

软件需求规格说明书



|  |  |
| --- | --- |
|  |  |

项目主题 学校体育系统

**吴浩翔 范泽宇 刘培源 叶佳浩 董佳鑫**

**目录**

**1引言**

1.1编写目的

1.2项目背景

1.3名词定义

**2总体描述**

**3类图和CRC模型**

**4非功能性需求**

4.1性能需求

4.2安全性

4.3可维护性和可拓展性

4.4可靠性

4.5其他需求

**5数据流图**

**6验收准则**

**7UI原型**

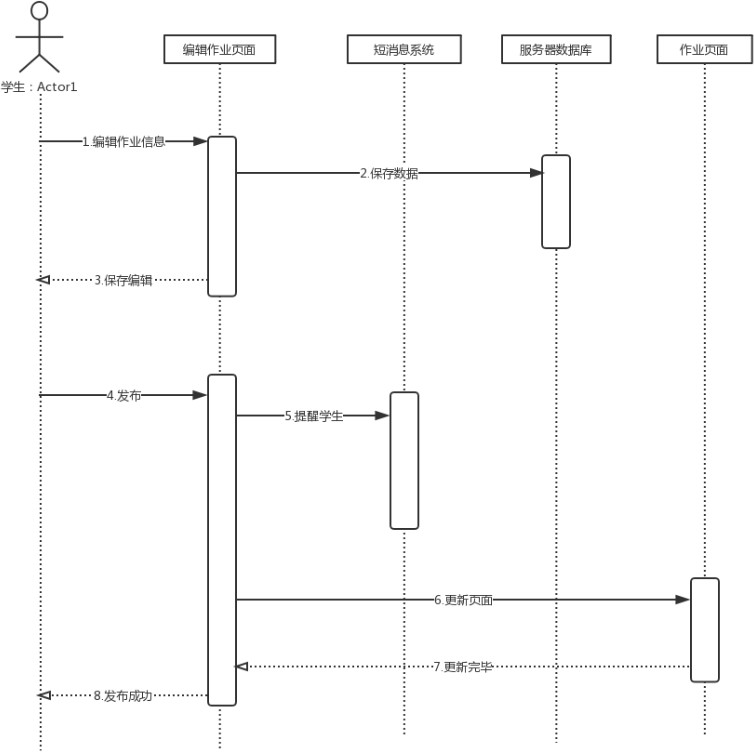
**3.3.5 设置作业**

**3.3.5.1 描述及优先级**

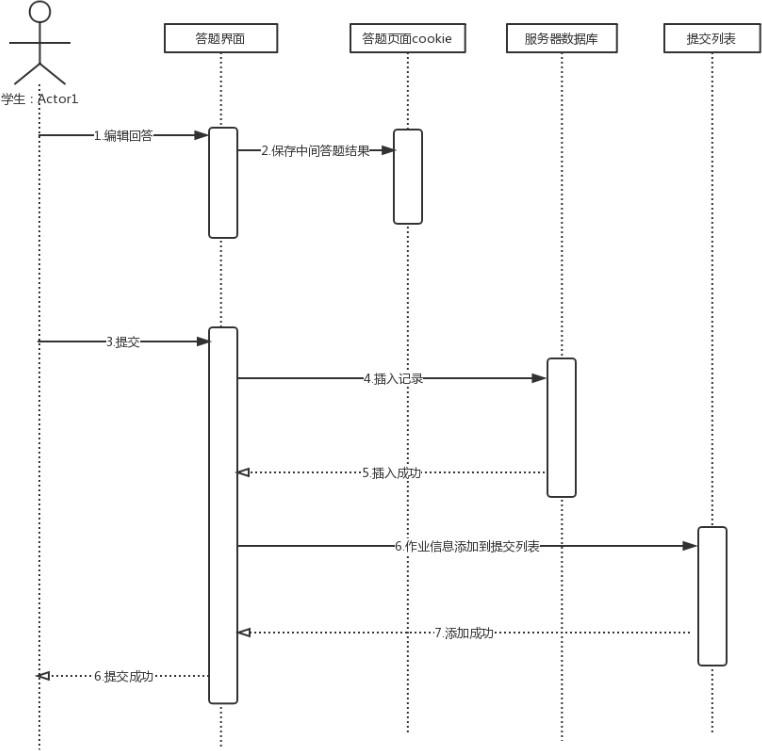
教师以及助教应该能在课程开展的过程中在教学系统上布置作业，并且可以把发布通知，让每个学生知道作业的相关信息。作业的形式要能由教师定制，可以设置题目的类型、学生提交作业的时间。学生能够通过系统提交作业，并可以跟踪作业的批改情况。学生可以通过设置主动收到系统提供的作业反馈。

