# 需求分析

一、需求分析：毕业设计辅助管理系统 - 任务管理模块

1. 业务功能分析：

- 教师角色：

- 发布任务：教师可以创建并发布任务，包括任务标题、描述、截止日期等信息。

- 分配任务：教师可以将任务分配给指定的学生或学生组，以便学生可以接收到任务。

- 接受任务反馈：教师可以接收学生对任务的反馈，包括文字描述、附件上传等。

- 审核任务结果：教师可以审核学生提交的任务结果，确定是否符合要求。

- 学生角色：

- 接收任务：学生可以接收到教师分配给他们的任务。

- 提交任务反馈：学生可以对任务进行反馈，包括文字描述、附件上传等。

- 查看任务结果：学生可以查看教师审核后的任务结果。

2. 输入输出说明：

- 教师发布任务：

- 输入：任务标题、任务描述、截止日期等任务信息。

- 输出：任务信息成功保存，任务在系统中可见。

- 教师分配任务：

- 输入：选择学生或学生组，将任务分配给他们。

- 输出：任务成功分配给指定学生或学生组。

- 学生提交任务反馈：

- 输入：文字描述、附件上传等任务反馈信息。

- 输出：任务反馈成功保存，教师可见该反馈信息。

- 教师审核任务结果：

- 输入：查看学生提交的任务反馈信息。

- 输出：审核结果，包括任务通过或未通过。

- 学生查看任务结果：

- 输入：查看自己提交任务的审核结果。

- 输出：任务审核结果，包括任务通过或未通过。

3. 界面设计：

- 教师任务管理界面：

- 提供发布任务按钮，用于创建并发布新的任务。

- 显示已发布的任务列表，包括任务标题、截止日期等信息。

- 提供分配任务按钮，用于将任务分配给指定的学生或学生组。

- 提供查看任务反馈按钮，以查看学生对任务的反馈信息。

- 提供审核任务结果的界面，用于对学生提交的任务结果进行审核。

4．学生任务管理界面：

- 显示已接收的任务列表，包括任务标题、截止日期等信息。

- 提供提交任务反馈的界面，包括文字描述、附件上传等功能。

- 显示任务审核结果，标识任务是否通过审核。

二、各分模块需求分析

**前端**

1.教师任务分配界面、教师任务接受界面

1. 界面设计：根据用户需求设计两个界面，任务分配和任务接收（老师）
2. 任务分配：老师将任务分配给每个学生，确保学生都收到了任务，并指明学生作业提交地址。
3. 任务提交：学生根据老师分配的任务进行选取、执行，完成后将任务命名、打包压缩后，通过老师指定的路径，将文件进行提交。
4. 任务接收：老师通过学生提交后的作业进行审核，符合要求的通过，未符合的进行文件返回，学生进行修改。
5. 用户界面要求：简单、快捷、高效。

