**目 录**

[**1 绪 论** 1](#_Toc38959182)

[1.1 研究的背景及意义 1](#_Toc38959183)

[1.1.1 选题的背景 1](#_Toc38959184)

[1.1.2 国内外研究现状 1](#_Toc38959185)

[1.1.3 研究的意义 2](#_Toc38959186)

[1.2 系统目标 3](#_Toc38959187)

[**2 功能需求** 4](#_Toc38959188)

[2.1 业务需求 4](#_Toc38959189)

[2.1.1 企业信息管理业务需求 4](#_Toc38959190)

[2.1.2 招商引资档案管理业务需求 5](#_Toc38959191)

[2.1.3 固定资产投资评估管理业务需求 7](#_Toc38959192)

[2.1.4 企业信息填报业务需求 9](#_Toc38959193)

[2.1.5 领导审阅统计管理业务需求 11](#_Toc38959194)

[2.2 功能需求 12](#_Toc38959195)

[2.2.1 角色分析 12](#_Toc38959196)

[2.2.2 功能描述 13](#_Toc38959197)

[2.2.3 企业信息填报模块功能分析 15](#_Toc38959198)

[2.2.4 企业信息管理模块功能分析 19](#_Toc38959199)

[2.2.5 固定资产评估信息管理模块功能分析 22](#_Toc38959200)

[2.2.6 招商引资档案管理模块详细功能分析 26](#_Toc38959201)

[2.2.7 领导审阅统计管理模块详细功能分析 28](#_Toc38959202)

[2.3 非功能需求 31](#_Toc38959203)

[2.3.1 环境需求 31](#_Toc38959204)

[2.3.2 性能需求 31](#_Toc38959205)

[2.3.3 安全需求 31](#_Toc38959206)

[**3 总体设计** 32](#_Toc38959207)

[3.1 系统设计的原则 32](#_Toc38959208)

[3.2 系统体系结构设计 32](#_Toc38959209)

[3.3 技术架构设计 33](#_Toc38959210)

[3.4 应用架构设计 34](#_Toc38959211)

[3.5 系统功能结构设计 35](#_Toc38959212)

[3.5.1 企业信息管理子系统功能结构设计 36](#_Toc38959213)

[3.5.2 企业信息填报子系统功能结构设计 37](#_Toc38959214)

[3.5.3 招商引资档案管理子系统功能结构设计 37](#_Toc38959215)

[3.5.4 固定资产投资评估管理子系统功能结构设计 38](#_Toc38959216)

[3.5.5 领导审阅统计子系统功能结构设计 39](#_Toc38959217)

[**4 数据库设计** 40](#_Toc38959218)

[4.1 概念结构设计 40](#_Toc38959219)

[4.1.1 设计思路 40](#_Toc38959220)

[4.1.2 实体属性设计 41](#_Toc38959221)

[4.2 逻辑结构设计 44](#_Toc38959222)

[4.2.1 设计思路 44](#_Toc38959223)

[4.2.2 逻辑模型 45](#_Toc38959224)

[4.3 物理结构设计 48](#_Toc38959225)

[4.3.1 存取方式 48](#_Toc38959226)

[4.3.2 DDL语言设计 49](#_Toc38959227)

[**5 界面设计** 53](#_Toc38959228)

[5.1 界面关系图 53](#_Toc38959229)

[5.2 界面设计成果 55](#_Toc38959230)

[5.2.1 主界面 55](#_Toc38959231)

[5.2.2 子界面 56](#_Toc38959232)

[**6 详细设计** 61](#_Toc38959233)

[6.1 系统主要功能模块介绍 61](#_Toc38959234)

[6.2 功能模块设计 61](#_Toc38959235)

[6.2.1 企业基本信息管理 61](#_Toc38959236)

[6.2.2 招商引资档案管理 68](#_Toc38959237)

[6.2.3 固定资产评估信息管理 74](#_Toc38959238)

[6.2.4 领导审阅统计管理 81](#_Toc38959239)

[6.2.5 企业信息填报系统 86](#_Toc38959240)

[**7 编码** 94](#_Toc38959241)

[7.1 代码实现与核心算法 94](#_Toc38959242)

[7.1.1 企业信息填报系统 94](#_Toc38959243)

[7.1.2 招商引资档案管理系统 96](#_Toc38959244)

[7.1.3 固定资产评估管理系统 97](#_Toc38959245)

[7.1.4 领导审阅统计管理系统 98](#_Toc38959246)

[7.1.5 企业基本信息管理系统 99](#_Toc38959247)

[7.2 代码优化分析 99](#_Toc38959248)

[**8 测试** 100](#_Toc38959249)

[8.1 测试方案设计 100](#_Toc38959250)

[8.1.1 测试策略 100](#_Toc38959251)

[8.1.2 测试资源 100](#_Toc38959252)

[8.1.3 关键测试点 100](#_Toc38959253)

[8.2 测试用例构建 101](#_Toc38959254)

[8.2.1 测试用例设计 101](#_Toc38959255)

[8.2.2 关键测试用例 101](#_Toc38959256)

[8.2.3 测试用例维护 104](#_Toc38959257)

[**9 总结与展望** 105](#_Toc38959258)

[9.1 设计工作总结 105](#_Toc38959259)

[9.2 未来工作展望 105](#_Toc38959260)

[**谢 辞** 106](#_Toc38959261)

[**参考文献** 107](#_Toc38959262)

[**附录A 外文翻译—原文部分** 109](#_Toc38959263)

[**附录B 外文翻译—译文部分** 113](#_Toc38959264)

[**附录C 软件使用说明书** 117](#_Toc38959265)

[**附录D 主要源代码** 118](#_Toc38959266)

1 绪 论

## 1.1 研究的背景及意义

### 1.1.1 选题的背景

高新技术产业开发区是我国在改革开放后建立在中大型城市和沿海地区的高新技术发展产业的开发区，是各级政府批准建立的科技工业园区，依托于智力、技术、环境、经济的大型综合平台。而我国日益发展的条件下，高新区企业不断扩张，同时也给高新区信息管理带来了不小的压力，早期的纸质办公和信息储备方式已然不再适用于如今的办公系统，因此在计算机信息技术不断快速发展的同时，使用计算机管理系统的对信息进行管理已然成为一件势在必行的工作。

高新区企业政策管理服务综合平台是为了管理企业信息，政策信息，审批信息，固定资产评估信息等信息而设计的。高新区作为企业和技术资源的中心，其囊括了相当庞大的信息量。如今，越来越多的企业机关采用计算的管理的方式来使用管理系统来对信息进行管理。根据调查显示，计算机管理系统的出现，成功地解决了曾经纸质办公的数据繁杂，易丢失，不易查找等问题，解决了缺乏信息管理手段，规范不到位的问题。

高新开发区是知识和技术的密集区，它是为了发展高新技术为目的而设置的特定区域，借助以先进的科技、资金和管理手段，实现软硬结合的局部优化，要求极大限度的打开生产，将科技成果转化为生产力。对于高新区的管理和运营，企业政策管理系统更加契合现代信息化的要求，充分地利用信息化带来的便捷性。为高新区工作人员提供一个企业信息管理方面，实现招商引资相关流程、资料、信息的网上管理，实现招商引资相关电子文件的生成、传输与共享全流程自动化管理，以及文件的安全传输与有效验证，实现企业信息的记录、检索、统计等一系列应用功能，解决工作人员在招商工作中数据遗漏丢失、资源分布零散、统计耗时耗力、信息查询不便等问题。

### 1.1.2 国内外研究现状

上世纪七十年代之后，管理系统开始飞速发展，尤其是西方发达国家对于管理系统的发展投入了大量精力进行开发研究，并取得了较好的成果并达到较高水准。管理系统在当今社会中扮演着越来越重要的角色，它是为达到相应的组织目的，集计算机技术，网络技术，管理科学以及系统科学为一体的重要人机交互系统。

20世纪70年代之后，管理系统在国外得到了飞速的进展，尤其是某些发达国家，他们的发展同时也带动了管理系统的发展，并结合实践和理论的条件下不断创新发散，取得了卓越的进展，达到了较高的水准。其发展过程大体分为局域网阶段和一体化阶段。其中在局域网阶段，运用的是计算机和程控交换机来解决不同设备的连接问题，从而实现一部分业务的自动化运作。至此，管理系统已经初成雏形，可以用结构化数据存储并处理对象，其强调的主要是数据的计算以及其统计能力。在一体化阶段，采用的是微型计算机、小型计算机以及计算站来实现办公的自动化，该阶段以网络为中心以处理数据及存储。20世纪90年代之后，信息技术突飞猛进，全球进入信息化时代。随着信息技术不断发展，网络技术已然成为了办公系统的主要发展特征，大大提高了办公效率，真正实现了办公的自动化。

对于国内而言，受技术发展所限，我国办公系统的发展要稍晚于欧美等发达国家，但是随着改革开放的不但深入，我国经济不断发展，并获得了相当的成果，这样带动了我国国民素质及信息技术的不断发展，可谓是后劲十足。我国在近三十年里管理系统得到了相当的发展，总体来说分为了三个阶段。第一个阶段主要是20世纪80年代开始，个人电脑进入市场，开始小范围普及，此时纸质办公可以形成电子文档以缩小储存空间。第二阶段是20世纪90年代中叶，新兴的网络技术带动了发展，网络技术与办公结合，实现了工作流程的自动化，信息开始能在单位内部进行流转，发布及管理，真正意义上实现了自动化。步入新世纪之后，智能化成为潮流，智能化结合管理系统的技术开始兴起，将目前“决策支持系统”的处理思路转向“智能决策系统”，逐步实现了管理系统的智能化。

随着计算机科学与信息技术的不断进步，计算机办公已然越发普及，各式各样的管理系统已经取代了曾经的人力办公，极好的解决了人力办公导致的诸多问题：安全性问题，储存问题，效率低下问题，冗余问题等等。本系统的开发与实现目的就是研究如何通过电脑信息技术快速且较好的解决管理系统的问题，集普遍性、目的性、独特性、创新性于一体，用于打造政府与企业互通的平台。随着世界经济和我国经济逐步向知识经济迈进，无论是基于工业经济的管理信息系统，还是面向少数专家和管理人员的专家系统或决策支持系统，其处理对象和服务对象，自身的系统结构，处理能力，都将不适应或不完全适应知识经济发展的要求。知识经济要求全体员工参与创新，这就使得管理信息系统向更高一级信息系统-知识信息系统发展进化成了一种必然趋势。

### 1.1.3 研究的意义

随着计算机的普及应用、网络技术的飞速发展，高新区政府需要采取必要的信息化手段来规范管理新兴产业，并提高企业管理效率。然而在一定程度上，仅仅是单一的零散化管理系统只能在一定程度上缓解纸质办公的冗余度高、信息残留严重、存储量过大导致的诸多问题，但是这种零散的管理系统还是存在诸多问题。该项目致力于完成管理系统的一体化，进一步提高生产效率，科学管理水平以及决策效能。我们有必要，也势必要建立一个信息管理系统，使企业信息管理变得更加自动化、规范化且流程化，避免纸质办公产生的一系列问题，能够精确、高效、及时处理各种问题。

## 1.2 系统目标

该项目的总体目标是为高新区建设标准统一、结构合理、功能完善、安全稳定的企业政策管理服务综合平台，实现高新招商引资工作的网络化运行，提高办事效率，形成信息共享、协同管理的招商引资机制。在为企业用户提供服务方面，该平台的目标是实现企业人员在系统中发起政策兑现申请，减少企业跑路次数与时间，为企业提供操作简便的企业扶持政策兑现申请服务平台，实现企业在平台中发起政策兑现申请，申请进度在平台中查看。

该项目有五大模块，分别是：企业信息填报系统、企业信息管理系统、招商引资档案管理系统、固定资产评估管理系统、领导审阅统计管理系统。以下分别简述五大模块的功能和作用。

（1）企业信息填报系统，企业人员可以对自己本企业的信息进行录入工作，录入的企业信息将进入数据库以供相关政府人员查看评估。企业基本信息上报完成后，某些属于固定资产投资公司的企业要进入固定资产评估环节，等待固定资产评估员评估完成后对企业申报政策信息进行上报，不属于固定资产投资公司的企业可以直接对其企业所需要的政策进行上报，然后等待其审核完成。审批流程全部通过后，企业可以正式开始实施企业对应政策所需要的项目，待项目完成后，企业通过该系统上报完成情况，最终完成上报，企业方面的任务全部完成。

（2）企业信息管理系统，这个模块面相的人群有两种，第一是企业用户，第二是政府的工作人员。企业用户自己要负责其企业的基本信息录入，包括企业名称，性质，责任人等信息，并上传相关附件。然后政府的工作人员可以对企业信息进行补录，查询以及一些政策的参数设定。通过完成这些目标，搭建一个相对完整的信息管理与贮存系统。

（3）招商引资档案管理系统，它是根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案查询。本模块功能需要系统管理员设置相关权限才能够进行操作与查询。

（4）固定资产管理系统，提供该系统是为固定资产投资公司建立第三方测评档案，方便高新区进行档案管理和投资公司管理。这个模块中同样包含有增删改查和导出几大功能，为的是建立一个方便管理人员编辑，审阅和统计固定资产投资公司的测评档案，也为了方便保存与管理。

（5）领导审阅统计系统，高新区领导可以通过该子系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该子系统，高新区领导可以对整个高新区进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

2 功能需求

## 2.1 业务需求

### 2.1.1企业信息管理业务需求

企业基本信息管理业务主要负责的是企业基本信息的录入，补录，基本信息的查询还有企业基本信息的管理等操作。本业务主要针对于使用该系统的政府工作人员以及企业人员，其中政府工作人员负责对各个企业的信息管理工作，而企业人员负责的是自己企业的基本信息录入，且根据权限要求不得访问其他企业的基本信息。在政府工作人员方面需要工作人员登录系统并核实身份权限后，进入属于自己的工作台选择企业信息管理选项，然后对自己需要的操作进行选定，并进入下一步操作。

（1）企业基本信息补录

如图2-1所示，在有需要的情况下，企业用户需要高新区工作人员对其企业信息进行新增，高新区工作人员根据其需求，登录系统在认证后，对企业信息进行录入。和企业基本信息功能相同，该部分也需要支持上传附件功能。流程图如下所示。

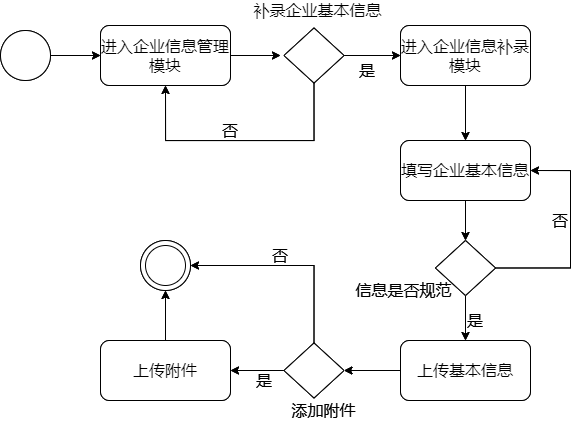


图2-1 企业基本信息补录流程

（2）企业基本信息查询

由于本系统针对的是整个高新区的信息管理，数据量存在一个非常庞大的问题，用户可能不能逐条用肉眼筛选符合条件的信息，此时，用户会需求一个查询的功能，通过关键字，针对不同的查询类别进行查询，如图 2-2所示，用户可以根据企业信息、企业性质、企业负责人、企业入住时间、签订政策等进行查询，系统返回符合查询信息的所有企业。

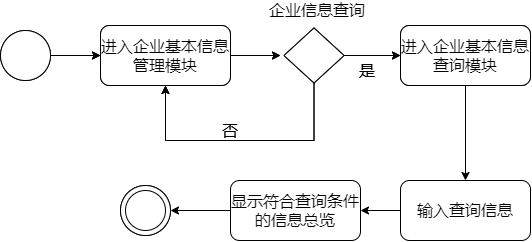


图2-2 企业基本信息查询流程

（3）企业基本信息管理

在某些特定的情况下，高新区工作人员和领导需求对企业信息的管理，包括对企业信息的查看，删除。由于针对整个高新区而言，企业信息可能体量非常庞大，不单单局限于在系统内查看企业信息，会需求使用Excel的方式来查看企业信息，因此该部分需要提供导出企业信息为Excel表格的功能以供选择。如图2-3，用户管理企业信息流程如图所示。

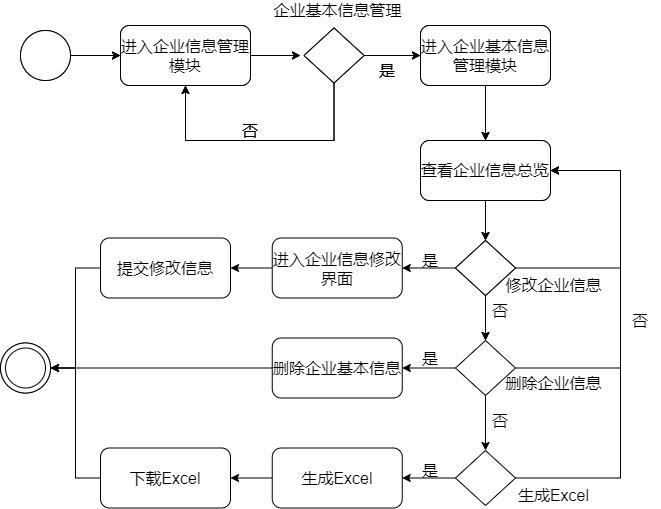


图2-3 企业基本信息管理流程

### 2.1.2招商引资档案管理业务需求

招商引资档案管理业务主要负责的是建立一个电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案查询。本模块功能需要系统管理员设置相关权限才能够进行操作与查询。档案管理中的档案主要是企业扶持政策兑现审批中审批结束的档案，管理人员可对档案进行新建、编辑和删除等操作。

（1）招商引资档案信息录入

在企业申请的项目完成后，上报项目完成情况报告，审批后档案局人员需要对企业信息进行录入，用以保留电子档，方便后期查找。由于需要入档，其信息以完备为必要条件，单纯的表单信息填写可能不足以备档，应此提供附件上传功能。如图2-4所示，为招商引资档案信息录入流程图，其录入信息必须符合相应标准方可进行上传。

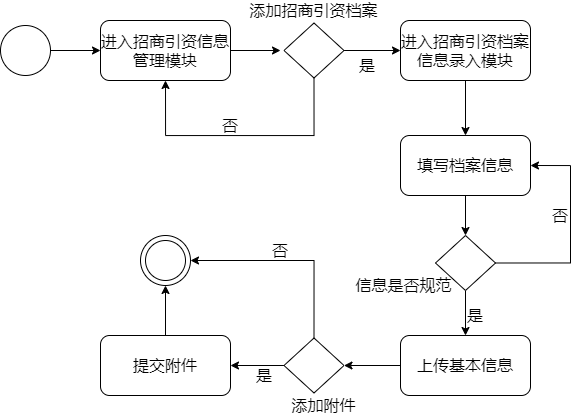


图2-4 招商引资档案信息录入流程

（2）招商引资档案信息查询

如图2-5所示，由于本系统针对的是整个高新区的信息管理，数据量存在一个非常庞大的问题，用户可能不能逐条用肉眼筛选符合条件的信息，此时，用户会需求一个查询的功能，通过关键字，针对不同的查询类别进行查询，如图 2-7所示，档案局人员可以根据责任人、责任处室、题名、时间等进行查询，系统返回符合查询信息的所有档案信息并罗列出来以供用户查看和使用。

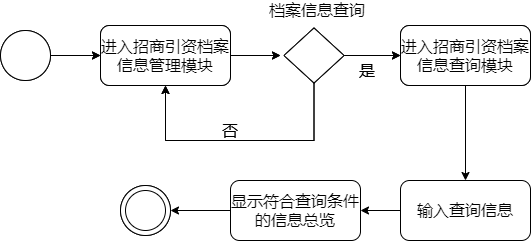


图2-5招商引资档案信息查询流程

（3）招商引资档案信息管理

如图2-6所示，根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案统计。该部分分为三个小模块：查询、删除、修改。其中编辑模块功能是，管理人员可对归档的档案进行编辑、表单查看和流程查看操作。其中删除模块功能是，管理人员可对档案进行删除操作，删除后的档案无法恢复。

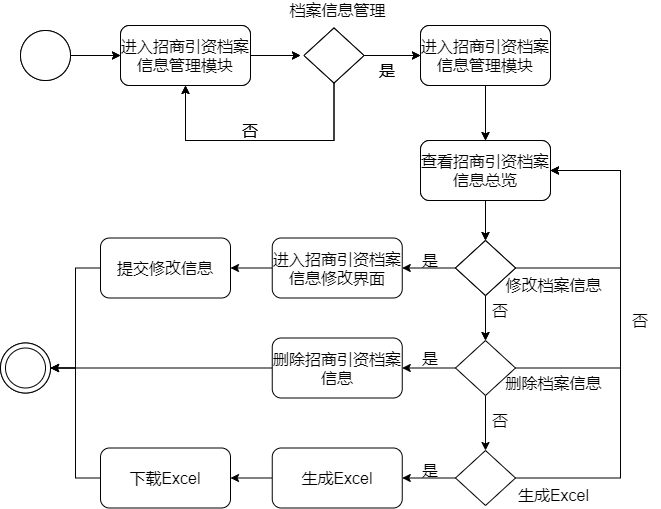


图2-6 招商引资档案信息管理流程

### 2.1.3固定资产投资评估管理业务需求

提供固定资产投资评估管理系统，为固定资产投资公司建立第三方测评档案，方便高新区进行档案管理和投资公司管理。固定资产投资评估管理系统包括新增、修改、删除、查询、导出Excel表格等功能。该部分业务主要的使用者是政府的工作人员，工作人员通过该部分业务可以非常简便的查看和使用第三方评估公司的评估报告，并以此对公司进行测评和管理。

（1）固定资产投资评估信息录入

如图2-8所示， 系统通过筛选，新增固定资产投资评估管理系统中企业信息为企业基本信息中企业类型为“固定资产投资”类型企业的记录，企业基本信息在新增记录时不能修改，只能针对第三方评估记录进行信息新增，包括第三方评估公司、评估时间、评估结果、评估附件等信息。

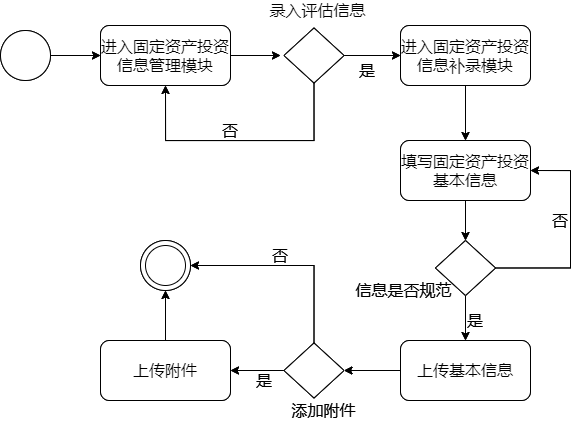


图2-8 固定资产投资评估信息录入流程

（2）固定资产投资评估信息管理

如图2-9所示，在该部分，管理人员可对固定资产评估记录进行修改，记录中企业基本信息不能修改，第三方评估公司、评估时间、评估结果、评估附件等信息可修改。另外管理人员可对固定资产评估记录进行删除，删除后该记录将不会在系统中显示。在固定资产投资评估管理系统中，管理人员可通过各种条件查询评估记录。

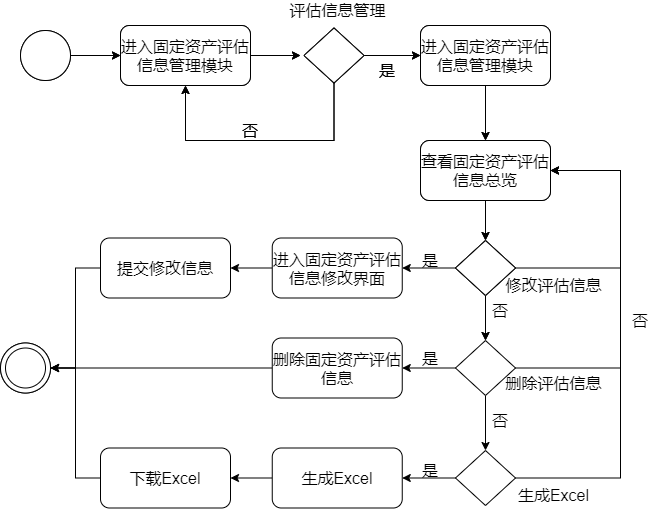


图2-9 固定资产投资评估信息管理流程

（3）固定资产投资评估信息查询

由于本系统针对的是整个高新区的信息管理，数据量存在一个非常庞大的问题，用户可能不能逐条用肉眼筛选符合条件的信息，此时，用户会需求一个查询的功能，通过关键字，针对不同的查询类别进行查询，如图 2-10所示，用户可以根据企业名称、时间段、企第三方评估公司等进行查询，系统返回符合查询信息的所有企业并罗列出来以供用户查看。

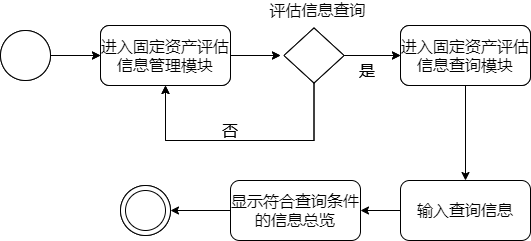


图2-9 固定资产投资评估信息查询流程

### 2.1.4企业信息填报业务需求

提供企业基本信息填报系统，为企业业务员提供向相关服务要求，首先，企业用户可以对本企业的基本信息进行相关的录入功能，通过系统的自动判断，如果属于固定资产投资公司，企业业务员等待政府相关审批，后对企业相关的政策进行上报，如果不属于固定资产投资公司可以直接进行政策上报。经过政府所有审批完成后，企业可以开始完成相关政策要求完成的项目，完成后对完成信息进行上报，最后政府相关单位进行审批归档。另外，企业业务员可以对其企业相关的申请进度进行查看，方便了解其企业的上报进度。

（1）企业基本信息录入

如图2-11所示，企业业务员在经过审批后登录账号进入系统，然后对企业基本信息进行录入，录入的基本信息要符合相关规范。如果企业业务员有相关需要，可以对本企业的相关附件进行上传，企业基本信息录入完成。

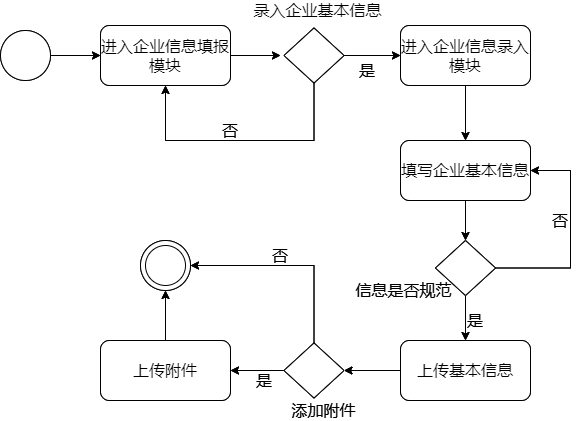


图2-11 企业基本信息录入流程

（2）政策申报

如图2-12所示，企业业务员在政府相关部门对其对应企业进行固定资产投资评估通过后，对其企业所需的政策进行申报，并填写相关信息进行上传。

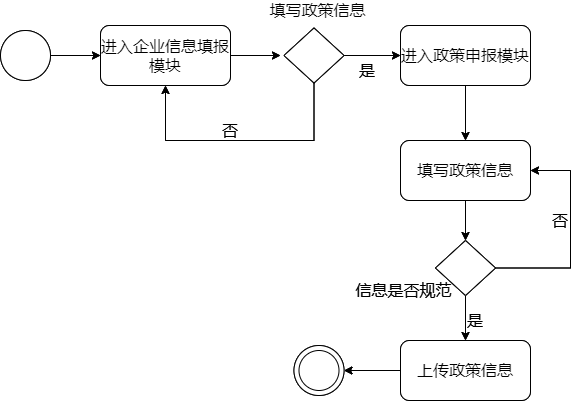


图2-12 政策申报流程

（3）完成情况上报

如图2-13所示，当企业完成对应政策要求的相关项目后，可以对完成信息进行上报，并等待政策相关部门进行核实入档，完成信息需要符合相关规范才能进行上传，否则无法上传。

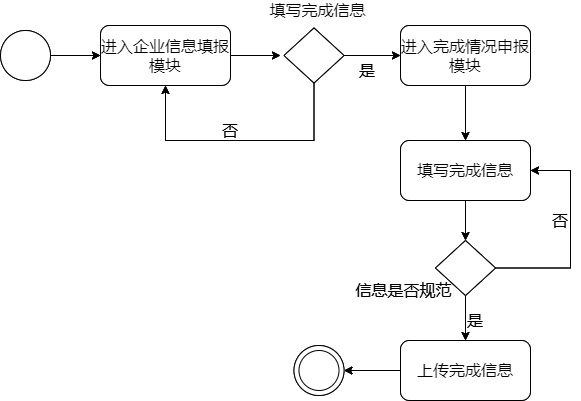


图2-13完成情况上报流程

（4）上报进度查询

如图2-14所示，企业业务员可以对其企业相关的申请进度进行查看，方便了解其企业的上报进度。

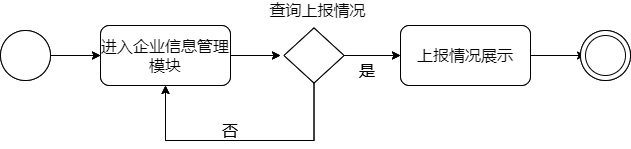


图2-14完成情况上报流程

### 2.1.5领导审阅统计管理业务需求

高新区相关领导可以通过该系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该系统，领导可以对整个高新区进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

（1）政策审阅

如图2-15所示，领导可以通过系统认证后，对政策审批员的政策审批信息进行审阅，如果不满意可以进行驳回，让政策审批员重新审批，如果满意，领导可以使其通过，让政府开始实施项目。

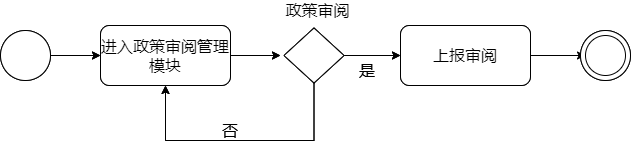


图2-15 固定资产投资评估信息新增流程

（2）统计分析查看

如图2-16所示，固定资产评估员在有需要的情况下，通过查询到需要修改的固定资产进行修改，其修改信息必须符合相应标准方可进行上传。

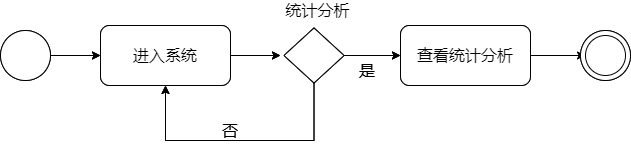


图2-16 固定资产投资评估信息编辑流程

## 2.2 功能需求

### 2.2.1 角色分析

企业业务员：这一部分角色负责的是他们自己企业的信息输入工作，政策申请以及完成情况上报。他们无权访问其他企业的任何信息，只能查询本企业目前所处的审批状态，权限等级低。

高新区工作人员：该部分角色在整个系统中扮演着非常重要的部分，他们在三大模块起着非常关键的作用。第一是企业信息管理模块，他们在这一部分负责对企业信息的补录，查询，信息管理和一些参数的设定，其中信息管理涉及企业信息的统计，导出和删除功能。第二是权限管理系统，在这一部分中高新区工作人员负责了对整个系统的角色权限的运维与管理。本系统对于权限要求较高，不同的用户对应不同权限，这需要工作人员进行管理。另外，高新区管理人员需要对角色进行认证，每当一个用户刚注册进入系统时，都需要提交一份权限认证申请，通过该份权限认证，高新区管理人员能够对其认证和管理，并分配其对应且合适的身份信息，允许其进入系统。

固定资产评估员：该角色面向固定资产评估管理系统，在这一部分中固定资产评估员负责了为固定资产投资公司建立第三方测评档案，方便高新区进行档案管理和投资公司管理，工作人员需要对评估信息进行录入，删除，修改和查询功能，在有需要的情况下他们还需要对评估信息进行导出工作。

档案局人员：招商局人员负责的主要是根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。本模块需要设置相关的权限才能进行操作，档案管理中的档案主要是企业扶持政策兑现审批中审批结束的档案，档案局人员可对档案进行新建、编辑和删除等操作。

政府审批人员：政府审批人员主要针对的是企业所申报的政策进行审批，观察审核其是否符合标准，是否能够达到申请规模，可以对企业申报的政策信息进行通过评议和驳回操作。

领导：领导负责的是对整个高新区的把控，领导能够通过统计面板来更好的决策整个系统的走向和发展路线。另外，领导能够查看所有已经存在的信息并针对其作出自己的判断。此外，领导还应当负责企业政策申报的把关作用，对政府审批员的工作进行认可，并最终同意申报政策能否实施。