优先级：高

**3.3.5.2 流程请求/响应时序图**



教师：Actor1



**3.3.5.3 用例文档**

|  |  |
| --- | --- |
| 用例名称 | 发布作业 |
| 行为角色 | 教师、助教 |
| 简要说明 | 教师或助教可以根据课程的需求发布作业，作业要求应该清楚的传达给  学生 |
| 前置条件 | 用户以教师或者助教的身份登录 |
| 后置条件 | 在服务器为作业创建一个新的 ID，根据 ID 创建新的存储目录。自动将  新作业的信息传递给学生，并且在课程的作业列表完成更新。 |

|  |  |
| --- | --- |
| 流程 |  |
| 异常处理 | 编辑的作业通知时网页发生了异常：可以实现内容的主动和自动保存 |
| 备注 | 1. 教师或助教在完成内容编辑以后可以设置发布的时间，系统可以在到达预定时间以后自动的更新。 2. 教师可以设置一个提醒时间，使得没有完成作业的学生可以在作业截止   时间之前再次得到通知 |

|  |  |
| --- | --- |
| 用例名称 | 提交作业 |
| 行为角色 | 学生 |
| 简要说明 | 学生通过系统按照课程要求提交作业 |
| 前置条件 | 用户以学生身份登录，且学生属于该门课程，提交作业时未超过最终时  限 |
| 后置条件 | 系统将提交的作业内容保存在数据库里，并做相应的记录 |
| 流程 |  |
| 异常处理 | 1. 学生在完成作业的过程中网站异常关闭：手动和系统自动保存作业。 2. 学生完成作业以后忘记提交：系统在截止日期到达时自动提交学生最后一次保存的作业。 |
| 备注 | 1. 在教师完成批改后会以短消息的形式通知学生，学生要能看到自己小   题的得分情况 |

