

# “银行业务管理系统”

## 系统设计与实现报告

姓名：哈依纳尔

学号：PB19111715

计算机科学与技术学院

中国科学技术大学

2022 年 6 月

## 目 录

1	概述 .....	1
1.1	系统目标 .....	1
1.2	需求说明 .....	1
1.3	本报告的主要贡献 .....	2
2	总体设计 .....	2
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
3	详细设计【可选】 .....	6
3.1	**** 模块 .....	6
3.2	**** 模块 .....	7
3.3	**** 模块 .....	7
4	实现与测试 .....	11
4.1	实现结果 .....	11
4.2	测试结果 .....	错误!未定义书签。
4.3	实现中的难点问题及解决【可选】 .....	15
5	总结与讨论 .....	15

# 1 概述

## 1.1 系统目标

某银行准备开发一个银行业务管理系统。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。

## 1.2 需求说明

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

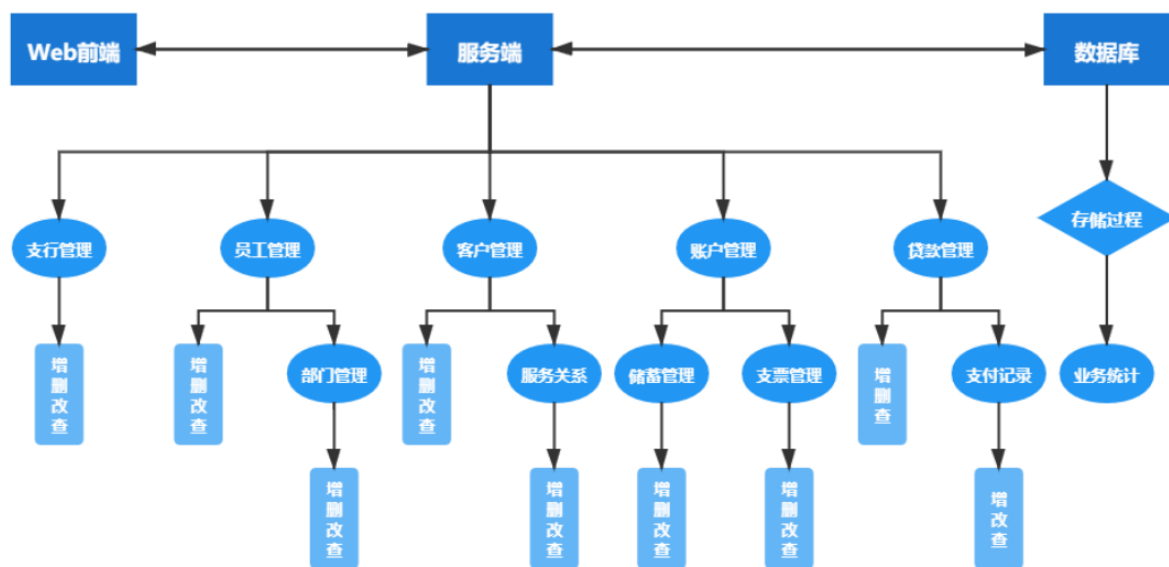
业务统计：按业务分类（储蓄、贷款）和时间（月、季、年）统计各个支行的业务总金额和用户数，要求对统计结果同时提供表格和曲线图两种可视化展示方式.

