



**本科毕业论文**

|  |
| --- |
| **基于Django与bootstrap个人收支管理系统** |
|  |

**龙燕**

**201430340117**

|  |  |
| --- | --- |
| 指导教师 | **杨章明 高级工程师**  **林旭东 副教授** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学院名称 |  | **软件学院** | 专业名称 |  | **软件工程** |
| 论文提交日期 |  | 2018年4月10日 | 论文答辩日期 |  | 2018年4月21日 |

# **摘 要**

近年来，人类社会科学技术不断发展，计算机科学越来越成熟，其强大的功能深深扎根于人们的内心。计算机技术已经运用到人类生活的方方面面，并且一直不断的发展，成为推动时代发展的工具。通过计算机这种途径来管理个人财务信息，可以实现系统化、规范化、科学化地个人收支管理，这种方式是传统手工操作记录无法实现的。为了适应现在社会的快节奏生活，满足人们日常方便快捷的记账管理，社会上出现了一些在线财务管理工具。个人收支管理系统不仅要能满足用户日常记账的基本操作，还有提供基本的财务分析功能，同时要确保用户信息的安全性。

本文主要分析了个人收支管理系统的实现。该系统使用浏览器/服务器模式（B/S）进行开发，用python和pycharm作为开发平台及工具，采用了Django+boostrap框架， 用来存储数据的数据库是mysql,是主要采用Javascript技术来实现前端页面的静态与动态显示，使用Django框架的视图（views）、模型（templates）、模板（models）这种模式分离处理业务逻辑、接口、数据，处理以便更好地控制业务请求和处理业务请求。

本系统使用邮箱验证的方式来验证注册用户的信息，注册邮箱是用户的唯一标识，一个邮箱不能重复注册，注册的时候系统通过发验证码的形式验证邮箱，假定用户不记得自己的密码，那么就可以通过系统的找回密码功能发送邮件找回。该系统主要由用户登录模块、个人信息模块、记账管理模块、分类设置模块和统计分析模块这五大模块构成。其中户登录模块主要包括用户注册、登录、忘记密码、修改密码还有系统退出功能。个人信息模块主要的功能是用户查看、修改、保存个人信息。记账管理模块的功能是实现个人收支信息的管理，用户可以添加、修改、删除和查询记账信息。分类设置模块主要实现收支类型的设置。统计分析模块主要实现用户收支数据的分析，包括收支分析和对账表。

关键词：个人收支管理 Django boostrap MVC

**The Analysis and Design of Flight Ticket Booking System**

Long Yan

(College of Software Engineering, South China Agricultural University, Guangzhou 510642, China)

**Abstract:** In recent years, human social science and technology have been continuously developed and computer science has become more mature. Its powerful functions are deeply rooted in people's hearts. Computer technology has been applied to all aspects of human life, and has continued to develop, becoming a tool to promote the development of the times. By managing personal financial information through a computer, you can achieve systematic, standardized, and scientific personal income and expenditure management. This method cannot be achieved by traditional manual operation records. In order to adapt to the fast-paced life of today's society and satisfy people's daily and convenient bookkeeping management, there have been some online financial management tools in the society. The personal income and expenditure management system must not only meet the basic operations of the user's daily bookkeeping, but also provide basic financial analysis functions, and at the same time ensure the security of user information.

This article mainly analyzes the realization of personal income and expenditure management system. The system uses browser/server mode (B/S) for development. It uses python and pycharm as development platforms and tools. It uses the Django+boostrap framework. The database used to store data is mysql. It uses Javascript technology to implement the front end. Static and dynamic display of pages, using Django framework views (templates), templates (templates), and models (models) to separate processing business logic, interfaces, data, and processing to better control business requests and process business requests.

This system uses the mailbox verification method to verify the registered user's information. The registered mailbox is the user's unique identifier. A mailbox cannot be re-registered. When the system is registered, the system verifies the mailbox by sending a verification code. Assume that the user does not remember his password. You can use the system's password recovery function to send messages back. The system is mainly composed of five modules: user login module, personal information module, accounting management module, classification setting module, and statistical analysis module. The user login module mainly includes user registration, login, forgotten password, password change, and system exit function. The main function of the personal information module is the user to view, modify, and save personal information. The function of the billing management module is to manage personal income and expenditure information. Users can add, modify, delete, and query billing information. The classification setting module mainly realizes setting of income and expenditure types. The statistical analysis module mainly implements the analysis of user revenue and expenditure data, including income and expenditure analysis and reconciliation tables.

**Key words:** Personal income and expenditure management Django Boostrap MVC

目 录

[1 引言 1](#_Toc510791115)

[1.1 选题背景 1](#_Toc510791116)

[1.2 开发本系统的意义 1](#_Toc510791117)

[1.3 本系统的设计思路 1](#_Toc510791118)

[1.4 本系统开发采用的研究方法 1](#_Toc510791119)

[2 需求分析 3](#_Toc510791120)

[2.1 系统架构 3](#_Toc510791121)

[2.2 功能模块用例分析 3](#_Toc510791122)

[2.2.1 注册用例 4](#_Toc510791123)

[2.2.2 登录用例 4](#_Toc510791124)

[2.2.3 添加分类用例 5](#_Toc510791125)

[2.2.4 修改分类用例 5](#_Toc510791126)

[2.2.5 删除分类用例 6](#_Toc510791127)

[2.2.6 记账用例 6](#_Toc510791128)

[2.2.7 查询账单用例 7](#_Toc510791129)

[2.2.8 修改账单用例 7](#_Toc510791130)

[2.2.9 删除账单用例 8](#_Toc510791131)

[2.2.10 查看收支分析用例 8](#_Toc510791132)

[2.2.11 查看对账表用例 9](#_Toc510791133)

[2.2.12 找回密码用例 9](#_Toc510791134)

[2.2.13 修改密码用例 10](#_Toc510791135)

[2.2.14 修改个人信息用例 10](#_Toc510791136)

[3 数据库设计 11](#_Toc510791137)

[3.1 实体E-R图设计 11](#_Toc510791138)

[3.1.1 用户信息实体E-R图 11](#_Toc510791139)

[3.1.2 记账分类实体E-R图 11](#_Toc510791140)

[3.1.3 账单信息实体E-R图 12](#_Toc510791141)

[3.1.4 验证码实体E-R图 12](#_Toc510791142)

[3.2 数据库表设计 13](#_Toc510791143)

[3.2.1 用户信息表 13](#_Toc510791144)

[3.2.2 记账分类信息表 14](#_Toc510791145)

[3.2.3 账单信息表 14](#_Toc510791146)

[3.2.4 验证码信息表 15](#_Toc510791147)

[4 系统设计 16](#_Toc510791148)

[4.1 系统目标 16](#_Toc510791149)

