实现中的难点问题及解决 . . . . .	16
<b>5</b>	<b>总结与讨论</b>	<b>16</b>

# 1 概述

## 1.1 系统目标

开发一个银行业务管理系统,该系统基于 MySQL,使用 Python 进行后端开发,使用 HTML+CSS+JavaScript 进行前端开发,主要以谷歌浏览器作为目标浏览器,实现客户、账户、贷款、业务以及其它方面的去求。

## 1.2 需求说明

银行有多个支行。各个支行位于某个城市,每个支行有唯一的名字。银行要监控每个支行的资产。

银行的客户通过其身份证号来标识。银行存储每个客户的姓名、联系电话以及家庭住址。为了安全起见,银行还要求客户提供一位联系人的信息,包括联系人姓名、手机号、Email 以及与客户的关系。客户可以有帐户,并且可以贷款。客户可能和某个银行员工发生联系,该员工是此客户的贷款负责人或银行帐户负责人。

银行员工也通过身份证号来标识。员工分为部门经理和普通员工,每个部门经理都负责领导其所在部门的员工,并且每个员工只允许在一个部门内工作。每个支行的管理机构存储每个员工的姓名、电话号码、家庭地址、所在的部门号、部门名称、部门类型及部门经理的身份证号。银行还需知道每个员工开始工作的日期,由此日期可以推知员工的雇佣期。

银行提供两类帐户——储蓄帐户和支票帐户。帐户可以由多个客户所共有,一个客户也可开设多个账户,但在一个支行内最多只能开设一个储蓄账户和一个支票账户。每个帐户被赋以唯一的帐户号。银行记录每个帐户的余额、开户日期、开户的支行名以及每个帐户所有者访问该帐户的最近日期。另外,每个储蓄帐户有利率和货币类型,且每个支票帐户有透支额。每笔贷款由某个分支机构发放,能被一个或多个客户所共有。每笔贷款用唯一的贷款号标识。银行需要知道每笔贷款所贷金额以及逐次支付的情况(银行将贷款分几次付给客户)。虽然贷款号不能唯一标识银行所有为贷款所付的款项,但可以唯一标识为某贷款所付的款项。对每次的付款需要记录日期和金额。

主要功能需求:

- 客户管理: 提供客户所有信息的增、删、改、查功能; 如果客户存在着关联账户或者贷款记录, 则不允许删除
- 账户管理: 提供账户开户、销户、修改、查询功能, 包括储蓄账户和支票账户; 账户号不允许修改
- 贷款管理: 提供贷款信息的增、删、查功能, 提供贷款发放功能; 贷款信息一旦添加成功后不允许修改; 要求能查询每笔贷款的当前状态 (未开始发放、发放中、已全部发放); 处于发放中状态的贷款记录不允许删除
- 业务统计: 按业务分类 (储蓄、贷款) 和时间 (月、季、年) 统计各个支行的业务总金额和用户数, 统计的结果以表格形式展示

## 1.3 本报告的主要贡献

- 建立数据库结构模型
- 展示数据库各个模块的实现
- 对各个模块进行测试评估

- 进行实验总结

## 2 总体设计

### 2.1 系统模块结构

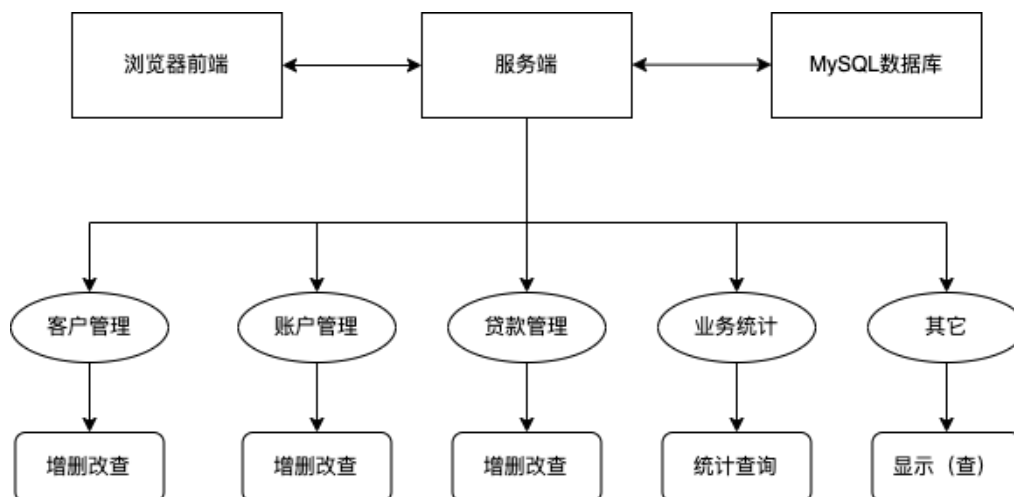


图 1: 系统模块结构示意图

- 实现采用浏览器/服务器架构,基于 MySQL 数据库,使用 Python 进行后端开发,使用 HTML+CS+JavaScript 进行前端开发
- 主要实现客户管理（增删改查）、账户管理（增删改查）、贷款管理（增删改查）、业务统计（查询并统计）及其它一些信息的查询功能
- 尝试开发用户登陆模块,但由于能力有限,功能还处于使用数据库记录注册账户信息阶段,并未涉及任何权限相关