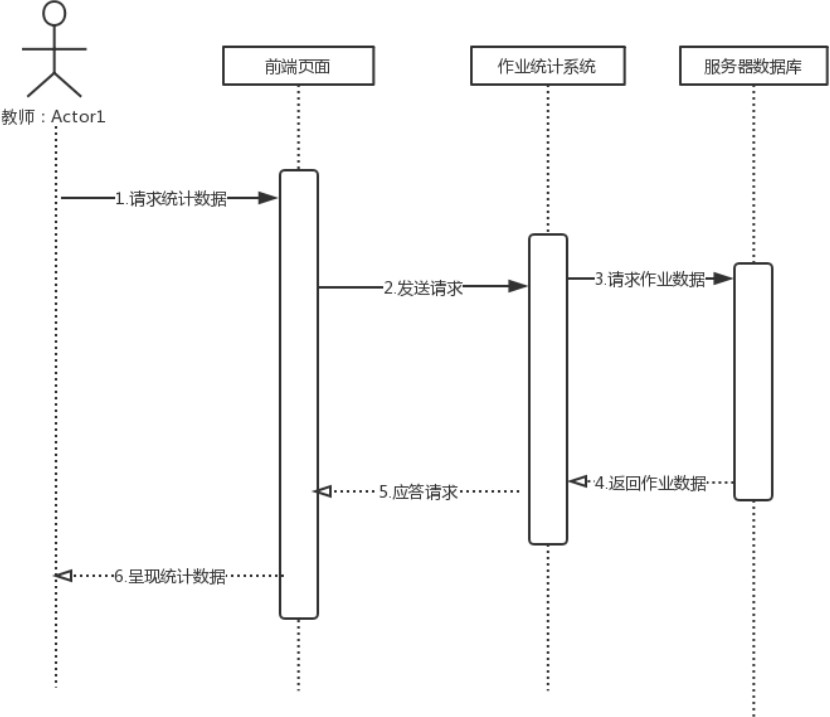
**3.3.6 作业统计信息反馈**

**3.3.6.1 描述及优先级**

教师在学生完成作业以后可以获取学生作业完成情况的统计信息，包括完成作业学生的比例、学生的平均分数，每道习题的得分情况。此外，教师或助教还可以搜索作业情况异常的学生。

优先级：中

**3.3.6.2 流程请求/响应时序图**



**3.3.6.3 用例文档**

|  |  |
| --- | --- |
| 用例名称 | 作业统计信息反馈 |
| 行为角色 | 老师 |
| 简要说明 | 老师可以得知学生完成作业的情况 |
| 前置条件 | 用户以老师或助教身份登录，且老师属于该门课程 |
| 后置条件 | 系统将数据库的相应数据传递给信息处理程序，等到处理完成以后再将信  息从前端展示出来。 |
| 流程 |  |
| 备注 | 1. 在教师可以自行设置参数来筛选异常的学生，筛选的方式包括多次未完成作业，多次完成作业得分过低。 2. 教师可以选择 checkbox 从而展示相应的信息 |

