政府机构网站软件

（计算机软件类综合性课程设计）

小组成员：

21071003 高立扬 组长

21071026 张馨予

19071007 马雪

完成日期：

2024/07/19

软件工程课程设计成绩评价表

**姓名** 在软件工程课程设计中： 指导老师：付利华

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 出勤情况10% | 小组合作5% | 工具使用5% | 系统分析、设计与实现、测试情况30% | 答辩、回答问题20% | 开发文档30% | 最终  成绩 |
| □全勤  □偶尔缺勤  □偶尔出勤  □从未出现 | □紧密合作  □合作较紧密  □偶尔合作  □从不合作 | □大量使用  □较多使用  □较少使用  □从未使用 | □设计方案合理、实现全部功能  □设计方案较合理，实现基本功能  □设计不合理，实现部分功能 | □准备充分、回答问题正确  □准备较充分、回答有部分错误  □没有准备、不能回答问题 | □条理清楚、符合规范  □条理较清楚、基本符合规范  □问题很大、不符合规范 |  |

**姓名** 在软件工程课程设计中： 指导老师：付利华

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 出勤情况10% | 小组合作5% | 工具使用5% | 系统分析、设计与实现、测试情况30% | 答辩、回答问题20% | 开发文档30% | 最终  成绩 |
| □全勤  □偶尔缺勤  □偶尔出勤  □从未出现 | □紧密合作  □合作较紧密  □偶尔合作  □从不合作 | □大量使用  □较多使用  □较少使用  □从未使用 | □设计方案合理、实现全部功能  □设计方案较合理，实现基本功能  □设计不合理，实现部分功能 | □准备充分、回答问题正确  □准备较充分、回答有部分错误  □没有准备、不能回答问题 | □条理清楚、符合规范  □条理较清楚、基本符合规范  □问题很大、不符合规范 |  |

**姓名** 在软件工程课程设计中： 指导老师：付利华

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 出勤情况10% | 小组合作5% | 工具使用5% | 系统分析、设计与实现、测试情况30% | 答辩、回答问题20% | 开发文档30% | 最终  成绩 |
| □全勤  □偶尔缺勤  □偶尔出勤  □从未出现 | □紧密合作  □合作较紧密  □偶尔合作  □从不合作 | □大量使用  □较多使用  □较少使用  □从未使用 | □设计方案合理、实现全部功能  □设计方案较合理，实现基本功能  □设计不合理，实现部分功能 | □准备充分、回答问题正确  □准备较充分、回答有部分错误  □没有准备、不能回答问题 | □条理清楚、符合规范  □条理较清楚、基本符合规范  □问题很大、不符合规范 |  |

**姓名** 在软件工程课程设计中： 指导老师：付利华

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 出勤情况10% | 小组合作5% | 工具使用5% | 系统分析、设计与实现、测试情况30% | 答辩、回答问题20% | 开发文档30% | 最终  成绩 |
| □全勤  □偶尔缺勤  □偶尔出勤  □从未出现 | □紧密合作  □合作较紧密  □偶尔合作  □从不合作 | □大量使用  □较多使用  □较少使用  □从未使用 | □设计方案合理、实现全部功能  □设计方案较合理，实现基本功能  □设计不合理，实现部分功能 | □准备充分、回答问题正确  □准备较充分、回答有部分错误  □没有准备、不能回答问题 | □条理清楚、符合规范  □条理较清楚、基本符合规范  □问题很大、不符合规范 |  |

目录

[政府机构网站软件 1](#_Toc1503)

[项目开发计划书 1](#_Toc16884)

[第一部分、引言 1](#_Toc10406)

[第二部分、项目概述 2](#_Toc9517)

[第三部分、实施计划 5](#_Toc22901)

[第四部分、支持条件 8](#_Toc19442)

[第五部分、专题计划要点 9](#_Toc24493)

[需求规格说明书 12](#_Toc22357)

[1．引言 12](#_Toc14980)

[2．任务概述 13](#_Toc19063)

[3．数据描述 13](#_Toc26188)

[4．功能需求 16](#_Toc1841)

[5．性能需求 22](#_Toc265)

[6．运行需求 23](#_Toc4312)

[7．其它需求 31](#_Toc28820)

[政府机构网站软件 32](#_Toc20056)

[设计说明书 32](#_Toc32467)

[开发团队：第四组 32](#_Toc3582)

[完成日期：2024.7.18 32](#_Toc17626)

[1、引言 35](#_Toc24736)

[2、概要设计 35](#_Toc17392)

[3、接口设计 50](#_Toc20236)

[4、数据库设计 55](#_Toc17349)

[5、界面设计 57](#_Toc17377)

[6、出错处理 73](#_Toc1893)

[测试计划 74](#_Toc5441)

[1．引言 74](#_Toc13989)

[2．任务概述 74](#_Toc13564)

[3．计划 75](#_Toc2082)

[4．测试项目说明 76](#_Toc801)

[5．评价 82](#_Toc22534)

[用户操作手册 83](#_Toc24092)

[1．引言 83](#_Toc31284)

[2．软件概述 83](#_Toc9423)

[3．运行环境 84](#_Toc16956)

[4.使用说明 84](#_Toc21565)

[5.运行说明 85](#_Toc22462)

[6.非常规过程 86](#_Toc29155)

[7.操作命令一览表 86](#_Toc490)

[8.程序文件（或命令文件）和数据文件一览表 86](#_Toc4385)

[9. 用户操作举例 87](#_Toc21896)

[项目开发记录 95](#_Toc25176)

[第一次会议： 96](#_Toc23882)

[第二次会议： 98](#_Toc14851)

[第三次会议： 99](#_Toc26142)

[第四次会议： 100](#_Toc23082)

[第五次会议： 101](#_Toc5981)

[第六次会议： 102](#_Toc19568)

[第七次会议： 103](#_Toc32034)

[第八次会议： 104](#_Toc31102)

[第九次会议： 105](#_Toc7811)

[项目开发总结 107](#_Toc20697)

[项目开发总结 108](#_Toc18065)

[项目开发总结 111](#_Toc11373)

**政府机构网站软件**

**项目开发计划书**

**第一部分、引言**

**1.1编写目的**

本计划编写目的是更清晰地理解政府机构网站软件分析项目的业务要求，明确项目需要做的工作，并为保证项目在预算、日期和质量方面的要求提供可执行的依据，包含了范围、进度安排、质量控制、预算和变更等在内的明确的计划和安排，以切实能保证项目能在控制中完成。

**1.2背景**

说明：

A、待开发的软件系统的名称：政府机构网站软件

B、本项目的提出者：北京工业大学计算机学院

C、开发者：第四组

D、用户：个人用户、法人用户、政府工作人员用户

E、实现该软件的计算机中心或计算机网络：HBuilder（开发平台），MySQL（数据库平台）

F、该软件与其他系统或其他机构的相互往来关系:本系统将是独立的系统，目前不与政府机构网站软件系统和其他资料系统提供接口，所产生的输出都是独立的。

**1.3定义**

略

**1.4 参考资料**

《软件工程课程设计指导书》

《软件工程课程设计》

《MySQL应用程序开发》

《HBuilder开发文档》等

**第二部分、项目概述**

**2.1 工作内容**

为完成本项目，需要按照需求分析、概要设计、详细设计、评审、编码、测试、安装和维护等不同的阶段来进行，其中，本计划不考虑维护阶段所做的工作。

需求分析明确本项目所开发产品的特性，并对不同的功能组进行划分，并得到用户方的确认。

设计阶段将该需求转化为计算机的模型，并且对实现的功能进行分配，详细设计还将提供各模块、任务、功能点的详细规划。

评审分布在项目的阶段点，是质量保证的一部分，评审活动用于确定和调整与需求的偏差，以确保最后的产品符合初始提出的要求。

编码实现将按照软件产品设计所描述的内容，编写代码实现软件各部分的功能。

测试部分包括对实现过程中的错误的修改、功能的改进的一些活动，同时包括了各子系统、模块、功能点的组合和连调。

以上的过程中，包含了不同阶段的文档输出工作，并且上一阶段的输出，通常作为下一阶段的输入而存在。

详细的工作包和任务的分配，请参考第二部分执行计划的工作分解结构。

**2.2主要参加人员**

小组成员：

主程序员：高立扬

辅助程序员：张馨予、马雪

文档管理员：马雪、张馨予、高立扬

**2.3 产品**

项目的最后的产品和可交付物包括最后完成的软件包、相关的文档、手册、培训材料等，分别如下：

**2.3.1 程序**

政府机构网站软件

**2.3.2 文件**

项目开发计划书

需求规格说明书

概要设计说明书

详细设计说明书

数据库设计说明书（或数据要求说明书）

测试计划

测试分析报告

项目开发总结

程序源代码

小组开发记录

**2.3.3 服务**

包括的服务集中在过程的后期，这些服务包括：

A、产品培训；

在软件系统正式使用之前，需要提供给甲方使用人员为期一天的培训，参与培训人员由甲方确认，具体的日期、方式，在项目的后期确定。

B、产品首次安装

提供给甲方首次的数据库服务器的安装，包括10名以内客户程序的安装。具体的日期，在项目的后期双方进行确认。

C、数据转换和系统初始化

包括系统的用户、权限等，负责提供首次数据的输入工作，其他一些牵涉到的数据转换的工作，可以通过协商、补充协议等形式来解决。

D、其他维护性的工作

合同中所包含的其他维护性的工作。

**2.3.4 未移交产品**

非移交的产品包括过程记录和过程文档，包括：

A、软件的源代码

程序的源代码不提供给甲方，但不包括在HTML页面中的Script。

B、安装程序工程

C、需求文档

D、过程评审记录

包括需求、设计、实现和验证阶段的评审记录、评审报告，都不提交给甲方。

E、设计和规划文档

包括产品设计、过程规划等方面的文档，不提供给甲方。

测试记录和测试报告

**2.4 验收标准**

A、书面材料

**个人报告**

每个学生提交个人课程设计报告（A4 打印稿，不少于20页，格式要求参看附录）交给组长。由组长连同小组报告打印报告一并按时交给老师。

小组项目开发记录本（含小组进行课程设计的活动日志与会议记录）。注：组长把小组报告、开发记录本连同所有成员的个人报告交给课代表，再由课代表统一交给老师。

B、软件与电子文档

**个人电子文档**

把含个人完成的电子文档交给小组负责保管文档的角色。个人电子文档中应当包含个人完成的开发源代码和下面“附录C、小组成员角色描述与负责完成的文档”中指定的文档和个人课程设计报告的电子文档。

小组项目电子文档

小组平时把运行系统的文档保存在组长安排的计算机硬盘小组项目专用目录中。完成项目后，把小组项目课程设计相关文档、可运行程序、源代码连同各个小组成员的文档分别复制到在小组项目专用的目录下不同的子目录及其每个成员的子目录中。个人和小组电子文档应当包括的内容参看后面附录E中的表E1和图E2，并可以参考附录C中的文档。

**2.5 完成项目的最迟期限**

7月22日前提交程序和项目文档

**2.6 本计划的批准者和批准日期**

批准者：付利华 批准日期：2024年7月

**第三部分、实施计划**

**3.1 工作任务的分门与人员分工**

主程序员：高立扬（组长）

工作：负责整个项目的开发工作，后端程序与数据库建立，前后端的部分交互工作，系统测试与调试和小组文档核查工作。

辅助程序员：张馨予、马雪

工作：主要负责软件的结构设计，软件需求分析，前端程序的开发，前后端的部分交互工作，用户功能测试与调试和小组文档撰写等工作。

文档管理员：马雪、张馨予、高立扬

工作：主要负责完成课设小组最终报告，小组电子文档管理和帮助系统开发等工作。

注释：P —— 参与人员；A —— 负责人员；S —— 确认审核人员；

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文档 | 阶段 | 角色承担的任务 | | |
| 代号 | 名称 | 高立扬  （主程序员） | 张馨予  （辅助程序员1） | 马雪  （辅助程序员2） |
| A0 | 项目调研 | P | P | A |
| A1 | 工具选择 | A |  |  |
| A2 | 角色分工 | A |  |  |
| A3 | 项目计划 | P | P | A |
| B1 | 需求调查 | P、系统管理员需求调查 | A、客户需求调查、需求规格说明书撰写 | P、客户需求调查 |
| B2 | 结果评估 | A、审核与确定 | P | P |
| B3 | 确定需求规格说明书 | P | P | A |
| C1 | 总体设计 | A、政府机构网站应用端 | P、软件架构设计 | P、政府机构网站应用端 |
| C2 | 接口设计 | A | P、政府机构网站应用接口 | P、客户端接口 |
| C3 | 运行设计 | A | P | P |
| C4 | 数据结构设计 | A、整体数据结构 | P、客户端数据结构 | P、业务数据结构 |
| C5 | 模块设计 | A、后端数据库设计、前后端模块交互 | P、整体规划、客户端模块设计 | P、整体规划、客户端模块设计 |
| D1 | 软件界面设计 | P | A | P |
| D2 | 功能模块设计 | S | P | A |
| D3 | 系统数据设计 | A,S | P | P |
| D4 | 数据库设计 | A | P | A，S |
| D5 | 编码 | A,S | P,S | P,S |
| E1 | 测试计划 | A |  |  |
| E2 | 单元测试 | 网站应用单元 | 客户应用单元 | 客户应用单元 |
| E3 | 集成测试 | A,S | P | P |
| F1 | 修改BUG | P,S | A | P |
| F2 | 完善功能 | A | P | P,S |
| F3 | 整理文档 | 个人报告 | 个人报告 | A、整体开发过程文档 |
| G1 | 交付软件、答辩 | P | P | P |

**3.2 接口人员**

高立扬：申请开发权限、连接数据库

**3.3 进度**

**初步时间安排**：

A、第一周完成需求分析，并规范需求规格说明书

B、第一周实现概要设计，并规范概要设计说明书

C、1周时间完成详细设计，并规范详细设计说明书

D、3天时间完成测试计划，规范测试计划书，同时发现存在的问题

E、4时间解决测试中遇到的问题，晚上软件系统

F、整理开发过程的所有数据、文档及程序，完成交付任务

**3.4 预算**

无

**3.5关键问题**

**影响整个项目成败的关键问题、技术难点和风险包括：**

1）使用提高生产率工具所产生的计划节余被过高地估计了。

2）预定日期提前，但没有对产品范围或可用资源作相应调整。

3）产品比估计的大（从工具的使用、功能点、文档管理等方面来看）。

4）最终用户最终发现产品不能令人满意，要求重新设计和重做。

5）小组成员之间的冲突导致通信差、设计差、界面错误和额外工作。

6）开发工具不像预料的那样易学、有效，开发者需要时间来创建有关工作，或改用新工具。

**第四部分、支持条件**

支持本项目的开发所需要的条件和设施包括：

**4.1 计算机系统支持**

**硬件环境：**

CPU : Inter® Core™ I5或更高频率

ROM: 1.00G或更高内存支持

**软件支持：**

开发所用的操作系统：Windows 10& Windows 11

开发平台：HBuilder

开发语言：javascript、vue

数据库平台：MySQL

项目管理工具：Project

分析设计工具：processon

**4.2 需要用户承担的工作**

在程序的需求、设计、编码实施、测试控制阶段需要有人参与并评审会议

在验收和维护阶段需要参与培训人员

**4.3 由外单位提出的条件**

略

**第五部分、专题计划要点**

**5.1 配置管理计划**

配置管理所关心的问题涉及以下三点：

1）仔细定义软件系统的交付物；

2）严格控制对可交付物的变更；

3）确保软件系统的可交付物与既定的或者经过核准修订的可交付物相一致。

**5.2 质量管理计划**

通过严格和规范的过程管理、文档化的流程开发，提高生产效率，为客户提供稳定、易用和符合要求的产品系列。

参考《软件工程课程设计指导书》认真完成课程设计，交付程序和相关文档。

**5.3 沟通计划**

本项目在规模上属于小项目，在人员安排和沟通方面都比较清晰和明确。在上面的职责分配中做了每人所参与、负责、评议的详细说明。

**项目成员**：

（主程序员） 高立扬

（辅助程序员）张馨予、马雪

（文档管理员）马雪、张馨予、高立扬

**沟通方式**：

本项目组成员之间的沟通方式包括：

1）电话；

2）电子邮件；

3）非正式的会谈或者正式的会议；

固定的电子邮件列表、电话、学号如下：

高立扬：439767274@qq.com;PHONE: 15101082900；21071003

张馨予：2428309412@qq.com;PHONE:18810131211；21071026

马雪: 13522038607 @163.com;PHONE: 13522038607,19071007

2024-7-17

**软件需求规格说明书**

**开发成员：高立扬、张馨予、马雪**

**日 期： 2024/7/17**

需求规格说明书

## 1．引言

### 1.1编写目的

本需求规格说明书旨在详细阐述政务网站的功能需求、性能需求、安全需求、用户界面需求以及非功能性需求，为网站的开发、测试、实施和维护提供明确的指导。本文档供设计人员、开发人员参考。