### 1.3 本报告的主要贡献

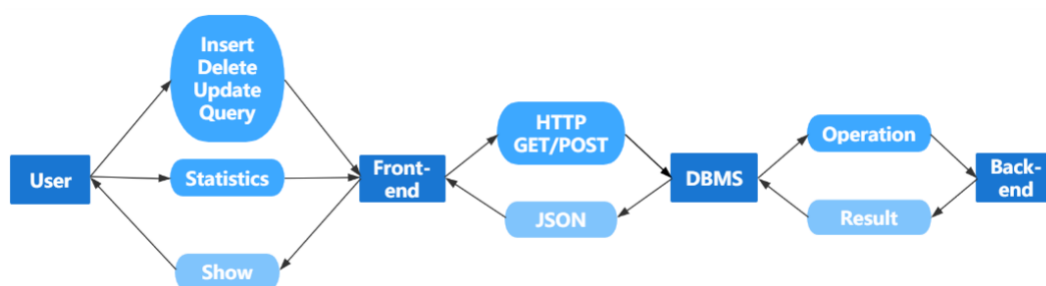
完成了网页设计，前端数据处理，后端数据库搭建，前后端连接。

## 2 总体设计

### 2.1 系统模块结构

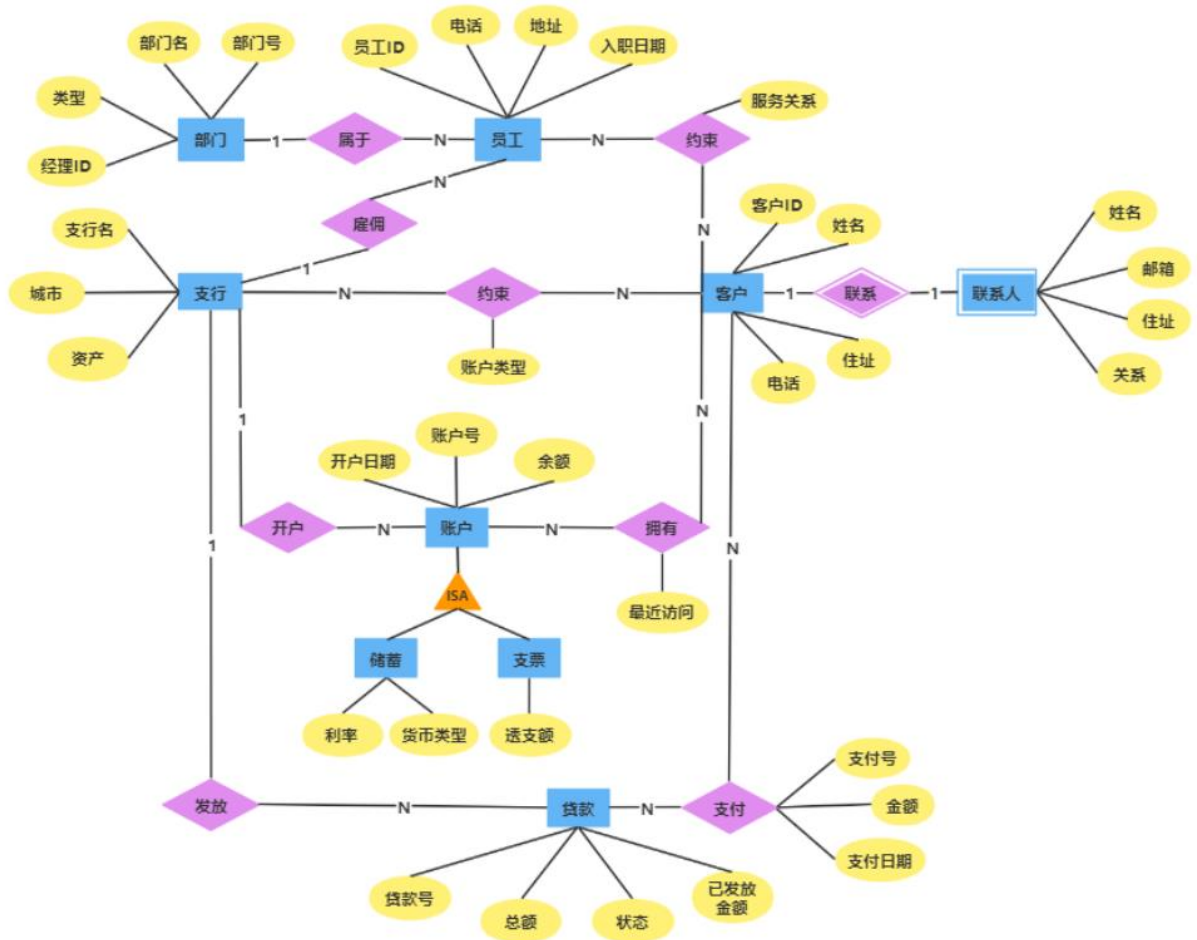