[4.2 系统的功能 16](#_Toc510791150)

[4.3 系统开发环境 16](#_Toc510791151)

[4.3.1 Python简介 16](#_Toc510791152)

[4.3.2 PyCharm简介 17](#_Toc510791153)

[4.3.3 MySQL简介 17](#_Toc510791154)

[4.3.4 Django简介 17](#_Toc510791155)

[4.4 系统的设计 18](#_Toc510791156)

[4.4.1 总体流程设计 18](#_Toc510791157)

[4.4.2 注册流程设计 20](#_Toc510791158)

[4.4.3 登录流程设计 21](#_Toc510791159)

[4.4.4 记账流程设计 21](#_Toc510791160)

[4.4.5 总体模块的设计 22](#_Toc510791161)

[4.5 前端界面模块主要设计与实现 22](#_Toc510791162)

[4.5.1 表单设计 22](#_Toc510791163)

[4.5.2 表单校验 22](#_Toc510791164)

[4.5.3 日期选择 24](#_Toc510791165)

[4.5.4 图表绘制 24](#_Toc510791166)

[4.5.5 网页动态刷新 25](#_Toc510791167)

[4.6 后台逻辑模块主要设计与实现 25](#_Toc510791168)

[4.6.1 处理请求 25](#_Toc510791169)

[4.6.2 发送邮箱验证信息 26](#_Toc510791170)

[4.6.3 分页显示 27](#_Toc510791171)

[4.6.4 拦截器 27](#_Toc510791172)

[4.6.5 数据库交互 28](#_Toc510791173)

[5 系统运行效果及功能说明 29](#_Toc510791174)

[5.1 注册 29](#_Toc510791175)

[5.2 登录 29](#_Toc510791176)

[5.3 找回密码 30](#_Toc510791177)

[5.4 修改密码 30](#_Toc510791178)

[5.5 修改个人资料 31](#_Toc510791179)

[5.6 分类设置 31](#_Toc510791180)

[5.7 记账 32](#_Toc510791181)

[5.8 账单管理 32](#_Toc510791182)

[5.9 收支分析 33](#_Toc510791183)

[5.10 对账表 34](#_Toc510791184)

[6 结论与展望 35](#_Toc510791185)

[参 考 文 献 36](#_Toc510791186)

[致 谢 37](#_Toc510791187)

引言

选题背景

在日常生活中，很多人经常会有一个困惑：为什么钱花得这么快，钱到底用在哪里了？那是因为很多人日常都没有记账，或者说曾经尝试过每天记录个人财务行为，但最后坚持不了。要想每天坚持记账不是一件容易的事，所有要是有一个还很简单、很便捷的方式记账，那么这件事将变得相对容易多了。然而传统的记账方式是通过纸张来记录，记录起来比较麻烦、而且数据容易丢失，统计和查看都需要花费大量的时间和精力(孙智勇，2015)。近年来，社会高速发展，人类进入信息时代，计算机成为日常生活中不可缺少的工具。使用计算机可以帮助我们提高效率、节省时间成本。为了方便、快捷的管理个人财务，一种现代化的个人财务管理系统对我们来说是必需的(陈颂丽，李海良，2010)。本文的目的就是研究这样的一个系统的设计和实现。

对于财务系统，现在生活中使用较多的是企业的财务管理系统，个人的财务管理系统比较少。据了解，现在网上的那些个人财务管理系统相对来说功能过于繁杂，有些还充斥着大量的广告。这或许能满足部分人的个人财务管理需求，但对于大部分人来说体验不好。本文研究的个人收支管理系统是面向普通的个人用户，能满足用户日常的基本记账管理功能，包括记账、查询数据、修改记录、统计分析等功能，同时，系统界面美观、操作简单、数据安全。

开发本系统的意义

为用户提供了快捷简单的财务管理方式，节约了用户的时间。

本系统的设计思路

（1）通过参考网上已有的个人财务管理系统，同时通过向周围的同学、朋友调查，了解用户对个人财务管理系统的基本要求，对毕业设计的题目进行需求分析。

（2）根据需求分析文档画出业务的流程图，确定相应的实体，画出E-R图。

（3）根据E-R图确定关系模式，撰写概要设计文档。

（4）设计数据库，在MySQL中创建数据库。

（5）使用python和pycharm开发平台实现个人收支管理系统的功能。

## 本系统开发采用的研究方法

该系统将会是基于B/S模式进行开发。开发过程中将会使用python语言,采用Django,mysql,bootstrap,Javascript等技术。

（1）整体设计采用Django+boostrap框架，结合运用Ajax和JQuery等技术实现。

（2）使用mysql数据库存储数据而不是Django内置的SQLite，提高性能。

（3）通过使用Ajax技术来改善表单验证方式，不再需要打开新页面，也不再需要将整个页面数据提交，减少用户等待时间。能实现异步与服务器进行交互，在交互过程中用户仍可继续操作(Jeremy Keith，2011)。

（4）调试服务器采用Django框架内置的服务器，不需要设置，修改代码保存即可刷新。

（5）设计过程中采用面向对象的设计思想，实现数据的封装整合。

（6）采用mysql数据库管理系统。逐步用上存储过程，使运行效率更高点。

需求分析

系统架构

系统主要包括“注册”、“登录”、“分类设置”、“记账”、“账单管理”、“统计分析”、“密码管理”和“个人信息管理”共8个模块，其中分类设置包括了添加、修改以及删除分类功能，账单管理包括了查询、修改和删除账单功能，统计分析模块包括了“收支分析”和“对账表”两个功能，密码管理模块包括了“忘记密码”和“修改密码”两个功能。用户只有注册和登录系统才能使用系统的所有功能。系统的总体架构如图1所示。

图1 系统总体架构图

功能模块用例分析

表1 个人收支管理系统功能表

|  |  |  |
| --- | --- | --- |
| 子模块 | 子功能 | 编号 |
| 注册 | 注册 | F1.1.1 |
| 登录 | 登录 | F1.2.1 |
| 分类设置 | 添加分类 | F1.3.1 |
| 修改分类 | F1.3.2 |
| 删除分类 | F1.3.3 |
| 记账 | 记账 | F1.4.1 |
| 账单管理 | 查询账单 | F1.5.1 |
| 修改账单 | F1.5.2 |
| 删除账单 | F1.5.3 |
| 统计分析 | 查看收支分析 | F1.6.1 |
| 查看对账表 | F1.6.2 |
| 密码管理 | 忘记密码 | F1.7.1 |
| 修改密码 | F1.7.2 |
| 个人信息管理 | 修改个人信息 | F1.8.1 |