### 2.2.2 功能描述

如图2-18所示，该项目总共可以分成5个子模块，分别是：对于企业业务员的企业信息填报模块，对于高新区工作人员的企业信息管理模块，以及对于固定资产评估员的固定资产投资评估信息管理模块，对于招商局工作人员的招商引资档案管理模块，另外还有领导的审阅统计管理。

（1）对于企业业务员的企业信息填报模块而言，需要相应企业注册并登录后才可以进行相关操作，进入相应系统后，企业人员可以对自己本企业的信息进行录入工作，录入的企业信息将进入数据库以供相关政府人员查看评估。企业基本信息上报完成后，某些属于固定资产投资公司的企业要进入固定资产评估环节，等待固定资产评估员评估完成后对企业申报政策信息进行上报，不属于固定资产投资公司的企业可以直接对其企业所需要的政策进行上报，然后等待其审核完成。审批流程全部通过后，企业可以正式开始实施企业对应政策所需要的项目，待项目完成后，企业通过该系统上报完成情况，最终完成上报，企业方面的任务全部完成。值得注意的是该部分用户的权限低，无法查看其他企业的任何信息。

（2）对于高新区工作人员的企业信息管理模块而言，高新区工作人员可以通过该系统对企业信息进行相应的管理，功能主要分为信息补录、查询、修改、删除、导出Excel文件，创业中心工作人员负责的主要工作是针对于企业信息的统计进行开展的，他们需要一系列的措施对企业的信息进行操作。所有由他们增、删、改、查的信息都将进入数据库，提供保存记录，也以便于上级领导查看和审阅。

（3）对于固定资产评估员的固定资产投资评估信息管理模块而言，固定资产评估员需要评估属于固定资产投资公司的企业，并对其进行审核批阅。该系统的主要功能是对于相关企业的固定资产风险评估信息的录入、查询、修改、删除和导出为Excel文件，固定资产评估员工作是围绕评估信息展开的，他们需要该子系统进行相关操作。由该系统最终确认的信息也都将进入数据库进行保存，以便下一步工作的展开。

（4）对于档案局人员的招商引资档案管理模块而言，这里的主要功能是围绕档案局的相关工作人员展开的。在企业业务员上报企业完成情况后，档案局人员对其完成情况进行入档，并根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案查询档案管理中的档案主要是企业扶持政策兑现审批中审批结束的档案，管理人员可对档案进行新建、编辑和删除等操作。

（5）对于领导的审阅统计管理而言，这个模块的主要功能是为高新区领导提供的。高新区领导可以通过该子系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该子系统，高新区领导可以对整个高新区进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

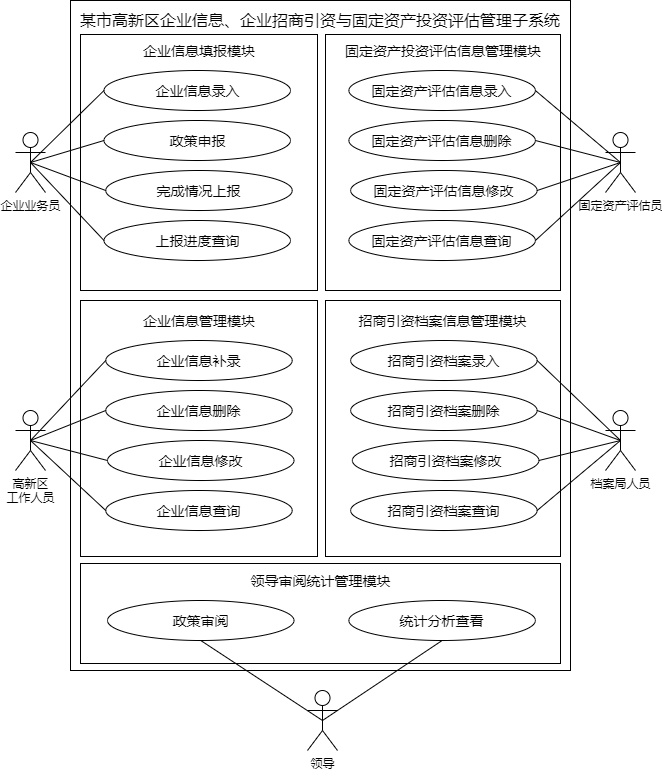


图2-18 系统总用例图

### 2.2.3 企业信息填报模块功能分析

如图2-19所示企业业务员可以通过该系统对本企业的信息进行录入，如果有附件需要补充可以进行上传，待企业信息审批完成后，企业业务员可以开始申报政策，等待政府审批人员审批完成后，可以开始实施项目，项目完成即可进行完成情况上报并归档。

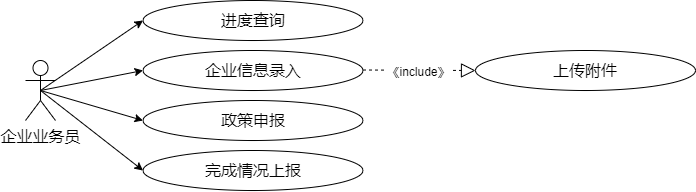


图2-19企业信息填报

如表2-1所示，展示的是企业基本信息录入的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对企业基本信息录入的用例进行了相关描述。

表2-1 企业基本信息录入

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0001 | 用例名称 | 企业基本信息录入 |
| 用例描述 | 提供企业基本信息录入功能，由企业对企业信息进行录入，每个企业只能录入或修改自己企业的信息，无法查看到其他企业信息，包括企业名称、企业性质、企业负责人、企业联系电话、企业入驻时间、签订政策、签订时间、政策完成情况、亮灯情况等信息。支持附件上传功能。 | | |
| 执行者 | 主要执行者：企业业务员 | | |
| 前置条件 | 企业业务员已经成功注册账号并且通过权限的认证并且进入了相关信息系统的管理平台。 | | |
| 后置条件 | 成功录入本企业基本信息后点击完成，跳转至企业基本信息管理界面后进入其他功能的选择界面。 | | |
| 涉众利益 | 1.用户可能担心本企业信息泄露； | | |
| 用例场景 | 企业基本信息添加界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面  2. 在侧边导航栏点击按钮进入企业基本信息录入界面  3. 填写信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入 | | |

续表2-1

|  |  |
| --- | --- |
| 基本事件流 | **字段列表（Filed List）**  1. ID  2．企业名称  3．企业注册地址  4．企业法人  5．企业责任人  6．企业电话  7．企业邮箱  8．注册资金  9．企业类型  10．企业注册时间  11．经营范围  12．企业等级  13．企业联系地址  14．企业邮编  15．开户银行  16．企业入驻时间  17．附件地址  18．添加账号  **业务规则（Business role）**   1. 企业名称不能为空 2. 企业性质为下拉表单，不能为空，在8类中进行选择：国有企业、集体所有制、私营企业、股份制企业、有限合伙企业、联营企业、外商投资企业、个人独资企业 3. 企业负责人不能为空 4. 企业入驻时间不能为空 5. 经营范围为下拉表单，在10类中进行选择：互联网、医疗、文娱、餐饮、金融、商贸、建筑、旅游、教育、其他 6. 联系地址不能为空 7. 邮编不能为空 8. 开户银行不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-2所示，展示的是政策申报的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对政策申报的用例进行了相关描述。

表2-2 政策申报

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0002 | 用例名称 | 政策申报 |
| 用例描述 | 企业需要填写申请的政策信息，其信息包括账号、申请单位、企业地址、邮政编码、联系人、开户行等信息，并上传至数据库，等待政府审批人员进行审核。 | | |
| 执行者 | 主要执行者：企业业务员 | | |

续表2-2

|  |  |
| --- | --- |
| 前置条件 | 企业业务员已经成功注册账号并且通过基本的认证并且进入了相关信息系统的管理平台。 |
| 后置条件 | 成功申报政策后，跳转至主界面，耐心等待政府审批 |
| 涉众利益 | 1.用户可能担心本企业信息泄露； |
| 用例场景 | 政策申报界面 |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面  2. 在侧边导航栏点击按钮进入政策申报界面  3. 填写信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1．id  2．申请单位  3．企业地址  4．邮政编码  5．联系人  6．开户行  7．账号  8．资金类别  9．已获得扶持资金  10．申请时间  11．项目所在国（地区）  12．项目实施时间  13．项目负责人  14．电话  15．申请条例  16．申请扶持内容  17．申请扶持金额  18．附件  **业务规则（Business role）**  1．账号不能为空  2．申请单位不能为空  3．企业地址不能为空  4．邮政编码不能为空 |

续表2-2

|  |  |
| --- | --- |
| 基本事件流 | 5．联系人不能为空  6．开户行不能为空  7．账号不能为空  8．资金类别不能为空  9．已获得扶持资金不能为空  10．申请时间不能为空  11．项目所在国（地区）不能为空  12．项目实施时间不能为空  13．项目负责人不能为空  14．电话不能为空  15．申请条例不能为空  16．申请扶持内容不能为空  17．申请扶持金额不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-3所示，展示的是完成情况上报的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对完成情况上报的用例进行了相关描述。

表2-3 完成情况上报

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0003 | 用例名称 | 完成情况上报 |
| 用例描述 | 企业业务员在企业完成相关项目后填报完成信息，等待企业完成信息审批后结束整体流程。 | | |
| 执行者 | 主要执行者：企业业务员 | | |
| 前置条件 | 企业业务员已经成功注册账号并且通过基本的认证并且进入了相关信息系统的管理平台。 | | |
| 后置条件 | 成功修改本企业基本信息后点击完成，跳转至完成上报界面。 | | |
| 涉众利益 | 1.用户可能担心本企业信息泄露； | | |
| 用例场景 | 完成情况上报界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面  2. 在侧边导航栏点击按钮进入完成情况上报界面  3. 填写相关完成情况上信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）** | | |

续表2-3

|  |  |
| --- | --- |
| 基本事件流 | 1．id  2．申请单位  3．企业地址  4．邮政编码  5．联系人  6．开户行  7．账号  8．资金类别  9．已获得扶持资金  10．申请时间  11．项目所在国（地区）  12．项目实施时间  13．项目负责人  14．电话  15．申请条例  16．申请扶持内容  17．申请扶持金额  **业务规则（Business role）**  1.企业名称不能为空  2.企业性质为下拉表单，不能为空，在8类中进行选择：国有企业、集体所有制、私营企业、股份制企业等  3.企业负责人不能为空  4.企业入驻时间不能为空  5.企业等级不能为空 |
| 特殊需求（Special Requirement） | 无 |

### 企业信息管理模块功能分析

如图2-20所示，高新区工作人员根据企业的要求能够对企业信息进行补录。另外可以对企业信息进行总览，并对不需要的企业信息进行删除操作，还可以对企业信息进行修改。由于信息体量的庞大，该模块还提供了对企业信息的查询功能及导出功能。

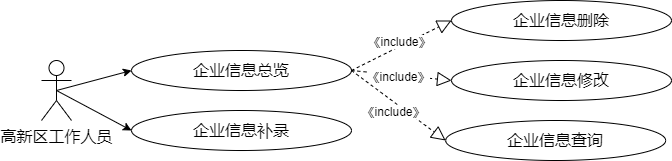


图2-20企业信息管理

如表2-4所示，展示的是企业基本信息补录的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对企业基本信息补录的用例进行了相关描述。

表2-4 企业基本信息补录

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0004 | 用例名称 | 企业基本信息补录 |
| 用例描述 | 提供企业基本信息补录功能，包括企业名称、企业性质、企业负责人、企业联系电话、企业入驻时间、签订政策、签订时间、政策完成情况、亮灯情况等信息。支持附件上传功能。 | | |
| 执行者 | 主要执行者：高新区创业中心工作人员 | | |
| 前置条件 | 高新区创业中心管理人员已经成功登录账号并拥有相关权限 | | |
| 后置条件 | 在补录基本信息后点击按钮上传信息，将数据上传至数据库 | | |
| 涉众利益 | 1. 操作不慎导致企业信息流出 2. 企业基本信息被误删 | | |
| 用例场景 | 企业基本信息补录界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面（高新区工作人员）  2. 在侧边导航栏点击按钮进入补录企业基本信息界面  3. 填写相关企业信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1. ID  2．企业名称  3．企业注册地址  4．企业法人  5．企业责任人  6．企业电话  7．企业邮箱  8．注册资金  9．企业类型  10．企业注册时间  11．经营范围  12．企业等级  13．企业联系地址  14．企业邮编  15．开户银行 | | |

续表2-4

|  |  |
| --- | --- |
| 基本事件流 | 16．企业入驻时间  17．附件地址  18．添加账号  **业务规则（Business role）**  1. 企业名称不能为空  2. 企业性质为下拉表单，不能为空，在8类中进行选择：国有企业、集体所有制、私营企业、股份制企业、有限合伙企业、联营企业、外商投资企业、个人独资企业  3. 企业负责人不能为空  4. 企业入驻时间不能为空  5. 经营范围为下拉表单，在10类中进行选择：互联网、医疗、文娱、餐饮、金融、商贸、建筑、旅游、教育、其他  6. 联系地址不能为空  7. 邮编不能为空 |
| 基本事件流 | 8. 开户银行不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-5所示，展示的是企业基本信息管理的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对企业基本信息管理的用例进行了相关描述。

表2-5 企业基本信息管理

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0005 | 用例名称 | 企业基本信息管理 |
| 用例描述 | 相关人员可通过企业名称、企业性质、企业负责人、企业入驻时间、签订政策、企业等级等条件对企业信息进行查询。支持对企业详情进行查看。在该部分中，管理人员可对企业基本信息进行管理，包括查询、修改、删除等操作。支持企业基本信息打印功能，支持企业基本信息导出成Excel表格功能。 | | |
| 执行者 | 主要执行者：高新区创业中心工作人员 | | |
| 前置条件 | 高新区创业中心管理人员已经成功登录账号并拥有相关权限 | | |
| 后置条件 | 1. 如果只是需要查询，完成相关操作后点击返回即可 2. 如果需要导出，需要查询后点击导出为Excel文件进行导出 3. 如果需要删除，需要查询后点击删除即可 | | |
| 涉众利益 | 1.操作不慎导致企业信息流出 | | |
| 用例场景 | 企业基本信息统计界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面（高新区工作人员）  2. 在侧边导航栏点击按钮进入企业基本信息统计界面  3. 填入相关信息，进行查询  4. 导出操作，查询后点击导出为Excel文件进行导出  5. 删除操作，需要查询后点击删除即可  **扩展流程（Extend Flow）** | | |

续表2-5

|  |  |
| --- | --- |
| 基本事件流 | 1a. 查询输入信息不规则：  1a.1系统提示输入信息不规则  1a.2重新检查数据后再次输入  **字段列表（Filed List）**  1. ID  2．企业名称  3．企业注册地址  4．企业法人  5．企业责任人  6．企业电话  7．企业邮箱  8．注册资金  9．企业类型  10．企业注册时间  11．经营范围  12．企业等级  13．企业联系地址  14．企业邮编  15．开户银行  16．企业入驻时间  17．附件地址  18．添加账号  **业务规则（Business role）**  1. 企业名称不能为空  2. 企业性质为下拉表单，不能为空，在8类中进行选择：国有企业、集体所有制、私营企业、股份制企业、有限合伙企业、联营企业、外商投资企业、个人独资企业  3. 企业负责人不能为空  4. 企业入驻时间不能为空  5. 经营范围为下拉表单，在10类中进行选择：互联网、医疗、  文娱、餐饮、金融、商贸、建筑、旅游、教育、其他  6. 联系地址不能为空  7. 邮编不能为空  8. 开户银行不能为空 |
| 特殊需求（Special Requirement） | 无 |

### 2.2.5 固定资产评估信息管理模块功能分析

如图2-21所示，固定资产评估员根据企业信息对企业进行固定资产评估并将固定资产信息录入系统。另外可以对固定资产评估信息进行总览，并对不需要的固定资产评估信息进行删除操作，还可以对固定资产评估信息进行修改。由于信息体量的庞大，该模块也提供了对固定资产评估信息的查询功能及导出功能。

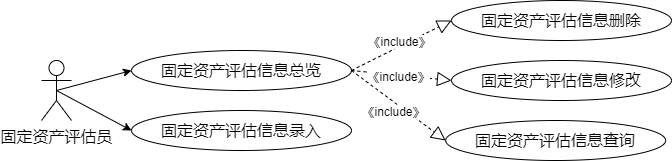


图2-21固定资产评估信息管理

如表2-6所示，展示的是固定资产评估基本信息录入的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对固定资产评估基本信息录入的用例进行了相关描述。

表2-6 固定资产评估基本信息录入

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0006 | 用例名称 | 固定资产评估基本信息录入 |
| 用例描述 | 在新增固定资产投资评估记录时，只能选择新增固定资产投资评估管理系统中企业信息为企业基本信息中企业类型为“固定资产投资”类型企业的记录，企业基本信息在新增记录时不能修改，只能针对第三方评估记录进行信息新增，包括第三方评估公司、评估时间、评估结果、评估附件等信息。 | | |
| 执行者 | 主要执行者：高新区创业中心工作人员 | | |
| 前置条件 | 高新区创业中心工作人员已经成功登录账号并拥有相关权限 | | |
| 后置条件 | 在录入基本信息后点击按钮上传信息，将数据上传至数据库 | | |
| 涉众利益 | 1.操作不慎导致固定资产评估信息流出 | | |
| 用例场景 | 固定资产评估信息录入界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入企业基本信息管理界面（高新区工作人员）  2. 在侧边导航栏点击按钮进入固定资产评估基本信息录入界面  3. 填写相关信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1.评估ID  2.评估公司 | | |

续表2-6

|  |  |
| --- | --- |
| 基本事件流 | 3.负责人  4.公司成立时间  5.公司地址  6.组织形式  7.邮箱  8.电话  9.经办人姓名  10.经办人电话  11.经办人邮箱  12.经办人地址  13.评估时间  14.附件地址  15.添加账号  **业务规则（Business role）**  1.评估ID不能为空  2.评估公司不能为空  3.负责人不能为空  4.公司成立时间不能为空  5.公司地址不能为空  6.组织形式不能为空  7.邮箱不能为空，格式应符合邮箱标准  8.电话不能为空，格式应符合电话标准  9.经办人姓名不能为空  10.经办人电话不能为空，格式应符合电话标准  11.经办人邮箱不能为空，格式应符合邮箱标准  12.经办人地址不能为空  13.评估时间不能为空  14.附件地址不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-7所示，展示的是固定资产评估基本信息录入的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对固定资产评估基本信息录入的用例进行了相关描述。

表2-7 固定资产评估基本信息总览

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0007 | 用例名称 | 固定资产评估基本信息总览 |
| 用例描述 | 在该部分，融合了三个较小的功能模块，分别是查询、修改、删除。管理人员可对固定资产评估记录进行修改，记录中企业基本信息不能修改，第三方评估公司、评估时间、评估结果、评估附件等信息可修改。另外管理人员可对固定资产评估记录进  行删除，删除后该记录将不会在系统中显示。在固定资产投资评 | | |

续表2-7

|  |  |
| --- | --- |
| 用例描述 | 估管理系统中，管理人员可通过企业名称、评估时间段、第三方评估公司等条件查询评估记录。该部分支持对查询出的固定资产投资评估记录进行导出操作，到处成Excel表格。 |
| 执行者 | 主要执行者：高新区创业中心工作人员 |
| 前置条件 | 高新区创业中心工作人员已经成功登录账号并拥有相关权限 |
| 后置条件 | 1.如果只是需要查询，完成相关操作后点击返回即可  2.如果需要导出，需要查询后点击导出为Excel文件进行导出  3.如果需要删除，需要查询后点击删除即可 |
| 涉众利益 | 1.操作不慎导致固定资产评估信息流出 |
| 用例场景 | 固定资产评估信息统计界面 |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入固定资产评估信息管理界面（高新区工作人员）  2. 在侧边导航栏点击按钮进入固定资产评估基本信息统计界面  3. 填入相关信息，进行查询  4. 导出操作，查询后点击导出为Excel文件进行导出  5. 删除操作，需要查询后点击删除即可  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1.评估ID  2.评估公司  3.负责人  4.公司成立时间  5.公司地址  6.组织形式  7.邮箱  8.电话  9.经办人姓名  10.经办人电话  11.经办人邮箱  12.经办人地址  13.评估时间  14.附件地址  15.添加账号  **业务规则（Business role）**  1.评估ID不能为空 |

续表2-7

|  |  |
| --- | --- |
| 基本事件流 | 2.评估公司不能为空  3.负责人不能为空  4.公司成立时间不能为空  5.公司地址不能为空  6.组织形式不能为空  7.邮箱不能为空，格式应符合邮箱标准  8.电话不能为空，格式应符合电话标准  9.经办人姓名不能为空  10.经办人电话不能为空，格式应符合电话标准  11.经办人邮箱不能为空，格式应符合邮箱标准  12.经办人地址不能为空  13.评估时间不能为空  14.附件地址不能为空 |
| 特殊需求（Special Requirement） | 无 |

### 2.2.6 招商引资档案管理模块详细功能分析

如图2-16所示，档案局人员根据企业完成信息对企业进行归档并将招商引资档案信息录入系统。另外可以对招商引资档案信息进行总览，并对不需要的招商引资档案信息进行删除操作，还可以对招商引资档案信息进行修改。

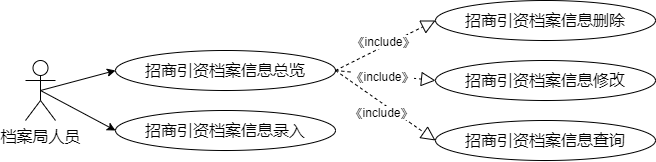


图2-17招商引资档案信息管理

如表2-8所示，展示的是招商引资基本信息录入的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对招商引资基本信息录入的用例进行了相关描述。

表2-8 招商引资基本信息录入

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0008 | 用例名称 | 招商引资基本信息录入 |
| 用例描述 | 根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案新增。 | | |
| 执行者 | 主要执行者：招商局工作人员 | | |
| 前置条件 | 招商局用户已经成功注册账号并且通过基本的认证并且进入了相关信息系统的管理平台。 | | |

续表2-8

|  |  |
| --- | --- |
| 后置条件 | 成功录入招商引资基本信息后点击完成，跳转至招商引资基本信息管理界面后进入其他功能的选择界面。 |
| 涉众利益 | 1.用户可能担心本招商引资信息泄露； |
| 用例场景 | 招商引资基本信息添加界面 |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入招商引资信息管理界面  2. 在侧边导航栏点击按钮进入招商引资信息录入界面  3. 填写信息  4. 点击完成录入数据库  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**   1. 档案ID 2. 档案责任人 3. 责任处室 4. 档案名 5. 入档时间 6. 附件路径   **业务规则（Business role）**  1. 档案ID不能为空  2. 档案责任人不能为空  3. 责任处室不能为空  4. 档案名不能为空  5. 入档时间不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-9所示，展示的是招商引资基本信息总览的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对招商引资基本信息总览的用例进行了相关描述。

表2-9 招商引资基本信息总览

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0009 | 用例名称 | 招商引资基本信息总览 |
| 用例描述 | 根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案统计。该部分分为三个小模块：查询、删除、修改。其中编辑模块功能是，管理人员可对归档的档案进行编辑、表单查看和流程查看操作。其中删除模块功 | | |

续表2-9

|  |  |
| --- | --- |
| 用例描述 | 能，管理人员可对档案进行删除操作，删除后的档案无法恢复。 |
| 执行者 | 主要执行者：招商局工作人员 |
| 前置条件 | 招商局工作人员已经成功登录账号并拥有相关权限 |
| 后置条件 | 1.如果只是需要查询，完成相关操作后点击返回即可  2.如果需要导出，需要查询后点击导出为Excel文件进行导出  3.如果需要删除，需要查询后点击删除即可 |
| 涉众利益 | 1.操作不慎导致招商引资基本信息流出 |
| 用例场景 | 招商引资基本信息统计界面 |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入招商引资基本信息管理界面（招商局工作人员）  2. 在侧边导航栏点击按钮进入招商引资基本信息统计界面  3. 填入相关信息，进行查询  4. 导出操作，查询后点击导出为Excel文件进行导出  5. 删除操作，需要查询后点击删除即可  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1. 档案ID  2. 档案责任人  3. 责任处室  4．档案名  5. 入档时间  6. 附件路径  **业务规则（Business role）**  1. 档案ID不能为空  2. 档案责任人不能为空  3. 责任处室不能为空  4. 档案名不能为空  5. 入档时间不能为空 |
| 特殊需求（Special Requirement） | 无 |

### 2.2.7 领导审阅统计管理模块详细功能分析

如图2-18所示，这个模块的主要功能是为高新区领导提供的。高新区领导可以通过该子系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该子系统，高新区领导可以对整个高新区进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

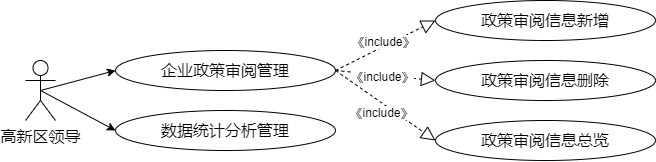


图2-18领导审阅统计管理模块

如表2-10所示，展示的是企业政策审阅管理的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对企业政策审阅管理的用例进行了相关描述。

表2-10 企业政策审阅管理

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0010 | 用例名称 | 企业政策审阅管理 |
| 用例描述 | 高新区领导对政策审批员审批过的信息进行审阅，并对其进行审阅，高新区领导的可以对其进行驳回或通过，驳回的信息重新回到政策审批员手中，并重新审批。 | | |
| 执行者 | 主要执行者：高新区领导 | | |
| 前置条件 | 高新区领导已经成功注册账号并且通过基本的认证并且进入了相关信息系统的管理平台。 | | |
| 后置条件 | 成功录入审阅基本信息后点击完成，跳转至审阅基本信息管理界面后进入其他功能的选择界面。 | | |
| 涉众利益 | 1.用户可能担心审批审阅信息泄露； | | |
| 用例场景 | 审批审阅信息管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 进入审批审阅信息管理界面  2. 在侧边导航栏点击按钮进入审批审阅信息录入界面  3. 删除操作，需要查询后点击删除即可  4. 查询操作，填写相关信息点击查询  **扩展流程（Extend Flow）**  1a. 输入信息不完整：  1a.1系统提示输入信息不完整  1a.2重新检查数据后再次输入  2a. 输入信息格式不正确：  2a.1系统提示输入信息格式不正确  2a.2重新检查数据后再次输入  **字段列表（Filed List）**  1.审阅ID  2.审阅时间  3.审阅结果  4.领导评语 | | |

续表2-10

|  |  |
| --- | --- |
| 基本事件流 | **业务规则（Business role）**  1.审阅时间不能为空  2.审阅结果不能为空  3.领导评语不能为空 |
| 特殊需求（Special Requirement） | 无 |

如表2-11所示，展示的是数据统计分析的用例描述。表中通过用例描述、执行者、前置条件基本事件流等方面对数据统计分析的用例进行了相关描述。

表2-11 数据统计分析

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0011 | 用例名称 | 数据统计分析 |
| 用例描述 | 提供统计分析功能，对企业政策管理服务综合平台中数据进行统计、分析，通过图表展现形式，为领导决策提供数据帮助。企业扶持政策兑现情况：按政策兑现类型、企业性质、政策兑现等情况进行统计分析。企业信息统计：按企业性质、是否贷款等情况进行统计分析。 | | |
| 执行者 | 主要执行者：高新区领导 | | |
| 前置条件 | 高新区领导已经成功注册账号并且通过基本的认证并且进入了相关信息系统的管理平台。 | | |
| 后置条件 | 成功录入审阅基本信息后点击完成，跳转至审阅基本信息管理界面后进入其他功能的选择界面。 | | |
| 涉众利益 | 1.操作不慎导致高新区信息流出 | | |
| 用例场景 | 数据统计分析界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 高新区企业领导进入系统 2. 点击左侧导航栏按钮进入，数据统计分析界面 3. 展示数据统计成果供领导分析查阅   **扩展流程（Extend Flow）**  无  **字段列表（Filed List）**  无  **业务规则（Business role）**  无 | | |
| 特殊需求（Special Requirement） | 无 | | |

## 

## 2.3 非功能需求

### 2.3.1环境需求

如表2-10所示，服务器端部署于阿里云ECS服务器基于Ubuntu18.04系统使用MySQL5.7数据库；如表2-11所示，客户端基于WindowXP或者以上的系统，使用IE6.0以上版本的浏览器即可使用。

表 2-10 服务器端环境

|  |  |
| --- | --- |
| 系统所使用数据库 | MySQL |
| 操作系统 | Ubuntu 18.04 |
| Web服务器 | 阿里云ECS服务器 |
| 数据库 | MySQL 5.7 |

表 2-11 客户端环境

|  |  |
| --- | --- |
| 操作系统 | Windows XP 或其以上版本操作系统 |
| 浏览器 | IE 6.0以及其以上版本（IE7.0、IE8.0） |
| 分辨率 | 最佳效果为1024×768像素 |

### 2.3.2 性能需求

高新区企业政策管理系统的构筑可以解决企业人员跑路次数，大大提高办事效率，所有与政策有关的相关事宜基本都可以自由的在该系统中解决。同时在另一方面，政府相关系统工作人员也减少了纸质办公的复杂度和冗余性。因此该系统针对与企业和政府两大群体进行服务。实现招商引资相关流程、资料、信息的网上管理，实现招商引资相关电子文件的生成、传输与共享全流程自动化管理，以及文件的安全传输与有效验证，实现企业信息的记录、检索、统计等一系列应用功能，解决工作人员在招商工作中数据遗漏丢失、资源分布零散、统计耗时耗力、信息查询不便等问题。

### 2.3.3 安全需求

由于牵涉到企业政策和企业、政府的一些相关隐私信息，该系统应该具备极高的安全保护能力。该政策管理系统应当具有灵活的权限管理设置，规定用户族群只能在自己的权限要求下进行相关操作，以免造成数据泄露等不必要的灾难。由于用户族群的特殊性，该系统对稳定性和可靠性有着相当高的需求。高新区企业政策管理系统需要具有出色的性能，要求所有的系统功能模块、服务接口进行严格的测试，满足系统性能的需求，防止出现储存数据资源的遗失问题，并在一定程度上应当采取数据备份和恢复策略。

3 总体设计

## 3.1系统设计的原则

以先进性、安全性、合理性、经济性、实用性等作为基本的系统设计原则，以此为标准规划整体架构。

（1）先进性。该系统以高新区为载体，高新区作为新兴科技、高新企业的集中区，拥有无法比拟的先进性。本系统作为高新区企业政策的管理系统，也需要在先进性方面有提升，在满足当前的功能需求下，具备一定的预见性，在今后一段时间内保持较高的技术先进性。

（2）安全性。由于本系统为高新区政府工作网站，其对安全性要求有着较高的标准，需要举报较高的安全性和保密性。对于权限认证也需要有较高的要求，并具有一定的容错和系统恢复能力以确保系统稳定且高效长期的正常运行。

（3）合理性。高新区企业政策管理平台同样也需要有合理性，要对整体功能进行详尽的分析，站在用户的立场上思考功能是否完备，并对整体系统进行平滑润滑，提升用户体验，不应指针对于某一种特定型号的设备进行设计。

（4）经济性。在经济方面考虑，为达成利益最大化，在满足合理性和功能完备性的基础上，尽可能压低系统开发成本，高度利用现有设备及资源进行开发。

（5）实用性。作为政府办公系统，实用性方面应当具备较高的标准，需要提供一个简单明了的交互感官，界面应当以清晰、简洁、有好为标准，不需要添加过多华而不实的杂项功能，便于用户灵活简便的使用，上手快、易学易用。具备政府风格的界面和政府相关的操作习惯和客户端界面，在处理事务时具备高效性。

## 3.2 系统体系结构设计

本管理系统主要采用的是B/S结构架构而成，这是一种Web兴起之后的网络结构模式，这种模式统一了客户端，将系统功能实现的核心部分集中在了服务器上，简化了系统的开发、维护以及使用。客户端只需要安装一个浏览器、服务器上安装MySQL数据库，浏览器对数据库进行数据交互。

随着信息时代的到来，网络技术不断发展成熟，B/S结构作为C/S体系的改变和促进，提升了设计的系统性能，更是有效的降低了资源成本，在处理模式方面大大的简化了客户端，用户只需要使用浏览器进行访问即可，具备更强的通用性，对环境依赖更小，开发维护更加简单。其工作模式是浏览器请求，服务器对其请求进行响应。

如图3-1所示，客户端先发送请求，比如用户使用客户端提交表单，向服务器发送请求，等待服务器响应。服务器接收到客户端发送来的请求进行处理，并产生响应，然后将响应发送给客户端浏览器，包括网页文件、图片、相关资源文件等等，最后浏览器执行相关文件，并将其展示给用户。