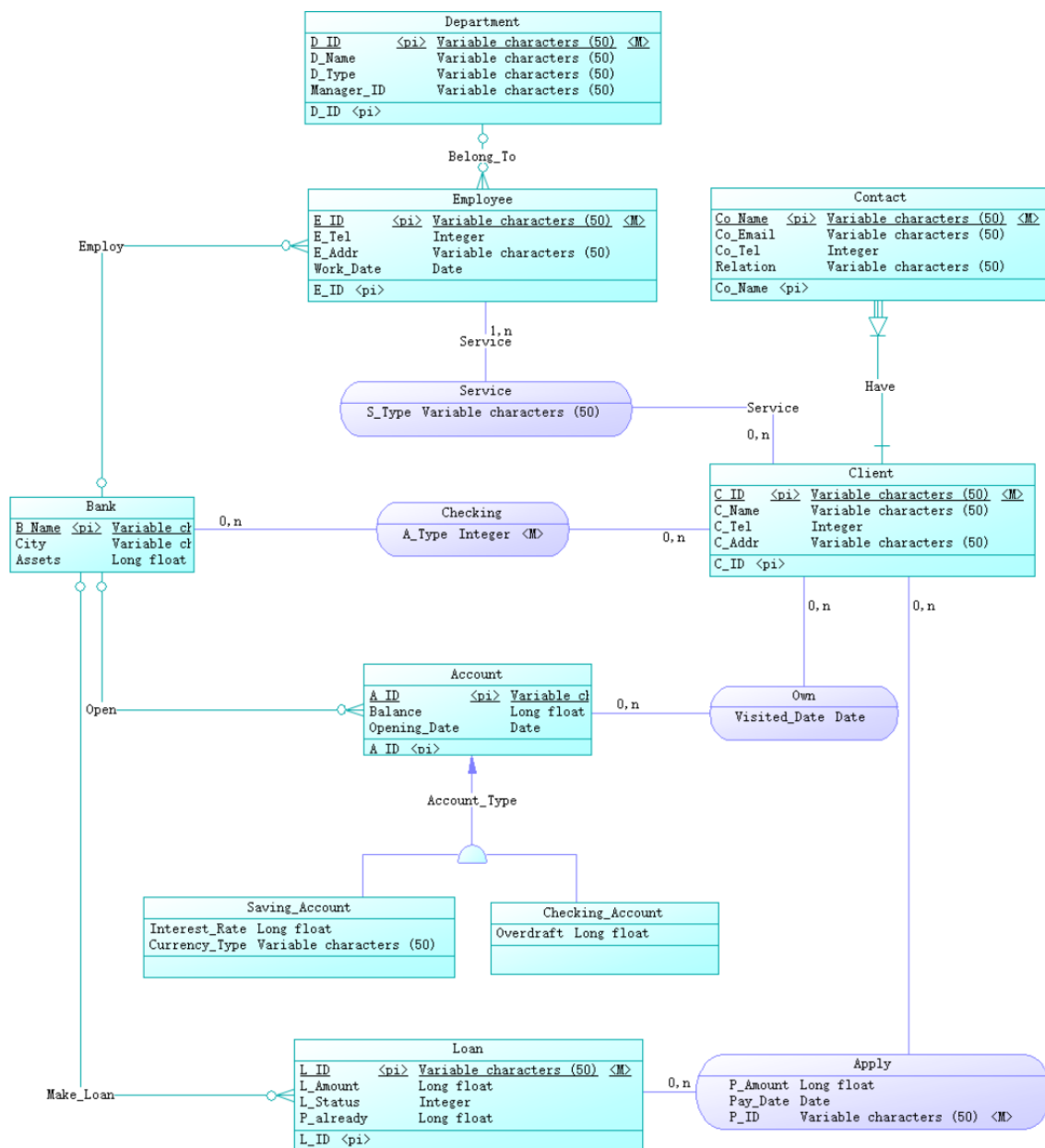
### 2.2 系统工作流程

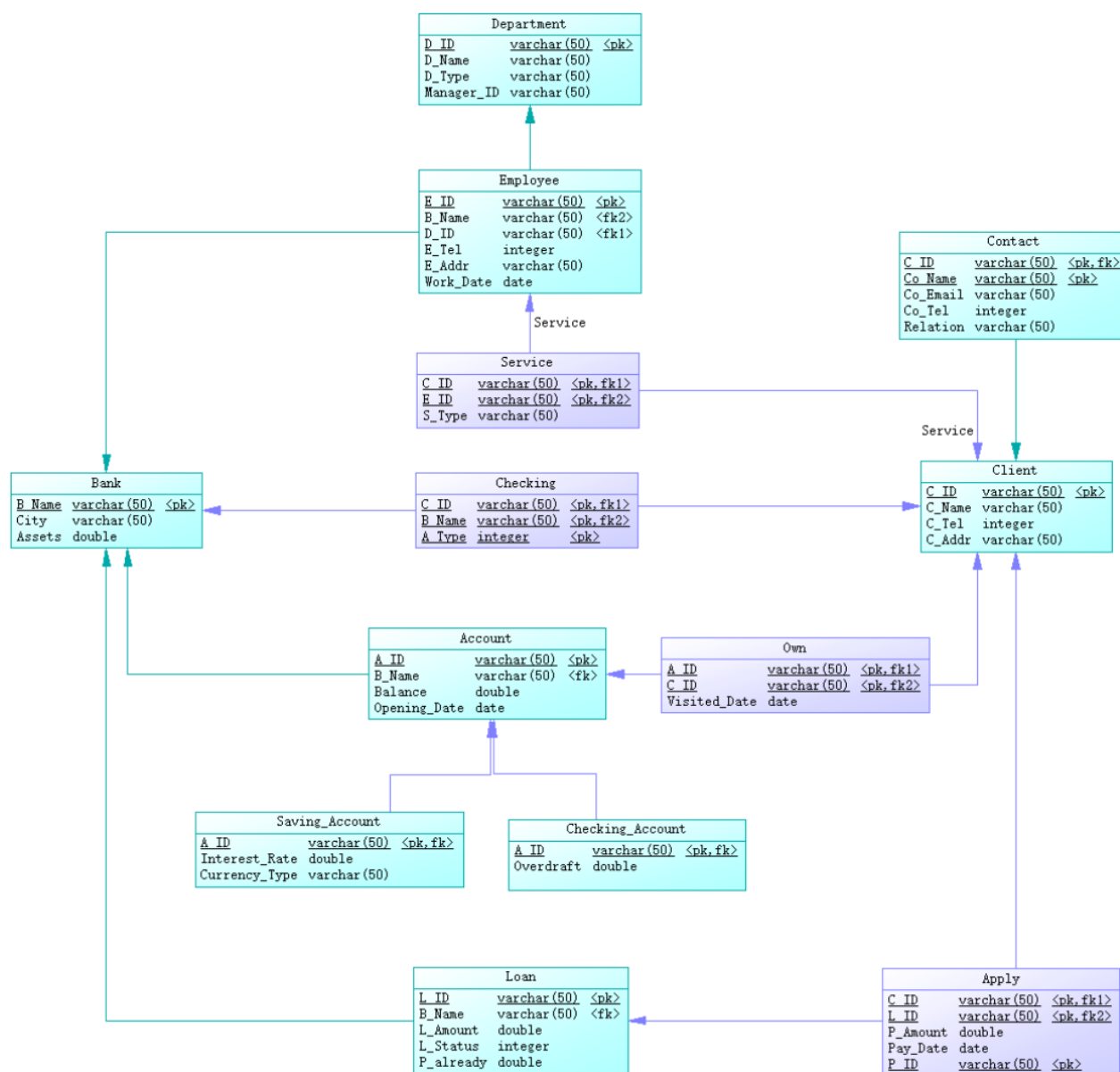


## 2.3 数据库设计

ER图:

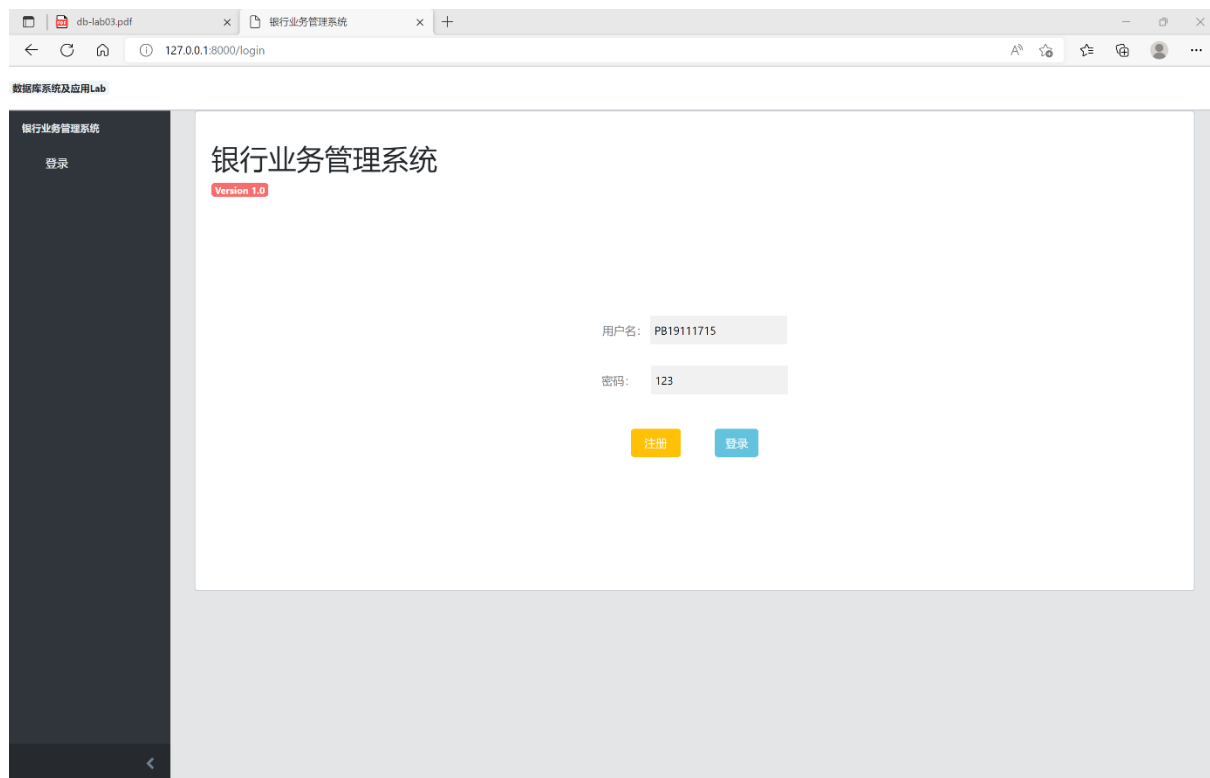






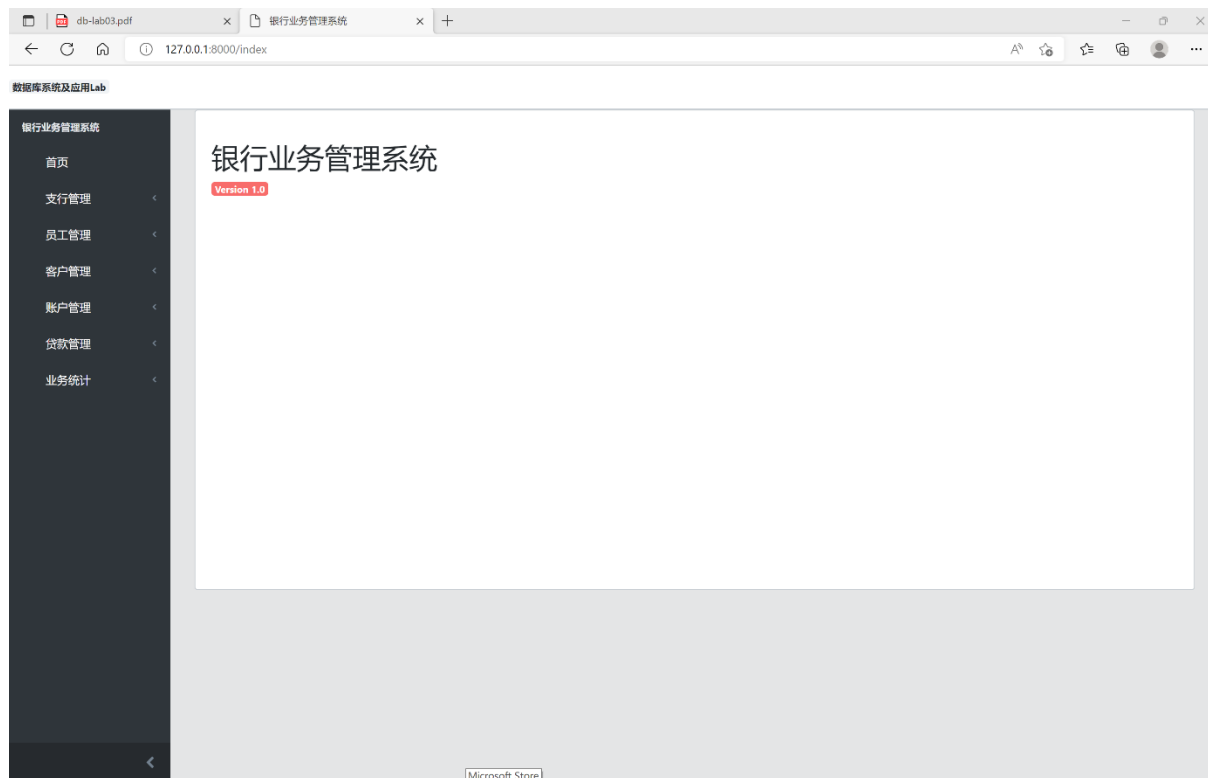
## 3 详细设计

### 3.1 登录模块

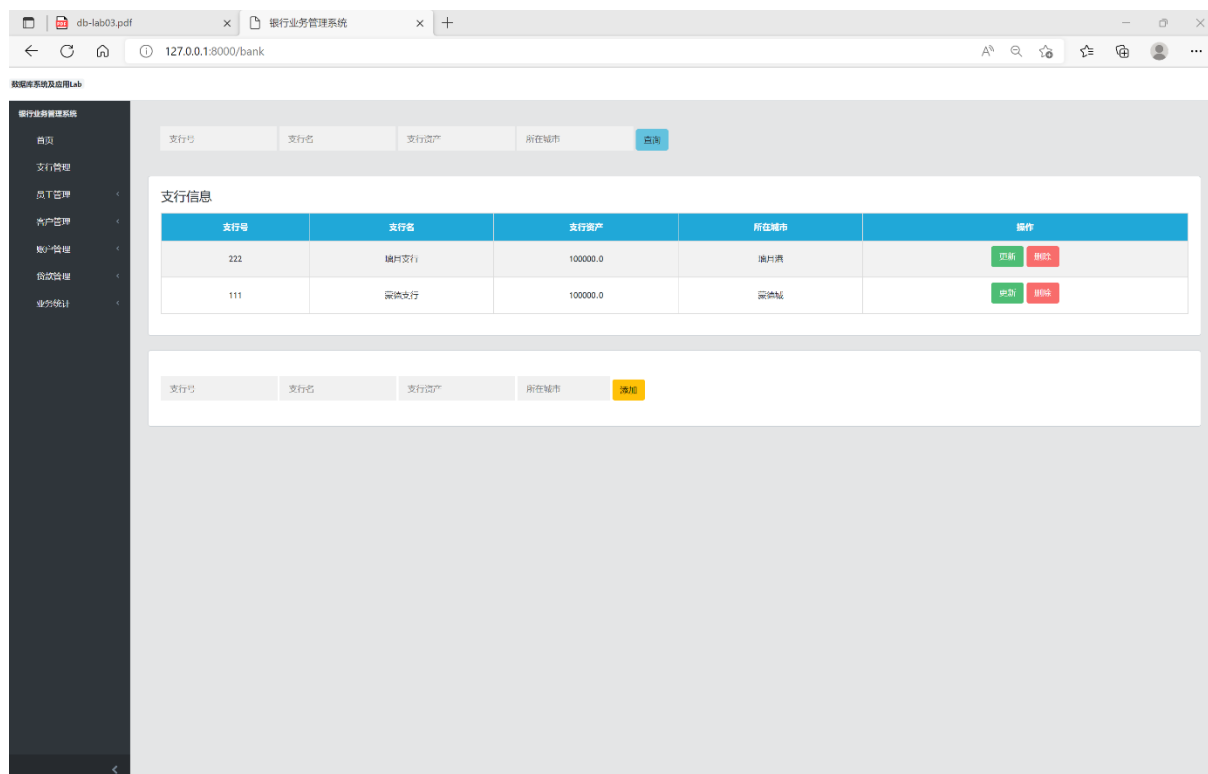




## 3.2 首页模块



## 3.3 支行模块



### 3.3 员工模块

db-lab03.pdf x 银行业务管理系统 x +

127.0.0.1:8000/employee

数据库系统及应用Lab

银行业务管理系统

- 首页
- 支行管理
- 员工管理
- 客户管理
- 账户管理
- 信贷管理
- 业务统计

员工ID 员工姓名 员工电话 员工住址 雇佣日期 所在支行 部门号 部门名称 部门类型 部门经理ID 查询

#### 员工管理

默认排序: 雇佣日期

员工ID	员工姓名	员工电话	员工住址	雇佣日期	所在支行	部门号	部门名称	部门类型	部门经理ID	操作
E001	孙楠	151515	翔月	2000-12-31	翔月支行	D001	行政部	行政	E001	更新 删除
E002	叶莉	141414	承德	2001-07-24	承德支行	D002	执行部	执行	E002	更新 删除

员工ID 员工姓名 员工电话 员工住址 雇佣日期 所在支行 部门号 添加

员工ID 员工姓名 员工电话 员工住址 雇佣日期 所在支行 部门号 部门名称 部门类型 部门经理ID 查询