2.任务管理登录界面需求分析学生任务编辑界面

目的：目的：

设计一个简单、清晰、引导明确的用户制作一个任务浏览、编辑界面。要求用

登录或注册界面。户可以在此界面完成查看、修改等功能。

功能：功能：

1区别用户身份（教师/学员）登录页面。1点击查看，显示具体信息。

2实现跳转到注册界面的功能。2建立新任务。

3具有用户账号输入模块。3对现有任务进行删除。

4具有用户密码验证模块。4对现有任务内容进行改动。

5确认注册密码。

6登录/注册成功提示窗口。

3.任务交互界面

界面设计：

1.教师审核界面显示学生提交的任务文件和任务说明，提供评分和评语的输入

框，以及通过和不通过的选择按钮。

2.教师反馈界面显示教师提供评分和评语的输入框，以及给学生建议和意见的文

本框。

3.学生查看审核与反馈界面显示教师的审核意见、评分和评语，以及是否通过任

务。

一、教师审核任务的需求：

1.教师能够查看学生提交的任务，任务文件和任务说明。

2.教师能够审核学生提交的任务，确认任务是否符合要求。

3.教师能够评价学生提交的任务，给出评分和评语。

4.教师能够给出审核结果，判断通过或不通过，并填写审核意见。

二、教师反馈的需求：

1.教师能够提供具体的建议和意见，帮助学生改进任务质量。

2.教师能够及时反馈给学生，以便学生对任务进行修改和完善。

三、学生查看审核与反馈的需求：

1.学生能够查看自己的任务进度，包括任务状态和教师评价。

2.学生能够查看教师的审核意见，包括通过或不通过的原因。

3.学生能够查看教师的评分和评语，了解自己的任务质量。

4.学生任务接受与提交界面

1. 任务列表展示：界面应该显示当前学生所分配到的任务列表。每个任务应该包含任务标题、任务描述、截止日期等信息，方便学生了解任务的内容和要求。

2. 任务状态标识：对于已接受的任务，界面应该有相应的状态标识，以区分已完成、未完成、审核中等状态，帮助学生了解任务的完成情况。

3. 接受（下载）任务：学生可以根据自己的意愿选择接受任务。界面应该提供相应的操作按钮，如"接受任务"，同时可以让学生下载任务的附件。

4. 任务详情查看：学生可以点击任务列表中的任务标题或详情按钮，查看任务的详细内容。界面应该以清晰的方式展示任务的描述、要求、截止日期等信息，以便学生了解任务的具体要求。

5. 提交任务：界面应该提供一个提交任务的入口。学生可以点击该按钮，上传自己完成的任务内容，如代码文件、文档、报告等。界面应该支持文件上传功能，方便学生提交任务材料。

6. 任务反馈与评价：界面应该提供相应的反馈与评价功能，对已完成的任务学生可以接受老师的评价，意见，反馈。

7. 用户友好性：界面应该具备良好的用户友好性，清晰易懂的界面设计、简洁明了的操作方式，能够提供良好的用户体验。

**后端**

1. 角色权限跳转功能

1. 功能描述：

- 用户通过登录页面进行身份验证，根据其角色的不同，跳转到相应的页面。

- 每个角色对应的页面可以是系统内置的页面，也可以是自定义的页面。

2. 角色定义：

- 系统管理员（admin）：拥有最高权限，可以访问系统的所有功能和页面。

- 普通用户（user）：拥有普通权限，可以访问系统的部分功能和页面。

- 其他自定义角色：根据具体需求，可以定义更多的角色，并分配相应的权限和页面。

3. 登录页面：

- 提供用户名和密码输入框，用户输入正确的用户名和密码进行登录。

- 提供登录按钮，点击后触发登录验证过程。

4. 登录验证：

- 后端应实现登录验证逻辑，包括用户名和密码的验证。

- 如果验证通过，根据用户的角色信息确定跳转到的页面。

- 如果验证失败，返回错误信息给前端，提示用户重新输入正确的用户名和密码。

5. 页面跳转：

- 登录成功后，根据用户的角色信息，跳转到相应的页面。

- 系统管理员跳转到系统管理页面，可访问系统的所有功能。

- 普通用户跳转到普通用户页面，可访问部分功能。

- 其他自定义角色根据需求进行相应的跳转。

6. 页面权限控制：

- 后端应实现页面级别的权限控制，确保用户只能访问其角色被授权的页面。

- 页面权限可以通过角色的配置或者角色与页面的映射关系进行控制。

- 当用户尝试访问未授权的页面时，应返回错误信息或跳转到无权限页面。

7. 错误处理：

- 后端应处理可能出现的异常情况，例如数据库连接失败、权限配置错误等。

- 在发生错误时，应返回友好的错误信息给前端，帮助用户快速定位和解决问题。

8. 安全性考虑：

- 用户密码应进行加密存储，以确保用户数据的安全性。

- 登录过程中应考虑防止常见的安全威胁，如跨站脚本攻击（XSS）和跨站请求伪造（CSRF）等。

任务上传（分配、新建）功能（教师）

上传（新建）任务：教师通过网页上传按钮，把所需文件上传，并把文件名和路径等信息保存到文件表中  
任务分配：教师通过网页分配按钮，把任务分配给对应学生，并将任务名保存到学生表中（可以自己选择想发布任务的对象）