注册用例

注册：用户通过系统的注册页面进行注册才能系统该系统。

用例ID：F1.1.1

用例名称：注册

参与者：用户

用例描述：用户注册该系统

前置任务：无

基本事件流： a、 进入登录页面，点击“注册账号”

b、 填写相关信息

c、 发送验证码邮件到用户所填写的邮箱

d、填写验证码

e、 点击注册

f、显示注册结果

g、 用例结束

说明：邮箱不能重复注册，否则会显示错误信息

登录用例

登录：验证用户身份，用户登录后才能使用该系统功能。

用例ID：F1.2.1

用例名称：登录

参与者：用户

用例描述：用户通过系统的登录界面填写账号密码登录

前置任务：必须先成功注册

基本事件流： a、 打开系统登录页面

b、 输入账号密码并提交

c、 显示登录结果

d、 用例结束

说明：账号和密码都正确才能登录

添加分类用例

添加分类：用户可以增加收入、支出的多种不同类型。

用例ID：F1.3.1

用例名称：添加分类

参与者：用户

用例描述：用户添加不同的收入支出分类

前置任务：成功登录系统

基本事件流： a、 点击“设置”下拉菜单

b、 选择“分类设置”进入分类设置界面

c、选择需要添加的分类

d、 填写分类名称

e、 点击“添加”按钮

f、显示添加分类结果

g、用例结束

修改分类用例

修改分类：用户可以修改收入、支出的类型。

用例ID：F1.3.2

用例名称：修改分类

参与者：用户

用例描述：用户修改收入、支出的分类

前置任务：成功登录系统

基本事件流： a、 点击“设置”下拉菜单

b、 选择“分类设置”进入分类设置界面

c、点击需要修改的分类后面的编辑图标

d、 填写修改的分类名称

e、 点击“修改”按钮

f、 显示修改分类结果

g、用例结束

删除分类用例

删除分类：用户可以删除收入、支出的类型。

用例ID：F1.3.3

用例名称：删除分类

参与者：用户

用例描述：用户删除收入、支出的分类

前置任务：成功登录系统

基本事件流： a、 点击“设置”下拉菜单

b、 选择“分类设置”进入分类设置界面

c、点击需要删除的分类后面的删除图标

d、 点击“确定”按钮

e、 显示删除分类结果

f、 用例结束

记账用例

记账：用户记录收入、支出信息。

用例ID：F1.4.1

用例名称：记账

参与者：用户

用例描述：用户记录日常收入支出

前置任务：成功登录系统

基本事件流： a、 点击“记账”进入记账界面

b、 选择记账分类

c、 填写相关信息

d、 点击“保存”按钮

e、 显示记账结果

f、 用例结束

查询账单用例

查询账单：用户多条件查询记账记录。

用例ID：F1.5.1

用例名称：查询账单

参与者：用户

用例描述：用户可以通过输入多种条件来查询记账记录

前置任务：成功登录系统

基本事件流： a、 进入主页界面的记账记录部分

b、 输入查询条件的相关信息

c、 点击“搜账”按钮

d、 列表显示对应的记账记录

e、 用例结束

修改账单用例

修改账单：用户可以修改记账记录。

用例ID：F1.5.2

用例名称：修改账单

参与者：用户

用例描述：用户修改记账记录的信息

前置任务：成功登录系统

基本事件流： a、 进入主页界面的记账记录部分

b、点击需要修改的记录后面的编辑图标

c、 填写相关信息

d、 点击“修改”按钮

e、显示修改记录的结果

f、 用例结束

删除账单用例

修改账单：用户可以删除记账记录。

用例ID：F1.5.3

用例名称：删除账单

参与者：用户

用例描述：用户删除记账记录

前置任务：成功登录系统

基本事件流： a、 进入主页界面的记账记录部分

b、点击需要删除的记录后面的删除图标

c、 点击“确定”按钮

d、显示删除记录的结果

e、 用例结束

查看收支分析用例

查看收支分析：用户可以查看收支分析图。

用例ID：F1.6.1

用例名称：查看收支分析

参与者：用户

用例描述：用户查看某段时间内的收支分析图

前置任务：成功登录系统

基本事件流： a、 点击“报表”下拉菜单

b、 选择“收支分析”进入收支分析界面

c、 选择一个时间段

d、点击“确定”按钮

e、 显示这段时间内的收支分析图

f、用例结束

查看对账表用例

查看对账表：用户可以查看对账表（每天流入流出金额统计表）。

用例ID：F1.6.2

用例名称：查看对账表

参与者：用户

用例描述：用户查看对账表

前置任务：成功登录系统

基本事件流： a、 点击“报表”下拉菜单

b、 选择“对账表”进入对账表界面

c、用例结束

找回密码用例

找回密码：忘记密码的用户可以找回密码。

用例ID：F1.7.1

用例名称：忘记密码

参与者：用户

用例描述：用户通过注册的邮箱找回密码

基本事件流： a、 进入登录页面

b、 点击“忘记密码”

c、输入注册的邮箱和其他信息

d、 点击“发送验证码”

e、 发送验证码到填写的邮箱

f、 点击“找回密码”

g、 重置密码成功

h、 用例结束

说明：填写的邮箱必须是注册的邮箱，才能找回密码

修改密码用例

修改密码：用户通过该途径修改个人密码。

用例ID：F1.7.2

用例名称：修改密码

参与者：用户

用例描述：用户登录后进行密码的修改

前置任务：成功登录系统

基本事件流： a、 点击“设置”下拉菜单

b、 选择“修改密码”进入修改密码界面

c、 输入原密码、新密码、确认密码

d、 点击“修改密码”按钮

e、 提示修改成功

f、 用例结束

说明：输入的原密码必须正确、新密码和确认密码相同才能修改密码

、

修改个人信息用例

修改个人信息：用户可以通过该途径修改个人信息。

用例ID：F1.8.1

用例名称：修改个人信息

参与者：会员用户

用例描述：用户修改个人信息

前置任务：成功登录系统

基本事件流： a、 点击“设置”下拉菜单

b、 选择“个人资料”进入个人资料界面

c、 修改相关信息

d、 点击“保存”按钮

**e**、显示修改个人信息的结果

f、 用例结束

数据库设计

## 实体E-R图设计

用户信息实体E-R图

用户注册系统的时候要求输入的一些个人信息，用户登录系统的时候需要输入账号和密码，用户信息实体包括用户ID、邮箱、密码、注册时间、生日、性别、QQ、个人介绍这些属性，用户信息实体E-R图如图2显示。



图2 用户E-R图

记账分类实体E-R图

用户可以设置自己的记账类型，系统具有收入、消费、借入、借出、还款、收债这六个固定的分类，用户可以根据自己的实际需要在这些分类下设置子分类，这种设计有利于用户对自己的财务有个比较清晰的分类。记账分类实体包括分类ID、用户ID、分类、子分类属性，记账分类实体E-R图如图3所示。