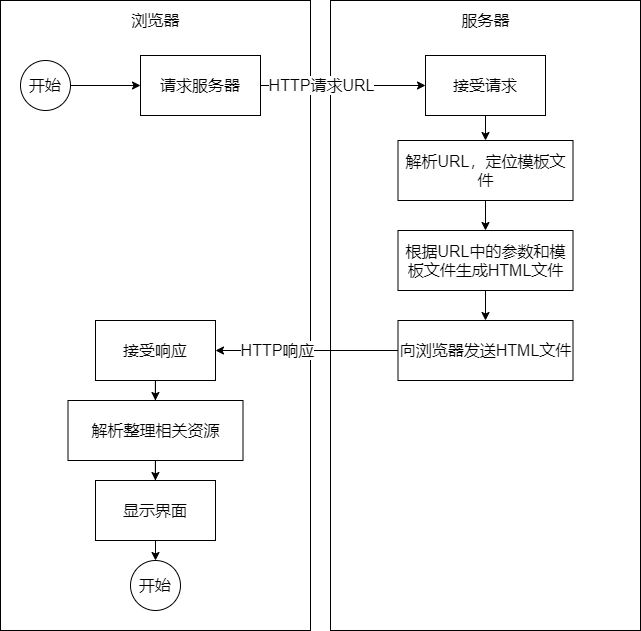


图3-1 B/S工作原理图

## 3.3 技术架构设计

项目使用的是SSM（Spring+SpringMVC+MyBatis）框架技术，前端界面采用的是Bootstrap框架技术。

SSM框架有Spring和My白提示两大开源框架整合而成，其中Spring在整个框架中执行工厂的任务，其核心思想是控制反转，在程序的设计过程中，程序员不再需要显示定义新对象，而是直接使用Spring框架完成。另外SpringMVC作为框架的一部分，其作用是类似于中介，将用户请求匹配到对应Controller处理请求。Mybatis是对JDBC的封装，可以是数据库底层操作变得更加透明，操作也更加方便。用户通过界面发送请求给控制器（Controller），控制器（Controller）通过调用业务层（Service）来处理业务逻辑，通过向持久层（Dao）与数据库进行交互，然后将处理结果交付业务层（Service），并将处理逻辑发送给处理器（Controller），控制器在调用视图层展示数据。

另外，前端采用的是Bootstrap框架技术，Bootstrap是基于HTML、CSS、JavaScript开发的，其最为重要的特性就是简单灵活，可以让Web的开发变得更加快捷方便。

如图3-2所示，展示的整个项目的技术架构设计图，整个项目由五层构成，分别是访问层、应用层、应用支撑层、数据资源层和网络层。其中访问层是用户使用PC端对系统进行访问，另外应用层面，分为企业信息录入、固定资产评估信息管理等五大模块构成。数据访问层是MySQL构成的，包括企业数据库、兑现政策库等等。



图3-2 技术架构设计图

## 3.4 应用架构设计

如图3-3所示，本系统采用的是典型的三层架构，分为界面层、业务逻辑层和数据访问层。其区分层次的目的是实现“高内聚低耦合”的思想。其中表示层主要显示数据和接受传输用户的数据为用户提供交互式的操作界面。另外业务逻辑层是对用户发送来的数据进行相关处理，分别存储。最后数据访问层是对数据库进行访问，实现数据库中对数据的增、删、改、查操作。

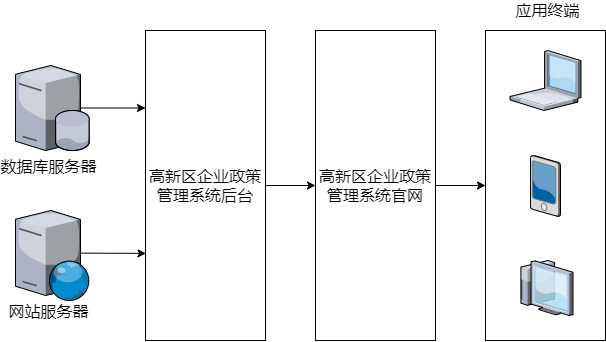


图3-3应用架构设计图

## 3.5 系统功能结构设计

如图3-4所示，该系统分为五大子系统，分别是：企业信息填报子系统、企业信息管理子系统、招商引资档案管理子系统、固定资产投资评估管理子系统、领导审阅统计管理子系统。

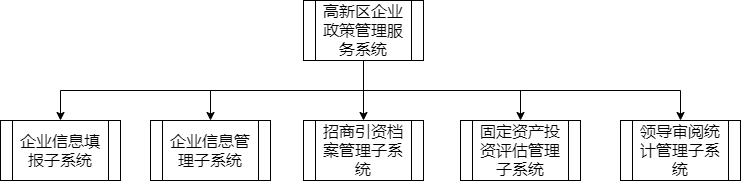


图3-4系统功能结构设计图

（1）招商引资档案管理子系统：招商引资档案管理业务主要负责的是建立一个电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案查询。本模块功能需要系统管理员设置相关权限才能够进行操作与查询。档案管理中的档案主要是企业扶持政策兑现审批中审批结束的档案，管理人员可对档案进行新建、编辑和删除等操作。

（2）固定资产投资评估管理子系统：提供固定资产投资评估管理系统，为固定资产投资公司建立第三方测评档案，方便高新区进行档案管理和投资公司管理。固定资产投资评估管理系统包括新增、修改、删除、查询、导出Excel表格等功能。该部分业务主要的使用者是政府的工作人员，工作人员通过该部分业务可以非常简便的查看和使用第三方评估公司的评估报告，并以此对公司进行测评和管理。

（3）企业信息填报子系统：需要相应企业注册并登录后才可以进行相关操作，进入相应系统后，企业人员可以对自己本企业的信息进行录入工作，录入的企业信息将进入数据库以供相关政府人员查看评估。企业基本信息上报完成后，某些属于固定资产投资公司的企业要进入固定资产评估环节，等待固定资产评估员评估完成后对企业申报政策信息进行上报，不属于固定资产投资公司的企业可以直接对其企业所需要的政策进行上报，然后等待其审核完成。审批流程全部通过后，企业可以正式开始实施企业对应政策所需要的项目，待项目完成后，企业通过该系统上报完成情况，最终完成上报，企业方面的任务全部完成。

（4）企业信息管理子系统：高新区工作人员可以通过该系统对企业信息进行相应的管理，功能主要分为信息补录、查询、修改、删除、导出Excel文件，创业中心工作人员负责的主要工作是针对于企业信息的统计进行开展的，他们需要一系列的措施对企业的信息进行操作。所有由他们增、删、改、查的信息都将进入数据库，提供保存记录，也以便于上级领导查看和审阅。

（5）审阅统计管理子系统：其主要功能是为高新区领导提供的。高新区领导可以通过该子系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该子系统，高新区领导可以对整个高新区进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

### 3.5.1 企业信息管理子系统功能结构设计

如图3-5所示，企业信息管理子系统分为两大模块，分别是企业基本信息补录模块和企业基本信息管理模块。

企业基本信息补录：提供企业基本信息补录功能，包括企业名称、企业性质、企业负责人、企业联系电话、企业入驻时间、签订政策、签订时间、政策完成情况、亮灯情况等信息。支持附件上传功能。

企业基本信息管理：相关人员可通过企业名称、企业性质、企业负责人、企业入驻时间、签订政策、企业等级等条件对企业信息进行查询。支持对企业详情进行查看。在该部分中，管理人员可对企业基本信息进行管理，包括查询、修改、删除等操作。支持企业基本信息打印功能，支持企业基本信息导出成Excel表格功能。

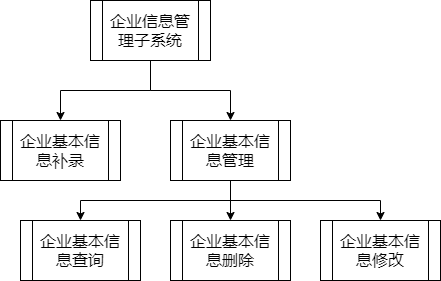


图3-5企业信息管理子系统功能结构图

### 3.5.2 企业信息填报子系统功能结构设计

如图3-6所示，企业信息填报子系统分为四大模块，分别是企业基本信息录入模块、政策申报模块、完成信息上报模块和审批进度查询查询。

企业基本信息录入：提供企业基本信息录入功能，由企业对企业信息进行录入，每个企业只能录入或修改自己企业的信息，无法查看到其他企业信息，包括企业名称、企业性质、企业负责人、企业联系电话、企业入驻时间、签订政策、签订时间、政策完成情况、亮灯情况等信息。支持附件上传功能；

政策申报：企业需要填写申请的政策信息，其信息包括账号、申请单位、企业地址、邮政编码、联系人、开户行等信息，并上传至数据库，等待政府审批人员进行审核；

完成情况上报：对应企业完成相应企业任务后，对完成情况进行上报，并上传至数据库，等待档案局工作人员归档。

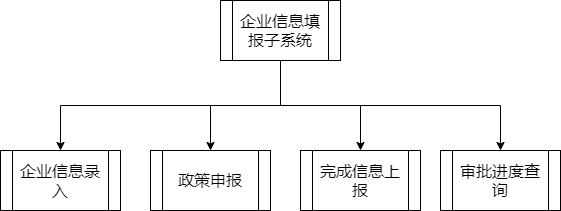


图3-6企业信息填报子系统功能结构图

### 3.5.3 招商引资档案管理子系统功能结构设计

如图3-7所示，招商引资档案管理子系统分为两大模块，分别是招商引资档案信息录入模块和招商引资档案信息管理模块。

档案信息录入：根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案录入；

档案信息管理：根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案统计。该部分分为三个小模块：查询、删除、修改。其中编辑模块功能是，管理人员可对归档的档案进行编辑、表单查看和流程查看操作。其中删除模块功能是，管理人员可对档案进行删除操作，删除后的档案无法恢复。

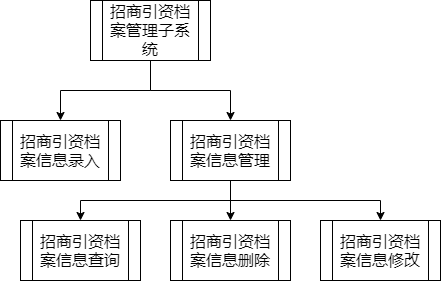


图3-7招商引资档案管理子系统功能结构图

### 3.5.4 固定资产投资评估管理子系统功能结构设计

如图3-8所示，固定资产评估管理管理子系统分为两大模块，分别是固定资产评估信息录入模块和固定资产评估信息管理模块。

评估信息录入：在新增固定资产投资评估记录时，只能选择新增固定资产投资评估管理系统中企业信息为企业基本信息中企业类型为“固定资产投资”类型企业的记录，企业基本信息在新增记录时不能修改，只能针对第三方评估记录进行信息新增，包括第三方评估公司、评估时间、评估结果、评估附件等信息；

评估信息管理：在该部分，融合了三个较小的功能模块，分别是查询、修改、删除。固定资产评估员可对固定资产评估记录进行修改，记录中企业基本信息不能修改，第三方评估公司、评估时间、评估结果、评估附件等信息可修改。另外管理人员可对固定资产评估记录进行删除，删除后该记录将不会在系统中显示。

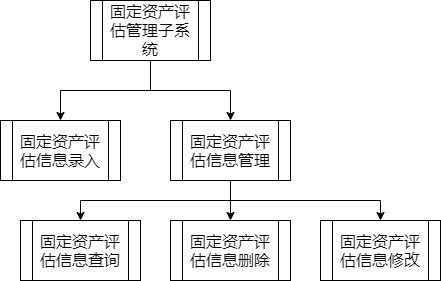


图3-8固定资产投资评估管理子系统功能结构图

### 3.5.5 领导审阅统计子系统功能结构设计

如图3-9所示，企业信息管理子系统分为两大模块，分别是统计分析查看信息录入模块和审批审阅信息管理模块。

审批审阅信息管理：高新区领导对政策审批员审批过的信息进行审阅，并对其进行审阅，高新区领导的可以对其进行驳回或通过，驳回的信息重新回到政策审批员手中，并重新审批。另外领导对不需要的审批审阅信息删除；

评估信息管理：提供统计分析功能，对企业政策管理服务综合平台中数据进行统计、分析，通过图表展现形式，为领导决策提供数据帮助。企业扶持政策兑现情况：按政策兑现类型、企业性质、政策兑现等情况进行统计分析，其中企业信息统计：按企业性质、是否贷款等情况进行统计分析。

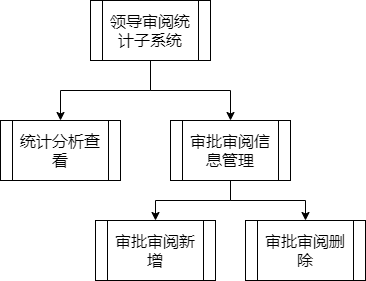


图3-9领导审阅统计子系统功能结构图

4 数据库设计

## 4.1 概念结构设计

### 4.1.1 设计思路

本系统中存在着7个实体，分别是：企业信息，申报政策信息，固定资产评估信息，政策审批信息，领导审阅信息，项目完成信息，招商引资档案信息。其中以企业信息为核心，一家企业对应着一份企业信息，而每一家企业只能申报一份政策信息。另外，属性为固定资产投资公司的企业相应要求一份固定资产评估信息，当其完成政策申报后，需要由政府审批人员对其申报的政策进行审批，然后生成一份审批信息，审批信息进入领导审阅环节，领导审阅后生成一份与之对应的领导审阅信息，项目政策的审阅审批环节正式结束，企业项目可以正式进入实施环节，当企业完成了申报政策所需到达到的标准，即项目完成后，可以填报企业完成信息并提交至政府有关部门进行审核，政府审核后进行与之对应的招商引资档案并归档收录。

上述为整个系统的流程概述，通过分析可以知道，如图4-1所示，在该系统中基本上所有的关系均为一对一关系，即一个企业信息对应一份申报政策信息，同时拥有一份固定资产评估信息、政策审批信息、领导审阅信息、项目完成信息和招商引资档案。数据库表关系较为单一，可以采用每张表的主键使用相同的字符串的方式来体现对应关系。

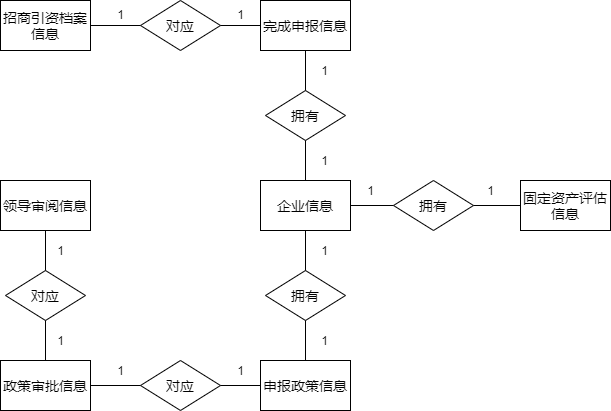


图4-1 总体E-R图

### 4.1.2 实体属性设计

如图4-2所示，企业信息表拥有18个字段，分别是ID、企业名称、企业注册地址、企业法人、企业责任人、企业电话、企业邮箱、注册资金、企业类型、企业注册时间、经营范围、企业等级、企业联系地址、企业邮编、开户银行、企业入驻时间、附件地址、添加账号。

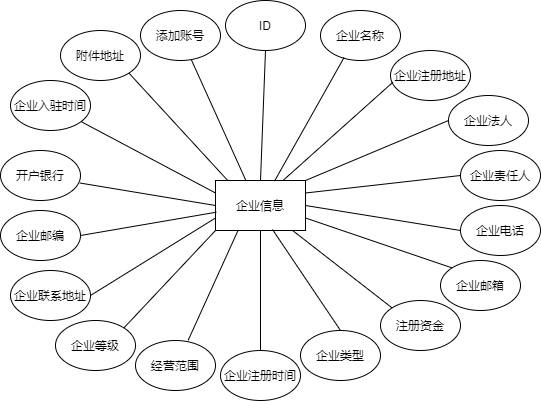


图4-2 企业信息实体图

如图4-3所示，企业信息表拥有18个字段，分别是Id、申请单位、企业地址、邮政编码、联系人、开户行、账号、资金类别、已获得扶持资金、申请时间、项目所在国（地区）、项目实施时间、项目负责人、电话、申请条例、申请扶持内容、申请扶持金额、附件。

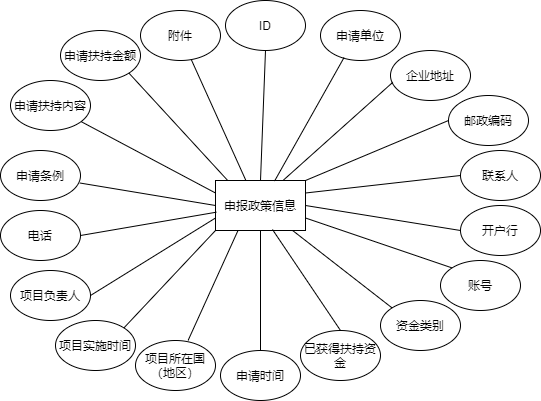


图4-3 申报政策信息实体图

如图4-4所示，企业信息表拥有6个字段，分别是档案ID、档案责任人、责任处室、档案名、入档时间、附件路径。

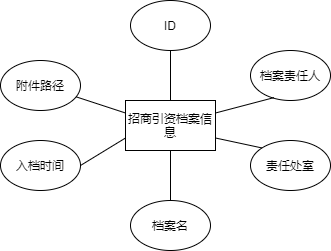


图4-4 招商引资档案信息实体图

如图4-5所示，企业信息表拥有15个字段，分别是评估ID、评估公司、负责人、公司成立时间、公司地址、组织形式、邮箱、电话、经办人姓名、经办人电话、经办人邮箱、经办人地址、评估时间、附件地址、添加账号。

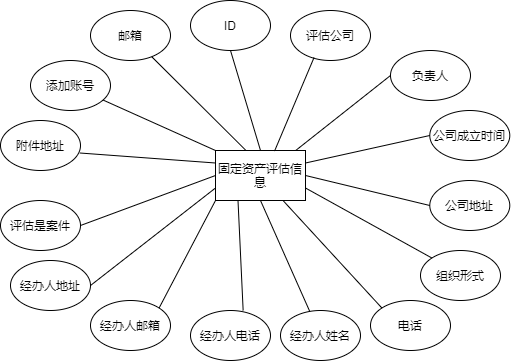


图4-5 固定值产评估信息实体图

如图4-6所示，企业信息表拥有10个字段，分别是审批id审批企业、经办人、责任人、联系电话、审批机关、审批时间、审批结果、审批意见、附件地址。

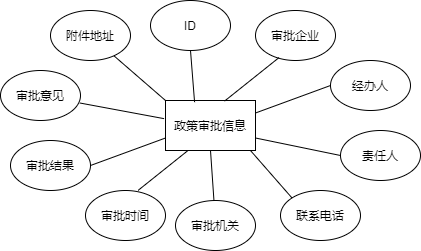


图4-6政策审批信息实体图

如图4-7所示，企业信息表拥有4个字段，分别是ID、审阅时间、审阅结果、领导评语。

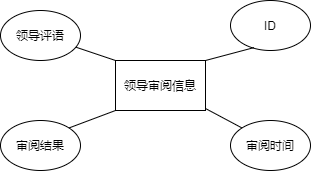


图4-7 领导审阅信息实体图

如图4-8所示，企业信息表拥有15个字段，分别是完成id、企业名称、企业法人、联系电话、企业类型、企业地址、申请政策、扶持金额、实施时间、完成时间、工作内容。

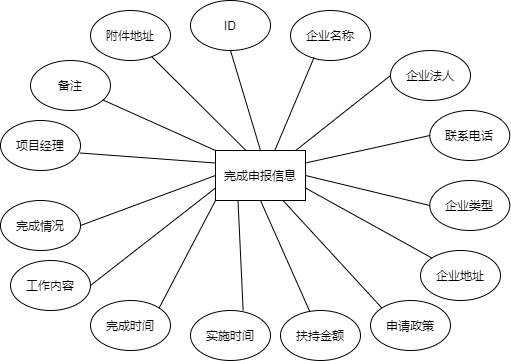


图4-8 完成情况申报信息实体图

## 4.2 逻辑结构设计

### 4.2.1 设计思路

根据实体属性设计可以发现，本系统总共需要7张表，其逻辑模型如下所示：

（1）企业信息：（ID，企业名称，企业注册地址，企业法人，企业责任人，企业电话，企业邮箱，注册资金，企业类型，企业注册时间，经营范围，企业等级，企业联系地址，企业邮编，开户银行，企业入驻时间，附件地址，添加账号）；

（2）固定资产评估信息：（评估ID，评估公司，负责人，公司成立时间，公司地址，组织形式，邮箱，电话，经办人姓名，经办人电话，经办人邮箱，经办人地址，评估时间，附件地址，添加账号）；

（3）完成申报信息：（完成id，企业名称，企业法人，联系电话，企业类型，企业地址，申请政策，扶持金额，实施时间，完成时间，工作内容，完成情况，项目经理，备注，附件地址）；

（4）招商引资档案信息：（档案ID，档案责任人，责任处室，档案名，入档时间，附件路径）；

（5）申报政策信息：（Id，申请单位，企业地址，邮政编码，联系人，开户行，账号，资金类别，已获得扶持资金，申请时间，项目所在国（地区），项目实施时间，项目负责人，电话，申请条例，申请扶持内容，申请扶持金额，附件）；

（6）政策审批信息：（审批id，审批企业，经办人，责任人，联系电话，审批机关，审批时间，审批结果，审批意见，附件地址）；

（7）领导审阅信息：（审阅ID，审阅时间，审阅结果，领导评语）

### 4.2.2 逻辑模型

（1）企业信息表，如表4-1所示

表4-1 企业信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | e\_id | 整型 | 11 | 是 | ID |
| 2 | e\_name | 字符串型 | 255 | 否 | 企业名称 |
| 3 | e\_regaddress | 字符串型 | 255 | 否 | 企业注册地址 |
| 4 | e\_legalman | 字符串型 | 255 | 否 | 企业法人 |
| 5 | e\_principal | 字符串型 | 255 | 否 | 企业责任人 |
| 6 | e\_phone | 字符串型 | 255 | 否 | 企业电话 |
| 7 | e\_email | 字符串型 | 255 | 否 | 企业邮箱 |
| 8 | e\_fund | 整型 | 255 | 否 | 注册资金 |
| 9 | e\_type | 字符串型 | 255 | 否 | 企业类型 |
| 10 | e\_signtime | 日期型 | 0 | 否 | 企业注册时间 |
| 11 | e\_scope | 字符串型 | 255 | 否 | 经营范围 |
| 12 | e\_rank | 字符串型 | 255 | 否 | 企业等级 |
| 13 | e\_conaddress | 字符串型 | 255 | 否 | 企业联系地址 |
| 14 | e\_postcode | 字符串型 | 255 | 否 | 企业邮编 |

续表4-1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 15 | e\_account | 字符串型 | 255 | 否 | 开户银行 |
| 16 | e\_intime | 日期型 | 0 | 否 | 企业入驻时间 |
| 17 | e\_path | 字符串型 | 255 | 否 | 附件地址 |
| 18 | username | 字符串型 | 255 | 否 | 添加账号 |

（2）固定资产评估信息表，如表4-2所示

表4-2 固定资产评估信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | a\_id | 整型 | 11 | 是 | 评估ID |
| 2 | a\_name | 字符串型 | 255 | 否 | 评估公司 |
| 3 | a\_boss | 字符串型 | 255 | 否 | 负责人 |
| 4 | a\_signtime | 日期型 | 0 | 否 | 公司成立时间 |
| 5 | a\_address | 字符串型 | 255 | 否 | 公司地址 |
| 6 | a\_type | 字符串型 | 255 | 否 | 组织形式 |
| 7 | a\_email | 字符串型 | 255 | 否 | 邮箱 |
| 8 | a\_phone | 字符串型 | 255 | 否 | 电话 |
| 9 | a\_o\_name | 字符串型 | 255 | 否 | 经办人姓名 |
| 10 | a\_o\_phone | 字符串型 | 255 | 否 | 经办人电话 |
| 11 | a\_o\_email | 字符串型 | 255 | 否 | 经办人邮箱 |
| 12 | a\_o\_address | 字符串型 | 255 | 否 | 经办人地址 |
| 13 | a\_time | 日期型 | 0 | 否 | 评估时间 |
| 14 | a\_path | 字符串型 | 255 | 否 | 附件地址 |
| 15 | username | 字符串型 | 255 | 否 | 添加账号 |

（3）招商引资档案信息表，如表4-3所示

表4-3 招商引资档案表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | f\_id | 整型 | 11 | 是 | 档案ID |
| 2 | f\_dutyofficer | 字符串型 | 255 | 否 | 档案责任人 |
| 3 | f\_response | 字符串型 | 255 | 否 | 责任处室 |
| 4 | f\_title | 字符串型 | 255 | 否 | 档案名 |
| 5 | f\_time | 日期型 | 0 | 否 | 入档时间 |
| 6 | f\_path | 字符串型 | 255 | 否 | 附件路径 |

（4）申报政策信息表，如表4-4所示

表4-4 申报政策信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | p\_id | 整型 | 11 | 是 | id |
| 2 | p\_enterprise | 字符串型 | 255 | 否 | 申请单位 |
| 3 | p\_address | 字符串型 | 255 | 否 | 企业地址 |
| 4 | p\_postcode | 字符串型 | 255 | 否 | 邮政编码 |
| 5 | p\_man | 字符串型 | 255 | 否 | 联系人 |
| 6 | p\_bank | 字符串型 | 255 | 否 | 开户行 |
| 7 | p\_account | 字符串型 | 255 | 否 | 账号 |
| 8 | p\_type | 字符串型 | 255 | 否 | 资金类别 |
| 9 | p\_gotmoney | 字符串型 | 255 | 否 | 已获得扶持资金 |
| 10 | p\_applytime | 日期型 | 0 | 否 | 申请时间 |
| 11 | p\_area | 字符串型 | 255 | 否 | 项目所在国（地区） |
| 12 | p\_starttime | 日期型 | 0 | 否 | 项目实施时间 |
| 13 | p\_principal | 字符串型 | 255 | 否 | 项目负责人 |
| 14 | p\_phone | 字符串型 | 255 | 否 | 电话 |
| 15 | p\_policy | 字符串型 | 255 | 否 | 申请条例 |
| 16 | p\_contain | 字符串型 | 1000 | 否 | 申请扶持内容 |
| 17 | p\_money | 整型 | 255 | 否 | 申请扶持金额 |
| 18 | p\_path | 字符串型 | 255 | 否 | 附件 |

（5）政策审批信息表，如表4-5所示

表4-5：政策审批信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | ex\_id | 整型 | 11 | 是 | 审批id |
| 2 | ex\_enterprise | 字符串型 | 255 | 否 | 审批企业 |
| 3 | ex\_man | 字符串型 | 255 | 否 | 经办人 |
| 4 | ex\_principal | 字符串型 | 255 | 否 | 责任人 |
| 5 | ex\_phone | 字符串型 | 255 | 否 | 联系电话 |
| 6 | ex\_unit | 字符串型 | 255 | 否 | 审批机关 |
| 7 | ex\_time | 日期型 | 0 | 否 | 审批时间 |
| 8 | ex\_result | 字符串型 | 255 | 否 | 审批结果 |
| 9 | ex\_opinion | 字符串型 | 1000 | 否 | 审批意见 |

（6）领导审阅信息表，如表4-6所示

表4-6：领导审阅信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | c\_id | 整型 | 11 | 是 | 审阅ID |
| 2 | c\_time | 日期型 | 0 | 否 | 审阅时间 |
| 3 | c\_result | 字符串型 | 255 | 否 | 审阅结果 |
| 4 | c\_remark | 字符串型 | 1000 | 否 | 领导评语 |

（7）完成申报信息表，如表4-7所示

表4-7：完成申报信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | com\_id | 整型 | 11 | 是 | 完成id |
| 2 | com\_enterprise | 字符串型 | 255 | 否 | 企业名称 |
| 3 | com\_man | 字符串型 | 255 | 否 | 企业法人 |
| 4 | com\_phone | 字符串型 | 255 | 否 | 联系电话 |
| 5 | com\_type | 字符串型 | 255 | 否 | 企业类型 |
| 6 | com\_address | 字符串型 | 255 | 否 | 企业地址 |
| 7 | com\_policy | 字符串型 | 255 | 否 | 申请政策 |
| 8 | com\_money | 整型 | 255 | 否 | 扶持金额 |
| 9 | com\_starttime | 日期型 | 0 | 否 | 实施时间 |
| 10 | com\_comtime | 日期型 | 0 | 否 | 完成时间 |
| 11 | com\_contain | 字符串型 | 255 | 否 | 工作内容 |
| 12 | com\_condition | 字符串型 | 1000 | 否 | 完成情况 |
| 13 | com\_manager | 字符串型 | 255 | 否 | 项目经理 |
| 14 | com\_remark | 字符串型 | 1000 | 否 | 备注 |
| 15 | com\_path | 字符串型 | 255 | 否 | 附件地址 |

## 4.3 物理结构设计

### 4.3.1 存取方式

本项目使用MySQL数据库。MySQL是一个关系型数据库管理系统，关联数据库将数据存储在不同的表中，而不是将所有的数据存储在一起，显著的提高了灵活性，也增加了速度。MySQL数据库拥有众多优点：

（1）运行速度快，MySQL体积小，命令执行的速度快；

（2）使用成本低。MySQL是开源的，且提供免费版本，对大多数用户来说大大降低了使用成本；

（3）使用容易。与其他大型数据库的设置和管理相比，其复杂程度较低，易于使用；

（4）可移植性强。MySQL能够运行与多种系统平台上，如windouws，Linux，Unix等；

（5）适用更多用户。MySQL支持最常用的数据管理功能，适用于中小型企业甚至大型网站应用。

### 4.3.2 DDL语言设计

（1）企业信息表

CREATE TABLE `enterprise` (

`e\_id` int(11) NOT NULL COMMENT 'ID',

`e\_name` varchar(255) DEFAULT NULL COMMENT '企业名称',

`e\_regaddress` varchar(255) DEFAULT NULL COMMENT '企业注册地址',

`e\_legalman` varchar(255) DEFAULT NULL COMMENT '企业法人',

`e\_principal` varchar(255) DEFAULT NULL COMMENT '企业责任人',

`e\_phone` varchar(255) DEFAULT NULL COMMENT '企业电话',

`e\_email` varchar(255) DEFAULT NULL COMMENT '企业邮箱',

`e\_fund` int(255) DEFAULT NULL COMMENT '注册资金',

`e\_type` varchar(255) DEFAULT NULL COMMENT '企业类型',

`e\_signtime` date DEFAULT NULL COMMENT '企业注册时间',

`e\_scope` varchar(255) DEFAULT NULL COMMENT '经营范围',

`e\_rank` varchar(255) DEFAULT NULL COMMENT '企业等级',

`e\_conaddress` varchar(255) DEFAULT NULL COMMENT '企业联系地址',

`e\_postcode` varchar(255) DEFAULT NULL COMMENT '企业邮编',

`e\_account` varchar(255) DEFAULT NULL COMMENT '开户银行',

`e\_intime` date DEFAULT NULL COMMENT '企业入驻时间',

`e\_path` varchar(255) DEFAULT NULL COMMENT '附件地址',

`username` varchar(255) DEFAULT NULL COMMENT '添加账号',

PRIMARY KEY (`e\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（2）固定资产评估信息表

CREATE TABLE `assess` (

`a\_id` int(11) NOT NULL COMMENT '评估ID',

`a\_name` varchar(255) DEFAULT NULL COMMENT '评估公司',

`a\_boss` varchar(255) DEFAULT NULL COMMENT '负责人',

`a\_signtime` date DEFAULT NULL COMMENT '公司成立时间',

`a\_address` varchar(255) DEFAULT NULL COMMENT '公司地址',

`a\_type` varchar(255) DEFAULT NULL COMMENT '组织形式',

`a\_email` varchar(255) DEFAULT NULL COMMENT '邮箱',

`a\_phone` varchar(255) DEFAULT NULL COMMENT '电话',

`a\_o\_name` varchar(255) DEFAULT NULL COMMENT '经办人姓名',

`a\_o\_phone` varchar(255) DEFAULT NULL COMMENT '经办人电话',

`a\_o\_email` varchar(255) DEFAULT NULL COMMENT '经办人邮箱',

`a\_o\_address` varchar(255) DEFAULT NULL COMMENT '经办人地址',

`a\_time` date DEFAULT NULL COMMENT '评估时间',

`a\_path` varchar(255) DEFAULT NULL COMMENT '附件地址',

`username` varchar(255) DEFAULT NULL COMMENT '添加账号',

PRIMARY KEY (`a\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（3）招商引资档案信息表