### 2.2 系统工作流程

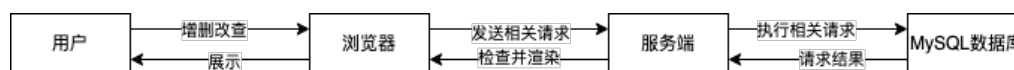


图 2: 系统工作流程示意图

- 增加
  - 用户在输入框中输入信息（前端进行一些检查，例如必填项）
  - 用户提交信息后，后端根据内容生成 MySQL 语句
  - 对数据库执行 MySQL 语句，将数据写入数据库
  - 后端对数据库的执行情况进行检查，并将结果返回用户

- 用户点击要删除的信息所对应的删除按钮
- 前端将对应要删除的信息传输给后端
- 后端根据信息生成 MySQL 语句并执行
- 根据数据库的执行情况进行检查，并将结果返回用户

- 用户点击要修改的信息对应的修改按钮
- 用户填写新的信息
- 前端将要修改的信息和修改后的信息传输给后端
- 后端根据信息生成 MySQL 语句并执行
- 根据数据库的执行情况进行检查，并将结果返回用户

- 用户在输入框中输入查询信息
- 用户提交查询信息
- 前端将查询信息传输给后端
- 后端根据信息生成 MySQL 语句并执行
- 对查询结果进行检查，并将结果返回用户

```

1  # 账户(账户号, 支行名, 余额, 开户日期)
2  create table Account
3  (
4      Account_ID varchar(50) not null,
5      Bank_Name  varchar(50) not null,
6      Balance    float(15),
7      Opening_Date date,
8      primary key (Account_ID)
9  );
10
11 # 支付(用户号, 贷款号, 支付号, 支付金额, 支付日期)
12 create table Pay
13 (
14     Client_ID varchar(50) not null,
15     Loan_ID   varchar(50),
16     Pay_ID    varchar(50) not null,
17     Pay_Amount float(15),
18     Pay_Date  date,
19     primary key (Client_ID, Loan_ID, Pay_ID)
20 );
21
22 # 支行(支行, 城市, 资产)
23 create table Bank(
24     Bank_Name varchar(50) not null,
25     City      varchar(50) not null,
26     Assets    float(15) not null,
27     primary key (Bank_Name)
28 );
29
30 # 支票账户(账户号, 透支额)
31 create table Cheque_Account
32 (
33     Account_ID varchar(50) not null,
34     Overdraft  float(15),
35     primary key (Account_ID)
36 );
37
38 # 客户(客户号, 客户名, 客户电话, 客户地址)
39 create table Client
40 (
41     Client_ID varchar(50) not null,
42     Client_Name varchar(50) not null,
43     Client_Tel  varchar(50),
44     Client_Address varchar(50),
45     primary key (Client_ID)
46 );
47

```

```

48 # 联系人(客户号, 联系人姓名, 联系人邮箱, 联系人电话, 联系人和客户的关系)
49 create table Contact
50 (
51     Client_ID varchar(50) not null,
52     Contact_Name varchar(50) not null,
53     Contact_Email varchar(50),
54     Contact_Tel varchar(50),
55     Relation varchar(50),
56     primary key (Client_ID, Contact_Name)
57 );
58
59 # 部门(部门号, 部门名, 部门类型, 经理身份证号)
60 create table Department
61 (
62     Department_ID varchar(50) not null,
63     Department_Name varchar(50) not null,
64     Department_Type varchar(50),
65     Manager_ID varchar(50),
66     primary key (Department_ID)
67 );
68
69 # 员工(员工号, 员工名, 支行名, 部门名, 员工电话, 员工地址, 入职日期)
70 create table Employee
71 (
72     Employee_ID varchar(50) not null,
73     Employee_Name varchar(50) not null,
74     Bank_Name varchar(50) not null,
75     Department_ID varchar(50),
76     Employee_Tel varchar(50),
77     Employee_Address varchar(50),
78     Work_Date date,
79     primary key (Employee_ID)
80 );
81
82 # 贷款(贷款号, 支行名, 贷款总额, 状态, 已支付贷款)
83 create table Loan
84 (
85     Loan_ID varchar(50) not null,
86     Bank_Name varchar(50) not null,
87     Loan_Amount float(15) not null,
88     Loan_Status int default 0 not null,
89     Pay_already float(15) not null,
90     Opening_Date date,
91     primary key (Loan_ID)
92 );
93
94 # 持有(客户号, 最近访问日期, 账户名)
95 create table Own