图3 记账分类E-R图

账单信息实体E-R图

用户使用账单管理功能对账单进行修改的时候，系统直接引用这条记账记录的账单信息实体属性。账单信息实体包括账单ID、用户ID、记账分类、记账子分类、金额、日期、途径、备注、创建时间、更新时间等属性，账单实体E-R图如图4所示。



图4 账单E-R图

验证码实体E-R图

用户进行注册和找回密码时，系统会发送验证码到用户指定的邮箱，系统对用户输入的验证码进行验证码，验证成功才能成功注册和找回密码。验证码实体包括验证码ID、邮箱、code、操作（注册或找回密码）、创建时间等属性，验证码实体E-R图如图5所示。



图5 验证码E-R图

## 数据库表设计

用户信息表

会员用户信息表属性：用户ID、邮箱、密码、注册时间、生日、性别、QQ、个人介绍。其中用户ID为主键，性别有男、女和保密三个值，分别由male、female和secret表示，性别默认为secret。表结构如表2所示。

表2 用户信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | | 域名 | 域名描述 | 数据类型 | 长度 | 备注 |
| 1 | id | | 用户ID | char | 11 | 主键 |
| 3 | emai | | 邮箱 | varchar | 20 | not null |
| 4 | password | | 密码 | varchar | 15 | not null |
| 5 | username | | 用户名 | varchar | 20 | not null |
| 6 | register\_date | | 注册时间 | datetime | 0 | not null |
| 7 | birthday | | 生日 | datetime | 0 |  |
| 8 | gender | | 性别 | varchar | 6 | default="secret" check(male, female, secret) |
| 9 | qq | | QQ | int | 15 |  |
| 10 | introduction | | 个人介绍 | varchar | 100 |  |

记账分类信息表

记账分类信息表属性：分类ID、用户ID、分类、子分类，其中分类ID为主键。表结构如表3所示。

表3 记账分类信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | | 域名 | 域名描述 | 数据类型 | 长度 | 备注 |
| 1 | id | | 分类ID | int | 11 | 主键 |
| 2 | userid | | 用户ID | varchar | 11 | not null |
| 3 | accountingtype | | 分类 | varchar | 20 | check(borrowed, consumption, debtcollection, lend, repayment,income) |
| 4 | category | | 子分类 | varchar | 20 | not null |

账单信息表

账单信息表属性：验证码ID、邮箱、code、操作（注册或找回密码）。其中账单ID为主键。账单信息表结构如表4所示。

表4 账单信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | | 域名 | 域名描述 | 数据类型 | 长度 | 备注 |
| 1 | id | | 账单ID | int | 11 | 主键 |
| 2 | userid | | 用户ID | varchar | 11 | not null |
| 3 | accountingtype | | 记账分类 | varchar | 2 | not null |
| 4 | create\_date | | 创建时间 | datetime | 0 | not null |
| 5 | update\_date | | 更新时间 | datetime | 0 | not null |
| 6 | date | | 日期 | date | 0 | not null |
| 7 | category | | 记账子分类 | varchar | 20 | not null |
| 8 | amount | | 金额 | double | 0 | not null |
| 9 | way | | 途径 | varchar | 20 |  |
| 10 | remark | | 备注 | varchar | 50 |  |

验证码信息表

验证码表属性：验证码ID、邮箱、code、操作、创建时间。其中验证码ID为主键，操作有‘0’和‘1’两个值，‘0’代表 “注册”操作，‘1’代表 “找回密码”操作，其他属性值不能为空。表结构如表5所示。

表5 常用联系人信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | | 域名 | 域名描述 | 数据类型 | 长度 | 备注 |
| 1 | id | | 验证码ID | int | 11 | 主键 |
| 2 | email | | 邮箱 | varchar | 20 | not null |
| 3 | create\_date | | 创建时间 | datetime | 0 | not null |
| 4 | code | | code | varchar | 6 | not null |
| 5 | action | | 操作 | varchar | 6 | not null |

系统设计

系统目标

开发一个面向个人的、操作简单、不会充斥着理财产品广告、能满足大多数人的需要的个人财务管理系统。该系统的主要功能是记录管理个人财务信息，满足用户分类添加、删除、查询、修改资金信息，同时还能统计分析用户的财务行为，通过图标很直观地展示给用户。用户通过使用该系统进行记账管理，可以对自己的财务状况有一个很详细、很直观的认识。

将所学的软件开发知识进行实战开发系统，将知识运用到实际中去，提高自己的编程能力。

## 系统的功能

该收支管理系统是一个面向个人的财务管理系统，主要有一下几个功能：

（1）账号系统：用户要使用该系统，首先要注册帐号登录，支持忘记密码找回密码功能和修改密码；

（2）设置：个人基本信息设置、收支分类的设置；

（3）记账功能：记账记录收入、支出、借入、借出、还款、收债；

（4）收支分析：用户可以查看任何时间段之间的日常收支分析图、对账表；

（5）搜账：支持用户查询个人财务行为；

（6）记账管理：就有管理账单的功能，支持账单的修改、删除。

## 系统开发环境

本文系统在windows10下开发，使用PyCharm开发环境，采用MySQL5.7数据库存储数据，使用 Django内置的服务器、开发环境是Python 2.7。

Python简介

Python是面向对象的一种解释型计算机编程语言，在开发的过程中不需要进行编译，由荷兰的Guido van Rossum于1989年发明，于1991年首次公开发行。

Python语法的一个特点是简洁明了，它被强制使用空格作为语句的缩进。

Python有一个丰富而强大的库。它通常被称为胶水语言，可以很容易地将其他语言中的各种模块(特别是C/ C++)链接起来（Wesley Chun，2016）。一个常见的应用场景是使用Python很快地生成程序的原型(有时甚至是应用程序的最终接口)，然后用更合适的编程语言来重写它，例如在3D游戏中有特殊要求的图形渲染模块，如果性能要求比较高，那么可以用C/ C++来重写，然后封装到Python可以调用的扩展库中。应该注意的是，在使用扩展库时，有时需要考虑跨平台无法实现的问题(Eric Matthes，2016)。

PyCharm简介

PyCharm是一个Python IDE，它有一整套工具，可以帮助用户在使用Python语言时提高他们的工作效率，比如项目管理、调试、代码跳转、单元测试、语法高亮显示、智能提示、代码跳转、自动完成、版本控制。此外，该IDE还提供了支持Django框架下的专业web开发的高级功能。

PyCharm是由JetBrains创建的。VS2010的重构插件Resharper也是来自JetBrains。在支持谷歌应用引擎的同时，PyCharm支持IronPython。这些特性，在高级代码分析程序的支持下，使PyCharm成为Python专业开发人员和初创企业的强大工具。