### 1.2项目背景

随着信息技术的快速发展和互联网的普及，政务网站已成为政府与公众之间沟通的重要桥梁。政府机构网站不仅提供了政府相关政策法规的宣传，还能提供公共服务和解决政务问题的渠道，同时也是政府对公众负责的一个重要平台。然而，现有的政务网站面临着一些问题，如功能不完善、界面不友好、安全性不足等，这些问题严重影响了政府工作的透明度、效率和公众满意度。

为了解决现有政务网站存在的问题，我们计划开发一款功能完善、界面友好、安全可靠的政务网站小程序。这款小程序将包括政府业务办理、公共服务导航、政策发布、反馈投诉等功能，以满足公众对政府服务的多样化需求。通过优化和改进政务网站的用户体验和服务质量，我们希望能提升政府工作的透明度和效率，增强公众对政府的信任和满意度。

在这个项目中，我们将结合最新的软件设计理念和技术手段，采用用户研究和需求分析等方法，确保政务网站小程序能够真正满足公众的需求。同时，我们将注重安全性和可靠性，保障政务网站小程序的数据和信息安全。通过此项目的实施，我们期待能够为政府机构提供一种全新的、高效的沟通方式，促进政府与公众之间的交流与互动。

### 1.3定义

政务网站：指政府部门通过互联网向公众提供政务信息、在线服务、互动交流等功能的网站。

用户：指使用政务网站的公众、企业等。

管理员：指负责政务网站内容维护、功能设置、安全监控等工作的政府工作人员。

### 1.4参考资料

1. 《国家电子政务工程建设项目管理暂行办法》
2. 《政府网站发展指引》
3. 相关法律法规和技术标准

## 2．任务概述

### 2.1目标

提供全面、准确、及时的政务信息，满足公众对政务信息的查询需求。

实现政务服务的在线办理，提高政府服务效率和用户体验。

加强政府与公众的互动交流，收集社情民意，回应社会关切。

保障政务网站的安全稳定运行，防止信息泄露和非法入侵。

### 2.2运行环境

描述了本软件的运行环境，一般包括：

硬件平台：PC台式机或笔记本电脑

运行时占用内存：略；

安装所需硬盘：略；

打印机：可选。

操作系统和版本：windows XP win11；

支撑环境：uniapp微信开发者工具、云数据库；

其它与该软件有关的软件组件：无；

与该软件共存的应用程序：无；

### 2.3条件与限制

遵循国家相关法律法规和政策要求，确保政务网站的合法合规。

遵循相关技术标准和规范，确保政务网站的技术可行性和可维护性。

## 3．数据描述

### 3.1静态数据

用户表（用户ID、用户名、昵称、手机号、密码、头像）

企业表（企业ID、企业名称、企业法人、联系电话、城市、法人身份证号、法人居住地）

业务表（用户ID、政府部门人员表ID、业务ID、业务名称、业务开始时间、业务信息、业务办理状态）

户籍表（姓名、用户ID、出生日期、籍贯、居住地、身份证）

社保卡（用户ID、社保卡号、姓名、性别、居住地、电话）

投诉表（投诉ID、业务ID、投诉开始时间、投诉标题、投诉信息、反馈状态、反馈信息）

### 3.2动态数据

一、普通用户

1. 登录

输入数据：账号、密码

输出数据：登录成功/失败提示

2. 退出登录

输入数据：无（确认操作）

输出数据：退出登录成功提示

3. 修改密码

输入数据：旧密码、新密码、确认新密码

输出数据：密码修改成功/失败提示

4. 找回密码

输入数据：手机号/邮箱、验证码、新密码、确认新密码

输出数据：密码重置成功/失败提示

5. 办理业务

输入数据：业务类型、申请材料、缴费信息（如果有）

输出数据：业务办理成功/失败提示、受理编号（如果有）

6. 站内搜索（业务查询）

输入数据：查询关键词

输出数据：搜索结果列表

7. 业务进度查询

输入数据：业务受理编号

输出数据：业务进度详情

8. 我要投诉

输入数据：投诉问题类型、投诉内容

输出数据：投诉提交成功提示

9. 注册

输入数据：账号、密码、确认密码、个人信息（如邮箱、手机号等）

输出数据：注册成功/失败提示

二、企业法人

（输入和输出数据基本与普通用户相同，但业务类型不同）

三、政府部门人员

1. 政务办理

输入数据：业务事项、审批材料

输出数据：审批结果、归档信息

2. 政务查询

输入数据：查询关键词

输出数据：政务信息详情

3. 处理反馈

输入数据：投诉内容、处理意见

输出数据：反馈处理结果

### 3.3数据库介绍

#### FastAdmin是一款基于PHP+Bootstrap的开源后台框架，它采用了Apache2商业友好开源协议，专为开发者精心打造，旨在提高开发效率并简化Web应用程序的开发。FastAdmin基于ThinkPHP框架进行开发，并集成了Bootstrap等前端框架，使得开发出的后台系统既快速又美观。同时，FastAdmin支持免费商用，无需单独授权，为开发者提供了极大的便利。在FastAdmin中，PHP数据库交互是Web应用程序开发的重要组成部分。FastAdmin支持多种数据库，但最常用的是MySQL数据库。PHP通过MySQLi或PDO扩展与MySQL数据库进行交互，这两种扩展都提供了丰富的函数和方法来执行SQL语句、处理查询结果等。

### 特性：

1. 一键生成CRUD

FastAdmin具有强大的一键生成功能，可以根据数据库表一键自动生成控制器、模型、视图、JS、语言包、菜单、回收站等。这大大节省了开发者的时间，提高了开发效率。

2. 权限管理系统

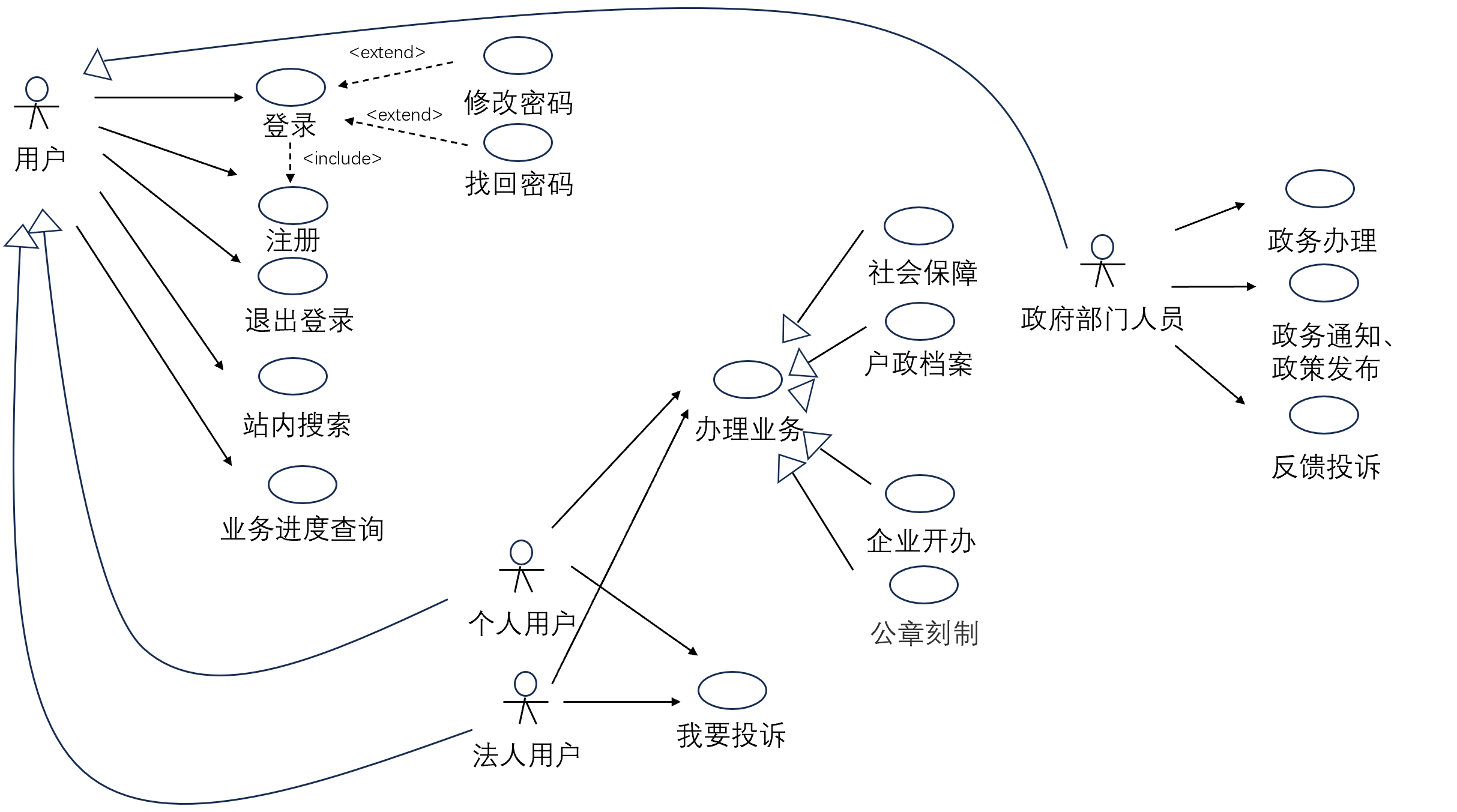
FastAdmin基于Auth验证的权限管理系统，支持无限级父子级权限继承。父级管理员可以任意增删改子级管理员及权限设置，支持单管理员多角色，支持管理子级数据或个人数据。这使得FastAdmin在权限管理方面非常灵活和强大。

3. 数据库管理工具

FastAdmin提供了数据库管理工具，允许开发者在后台直接管理数据库，包括查看数据库结构、执行SQL语句、备份和恢复数据库等。这些工具使得数据库管理更加方便和高效。

## 4．功能需求

### 4.1功能划分



### 4.2功能描述

|  |  |
| --- | --- |
| **用例名称：login（登录）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户访问政务网站的登录页面。 | |
| 1.2后置条件:如果此用例执行成功，用户成功登录；如果执行不成功，系统状态不变。 | |
| 1.3主事件流 | |
| 1)当用户点击“登录”时，此用例开始。 | |
| 2)用户输入必要信息(账号和密码)。 | |
| 3)系统验证身份。 | |
| 4)系统用户成功登录。 | |
| 1.4备选事件流 | |
| E-1:若系统在验证身份时检测到登录信息不正确，系统给出提示信息“账号或密码错误，请重试！”，此用例结束。 | |

|  |  |
| --- | --- |
| **用例名称：Registration（注册）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户访问政务网站的注册页面。 | |
| 1.2后置条件:如果此用例执行成功，用户成功注册，则此用户的信息增加到数据库中；如果执行不成功，系统状态不变。 | |
| 1.3主事件流 | |
| 1)当用户点击“注册”时，此用例开始。 | |
| 2)用户输入必要个人信息。 | |
| 3)系统确认用户信息，并创建新账户。 | |
| 1.4备选事件流 | |
| E-1:若用户在系统中填写的个人信息不正确或不全，系统给出提示信息“信息错误/不能有空白信息”，此用例结束。 | |

|  |  |
| --- | --- |
| **用例名称：Change password（修改密码）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户已有注册账户。 | |
| 1.2后置条件:如果此用例执行成功，用户成功修改密码；如果执行不成功，密码不变。 | |
| 1.3主事件流 | |
| 1)当用户点击“修改密码”时，此用例开始。 | |
| 2)用户输入旧密码。 | |
| 3)系统验证原密码正确。 | |
| 4)用户输入新密码。 | |
| 5）用户确认修改。 | |
| 1.4备选事件流 | |
| E-1:若系统验证旧密码不正确，系统给出提示信息“原密码错误，请重试！”，直到成功设置新密码此用例结束。 | |

|  |  |
| --- | --- |
| **用例名称：Retrieve password（找回密码）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户忘记账户密码。 | |
| 1.2后置条件:如果此用例执行成功，用户成功找回密码，如果执行不成功，则密码没有找回。 | |
| 1.3主事件流 | |
| 1)当用户点击“忘记密码”时，此用例开始。 | |
| 2)用户填写手号/邮箱。 | |
| 3)系统发送验证码给用户。 | |
| 4)用户接收并填写验证码。 | |
| 5）用户设置新密码。 | |
| 6）用户确认修改。 | |
| 1.4备选事件流 | |
| E-1:若用户填写验证码不正确，系统给出提示信息“验证码错误，请重试！”，直到成功填写验证码结束。 | |
| E-2：若系统在60s内未发送验证码，则用户重新在60s内获得新的验证码，直到获得到验证码结束。 | |

|  |  |
| --- | --- |
| **用例名称：Search（站内搜索）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户已经成功登录并点击进入搜索界面。 | |
| 1.2后置条件:如果此用例执行成功，则页面跳转到搜索的功能页面，如果执行不成功，则依旧在初始页面。 | |
| 1.3主事件流 | |
| 1)当用户点击“搜索”时，此用例开始。 | |
| 2)用户填写要搜索的内容并回车。 | |
| 3)系统跳转到搜索到的页面。 | |
| 1.4备选事件流 | |
| E-1:若用户搜索的内容并不是政务网站中能够提供的服务，系统给出提示信息“未找到相关服务！”，此用例结束。 | |

|  |  |
| --- | --- |
| **用例名称：Service progress query（业务进度查询）** | **执行者：个人用户、法人用户、政府部门人员** |
| 1.1前置条件:用户已经成功登录。 | |
| 1.2后置条件:如果此用例执行成功，则成功查看业务办理进度，如果执行不成功，则依旧在初始页面。 | |
| 1.3主事件流 | |
| 1)当用户点击“我的业务”时，此用例开始。 | |
| 2)用户选择要查询的业务种类。 | |
| 3)用户点击业务进度查询，跳转到业务进度页面。 | |
| 1.4备选事件流 | |
| E-1:若该业务无法办理，用户查询到的业务进度显示为“无法办理”以及无法办理的原因，此用例结束。 | |

|  |  |
| --- | --- |
| **用例名称：Complaint（我要投诉）** | **执行者：个人用户、法人用户** |
| 1.1前置条件:用户已经成功登录并点击进入“我要投诉”界面。 | |
| 1.2后置条件:如果此用例执行成功，则成功提交投诉内容或反馈内容，如果执行不成功，则未提交。 | |
| 1.3主事件流 | |
| 1)当用户点击“投诉举报”时，此用例开始。 | |
| 2)用户选择要投诉的问题类型。 | |
| 3)系统跳转到投诉内容填写页面。 | |
| 4)用户填写具体投诉内容并提交。 | |

|  |  |
| --- | --- |
| **用例名称：Business handling（业务办理）** | **执行者：个人用户、法人用户** |
| 1.1前置条件:用户已经成功登录。 | |
| 1.2后置条件:如果此用例执行成功，则成功提交要办理的业务，如果执行不成功，则未提交。 | |
| 1.3主事件流： | |
| 1.3.1个人用户： | |
| 1）当用户点击“我的”时，此用例开始。 | |
| 2）系统显示不同的业务类型 | |
| 1.3.1.1社会保障： | |
| 1）用户点击“社会保障”转到具体业务 | |
| 2）点击“社保卡申请”的“办理”或“社保卡挂失”的“办理” | |
| 3）系统提示申请已提交 | |
| 1.3.1.2户政档案： | |
| 1）用户点击“户政档案”，转到具体业务 | |
| 2）点击申请“开具户籍证明”的线上办理 | |
| 3）选择申请情形 | |
| ——“个人办理”“委托人办理”“户籍注销” | |
| ——“父母双方婚内生育”“不符合计划生育政策、未办理《出生医学证明》的无户口”“我国公民与外国人、无国籍人非婚生育的无户口人员” | |
| 4）用户根据对应情形提交相关材料 | |
| 5）用户点击提交申请 | |
| 6）系统提示申请已提交 | |
| 1.3.2法人用户： | |
| 1）当用户点击“业务办理”时，此用例开始。 | |
| 2）系统显示不同的业务类型 | |
| 1.3.2.1企业开办： | |
| 1）用户点击“企业开办”转到具体业务 | |
| 2）点击申请“企业开办”或“公章刻制备案”的线下办理 | |
| 3）系统转到信息填写界面 | |
| 4）用户填写相关团体、法人信息、业务情形 | |
| 5）提交申请材料 | |
| 6）用户点击申请 | |
| 7）系统提示申请已提交 | |

|  |  |
| --- | --- |
| **用例名称：Government affairs transaction（政务办理）** | **执行者：政府部门人员** |
| 1.1前置条件:政府部门人员已经成功登录并且用户已经提交符合业务办理的材料。 | |
| 1.2后置条件:如果此用例执行成功，则成功处理政务，如果执行不成功，则无法成功处理政务并反馈给用户无法处理的原因。 | |
| 1.3主事件流 | |
| 1)当政府部门人员点击“政务办理”时，此用例开始。 | |
| 2)政府部门人员选择要办理的政务。 | |
| 3)政府部门人员审批处理材料。 | |
| 4)政府部门人员将结果反馈给用户并归档。 | |

|  |  |
| --- | --- |
| **用例名称：Feedback complaint（反馈投诉）** | **执行者：政府部门人员** |
| 1.1前置条件:政府部门人员已经成功登录并且有需要处理的投诉或反馈。 | |
| 1.2后置条件:如果此用例执行成功，则成功反馈投诉，如果执行不成功，则反馈无法处理的原因。 | |
| 1.3主事件流 | |
| 1)当政府部门人员点击“反馈投诉”时，此用例开始。 | |
| 2)政府部门人员选择要处理的投诉和反馈问题类型。 | |
| 3)政府部门人员填写处理的方式与反馈的内容并点击发布。 | |