```

```

96  (
97      Client_ID varchar(50) not null,
98      Visited_Date date,
99      Account_ID varchar(50),
100     primary key (Client_ID, Account_ID)
101 );
102
103 # 开户约束(客户号, 银行名, 账户类型)
104 create table Checking
105 (
106     Client_ID varchar(50) not null,
107     Bank_Name varchar(50) not null,
108     Account_Type int not null,
109     primary key (Client_ID, Bank_Name, Account_Type)
110 );
111
112 # 储蓄账户(账户号, 利率, 货币类型)
113 create table Saving_Account
114 (
115     Account_ID varchar(50) not null,
116     Interest_Rate float(15),
117     Currency_Type varchar(50),
118     primary key (Account_ID)
119 );
120
121 # 服务(客户号, 员工号, 服务类型: 该员工是此客户的贷款负责人或银行帐户负责人)
122 create table Service
123 (
124     Client_ID varchar(50) not null,
125     Employee_ID varchar(50) not null,
126     Service_Type varchar(50),
127     primary key (Client_ID, Employee_ID)
128 );
129
130 alter table Account add constraint FK_Open foreign key (Bank_Name)
131     references Bank (Bank_Name) on delete restrict on update restrict;
132
133 alter table Pay add constraint FK_Apply foreign key (Client_ID)
134     references Client (Client_ID) on delete restrict on update restrict;
135
136 alter table Cheque_Account add constraint FK_Account_Type foreign key (Account_ID)
137     references Account (Account_ID) on delete restrict on update restrict;
138
139 alter table Contact add constraint FK_Have foreign key (Client_ID)
140     references Client (Client_ID) on delete restrict on update restrict;
141
142 alter table Employee add constraint FK_Belong_To foreign key (Department_ID)
143     references Department (Department_ID) on delete restrict on update restrict;

```



```

144
145 alter table Employee add constraint FK_Employ foreign key (Bank_Name)
146     references Bank (Bank_Name) on delete restrict on update restrict;
147
148 alter table Loan add constraint FK_Make_Loan foreign key (Bank_Name)
149     references Bank (Bank_Name) on delete restrict on update restrict;
150
151 alter table Own add constraint FK_Own1 foreign key (Client_ID)
152     references Client (Client_ID) on delete restrict on update restrict;
153
154 alter table Own add constraint FK_Own2 foreign key (Account_ID)
155     references Account (Account_ID) on delete restrict on update restrict;
156
157 alter table Saving_Account add constraint FK_Account_Type2 foreign key (Account_ID)
158     references Account (Account_ID) on delete restrict on update restrict;
159
160 alter table Service add constraint FK_Service foreign key (Client_ID)
161     references Client (Client_ID) on delete restrict on update restrict;
162
163 alter table Service add constraint FK_Service2 foreign key (Employee_ID)
164     references Employee (Employee_ID) on delete restrict on update restrict;
165
166 alter table Checking add constraint FK_Checking1 foreign key (Client_ID)
167     references Client (Client_ID) on delete restrict on update restrict;
168
169 alter table Checking add constraint FK_Checking2 foreign key (Bank_Name)
170     references Bank (Bank_Name) on delete restrict on update restrict;

```

### 3 详细设计（部分）

因为个人设计得较为简单，输入基本都是必要的信息，输出基本都是响应的页面，程序流程图也较为简单，所以就不详细展开每个模块单独讲了。在这里对几个个人认为需要说明的点进行简单说明

- 每个部分要尽力处理可能出现的能想到的异常情况，其余的异常情况要显示在 log 中。例如：删除一个有账户的客户是非法的、删除一个未结算清的贷款是非法的、贷款支付金额是不能大于贷款金额的、金额是不能为负的等等等等。
- 用户提供信息有误的情况/用户操作非法的情况，需要提示用户，例如：