MySQL简介

My是MySQL联合创始人之一Monty Widenius的女儿的名字。MySQL是My和SQL的结合，这是MySQL命名的来源。

MySQL是一个关系型数据库管理系统。它是Oracle 旗下的产品，是一款支持的开源的数据库软件。这意味着任何人都可以不用花一分钱就可以使用MySQL。此外，如果有必要，您还可以更改其源代码或执行二次开发以满足您的需求。虽然MySQL是开源软件，但是也可以通过从Oracle购买商业许可版本，用于高级支持服务(对于特殊企业用户)。

与其他数据库软件(如Microsoft SQL Server或Oracle数据库)相比，MySQL是非常容易学习掌握的。在各种平台上运行比如UNIX、Linux、Windows等MySQL都是可以运行的。它可以安装在服务器上，甚至可以安装在桌面系统上。此外，MySQL是可靠的、可扩展的和快速的(Baron Scbwartz，Peter Zaitsev，vadimTkacbenko，1023)。

如果想要开发一个网站或web应用程序，MySQL是一个值得强烈推荐的、一个不错的选择（Abraham Silberschatz，Henry F.Korth，S.Sudarshan，2015）。

Django简介

Django是一个用Python编写、开放源码的web应用程序框架。它最初开发是为了管理劳伦斯出版集团的基于新闻内容的一些网站，即CMS(内容管理系统)软件。它于2005年7月在BSD许可下发布。这个框架是以来Django Reinhardt（来自比利时的吉他手）来命名的。

Django是基于MVC结构的一个框架。但是在Django中，接受用户输入的控制器的部分是由框架本身处理的，所以Django更关心模型（Model）、模板(Template)和视图（Views），也被称为 MTV模式(Jeff Forcier，Paul Bissex，Wesley Chun，2009)。

Django视图没有处理用户的输入，但只决定向用户显示哪些数据，Django模板只决定如何呈现Django视图指定的那部分数据。或者，Django进一步将MVC中的视图分解为Django视图、Django模板的两部分，它们分别决定“显示什么数据”和“如何显示”，这样Django的模板就可以根据需要替换，而不只是只能使用内置的模板。而MVC的控制器部分，它是由Django框架的URLconf实现的。URLconf机制使用正则表达式来匹配URL，然后调用相应的Python函数。URLconf没有对URL的规则作任何限制。您可以设计任何风格的URL样式，无论是传统的、RESTful的还是其他的。框架封装了控制层，你只需要的数据交互这一层进行数据库表的读取、写入、删除和更新等操作（王冉阳，2009）。

## 系统的设计

总体流程设计

总体设计流程图由个人收支管理系统登录界面开始，没有注册的用户可以登录的注册接口跳转到注册界面，注册成功则直接进入系统的首页，已经注册的用户可以通过登录页面登录进入系统。用户在已注册但是不记得登陆密码的情况下，可以通过忘记密码重置密码界面找回密码，找回密码后登录系统（钱乐秋，赵文耘，朱俊珏，2014）。

登录成功后系统将进入系统首页页面（系统默认选中导航栏的“首页”）。该主页有资产概况、收支表、记账记录等功能，记账记录可以输入多条件进行查询、也可以进行修改和删除。

点击导航栏的“记账”进入记账页面，可以选择“收入”、“消费”、“借入”、“借出”、“还款”、“收债”这六种分类进行记账。

点击导航栏的“报表”下拉菜单可以选择查看收支分析或对账表

点击导航栏的“设置”下拉菜单可以选择分类设置对记账分类进行管理，选择修改密码可以修改系统的登录密码，选择个人资料可以查看及修改个人资料

整体架构如图6所示。

图6 总体流程图

注册流程设计

用户在注册页面输入用户名、常用邮箱、密码和和确认密码后点击“发送验证码”，进行邮箱验证，系统发送验证码到注册邮箱，用户在注册页面输入验证码后点击“注册”，验证通过则注册成功，否则注册失败。



图7注册流程图

登录流程设计

用户在登录页面输入账号（注册邮箱）和密码后点击登录，账号和密码都正确则登录成功，否则登录失败。



图8 登录流程图

记账流程设计

选择记账进入记账模块，选择记账分类然后填写记账的相关信息，点击保存即完成记录一条账单。



图9 记账流程图

总体模块的设计

本系统主要是由两个模块构成的：前端界面模块和后台逻辑模块。

（1）前端界面模块

前端界面模块实现了用户可操作的页面，包括了登录页面、注册页面、找回密码页面、记账记录展示页面、记账页面、报表页面、分类设置页面、个人资料页面等。

（2）后台逻辑模块

后台逻辑模块是系统的关键，前端与数据库的交互都是要通过后台逻辑模块处理，起到相当于中央处理机制的作用。后台逻辑模块还实现了对前端请求的过滤，信息合法性的校验等功能。

## 前端界面模块主要设计与实现

表单设计

使用bootstrap框架，快速设计样式优美的表单(Adam Freeman,2014)。

//需要导进bootstrap框架

<!-- Bootstrap 核心 CSS 文件 -->

<link href="https://cdn.bootcss.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="stylesheet">

<!-- 最新的 Bootstrap 核心 JavaScript 文件 -->

<script src="https://cdn.bootcss.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

//登录页面的html代码

<form class="form-horizontal col-sm-offset-3 col-md-offset-3">  
<div class="form-group">   
 <input class="form-control required" type="text" placeholder="请输入账号（注册邮箱）" >  
</div>   
<div class="form-group">   
 <input class="form-control required" type="password" placeholder="请输入密码" name="password" >  
</div>   
<div class="form-group">   
 <input type="submit" class="btn btn-success" value="登录 ">   
</div>

</form>

表单校验

通过前端bootstrapValidator插件对表单进行检查，能够有效地验证信息的合法性，避免了后台的无效请求，同时减少了数据库的非法操作（徐涛，2014）。

//引入bootstrapValidator插件

<script src="https://cdn.bootcss.com/bootstrap-validator/0.5.3/js/bootstrapValidator.min.js"></script>

<script src="https://cdn.bootcss.com/bootstrap-validator/0.5.3/css/bootstrapValidator.min.css"></script>

//登录表单的校验

$(function () {

$('#login\_form').bootstrapValidator({

message: 'This value is not valid',

feedbackIcons: {

valid: 'glyphicon glyphicon-ok',

invalid: 'glyphicon glyphicon-remove',

validating: 'glyphicon glyphicon-refresh'

},

fields: {

email: {

message: '账号验证失败',

validators: {

notEmpty: {

message: '账号不能为空'

},

emailAddress: {

message: '邮箱格式不正确'

}

}

},

password: {

message: '密码验证失败',

validators: {

notEmpty: {

message: '密码不能为空'

},

stringLength: {

min: 6,

max: 16,

message: '密码长度在6到16之间'

},

different: {

field: 'Username',

message: '密码不能和邮箱相同'