## 5．性能需求

### 5.1数据精确度

1、正确性需求

系统正确性需求主要包括：

系统能够把用户的个人信息，政务的信息，投诉反馈信息准确地导入数据库中

客户查询时能够从数据库中准确地读取并且在界面显示

2、精度需求

输入，输出，存储数据达到单浮点数的精度即可，传输达到1%的误差以内即可。

软件的输入精度:小数点后保留5位有效数字

输出数据精度的要求: 小数点后保留5位有效数字

传输过程中的精度: 小数点后保留5位有效数字

### 5.2时间特性

时间特征需求

响应时间:0.5秒以内

更新处理时间:0.5秒钟以内

数据的转换和传送时间: 1秒以内

解题时间: 1秒以内

### 5.3适应性

1、稳定性需求

该系统部署后，在硬件和支持软件没有发生变化的情况下，能够一直保持运行状态，直到系统被升级或替代

2、灵活性需求

操作方式上的变化:该软件实用与现在流行的操作系统

3、扩展性需求

支持环境的扩展：系统运行所要求的操作系统可以从Windows 平台扩展到Mac平台等

## 6．运行需求

### 6.1用户界面

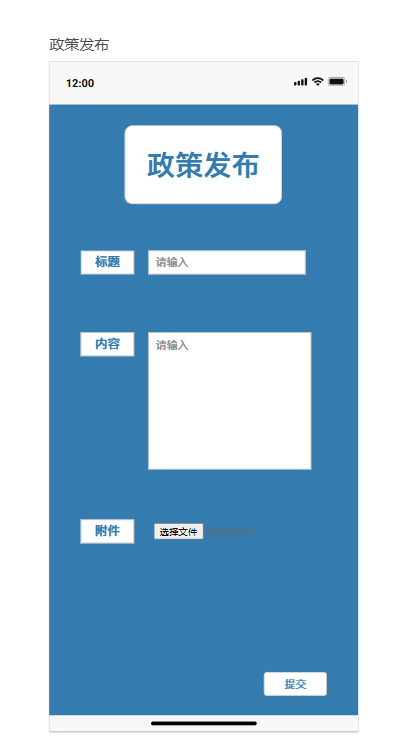
系统界面通过客户端来访问。客户端要求界面清晰，而且比较美观，其界面界面布局要合理。

能够顺利显示，显示服务器上的中文，英文，和附件信息以及用户反馈到服务器上的信息。



### 529c492f9f27710451b1b405b663b9a17eee2bcf7ad643bbd05f77daa12751





### 6.2硬件接口

政务网站应具备良好的硬件接口兼容性，能够支持多种主流的服务器、存储和网络设备。同时，应提供详细的硬件接口规范，以便与第三方硬件进行无缝对接。

### 6.3软件接口

API支持：

提供丰富的API接口，支持小程序与后端服务器、数据库以及第三方应用进行数据交换和集成。

API接口应具备良好的稳定性和可扩展性，满足不断增长的业务需求。

数据格式：

支持主流的数据格式，如JSON、XML等，方便小程序与后端进行数据交互。

数据格式应规范统一，避免出现数据格式不一致导致的错误。

1. vm.$api.getConfig = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.common/init', params);
2. vm.$api.getCaptcha = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.common/captcha', params);
3. vm.$api.getEmsSend = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.ems/send', params);
4. vm.$api.getSmsSend = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.sms/send', params);
5. vm.$api.getArchives = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.archives/index', params);
6. vm.$api.getArchivesDetail = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.archives/detail', params);
7. vm.$api.getArchivesVote = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.archives/vote', params);
8. vm.$api.getArchivesOrder = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.archives/order', params);
9. vm.$api.getChannel = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.archives/get\_channel', params);
10. vm.$api.getChannelFields = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.archives/get\_channel\_fields', params);
11. vm.$api.archivesPost = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.archives/archives\_post', params);
12. vm.$api.myArchives = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.archives/my', params);
13. vm.$api.deleteArchives = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.archives/delete', params);
14. vm.$api.getUserIndex = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.user/index', params);
15. vm.$api.getUserProfile = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.user/profile', params);
16. vm.$api.goUserLogout = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.user/logout', params);
17. vm.$api.goUserAvatar = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.user/avatar', params);
18. vm.$api.getUserInfo = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.user/userInfo', params);
19. vm.$api.getMyComment = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.my/comment', params);
20. vm.$api.getMyboutus = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.my/aboutus', params);
21. vm.$api.getOrder = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.my/order', params);
22. vm.$api.getMyagree = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.my/agreement', params);
23. vm.$api.goCommentPost = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.comment/post', params);
24. vm.$api.goCommentIndex = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.comment/index', params);
25. vm.$api.goLogin = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/login', params);
26. vm.$api.mobilelogin = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/mobilelogin', params);
27. vm.$api.goRegister = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/register', params);
28. vm.$api.goResetpwd = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/resetpwd', params);
29. vm.$api.gowxLogin = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/wxLogin', params);
30. vm.$api.goAppLogin = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/appLogin', params);
31. vm.$api.getWechatMobile = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.login/getWechatMobile', params);
32. vm.$api.getAuthUrl = async (params = {}) =**>** await vm.$u.get('/addons/third/api/getAuthUrl', params);
33. vm.$api.goAuthCallback = async (params = {}) =**>** await vm.$u.post('/addons/third/api/callback', params);
34. vm.$api.getBindList = async (params = {}) =**>** await vm.$u.get('/addons/third/api/getBindList', params);
35. vm.$api.goUnbind = async (params = {}) =**>** await vm.$u.post('/addons/third/api/unbind', params);
36. vm.$api.goThirdAccount = async (params = {}) =**>** await vm.$u.post('/addons/third/api/account', params);
37. vm.$api.getMoneyLogs = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.the\_logs/money', params);
38. vm.$api.getScoreLogs = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.the\_logs/score', params);
39. vm.$api.selectpage = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.common/selectpage', params);
40. vm.$api.search = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.search/index', params);
41. vm.$api.signinConfig = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/index', params);
42. vm.$api.monthSign = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/monthSign', params);
43. vm.$api.dosign = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/dosign', params);
44. vm.$api.fillup = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/fillup', params);
45. vm.$api.signRank = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/rank', params);
46. vm.$api.signLog = async (params = {}) =**>** await vm.$u.get('/addons/signin/api.index/signLog', params);
47. vm.$api.formField = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.diyform/index', params);
48. vm.$api.postForm = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.diyform/postForm', params);
49. vm.$api.formList = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.diyform/formList', params);
50. vm.$api.formShow = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.diyform/show', params);
51. vm.$api.tagIndex = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.tag/index', params);
52. vm.$api.getCategory = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.common/getCategory', params);
53. vm.$api.getSigned = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.user/getSigned', params);
54. vm.$api.getVipIndex = async (params = {}) =**>** await vm.$u.get('/addons/vip/api.index/index', params);
55. vm.$api.goVipSubmit = async (params = {}) =**>** await vm.$u.get('/addons/vip/api.order/submit', params);
56. vm.$api.getCollection = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.collection/index', params);
57. vm.$api.addCollection = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.collection/create', params);
58. vm.$api.delCollection = async (params = {}) =**>** await vm.$u.post('/addons/cms/api.collection/delete', params);
59. vm.$api.specialList = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.special/special', params);
60. vm.$api.specialIndex = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.special/index', params);
61. vm.$api.getPageDetail = async (params = {}) =**>** await vm.$u.get('/addons/cms/api.page/detail', params);
62. vm.$api.goUpload = async (params = {}) =**>** await upload(vm, params);

### 6.4故障处理

故障检测：

实时监控小程序及其连接的硬件设备、后端服务器的运行状态。

通过日志记录、异常捕获等方式及时发现故障。

故障定位：

提供详细的故障信息和日志记录，帮助管理员快速定位问题所在。

支持远程调试和诊断功能，提高故障处理效率。

故障恢复：

提供自动恢复机制，对于可预见的故障能够自动进行修复。

对于无法自动恢复的故障，应提供清晰的恢复流程和操作指南。

故障预防：

定期进行系统维护和升级，确保小程序及其连接的设备、服务器保持最佳状态。

加强安全管理和培训，提高管理员和用户的安全意识。

## 7．其它需求

本软件开发完成，需要交付给客户的包括文档和安装程序。文档包括：需求规格说明书、设计说明书、测试报告、用户手册。

政府机构网站软件

设计说明书

开发团队：第四组

完成日期：2024.7.18

## 1、引言

### 1.1编写目的

从软件的阶段开发正式进入软件的实际开发阶段，本阶段完成系统的设计并明确系统的数据结构和软件结构。在软件的设计阶段只要是把一个软件的需求转化为软件的表示的过程，这种表示只是描绘出软件的总体概貌。继而进行详细的说明。本说明集合了软件的概要设计也详细设计于一身，进一步说明软件在设计阶段的情况。

### 1.2定义

**本设计说明包含：**

概要设计

接口设计

数据结构设计

数据库设计

界面设计

出错管理

### 1.3参考资料

《软件工程设计指导书》 北京工业大学出版社

《UML与系统分析设计》 张龙祥 人民邮电出版社

## 2、概要设计

### 2.1 需求规定

本系统的主要输入输出项目为用户的个人信息、用户政务业务的办理信息、政务新闻信息、办事指南信息等。处理的主要功能有对数据的管理（用户的注册，修改，找回。业务办理的输入，查询，用户反馈信息），支持图形化界面的操作。详细说明可见《需求分析说明说》。

### 2.2 功能模块描述

#### 2.2.1 用户信息管理

2.2.1.1 用户注册

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 用户注册 | | 子系统名称 | 用户信息管理 |
| 接口说明 | 输入 | | 用户名、密码、手机号、验证码 | |
|
| 输出 | | A 信息填写错误提示  B 注册完毕成功 | |
|
| 功能说明 | 当用户要在系统中注册自己的信息时，通过本模块实现输入个人的基本信息。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 |  | | |
|
| 被调用模块 |  | | |
|

2.2.1.2 用户登录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 用户登录 | | 子系统名称 | 用户信息管理 |
| 接口说明 | 输入 | | 用户身份、账号、密码/用户身份、手机号、验证码 | |
|
| 输出 | | A密码、用户名或手机号、验证码错误提示信息  B登录成功信息。 | |
|
| 功能说明 | 当用户要在系统中登录自己的账号时，通过本模块实现登录。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户注册 | | |
|
| 被调用模块 |  | | |
|

2.2.1.3 用户找回密码

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 用户找回密码 | | 子系统名称 | 用户信息管理 |
| 接口说明 | 输入 | | 邮箱、验证码/手机号、验证码、新密码 | |
|
| 输出 | | A信息填写错误提示  B信息修改成功提示 | |
|
| 功能说明 | 当用户要在系统中登录自己的账号时忘记密码，用户可通过本模块找回并重新设置密码。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户注册 | | |
|
| 被调用模块 |  | | |
|

2.2.1.4 用户退出登录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 用户退出登录 | | 子系统名称 | 用户信息管理 |
| 接口说明 | 输入 | | A鼠标点击退出按钮事件 | |
|
| 输出 | | A 确定退出按钮 B退出成功按钮 | |
|
| 功能说明 | 当用户要从系统退出时，点击按钮退出。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录 | | |
|
| 被调用模块 |  | | |
|

2.2.1.5 用户个人资料查看

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 用户个人资料查看 | | 子系统名称 | 用户信息管理 |
| 接口说明 | 输入 | | A鼠标点击个人资料按钮事件 | |
|
| 输出 | | 头像、用户名、昵称、个人介绍 | |
|
| 功能说明 | 当用户要查看个人资料时，点击个人资料按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录 | | |
|
| 被调用模块 |  | | |
|

#### 2.2.2 业务信息管理

2.2.2.1业务办理-社保登记

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 社保登记（个人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 姓名、出生日期、性别、社保所在地、身份证照片 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当个人用户需要办理社保登记业务时，提交相关个人信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（个人） | | |
|
| 被调用模块 |  | | |
|

2.2.2.2 业务办理-社保挂失

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 社保挂失（个人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 姓名、身份证号、性别、现居住地、电话号码、社保号 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当个人用户需要办理社保挂失业务时，提交相关个人信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（个人） | | |
|
| 被调用模块 |  | | |
|

2.2.2.3 业务办理-开具户籍证明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 开具户籍证明（个人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 姓名、居住地、籍贯、出生日期、身份证、附件材料 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当个人用户需要办理开具户籍证明业务时，提交相关个人信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（个人） | | |
|
| 被调用模块 |  | | |

2.2.2.4 业务办理-企业开办

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 企业开办（法人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 附件材料、企业名称、法人居住地、城市、法人身份证号、企业法人 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当法人用户需要办理企业开办业务时，提交相关企业信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（法人） | | |
|
| 被调用模块 |  | | |

2.2.2.5 业务办理-公章刻制备案

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 公章刻制备案（法人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 企业名称、企业法人、法人身份证号 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当法人用户需要办理公章刻制备案业务时，提交相关企业信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（个人） | | |
|
| 被调用模块 |  | | |

2.2.2.6 业务办理-投诉举报

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 投诉举报（个人、法人用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 标题、图片、投诉内容 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当个人、法人用户需要进行投诉举报时，提交相关投诉信息，进行业务申请。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（个人、法人） | | |
|
| 被调用模块 |  | | |

2.2.2.7 业务办理-政务处理

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 政务处理（政府工作人员用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 标题、图片、处理内容 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当政府工作人员需要进行政务处理时，提交相关处理信息。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（政府工作人员）、业务办理（个人、法人） | | |
|
| 被调用模块 |  | | |

2.2.2.8 业务办理-反馈处理

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 反馈处理（政府工作人员用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 标题、图片、处理内容 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当政府工作人员需要进行反馈处理时，提交相关处理信息。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（政府工作人员）、投诉举报（个人、法人） | | |
|
| 被调用模块 |  | | |

2.2.2.9 业务办理-我发布的文章

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 我发布的文章（政府工作人员用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | A鼠标点击我发布的文章按钮事件 | |
|
| 输出 | | A显示我发布的文章 | |
|
| 功能说明 | 当政府工作人员需要查看我发布的文章时，点击我发布的文章按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（政府工作人员）、发布文章 | | |
|
| 被调用模块 |  | | |

2.2.2.10 业务办理-发布文章

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 发布文章（政府工作人员用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | 栏目、发布内容 | |
|
| 输出 | | A提交成功提示 | |
|
| 功能说明 | 当政府工作人员需要发布相关政策文章时，提交相关内容信息。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录（政府工作人员） | | |
|
| 被调用模块 |  | | |