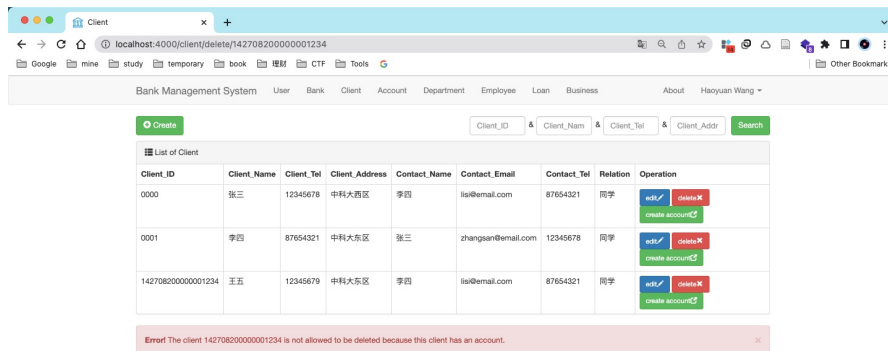


图 4: 异常情况提示示例

- 对于信息的正确性和合理性，前端和后端都应尽力去检查（这点我做的可能不够完美，但我尽力地做了）

## 4 实现与测试

### 4.1 实现结果

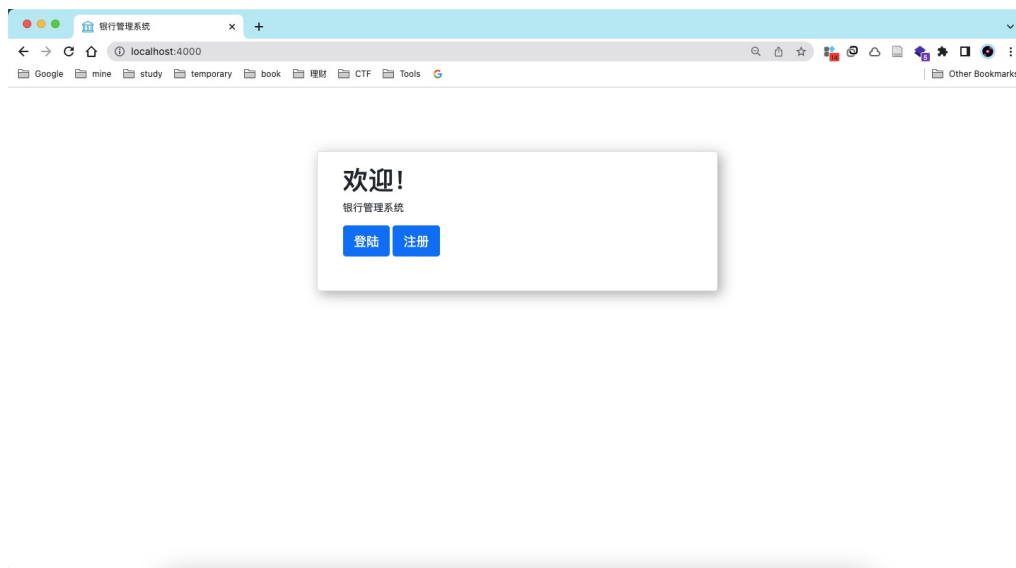


图 5: 首页

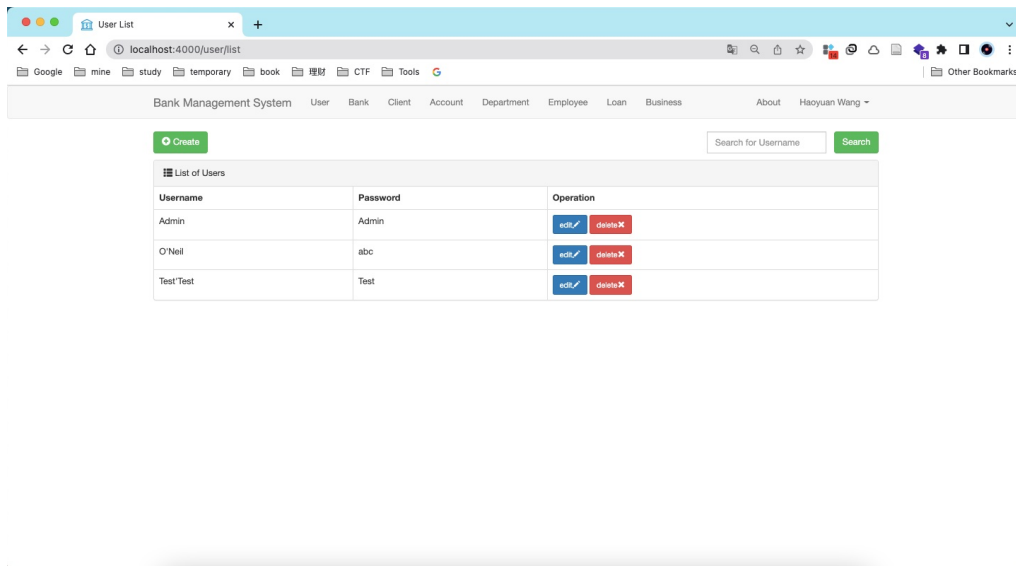


图 6: 系统用户列表

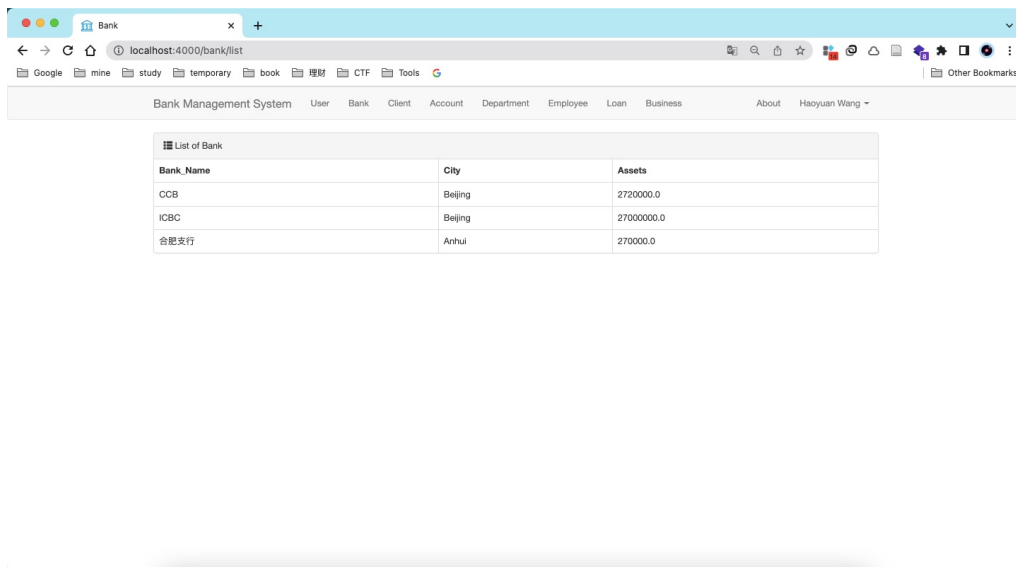


图 7: 支行列表

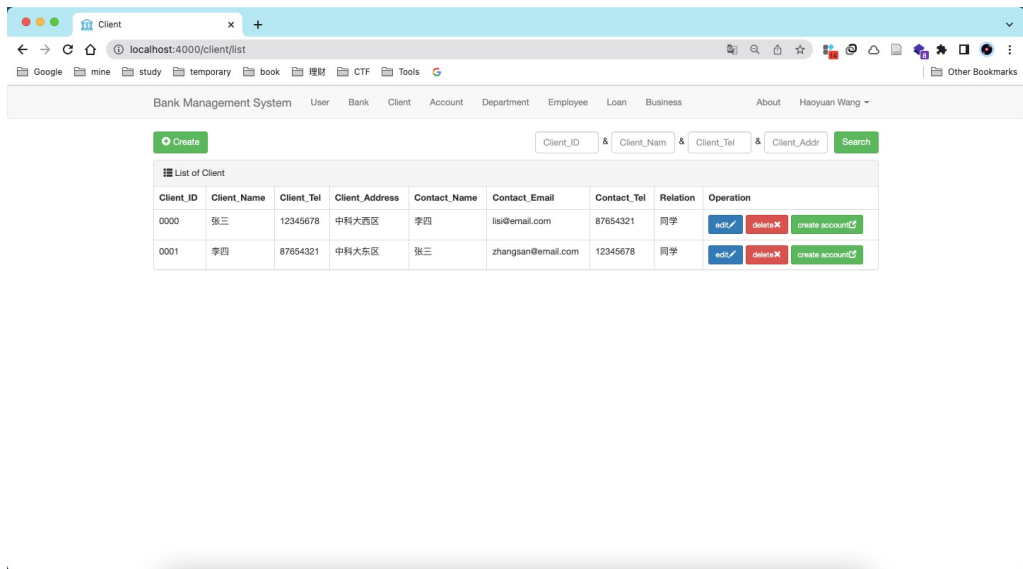


图 8: 客户列表

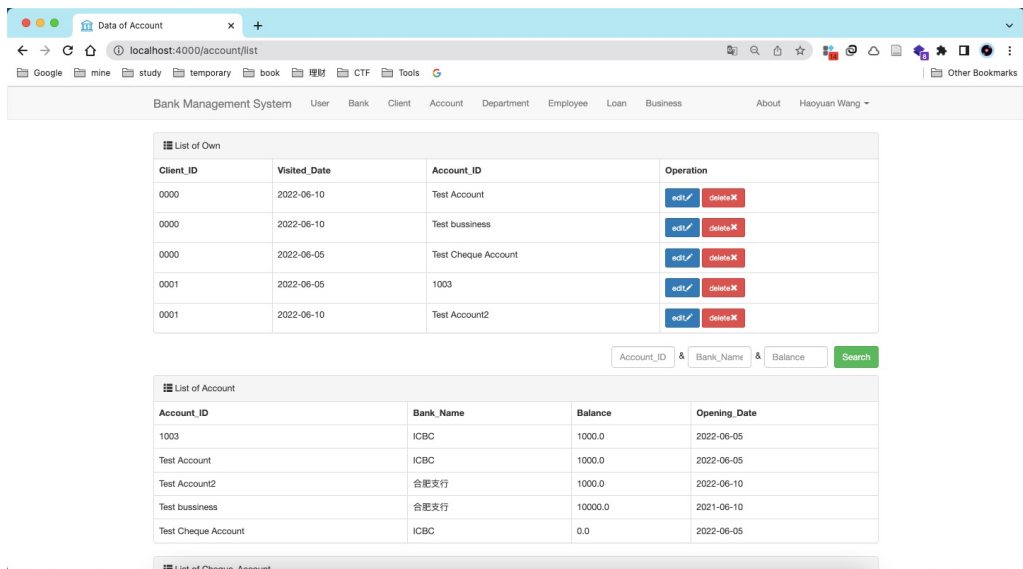


图 9: 账户列表

Bank Management System   User   Bank   Client   Account   Department   Employee   Loan   Business   About   Haoyuan Wang ▾

List of Department