CREATE TABLE `file` (

`f\_id` int(11) NOT NULL COMMENT '档案ID',

`f\_dutyofficer` varchar(255) DEFAULT NULL COMMENT '档案责任人',

`f\_response` varchar(255) DEFAULT NULL COMMENT '责任处室',

`f\_title` varchar(255) DEFAULT NULL COMMENT '档案名',

`f\_time` date DEFAULT NULL COMMENT '入档时间',

`f\_path` varchar(255) DEFAULT NULL COMMENT '附件路径',

PRIMARY KEY (`f\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（4）申报政策信息表

CREATE TABLE `policy` (

`p\_id` int(11) NOT NULL COMMENT 'id',

`p\_enterprise` varchar(255) DEFAULT NULL COMMENT '申请单位',

`p\_address` varchar(255) DEFAULT NULL COMMENT '企业地址',

`p\_postcode` varchar(255) DEFAULT NULL COMMENT '邮政编码',

`p\_man` varchar(255) DEFAULT NULL COMMENT '联系人',

`p\_bank` varchar(255) DEFAULT NULL COMMENT '开户行',

`p\_account` varchar(255) DEFAULT NULL COMMENT '账号',

`p\_type` varchar(255) DEFAULT NULL COMMENT '资金类别',

`p\_gotmoney` varchar(255) DEFAULT NULL COMMENT '已获得扶持资金',

`p\_applytime` date DEFAULT NULL COMMENT '申请时间',

`p\_area` varchar(255) DEFAULT NULL COMMENT '项目所在国（地区）',

`p\_starttime` date DEFAULT NULL COMMENT '项目实施时间',

`p\_principal` varchar(255) DEFAULT NULL COMMENT '项目负责人',

`p\_phone` varchar(255) DEFAULT NULL COMMENT '电话',

`p\_policy` varchar(255) DEFAULT NULL COMMENT '申请条例',

`p\_contain` varchar(1000) DEFAULT NULL COMMENT '申请扶持内容',

`p\_money` int(255) DEFAULT NULL COMMENT '申请扶持金额',

`p\_path` varchar(255) DEFAULT NULL COMMENT '附件',

PRIMARY KEY (`p\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（5）政策审批信息表

CREATE TABLE `examine` (

`ex\_id` int(11) NOT NULL COMMENT '审批id',

`ex\_enterprise` varchar(255) DEFAULT NULL COMMENT '审批企业',

`ex\_man` varchar(255) DEFAULT NULL COMMENT '经办人',

`ex\_principal` varchar(255) DEFAULT NULL COMMENT '责任人',

`ex\_phone` varchar(255) DEFAULT NULL COMMENT '联系电话',

`ex\_unit` varchar(255) DEFAULT NULL COMMENT '审批机关',

`ex\_time` date DEFAULT NULL COMMENT '审批时间',

`ex\_result` varchar(255) DEFAULT NULL COMMENT '审批结果',

`ex\_opinion` varchar(1000) DEFAULT NULL COMMENT '审批意见',

`ex\_path` varchar(255) DEFAULT NULL COMMENT '附件地址',

PRIMARY KEY (`ex\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（6）领导审阅信息表

CREATE TABLE `check` (

`c\_id` int(11) NOT NULL COMMENT '审阅ID',

`c\_time` date DEFAULT NULL COMMENT '审阅时间',

`c\_result` varchar(255) DEFAULT NULL COMMENT '审阅结果',

`c\_remark` varchar(1000) DEFAULT NULL COMMENT '领导评语',

PRIMARY KEY (`c\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（7）完成申报信息表

CREATE TABLE `complete` (

`com\_id` int(11) NOT NULL COMMENT '完成id',

`com\_enterprise` varchar(255) DEFAULT NULL COMMENT '企业名称',

`com\_man` varchar(255) DEFAULT NULL COMMENT '企业法人',

`com\_phone` varchar(255) DEFAULT NULL COMMENT '联系电话',

`com\_type` varchar(255) DEFAULT NULL COMMENT '企业类型',

`com\_address` varchar(255) DEFAULT NULL COMMENT '企业地址',

`com\_policy` varchar(255) DEFAULT NULL COMMENT '申请政策',

`com\_money` int(255) DEFAULT NULL COMMENT '扶持金额',

`com\_starttime` date DEFAULT NULL COMMENT '实施时间',

`com\_comtime` date DEFAULT NULL COMMENT '完成时间',

`com\_contain` varchar(255) DEFAULT NULL COMMENT '工作内容',

`com\_condition` varchar(1000) DEFAULT NULL COMMENT '完成情况',

`com\_manager` varchar(255) DEFAULT NULL COMMENT '项目经理',

`com\_remark` varchar(1000) DEFAULT NULL COMMENT '备注',

`com\_path` varchar(255) DEFAULT NULL COMMENT '附件地址',

PRIMARY KEY (`com\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

5 界面设计

## 5.1 界面关系图

如图5-1所示，企业业务员通过注册以及相关角色审批后可以进入系统，并可以在工作台中选择功能，其中包括企业信息上传、政策申报、审批进度查询和完成情况上报等功能。

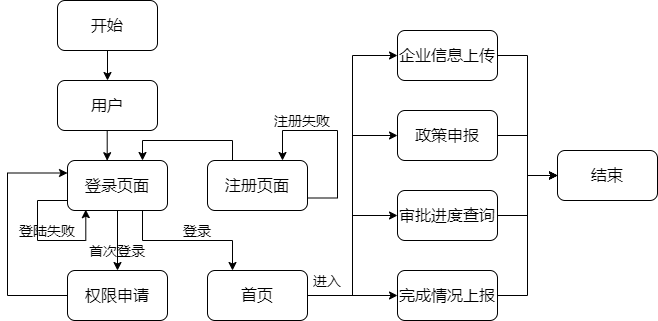


图5-1 企业业务员界面关系图

如图5-2所示，固定资产评估员通过注册以及相关角色审批后可以进入系统，并可以在工作台中选择功能，其中包括通讯录、邮件管理、固定资产评估等功能。

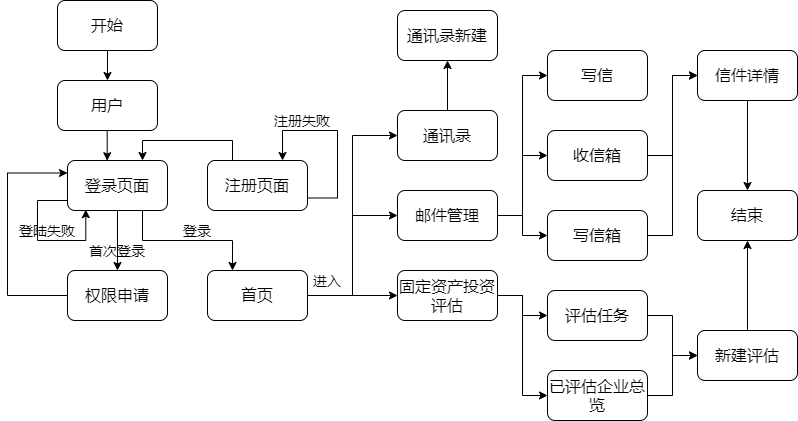


图5-2 固定资产评估员界面关系图

如图5-3所示，政策审批员通过注册以及相关角色审批后可以进入系统，并可以在工作台中选择功能，其中包括通讯录、邮件管理、扶持政策兑现审批等功能。

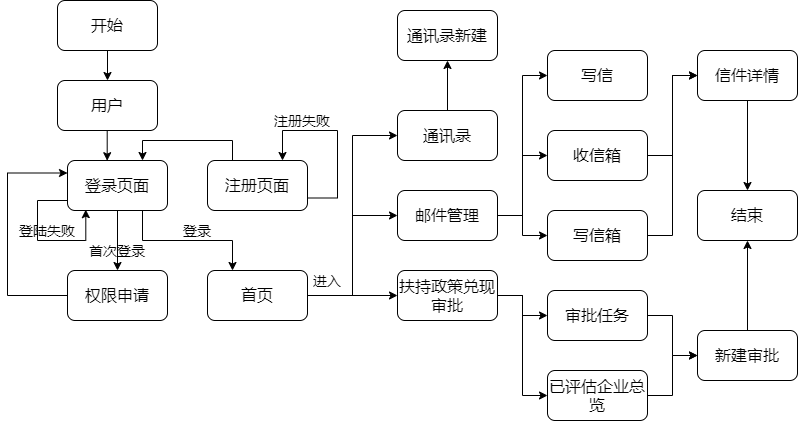


图5-3 政策审批员界面关系图

如图5-4所示，档案局人员通过注册以及相关角色审批后可以进入系统，并可以在工作台中选择功能，其中包括通讯录、邮件管理、招商引资档案管理等功能。

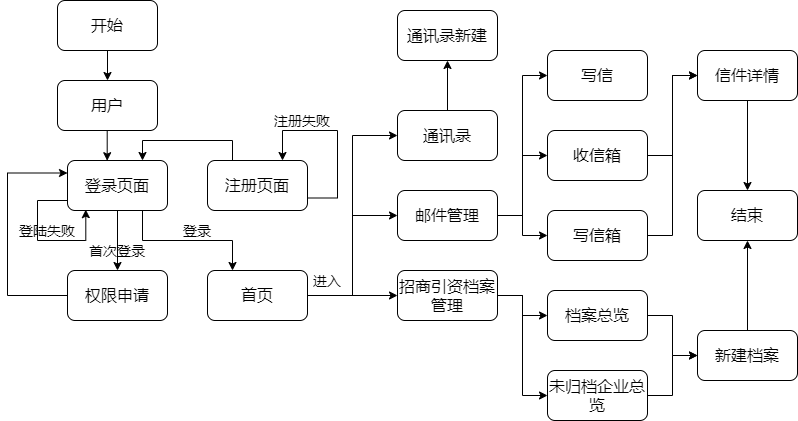


图5-4 档案局人员界面关系图

如图5-5所示，领导通过注册以及相关角色审批后可以进入系统，并可以在工作台中选择功能，其中包括通讯录、邮件管理、招商引资档案总览、已审批项目总览、在办业务总览、企业信息总览、档案总览、固定资产评估总览、审批审阅等功能。

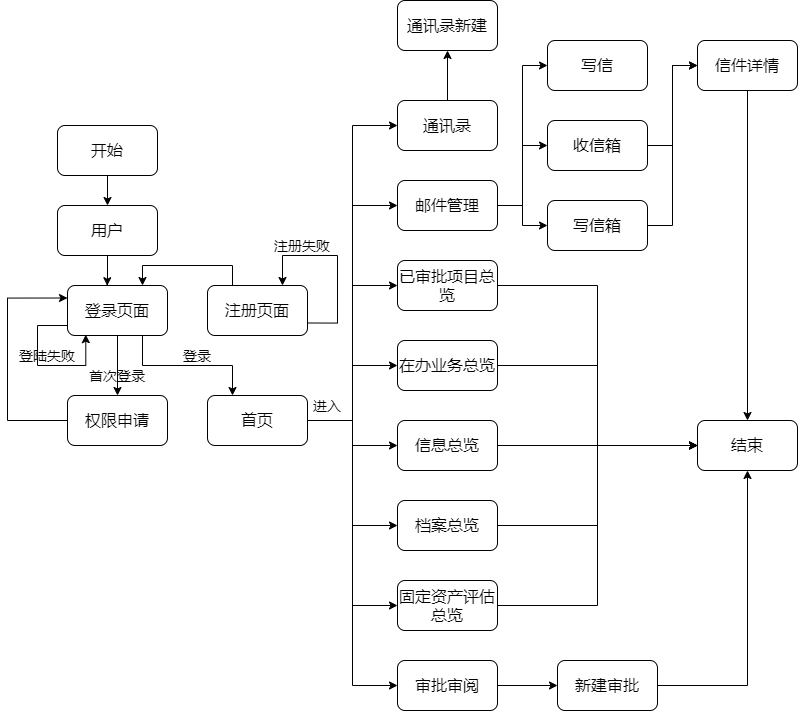


图5-5 领导界面关系图

## 5.2 界面设计成果

### 5.2.1 主界面

如图5-6所示，项目整体上部为顶部导航栏，左侧为主要的目录栏，针对不同的用户，该部分会展示不同的功能，用户通过点击左侧目录栏的功能选项进入功能，中间是主题部分，主要功能、表格、信息、表单等等组件都主要展示在这一个部分。



图5-6 模板详情分割功能图

### 5.2.2 子界面

如图5-7所示，为本项目的登录子界面，如果用户没有账号且需要注册，可以通过点击注册按钮进入注册界面，如图5-8所示，并支持点击按钮返回登录界面进行登录。

图5-7 登录子界面图 图5-8 注册子界面图

如图5-9所示，为本项目的主界面，政府工作人员（包括高新区工作人员、档案局人员、政策审批员、固定资产评估员、领导）可以对其页面进行相关浏览操作，并可以查看企业总体信息和相关的统计分析等等。

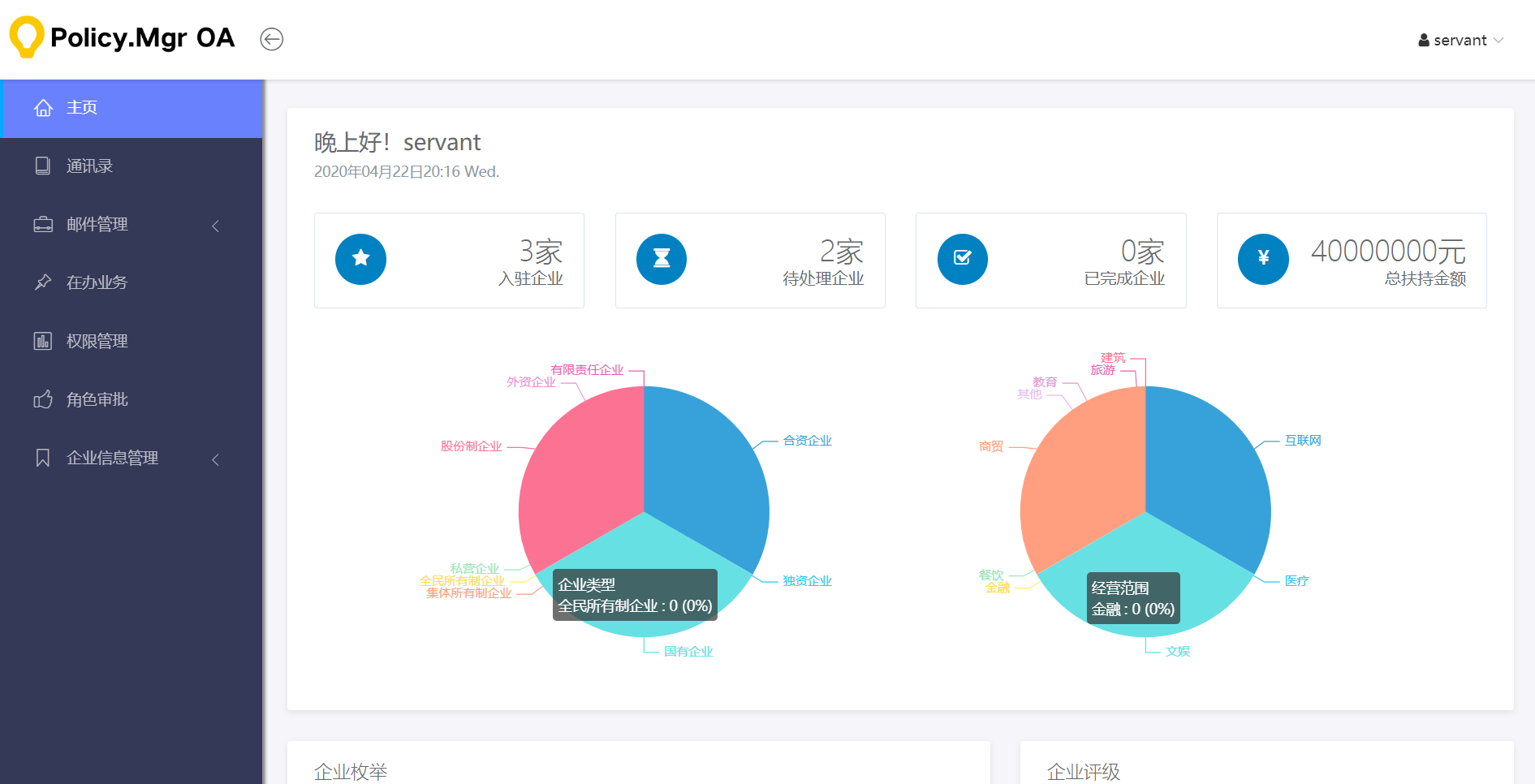


图5-9 主界面图

如图5-10所示，高新区工作人员通过该界面对整体企业信息进行浏览，可以通过表格右侧对相关信息进行有关操作，包括：详情、修改、删除的操作。

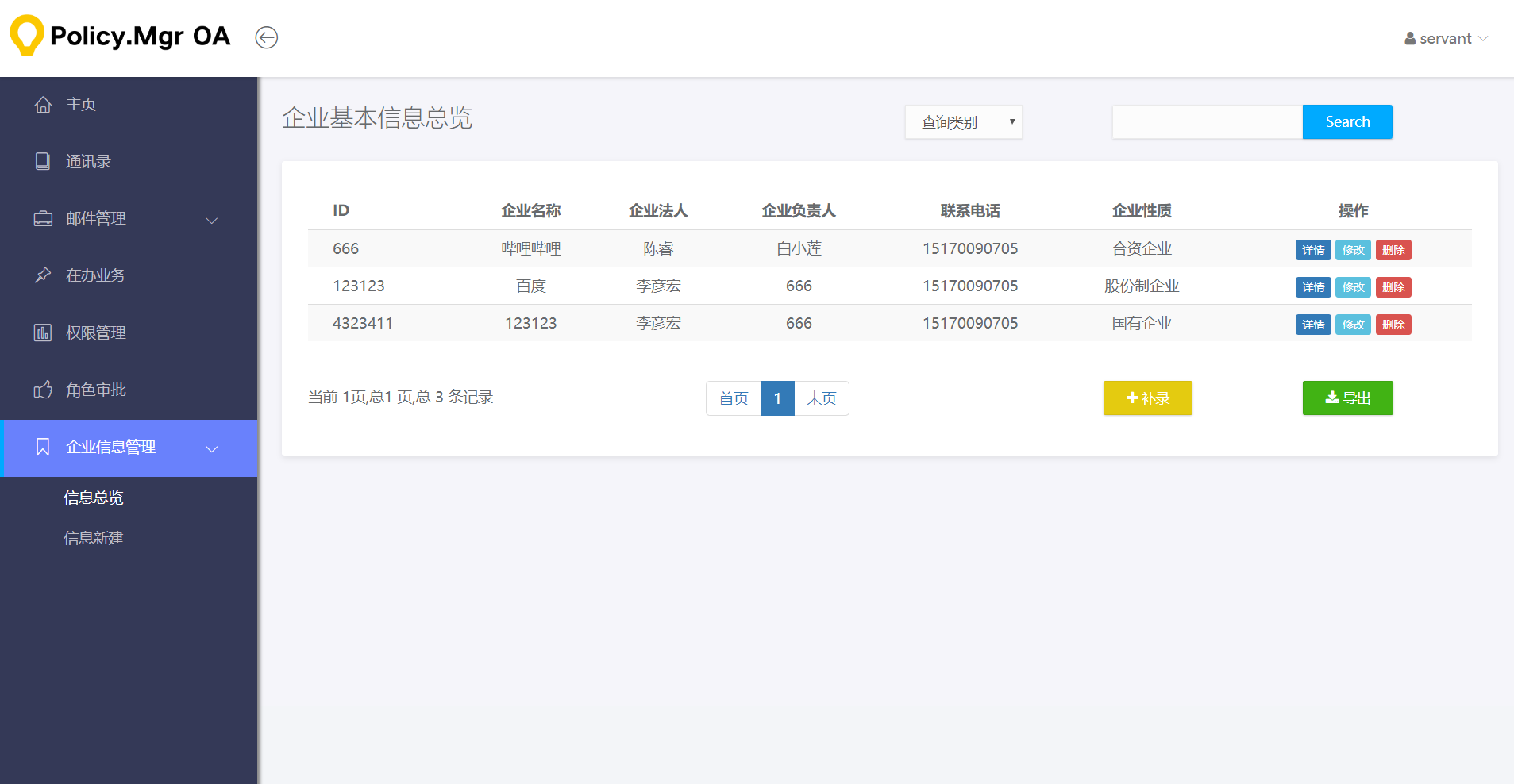


图5-10 企业信息总览子界面图

如图5-11所示，高新区管理人员可以对企业信息进行补录，通过填写表单点击按钮包装数据进入数据库。

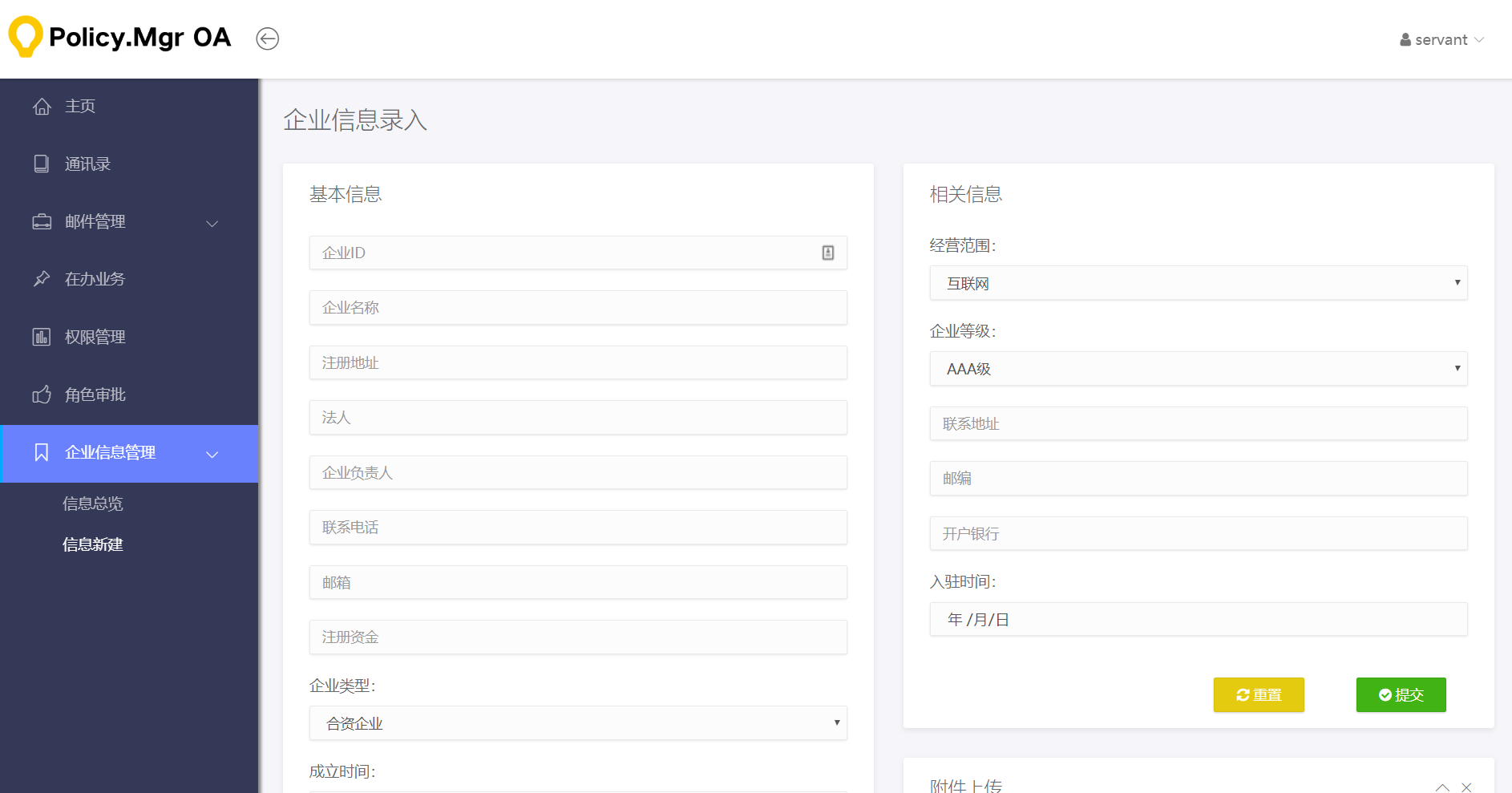


图5-11 企业信息补录子界面图

如图5-12所示，政策审批员可以通过该平台对政策进行审批，通过点击后进入功能界面，在左侧查看项目有关信息，在右侧可以填写审批信息点击按钮，分装数据后录入数据库。



图5-12 申请政策审批子界面图

如图5-13所示，用户可能会有一些不符合规范的操作，需要有操作界面用于提示用户操作错误的原因。

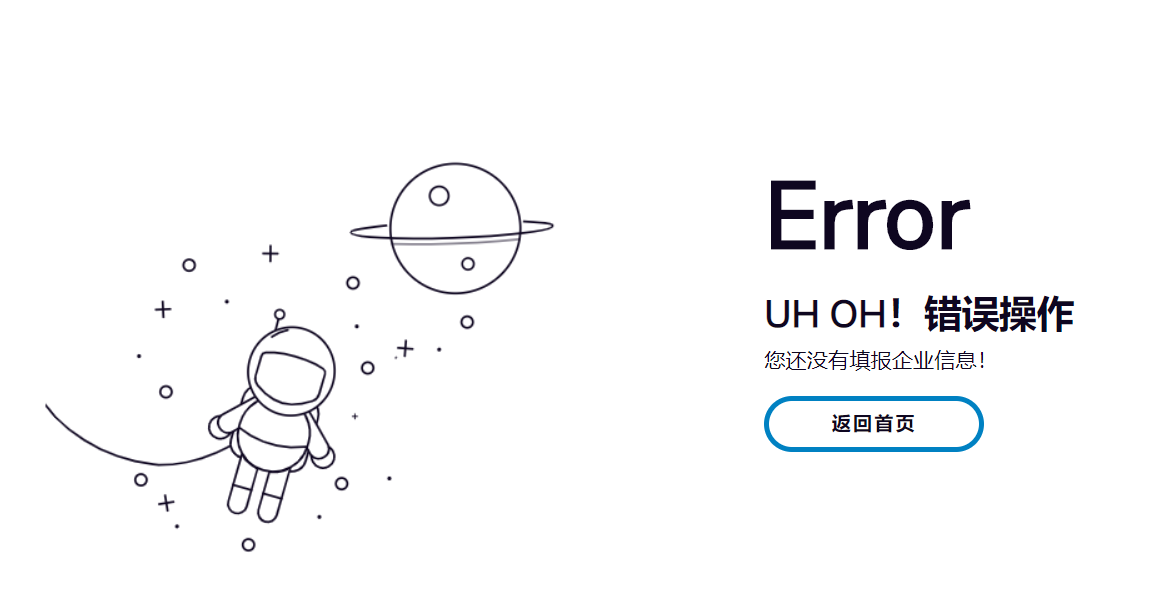


图5-13 操作错误界面图

如图5-14所示，政府工作人员（包括高新区工作人员、档案局人员、政策审批员、固定资产评估员、领导）可以通过该界面写邮件给对应用户，并可以在主体部分选择写信方式，可以选择普通邮件和群发邮件。