2.2.2.11 业务办理-我的业务

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 我的业务（个人、法人、政府工作人员用户） | | 子系统名称 | 业务信息管理 |
| 接口说明 | 输入 | | A鼠标点击我的业务按钮事件 | |
|
| 输出 | | A显示我提交的业务 | |
|
| 功能说明 | 当用户需要查看我提交的内容时，点击我的业务按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录 | | |
|
| 被调用模块 |  | | |

#### 2.2.3 客户功能描述

2.2.3.1我发表的评论

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 我发表的评论 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | A鼠标点击我发表的评论按钮事件 | |
|
| 输出 | | A显示我评论的位置与内容 | |
|
| 功能说明 | 当用户需要查看我发表的评论时，点击我发表的评论按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 发表评论、用户登录 | | |
|
| 被调用模块 | 评论数据 | | |
|

2.2.3.2 发表评论

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 发表评论 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | 评论内容、点击提交 | |
|
| 输出 | | 评论成功 | |
|
| 功能说明 | 用户在发表评论处输入评论内容。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 客户功能描述、用户登录 | | |
|
| 被调用模块 | 网页内容 | | |
|

2.2.3.3点赞

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 我发表的评论 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | A鼠标点击点赞按钮事件 | |
|
| 输出 | | A显示点赞成功 | |
|
| 功能说明 | 当用户需要点赞时，点击点赞按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 用户登录 | | |
|
| 被调用模块 | 网页内容 | | |
|

2.2.3.4 收藏

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 收藏 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | A鼠标点击收藏按钮事件 | |
|
| 输出 | | A显示收藏成功 | |
|
| 功能说明 | 当用户需要收藏时，点击收藏按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 客户功能描述、用户登录 | | |
|
| 被调用模块 | 网页内容 | | |

2.2.3.5 我的收藏

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 我的收藏 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | A鼠标点击我的收藏按钮事件 | |
|
| 输出 | | A显示我收藏内容 | |
|
| 功能说明 | 当用户需要查看我的收藏时，点击我的收藏按钮。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 收藏、用户登录 | | |
|
| 被调用模块 | 收藏数据 | | |
|
|

2.2.3.6 站内搜索

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 站内搜索 | | 子系统名称 | 客户功能描述 |
| 接口说明 | 输入 | | 搜索内容 | |
|
| 输出 | | 搜索出来的详细内容 | |
|
| 功能说明 | 当用户需要搜索内容时，在搜索框输入文字后，显示搜索出来的详细内容 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 |  | | |
|
| 被调用模块 | 网页数据 | | |
|
|

#### 2.2.4 系统管理员功能描述

2.2.4.1 登录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 登录 | | 子系统名称 | 系统管理员功能描述 |
| 接口说明 | 输入 | | 用户名，密码 | |
|
| 输出 | | 1. 错误提示 2. 成功提示 | |
|
| 功能说明 | 当用户输入时：  情况1：用户名存在，密码正确，类型正确🡪成功提示  情况2：其他🡪错误提示 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 管理员功能描述 | | |
|
| 被调用模块 | 管理员数据调用 | | |
|

2.2.4.2 查看所有用户信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 查看所有用户信息 | | 子系统名称 | 系统管理员功能描述 |
| 接口说明 | 输入 | | “会员管理”按钮 | |
|
| 输出 | | 所有用户信息 | |
|
| 功能说明 | 管理员点击会员管理按钮，可以看到所有用户信息 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 管理员功能描述 | | |
|
| 被调用模块 | 用户信息 | | |

2.2.4.3 查看所有网页信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 查看所有网页信息 | | 子系统名称 | 系统管理员功能描述 |
| 接口说明 | 输入 | | “栏目管理”按钮 | |
|
| 输出 | | 所有网页信息 | |
|
| 功能说明 | 管理员点击栏目管理按钮，可以看到所有网页信息 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 管理员功能描述 | | |
|
| 被调用模块 | 网页数据调用 | | |
|

2.2.4.4 评论管理

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 评论管理 | | 子系统名称 | 系统管理员功能描述 |
| 接口说明 | 输入 | | 点击“编辑”按钮，对状态进行选择 | |
|
| 输出 | | 无 | |
|
| 功能说明 | 管理员点击编辑按钮，若该评论审核通过，则在状态处选择正常，则能将该评论正常发布在网站中；若该评论审核不通过，则在状态处选择隐藏，则不能显示在网站中。 | | | |
|
|
|
|
|
| 运行环境说明 | 用Hbuilder调用MySQL实现。 | | | |
|
| 调用关系说明 | 调用模块 | 管理员功能描述 | | |
|
| 被调用模块 | 评论数据调用 | | |
|

### 2.3 运行环境

Hbuilder、fastadmin

### 2.4 基本设计概念及处理流程

#### 2.4.1 软件体系结构图



## 3、接口设计

### 3.1 内部接口

用来处理界面与后台数据库操作中数据的传输，数据库的操作由SQL操作完成。

1. **private** string m\_UserId;          // 用户ID
2. **private** string m\_UserName;        // 用户名
3. **private** string m\_Password;        // 密码
4. **private** string m\_UserInfo;        // 用户信息
5. **private** string m\_CorporateName;   // 所属企业名称
6. **private** string m\_Department;      // 所属政府部门
7. **private** string m\_CorporateId;     // 企业ID
8. **private** string m\_EnterpriseName;  // 企业名称
9. **private** string m\_LegalPerson;     // 企业法人
10. **private** string m\_ContactPhone;    // 联系电话
11. **private** string m\_ServiceId;       // 业务ID
12. **private** string m\_ServiceName;     // 业务名称
13. **private** DateTime m\_StartTime;     // 业务开始时间
14. **private** string m\_ServiceInfo;     // 业务信息
15. **private** string m\_ServiceStatus;   // 业务办理状态
16. **private** string m\_ResidenceName;   // 用户姓名（户籍表）
17. **private** **int** m\_Age;                // 年龄
18. **private** string m\_NativePlace;     // 籍贯
19. **private** string m\_Residence;       // 居住地
20. **private** string m\_SocialSecurityNumber; // 社保卡号
21. **private** string m\_Phone;               // 电话
22. **private** string m\_ComplaintId;         // 投诉ID
23. **private** DateTime m\_ComplaintStartTime; // 投诉开始时间
24. **private** string m\_ComplaintInfo;       // 投诉信息
25. **private** string m\_FeedbackStatus;      // 反馈状态
26. **private** string m\_FeedbackInfo;        // 反馈信息
27. **private** string m\_ComplaintType;       // 投诉类型
29. // 检查旧密码是否正确
30. **public** bool CheckOldPassword(string userId, string oldPassword){}
31. // 更新用户的密码
32. **public** bool UpdatePassword(string userId, string newPassword){}
33. // 根据特定条件查找用户的信息
34. **public** UserInfo FindUserInfo(string userId){}
35. // 添加业务记录
36. **public** bool AddBusinessRecord(string userId, BusinessDetails businessDetails){}
37. // 添加投诉反馈
38. **public** bool AddComplaintFeedback(string complaintId, string feedback){}
39. // 获取用户信息
40. **public** UserInfo GetUserInfo(string userId){}
41. // 获取用户的业务列表
42. **public** List<BusinessDetails> GetUserBusiness(string userId){}
43. // 获取用户的反馈信息
44. **public** List<FeedbackDetails> GetUserFeedback(string userId){}
46. **public** function get\_fields\_html()//获取栏目列表
47. **public** function check\_element\_available()//检测元素是否可用
48. **public** function suggestion()//搜索建议
49. **public** function lang()//加载语言包
50. **public** function upload()//上传文件
51. **public** function weigh()//通用排序
52. **public** function wipecache()//清空系统缓存
53. **public** function category()//读取分类数据,联动列表
54. **public** function area()//读取省市区数据,联动列表
55. **public** function icon()//生成后缀图标
56. **public** function selectpage()//查找搜索
57. **public** function index()//后台首页显示
58. **public** function login()//管理员登录
59. **public** function logout()//退出登录
60. **public** function index()//查看
61. **public** function add()//添加
62. **public** function edit($ids = **null**)//编辑
63. **public** function del($ids = "")//删除
64. **public** function multi($ids = "")//批量更新
65. **public** function detail($ids)//详情
66. **public** function edit($ids = **null**)//编辑
67. **public** function roletree()//获取角色权限
68. **public** function get\_template\_list()//获取模板列表
69. **public** function check\_content\_islegal()//检查内容是否包含违禁词
70. **public** function get\_content\_keywords()//获取关键字
71. **public** function get\_fields\_list()//获取字段列表

### 3.2 外部接口

1. vm.$api.getConfig = async (params = {}) => await vm.$u.get('/addons/cms/api.common/init', params);
2. vm.$api.getCaptcha = async (params = {}) => await vm.$u.get('/addons/cms/api.common/captcha', params);
3. vm.$api.getEmsSend = async (params = {}) => await vm.$u.post('/addons/cms/api.ems/send', params);
4. vm.$api.getSmsSend = async (params = {}) => await vm.$u.post('/addons/cms/api.sms/send', params);
5. vm.$api.getArchives = async (params = {}) => await vm.$u.get('/addons/cms/api.archives/index', params);
6. vm.$api.getArchivesDetail = async (params = {}) => await vm.$u.post('/addons/cms/api.archives/detail', params);
7. vm.$api.getArchivesVote = async (params = {}) => await vm.$u.post('/addons/cms/api.archives/vote', params);
8. vm.$api.getArchivesOrder = async (params = {}) => await vm.$u.post('/addons/cms/api.archives/order', params);
9. vm.$api.getChannel = async (params = {}) => await vm.$u.get('/addons/cms/api.archives/get\_channel', params);
10. vm.$api.getChannelFields = async (params = {}) => await vm.$u.get('/addons/cms/api.archives/get\_channel\_fields', params);
11. vm.$api.archivesPost = async (params = {}) => await vm.$u.post('/addons/cms/api.archives/archives\_post', params);
12. vm.$api.myArchives = async (params = {}) => await vm.$u.get('/addons/cms/api.archives/my', params);
13. vm.$api.deleteArchives = async (params = {}) => await vm.$u.post('/addons/cms/api.archives/delete', params);
14. vm.$api.getUserIndex = async (params = {}) => await vm.$u.get('/addons/cms/api.user/index', params);
15. vm.$api.getUserProfile = async (params = {}) => await vm.$u.post('/addons/cms/api.user/profile', params);
16. vm.$api.goUserLogout = async (params = {}) => await vm.$u.get('/addons/cms/api.user/logout', params);
17. vm.$api.goUserAvatar = async (params = {}) => await vm.$u.post('/addons/cms/api.user/avatar', params);
18. vm.$api.getUserInfo = async (params = {}) => await vm.$u.post('/addons/cms/api.user/userInfo', params);
19. vm.$api.getMyComment = async (params = {}) => await vm.$u.get('/addons/cms/api.my/comment', params);
20. vm.$api.getMyboutus = async (params = {}) => await vm.$u.get('/addons/cms/api.my/aboutus', params);
21. vm.$api.getOrder = async (params = {}) => await vm.$u.get('/addons/cms/api.my/order', params);
22. vm.$api.getMyagree = async (params = {}) => await vm.$u.get('/addons/cms/api.my/agreement', params);
23. vm.$api.goCommentPost = async (params = {}) => await vm.$u.post('/addons/cms/api.comment/post', params);
24. vm.$api.goCommentIndex = async (params = {}) => await vm.$u.post('/addons/cms/api.comment/index', params);
25. vm.$api.goLogin = async (params = {}) => await vm.$u.post('/addons/cms/api.login/login', params);
26. vm.$api.mobilelogin = async (params = {}) => await vm.$u.post('/addons/cms/api.login/mobilelogin', params);
27. vm.$api.goRegister = async (params = {}) => await vm.$u.post('/addons/cms/api.login/register', params);
28. vm.$api.goResetpwd = async (params = {}) => await vm.$u.post('/addons/cms/api.login/resetpwd', params);
29. vm.$api.gowxLogin = async (params = {}) => await vm.$u.post('/addons/cms/api.login/wxLogin', params);
30. vm.$api.goAppLogin = async (params = {}) => await vm.$u.post('/addons/cms/api.login/appLogin', params);
31. vm.$api.getWechatMobile = async (params = {}) => await vm.$u.post('/addons/cms/api.login/getWechatMobile', params);
32. vm.$api.getAuthUrl = async (params = {}) => await vm.$u.get('/addons/third/api/getAuthUrl', params);
33. vm.$api.goAuthCallback = async (params = {}) => await vm.$u.post('/addons/third/api/callback', params);
34. vm.$api.getBindList = async (params = {}) => await vm.$u.get('/addons/third/api/getBindList', params);
35. vm.$api.goUnbind = async (params = {}) => await vm.$u.post('/addons/third/api/unbind', params);
36. vm.$api.goThirdAccount = async (params = {}) => await vm.$u.post('/addons/third/api/account', params);
37. vm.$api.getMoneyLogs = async (params = {}) => await vm.$u.get('/addons/cms/api.the\_logs/money', params);
38. vm.$api.getScoreLogs = async (params = {}) => await vm.$u.get('/addons/cms/api.the\_logs/score', params);
39. vm.$api.selectpage = async (params = {}) => await vm.$u.get('/addons/cms/api.common/selectpage', params);
40. vm.$api.search = async (params = {}) => await vm.$u.get('/addons/cms/api.search/index', params);
41. vm.$api.signinConfig = async (params = {}) => await vm.$u.get('/addons/signin/api.index/index', params);
42. vm.$api.monthSign = async (params = {}) => await vm.$u.get('/addons/signin/api.index/monthSign', params);
43. vm.$api.dosign = async (params = {}) => await vm.$u.get('/addons/signin/api.index/dosign', params);
44. vm.$api.fillup = async (params = {}) => await vm.$u.get('/addons/signin/api.index/fillup', params);
45. vm.$api.signRank = async (params = {}) => await vm.$u.get('/addons/signin/api.index/rank', params);
46. vm.$api.signLog = async (params = {}) => await vm.$u.get('/addons/signin/api.index/signLog', params);
47. vm.$api.formField = async (params = {}) => await vm.$u.get('/addons/cms/api.diyform/index', params);
48. vm.$api.postForm = async (params = {}) => await vm.$u.post('/addons/cms/api.diyform/postForm', params);
49. vm.$api.formList = async (params = {}) => await vm.$u.get('/addons/cms/api.diyform/formList', params);
50. vm.$api.formShow = async (params = {}) => await vm.$u.get('/addons/cms/api.diyform/show', params);
51. vm.$api.tagIndex = async (params = {}) => await vm.$u.get('/addons/cms/api.tag/index', params);
52. vm.$api.getCategory = async (params = {}) => await vm.$u.get('/addons/cms/api.common/getCategory', params);
53. vm.$api.getSigned = async (params = {}) => await vm.$u.post('/addons/cms/api.user/getSigned', params);
54. vm.$api.getVipIndex = async (params = {}) => await vm.$u.get('/addons/vip/api.index/index', params);
55. vm.$api.goVipSubmit = async (params = {}) => await vm.$u.get('/addons/vip/api.order/submit', params);
56. vm.$api.getCollection = async (params = {}) => await vm.$u.get('/addons/cms/api.collection/index', params);
57. vm.$api.addCollection = async (params = {}) => await vm.$u.post('/addons/cms/api.collection/create', params);
58. vm.$api.delCollection = async (params = {}) => await vm.$u.post('/addons/cms/api.collection/delete', params);
59. vm.$api.specialList = async (params = {}) => await vm.$u.get('/addons/cms/api.special/special', params);
60. vm.$api.specialIndex = async (params = {}) => await vm.$u.get('/addons/cms/api.special/index', params);
61. vm.$api.getPageDetail = async (params = {}) => await vm.$u.get('/addons/cms/api.page/detail', params);
62. vm.$api.goUpload = async (params = {}) => await upload(vm, params);

## 4、数据库设计

### 4.1 引言

本部分说明“政府机构网站软件”的数据库设计、结构情况，用于开发人员进行项目设计，同时也为后续的数据库维护工作提供了良好的使用说明，也可以作为未来版本升级时的重要参考资料。