#### 部门管理

部门号	部门名称	部门类型	部门经理ID	操作
D001	行政部	行政	E001	更新 删除
D002	执行部	执行	E002	更新 删除

部门号 部门名称 部门类型 部门经理ID 添加

### 3.3 客户模块

db-lab03.pdf x 银行业务管理系统 x +

127.0.0.1:8000/client

数据库系统及应用Lab

银行业务管理系统

- 首页
- 支行管理
- 员工管理
- 客户管理
- 账户管理
- 信贷管理
- 业务统计

客户ID 客户姓名 客户电话 客户地址 联系人姓名 联系人电话 联系人邮箱 关系 查询

#### 客户管理

客户ID	客户姓名	客户电话	客户地址	联系人姓名	联系人电话	联系人邮箱	关系	操作
C001	张	111	包河区	张家福	112	111@qq.com	家人	更新 删除
C001	连斗东	131313	安徽	郭亚	121212	1212@qq.com	兄弟	更新 删除
C002	李	123123	贵州	熊梦	123	321@qq.com	朋友	更新 删除
C003	123	321	3	3	3	3	3	更新 删除

客户ID 客户姓名 客户电话 客户地址 联系人姓名 联系人电话 联系人邮箱 关系 添加

客户ID 员工ID 部门类型 查询

#### 服务关系

客户ID	员工ID	服务类型	操作
C001	E002	朋友	更新 删除