}

}

}

}

});

});

日期选择

通过引进bootstrap-datetimepicker插件轻松实现日期的选择，格式化日期的输入(David Cochran，Ian Whitley，2015)。

//引进bootstrap-datetimepicker插件

<link href="{% static 'online/static/css/bootstrap-datetimepicker.min.css' %}" rel="stylesheet" />

<script src="{% static 'online/static/js/bootstrap-datetimepicker.min.js' %}"></script>

<script src="{% static 'online/static/js/bootstrap-datetimepicker.zh-CN.js' %}"></script>

//js代码

$("[name='datetimepicker']").datetimepicker({

language: 'zh-CN',//显示中文

format: 'yyyy-mm-dd',//显示格式

minView: "month",//设置只显示到月份

initialDate: new Date(),

autoclose: true,//选中自动关闭

todayBtn: true,//显示今日按钮

locale: moment.locale('zh-cn')

});

图表绘制

引入基于html5 canvas技术的图表插件Chart.js,不需要依赖任何外部工具库即可制作出色的响应式图表。

//引进bootstrap-datetimepicker插件

<link href="{% static 'online/static/css/bootstrap-datetimepicker.min.css' %}" rel="stylesheet" />

<script src="{% static 'online/static/js/bootstrap-datetimepicker.min.js' %}"></script>

<script src="{% static 'online/static/js/bootstrap-datetimepicker.zh-CN.js' %}"></script>

//js代码

$("[name='datetimepicker']").datetimepicker({

language: 'zh-CN',//显示中文

format: 'yyyy-mm-dd',//显示格式

minView: "month",//设置只显示到月份

initialDate: new Date(),

autoclose: true,//选中自动关闭

todayBtn: true,//显示今日按钮

locale: moment.locale('zh-cn')

});

网页动态刷新

使用jQuery框架的Ajax服务器交换数据技术，在不重新加载整个页面的情况下实现对网页的局部刷新功能(单东林，张晓菲，魏然，2015)。

//引进jQuery框架

<!-- jQuery文件，务必在bootstrap.min.js 之前引入 -->

<script src="https://cdn.bootcss.com/jquery/2.1.1/jquery.min.js"></script>

//多条件查询账单的js代码

//用户输入查询 条件点击搜账，只有列表数据局部更新，整个页面没有重新加载

$("#searchbtn").click(function () {

document.getElementById('overview').style.display='none';

var postData= {};

$(this).parents("[name='searchform']").find('input,select').each(function () {

var v = $(this).val();

var n = $(this).attr('name');

postData[n] = v;

});

$.ajax({

url: '/online/search\_accounting/',

type: 'POST',

data: postData,

success:function(data,status){

$("#mysearch").html(data);//将id为#mysearch的标签的html替换为返回的data

for(var i = 1;i<$("#myrecord tr").length;i++){

$('#myrecord tr:eq('+i+') td:first').text(i);

}

}

})

});

## 后台逻辑模块主要设计与实现

### 处理请求

后台用视图函数统一接受、处理前端的请求以及将结果返回前端页面，在url.py中设置视图函数相关的URL，即是不同的请求对应不同的处理函数。当前端发出一个请求时，系统会根据请求路径发送数据到相应的视图函数进行处理。

def login(request): //登录页面的后台处理代码

if request.method == 'GET':

return render\_to\_response('login.html')

state = None

if request.method == 'POST':

email = request.POST.get('email', '')

password = request.POST.get('password', '')

user = User.objects.filter(email=email, password=password)

if user:

response = HttpResponseRedirect(reverse('home'))

request.session['email'] = email

return response

else:

state = 'not\_exist\_or\_password\_error'

content = {

'active\_menu': 'index',

'state': state,

'user': None

}

return render(request, 'login.html', content)

### 发送邮箱验证信息

在注册和找回密码的时候需要进行邮箱验证，系统生成验证码发送到用户提供的邮箱，用户输入验证码，后台校验成功才能成功注册和找回密码。

//给邮箱发送验证码

code\_list = []

for i in range(10): # 0-9数字

code\_list.append(str(i))

for i in range(65, 91): # A-Z

code\_list.append(chr(i))

for i in range(97, 123): # a-z

code\_list.append(chr(i))

myslice = random.sample(code\_list, 6) # 从list中随机获取6个元素，作为一个片断返回

verification\_code = ''.join(myslice) # list to string

new\_email = Verification\_code.objects.create(email=email, code=verification\_code, action='1')

new\_email.save()

email\_title = "记账本验证码"

email\_body = "您正在找回记账本密码，这是您的验证码：{0}".format(verification\_code) # 发送邮件

status = send\_mail(email\_title, email\_body, EMAIL\_FROM, [email])

### 分页显示

从数据库中提取数据，设置每页显示的数量，分页计算页面数量，再返回到前端分析显示（齐金刚，李滔，李晋军，2014）。

//记账列表分页显示

after\_range\_num = 5 # 当前页前显示5页

befor\_range\_num = 4 # 当前页后显示4页

try: # 如果请求的页码少于1或者类型错误，则跳转到第1页

page = int(request.POST.get("page", 1))

if page < 1:

page = 1

except ValueError:

page = 1

paginator = Paginator(accounts, 10) # 设置books在每页显示的数量

try: # 跳转到请求页面，如果该页不存在或者超过则跳转到尾页

accounts\_list = paginator.page(page)

except(EmptyPage, InvalidPage, PageNotAnInteger):

accounts\_list = paginator.page(paginator.num\_pages)

if page >= after\_range\_num:

page\_range = list(paginator.page\_range)[page - after\_range\_num:page + befor\_range\_num]

else:

page\_range = list(paginator.page\_range)[0:int(page) + befor\_range\_num]

template = 'searchaccounting.html'

return render(request, template, {"accounts": accounts,"page\_range": page\_range})

### 拦截器

用户需要登录了才能使用该系统，未登录的用户将被拦截到登录页面。

//拦截未登录的用户

from django.shortcuts import HttpResponseRedirect

from django.shortcuts import render\_to\_response

try:

from django.utils.deprecation import MiddlewareMixin # Django 1.10.x

except ImportError:

MiddlewareMixin = object # Django 1.4.x - Django 1.9.x

class LoginMiddleware(MiddlewareMixin):

def process\_request(self, request):

if not request.path.startswith(('/online/login', '/online/registered','/online/retrievepassword', '/online/send\_retrievepassword\_verification\_code', '/online/send\_registered\_verification\_code')):

if request.session.get('email', None):

pass

else:

return HttpResponseRedirect('/online/login')