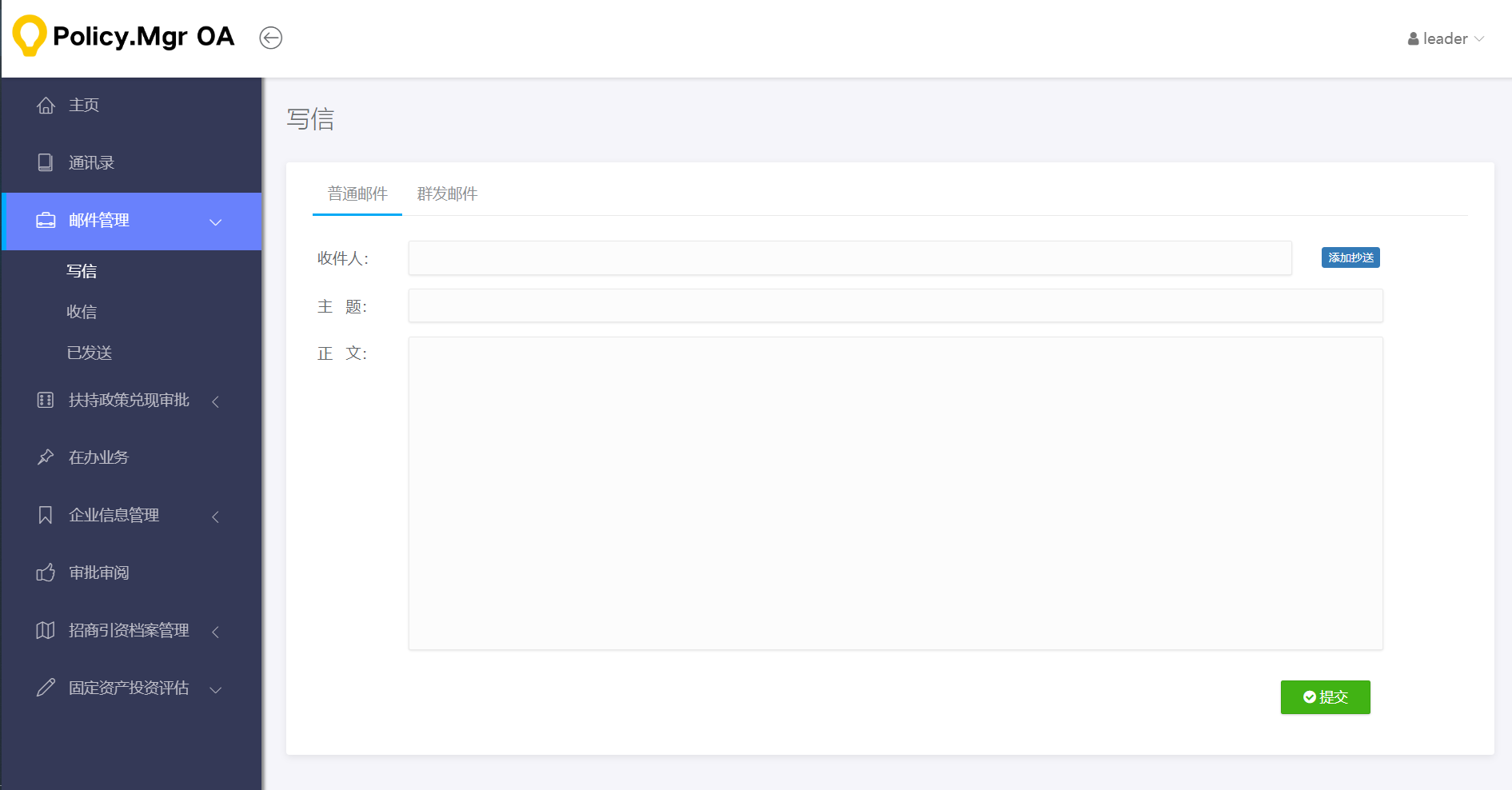


图5-14 写信子界面图

如图5-15所示，用户可以在这个界面查看个人信息并对自己的信息进行修改、任务的查看等等功能。

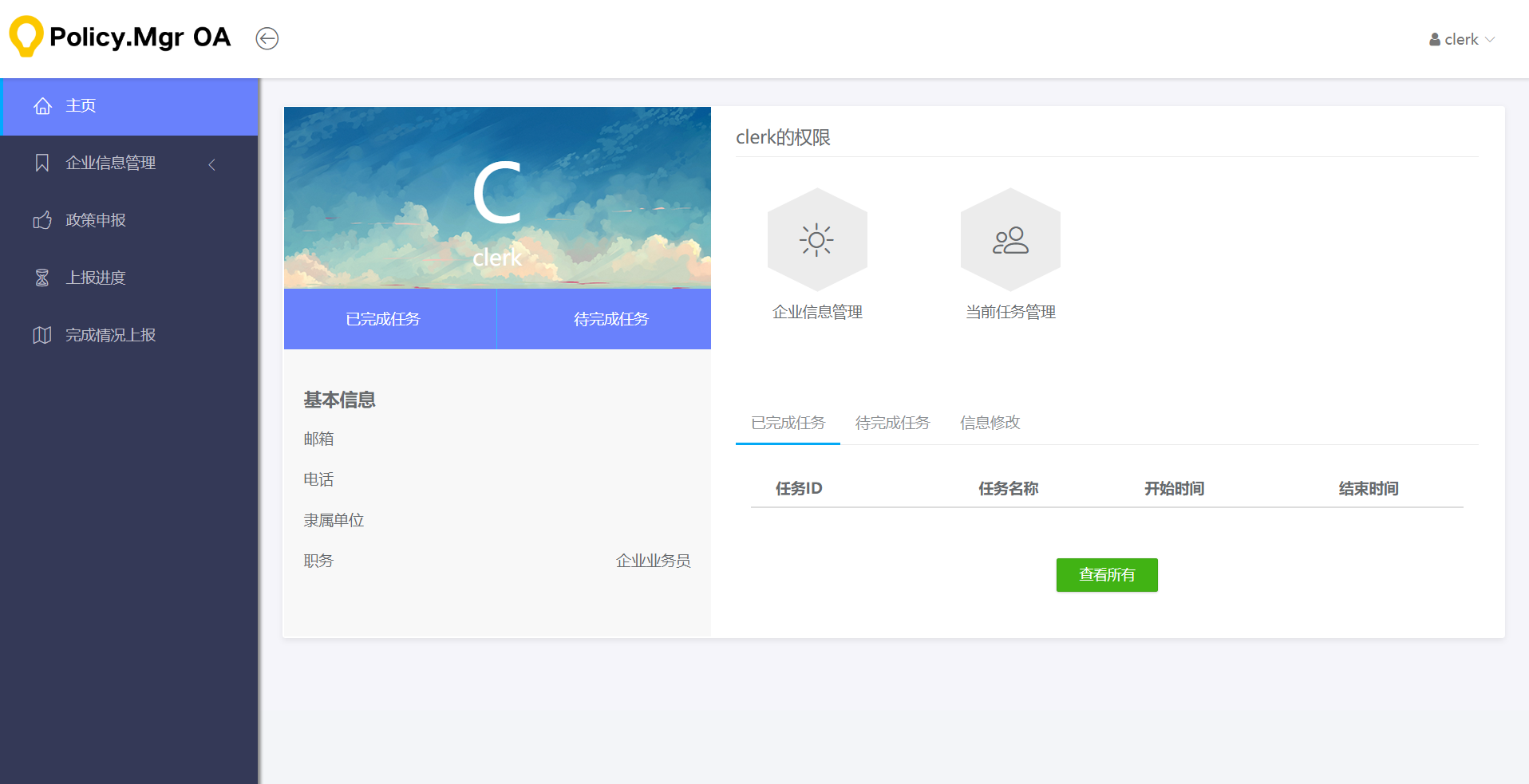


图5-15 个人信息界面图

如图5-16所示，企业业务员可以通过该界面查看上报进度，在该界面中，正在进行的步骤会显示为蓝色，另外在这个界面有催办按钮以供用户使用。

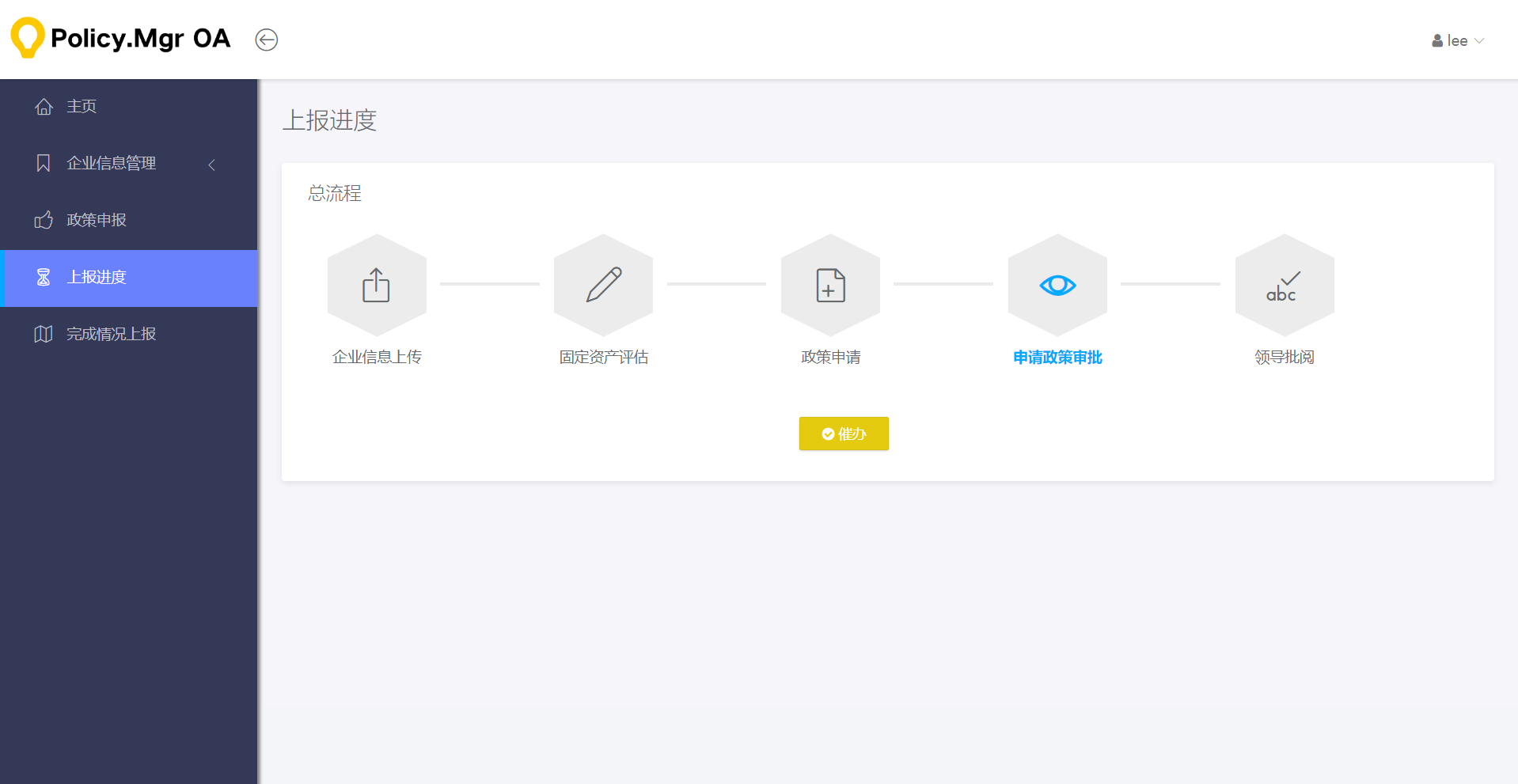


图5-16 上报进度查询界面图

6 详细设计

## 6.1 系统主要功能模块介绍

模块一，企业基本信息管理系统。企业基本信息管理系统包括企业基本信息补录、企业基本信息查询、企业基本信息管理等功能。

模块二，招商引资档案管理系统。根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。在档案管理中可通过责任者、责任处室、题名、时间等条件进行档案查询。本模块功能需要系统管理员设置相关权限才能够进行操作与查询。档案管理中的档案主要是企业扶持政策兑现审批中审批结束的档案，管理人员可对档案进行新建、编辑和删除等操作。

模块三，固定资产投资评估系统。提供固定资产投资评估管理系统，为固定资产投资公司建立第三方测评档案，方便高新区进行档案管理和投资公司管理。固定资产投资评估管理系统包括新增、修改、删除、查询、导出Excel表格等功能。

模块四，企业信息填报系统。企业业务员在该模块进行相关的业务操作，包括：企业信息上传，扶持政策申请，完成情况上报以及上报进度查询等功能。

模块五，领导审阅统计系统。高新区领导可以通过该子系统对企业政策信息进行相关审阅，如果有不合适的地方可以直接进行驳回，让相关政策审批员重新进行审批。另外通过该子系统，高新区领导进行相关数据统计分析，方便领导对整个高新区进行把控、决策。

## 6.2 功能模块设计

### 6.2.1企业基本信息管理

（1）功能结构设计

企业基本信息补录：提供企业基本信息补录功能，由企业对企业信息进行录入，每个企业只能录入或修改自己企业的信息，无法查看到其他企业信息，包括企业名称、企业性质、企业负责人等信息。支持附件上传功能。

企业基本信息统计：相关人员可通过企业名称、企业性质、企业负责人、企业入驻时间、签订政策、亮灯情况、企业等级等条件对企业信息进行查询。支持对企业详情进行查看。在该部分中，管理人员可对企业基本信息进行管理，包括查询、修改、删除等操作。支持企业基本信息打印功能，支持企业基本信息导出成Excel表格功能；

企业基本信息修改：提供企业基本信息修改功能，包括企业名称、企业性质、企业负责人、企业联系电话、企业入驻时间、签订政策、签订时间、政策完成情况、亮灯情况等信息。支持附件上传功能；

企业基本信息查看：企业人员登入系统后可以查看自己本企业的基本信息，如果有需求的情况下，可以进行下一个用例对本企业的基本信息进行修改。

企业基本信息管理如图6-1所示：

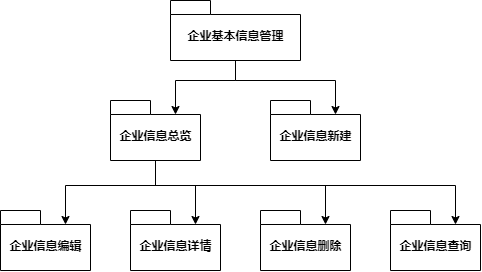


图6-1企业基本信息管理包图

（2）类设计

企业基本信息类图如图6-2所示：

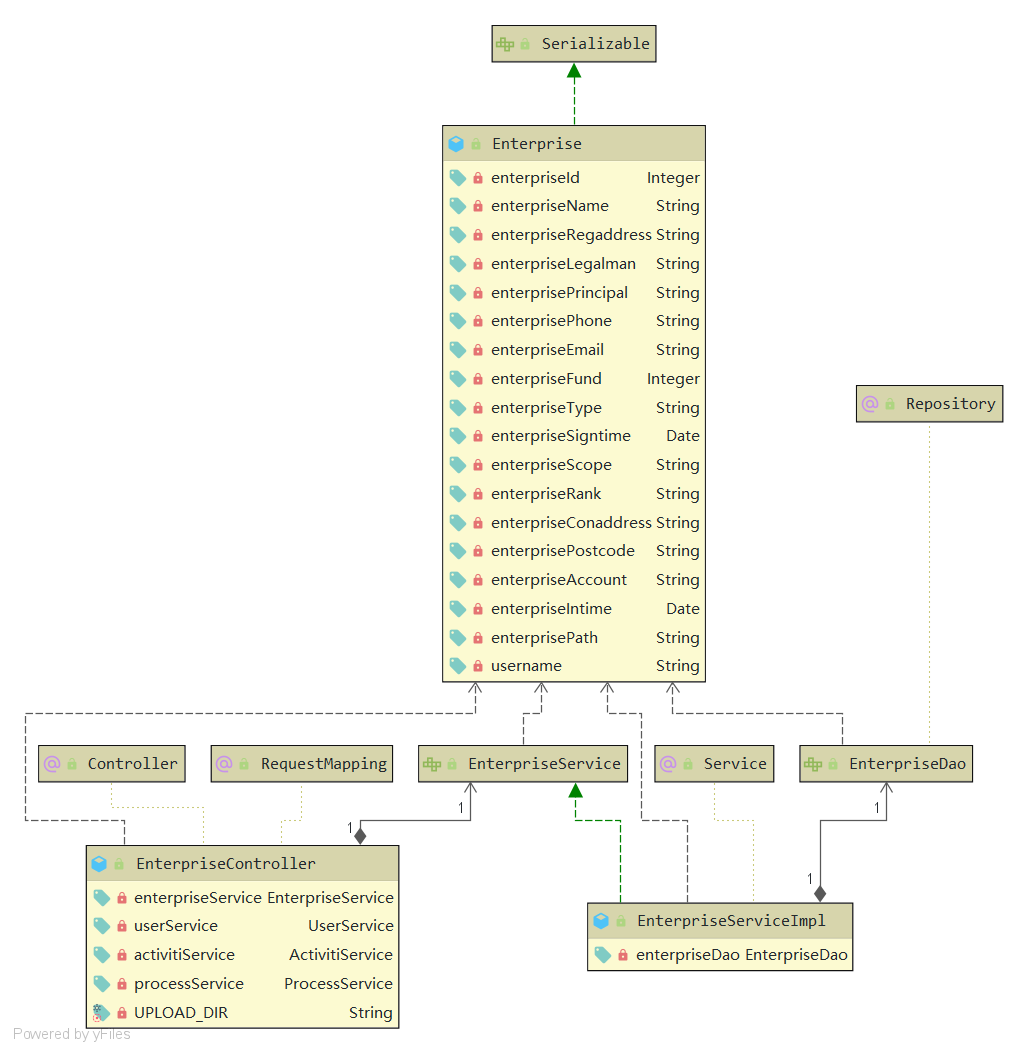


图6-2 企业基本信息类图

表6-1：企业基本信息实体类表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 类代号（code） | Enterprise | 类名称（中文） | | 企业基本信息实体类 | |
| 类功能描述 | 详细定义企业的基本信息，并使用Getter和Setter方法设置获取和设置方法 | | | | |
| 所在包名称 | com.policymanage.entity包 | | | | |
| 类属性 | 类型 | 属性名称 | 数据库段名 | | 其他 |
| enterpriseId | Integer | ID | e\_id | | 无 |
| enterpriseName | String | 企业名称 | e\_name | | 无 |
| enterpriseRegaddress | String | 企业注册地址 | e\_regaddress | | 无 |
| enterpriseLegalman | String | 企业法人 | e\_legalman | | 无 |
| enterprisePrincipal | String | 企业责任人 | e\_principal | | 无 |
| enterprisePhone | String | 企业电话 | e\_phone | | 无 |
| enterpriseEmail | String | 企业邮箱 | e\_email | | 无 |
| enterpriseFund | Integer | 注册资金 | e\_fund | | 无 |
| enterpriseType | String | 企业类型 | e\_type | | 无 |
| enterpriseSigntime | Date | 企业注册时间 | e\_signtime | | 无 |
| enterpriseScope | String | 经营范围 | e\_scope | | 无 |
| enterpriseRank | String | 企业等级 | e\_rank | | 无 |
| enterpriseConaddress | String | 企业联系地址 | e\_conaddress | | 无 |
| enterprisePostcode | String | 企业邮编 | e\_postcode | | 无 |
| enterpriseAccount | String | 开户银行 | e\_account | | 无 |
| enterpriseIntime | Date | 企业入驻时间 | e\_intime | | 无 |
| enterprisePath | String | 附件地址 | e\_path | | 无 |
| username | String | 添加账号 | username | | 无 |
| 主要实现方法 | Getter()方法，Setter()方法, toString()方法 | | | | |

表6-2 企业基本信息数据操作类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseDao | | 类名称（中文） | 企业基本信息数据操作类 |
| 类功能描述 | 对企业的基本信息进行相关的数据库操作 | | | |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有企业信息 |
| public | insertEnterprise | Enterprise | void | 添加企业信息 |
| public | findById | Integer | Enterprise | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |

续表6-2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | findById | Integer | Enterprise | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | updateEnterprise | Enterprise | void | 更新企业信息 |
| public | deletEnterprise | Integer | void | 删除企业信息 |
| public | findByName | String | List<Enterprise> | 通过Name查询 |
| public | findByType | String | List<Enterprise> | 通过Type查询 |
| public | findByPrincipal | String | List<Enterprise> | 通过Principal查询 |
| public | findByAccount | String | List<Enterprise> | 通过Account查询 |
| public | findByLegalman | String | List<Enterprise> | 通过Legalman查询 |
| public | findByRank | String | List<Enterprise> | 通过Rank查询 |
| public | findByInTime | Date | List<Enterprise> | 通过InTime查询 |
| public | findByUser | String | Enterprise | 通过User查询 |

表6-3 企业基本信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseService | | 类名称（中文） | 企业基本信息业务逻辑类 |
| 类功能描述 | 对企业的基本信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有企业信息 |
| public | insertEnterprise | Enterprise | void | 添加企业信息 |
| public | findById | Integer | Enterprise | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | updateEnterprise | Enterprise | void | 更新企业信息 |
| public | deletEnterprise | Integer | void | 删除企业信息 |
| public | queryEnterprise | Integer, String | List<Enterprise> | 根据下拉表单查询 |
| public | AddQuery | Integer | boolean | 添加信息判空 |
| public | findByUsername | String | Enterprise | 查询对应用户名 |

表6-4 企业基本信息控制器类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseController | | 类名称（中文） | 企业基本信息控制器类 |
| 类功能描述 | 对企业的基本信息进行相关的操作，负责处理分发的请求 | | | |
| 所在包名称 | com.policymanage.controller包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |

续表6-4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | getlist | ModelMap, Integer, Integer | String | 携带数据跳转总览界面 |
| public | skipEnterprise | 无 | String | 跳转至企业信息新增 |
| public | insertEnterprise | Enterprise | Message | 添加企业信息 |
| public | uploadFileById | MultipartFile, Integer | String | 上传文件 |
| public | getEnterprise | Integer, HttpServletRequest,  Model | String | 详情界面传值 |
| public | deleteEnterprise | Integer | String | 删除企业信息 |
| public | updateEnterprise | Enterprise | String | 编辑企业信息 |
| public | queryEnterprise | Integer, String, ModelMap | String | 查询展示数据分页列表 |
| public | skipDashboard | 无 | String | 跳转回总览界面 |
| public | excelDown | HttpServletResponse, HttpServletRequest | String | 下载企业信息转为Excel |

（3）顺序图设计

企业信息添加时序图如图6-3所示：

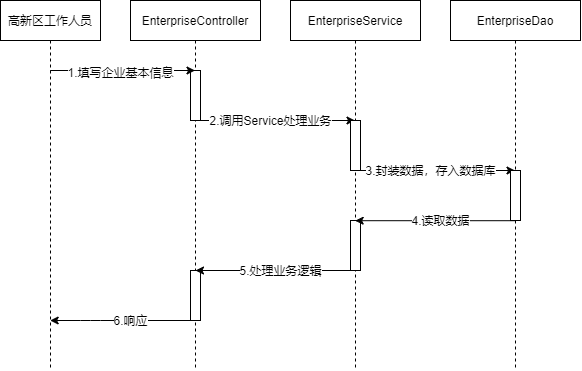


图6-3 添加企业信息时序图

企业信息查询时序图如图6-4所示：

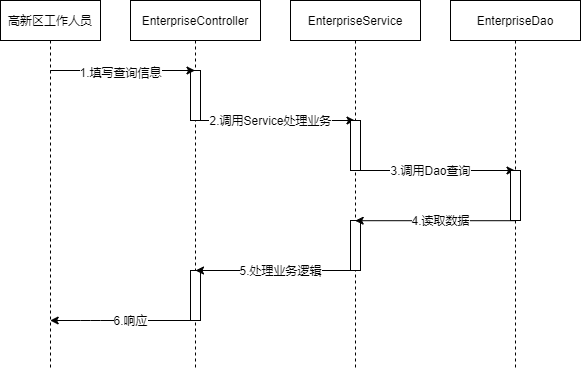


图6-4 查询企业信息时序图

企业信息修改时序图如图6-5所示：

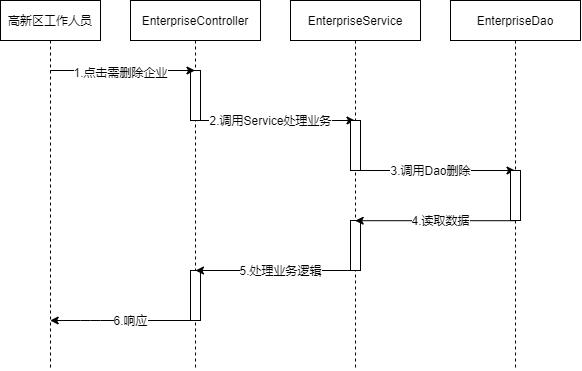


图6-5 修改企业信息时序图

（4）核心处理流程设计

企业基本信息补录流程如图6-6所示：

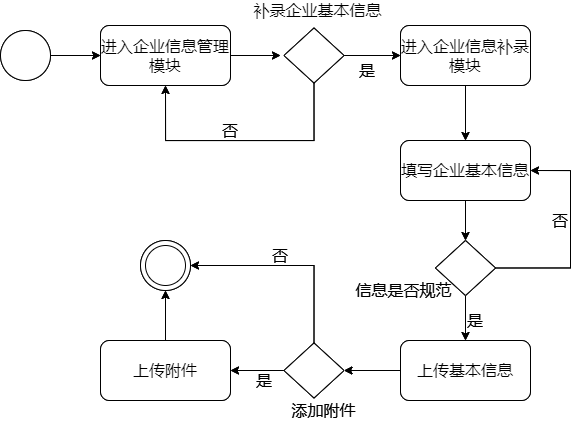


图6-6企业基本信息补录活动图

企业基本信息管理流程如图6-7所示：

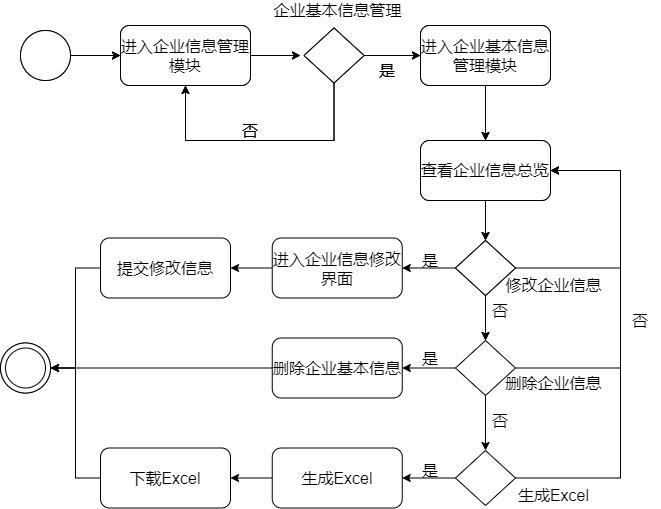


图6-7企业基本信息管理活动图

企业基本信息查询流程如图6-8所示：

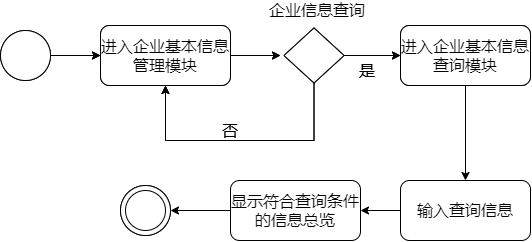


图6-8企业基本信息查询活动图

### 6.2.2招商引资档案管理

（1）功能结构设计

档案信息录入：根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。

档案信息查看：根据高新区实际情况建立电子档案数据库，实现各类企业扶持政策兑现处理过程档案管理。该部分分为三个小模块：查询、删除、修改；

档案信息删除：管理人员可对档案进行删除操作，删除后的档案无法恢复；

档案信息编辑：管理人员可对归档的档案进行编辑、表单查看和流程查看操作。

招商引资信息管理如图6-9所示：

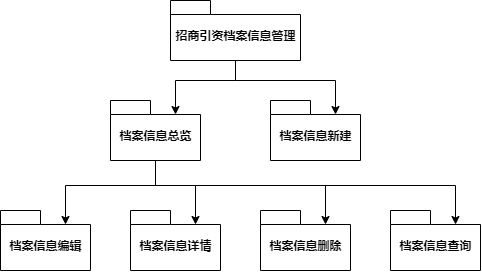


图6-9招商引资信息管理包图

（2）类图设计

招商引资信息管理类图如图6-10所示：

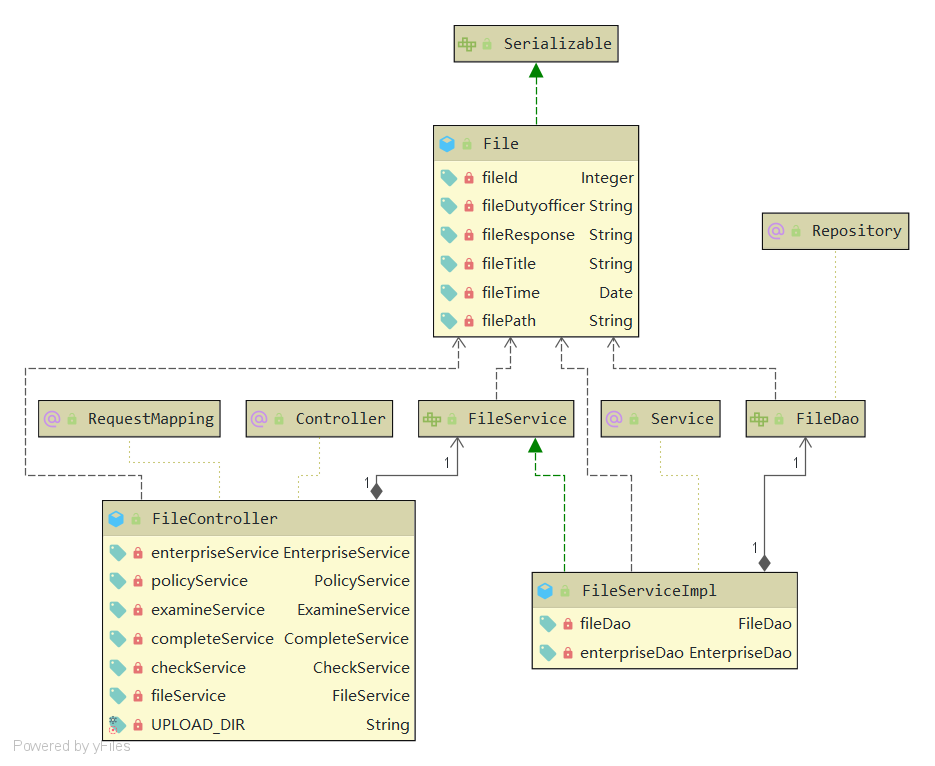


图6-10招商引资信息管理类图

表6-5 招商引资档案信息实体类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类代号（code） | File | | 类名称（中文） | 招商引资档案信息实体类 | | |
| 类功能描述 | 详细定义招商引资档案的基本信息，并使用Getter和Setter方法设置获取和设置方法 | | | | | |
| 所在包名称 | com.policymanage.entity包 | | | | | |
| 类属性 | 类型 | 属性名称 | | | 数据库段名 | 其他 |
| fileId | Integer | 档案id | | | f\_id | 无 |
| fileDutyofficer | String | 档案责任人 | | | f\_dutyofficer | 无 |
| fileResponse | String | 档案责任处室名 | | | f\_response | 无 |
| fileTitle | String | 档案题名 | | | f\_title | 无 |
| fileTime | Date | 档案时间 | | | f\_time | 无 |
| filePath | String | 附件链接 | | | f\_path | 无 |
| 主要实现方法 | Getter()方法，Setter()方法, toString()方法 | | | | | |

表6-6 招商引资档案信息数据操作类表

|  |  |  |  |
| --- | --- | --- | --- |
| 类代号 | FileDao | 类名称（中文） | 招商引资档案信息数据操作类 |
| 类功能描述 | 对招商引资档案的基本信息进行相关的数据库操作 | | |

续表6-6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<File> | 查询所有档案信息 |
| public | insertFile | File | void | 添加档案信息 |
| public | findById | Integer | File | 通过ID查询 |
| public | updateFile | File | void | 更新档案信息 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | deletFile | Integer | void | 删除档案信息 |
| public | findByDutyofficer | String | List< File > | 通过责任人查询 |
| public | findByResponse | String | List< File > | 通过Response查询 |
| public | findByTitle | String | List< File > | 通过Title查询 |
| public | findByTime | Date | List< File > | 通过Time查询 |

表6-7 招商引资档案信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | FileService | | 类名称（中文） | 招商引资档案信息业务逻辑类 |
| 类功能描述 | 对招商引资档案的基本信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有档案信息 |
| public | findEnterpriseById | Integer | File | 根据ID查询企业 |
| public | insertEnterprise | File | void | 添加档案信息 |
| public | findById | Integer | File | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | updateFile | File | void | 更新档案信息 |
| public | deletFile | Integer | void | 删除档案信息 |
| public | queryFile | Integer, String | List<File> | 根据下拉表单查询 |
| public | AddqueryEnterprise | Integer | boolean | 添加信息判空 |
| public | AddQueryFile | Integer | boolean | 添加信息判空 |

表6-8 招商引资档案信息控制器类表

|  |  |  |  |
| --- | --- | --- | --- |
| 类代号 | FileController | 类名称（中文） | 招商引资档案信息控制器类 |

续表6-8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类功能描述 | 对招商引资档案的基本信息进行相关的操作，负责处理分发的请求 | | | |
| 所在包名称 | com.policymanage.controller包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | skipToAddFile | Integer, Model | String | 携带数据跳转新增 |
| public | getlist | ModelMap, Integer,Integer | String | 携带数据跳转总览 |
| public | insertFile | File | Message | 添加档案信息 |
| public | downloadFile | Integer,HttpServletRequest,HttpServletRequest | void | 下载档案信息 |
| public | uploadFileById | MultipartFile, Integer | String | 上传文件 |
| public | getFile | Integer,HttpServletRequest,Model | String | 详情界面传值 |
| public | deleteFile | Integer | String | 删除档案信息 |
| public | editFile | File, Model | String | 携带数据跳转修改 |
| public | updateFile | File | String | 编辑档案信息 |
| public | queryFile | Integer, String, ModelMap | String | 查询展示数据分页列表 |
| public | skipDashboard | 无 | String | 跳转回总览界面 |