Department_ID	Department_Name	Department_Type	Manager_ID
0000	Test	Test	1000
0001	Sale	Sale	1001
0002	Personnel	Personnel	1002

图 10: 部门列表

Bank Management System   User   Bank   Client   Account   Department   Employee   Loan   Business   About   Haoyuan Wang ▾

Create

List of Employee

Employee_ID	Employee_Name	Bank_Name	Department_ID	Employee_Tel	Employee_Address	Work_Date
1000	Test	合肥支行	0000	10010001000	安徽合肥中科大	2022-05-26

图 11: 职员列表

Bank Management System User Bank Client Account Department Employee Loan Business About Hasyuan Wang

Create Loan\_ID & Bank\_Name & Loan\_Amount & Loan\_Status & Pay\_already Search

List of Loan

Loan_ID	Bank_Name	Loan_Amount	Loan_Status	Pay_already	Opening_Date	Operation
Test Delete	合肥支行	2000.0	Being issued	1000.0	2022-06-10	delete

Create

List of Loan-Pay

Client_ID	Loan_ID	Pay_ID	Pay_Amount	Pay_Date
0000	1000	1000	1000.0	2022-06-06
0000	1000	1001	0.0	2022-06-07
0000	Test Delete	1002	1000.0	2022-06-10
0000	Test Delete2	1003	500.0	2022-06-10
0000	Test Delete2	1004	500.0	2022-06-10

图 12: 贷款列表

Bank Management System User Bank Client Account Department Employee Loan Business About Hasyuan Wang