### 4.2 支持软件

数据库服务器：MySQL

应用程序端：Navicat

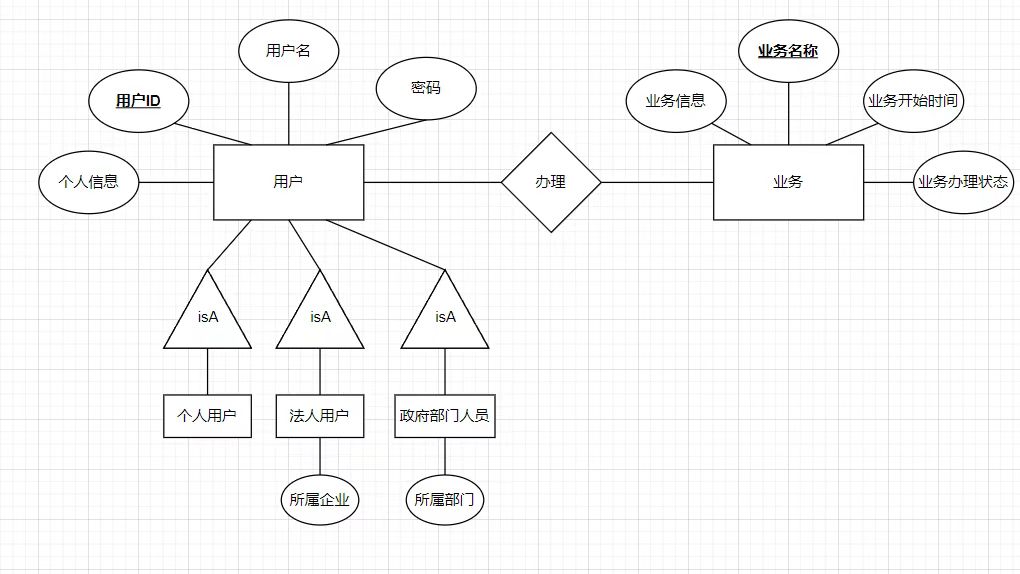
### 4.3 结构设计

#### 4.3.1 概念结构

略。

#### 4.3.2 逻辑结构

4.3.2.1 ER图



4.3.2.2 表的设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 业务 | | | | |
| 字段列名 | 中文名 | 属性 | 主键 | 约束 |
| I\_name | 业务名称 | VARCHAR | √ | NOT NULL, UNIQUE |
| I\_time | 业务开始时间 | DateTime |  | NOT NULL |
| I\_info | 业务信息 | VARCHAR |  |  |
| I\_state | 业务办理状态 | BOOLEAN |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 个人用户 | | | | |
| 字段列名 | 中文名 | 属性 | 主键 | 约束 |
| U\_id | 用户ID | VARCHAR | √ | NOT NULL, UNIQUE |
| U\_info | 个人信息 | VARCHAR |  |  |
| U\_name | 用户名 | VARCHAR |  | NOT NULL |
| U\_pwd | 密码 | VARCHAR |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 法人用户 | | | | |
| 字段列名 | 中文名 | 属性 | 主键 | 约束 |
| U\_id | 用户ID | VARCHAR | √ | NOT NULL, UNIQUE |
| U\_info | 个人信息 | VARCHAR |  |  |
| U\_name | 用户名 | VARCHAR |  | NOT NULL |
| U\_pwd | 密码 | VARCHAR |  | NOT NULL |
| development | 所属企业 | VARCHAR |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 政府部门人员 | | | | |
| 字段列名 | 中文名 | 属性 | 主键 | 约束 |
| U\_id | 用户ID | VARCHAR | √ | NOT NULL, UNIQUE |
| U\_info | 个人信息 | VARCHAR |  |  |
| U\_name | 用户名 | VARCHAR |  | NOT NULL |
| U\_pwd | 密码 | VARCHAR |  | NOT NULL |
| department | 所属部门 | VARCHAR |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 办理 | | | | |
| 字段列名 | 中文名 | 属性 | 外键 | 约束 |
| U\_id | 用户ID | VARCHAR | √ | NOT NULL, UNIQUE |
| I\_name | 业务名称 | VARCHAR | √ | NOT NULL, UNIQUE |

## 5、界面设计

### 5.1 引言

本部分说明“政府机构网站软件”的界面设计情况，用于开发人员进行界面设计。

### 5.2 界面详细设计

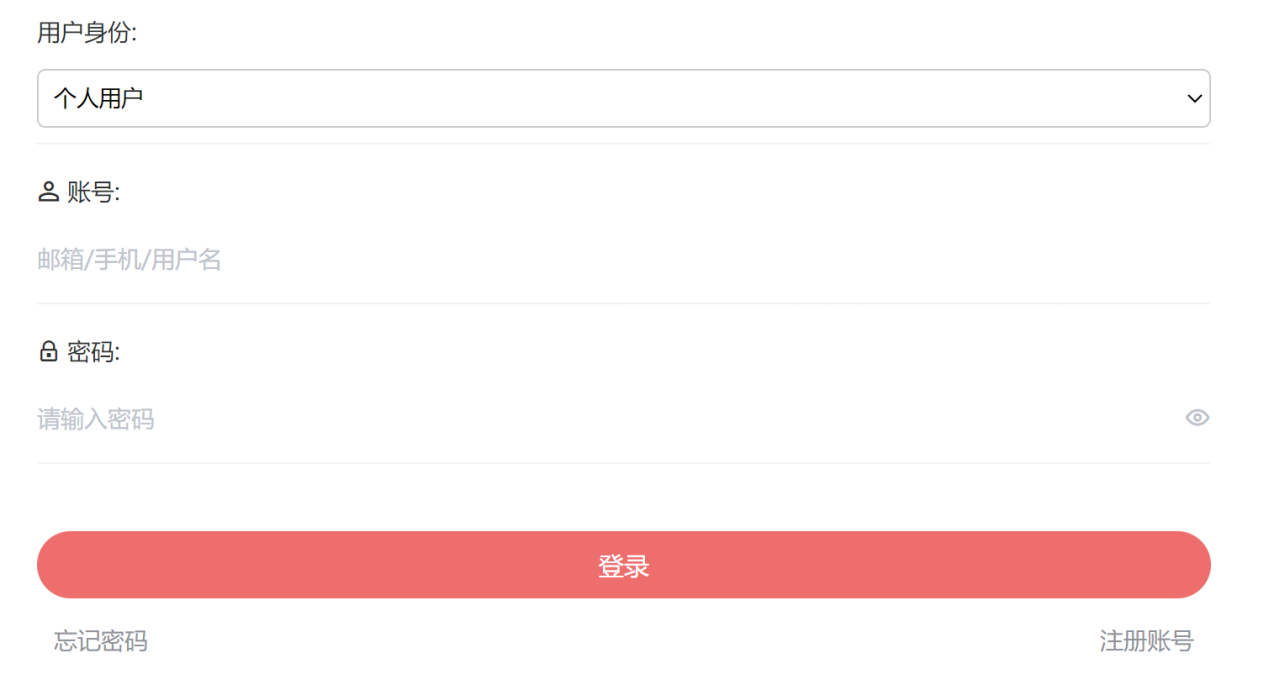
#### 5.2.1 登录界面：

界面视图：

手机号登录：



账号登录：



对象功能及操作方式：略。

#### 5.2.2 注册界面

界面视图：



对象功能及操作方式：略

#### 5.2.3 找回密码界面：

界面视图：



对象功能及操作方式：

菜单栏：功能列表

#### 5.2.4 个人用户操作主界面：

界面视图：



对象功能及操作方式：略

#### 5.2.5 个人社保登记界面：

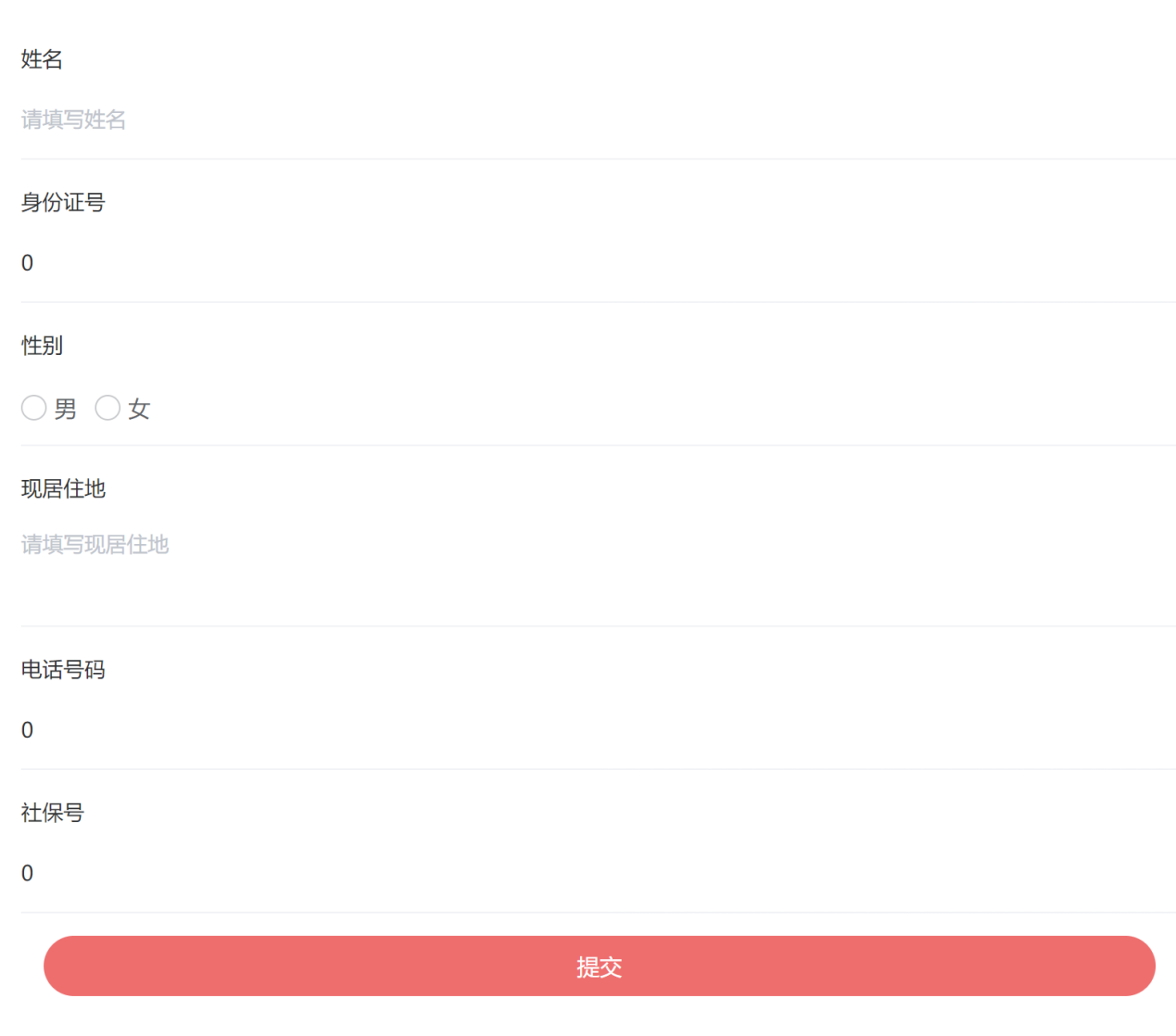
界面视图：



对象功能及操作方式：略

#### 5.2.6 个人社保挂失界面：

界面视图：



对象功能及操作方式：

#### 5.2.7 个人开具户籍证明界面：

界面视图：



对象功能及操作方式：

#### 5.2.8 法人用户操作主界面：

界面视图：



对象功能及操作方式：

#### 5.2.9 法人企业开办界面：

界面视图：



对象功能及操作方式：

#### 5.2.10 法人公章刻制备案界面：

界面视图：



对象功能及操作方式：

#### 5.2.11 我的业务界面：

界面视图：



对象功能及操作方式：

#### 5.2.12 投诉举报界面：

界面视图：



对象功能及操作方式：

#### 5.2.13 个人资料界面

界面视图：



#### 5.2.14 我的业务界面：

界面视图：



对象功能及操作方式：

#### 5.2.15 政府工作人员用户界面：

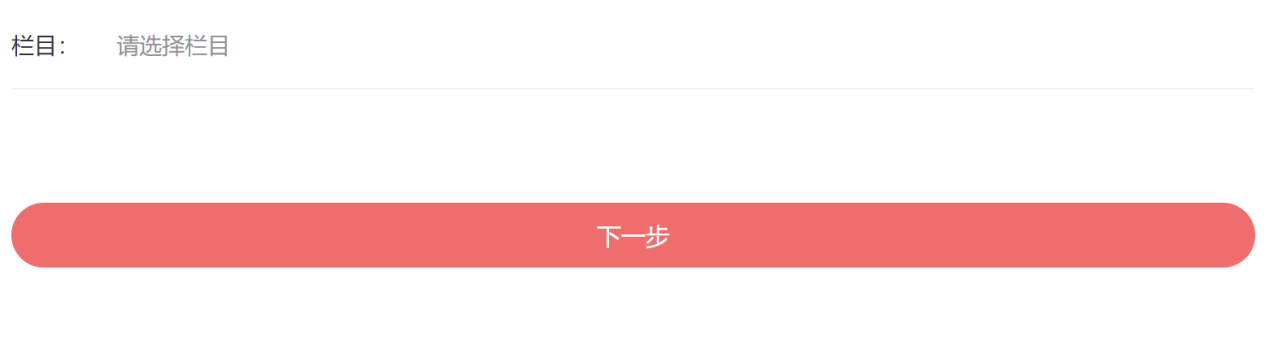
界面视图：



对象功能及操作方式：

#### 5.2.16 政府工作人员发布文章界面

界面视图：





#### 5.2.17 评论界面：

界面视图：





对象功能及操作方式：

#### 5.2.18 我的收藏界面：

界面视图：



对象功能及操作方式：

#### 5.2.19 网页界面







#### 5.2.20 搜索界面



## 6、出错处理

### 6.1 用户输入格式出错处理

E-mail地址格式错误：[符合xxx@xxx.com,xxx@xxx.com.cn,xxx@xxx.cn,xxx.xxx@xxx.com.cn](mailto:符合xxx@xxx.com,xxx@xxx.com.cn,xxx@xxx.cn,xxx.xxx@xxx.com.cn)等正确的地址格式，否则提示输入错误重新输入；

身份证号格式错误：符合中华人民共和国身份证号格式规范，否则提示错误并重新输入；

电话号码格式错误：固定电话及手机位数需符合实际标准，否则提示输入错误并重新输入。

**测试计划**

## 1．引言

### 1.1编写目的

本测试计划重点在于及时发现软件中的漏洞，从而便于程序员进行针对性维护，确保预期功能正常运作。

### 1.2项目背景

项目来源：北京工业大学计算机科学与技术（实验班）210710班第四组

委托单位及主管部门：北京工业大学计算机学院

### 1.3定义

黑盒测试；白盒测试

### 1.4参考资料

软件工程课程设计指导书，北京工业大学 计算机学院

## 2．任务概述

### 2.1目标

软件测试的目标在于使用人工或自动的手段来运行或测定某个软件系统的过程，其目的是在于检验它是否满足规定的需求或弄清预期的结果与实际的结果之间的差别。

### 2.2运行环境

操作系统：Windows 10/ Windows 11

运行环境：任意类型浏览器

### 2.3需求概述

乙方应当在呈交最终产品之前，完成软件测试工作，并针对测试中的漏洞进行修复。

### 2.4条件与限制

条件：已有的文档、测试成员组、测试环境、测试软件。

限制：无。

## 3．计划

### 3.1测试方案

在线政务系统的主要测试内容，基本可以理解为：用户填写业务、用户查看业务。因此测试点也侧重于该二方面。此外，还有政务文章阅读相关的测试，例如：点赞、收藏、评论。

### 3.2测试项目

|  |  |  |  |
| --- | --- | --- | --- |
| 测试内容 | 测试名称 | 目的 | 进度 |
| 评论功能 | 评论测试 | 测试评论内容是否能发表且显示在对应文章中 | 填写、发表、审核、展示 |
| 业务办理 | 业务办理测试 | 测试用户办理的业务是否能提交，在该过程中测试非法输入，提交完成后，业务应当能正确地显示出来 | 填写、上传业务、等待审核、查看 |
| 政务文章发表 | 发文测试 | 测试政务用户发布政策文章是否能提交，在该过程中测试非法输入，提交完成后，文章应当能正确地在主页或对应栏目显示出来 | 填写、发表、查看 |