（3）顺序图设计

删除档案时序图如图6-11所示：

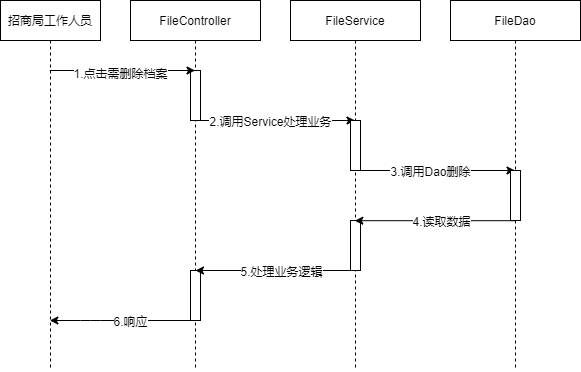


图6-11 删除档案信息时序图

新增档案时序图如图6-12所示：

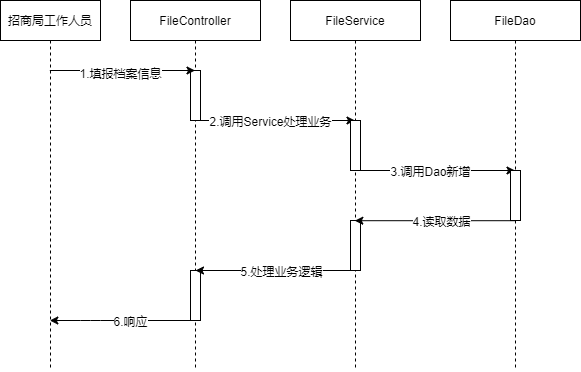


图6-12 新增档案信息时序图

新增档案时序图如图6-12所示：

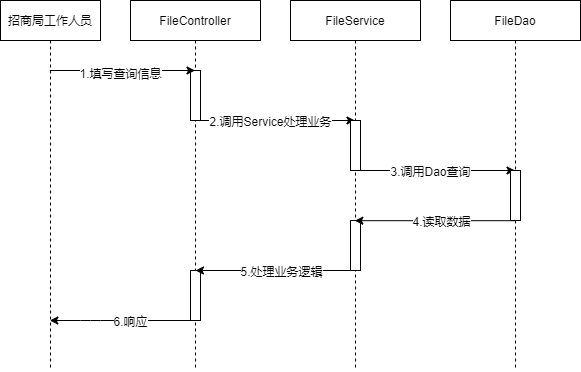


图6-13查询档案信息时序图

（4）核心处理流程设计

招商引资档案信息录入流程如图6-14所示：

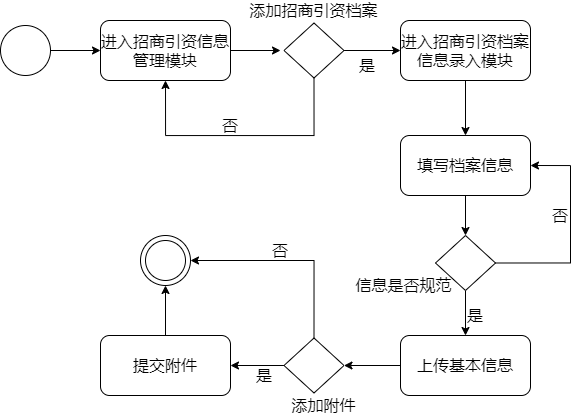


图6-14 招商引资档案信息录入活动图

招商引资档案信息管理流程如图6-15所示：

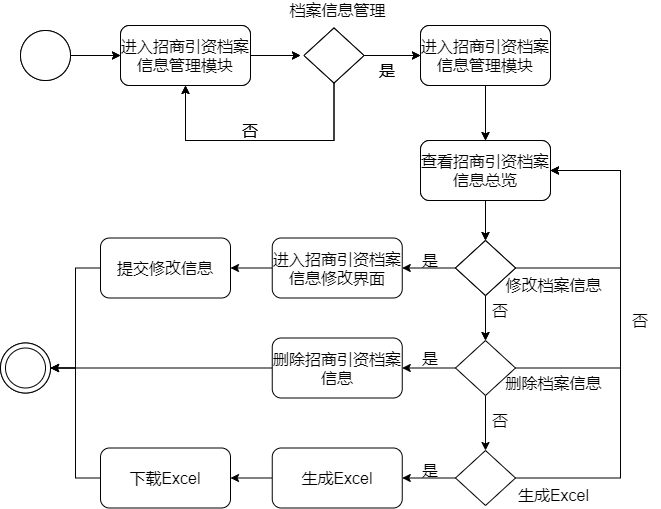


图6-15 招商引资档案信息管理活动图

招商引资档案信息查询、导出流程如图6-16所示：

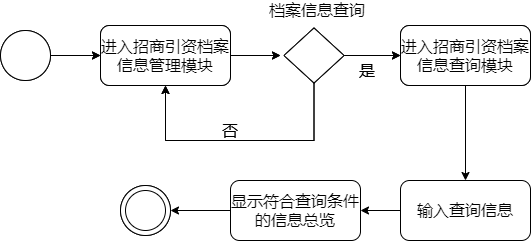


图6-16 招商引资档案信息查询、导出活动图

### 6.2.3固定资产评估信息管理

（1）功能结构设计

固定资产评估如图6-17所示：

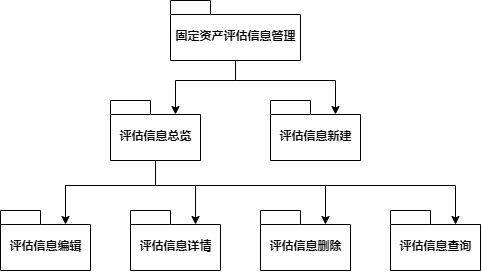


图6-17 固定资产评估包图

（2）类图设计

固定资产评估类图如图6-18所示：

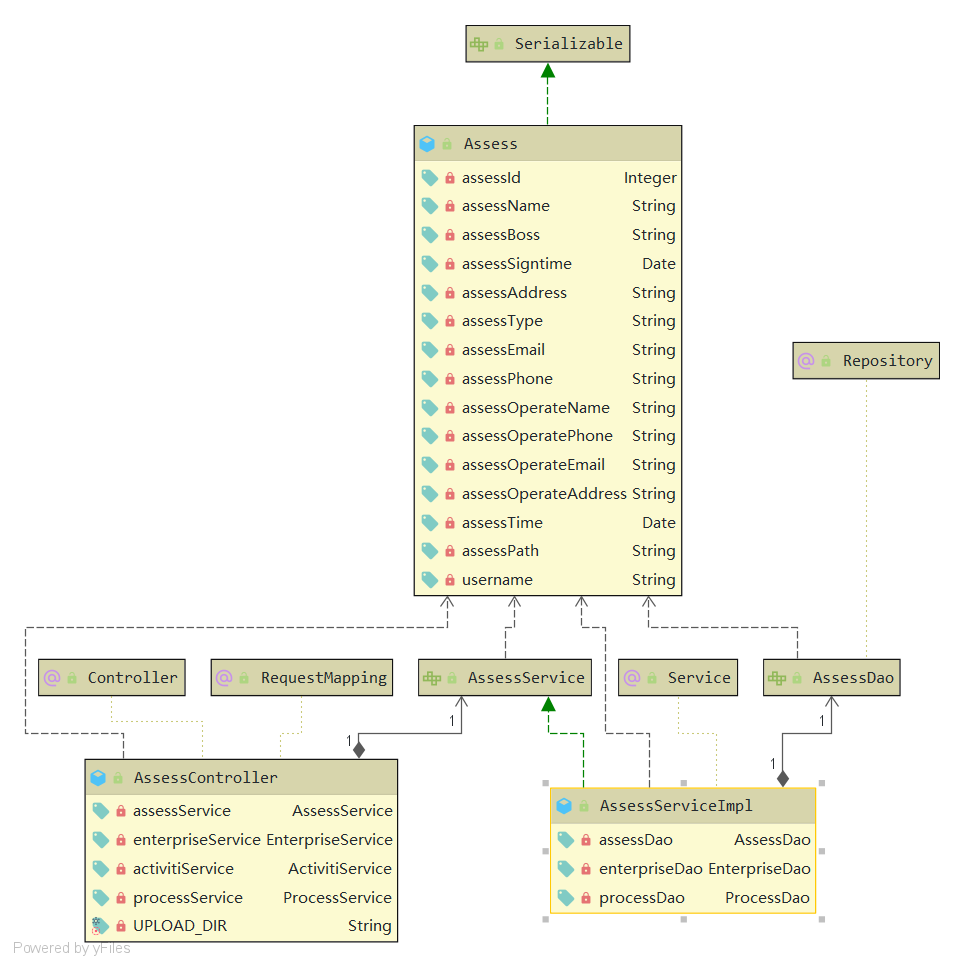


图6-18 固定资产评估类图

表6-9 固定资产评估信息实体类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类代号（code） | Assess | | 类名称（中文） | 固定资产评估信息实体类 | | |
| 类功能描述 | 详细固定资产评估企业的基本信息，并使用Getter和Setter方法设置获取和设置方法 | | | | | |
| 所在包名称 | com.policymanage.entity包 | | | | | |
| 类属性 | 类型 | 属性名称 | | | 数据库段名 | 其他 |
| assessId | Integer | 评估ID | | | a\_id | 无 |
| assessName | String | 评估公司 | | | a\_name | 无 |
| assessBoss | String | 负责人 | | | a\_boss | 无 |
| assessSigntime | Date | 公司成立时间 | | | a\_signtime | 无 |
| assessAddress | String | 公司地址 | | | a\_address | 无 |
| assessType | String | 组织形式 | | | a\_type | 无 |
| assessEmail | String | 邮箱 | | | a\_email | 无 |
| assessPhone | String | 电话 | | | a\_phone | 无 |
| assessOperateName | String | 经办人姓名 | | | a\_o\_name | 无 |
| assessOperatePhone | String | 经办人电话 | | | a\_o\_phone | 无 |
| assessOperateEmail | String | 经办人邮箱 | | | a\_o\_email | 无 |
| assessOperateAddress | String | 经办人地址 | | | a\_o\_address | 无 |
| assessTime | Date | 评估时间 | | | a\_time | 无 |
| assessPath | String | 附件地址 | | | a\_path | 无 |
| username | String | 添加账号 | | | username | 无 |
| 主要实现方法 | Getter()方法，Setter()方法, toString()方法 | | | | | |

表6-10 固定资产评估信息数据操作类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | AssessDao | | 类名称（中文） | 固定资产评估信息数据操作类 |
| 类功能描述 | 对固定资产评估的基本信息进行相关的数据库操作 | | | |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Assess> | 查询所有评估信息 |
| public | insertAssess | File | void | 添加评估信息 |
| public | findById | Integer | Assess | 通过ID查询 |
| public | updateAssess | File | void | 更新档案信息 |
| public | updatePathById | String, Integer | void | 上传附件链接 |

续表6-10

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | deleAssess | Integer | void | 删除评估信息 |
| public | findByName | String | List< Assess> | 通过Name查询 |
| public | findByTime | String | List< Assess> | 通过Time查询 |

表6-11 固定资产评估信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | AssessService | | 类名称（中文） | 固定资产评估信息业务逻辑类 |
| 类功能描述 | 对固定资产评估的基本信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有信息 |
| public | findEnterpriseById | Integer | Enterprise | 根据ID查询评估 |
| public | insertAssess | File | void | 添加评估信息 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | findById | Integer | Assess | 根据ID查询 |
| public | updateAssess | Assess | void | 更新评估信息 |
| public | deletAssess | Integer | void | 删除评估信息 |
| public | queryAssess | Integer, String | List<Assess> | 根据表单查询 |
| public | queryByTimeBucket | String, String | List<Assess> | 根据时间段查询 |
| public | AddQueryEnterprise | Integer | boolean | 判断企业存在 |
| public | AddQueryAssess | Intteger | boolean | 判断评估存在 |
| public | findExistButAssess | 无 | List<Enterprise> | 查询已有信息但没有评估的企业 |
| public | findMyAssess | String | List<Enterprise > | 查询评估任务 |

表6-12 固定资产评估信息控制器类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | AssessController | | 类名称（中文） | 固定资产评估信息控制器类 |
| 类功能描述 | 对固定资产评估的基本信息进行相关的操作，负责处理分发的请求 | | | |
| 所在包名称 | com.policymanage.controller包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | getlist | ModelMap, Integer, Integer | String | 携带数据跳转总览界面 |
| public | skipAssess | 无 | String | 跳转至评估新增 |

续表6-12

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | insertAssess | File | Message | 添加评估信息 |
| public | uploadFileById | MultipartFile, Integer | String | 上传文件 |
| public | getFile | Integer,HttpServletRequest | String | 详情界面传值 |
| public | deleteAssess | Integer | String | 删除评估信息 |
| public | editAssess | File, Model | String | 携带数据跳转修改界面 |
| public | updateAssess | File | String | 编辑评估信息 |
| public | queryEnterprise | Integer, String | String | 查询展示数据分页列表 |
| public | skipDashboard | 无 | String | 跳转回总览界面 |
| public | queryByTimeBucket | String, String, Model | String | 时间段查询并携带数据 |
| public | getToEvaluate | ModelMap, Integer, Integer | String | 跳转至待评估企业列表 |
| public | assessEnterprise | Enterprise, Model | String | 携带数据跳转评估界面 |
| public | excelDown | HttpServletResponse,HttpServletRequest | String | 下载所有企业信息为Excel |

（3）顺序图设计

查询评估信息时序图如图6-19所示：

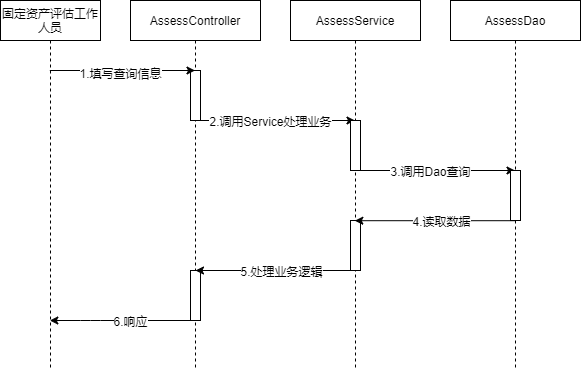


图6-19 查询评估信息时序图

删除评估信息时序图如图6-20所示：

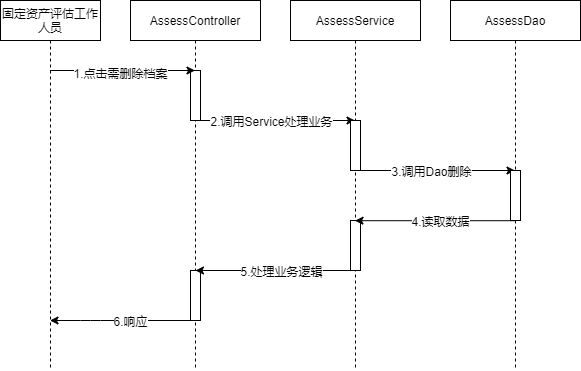


图6-20 删除评估信息时序图

修改评估信息时序图如图6-21所示：

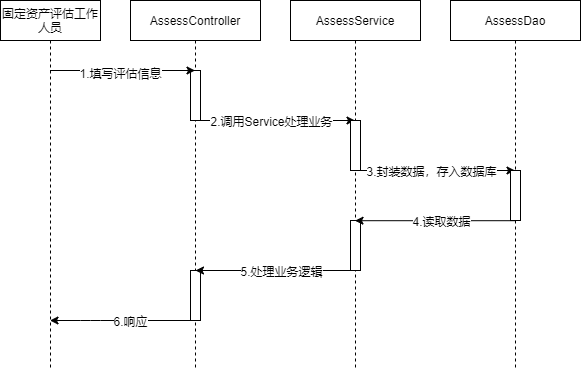


图6-21 修改评估信息时序图

（4）核心处理流程设计

固定资产评估信息新增流程如图6-22所示：

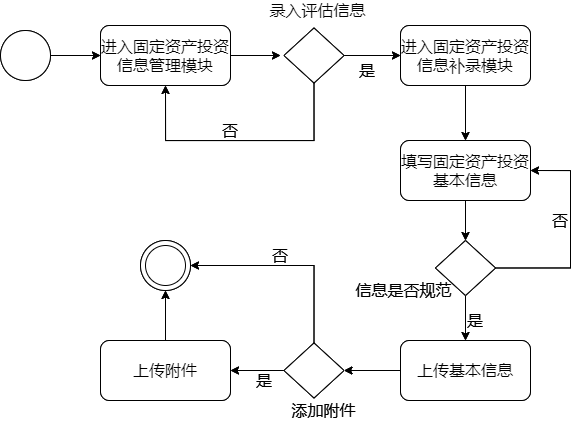


图6-22 固定资产投资评估信息新增活动图

固定资产评估信息编辑流程如图6-24所示：

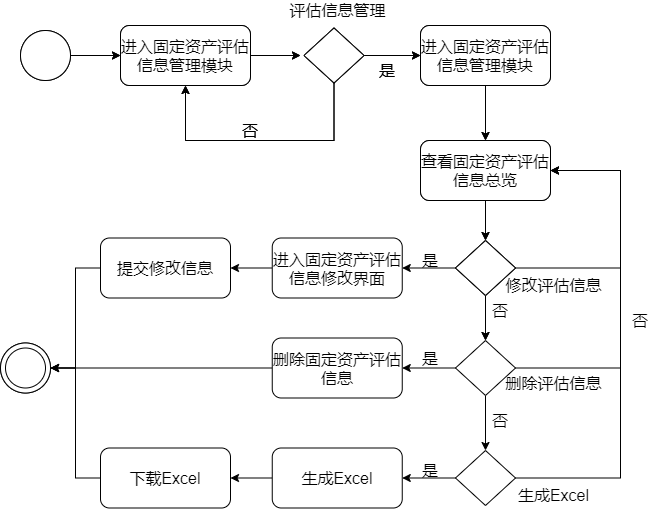


图6-24固定资产投资评估信息编辑活动图

固定资产评估信息查询、导出流程如图6-25所示：

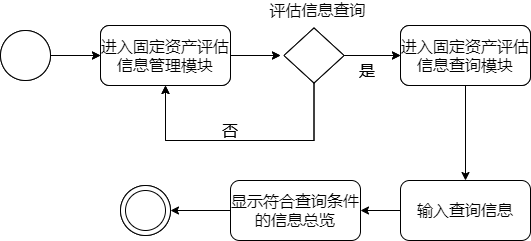


图6-23 固定资产投资评估信息查询、导出活动图

### 6.2.4领导审阅统计管理

（1）功能结构设计

审批审阅信息管理：高新区领导对政策审批员审批过的信息进行审阅，并对其进行审阅，高新区领导的可以对其进行驳回或通过，驳回的信息重新回到政策审批员手中，并重新审批。另外领导对不需要的审批审阅信息删除；

评估信息管理：提供统计分析功能，对企业政策管理服务综合平台中数据进行统计、分析，通过图表展现形式，为领导决策提供数据帮助。

领导审阅统计管理如图6-25所示：

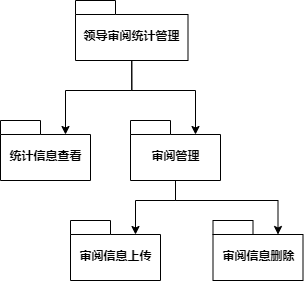


图6-25 领导审阅统计管理包图

（2）类图设计

领导审阅统计类图如图6-26所示：

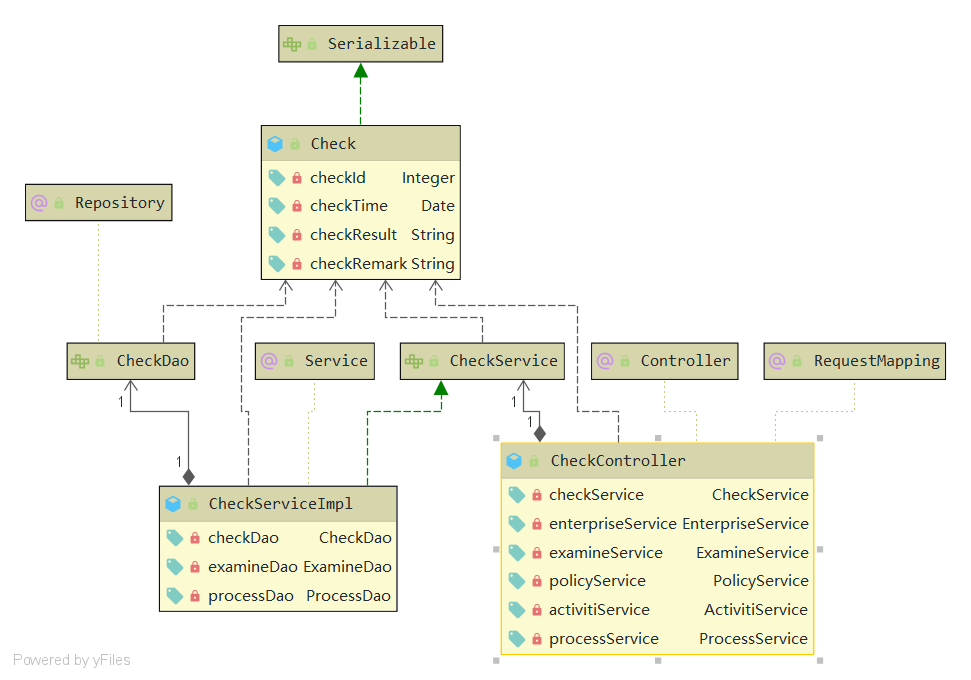


图6-26 领导审阅统计管理类图

表6-13 领导审阅信息实体类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类代号（code） | Check | | 类名称（中文） | 领导审阅信息实体类 | | |
| 类功能描述 | 详细定义领导审阅信息，并使用Getter和Setter方法设置获取和设置方法 | | | | | |
| 所在包名称 | com.policymanage.entity包 | | | | | |
| 类属性 | 类型 | 属性名称 | | | 数据库段名 | 其他 |
| checkId | Integer | 审阅ID | | | c\_id | 无 |
| checkTime | Date | 审阅时间 | | | c\_time | 无 |
| checkResult | String | 审阅结果 | | | c\_result | 无 |
| checkRemark | String | 领导评语 | | | c\_remark | 无 |

表6-14 领导审阅信息数据操作类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | CheckDao | | 类名称（中文） | 领导审阅信息数据操作类 |
| 类功能描述 | 对领导审阅信息信息进行相关的数据库操作 | | | |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Check> | 查询所有审阅信息 |
| public | insertCheck | Check | void | 添加审阅信息 |
| public | findById | Integer | Check | 通过ID查询 |
| public | updateCheck | Check | void | 更新审阅信息 |

表6-15 领导审阅信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | CheckService | | 类名称（中文） | 领导审阅信息业务逻辑类 |
| 类功能描述 | 对领导审阅信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Check> | 查询所有信息 |
| public | findAllNoCheck | 无 | List<Examine> | 查询未审阅信息 |
| public | insertCheck | Check | void | 添加评估信息 |
| public | updateCheck | Check | void | 更新审阅信息 |
| public | findById | Integer | Check | 根据ID查询 |
| public | findAllNone | 无 | List<Examine> | 查询领导未审阅 |

表6-16 领导审阅信息控制器类表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 类代号 | CheckController | | 类名称（中文） | | 领导审阅信息控制器类 |
| 类功能描述 | 对领导审阅信息进行相关的操作，负责处理分发的请求 | | | | |
| 所在包名称 | com.policymanage.controller包 | | | | |
| 方法属性 | 方法名 | 输入 | | 输出 | 方法功能描述 |
| public | getlist | ModelMap, Integer, Integer | | String | 携带数据跳转总览界面 |
| public | insertCheck | Integer, HttpServletRequest, check | | String | 获取所有企业审批申请情况 |
| public | insertAssess | Check | | void | 添加审阅信息 |

（3）顺序图设计

添加审阅信息时序图如图6-27所示：

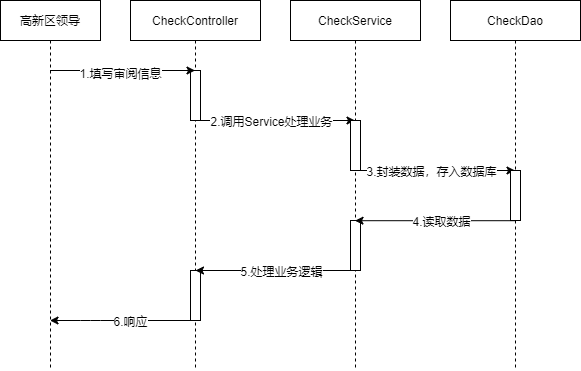


图6-27 添加审阅信息时序图

领导查询审阅统计时序图如图6-28所示：

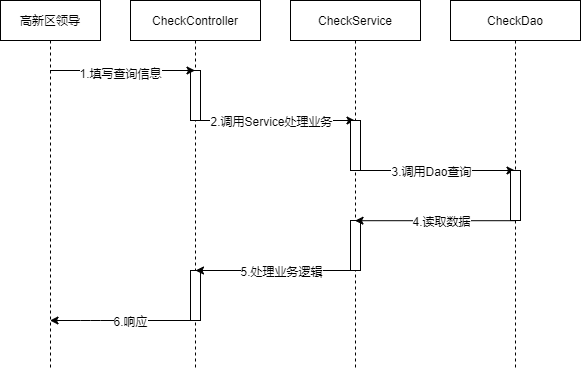


图6-28 查询审阅信息时序图

（4）核心处理流程设计

审阅信息新增流程如图6-29所示：

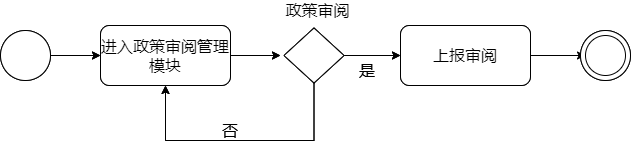


图6-29 审阅信息新增活动图

统计分析流程如图6-30所示：

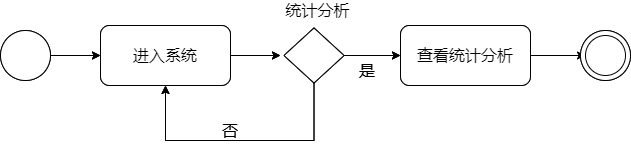


图6-30 统计分析活动图

审阅信息管理流程如图6-31所示：

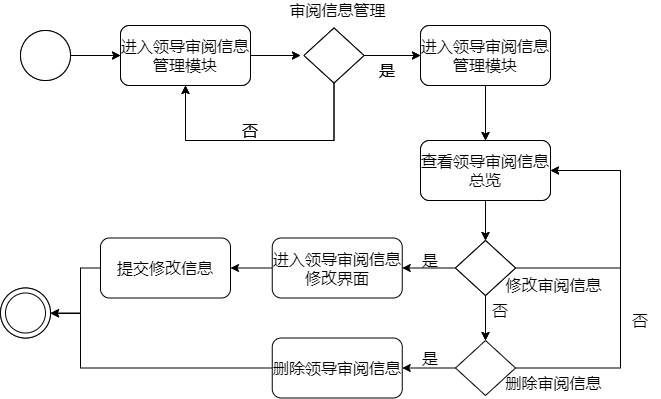


图6-31 审阅信息管理活动图

### 6.2.5企业信息填报系统

（1）功能结构设计

企业信息填报如图6-25所示：

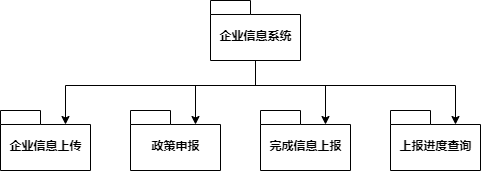


图6-25企业信息填报系统包图

（2）类图设计

企业信息填报类图如图6-26所示：

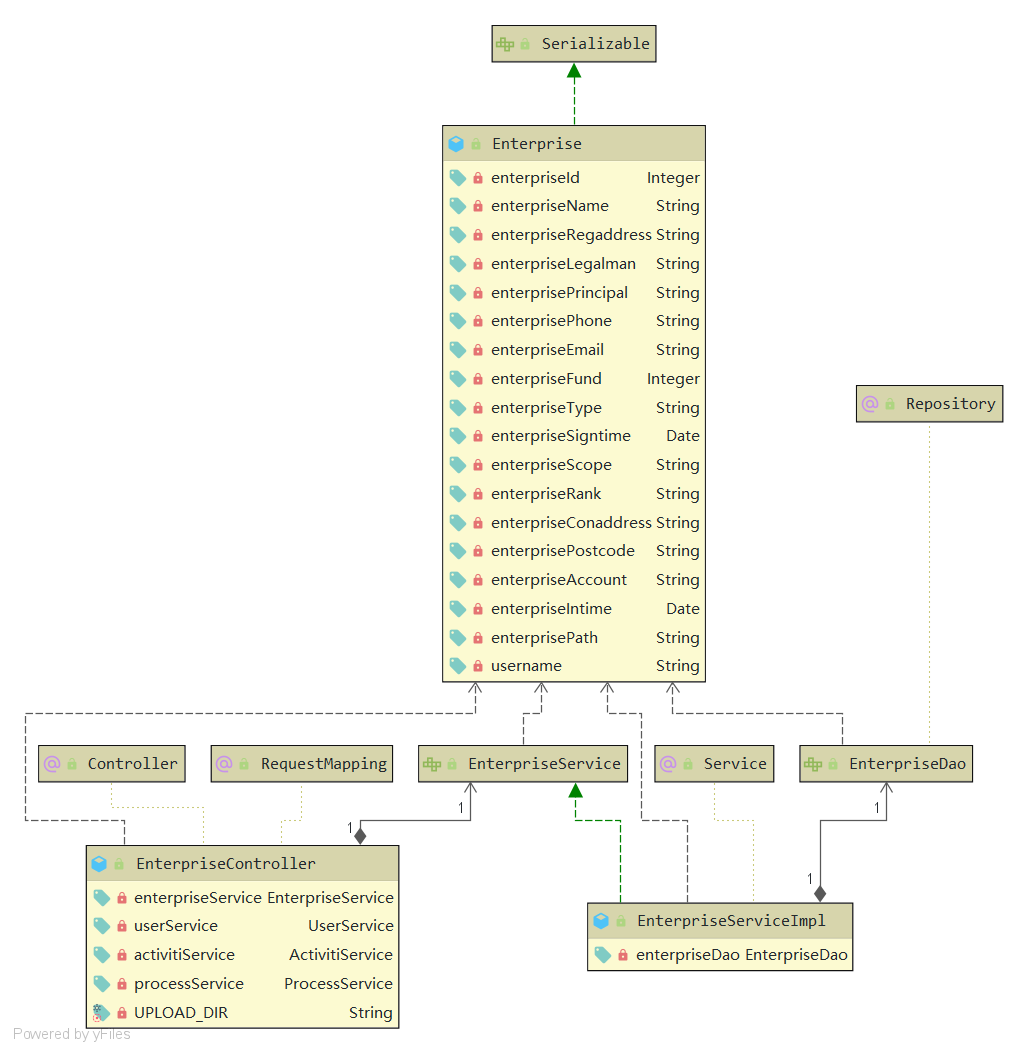


图6-27 企业基本信息类图

表6-17 企业基本信息实体类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类代号（code） | Enterprise | | 类名称（中文） | 企业基本信息实体类 | | |
| 类功能描述 | 详细定义企业的基本信息，并使用Getter和Setter方法设置获取和设置方法 | | | | | |
| 所在包名称 | com.policymanage.entity包 | | | | | |
| 类属性 | 类型 | 属性名称 | | | 数据库段名 | 其他 |
| enterpriseId | Integer | ID | | | e\_id | 无 |
| enterpriseName | String | 企业名称 | | | e\_name | 无 |
| enterpriseRegaddress | String | 企业注册地址 | | | e\_regaddress | 无 |
| enterpriseLegalman | String | 企业法人 | | | e\_legalman | 无 |
| enterprisePrincipal | String | 企业责任人 | | | e\_principal | 无 |
| enterprisePhone | String | 企业电话 | | | e\_phone | 无 |
| enterpriseEmail | String | 企业邮箱 | | | e\_email | 无 |
| enterpriseFund | Integer | 注册资金 | | | e\_fund | 无 |
| enterpriseType | String | 企业类型 | | | e\_type | 无 |
| enterpriseSigntime | Date | 企业注册时间 | | | e\_signtime | 无 |
| enterpriseScope | String | 经营范围 | | | e\_scope | 无 |
| enterpriseRank | String | 企业等级 | | | e\_rank | 无 |
| enterpriseConaddress | String | 企业联系地址 | | | e\_conaddress | 无 |
| enterprisePostcode | String | 企业邮编 | | | e\_postcode | 无 |
| enterpriseAccount | String | 开户银行 | | | e\_account | 无 |
| enterpriseIntime | Date | 企业入驻时间 | | | e\_intime | 无 |
| enterprisePath | String | 附件地址 | | | e\_path | 无 |
| username | String | 添加账号 | | | username | 无 |
| 主要实现方法 | Getter()方法，Setter()方法, toString()方法 | | | | | |

表6-18 当前任务信息实体类表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 类代号（code） | Process | 类名称（中文） | 当前任务信息实体类 | | |
| 类功能描述 | 详细定义企业的基本信息，并使用Getter和Setter方法设置获取和设置方法 | | | | |
| 所在包名称 | com.policymanage.activiti.entity包 | | | | |
| 类属性 | 类型 | 属性名称 | | 数据库段名 | 其他 |
| ID\_ | String | ID\_ | | ID\_ | 无 |
| REV\_ | Integer | 版本号 | | REV\_ | 无 |

续表6-18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EXECUTION\_ID\_ | String | 实例ID | EXECUTION\_ID\_ | 无 |
| PROC\_INST\_ID\_ | String | 路程ID | PROC\_INST\_ID\_ | 无 |
| PROC\_DEF\_ID\_ | String | 定义ID | PROC\_DEF\_ID\_ | 无 |
| NAME\_ | String | 任务名称 | NAME\_ | 无 |
| PARENT\_TASK\_ID\_ | String | 父节点ID | PARENT\_TASK\_ID\_ | 无 |
| DESCRIPTION\_ | String | 任务描述 | DESCRIPTION\_ | 无 |
| TASK\_DEF\_KEY\_ | String | 任务定义Key | TASK\_DEF\_KEY\_ | 无 |
| TASK\_DEF\_KEY\_ | String | 任务定义Key | TASK\_DEF\_KEY\_ | 无 |
| OWNER\_ | String | 所属人 | OWNER\_ | 无 |
| ASSIGNEE\_ | String | 代理人 | ASSIGNEE\_ | 无 |
| DELEGATION\_ | String | 代理团队 | DELEGATION\_ | 无 |
| PRIORITY\_ | Integer | 优先权 | PRIORITY\_ | 无 |
| CREATE\_TIME\_ | Timestamp | 创建时间 | CREATE\_TIME\_ | 无 |
| DUE\_DATE\_ | Datetime | 执行时间 | DUE\_DATE\_ | 无 |
| CATEGORY\_ | String | 种类 | CATEGORY\_ | 无 |
| SUSPENSION\_STATE\_ | Integer | 暂停状态 | SUSPENSION\_STATE\_ | 无 |
| TENANT\_ID\_ | String | 使用者ID | TENANT\_ID\_ | 无 |
| FORM\_KEY\_ | String | 表单Key | FORM\_KEY\_ | 无 |

表6-19 企业基本信息数据操作类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseDao | | 类名称（中文） | 企业基本信息数据操作类 |
| 类功能描述 | 对企业的基本信息进行相关的数据库操作 | | | |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有企业信息 |
| public | insertEnterprise | Enterprise | void | 添加企业信息 |
| public | findById | Integer | Enterprise | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | updateEnterprise | Enterprise | void | 更新企业信息 |
| public | deletEnterprise | Integer | void | 删除企业信息 |
| public | findByName | String | List<Enterprise> | 通过Name查询 |

续表6-19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | findByType | String | List<Enterprise> | 通过Type查询 |
| public | findByPrincipal | String | List<Enterprise> | 通过Principal查询 |
| public | findByAccount | String | List<Enterprise> | 通过Account查询 |
| public | findByLegalman | String | List<Enterprise> | 通过法人查询 |
| public | findByRank | String | List<Enterprise> | 通过Rank查询 |
| public | findByInTime | Date | List<Enterprise> | 通过InTime查询 |
| public | findByUser | String | Enterprise | 通过User查询 |

表6-20 企业基本信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseService | | 类名称（中文） | 企业基本信息业务逻辑类 |
| 类功能描述 | 对企业的基本信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Enterprise> | 查询所有信息 |
| public | insertEnterprise | Enterprise | void | 添加企业信息 |
| public | findById | Integer | Enterprise | 通过ID查询 |
| public | updatePathById | String, Integer | void | 上传附件链接 |
| public | updateEnterprise | Enterprise | void | 更新企业信息 |
| public | deletEnterprise | Integer | void | 删除企业信息 |
| public | queryEnterprise | Integer, String | List<Enterprise> | 根据表单查询 |
| public | AddQuery | Integer | boolean | 添加信息判空 |
| public | findByUsername | String | Enterprise | 查询对应用户名 |