客户ID 员工ID 部门类型 添加

### 3.3 账户模块

db-lab03.pdf x 银行业务管理系统 x +

127.0.0.1:8000/account

数据库系统及应用Lab

银行业务管理系统

- 首页
- 支行管理
- 员工管理
- 客户管理
- 账户管理
- 贷款管理
- 业务统计

账户号 客户ID 客户姓名 开户支行 开户时间 账户余额 最近访问时间 利率 货币类型 查询

#### 储蓄账户管理

默认排序: 升序, 降序

账户号	客户ID	客户姓名	开户支行	开户时间	账户余额	最近访问时间	利率	货币类型	操作
A102	C001	趙小虎	福州支行	2002-10-10	2000.0	2022-06-19	0.1	固定	更新 删除
A105	C003	123	福州支行	2003-10-10	300.0	2022-06-19	0.12	浮动	更新 删除
A106	C003	123	宁德支行	2003-12-12	400.0	2022-06-19	0.13	固定	更新 删除

账户号 客户ID 客户姓名 开户支行 开户时间 账户余额 最近访问时间 透支额度 查询

#### 支票账户管理

默认排序: 升序, 降序

账户号	客户ID	客户姓名	开户支行	开户时间	账户余额	最近访问时间	透支额度	操作
A103	C002	李	福州支行	2002-12-12	1500.0	2022-06-19	3000.0	更新 删除