说明：“业务办理”包括了：社保登记、社保挂失、开具户籍证明、投诉举报、企业开办、政务处理、反馈处理，这些功能的底层逻辑相同，因此共用一套测试，下方测试说明也举其中一例。

### 3.3测试准备

由于该项目呈现形式为web网页，因此测试人员仅需注册账号，并在网页内针对测试内容进行相应测试即可。

### 3.4测试机构及人员

测试机构名称：北京工业大学计算机科学与技术（实验班）210710班第四组

负责人：高立扬

职责：

高立扬：软件测试总负责人

张馨予 马雪：测试组成员

## 4．测试项目说明

### 测试项目：评论测试

### 4.1测试项目名称及测试内容

名称：评论测试

测试内容：判断用户输入合法性，若不合法，禁止发表；若合法，允许发表。发表后，通过系统管理员的审核，评论应当能正确显示在评论区，用户也能自行查看评论记录。

### 4.2测试用例

#### 4.2.1输入

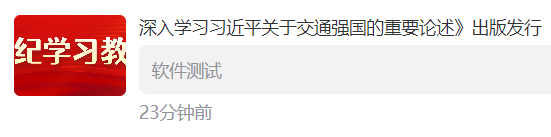
①不输入任何内容，点击立即评论 ②输入一些评论，点击立即评论

#### 4.2.2输出

①评论操作被禁止，因为没有填写评论内容



②评论成功，后台等待审核，审核完毕可以看到评论



#### 4.2.3步骤及操作

①进行测试样例的输入；

②点击“立即评论”按钮；

③若成功发表，审核过后，在文章评论区和“我的评论”可以看到评论。

#### 4.2.4允许偏差

评论发表仅分为成功和不成功，无偏差。

### 4.3进度

填写、发表、审核、展示

### 4.4条件

无特殊条件，仅需在web注册账号即可测试。

### 4.5测试资料

无。

### 测试项目：业务办理测试

### 4.1测试项目名称及测试内容

名称：业务办理测试

测试内容：以开具户籍证明为例，测试用户办理的业务是否能提交，在该过程中测试非法输入，提交完成后，业务应当能正确地显示出来。

### 4.2测试用例

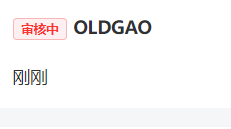
#### 4.2.1输入

①不输入任何必须输入的项目 ②输入必须输入的项目

#### 4.2.2输出

①无法提交

②提交成功，在“我的业务”可以看到对应业务





#### 4.2.3步骤及操作

①点击“开具户籍证明”

②填写相应的数据，点击提交

③提交成功，在“我的业务”可以看到对应业务

#### 4.2.4允许偏差

业务申请上传仅分为成功和不成功，无偏差。

### 4.3进度

填写、上传业务、等待审核、查看

### 4.4条件

无特殊条件，仅需在web注册账号即可测试。

### 4.5测试资料

无。

### 测试项目：发文测试

### 4.1测试项目名称及测试内容

名称：发文测试

测试内容：测试政务用户发布政策文章是否能提交，在该过程中测试非法输入，提交完成后，文章应当能正确地在主页或对应栏目显示出来

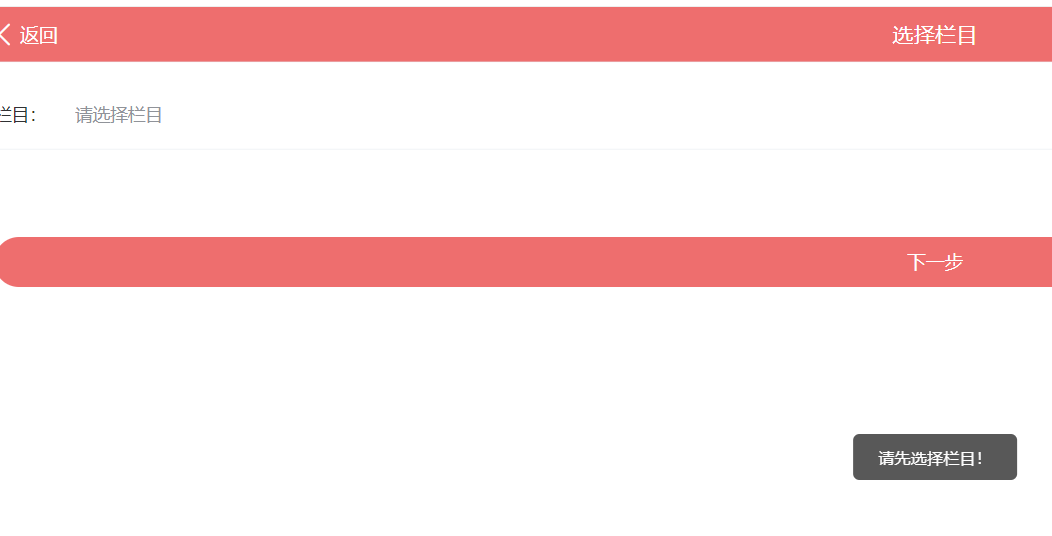
### 4.2测试用例

#### 4.2.1输入

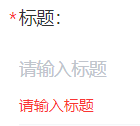
①不选择政务类型 ②选择政务类型，不填写内容 ③选择政务类型，填写内容

#### 4.2.2输出

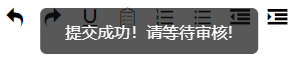
①不允许进行政务内容填写



②不允许提交



③发布后，后台管理员审核后可以在首页或对应栏目看到文章



#### 4.2.3步骤及操作

①选择/不选择政务栏目

②填写/不填写必填内容

③提交文章

④在对应栏目查看文章

#### 4.2.4允许偏差

政务文章发表仅分为成功和不成功，无偏差。

### 4.3进度

填写、发表、查看

### 4.4条件

无特殊条件，仅需在web注册账号即可测试。

### 4.5测试资料

无。

## 5．评价

### 5.1范围

测试完毕，没有发现软件在测试任务上有任何漏洞。

### 5.2准则

评价准则为，测试任务的输出结果是否能按照预期被输出，显示在指定的区域，不会错误地出现或错误地出现在其他区域。

**用户操作手册**

## 1．引言

### 1.1编写目的

该手册为北京工业大学 计算机学院 计算机科学与技术（实验班）软件工程课设小组第四组的 政务系统 专用。目的在于让用户快速上手该软件。

### 1.2项目背景

项目来源：北京工业大学计算机科学与技术（实验班）210710班第四组

委托单位及主管部门：北京工业大学计算机学院

### 1.3定义

无。

### 1.4参考资料

软件工程课程设计指导书，北京工业大学 计算机学院

## 2．软件概述

### 2.1目标

目标在于为用户打造一个线上政务平台，个人用户可以办理社保与户籍相关的业务，还可以进行投诉；法人用户可以开办与企业相关的业务，如：企业开办、企业公章；政务人员可以进行政务审批和发表。用户均可以在该软件中查看实时更新的政务。

### 2.2功能

用户可以查看政务，进行评论收藏点赞；三种类型的用户可以进行对应的业务处理。

### 2.3性能

1. 数据精确度

暂无需要精确度的数据。

1. 时间特性

处理时间可以忽略不计，响应非常快。

1. 灵活性

仅需点击和输入即可，交互界面友好，可以快速上手。

## 3．运行环境

### 3.1硬件

无特殊需求。只需要准备好一台计算机或智能手机、平板等即可。

### 3.2支持软件

支持任意主流类型的浏览器。

## 4.使用说明

### 4.1安装和初始化

无需安装，访问网页即可。因此无需给出后续说明。

### 4.2输入

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.2.1数据背景

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.2.2数据格式

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.2.3输入举例

无。原因在4.1中已说明。软件输入和使用请关注9.。

### 4.3输出

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.3.1数据背景

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.3.2数据格式

无。原因在4.1中已说明。软件输入和使用请关注9.。

#### 4.3.3举例

无。原因在4.1中已说明。软件输入和使用请关注9.。

### 4.4出错和恢复

无。原因在4.1中已说明。软件输入和使用请关注9.。

### 4.5求助查询

无。原因在4.1中已说明。软件输入和使用请关注9.。

## 5.运行说明

### 5.1运行表

用户只需点开网址即可运行该软件，无需任何其他操作。因此下方不再给出运行说明。

### 5.2运行步骤

无。原因在5.1中已说明。

#### 5.2.1运行控制

无。原因在5.1中已说明。

#### 5.2.2操作信息

无。原因在5.1中已说明。

#### 5.2.3输入/输出文件

无。原因在5.1中已说明。

#### 5.2.4启动或恢复过程

无。原因在5.1中已说明。

## 6.非常规过程

若用户输入非法，软件会进行对应的提示，并阻止该操作的进行，直至用户正确操作为止。若用户中断网络连接，则用户正在填写的内容不会保留，其余内容均会保留。

## 7.操作命令一览表

无。

## 8.程序文件（或命令文件）和数据文件一览表

（按照首字母升序）

.eslintignore web组件

.hbuilderx IDE组件

App.vue web主文件

common 存放vue页面的文件夹

components 存放vue页面的文件夹

hybrid 存放vue页面的文件夹

main.js js主脚本

manifest.json 配置文件

node\_modules 存放vue页面的文件夹

package-lock.json 配置文件

package.json 配置文件

pages 存放vue页面的文件夹（主要）

pages.json 配置文件

project.config.json 配置文件

project.private.config.json 配置文件

static 存放vue页面的文件夹

store 存放vue页面的文件夹

template.h5.html html脚本

uni.scss uniapp文件

uniCloud-aliyun uniapp云文件夹

uni\_modules uniapp组件

unpackage uniapp组件

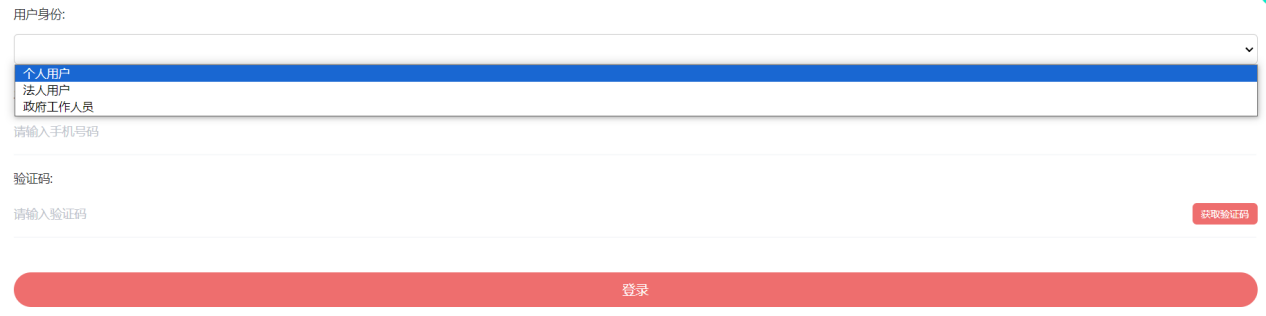
uview-ui uniapp组件

vue.config.js 配置脚本

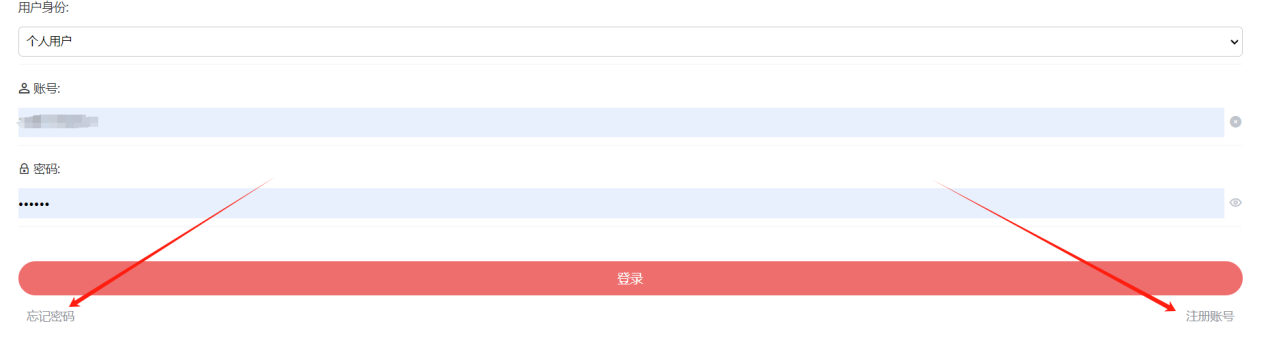
## 用户操作举例

本软件需要登录才可进行一些个人服务。用户一共分为三种：个人、法人、政府人员。每种用户有独特的业务。当然，查看政务和评论收藏，是三种用户都可以进行的。下面将针对每种用户各自的独特功能，选其一进行举例展示，公共功能均会展示。