表6-21 企业基本信息控制器类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | EnterpriseController | | 类名称（中文） | 企业基本信息控制器类 |
| 类功能描述 | 对企业的基本信息进行相关的操作负责处理分发的请求 | | | |
| 所在包名称 | com.policymanage.controller包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | getlist | ModelMap, Integer, Integer | String | 携带数据跳转 |
| public | skipEnterprise | 无 | String | 跳转至信息新增 |
| public | insertEnterprise | Enterprise | Message | 添加企业信息 |
| public | uploadFileById | MultipartFile, Integer | String | 上传文件 |

续表6-21

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | getEnterprise | Integer,HttpServletRequest,Model | String | 详情界面传值 |
| public | deleteEnterprise | Integer | String | 删除企业信息 |
| public | editEnterprise | Enterprise, Model | String | 携带数据跳转修改界面 |
| public | updateEnterprise | Enterprise | String | 编辑企业信息 |
| public | queryEnterprise | Integer, String, ModelMap | String | 查询展示数据列表 |
| public | skipDashboard | 无 | String | 跳转回总览界面 |
| public | excelDown | HttpServletResponse,HttpServletRequest | String | 下载企业信息转为Excel |

表6-22 当前任务信息数据操作类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | ProcessDao | | 类名称（中文） | 当前任务信息数据操作类 |
| 类功能描述 | 对当前任务的基本信息进行相关的数据库操作 | | | |
| 所在包名称 | com.policymanage.dao包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Process> | 查询所有进程 |
| public | findByASSIGNEE | String | List<Process> | 通过责任人查询任务 |
| public | getIdByAssignee | String | Process | 获取当前流程 |
| public | getKeyByProcess | String | String | 获取当前任务Key |
| public | updateBUSIKeyById | Integer, String | void | 注入act业务ID |
| public | findeIdByBUSIKey | Integer | String | 根据Key反向查询 |
| public | findProById | String | String | 根据ID查询所属流程 |
| public | findByEXEId | String | Process | 通过EXECUTION查询 |

表6-23 当前任务信息业务逻辑类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | ProcessService | | 类名称（中文） | 当前任务信息业务逻辑类 |
| 类功能描述 | 对当前任务信息进行相关的业务逻辑操作 | | | |
| 所在包名称 | com.policymanage.activti.service包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | findAll | 无 | List<Process> | 查询所有进程 |

续表6-23

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| public | findByASSIGNEE | String | List<Process> | 查询责任人所属任务 |
| public | getIdByAssignee | String | String | 获取当前任务ID |
| public | updateByBUSI | String, Integer | void | 将信息注入act表 |
| public | findProByBUSI | Integer | String | 根据业务查询流程 |

表6-24 当前任务信息控制器类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类代号 | ProcessController | | 类名称（中文） | 当前任务信息控制器类 |
| 类功能描述 | 对当前任务信息进行相关的操作，负责处理分发的请求 | | | |
| 所在包名称 | com.policymanage.controller包 | | | |
| 方法属性 | 方法名 | 输入 | 输出 | 方法功能描述 |
| public | SubmitFlow | String, MultipartFile | String | 上传Zip并部署工作流 |
| public | getlist | ModelMap, Integer, Integer | String | 跳转至TaskIndex界面 |
| public | getMyProcess | Model | Message | 获取我的任务 |

（3）顺序图设计

企业信息录入时序图如图6-28所示：

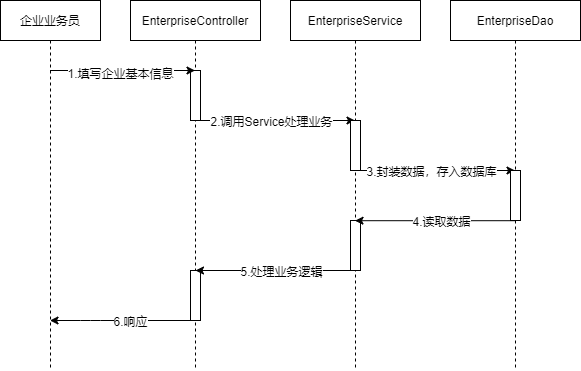


图6-28 企业信息录入顺序图

企业政策信息填报时序图如图6-29所示：

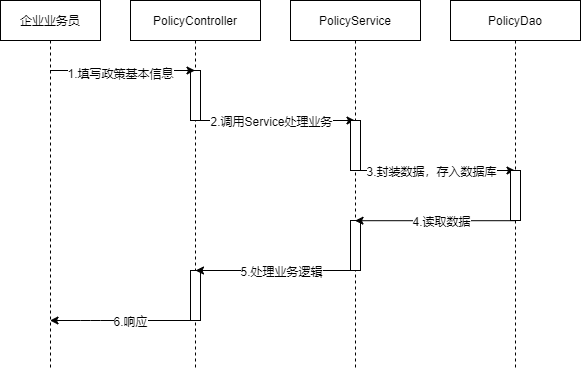


图6-29 企业政策信息填报顺序图

企业完成信息填报时序图如图6-30所示：

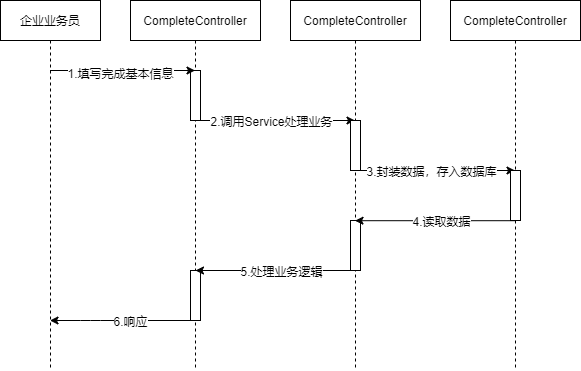


图6-30 企业完成信息填报顺序图

（4）核心处理流程设计

企业信息录入流程如图6-31所示：

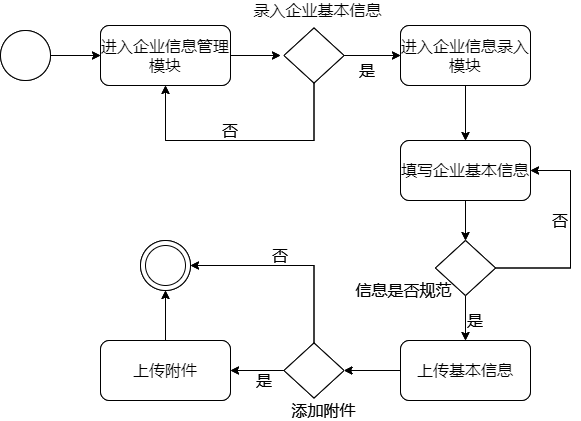


图6-31企业信息录入活动图

企业政策申报流程如图6-32所示：

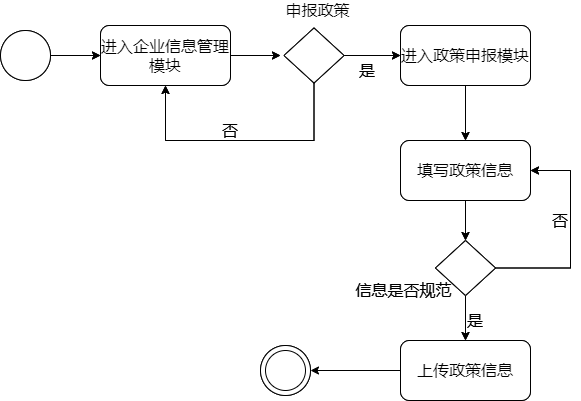


图6-32政策申报活动图

7 编码

## 7.1代码实现与核心算法

### 7.1.1企业信息填报系统

（1）启动流程添加企业基本信息核心代码

// 调用方法，启动流程

activitiService.startProcess("MainFlow", variables);

/\*处理任务一：提交企业信息\*/

// 将业务ID注入流程信息中

processService.updateBUSIKeyById(enterprise.getEnterpriseId(), userName);

// 获取当前流程ID

String proByBUSI = processService.findProByBUSI(enterprise.getEnterpriseId());

// 获取一个随机评估员信息

List<User> user = userService.findAllAssess();

int t = new Random().nextInt(user.size());

User sample = user.get(t);

String assessId = sample.getUsername();

variables.put("assessId", assessId);

// 处理任务

activitiService.completeTask(proByBUSI, variables);

Message msg = new Message("yes");

return msg;

（2）附件上传核心代码

@ResponseBody

@RequestMapping("/uploadFile")

public String uploadFileById(@RequestParam("file") MultipartFile file,

@RequestParam("enterpriseId") Integer enterpriseId) {

String path = UploadUtils.upload(UPLOAD\_DIR, file);

enterpriseService.updatePathById(path, enterpriseId);

return "redirect:getAllInfo";

}

（3）读取Word表格信息核心代码

public static void parseWord(String filePath) throws Exception {

Iterator<XWPFTable> it = getWordContent(filePath);

while(it.hasNext()){

XWPFTable table = it.next();

List<XWPFTableRow> rows = table.getRows();

//读取每一行数据

for (int i = 0; i < rows.size(); i++) {

XWPFTableRow row = rows.get(i);

//读取每一列数据

List<XWPFTableCell> cells = row.getTableCells();

for (int j = 0; j < cells.size(); j++) {

XWPFTableCell cell = cells.get(j);

//输出当前的单元格的数据

System.out.print(cell.getText() + "\t");

}

System.out.println();

}

}

}

（4）导出Excel核心代码

@RequestMapping("/excelDown")

public String excelDown(HttpServletResponse response,

HttpServletRequest request) throws Exception {

/\*获取Web项目根目录\*/

String path = request.getServletContext().getRealPath("/");

/\*定义模板的位置\*/

String url = path + "/model/Enterprise\_model.xls";

/\*获取全部数据\*/

List<Enterprise> list = enterpriseService.findAll();

/\*将数据导入Excel模板\*/

Workbook workbook = FillDataInModel.EnterpriseDataExcel(list, url);

ResponseUtils.export(response, workbook, "企业信息.xls");

return null;

}

### 7.1.2招商引资档案管理系统

（1）导出档案核心代码

@ResponseBody

@RequestMapping("/downloadFile")

public void downloadFile(Integer id, HttpServletResponse response, HttpServletRequest request) throws Exception {

String str = id.toString();

Enterprise enterprise = enterpriseService.findById(id);

Examine examine = examineService.findById(id);

Policy policy = policyService.findById(str);

Check check = checkService.findById(id);

Complete complete = completeService.findById(id);

/\*获取Web项目根目录\*/

String path = request.getServletContext().getRealPath("\\");

/\*定义模板的位置\*/

String url = path + "\\model\\File.xls";

Workbook workbook = FillDataInModel.FileExcel(enterprise, policy, examine, check, complete, url);

ResponseUtils.export(response, workbook, "档案信息.xls");

}

（2）下载所有档案信息为Excel核心代码

@RequestMapping("/download")

public void download(Integer fileId, HttpServletRequest request, HttpServletResponse response) throws Exception {

// 通过fileID查询当前附件路径

File FileById = fileService.findById(fileId);

String filePath = FileById.getFilePath();

// 获取文件的绝对路径名称

String path = UPLOAD\_DIR + "\\" + filePath;

// 获取后缀名

String suffix = filePath.substring(filePath.lastIndexOf(".") + 1);

String filename = "档案." + suffix;

ResponseUtils.download(path, filename, response);

}

### 7.1.3固定资产评估管理系统

（1）时间段查询核心代码

public List<Assess> queryByTimeBucket(String beginDate, String endDate) {

List<Assess> list = new ArrayList<>();

try {

/\*获取时间段内所有时间集合\*/

List<String> dateStrs = DateUtils.getDayListOfDateStr(beginDate, endDate);

/\*循环遍历集合内所有时间\*/

assert dateStrs != null;

for (String dateStr : dateStrs) {

/\*将dateStr转换为Date格式\*/

Date date = DateUtils.parseStrToDate(dateStr);

/\*通过转换后的Date数据对数据库进行查询并存储至assessList\*/

List<Assess> assessList = assessDao.findByTime(date);

/\*遍历assessList将其中元素加入list中\*/

list.addAll(assessList);

}

return list;

} catch (ParseException e) {

e.printStackTrace();

return null;

}

}

（2）新建评估传递流程核心代码

@ResponseBody

@RequestMapping("/insertAssess")

public Message insertAssess(Assess assess, @RequestParam("userName")String userName) {

// 添加评估信息

assessService.insertAssess(assess);

/\*处理任务二：固定资产评估\*/

// 通过ID查询对应企业信息的用户账号

Enterprise enterprise = enterpriseService.findById(assess.getAssessId());

String username = enterprise.getUsername();

// 将用户账号写入map集合注入流程

Map<String, Object> variables = new HashMap<>();

variables.put("clerkId", username);

// 获得流程ID

String proByBUSI = processService.findProByBUSI(assess.getAssessId());

// 处理任务

activitiService.completeTask(proByBUSI, variables);

/\*向前端传递信息\*/

Message msg = new Message("yes");

return msg;

}

### 7.1.4领导审阅统计管理系统

（1）添加评阅信息处理分支核心代码

@RequestMapping("/check")

public void check(Check check) {

// 获取check的id，判断其是否存在

Check i = checkService.findById(check.getCheckId());

if (i == null) {

// 添加审阅信息

checkService.insertCheck(check);

}

else {

// 修改审批信息

checkService.updateCheck(check);

}

/\*处理任务五：政策审阅\*/

Map<String, Object> variables = new HashMap<>();

// 获取check中的审批结果，并判断分支

String checkResult = check.getCheckResult();

if (checkResult.equals("通过")) {

variables.put("approve", true);

} else {

variables.put("approve", false);

}

// 获得流程ID

String proByBUSI = processService.findProByBUSI(check.getCheckId());

// 处理任务

activitiService.completeTask(proByBUSI, variables);

}

### 7.1.5企业基本信息管理系统

（1）编辑企业信息核心代码

@RequestMapping("/deleteEnterprise")

public String deleteEnterprise(Integer enterpriseId) {

enterpriseService.deleteEnterprise(enterpriseId);

return "redirect:getAllInfo";

}

（2）删除企业信息核心代码

@RequestMapping("/editEnterprise")

public String editEnterprise(Enterprise enterprise, Model model){

model.addAttribute("enterprise", enterpriseService.findById(enterprise.getEnterpriseId()));

return "EnterpriseEdit";

}

## 7.2 代码优化分析

为避免很多重复代码造成的代码冗余，可以多写几个工具类定义静态方法，提供其他类使用如：

DateUtils为时间工具类，负责的是时间格式转换，计算时间差等多用途方法，在整个项目开发中多次使用，避免了代码的冗余；FillDateModel是数据注入工具，用于配合Excel的导出，它可以将数据库中的数据注入以Excel为模板的文件中，而项目中多次需要数据的导出，成功规避冗余问题；ResponseUtils解决的是导出文件下载的问题，其对项目开发的作用与上述类似；UploadUtils是数据上传工具，本系统中三个子系统均要求有附件上传功能，解决了代码冗余。

8 测试

## 8.1 测试方案设计

### 8.1.1 测试策略

首先将该系统划分为多个小模块分别进行单元测试。由于该项目总体结构为SSM框架，有Spring，SpringMVC，Mybatis整合而成，并非一蹴而就，针对于Spring，SpringMVC以及Mybatis的整合过程就可以进行单元测试，以免在后续的开发中有读取配置文件失败等问题，一步一步一边进行单元测试一边进行开发能够大大提高开发效率以及代码正确度。在进行Dao层开发时，由于需要对数据库直接进行操作，也需要一定的单元测试以确保数据库操作的正确。

此外，在进行界面开发时，应当同时进行界面测试，以确保布局的合理性以及整体风格的一致性，站在用户的角度，全方位考虑使用者对于交互的需求，考虑各个控件的存在是否合理，位置是否符合交互便捷性，整体风格是否通俗易懂，色彩搭配是否完美等等。

在开发控制层时，对层内的各个控制器进行测试，主要是测试其业务调度是否合理，是否符合项目要求，能否成功接受客户端传来的请求，能否成功将数据传给视图层用于视觉的展示。

在整合所有前后端后，通过使用Chrome浏览器自带的开发者工具进行测试，使用Network空间测试观察Ajax的数据传输是否能够加载，解析成功，是否报错。所有模块组装后，通过系统测试方法，并结合项目需求分析，详细的测试每个需求能否在开发完成的系统中运行成功，如果均能够测试通过，实现相应要求，即完成测试。

### 8.1.2 测试资源

软件资源：junit-4.12依赖，Chrome浏览器

人力资源：软件开发者本人

### 8.1.3 关键测试点

测试点一：Mybatis，Spring，SpringMVC整合，配置文件读取；

测试点二：Dao等内各类对数据库的操作，使用单元测试，控制台输入输出来测试能都正常运作；

测试点三：Ajax数据封装为JSON，以及其解析；

测试点四：时间格式转换工具类的测试；

测试点五：前端界面测试，测试其合理性；

## 8.2 测试用例构建

### 8.2.1 测试用例设计

分模块进行单元测试，本系统分为五大子系统，每个模块针对其可能出现的各种输入情况进行测设，包括正常输入，错误输入，重复输入以及格式错误输入等问题。通过这种方式对软件系统进行相应测试并完成测试用例。

在开发过程中要一边测试以便完成项目，以避免在模块开发完成后出现问题，这样排除bug的难度将大大提高。例如，在开发Dao层是，由于要处理数据库相关操作，容易由于MySQL语言的错误或者没有对应清楚而产生一些列问题，这就是一个非常重要的测试点，如图8-1所示。

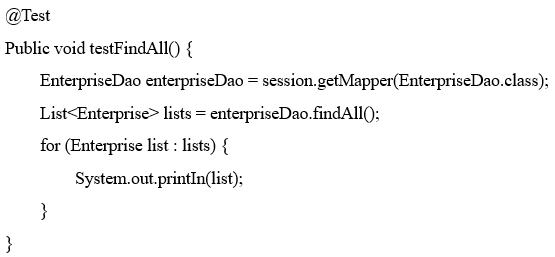


图8-1 查询所有企业信息测试方法

用这种方式进行测试，可以大大减轻开发完成后排除bug的难度，有效提高开发效率。此外，出了一些输入性的错误，应针对用户容易出现的逻辑性问题进行排除，比如：企业业务员还没有上报企业信息就进行相关的政策信息申报，这在现实生活中是不符合逻辑的操作，应对其进行相关测试并完善。

在该系统中，容易出现的就是逻辑错误而产生的问题，应对此进行相关的测试工作。还有就是信息输入等过程中，应对每一个表单可能出现的所有错误输入、重复输入、不输入进行测试。

### 8.2.2 关键测试用例

表8-1 登录模块测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **登录模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | 邮箱：空  密码：19980705 | 邮箱不能为空 | 成功 |

续表8-1

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | 邮箱：[61958605@qq.com](mailto:61958605@qq.com)  密码：空 | 密码不能为空 | 成功 |
| 3 | 邮箱：[61958605@qq.com](mailto:61958605@qq.com)  密码：19980705 | 密码正确进入系统 | 成功 |
| 4 | 邮箱：[61958605@qq.com](mailto:61958605@qq.com)  密码：123456 | 密码错误，提示 | 成功 |

表8-2 注册模块测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **注册模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | ID：空；邮箱：[619580605@qq.com](mailto:619580605@qq.com)；密码：199875 | ID不能为空 | 成功 |
| 2 | ID：12312312；邮箱：空；密码：199875 | 邮箱不能为空 | 成功 |
| 3 | ID：12312312；邮箱：[61958605@qq.com](mailto:61958605@qq.com)；密码：空 | 密码不能为空 | 成功 |
| 4 | ID：1；邮箱：[61958605@qq.com](mailto:61958605@qq.com)；密码：19980705 | ID必须为8-12位 | 成功 |
| 5 | ID：12312312；邮箱：[619580605](mailto:61958605@qq.com)；密码：123456 | 邮箱格式有误 | 成功 |
| 6 | ID：12312312；邮箱：[61958605@qq.com](mailto:61958605@qq.com)；密码：1 | 密码必须为6-18位 | 成功 |
| 7 | ID：12312312；邮箱：[61958605@qq.com](mailto:61958605@qq.com)；密码：12323422 | 密码必须包含字母数字 | 成功 |

表8-3 企业信息录入测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **企业信息录入模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | ID：空 | 企业ID不能为空 | 成功 |
| 2 | ID：1 | 企业ID必须为6-10位的纯数字! | 成功 |
| 3 | ID：E | 企业ID必须为6-10位的纯数字! | 成功 |
| 4 | ID：2343243 | 正确输入 | 成功 |
| 5 | 企业名称：空 | 企业名称不能为空! | 成功 |
| 6 | 企业负责人：空 | 企业负责人不能为空! | 成功 |
| 7 | 联系电话：1231 | 联系电话格式有误! | 成功 |
| 8 | 联系电话：15170090705 | 正确输入 | 成功 |
| 9 | 企业入驻时间：空 | 企业入驻时间不能为空! | 成功 |
| 10 | 签订政策：空 | 签订政策不能为空! | 成功 |
| 11 | 签订政策时间：空 | 签订政策时间不能为空! | 成功 |

表8-4 招商引资档案录入测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **招商引资档案录入模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | ID：空 | 企业ID不能为空 | 成功 |
| 2 | ID：1 | 企业ID必须为6-10位的纯数字! | 成功 |
| 3 | ID：E | 企业ID必须为6-10位的纯数字! | 成功 |
| 4 | ID：2343243 | 正确输入 | 成功 |
| 5 | 责任人：空 | 责任人不能为空! | 成功 |
| 6 | 责任处室：空 | 责任处室不能为空! | 成功 |
| 7 | 档案题名：空 | 档案题名不能为空! | 成功 |
| 8 | 档案写入时间：空 | 档案写入时间不能为空! | 成功 |

表8-5 固定资产投资评估录入测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **固定资产投资评估录入模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | ID：空 | 企业ID不能为空 | 成功 |
| 2 | ID：1 | 企业ID必须为6-10位的纯数字! | 成功 |
| 3 | ID：E | 企业ID必须为6-10位的纯数字! | 成功 |
| 4 | ID：2343243 | 正确输入 | 成功 |
| 5 | 第三方评估公司名称：空 | 第三方评估公司名称不能为空! | 成功 |
| 6 | 第三方评估公司联系电话：空 | 第三方评估公司联系电话不能为空! | 成功 |
| 7 | 第三方评估公司登记机关：空 | 第三方评估公司登记机关不能为空! | 成功 |
| 8 | 工商注册号：空 | 工商注册号不能为空! | 成功 |
| 9 | 评估时间：空 | 评估时间不能为空! | 成功 |
| 10 | 评估结果：空 | 评估结果不能为空! | 成功 |

表8-7 企业政策申报测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **固定资产投资评估录入模块** | | | |
| 编号 | 输入条件 | 预期输出 | 确认输出 |
| 1 | 申请单位：空 | 申请单位不能为空！ | 成功 |
| 2 | 申请单位：腾讯 | 正确输入 | 成功 |
| 3 | 申请单位：123456456456456 | 申请单位名字过长！ | 成功 |
| 4 | 企业地址：空 | 企业地址不能为空！ | 成功 |
| 5 | 企业地址：123123 | 企业地址不能为纯数字！ | 成功 |
| 6 | 企业地址：省市西湖区兰宫路228号 | 企业地址 | 成功 |
| 7 | 邮政编码：空 | 邮政编码不能为空！ | 成功 |
| 8 | 邮政编码：Hello | 邮政编码必须为纯数字！ | 成功 |
| 9 | 邮政编码：330000 | 正确输入 | 成功 |
| 10 | 联系人：空 | 联系人不能为空！ | 成功 |
| 11 | 联系人：12 | 联系人不能为数字 | 成功 |
| 12 | 联系人：李小二 | 正确输入 | 成功 |

续表8-7

|  |  |  |  |
| --- | --- | --- | --- |
| 13 | 开户行：空 | 开户行不能为空！ | 成功 |
| 14 | 开户行：中国银行 | 正确输入 | 成功 |
| 15 | 账号：空 | 账号不能为空！ | 成功 |
| 16 | 账号：1220 1029 3293 123 | 正确输入 | 成功 |
| 17 | 资金类别：空 | 资金类别不能为空！ | 成功 |
| 18 | 资金类别：扶持资金 | 正确输入 | 成功 |
| 19 | 已获得扶持资金：空 | 已获得扶持资金不能为空！ | 成功 |
| 20 | 已获得扶持资金：100000 | 正确输入 | 成功 |
| 21 | 申请时间：空 | 申请时间不能为空！ | 成功 |
| 22 | 申请时间：2020-4-1 | 正确输入 | 成功 |
| 23 | 项目所在国（地区）：空 | 项目所在国（地区）不能为空！ | 成功 |
| 24 | 项目所在国（地区）：123123 | 项目所在国（地区）不能为纯数字！ | 成功 |
| 25 | 项目所在国（地区）：中国 | 正确输入 | 成功 |
| 26 | 项目实施时间：空 | 项目实施时间不能为空！ | 成功 |
| 27 | 项目实施时间：2020-4-1 | 正确输入 | 成功 |
| 28 | 项目负责人：空 | 项目负责人不能为空！ | 成功 |
| 29 | 项目负责人 | 项目负责人 | 成功 |
| 30 | 电话：空 | 电话不能为空！ | 成功 |
| 31 | 电话：ajsneb | 电话格式不正确 | 成功 |
| 32 | 电话：13187763847 | 正确输入 | 成功 |
| 33 | 申请条例：空 | 申请条例不能为空！ | 成功 |
| 34 | 申请条例：某市科技扶持政策条例 | 申请条例 | 成功 |

### 8.2.3 测试用例维护

在多种情况下可能会导致测试用例的变更，如：软件需求变更，测试需求的遗漏，测试用例的遗漏，用户反馈缺陷等。

（1）软件需求变更

软件需求的变更可能会导致某些功能的增加、删除、修改等等变化，那么随着变化的发生，应当对种种变化进行管理和维护测试用例，在变化后的情况下，测试用例也需要进行响应的变更；

（2）测试需求和测试用例的遗漏

在测试过程中，可能会处于不全面或不仔细等问题，测试用例没有覆盖全面，从而导致一些列问题，那么根据问题发生的改变，要对测设用例进行相应的维护、修改等操作；

（3）用户反馈缺陷

该系统在交付后可能会出现一些瑕疵，用户在时候用会对其进行相应的反馈，那么要对系统进行一些补充和修改，同时也需要补充和修改测试用例。

9 总结与展望

## 9.1 设计工作总结

通过本项目的设计与实现，我从中学到并掌握了许多。首先是对软件开发团队合作的重要性的认知，本次毕业设计实在团队合作的基础上进行完成的，光是一个人的力量是远远不能完成这个项目的。只有通过团队之间不断地讨论和探索，才能对项目的开发起到异常积极的作用。

此外，通过官方文档自学相关插件的开发相当重要。Activiti插件作为本系统相当重要的基石，对于整个项目流程的部署和完善起到了非常关键的作用，但是对于项目开发来说，这是我首次使用Activiti插件进行开发，并没有使用经验。通过文档的学习，我学会了如何使用Activiti插件对流程进行开发，所以说，通过官方文档的自学对整个项目的开发来说是非常重要的。

除此之外，项目开发前的实地调研对于整个项目开发来说也异常关键。只有通过项目前详细的实地调研，需求分析才能更加合理，需求分析的合理性决定了后面整个项目的设计与实现的难度，避免了重复开发和维护的难度。

## 9.2 未来工作展望

虽然毕业设计已经完成，但这仅仅是告一段落。本系统还有诸多问题亟需解决，需要在后面的实际应用中不断完善，在以下方面还需要进一步的研发：

（1）本项目在开发过程中更注重对于整体流程的设计，可能在数据的统计等加强用户体验、丰富系统功能等方面进行深入思考，使得整个系统的功能稍微有些简略，在数据的整理和运营方面还需要加强；

（2）在另一方面，对于企业信息，政策信息方面的数据库表字段的完整性可能有所欠缺，需要对其进行完善，提高信息的完整性；

（3）系统在一些人性化方面还存在一些问题，比如领导在查看某一个企业的信息时没有一次展示所有的相关信息，需要点击功能后才能展示其他信息，增加了操作的繁琐程度，故需要增加系统的人性化设计开发；

（4）在用户可能的所有操作方面可能没有考虑全面，用户可能会有一些操作导致系统的异常，这些问题在不断的测试和实际操作中预期可以解决；

（5）可以在本系统的开发基础上增加更多功能，加强系统丰富度，提升用户体验，比如：企业业务员可以申请修改企业信息，不用再联系高新区工作人员进行修改；企业业务员可以在一定时间范围内对工作进程较慢的政府部门进行催办操作等等。

这些仍然存在的不足将在未来不断的测试运营和实际操作过程中将会不断补充、修改并完善。

谢 辞

参考文献

[1]侯俊.企业资金结算管理系统的开发与实现[J].现代营销（经营版），2019.

[2]孙术超.给予JAVA的开发和应用[J].计算机产品与流通，2018.

[3]杨瑞东，王云峰，张海英.Spring新特性之Java Config在Web开发中的应用[J].微型机与应用，2017.

[4]姜维，王岩.基于JavaWeb+JDBC的校园服务平台的功能与实现[J].现代农机，2018.

[5]汪永松.JavaWeb开发技巧之项目模板[J].电脑编程技巧与维护，2020.

[6]刘志洋.ajax技术在Web程序开发中的运用探讨[J].轻纺工业与技术，2020.

[7]王菲露，李军，宋杨，胡勇，陈玉峰.基于Ajax技术的ECharts实时图形报表实现[J].黑龙江工业学院学报（综合版），2019.

[8]袁灵华.政府机构办公自动化（OA）系统安全策略探究[J].信息与电脑（理论版），2019.

[9]李珍，童莲.基于JavaEE的校园OA系统的设计与实现[J].花炮科技与市场，2019.

[10]Bhaskar Kumar, Kumar Abhishek, Akshay Deepak, M.P.Singh. Implementation of Interactive Real Time Online Co-shopping Using Push AJAX[J].Procedia Computer Science，2016, 89: 473-482.

[11]Ying Xin Liu, Xi Yuan Li. Design and Implementation of a Business Platform System Based on Java[J].Procedia Computer Science，2020, 166: 150-153.

[12]张忠，宋嘉诚，黄隽瑶.基于JavaEE物品仓储管理系统设计[J].电脑编程技巧与维护，2020.

[13]赵国平，夏平平.基于JavaEE的人事管理系统的设计与实现[J].现代工业经济和信息化，2019.

[14]季昆， 孟丽丽，薛迁，孙晓伟，王才华.基于JavaWeb的产品质量检测预警系统的设计与实现[J].数字技术与应用，2019.

[15]肖桃，王致力，张沂蒙，张紫琳.基于JavaWeb的研究生报考信息系统的设计与实现[J].科技视界，2019.

[16]周启云，许新华，付昱榕，刘文文，刘梦宇.基于Echarts的可视化学情分析系统设计与实现[J].信息技术与信息化，2020.

[17]周玮祎.基于ECharts的市场分析设计与实现[J].电子技术应用，2019.

[18]李光明，房靖力.基于JavaWeb的推荐数据后台管理系统的设计与实现[J].电脑知识与技术，2020.

[19]莫运襟.小议基于SSM的水电站设备管理系统的设计及实现[J].轻工科技，2019.

[20]马梓昂，贾克斌.基于Web的高性能智能快递柜管理系统[J].计算机应用与软件，2020.

[21]彭勇，陈俞强.基于SSM的实验室管理系统的研究与设计[J].微型电脑应用，2020.

[22]倪晓锋.基于SSM和Shiro的火电SIS系统的设计与实现[J].工业控制计算机，2019.

[23]傅安中，王祥荣.基于Apache Shiro的储物管理系统权限管理研究与实现[J].电脑编程技巧与维护，2020.

[24]王杉文.基于SpringBoot+Shiro的权限管理实现[J].电脑编程技巧与维护，2019.

[25]陈宇收，饶宏博，徐亮.[基于Shiro的权限管理机制研究](https://kns.cnki.net/kns/detail/detail.aspx?QueryID=75&CurRec=15&recid=&FileName=DNBC201906014&DbName=CJFDLAST2019&DbCode=CJFQ&yx=&pr=CJFT2019;&URLID=&bsm=QK0203;T0304;)[J].电脑编程技巧与维护，2019.

[25]刘聪，李世川，吕雪峰.基于Activiti+MongoDB的后勤自由流程信息系统[J].计算机系统应用，20.

[26]付晓明.企业管理系统集成Activiti工作流引擎的研究[J].电脑知识与技术，2019.

[27]贺雪梅.基于Activiti5工作流技术的定时任务在审批流程中的设计与实现[J].电子技术与软件工程，2019.