#### 储蓄账户

账户号 客户ID 开户支行 开户时间 账户余额 利率 货币类型 添加

#### 支票账户

账户号 客户ID 开户支行 开户时间 账户余额 透支额度 添加

### 3.3 贷款模块

db-lab03.pdf x 银行业务管理系统 x +

127.0.0.1:8000/debt

数据库系统及应用Lab

银行业务管理系统

- 首页
- 支行管理
- 员工管理
- 客户管理
- 账户管理
- 贷款管理
- 业务统计

贷款号 发放支行 贷款金额 贷款状态 查询

#### 贷款管理

默认排序: 升序, 降序

贷款状态: 0 - 未发放; 1 - 发放中; 2 - 发放完毕; 3 - 过期

贷款号	发放支行	贷款金额	贷款状态	操作
001	福州支行	1000.0	1	删除
002	宁德支行	1500.0	2	删除
003	宁德支行	100.0	0	删除

贷款号 发放支行 贷款金额 添加

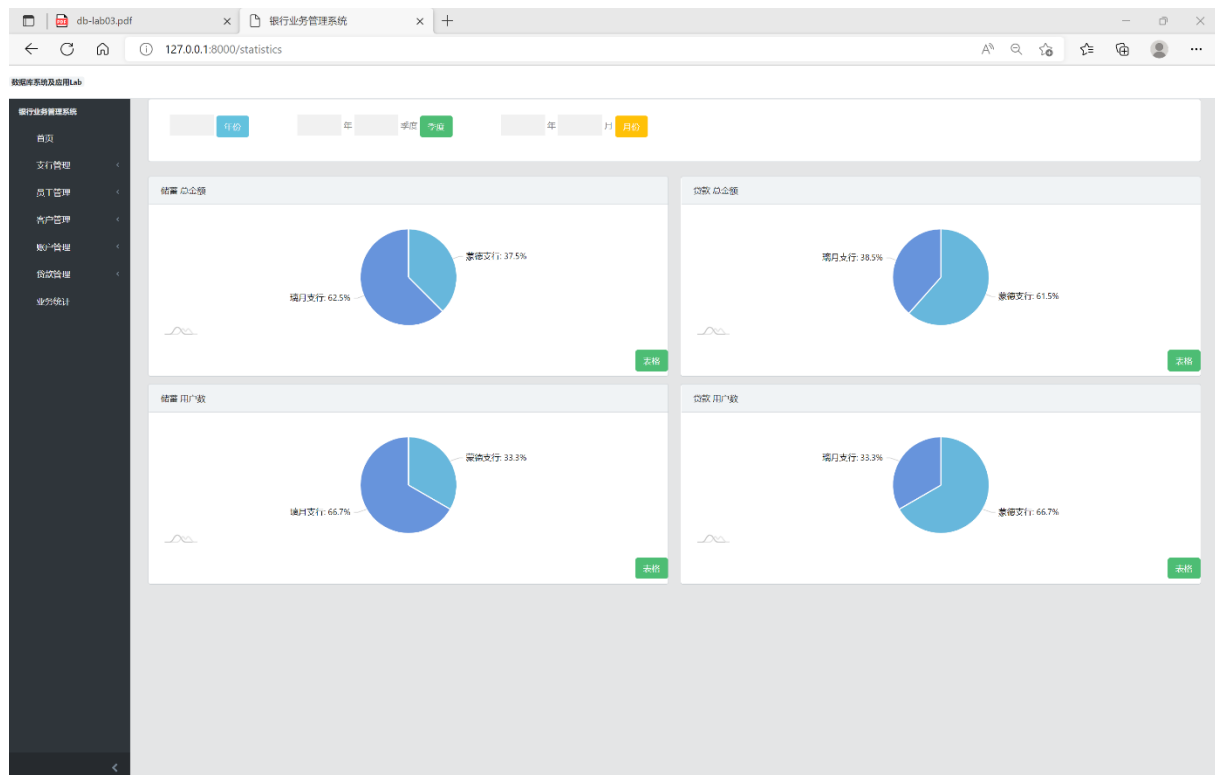
贷款号 客户ID 支付号 支付金额 支付日期 查询

#### 支付记录

支付号	贷款号	客户ID	支付金额	支付日期	操作
001	001	001	200.0	2005-10-10	更新
002	002	C001	1500.0	2003-01-01	更新

待发放贷款号 发放客户ID 支付号 发放日期 发放金额 添加

### 3.3 业务统计模块



## 4 结果

### 4.1 数据库实现

account x checking bank checking\_account

1 • SELECT \* FROM bank.account;

Result Grid Filter Rows: Edit

	A_ID	B_Name	Balance	Opening_Date
▶	A101	蒙德支行	60000	2002-01-01
	A102	璃月支行	2000	2002-10-10
	A103	璃月支行	1500	2002-12-12
	A105	璃月支行	300	2003-10-10
	A106	蒙德支行	400	2003-12-12
*	NULL	NULL	NULL	NULL

bank x checking\_account client

1 • SELECT \* FROM bank.bank;