List of bank\_account\_year

ICBC

Year	#Account
2022	3

合肥支行

Year	#Account
2021	1
2022	1

List of bank\_balance\_year

ICBC

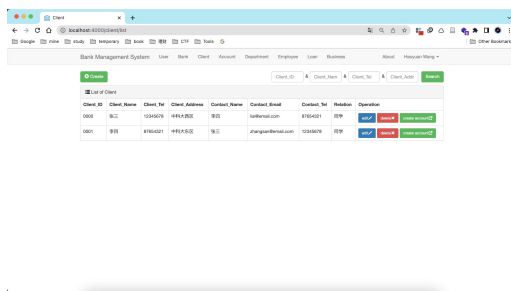
Year	Balance
2022	2000.0

合肥支行

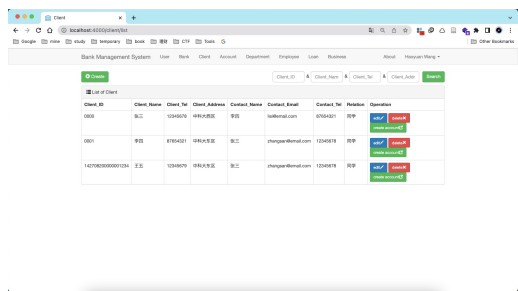
Year	Balance
2021	10000.0
2022	1000.0

图 13: 业务统计（因页面太长不太完整）

## 4.2 测试结果

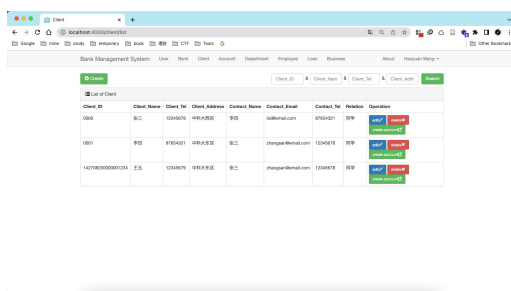


(a) 之前

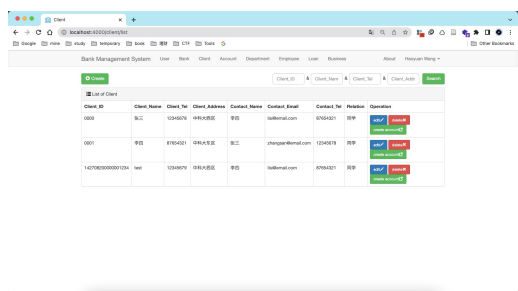


(b) 之后

图 14: 增加客户

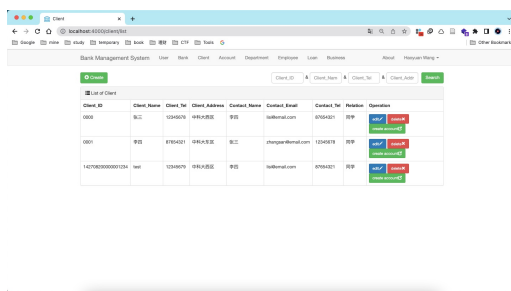


(a) 之前

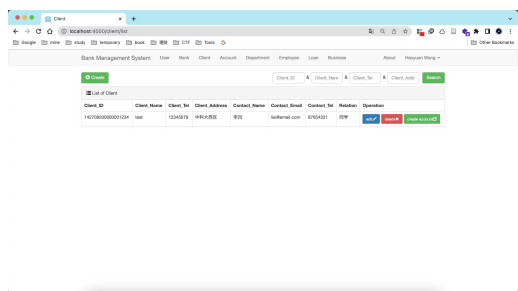


(b) 之后

图 15: 修改客户信息



(a) 之前



(b) 之后

图 16: 客户查询

**List of Open**

Client ID	Validated Date	Account ID	Operation
0000	2020-06-15	Test Account	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-15	Test business	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-05	Test Cheque Account	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-05	1000	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-15	Test AccountID	<a href="#">cancel</a> <a href="#">cancel</a>

**List of Account**

Account ID	Bank Name	Balance	Opening Date
1000	ICBC	1000.0	2020-06-05
Test Account	ICBC	1000.0	2020-06-05
Test AccountID	ICBC	1000.0	2020-06-10
Test business	ICBC	10000.0	2021-06-10
Test Cheque Account	ICBC	0.0	2020-06-05

(a) 之前

**List of Open**

Client ID	Validated Date	Account ID	Operation
0000	2020-06-15	Test Account	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-15	Test business	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-05	Test Cheque Account	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-05	1000	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-15	Test AccountID	<a href="#">cancel</a> <a href="#">cancel</a>
10710000000000000000	2020-06-15	Test Account for report	<a href="#">cancel</a> <a href="#">cancel</a>

**List of Account**

Account ID	Bank Name	Balance	Opening Date
1000	ICBC	1000.0	2020-06-05
Test Account	ICBC	1000.0	2020-06-05
Test Account for report	ICBC	10000.0	2020-06-17
Test AccountID	ICBC	1000.0	2020-06-10
Test business	ICBC	10000.0	2021-06-10

(b) 之后

图 17: 账户开户

**List of Account**

Account ID	Bank Name	Balance	Opening Date
1000	ICBC	1000.0	2020-06-05
Test Account	ICBC	1000.0	2020-06-05
Test Account for report	ICBC	10000.0	2020-06-17
Test AccountID	ICBC	1000.0	2020-06-10
Test business	ICBC	10000.0	2021-06-10

**List of Cheque Account**

Account ID	Overdraft
Test Account for report	1000.0
Test Cheque Account	1000.0

**List of Saving Account**

Account ID	Interest Rate	Currency Type
1000	0.00	人民币
Test Account	1.0	人民币
Test AccountID	0.00	人民币
Test business	0.00	人民币

(a) 之前

**List of Account**

Account ID	Bank Name	Balance	Opening Date
1000	ICBC	1000.0	2020-06-05
Test Account	ICBC	1000.0	2020-06-05
Test Account for report	ICBC	10000.0	2020-06-17
Test AccountID	ICBC	1000.0	2020-06-10
Test business	ICBC	10000.0	2021-06-10
Test Cheque Account	ICBC	0.0	2020-06-05