[28]Shu Lin Yang.Design of Task Workflow Based on Activiti Technology[J]. Applied Mechanics and Materials，2015.

[29]Da Wei Zhang.Production Process Modeling and Realizing Based on Activiti[J]. Applied Mechanics and Materials，2016.

[30]张学诚. Java WEB开发中的中文乱码问题解决方法 [J].电脑知识与技术，2018.

附录A 外文翻译—原文部分

*Procedia Computer Science* Volume 89, 2016, Pages 473-482

Implementation of Interactive Real Time Online Co-shopping Using Push AJAX

**Abstract**

Online shopping has witnessed explosive growth: there are hundreds of millions of online shoppers that are part of an online retail industry worth of hundreds of billions of dollars. While online shopping has its own advantages(such as convenience, product reviews, price and selection range), recent focus has been in using technology to mimic the social interactions found in physical shopping malls and stores in what is called as "social shopping". Put simply, in social shopping shoppers' friends become involved in the shopping experience. An important aspect of social shopping is "co-shopping" where ad-hoc collaborative shopping groups are formed in which one person can drive an online shopping experience for one or more other people. This requires a mechanism that provisions multiple users to conduct real time online shopping collaboratively from multiple locations. For example, family members at remote locations can view the product online simultaneously as if they are shopping together in real time. To this end we have designed a special collaborative browsing window in which all the co-shoppers need to login. One user becomes the leader and invites others from his friends' list for shopping. Other users need to accept this invitation and join the shopping. The leader upon liking a product initiates the co-shopping by clicking on "Share this product with selected friends" button. On clicking this button an AJAX request, which contains URL of the product and list of IDs of selected friends, goes to the server. The server receives the URL of the product and list of IDs of friends and pushes this URL to all the selected friends in an asynchronous manner using PUSH AJAX technology. This URL gets loaded into an HTML iFrame on every client's browsing window. Shoppers can communicate with each other using a simple chat window.

**1. Introduction**

E-commerce and wide penetration of internet has changed the way we shop today. The number of shoppers coming online for retailing has reached hundreds of millions and continues to grow. As per a report by China Internet Network Information Center", the number of users shopping online in China had reached 242 million in 2012 itself. The same figure for USA stands at 202 million in 2015t While India has the second largest internet base of about 354 million users as of June 2015\*, the penetration of e-commerce is low as compared to developed countries. However, the number of online shoppers is growing at an explosive rate of 6 million new entrants every months and is expected to cross 100 million by 2016. It is not just the number of online shoppers running into hundreds of millions, but also the online retailing is running into hundreds of billions of dollars. For instance, the online spending in USA was around 307 billion dollars in 2014\*\* and the same figure for China is around 300 billion dollars for 2013. India is catching up fast and its e-commerce market is estimated to grow at a compounded annual growth rate (CAGR) of 63% to reach 8.5 billion dollars in 2016.

1.1 *Need for this design*

Shopping is a social activity in which a group of people view the product and discuss about the product before buying it. Discussions and reviews make the buyer more comfortable in buying the product. The same ideology applies to Online Shopping as well. However, till now no good work exists that enables a customer to perform interactive real time co-shopping. All of the implemented works either use outdated technologies, raise performance issues, use open APIs such as Google Maps, or fail to serve in real time situations. More on the existing works on co-shopping are discussed in the "Related Works" section on the next page. Therefore, a method that can provision online co-shopping is required to make online shopping more real. Real time online co-shopping can be easily implemented using PUSH AJAX technology.

1.2 *Push AJAX*

HTTP supports unidirectional communication from client to server. It does not have provision for the "server-push" feature that notifies clients immediately if any change happens on server-side. Most designers today implement server-push by client-side polling, in which the client keeps on polling the server every fixed interval of time to get the latest server data. However, this design generates a lot of unnecessary traffic because clients keep on repeatedly polling the server even though there are no server side updates.

PUSH AJAX provides light weight, reliable and secure way to push notifications asynchronously from client to server. The reliable mechanism facilitates notifications delivery, fault-tolerant and high-availability supports. This part of implementation carries no application data, thus making it secure. Thus, real time online co-shopping can be easily implemented using PUSH AJAX technology.

PUSH AJAX uses a technique called Long Poll to update server data to clients. Once a client establishes the connection, the server gives an illusion to the client that more of data is still to come and holds back the connection. When the data is available the server sends the data through the held connection. At last, when the co-browsing ends, the server closes the connection. This way, real-time communication is achieved from server to client.

Long Poll technique has two significant disadvantages. Firstly, by default, browser allows only 2 connections per server. Since one connection is busy with PUSH AJAX, only one connection is remaining to service client's request. This may cause service delays. Also, uncontrollable factors like network delays cannot be handled and result in service delays.

In order to achieve AJAX PUSH in this article, ICEpush libraries and APIs are used. ICEpush provides lightweight asynchronous PUSH AJAX using long polling technology as described above. It provides two main jars to achieve PUSH AJAX-icefaces.jar and icepush.jar. In order to use ICEpush libraries, a proper well-defined framework needs to be followed. ICEpush internally uses JSF framework that connects UI widgets with data sources and server-side event handlers. This is discussed in implementation part in detail.

1.3 *Design summary*

In summary, the online co-shopping works as follows:

a. leader user invites his online friends for co-shopping, (Offline friends cannot be invited.)

b. leader clicks on "Share this product with friends" button upon liking a product,

c. an AJAX request containing list of friends and URL of the product goes to the server,

d. the server manipulates the request by pushing the URI to all the friend's browsers.

e. Server pushes the URI into each client's HTML iFrame using PUSH AJAX.

f. Shoppers can communicate with each other using a chat window.

**2. Related Work**

The growing E-commerce industry in online shopping attracted good amount of works on bridging the gap between buyers and sellers. However, very few works have been done on implementing an interactive real time co-shopping. In 2007, Khoury, Michel, Xiaojun Shen, Shirmohammadi.S proposed a "A Peer-to-Peer Collaborative Virtual Environment for E-Commerce". However, it fails to implement interactive realtime co-shopping with a group of friends. Moreover, it also relies on third party VRML plugins for peer-to-peer communication to happen. In2012, Jen-Hao Hsiao, Li-Jia Li proposed "On visual similarity based interactive product recommendation for online shopping". However, this design only provides a mechanism to make product recommendation method interactive based on similarity. It lacks implementation of real time shopping.

Richard Han, Veronique Perret, Mahmoud Naghshine in 2000 proposed a XML splitter based Multi-Device Collaborative Web Browsing. However, this design introduces a proxy to manipulate the requests, create and maintain sessions and split the XML pages. This proxy is an extra over head. Moreover, this design implements client side using an Applet based User Interface, which is hardly used these days. Applets are criticized for their Plug-In and JRE requirements, their startup performances and client side security issues. Modern browsers these days provide client server interface much better than what applets used to provide. Also, this approach is based on hidden client pull technology and assumes that universal push solution from server to client does not exist, which is a wrong assumption. AJAX PUSH is a technique that facilitates data push from server to clients when fresh data is available at the server.

Maria Aneiros, Vladimir Estivill-Castro (2005) proposed a Group Unified History (GUH) based collaborative web browsing. In this design, every time the user sends an HTTP request, a proxy server manipulates the request and passes it to GUH server. The GUH server then pushes all this information to the client updated history, and allows users to visit sites that others have judged or commented positively. However, this design is adds two extra servers-proxy and GUH server, thus making the design complex. Moreover, this design is specific to educational needs and would have to undergo lots of design changes to accomplish collaborative online shopping.

K. Maly, M. Zubair, and L. Li (2001) presented a design that removes the proxy server and describes an approach that supports a graphical Web browser with Java and Java Script. However, this design requires client side users to have JAVA enabled browsers and that form mutual exclusive groups of surfers that share the same experience. Moreover, this design again uses Applet based client side User linterface, which is outdated. Also, changes at the server are monitored through a JavaScript Co-Pointer. The Co-pointer's relative position is monitored to measure the changes in the browsing window. This is an outdated technology because there are better mechanisms to measure the changes in the browsing window.

Alan W. Esenther 2002 presents a simple light weight web based real time collaborative web browser. Its instant messaging system provides a quick, simple and safe way to co-browse. Its' non-intrusive nature facilitates collaboration over existing web content without modification. However, this design cites some critical disadvantages and limitations. Pixel positions and resolutions of each user's browsing window needs to be same. So it is preferred to use the same browser brand on each client. Also, the server side co-browsing window logic needs to be placed in same domain as all web pages that are to be shared during the collaborative session.

Dietwig Lowet, Daniel Goergen 2009 Co-Browsing Dynamic Web Pages, illustrates a magnificent way to implement co-browsing using JavaScript DOM synchronization. However, this design requires Firefox configuration to allow trusted JavaScript to receive cross domain scripting access and use the Firefox socket mechanism. This cross-domain scripting permission can lead to execution of unwanted JavaScript and poses serious security issues. This is the reason, cross domain scripting permissions are disabled by default. Jason J. Jung 2011, proposed a ContextGrid based contextual mashup-based collaborative browsing system in which a ContextGrid Model is used to compare user contexts. Also, for better understanding of user context, open APIs such as Google Map APIs, are used to fetch additional information through various sources. This becomes a tedious approach to compare user contexts through various APIs to implement co-browsing because generating queries and filtering the query result and finding out relevant results will add to cost and complexity.

Raphael O. Santos, Roberta L. Gomes, Felip F. Oliveira, Magnos Martinello,2011 proposed a magnificent design to achieve Lightweight Collaborative Web Browsing by developing OCEAN prototype. This design achievesco-browsing considering performance, responsiveness, scalability and user interface in an efficient way. Also, this architecture avoids bottlenecks on the server side, taking advantage of user's bandwidth in order to promote synchronous collaborative experiences. However, this design cites a concern related to feasibility of this design in real scenario due to unclear infrastructure. Moreover, this design is not suitable for dynamic web applications like online shopping.

Many research has happened in the field of co-browsing, however, most of them are based on master slave model, proxy based model or JavaScript DOM synchronization based model, which are either outdated ways or complex ways to achieve server-client communication. There has been no model proposed with support for interactive realtime co-browsing, which is browser independent, simple to implement and light weight, a design that can facilitate requirements of realtime online shopping.

2.1 *Comparison with Flipkart Ping App*

India's E-Commerce giant Flipkart recently launched Ping, a new feature in its mobile app that allows users to chat with their friends while shopping, thus giving a start to a collaborative shopping experience. Ping allows users to have real-time conversations with friends in order to get feedback about a product they plan on buying, or help friends get in on the decision of what to buy. Using this feature, users can drag and drop products directly into conversations, share their screen, share products with friends from their cart, wish list, gallery and even allows them to take photos.

However, this feature is limited to mobile phones without desktop support. Today, majority of shoppers prefer to shop on desktop for better visibility of product and their details. Online shopping on desktop gives users a wide interface for viewing the product details, its ratings and reviews, discount offers on various retailers and alternative options to the product that a shopping on mobile phone fails to give. Moreover, this app requires window change to go from chat window to product details window. Therefore, Flipkart's Ping is just a beginning to real time online collaborative shopping and the need for a desktop version of co-shopping still remains. The implementation discussed in the paper, real time online collaborative shopping is achieved on desktop. All the product details and chat window are implemented into the same collaborative window and no window change is required. Using this desktop version of online co-shopping, users can have a better view of product details, its ratings and reviews, discount offers on various retailers and alternative options to the product.

附录B 外文翻译—译文部分

《计算机科学进展》第89卷，2016年，第473-482页

利用Push Ajax技术实现交互式网上购物

**摘要**

伴随着数据的爆发式增长，网上购物应运而生：数亿网上购物者参与了价值数千亿美元的网上零售行业，为其贡献了自己的一份力量。虽然网上购物有其自身的优势（如采购方便、产品评论、价格低廉和选择范围广），但是，帮随着网络技术的发展，新的技术不断更迭，而如今大众更希望是使用技术来模拟实体购物中心和商店中的社交互动，即所谓的“社交购物”。简单来说，在社交购物中，购物者与他们的朋友都参与到购物体验中。所谓社交购物的一个重要方面便是“联合购物”，在这里，每一个购物者可以为一个或多个其他参与购物的顾客提高网上购物的体验。而这需要一种机制，让多个用户在多个地点协同进行实时的网上购物。就比如说，在偏远地区的家庭成员可以同时在线查看产品，就像他们一起实时购物一样。为此，我们设计了一个特殊的协同浏览窗口，所有的共同购物者都需要登录。一个顾客成为了类似一个领导者的作用，他要负责从他的朋友列表中邀请其他人来购物。其他用户需要接受此邀请并加入购物。用户点赞后，点击“与选定的朋友分享该产品”按钮，开始共同购物。单击此按钮，一个Ajax请求(包含产品的URL和所选好友的id列表)将转到服务器。服务器接收产品的URL和好友的id列表，并使用Push Ajax技术以异步方式将此URL推送给所有选中的好友。这个URL被加载到每个客户端的浏览窗口的HTML的<iframe>标签中。购物者可以通过一个简单的聊天窗口互相交流。

**1. 简介**

电子商务和互联网的广泛普及改变了我们如今的购物方式。网上购物的人数已达数亿人，而且仍然在继续增长。据中国互联网络信息中心的一份报告显示：2012年中国的网上购物用户数量已经达到2.42亿，美国在2015年的数据为2.02亿，而印度作为全球第二大互联网基地，截至2015年6月，虽然印度的互联网用户已达到约3.54亿人，但其电子商务的渗透率依然低于发达国家，但是，印度的网购人数正以每月600万的爆炸式增长，预计到2016年将突破1亿。不仅仅是网上购物的人数达到了数亿的规模，网上零售额也已然达到了数千亿美元。例如，2014年美国的在线消费约为3070亿美元，而2013年中国的在线消费约为3000亿美元。印度正在迎头赶上，其电子商务市场预计将以63%的复合年增长率（CAGR）在2016年达到85亿美元。

1.1 设计需求

购物是一种社会性活动，人们会在决定购买之前一起查看并讨论要购买的产品。这种一起讨论一起沟通和查看的过程会让购买者更愿意购买商品。这种思想同样也适用于网上购物的方式。然而，到目前为止，还没有这种产品可以成功的让客户体验交互式的实时共同购物体验。有部分已经实现的产品要么使用的是过时的技术，有时在性能方面会产生问题，诸如谷歌映射之类的开放api而制作的产品，要么无法在实时情况下提供这种服务。更多关于共同购物的现有作品将在下一页的“相关作品”一节中讨论。因此，需要一种能够提供网上购物的方法，使网上购物更加真实。因此，本项目选择使用Push Ajax技术可以轻松实现实时在线购物系统。

1.2 Push Ajax技术

HTTP支持从客户端到服务器的单向通信。但是它没有提供服务器反向推送的功能，如果服务器端发生任何信息变更，该项功能会立即传递给客户端。现在的大多数设计人员都实现了客户端轮询式的服务器推送，这种方式客户端每隔一段固定的时间会轮询服务器，用来获取最新的服务器数据。但是，这种设计会产生大量不需要的流量，因为即使没有服务器端更新，客户机也会重复地轮询服务器。

Push Ajax技术提供了一种轻量级、可靠和安全的方式来异步地将通知从客户机推送到服务器。它提供了一种可靠的机制促进了通知传递、容错和高可用性支持，实现不携带任何应用程序数据进行推送，从而安全性有效提高。因此，使用Push Ajax技术可以轻松地实现实时在线购物系统。

Push Ajax技术使用一种称为长轮询的技术将服务器数据更新到客户端。一旦客户端建立了连接，服务器就会给客户端一种错觉，以为还有更多的数据要来，并阻止连接。当数据可用时，服务器通过持有的连接发送数据。最后，当共同浏览结束时，服务器关闭连接。通过这种方式，可以实现从服务器到客户机的实时通信。

但是长轮询技术有两个明显的缺点。首先，默认情况下，每个服务器只允许2个连接。由于一个连接忙于使用Push Ajax技术，所以只剩下一个连接服务于客户机的请求。这可能会导致服务延迟。此外，无法控制的因素，如网络延迟，导致服务延迟。

为了在本文中实现Ajax推送，使用了ICEpush库和api。如前所述，ICEpush使用长轮询技术提供轻量级异步推送Ajax。它提供了两个主jar来实现Push Ajax-icefaces.jar和icepush.jar。为了使用ICEpush库，需要遵循一个适当实用的框架。ICEpush内部使用JSF框架将UI小部件与数据源和服务器端事件处理程序连接起来。这将在实现部分中详细讨论。

1.3 设计总结

综上所述，网上购物的工作流程如下:

a. 群主用户邀请他的线上朋友一起购物(线下朋友不能被邀请)；

b. 发现某一个中意的产品后，群主用户点击“与朋友分享这个产品”按钮；

c. 一个包含好友列表和产品URL的Ajax请求被发送到服务器；

d. 服务器通过将URL推送到朋友的所有浏览器来操作请求；

e. 服务器使用将URL推入每个客户端的HTML<iframe>标签体内。

f. 购物者可以通过聊天窗口互相交流。

**2. 相关背景**

电子商务在网上购物中的发展吸引了大量的创新用来拉近商家和买家的距离。然而，实现交互式实时共同购物的项目却非常少。在2007年，Khoury, Michel, Xiaojun Shen, Shirmohammadi.S提出了一个“对等协作的电子商务虚拟环境”。然而，它无法实现与一群朋友在一起的实时互动购物。此外，它还依赖于第三方VRML插件进行点对点通信。2012年，Li-Jia Li、Jen-Hao Hsiao还提出“基于视觉相似度的交互产品推荐在线购物”。但是，本设计只提供了一种基于相似性进行交互使产品互相推荐方法的方法，但是它却缺乏实时购物体验的实现。

Richard Han, Veronique Perret, Mahmoud Naghshine在2000年提出了一种基于XML拆分器的多设备协作Web浏览。但是，这种设计引入了一个代理来操作请求、创建和维护会话以及分割XML页面，然而这个代理却超出了预算。此外，本设计使用基于Applet的用户界面来实现客户端，这在当今科技社会中是非常少见的。但是Applet因其插件和JRE需求、启动性能和客户端安全问题而广受批评。现在的浏览器提供了比Applet更好的客户机服务器接口，此外，这种方法基于隐藏的客户端数据传输技术，并假从服务器到客户端的通用推送解决方案是不存在的，然而这是一个错误的假设。Ajax Push为一种技术，当服务器上有新的数据可用时，它会促使数据从服务器推送到客户端。

Maria Aneiros, Vladimir Estivill-Castro在2005年提出了一个基于协同web浏览名为Group Unified History (GUH)服务器的设计。在此设计中，每次用户发送HTTP请求时，代理服务器都会接收并操作该请求并将其传递给GUH服务器。然后，GUH服务器将所有这些信息推送送到客户端更新的历史记录中，并允许用户访问其他用户评论的站点。但是，本设计增加了两个额外的服务器-代理和GUH服务器，使得设计变得更加复杂。此外，这个设计是专门针对教育需求的，需要进行大量的设计更改才能实现协同在线购物系统。

K. Maly、M. Zubair和L. Li在2001年提出了一种删除代理服务器的设计，并描述了一种使用Java和JavaScript支持图形化Web浏览器的方法。然而，这种设计要求客户端用户使用支持Java的浏览器，并形成相同体验的用户组。此外，本设计再次使用了过时的基于Applet的客户端用户界面。此外，服务器上的数据变更将通过JavaScript共用指针进行监视，共用指针的相对位置被监视，用来检测浏览窗口中的变化。这是一项过时的技术，因为有更好的方法来检测浏览窗口的变化。

Alan W. Esenther在2002年提出了一种简单的基于轻量级web的实时协作web浏览器。它的即时通讯系统提供了一种快速、简单和安全的共同浏览方式。它的非侵入性本质上促进了对现有web内容的协作，而无需修改。然而，这种设计也存在一些严重的缺点和局限性。每个用户的浏览窗口的像素位置和分辨率需要相同。因此，最好在每个客户机上使用相同的浏览器品牌。此外，服务器端共同浏览窗口逻辑需要放置在与协作会话期间共享的所有web页面相同的域中。

Dietwig Lowet, Daniel Goergen 在2009年研究共同浏览动态网页技术，演示了一种使用JavaScript DOM同步实现共同浏览的出色方法。但是，这种设计要求Firefox配置允许受信任的JavaScript接收跨域脚本访问并使用Firefox套接字机制。这种跨域脚本权限可能导致执行不需要的JavaScript，并造成严重的安全问题。这也就是跨域脚本权限在默认情况下被禁用的原因。Jason J. Jung 在2011年提出了一种基于ContextGrid的基于上下文mashup的协同浏览的系统，其中使用ContextGrid模型来比较用户上下文。为了更好地理解用户上下文，还使用开放api（如谷歌映射api）来通过各种来源获取附加信息。通过各种api来比较用户上下文以实现共同浏览，这是一种非常无趣的方法，因为生成查询和过滤查询结果并找出相关结果将增加成本和复杂性。

Raphael O. Santos, Roberta L. Gomes, Felip F. Oliveira, Magnos Martinello在2011年提出了一个宏伟的设计，通过开发OCEAN原型来实现轻量级的协同Web浏览。这种设计在性能、响应能力、可扩展性和用户界面方面都达到了高效的协同浏览。此外，这种架构避免了服务器端的瓶颈，利用用户的带宽来促进同步协作体验。然而，由于基础设施不清楚，本设计提除了一个与该设计在实际场景中的可行性相关的解决方法。此外，这种设计并不适合像在线购物这样的动态web应用程序。

协同浏览领域的研究很多，但多基于主从模型、基于代理的模型或基于Javascript DOM同步的模型，这些不是过时的方法就是复杂的服务器-客户端通信方式。但是目前还没有提出支持交互式实时协同浏览的模型，这种模型与浏览器无关，实现简单，重量轻，能够满足实时网上购物的需求。

2.1 比较Flipkart旗下Ping应用

印度电子商务巨头Flipkart最近推出了名为Ping的软件程序，这是其移动应用程序的一项新功能，允许用户在购物时与朋友聊天，从而开启了协同购物体验。Ping允许用户与朋友进行实时对话，以便获取关于他们对于产品的反馈，或者帮助朋友做出购买决定。使用这一功能，用户可以直接将产品拖放到对话框中，分享他们的屏幕，从购物车中与朋友门分享产品、愿望清单、图库甚至允许他们拍摄。

但是，这个功能仅限于没有桌面支持的移动电话。而今天，大多数购物者更喜欢在电脑端上进行购物，以更好地查看产品及其细节。电脑端在线购物为用户提供了一个屏幕更大，视野更宽的界面，用户可以查看产品的详细信息、评分、评论以及各种零售商的折扣优惠和那些手机购物无法提供的其他产品选项。此外，该应用程序需要更改窗口才能从聊天窗口转到产品详细信息窗口。因此，Flipkart的Ping只是实时在线协同购物的一个开始，对桌面版协同购物的需求仍然存在。本文讨论的实现方法是在桌面环境下实现实时在线协同购物。所有的产品细节和聊天窗口都实现在同一个协作窗口中，不需要更改窗口。使用这个桌面版的网上购物，用户可以更好地了解产品细节，它的评级和评论，各种零售商的折扣优惠和产品的替代选择。

附录C 软件使用说明书

该系统面向6种不同的用户，分别是企业业务员、政府审批员、档案局人员、高新区工作人员、固定资产评估员和高新区相关领导。

（1）企业业务员

对于企业业务员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。功能包括：企业基本信息录入、企业政策申报、完成信息上传等等。

（2）政府审批员

对于政府审批员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。功能包括：对企业申报政策的增删改查。

（3）档案局人员

对于档案局人员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。功能包括：对企业完成信息的审查和对档案的增删改查操作。

（4）高新区工作人员

对于高新区工作人员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。功能包括：对角色申请和账号注册的管理、权限管理以及对企业基本信息的补录等功能。

（5）固定资产评估员

对于固定资产评估员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。功能包括：对某些属于固定资产评估公司的企业进行相关固定资产评估，并对其信息进行增删改查等操作。

（6）高新区相关领导

对于档案局人员而言，首次进入系统需要注册账号并提交相关信息和角色申请至已存在的高新区工作人员处，通过审批后方可进入系统。进入系统后根据需要可以在左侧的功能列表中获取到所需功能后点击进入功能即可。

附录D 主要源代码

（1）启动流程添加企业基本信息核心代码

// 调用方法，启动流程

activitiService.startProcess("MainFlow", variables);

/\*处理任务一：提交企业信息\*/

// 将业务ID注入流程信息中

processService.updateBUSIKeyById(enterprise.getEnterpriseId(), userName);

// 获取当前流程ID

String proByBUSI = processService.findProByBUSI(enterprise.getEnterpriseId());

// 获取一个随机评估员信息

List<User> user = userService.findAllAssess();

int t = new Random().nextInt(user.size());

User sample = user.get(t);

String assessId = sample.getUsername();

variables.put("assessId", assessId);

// 处理任务

activitiService.completeTask(proByBUSI, variables);

Message msg = new Message("yes");

return msg;

（2）附件上传核心代码

@ResponseBody

@RequestMapping("/uploadFile")

public String uploadFileById(@RequestParam("file") MultipartFile file,

@RequestParam("enterpriseId") Integer enterpriseId) {

String path = UploadUtils.upload(UPLOAD\_DIR, file);

enterpriseService.updatePathById(path, enterpriseId);

return "redirect:getAllInfo";

}

（3）读取Word表格信息核心代码

public static void parseWord(String filePath) throws Exception {

Iterator<XWPFTable> it = getWordContent(filePath);

while(it.hasNext()){

XWPFTable table = it.next();

List<XWPFTableRow> rows = table.getRows();

//读取每一行数据

for (int i = 0; i < rows.size(); i++) {

XWPFTableRow row = rows.get(i);

//读取每一列数据

List<XWPFTableCell> cells = row.getTableCells();

for (int j = 0; j < cells.size(); j++) {

XWPFTableCell cell = cells.get(j);

//输出当前的单元格的数据

System.out.print(cell.getText() + "\t");

}

System.out.println();

}

}

}

（4）导出Excel核心代码

@RequestMapping("/excelDown")

public String excelDown(HttpServletResponse response,

HttpServletRequest request) throws Exception {

/\*获取Web项目根目录\*/

String path = request.getServletContext().getRealPath("/");

/\*定义模板的位置\*/

String url = path + "/model/Enterprise\_model.xls";

/\*获取全部数据\*/

List<Enterprise> list = enterpriseService.findAll();

/\*将数据导入Excel模板\*/

Workbook workbook = FillDataInModel.EnterpriseDataExcel(list, url);

ResponseUtils.export(response, workbook, "企业信息.xls");

return null;

}

（5）导出档案核心代码

@ResponseBody

@RequestMapping("/downloadFile")

public void downloadFile(Integer id, HttpServletResponse response, HttpServletRequest request) throws Exception {

String str = id.toString();

Enterprise enterprise = enterpriseService.findById(id);

Examine examine = examineService.findById(id);

Policy policy = policyService.findById(str);

Check check = checkService.findById(id);

Complete complete = completeService.findById(id);

/\*获取Web项目根目录\*/

String path = request.getServletContext().getRealPath("\\");

/\*定义模板的位置\*/

String url = path + "\\model\\File.xls";

Workbook workbook = FillDataInModel.FileExcel(enterprise, policy, examine, check, complete, url);

ResponseUtils.export(response, workbook, "档案信息.xls");

}

（6）下载所有档案信息为Excel核心代码

@RequestMapping("/download")

public void download(Integer fileId, HttpServletRequest request, HttpServletResponse response) throws Exception {

// 通过fileID查询当前附件路径

File FileById = fileService.findById(fileId);

String filePath = FileById.getFilePath();

// 获取文件的绝对路径名称

String path = UPLOAD\_DIR + "\\" + filePath;

// 获取后缀名

String suffix = filePath.substring(filePath.lastIndexOf(".") + 1);

String filename = "档案." + suffix;

ResponseUtils.download(path, filename, response);

}

（7）时间段查询核心代码

public List<Assess> queryByTimeBucket(String beginDate, String endDate) {

List<Assess> list = new ArrayList<>();

try {

/\*获取时间段内所有时间集合\*/

List<String> dateStrs = DateUtils.getDayListOfDateStr(beginDate, endDate);

/\*循环遍历集合内所有时间\*/

assert dateStrs != null;

for (String dateStr : dateStrs) {

/\*将dateStr转换为Date格式\*/

Date date = DateUtils.parseStrToDate(dateStr);

/\*通过转换后的Date数据对数据库进行查询并存储至assessList\*/

List<Assess> assessList = assessDao.findByTime(date);

/\*遍历assessList将其中元素加入list中\*/

list.addAll(assessList);

}

return list;

} catch (ParseException e) {

e.printStackTrace();

return null;

}

}

（8）新建评估传递流程核心代码

@ResponseBody

@RequestMapping("/insertAssess")

public Message insertAssess(Assess assess, @RequestParam("userName")String userName) {

// 添加评估信息

assessService.insertAssess(assess);

/\*处理任务二：固定资产评估\*/

// 通过ID查询对应企业信息的用户账号

Enterprise enterprise = enterpriseService.findById(assess.getAssessId());

String username = enterprise.getUsername();

// 将用户账号写入map集合注入流程

Map<String, Object> variables = new HashMap<>();

variables.put("clerkId", username);

// 获得流程ID

String proByBUSI = processService.findProByBUSI(assess.getAssessId());

// 处理任务

activitiService.completeTask(proByBUSI, variables);

/\*向前端传递信息\*/

Message msg = new Message("yes");

return msg;

}

（9）添加评阅信息处理分支核心代码

@RequestMapping("/check")

public void check(Check check) {

// 获取check的id，判断其是否存在

Check i = checkService.findById(check.getCheckId());

if (i == null) {

// 添加审阅信息

checkService.insertCheck(check);

}

else {

// 修改审批信息

checkService.updateCheck(check);

}

/\*处理任务五：政策审阅\*/

Map<String, Object> variables = new HashMap<>();

// 获取check中的审批结果，并判断分支

String checkResult = check.getCheckResult();

if (checkResult.equals("通过")) {

variables.put("approve", true);

} else {

variables.put("approve", false);

}

// 获得流程ID

String proByBUSI = processService.findProByBUSI(check.getCheckId());

// 处理任务

activitiService.completeTask(proByBUSI, variables);

}

（10）编辑企业信息核心代码

@RequestMapping("/deleteEnterprise")

public String deleteEnterprise(Integer enterpriseId) {

enterpriseService.deleteEnterprise(enterpriseId);

return "redirect:getAllInfo";

}

（11）删除企业信息核心代码

@RequestMapping("/editEnterprise")

public String editEnterprise(Enterprise enterprise, Model model){

model.addAttribute("enterprise", enterpriseService.findById(enterprise.getEnterpriseId()));

return "EnterpriseEdit";

}

（12）获取Word单元格内容

public static List<String> completeWordParse(String filePath) throws Exception {

Iterator<XWPFTable> it = getWordContent(filePath);

List<String> list = new ArrayList<>();

while(it.hasNext()){

XWPFTable table = it.next();

List<XWPFTableRow> rows = table.getRows();

list.add(getRowText(rows, 0, 1));

list.add(getRowText(rows, 0, 3));

list.add(getRowText(rows, 0, 5));

list.add(getRowText(rows, 1, 1));

list.add(getRowText(rows, 1, 3));

list.add(getRowText(rows, 2, 1));

list.add(getRowText(rows, 2, 3));

list.add(getRowText(rows, 3, 1));

list.add(getRowText(rows, 3, 3));

list.add(getRowText(rows, 5, 0));

list.add(getRowText(rows, 6, 1));

list.add(getRowText(rows, 6, 3));

list.add(getRowText(rows, 8, 0));

}

return list;

}

（13）企业信息分页展示

@RequestMapping("/getAllInfo")

public String getlist(ModelMap model,

/\*页码\*/

@RequestParam(defaultValue = "1", required = true, value="pn") Integer pn,

/\*条数\*/

@RequestParam(defaultValue = "5", required = true, value="ps") Integer ps

) {

PageHelper.startPage(pn, ps);

List<Enterprise> enterprise = enterpriseService.findAll();

PageInfo<Enterprise> pageInfo = new PageInfo<Enterprise>(enterprise);

model.addAttribute("pageInfo", pageInfo);

return "EnterpriseIndex";

}

（14）添加政策审批

/\*处理任务四：政策审批\*/

Map<String, Object> variables = new HashMap<>();

// 获取examine中的审批结果，并判断分支

String examineResult = examine.getExamineResult();

if (examineResult.equals("通过")) {

variables.put("exam", true);

variables.put("leaderId", "leader");

} else {

// 找到申请政策的用户

Integer id = examine.getExamineId();

Enterprise enterprise = enterpriseService.findById(id);

String userName = enterprise.getUsername();

variables.put("exam", false);

variables.put("clerkId", userName);

}

// 获得流程ID

String proByBUSI = processService.findProByBUSI(examine.getExamineId());

// 处理任务

activitiService.completeTask(proByBUSI, variables);