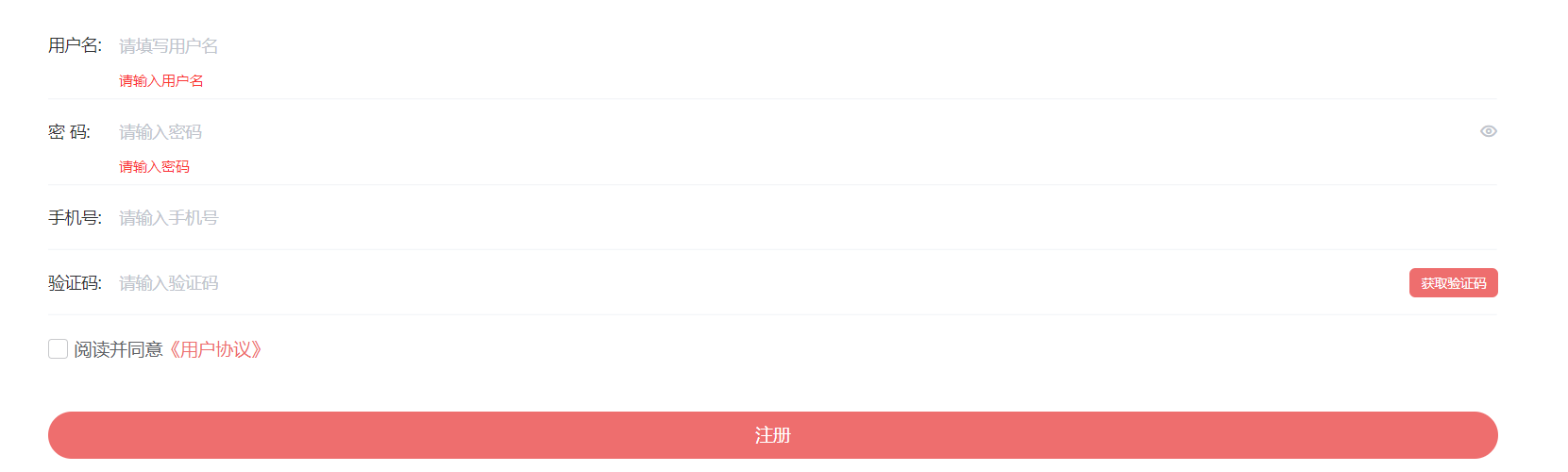
* 1. 注册和登录



首先是手机号快速登录：选择用户类型，填写手机号获取验证码并输入后，即可登录



在快速登录页面的右下角，可以选择账号登录，账号登录的右下角可以进行注册，左下角可以修改账号密码（不再赘述）下方是注册界面，用户填写后即可注册登录。



* 1. 查看文章

任意类型用户登录后，即可查看到主页的文章，点击其中一个即可查阅详细内容。



* 1. 评论

文章的下方有评论区，可以查看到他人的评论，也可以自己发表评论或者回复他人的评论。填写后，点击“立即评论”，等待审核完毕后，即可成功显示。



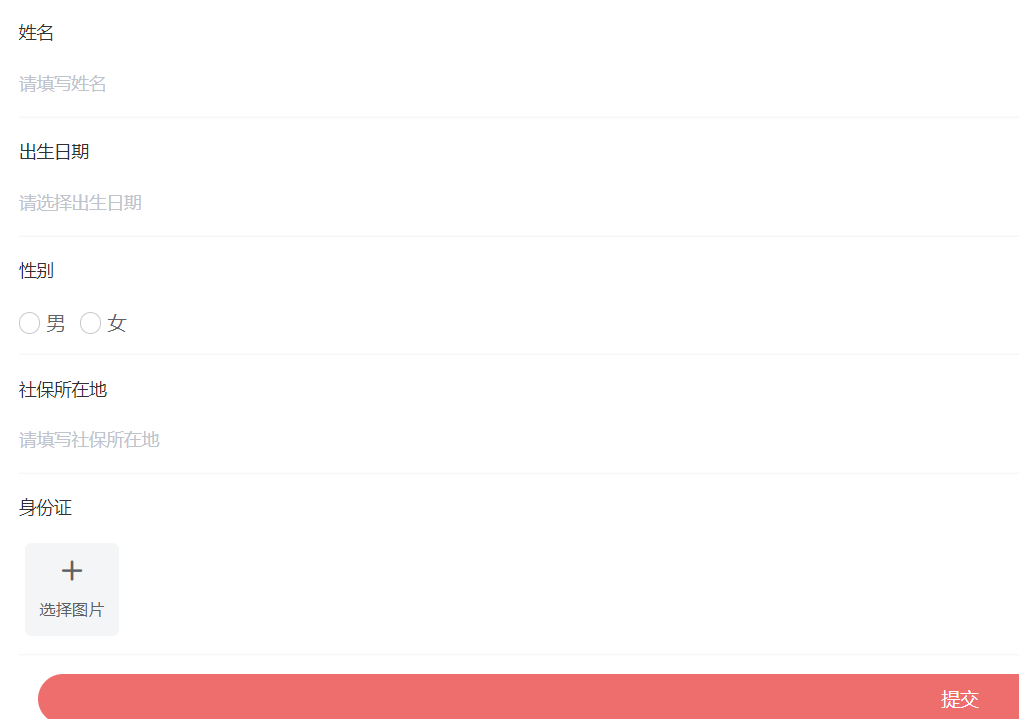
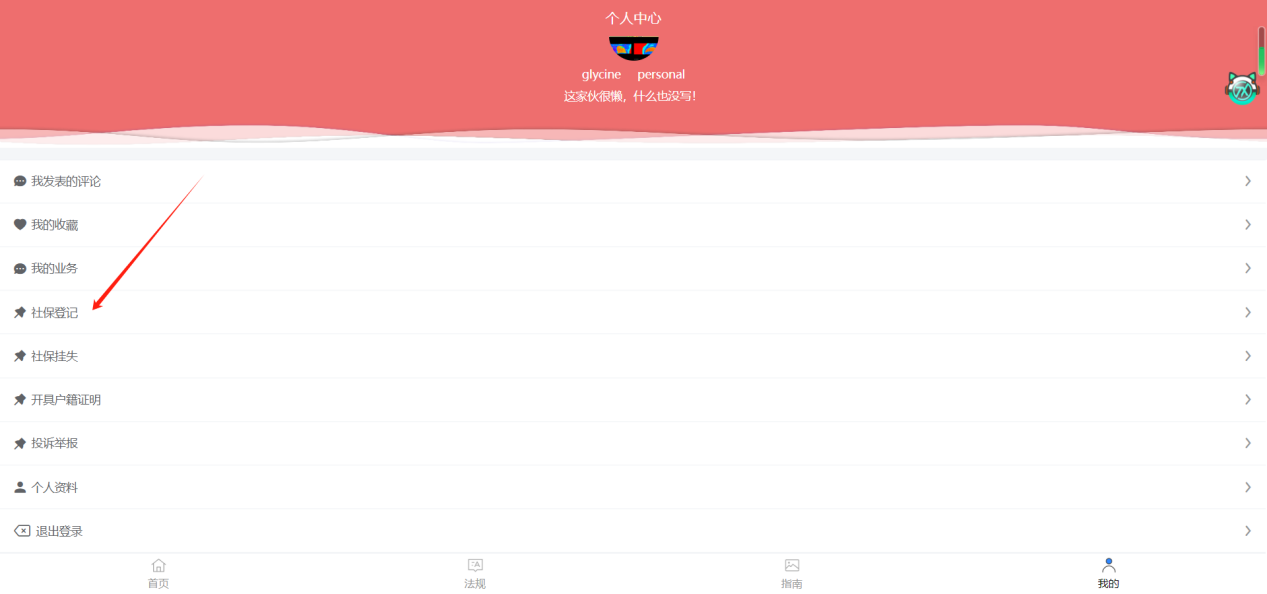
* 1. 收藏

点击收藏，可以在“我的”-“我的收藏”看到对应文章



* 1. 【个人用户】社保登记

“我的”-“社保登记”

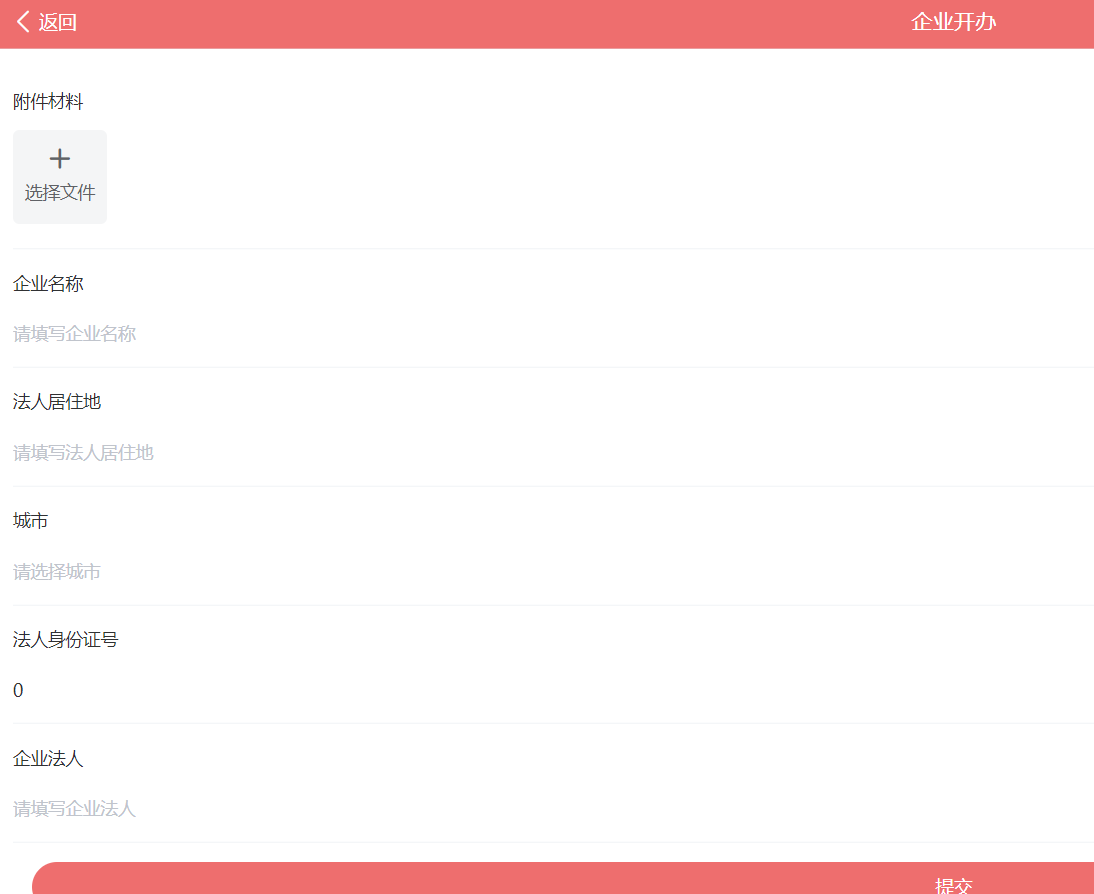
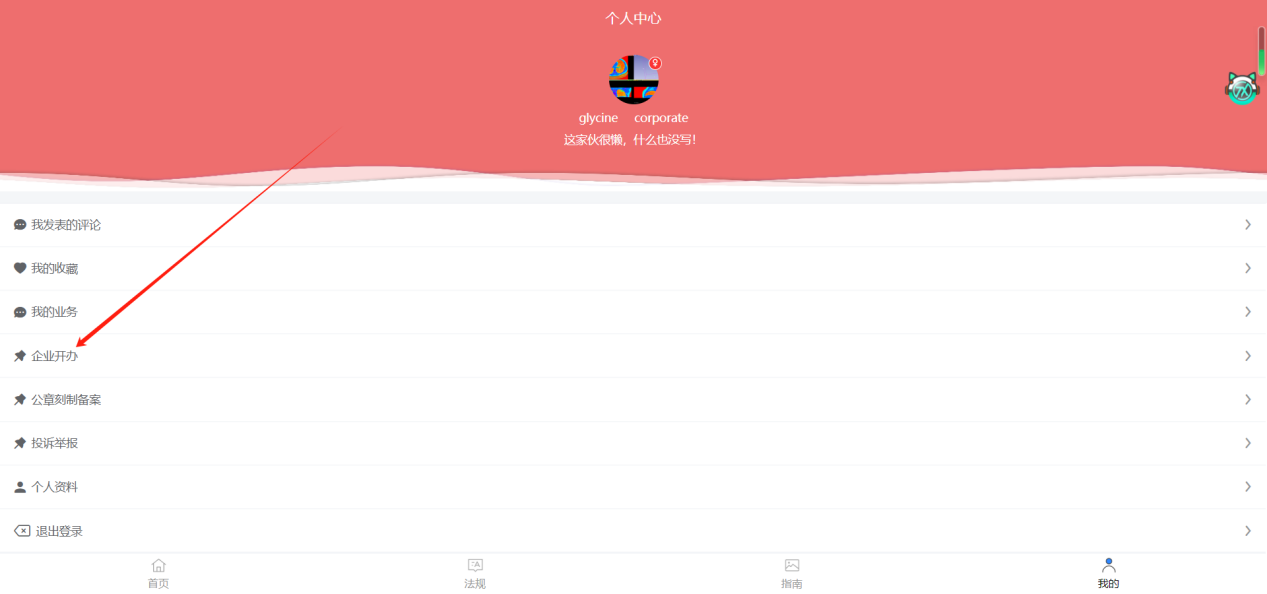


填写好之后提交，即可在“我的”-“我的业务”中查看业务审核状态，点击可以查看详情

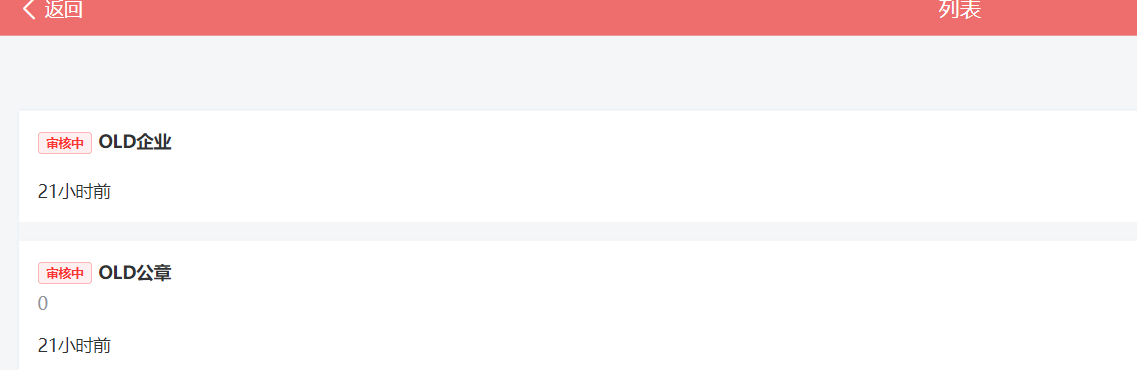


* 1. 【法人用户】企业开办

“我的”-“企业开办”

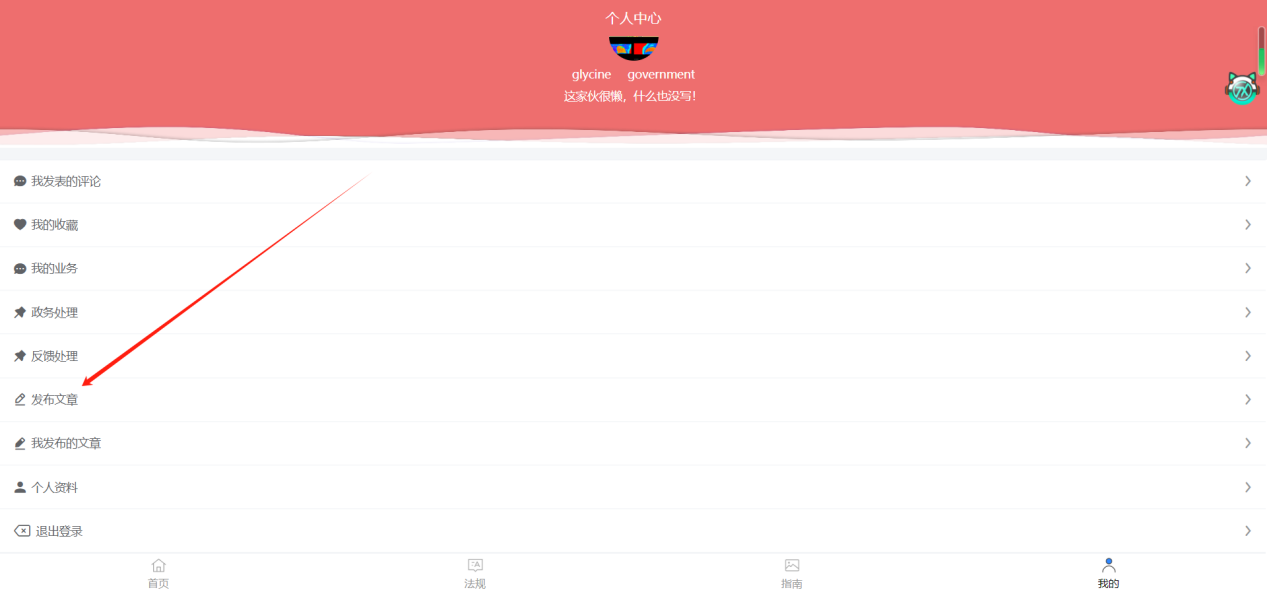


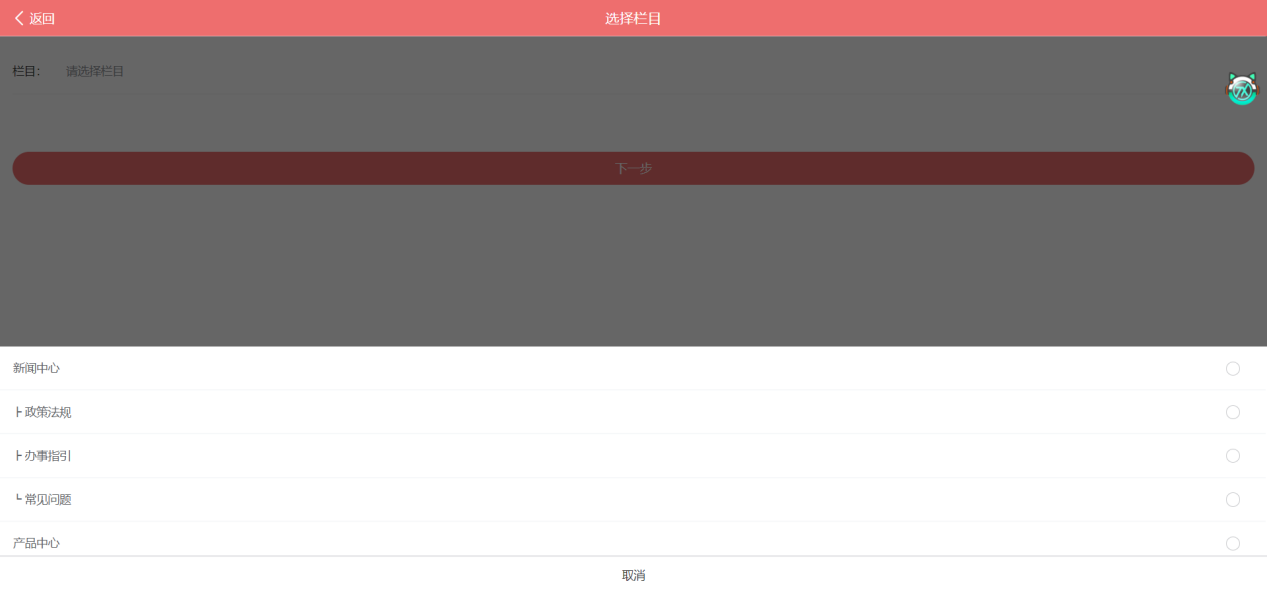
填写好之后提交，即可在“我的”-“我的业务”中查看业务审核状态，点击可以查看详情