### 数据库交互

后台直接使用Django封装好的函数来进行数据库操作，只需要简单的配置就可以使用MySQL数据库，创建或修改类只需要同步数据库就能修改对应的表设计。

//配置MySQL数据库

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'NAME': 'accountingsystem',

'USER': 'root',

'PASSWORD': '\*\*\*\*\*\*',

'HOST': '127.0.0.1',

'PORT': '3306',

}

}

//创建类，同步数据库即可生成对应的表

class Verification\_code(models.Model):

id = models.AutoField(primary\_key=True)

email = models.CharField(max\_length=30)

create\_date = models.DateTimeField(auto\_now\_add=True)

code = models.CharField(max\_length=6)

action = models.CharField(max\_length=6)

//数据库操作

//查询操作

Verification\_code.objects.filter(email=email, action='0').values("code").order\_by('-create\_date')[:1][0]['code']:

//添加操作

User.objects.create(email=email, username=username, password=password, )

//更新操作

User.objects.filter(id=userid).update(password=password)

系统运行效果及功能说明

## 注册

（1）注册界面



图10 注册界面图

（2）注册功能说明

如图10所示，用户根据提示填写个人信息，当信息不符合要求时，将无法点击注册按钮；注册过的邮箱无法重复注册，两次输入的密码不相同也无法进行注册，必须保持一致。

## 登录

（1）登录界面



图11登录界面图

（2）登录功能说明

如图11所示，用户输入账号和密码登录，没有注册的用户可以点击“注册账号”进行注册，忘记密码则可以点击“忘记没密码”找回密码。登录成功将跳转至系统首页。

## 找回密码

（1）找回密码界面



图12找回密码界面图

（2）找回密码功能说明

如图12所示，用户输入注册的邮箱，输入密码和确认密码，点击“发送验证码”，系统将验证码发送到邮箱中，用户输入验证码后点击“找回密码”，找回密码成功则跳转至系统首页。找回密码成功的条件时按照要求输入相关信息。

## 修改密码

（1）修改密码界面



图13 修改密码界面图

（2）修改密码功能说明

如图13所示，用户要登录才能修改密码，输入的旧密码要正确，两次输入的新密码符合要求并且保持一致才能修改成功。

## 修改个人资料

（1）修改个人资料界面



图14 修改个人信息界面图

（2）修改个人资料功能说明

如图14所示，用户个人资料中的邮箱是不能修改的，其他信息可以在输入框中编辑完后点击“保存”即可修改个人资料。

## 分类设置

（1）分类设置界面



图15 分类设置界面图

（2）分类设置功能说明

如图15所示，用户可以选择添加收入、支出、借入、借出、还款、收债六种分类，点击相应的分类弹出输入框输入分类名称点击添加即可，不能重复添加相同的分类。点击具体分类后面的编辑图标可对该分类进行编辑，点击删除图标可删除该分类。

## 记账

（1）记账界面



图16 记账界面图

（2）记账功能说明

如图16所示，用户登录系统，点击导航菜单栏的“记账”进入记账功能界面。用户可以选择收入、支出、借入、借出、还款、收债六种分分类分别进行记账，输入日期、金额、账户、备注，选择分类点击“保存”即可添加一条记账记录。其中日期默认选择当前日期，同时用户也可以选择一个自己想要的日期；金额显示用户只能输入数字，不能输入其他字符；分类只能选择该大分类下面的子分类，不支持手动输入其他的分类，如果想要增加分类，可以先通过分类设置功能设置；账户是指用户此次使用的收入或支出的金融账户，比如支付宝、信用卡等。