**List of Cheque Account**

Account ID	Overdraft
Test Account for report	1000.0
Test Cheque Account	1000.0
Test Cheque Account	1000.0

**List of Saving Account**

Account ID	Interest Rate	Currency Type
1000	0.00	人民币
Test Account	1.0	人民币
Test AccountID	0.00	人民币
Test business	0.00	人民币

(b) 之后

图 18: 修改账户信息

**List of Account**

Account ID	Bank Name	Balance	Opening Date
1000	ICBC	1000.0	2020-06-05
Test Account	ICBC	1000.0	2020-06-05
Test Account for report	ICBC	10000.0	2020-06-17
Test AccountID	ICBC	1000.0	2020-06-10
Test business	ICBC	10000.0	2021-06-10

**List of Cheque Account**

Account ID	Overdraft
Test Account for report	1000.0
Test Cheque Account	1000.0

**List of Saving Account**

Account ID	Interest Rate	Currency Type
1000	0.00	人民币
Test Account	1.0	人民币
Test AccountID	0.00	人民币
Test business	0.00	人民币

(a) 之前

**List of Open**

Client ID	Validated Date	Account ID	Operation
0000	2020-06-15	Test Account	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-15	Test business	<a href="#">cancel</a> <a href="#">cancel</a>
0000	2020-06-05	Test Cheque Account	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-05	1000	<a href="#">cancel</a> <a href="#">cancel</a>
0001	2020-06-15	Test AccountID	<a href="#">cancel</a> <a href="#">cancel</a>
10710000000000000000	2020-06-15	Test Account for report	<a href="#">cancel</a> <a href="#">cancel</a>

**List of Account**

Account ID	Bank Name	Balance	Opening Date
Test Account for report	ICBC	10000.0	2020-06-17

**List of Cheque Account**

Account ID	Overdraft
Test Account for report	1000.0

**List of Saving Account**

Account ID	Interest Rate	Currency Type
Test Account for report	1.0	人民币

(b) 之后

图 19: 查询账户





Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(a) 之前

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(b) 之后

图 23: 贷款查询

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(a) 未发放贷款

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(b) 发放部分贷款

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(c) 发放所有贷款

图 24: 贷款发放

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(a) 之前

Loan ID	Bank Name	Loan Amount	Loan Status	Pay already	Opening Date	Operation
1010	浦发银行	2000.0	Not issued	0.0	2020-06-10	Cancel
1000	浦发银行	2000.0	Being issued	1000.0	2020-06-10	Cancel
1005	浦发银行	2000.0	Issued	0.0	2020-06-10	Cancel

Client ID	Loan ID	Pay ID	Pay Amount	Pay Date
0000	1000	1000	1000.0	2020-06-06
0000	1000	1001	0.0	2020-06-07
0000	1000	1002	1000.0	2020-06-10
0000	Test Client02	1000	1000.0	2020-06-10
0000	Test Client02	1001	0.0	2020-06-10

(b) 之后

图 25: 删除贷款

### 4.3 实现中的难点问题及解决

- 不熟悉浏览器网页开发：主要使用 w3school 平台进行资料搜集和相关知识学习，同时向有相关经验的同学请教
- 前端不能正常渲染：经过检查和测试，会发现某些时候是自己布局写的有问题，而某些时候可能因为 cdn 服务不稳定导致（可以通过下载到本地解决）
- 在测试时功能异常：使用 Python 的 logging 包，在关键节点处记录 log，使用 log 进行 debug
- 总结的常见异常情况：
  - 前后端数据交互不正确（双向均有可能）

- MySQL 语句生成错误（例如字段忘记加单引号）
- 没有注意由于外键约束导致的特定语句执行顺序
- 实现对特殊字符（例如单引号）的处理：起初通过手动处理（将单引号替换为转义字符），后经过调研，发现 `pymysql` 包中有相关的处理函数，改为使用包中提供的函数进行处理

## 5 总结与讨论

- 本次实验通过使用 Python、MySQL、HTML、CSS、JavaScript 等诸多语言及工具，实现了一个简单的 B/S 模型的数据库系统，结合 lab2 进行数据的设计，体会到了数据库设计与实现的诸多原则，也在一定程度上领悟了背后的思想，进一步加深了我对数据库系统的理解，也更进一步地掌握了 MySQL 相关的技能。
- 有很多地方做得不够完善，有很大的进步空间。主要是很多约束做得还不够约束，在某些特定的条件下不会被满足。在这个方面我还需要做出更多的努力。
- 在以后和数据库相关的开发中，我会尽力去多开发数据库本身的效能，而非依靠后端开发来检验数据正确性、完整性。