Result Grid Filter Rows: Edit

	B_ID	B_Name	City	Assets
▶	222	璃月支行	璃月港	100000
	111	蒙德支行	蒙德城	100000
*	NULL	NULL	NULL	NULL

checking x bank checking\_a

1 • SELECT \* FROM bank.c

Result Grid Filter Rows: Edit

	C_ID	B_Name	A_Type
▶	001	璃月支行	1
	C001	璃月支行	1
	C002	璃月支行	1
	C003	璃月支行	1
	C001	蒙德支行	1
	C002	蒙德支行	1
	C003	蒙德支行	1
*	NULL	NULL	NULL

checking\_account x client contact department

1 • SELECT \* FROM bank.checking\_account;

Result Grid Filter Rows: Edit

	A_ID	Overdraft
▶	A103	3000
*	NULL	NULL

client

contact

department

employee

1

SELECT \* FROM bank.client;

Result Grid

Filter Rows:

Edit

	C_ID	C_Name	C_Tel	C_Addr
▶	001	张	111	包河区
	C001	迪卢克	131313	蒙德
	C002	琴	123123	璃月
	C003	123	321	3
*	NULL	NULL	NULL	NULL

contact

department

employee

loan

own

1

SELECT \* FROM bank.contact;

Result Grid

Filter Rows:

Edit

	C_ID	Co_Name	Co_Email	Co_Tel	Relation
▶	001	张家属	111@qq.com	112	家人
	C001	凯亚	1212@qq.com	121212	兄弟
	C002	派梦	321@qq.com	123	朋友
	C003	3	3	3	3
*	NULL	NULL	NULL	NULL	NULL

department

employee

loan

own

1

SELECT \* FROM bank.department;

Result Grid

Filter Rows:

Edit

	D_ID	D_Name	D_Type	Manager_ID
▶	D001	行政部	行政	E001
	D002	执行部	执行	E002
*	NULL	NULL	NULL	NULL

employee

loan

own

saving\_account

user

apply

1

SELECT \* FROM bank.employee;

Result Grid

Filter Rows:

Edit

Export/In

	E_ID	E_Name	B_Name	D_ID	E_Tel	E_Addr	Work_Date
▶	E001	钟离	璃月支行	D001	151515	璃月	2000-12-31
	E002	可莉	蒙德支行	D002	141414	蒙德	2001-07-24
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

loan | **own** | saving\_account | user | apply

1 • SELECT \* FROM bank.loan;

Result Grid | Filter Rows: | Edit:

	L_ID	B_Name	L_Amount	L_Status	P_already
▶	001	瑞月支行	1000	1	200
	002	蒙德支行	1500	2	1500
	003	蒙德支行	100	0	0
*	NULL	NULL	NULL	NULL	NULL

service |

1 • SELECT \* FROM bank.service;

Result Grid | Filter Rows: |

	C_ID	E_ID	S_Type
▶	C001	E002	朋友
*	NULL	NULL	NULL

own | **saving\_account** | user

1 • SELECT \* FROM bank.own;

Result Grid | Filter Rows: |

	C_ID	Visited_Date	A_ID
▶	C001	2022-06-19	A102
	C002	2022-06-19	A101
	C002	2022-06-19	A103
	C003	2022-06-19	A105
	C003	2022-06-19	A106
*	NULL	NULL	NULL

saving\_account | user | **apply** | ser

1 • SELECT \* FROM bank.saving\_acc

Result Grid | Filter Rows: |

	A_ID	Interest_Rate	Currency_Type
▶	A102	0.1	摩拉
	A105	0.12	摩拉
	A106	0.13	摩拉
*	NULL	NULL	NULL

user x apply service

1 • SELECT \* FROM bank.

Result Grid

	username	userkey
▶	11	11
	123	123
	PB19111715	123
*	NULL	NULL

apply x service

1 • SELECT \* FROM bank.apply;

Result Grid

	C_ID	L_ID	P_ID	P_Amount	Pay_Date
▶	001	001	001	200	2005-10-10
	C001	002	002	1500	2003-01-01
*	NULL	NULL	NULL	NULL	NULL

MySQL Workbench

MySQLproject x

File Edit View Query Database Server Tools

Navigator

SCHEMAS

Filter objects

bank

- Tables
  - account
  - apply
  - bank
  - checking
  - checking\_account
  - client
  - contact
  - department
  - employee
  - loan
  - own
  - saving\_account
  - service
  - user
- Views
- Stored Procedures
- Functions

db

labone

sakila

sys

world

Administration Schemas

Information

Table: checking

Columns:

C\_ID varchar(50) PK

B\_Name varchar(50) PK

A\_Type int PK



## 4.2 实现中的难点问题及解决

1. 没学过 web 开发，完全不会网页设计，从头学起花费了大量时间
2. 不会 python 数据库设计，从头学起花费了大量时间
3. 不会 python 连接前后端，从头学起花费了大量时间
4. 不会 python 设计语言，从头学起花费了大量时间
5. 设计的 Mysql 代码在使用中发现很多不足，修改花费了大量时间

## 5 总结与讨论

1. 实验收获：学会了 python 设计简单数据库，连接前后端，学会了简单的网页设计，学会了数据前后端流通。
2. 实验难度：非常难
3. 实验任务量：非常多