* 1. 【政务人员】发布文章

“我的”-“发布文章”





填写后发表，经过系统管理员的审批后，即可在对应栏目看到发表的政务文章。

# 项目开发记录

项目名称： 政务网站

开发人员：高立扬、马雪、张馨予

日 期： 2024/7/19

## 第一次会议：

**日期：**2024.3.7

**出席人员：**组内全体人员

**主题：**项目讨论，分工确定及课题选择

**讨论纪要：**分析了本次课设的目标，确定组员分工及各自侧重任务，完成课题选择

**结论：**

1、分工完成：组长，文档管理员，两辅助程序员（功能实现+界面设计+数据库设计）；

2、课设需解决的问题：工具选择+合作分工+进度跟踪+需求分析+设计+演示；

3、需要的开发文档：开发计划+需求规格说明书+软件设计说明书+测试计划+用户手册+项目开发总结+程序源代码+小组开发记录本+个人总结

4、答辩要求：出勤+演示+答辩。

5、题目确定：政务服务网站

6、明确项目开发过程：确定目标及初步方案--选择、准备开发平台及工具（开发语言+工具）--项目计划--需求分析--概要设计--详细设计--测试--完善+整理--演示+答辩

**问题：**开发工具及语言

**计划：**

1. 下周二之前完成开发工具及语言的确定并每人开始自主学习相关知识；
2. 完成第一次需求分析。
3. 每人认真分析课设指导书，详细了解课设要求及所做题目的要求。

**分工：**高立扬：数据库设计

马 雪：功能业务设计

张馨予：数据流图设计

## 第二次会议：

**日期：**2024-3-22

**出席人员：**组内全体人员

**主题：** 工具确定及任务安排

**讨论纪要：** 确定了所使用的主要工具及语言，明确每人下周的任务

**结论：**

1. 开发平台：hbuilderx
2. 开发语言：HTML,Javascript
3. 数据库平台：Mysql
4. 个人任务：文档管理员认真学习计算机软件文件规范；

其他成员学习工具的使用及语言，同时认真研究所选题目；

组长制定详细的项目计划；

**问题：**大家对软件及语言的使用需在设计开始之前熟悉。

**计划：**

1. 每人安装需要的软件并学习使用；
2. 组长做出项目计划，下周一大家讨论确定；

**分工：**共同完善第二次需求分析

## 第三次会议：

**日期：** 2024-5-13

**出席人员：** 全体

**主题：** 项目计划的确定

**讨论纪要：** 明确项目开发步骤以及人员分工

**结论：**

1、完成需求分析，并规范需求规格说明书

实现概要设计，并规范概要设计说明书

完成详细设计，并规范详细设计说明书

整理开发过程的所有数据、文档及程序，完成交付任务

2、人员分工：

高立扬：文档规范化书写；

马 雪：需求规格说明书及部分概要设计

张馨予：详细设计说明书及部分概要设计

**问题：**按照需求分析拆分完成文档编写

**分工：**

|  |  |
| --- | --- |
| 人员 | 任务 |
| 高立扬 | 角色分工+工具选择+接口设计+后端数据库设计+整体测试 |
| 马雪 | 需求调查+需求规格设计+模块设计+前端业务设计+文档整理 |
| 张馨予 | 需求调查+总体设计+数据设计+前端界面设计+文档整理 |

## 第四次会议：

**日期：** 2024-5-17

**出席人员：** 全体

**主题：** 需求分析讨论1

**讨论纪要：** 明确需求分析的内容、步骤及规范化需要，包括用况图、需求规格说明书的编写规范等，并进一步明确个人任务。

**结论：**

1.需求过程：获取🡪分析🡪确定🡪完善文档

2需求需要从用户及开发者两个角度考虑；

3需求包括功能需求及非功能需求

4用况图及用况描述

5.文档的规范化需要

6.个人任务的确定，详见分工说明

**问题：**

**计划：** 分组完成不同角色的需求分析，包括用户及系统管理员；每人需了解规范化需求分析的需要，最终完成说明文档。

**分工：**

用户需求分析： 马雪、张馨予；

系统管理员需求分析： 高立扬；

文档规范化管理： 马雪、张馨予。

## 第五次会议：

**日期：** 2024/5/24

**出席人员：** 全体

**主题：** 设计概述

**讨论纪要：** 本次会议主要的内容有：

1. 整体的设计任务（包括概要设计和详细设计）需要完成的内容；
2. 概要设计的内容；
3. 详细设计的内容；
4. 设计中的分工；

**结论：**

1. 设计任务正式进入软件的开发阶段，是编码的根本基础，故设计环节需认真做好概要设计和详细设计；
2. 概要设计需要完成软件的体系结构的设计；
3. 详细设计需要完成软件的数据结构设计、功能模块设计、界面设计、数据库设计以及接口设计；

**问题：** 设计文档的规范化书写，每部分所应包含的内容；

**计划：** 在需求的基础上，根据需要实现的功能的要求，逐步分析软件的结构、需要的数据以及功能的实现。

**分工：** 组长完成概要设计；其余成员按照各自需求的分析内容，完成自己的详细设计部分的内容。

## 第六次会议：

**日期：** 2024/6/27

**出席人员：** 全体

**主题：** 设计讨论

**讨论纪要：**

1. 确定了设计文档的书写规范；
2. 确定了设计过程中每个部分的内容；
3. 确定了人员分工；
4. 确定了设计的时间安排；

**结论：**

1. 设计文档包括的内容：引言、概要设计、接口设计、模块设计、数据结构设计、数据库设计、界面设计、出错处理；其中，概要设计主要包括功能描述及体系结构设计；
2. 每人根据各自负责的内容，完成设计工作的每个步骤；

**问题：** 前端业务功能不完善

数据设计有差错

数据未进行后续管理

**计划：**纠正问题，修改现有设计

## 第七次会议：

**日期：** 2024-7-2

**出席人员：** 全体

**主题：** 设计部分总结及编码安排

**讨论纪要：**

1. 完成了设计内容的修改、整合，实现所有成员对软件结构的清晰认识；
2. 确定编码工作的时间安排及人员分工；

**结论：**

1. 设计内容详见软件设计说明书；
2. 编码工作安排确定；

**问题：**

**计划：**

1、1周内完成编码；

2、原型调试，发现问题并解决；

**分工：**

高立扬：后端数据库修改、前后端交互

马 雪：个人、法人业务流程完善

张馨予：用户区分、政策发布功能代码

## 第八次会议：

**日期：** 2024-7-10

**出席人员：** 全体

**主题：** 设计报告的修改及原型开发

**讨论纪要：**

1. 完善设计报告的内容
2. 原型开发讨论
3. 测试工作的初步安排

**结论：**

1. 设计报告中有关详细设计的内容需要修改，其中接口设计、模块设及数据库设计的文档需要规范化；参考网上资料进行修改；具体安排见分工。
2. 模块设计内容需要修改；
3. 测试工作的初步安排：高立扬制定测试计划，所有成员完成测试任务。

**问题：**前后端交互不清晰

**计划：** 检查前完善设计报告并完成软件原型。

**分工：** 各自完善自己在设计工作中的报告及编码任务。

## 第九次会议：

**日期：** 2024-7-16

**出席人员：** 全体

**主题：** 原型完善和后续文档任务

**讨论纪要：** 讨论每人的文档任务

**结论：**

1. 原型已修改完善。
2. 组长需完成用户手册，测试计划及分析报告；
3. 所有成员需完成各自分工中的文档，包括需求说明书、设计说明书、开发计划说明、项目开发记录报告；
4. 所有成员需完成个人报告；
5. 自选项目个人记录。

**问题：**生成非本地网页

**计划：** 完成所有文档并整理、所有运行环境及代码文件，准备最终检查

**分工：**

|  |  |  |  |
| --- | --- | --- | --- |
| 文档\人员 | 高立扬 | 马 雪 | 张馨予 |
| 项目开发计划书 |  |  | √ |
| 需求规格说明书 |  | √ |  |
| 软件设计说明书 |  |  | √ |
| 测试计划 | √ |  |  |
| 用户手册 | √ |  |  |
| 项目开发记录表 |  | √ |  |
| 项目开发总结 | √ | √ | √ |

# 项目开发总结

软件工程课设 个人项目开发总结 21071003高立扬

本次软件工程课设，我担任第四组组长，负责整个项目的开发工作，后端程序与数据库建立，前后端的部分交互工作，系统测试与调试和小组文档核查工作。

其实真正做课设的时间非常短暂，在结束了软工的考试后，我立刻联系了南开大学程明明组进行了为期14天的线上二面考核，期间我还参加了北京理工大学医学技术学院的夏令营，和上海科技大学生物医学工程学院的夏令营，没有时间顾及太多软工课设的编写。虽然在此期间，我想了一些方案，搜寻了不少资料，奈何小组实力有限，并没有做好短时间内写好代码的措施，这也是我个人的失误。

结束了南开二面代码编写，从上海返回北京后，连续将近二十天的高强度生活已经让我筋疲力尽，然而看着进度几乎为零的课设，我明白自己还不是休息的时候。7月12日，在回京的高铁上，我就调动小组开始寻找开源框架，学习vue和后端开源fastadmin的使用，于2日内迅速搭建起了web页，后端在fastadmin框架上快速建立了数据库。在7月15日检查后，一边准备南开二面答辩、北工大夏令营面试、北师大AI学院夏令营，我一边调动组员基本完成了代码，撰写好了报告，尽最大的能力呈交出一份满意的课设作业。

我承认，在软工课设这门课上，我确实没有付出太多的努力，但是回首这将近一个月紧绷的神经，我认为一切都是值得的，我抗住了前所未有的压力，挑起了唯一一个三人小组的重担，尽最大能力安抚焦虑的组员。虽然没有完全实现预期的功能，但是我认为，我付出了我能付出的努力，我不后悔。

北理和上科大的夏令营，我也顺利拿到了优营，还被上科大BME的副院长主动联系，我觉得一个月苦都是值得的，这期间有崩溃，有无助，幸好我坚持住了，收获了宝贵的人生财富。

## 项目开发总结

姓名：19071007 马雪

日期： 2024/07/19

### 目标任务：

在本次的项目开发过程中 我主要负责：前期调研、部分代码编写、文档编写等任务。具体分工在开发记录中有所描述。

### **完成任务：**

我完成了前期调研工作，调研了北京市政务网站、重庆市政务网站、国家政务服务平台，确定了用户类型和具体功能，并从中挑选了一部分具有典型意义和交互流程的业务进行分析。

个人用户：社会保障、户籍管理、婚姻登记（后期未完成）

法人用户：企业开办、公章刻制备案

政府人员：政策发布、业务审批

个人和法人都具有投诉举报的功能，由政府人员进行反馈处理。

完成了前期文档编写，根据小组共同完成的需求分析报告，整理需求概要说明书、编写业务用例流程、数据管理、预计页面设计等工作。并补充完成了概要设计。

完成了分工内的代码编写，主要实现功能是前端的页面设计、业务显示，把前端业务管理内容存储到后端数据库。

完成了后期文档整理，项目开发记录跟踪，整理开会纪要并输出。

### **问题与改进：**

在前期设计时，对数据库的知识有些遗忘，所以数据管理设计有些不合理，前期业务挑选过多，忽略了开发时间和开发人员的限制。

代码编写时，遇到了一个很大的问题就是前后端交互的应用，和组内成员的进度不一致导致问题搁浅了很久。

* 1. 数据库设计优化

复习与回顾：重新复习了数据库设计的基础知识，特别是关系型数据库的设计原则、索引优化、表关联等关键内容。

重构设计：基于项目实际需求，重新评估并简化了数据库表结构，减少了不必要的冗余字段，优化了表之间的关联关系，确保数据的一致性和查询效率。

咨询与讨论：与团队成员进行深入讨论，借鉴其经验，共同完善数据库设计方案。

* 1. 前期业务挑选调整

优先级排序：根据用户需求和业务重要性，对挑选的业务进行优先级排序，优先开发核心和高频使用的功能。

迭代开发：将项目拆分为多个迭代周期，每个周期集中完成一部分业务功能，确保按时交付。

风险管理：加强风险管理，提前识别可能的风险点，制定应对措施，确保项目顺利推进。

### **总结**

在软件项目开发的旅程中，我深刻体会到了从需求分析、设计规划、编码实现到测试上线每一个环节的重要性和复杂性。这一过程不仅是对技术能力的考验，更是对项目管理、团队协作以及创新思维的一次全面锻炼。用户体验是软件产品成功的关键因素之一。我们需要更加关注用户的需求和反馈，不断优化产品的功能和界面设计，提升用户的使用体验和满意度。项目需求概述和软件设计说明也很重要，它们构成了项目成功的基石，对项目的顺利进行和最终成果的质量有着至关重要的影响。整个软件开发是一个漫长的过程，需要度过一道道的关卡。不仅仅是研发，从产品和测试的角度也是有很多内容要完成，这次的课设让我对软件项目设计开发有了更深刻的认识，也让我有了更清晰的工作思路能应对未来的挑战。

**项目开发总结**

姓名：21071026 张馨予

日期： 2024/07/19

1. **目标任务：**

在本次的项目开发过程中 我主要负责：前期调研、前端部分代码编写、文档编写、测试等任务。具体分工在开发记录中有所描述。

1. **完成任务：**

我完成了前期调研工作，调研了北京市政务网站、重庆市政务网站、国家政务服务平台的网页形式、业务类型与办理流程等，确定了用户类型和具体功能，并从中挑选了一部分具有典型意义和交互流程的业务进行分析与完成。

个人用户：社会保障、户籍管理、婚姻登记（后期未完成）

法人用户：企业开办、公章刻制备案

政府人员：政策发布、业务审批

个人和法人都具有投诉举报的功能，由政府人员进行反馈处理。

完成了前期的项目调研与前期文档编写，根据小组讨论出的设计思路，完成了总体框架的设计并撰写了概要设计与详细设计的文档、编写业务用例流程、辅助设计数据库、画用例图等。

完成了分工内的代码编写，主要实现功能是前端的页面设计、用户的相关操作与网站网页的页面设计，把前端内容存储到后端数据库。

完成了后期文档的撰写与整理，撰写了项目开发计划书和软件设计说明书。

1. **问题与改进：**

问题：

在前期的设计时，我们一开始设定的用户种类和实现的业务较多，没有考虑到开发时间和开发人员的限制，并且用户功能之间有些重复冗余，业务的分类也不是非常清晰。

之后在对数据库的设计上，由于对一些数据库知识的遗忘导致数据的设计有些不合理，一些一对多关系的冗余和应该存储哪些数据没有考虑清晰。

在后期的代码编写时，比较大的问题是前后端交互的应用，且组内成员的沟通不及时，进度不一致，导致开发延缓。

改进：

（1）重新优化设计

我们基于项目的开发需求，在原有的设计基础上进行了优化与删减，合并了一些功能相同的用户，例如从一开始的不同业务分别设计了政务人员来审批到最后合成一类政务人员用户。此外我们还将提交相似内容的业务种类进行删减，做到合并相似性，展现特殊性。

（2）数据库的回顾与改进

我们一同重新复习了数据库设计的基础知识，特别是关系型数据库的设计原则、索引优化、表关联等关键内容。基于项目需求，重新优化了数据库表结构，减少了不必要的冗余字段，优化了表之间的关联关系，确保数据的一致性。

（3）开发策略

我们加强了小组之间的讨论与合作，将讨论的频次提升，并将项目拆分成多个不同的模块与多个迭代周期，每个模块与周期集中完成一部分业务功能，确保按时交付。

1. **总结**

本次软件工程课程设计我收获颇多，我深入了解到了从需求分析、设计规划、编码实现到测试上线这一系列环节的关键性与复杂性。整个过程不仅仅锻炼了我们的代码编写能力，同时也对我们的项目管理、团队合作能力进行了提升。从一开始总体设计框架的不全面、不合理，到最后能够呈现出一个相对合理完整的软件，这个过程很漫长、也很不易。我们需要站在各个用户的不同角度，用软件工程的思路思考与优化项目，从实用性、便利性、合理性等角度出发思考问题。不仅是开发工作，测试与完善产品也是非常重要的，它能够帮助我们更加快速、显现化的反应问题所在，指导我们修改的方向。这次课设让我对软件项目设计与开发有了更深刻的认识，也将课程的理论知识应用到了实际开发项目中，对我未来的软件开发提供了清晰的工作思路与宝贵的经验。