2.任务下载（接受）功能

1. 业务功能分析:  
   分为两个方向的传输，分别是学生端根据需要接受教师端发来的任务，以及教师端需要下载学生端发来的任务反馈文件。  
   2、输入输出说明:  
   输入：数据库内存储的文件和任务。输出：学生下载任务接口，教师下载任务接口。输入输出是通过数据库做为一个中间桥梁。  
   3、定义所需要的外部资源：  
   流通的任务通过数据库进行存储。  
   4、系统实现所涉及的难点和关键点：  
   难点是要实现满足两个方向的传输，在一个共享平台进行多端传输对文件的资源分享和任务选择。其中关键点首先是学生和老师任务接口的分离，其次是建立数据库链接，最后是任务文件的传输。

3.任务编辑功能

学生的功能主要有以下三点：  
（1）查看学校题目列表，把老师上传的毕业论文任务书以列表的形式进行显示。  
（2）上交毕业设计：学生填写自己的论文题目，并添加本地的文件来进行毕业设计论文的上传 。  
（3）查看自己已上交毕业设计：把自己上传的文件信息已列表的形式进行显示。  
（4）修改自己上交的信息。

4.（学生）任务列表查看功能  
目的：供学生查看任务详细内容 接收任务  
功能：显示登录学生所分配到的任务信息，包含任务名称、任务内容、导师姓名、任务状态、截止时间等  
说明：依赖登录的学生用户对应信息、数据库信息调用  
界面设计：显示一个表格，有任务的名称、导师信息、任务内容、任务状态，可通过搜索任务名称来查看任务

5.（教师）任务列表查看功能  
目的：供教师查看自己已发布的任务 可通过页面查询和查看到学生的姓名、学号和任务完成状态。  
业务功能分析：显示教师发布的任务，包含学生姓名、学号、完成状态。可查询不同学生的完成情况。  
输入输出说明：本部分依赖的模块是登录页面。通过登录页面积累信息，输入到查看功能主页面。教师用户对应信息、数据库信息调用。  
界面设计：设计一个表格，依次包含学生姓名、学号、完成状态。可查看所有学生的信息。顶部有学生姓名与学号的查询区域。  
系统实现涉及的关键点：关键点是与前端的融合和对应。

1. 任务编辑功能（教师）删、改、查

设计一系列功能, 完成任务的<del>增</del>删改查

# 提供功能/API:

查询所有任务 `GET /scd/tasks/list`

新建一个任务 <del>`POST /scd/tasks/add`</del> 由其他组员完成

编辑单个任务 `PUT /scd/tasks/edit`

删除单个任务 `DELETE /scd/tasks/remove`

中止单个任务 `PUT /scd/tasks/abort`

查询单个任务 `GET /scd/tasks/{taskId}`

**数据库**

概述：数据库用来存储数据，管理数据资源。为了支持后端功能的实现，我目前定义以下数据表（暂定），数据库会随着项目开发的过程更改优化：

教师表：

|  |  |
| --- | --- |
| 键 | 说明 |
| 姓名 | 教师姓名 |
| ID号 | 唯一标识号 |
| 所属教研室 | 教研室名称 |

学生表：

|  |  |
| --- | --- |
| 键 | 说明 |
| 姓名 | 学生姓名 |
| 班级 | 所属班级 |
| ID号 | 唯一标识号 |
| 所属组 | 跟随的老师 |
| 任务组 | 被分配的任务 |

任务表：

|  |  |
| --- | --- |
| 键 | 说明 |
| 名称 | 任务名称 |
| ID号 | 唯一标识符 |
| 描述 | 任务说明 |
| 发布日期 | 任务发布日期 |
| 截至日期 | 任务截至日期 |
| 任务状态 | 任务是否完成，待处理等 |
| 任务发布者 | 教师姓名 |
| 任务参与者 | 参与任务的学生集合 |

任务反馈表

|  |  |
| --- | --- |
| 键 | 说明 |
| ID号 | 唯一标识符 |
| 关键字 | 评论标题 |
| 所属任务 | 关于某个任务的评论表 |
| 评论内容 | 内容 |
| 发布时间 | 评论发布时间 |
| 最后时间 | 最后一个评论时间 |
| 状态 | 评论关闭、开启、禁止等 |