## 账单管理

（1）账单管理界面

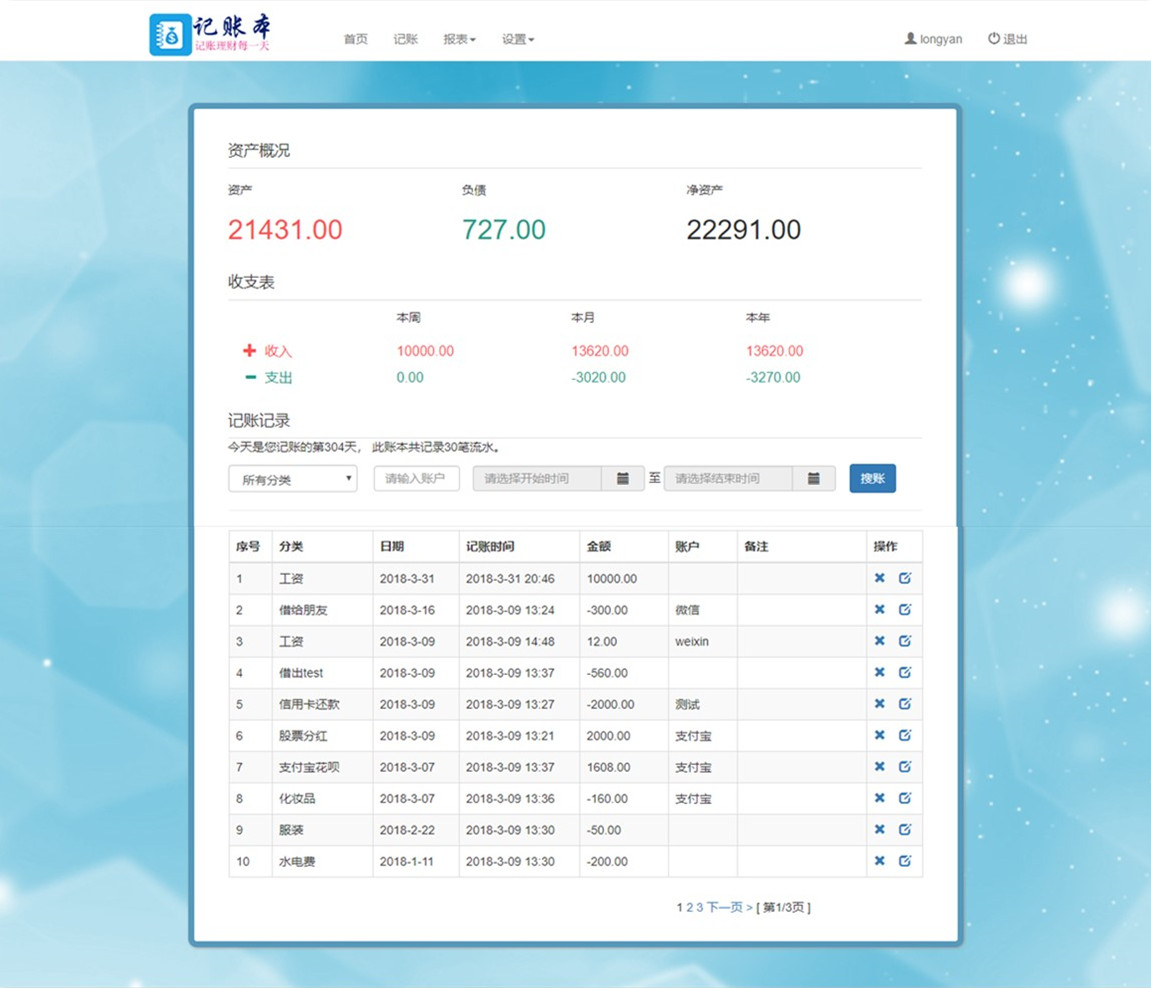


图17 账单管理界面图

（2）订单管理功能说明

如图17所示，显示当前用户的所有账单记录，可以进行多条件查询、修改删除账单操作。

## 收支分析

（1）收支分析图界面



图18 收支分析界面图

（2）收支分析功能说明

如图18所示，用户可以选择一个时间段，查看这段时间内的收支图表，对该段时间内的财务行为一目了然。

## 对账表

（1）对账表界面



图19 对账表界面图

（2）对账表功能说明

如图19所示，对账表包括日对账表和月对账表，直观显示每天、每月的财务流入流出的情况。

结论与展望

本论文介绍了课题的选题背景、需求分析、数据库设计到系统设计。该系统后台部分使用了Django框架，将程序划分为视图、模板、模型三部分，提高了业务处理效率。前端部分引进bootstrap、jQuery框架，使用了html、css、javascript技术可以轻易实现页面效果。

当然，本文系统还有很多地方可以改进：

（1）本文系统功能相对单薄，没有预算设置、超额提醒等功能，可以增加一些功能让系统更加满足用户使用。

（2）代码复用率很低，很多相似功能的代码无法做到复用。

（3）本人系统的收支分析统计的维度相对较少，可以增加一些维度和多种分析图来让用户更直观地了解自己日常地财务行为。

（4）前端html代码兼容性做得不够好，当屏幕大小变化了，页面不能做到完全自适应。

参 考 文 献

陈颂丽, 李海良. 基于MVC模式的个人财务管理系统的设计与实现[J]. 科技信息. 2010, 2:2-4

单东林, 张晓菲, 魏然. 锋利的jQuery（第2版）. 人民邮电出版社. 2015, 1:15-30

钱乐秋, 赵文耘, 朱俊珏. 软件工程. 清华大学出版社. 2014, 5:56-74

齐金刚, 李滔, 李晋军. Django框架Web数据查询分页技术研究[J]. 电子设计工程. 2014, 3:8-15

孙智勇, 面向高收入群体财务管理系统建设及应用研究[D]. 湖南大学. 2015, 10:2-3

王冉阳. 基于Django和Python的Web开发[J]. 电脑编程技巧与维护. 2009,1:45-50

徐涛. Web开发技术丛书 深入理解Bootstrap. 机械工业出版社, 2014, 2:156-157

Abraham Silberschatz, Henry F.Korth, S.Sudarshan. 数据库系统概念. 机械工业出版社. 2015, 10:30-35

Adam Freeman. HTML5权威指南. 人民邮电出版社.2014,8:20-130

Baron Scbwartz, Peter Zaitsev, vadimTkacbenko. 高性能MySQL. 电子工业出版社. 2013, 5:170-230

David Cochran, Ian Whitley.Bootstrap实战. 人民邮电出版社. 2015, 9:1-189

Eric Matthes. Python编程从入门到实践. 人民邮电出版社. 2016, 9:1-230

Jeff Forcier, Paul Bissex, Wesley Chun. Django Web开发指南. 机械工业出版社. 2009, 7:230-235

Jeremy Keith. JavaScript DOM 编程艺术（第2版）. 人民邮电出版社. 2011, 1:48-73

Wesley Chun. Python核心编程（第3版）. 人民邮电出版社. 2016, 5:60-75

致 谢

就要毕业告别学校了，在这里我首先要感谢我的指导老师林旭东老师，是他给予我帮助和指导，让我在规定地时间内完成了毕业设计。我觉得他是一名很认真负责地好老师，谢谢他。

其次，我要感谢学校这四年来地栽培，让我学到了很多专业知识，学会了如何做人做事。

接着，我要感谢所有遇到地同学、老师，是他们陪伴我走过了大学四年的美好时光。

然后，我要感谢我的家人，是他们给予我无限地爱和支持。

最后，非常感谢百忙之中抽出珍贵时间来参加论文审阅和答辩的各位老师和评委。谢谢大家。

**华南农业大学**

**本科生毕业论文成绩评定表**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学号 | 201430340117 | | | | | 姓名 | | 龙燕 | 专业 | 软件工程 | | |
| 毕业论文题目 | | | 基于Django与bootstrap个人收支管理系统 | | | | | | | | | |
| 指导教师评语  成绩（百分制）：　　　　　　　　　　　　　　　　　　　　指导教师签名：　　　　　　　　　　　　年　　月　　日 | | | | | | | | | | | | |
| 评  阅  人  评  语  及  成  绩  评  定 | | 成绩  评定  标准 | | **评分项目** | | | | | | | **分值** | **得分** |
| 选题  质量  20% | 1 | | 专业培养目标 | | | | 5 |  |
| 2 | | 课题难易度与工作量 | | | | 10 |  |
| 3 | | 理论意义或生产实践意义 | | | | 5 |  |
| 能力  水平40% | 4 | | 查阅文献资料与综合运用知识能力 | | | | 10 |  |
| 5 | | 研究方案的设计能力 | | | | 10 |  |
| 6 | | 研究方法和手段的运用能力 | | | | 10 |  |
| 7 | | 外文应用能力 | | | | 10 |  |
| 成果  质量40% | 8 | | 写作水平与写作规范 | | | | 20 |  |
| 9 | | 研究结果的理论或实际应用价值 | | | | 20 |  |
| 评阅人评语  成绩（百分制）：　　　　　　　　　　　　　评阅人签名：　　　　　　　　　　　　年　　月　　日 | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 答  辩小组  评语及  成  绩  评  定 | 评价项目 | 具体要求（A级标准） | 最高分 | 评分 | | | | |
| A | B | C | D | E |
| 论文质量 | 论文结构严谨，逻辑性强；有一定的学术价值或实用价值；文字表达准确流畅；论文格式规范；图表（或图纸）规范、符合要求。 | 60 | 55-60 | 49-54 | 43-48 | 37-42 | ≤36 |
|  |  |  |  |  |
| 论文报告、讲解 | 思路清晰；概念清楚，重点（创新点）突出；语言表达准确；报告时间、节奏掌握好。 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 答辩情况 | 答辩态度认真，能准确回答问题 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 答辩小组评语  是否同意通过论文答辩（打√）   1. 同意 2. 不同意   成绩（百分制）：　　　　　　　　 答辩小组成员（签名）：  　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　 年 月 日 | | | | | | | |
| 成绩总评 | 论文总评分数：  教学院长签名：  学院盖章：  年 月 日 | | | | | | | |

续上表：