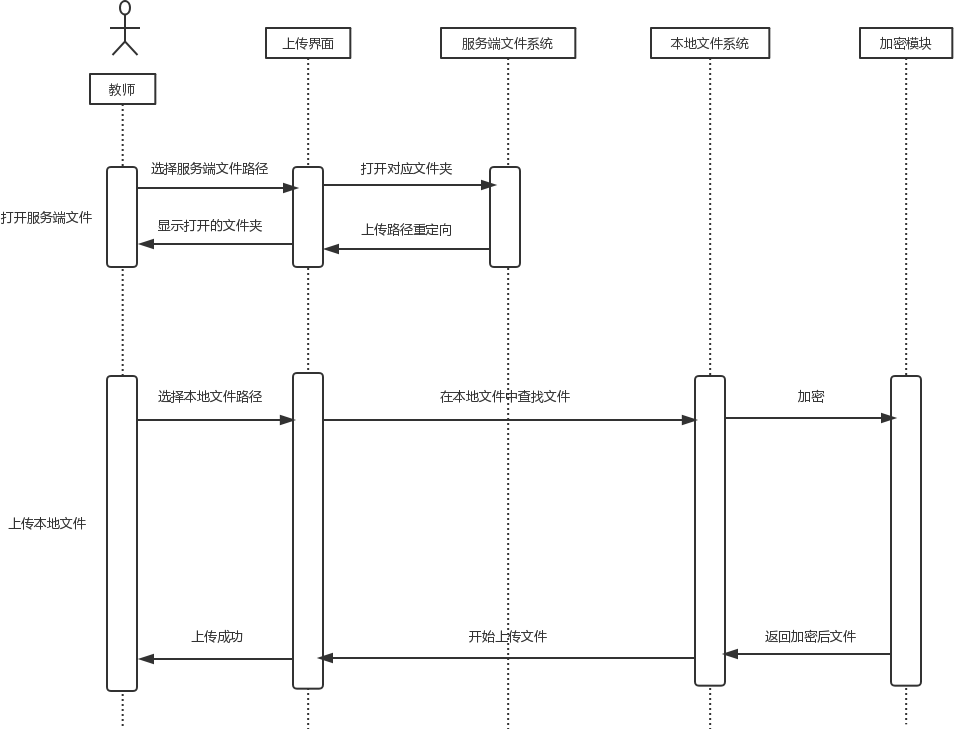
**3.3.7 教学文件管理**

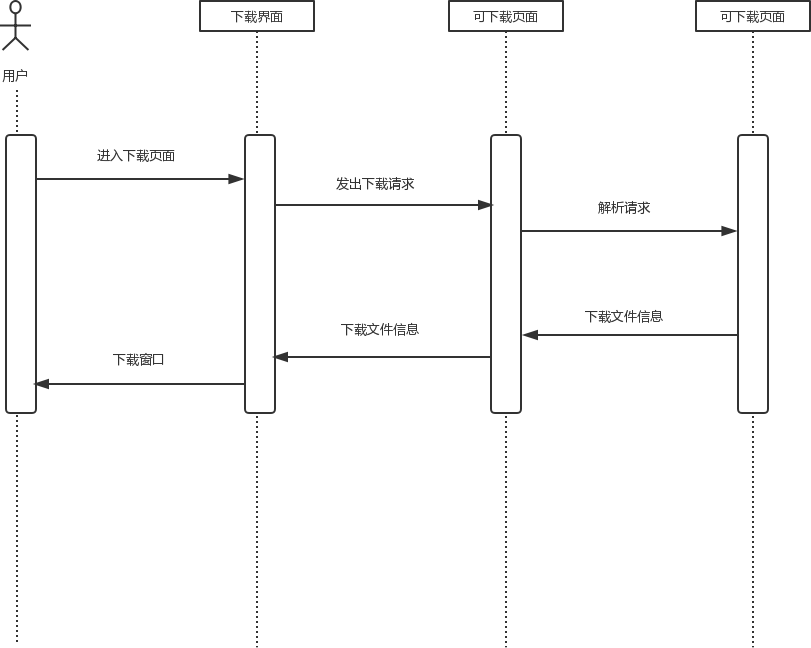
**3.3.7.1 描述及优先级**

在我们的教学系统中，为了方便同学们的学习，教师和助教应当可以在课程页面中上传课件、课程资料、往届同学作业、教学音频视频等等教学资源文件。我们的教学系统需要为教师、助教以及学生提供下载服务端存储的文件的功能。并且此功能需要保证一定的并发行，比如可以支持至少 50 人同时下载，并且人均的下载速度能够达标，比如至少 50kb/s。

优先级：高

**3.3.7.2 流程请求/响应时序图**





**3.3.7.3 用例文档**

|  |  |
| --- | --- |
| 用例名称 | 上传教学文件 |
| 操作角色 | 教师、助教 |
| 行为说明 | 教师或者助教可以通过该功能随时更新、上传和删除在服务端上供  学生参考的教学资源文件 |
| 前置条件 | 用户以教师或者助教的身份登陆 |
| 后置条件 | 上传的教学文件会保存在服务器数据库中，并更新相关页面 |
| 流程 |  |
| 异常处理 | 若上传了不支持的文件格式，则取消上传，并返回错误信息。若本地  文件路径错误导致无法找到资源文件则取消上传，并返回错误信息 |
| 备注 | 可支持的文件类型有文本文件 (.doc .docx .txt) ， 图 片 文 件 (.bmp .jpg .png)，视频文件(.mp4 .avi .mkv)，压缩文件(.rar .zip)和其他文件(.pdf .ppt .xlsx) |

|  |  |
| --- | --- |
| 用例名称 | 下载教学文件 |
| 操作角色 | 教师、助教、学生 |
| 行为说明 | 教师、助教和学生可以下载教师用户上传的教学资源以及学生上传  的作业文件 |
| 前置条件 | 用户以教师、助教或者学生的身份登陆 |
| 后置条件 | 创建下载任务 |

|  |  |
| --- | --- |
| 流程 |  |
| 异常处理 | 用户下载的文件资源失效：显示错误信息页面 |
| 备注 | 需要保证用户的下载速度，并且保证一定的下载并发性。除此之外，  未登录的用户不可下载文件资源 |

**4 非功能性需求**

4.1 性能需求

* 系统应保证运行稳定，尽量避免出现崩溃。
* 系统对于资源的利用率应该比较高效，避免大量占用移动端资源。
* 系统应该支持目前各种操作系统的移动端的正常访问。
* 系统应该能支持较高的并发访问，应能保证至少 1000位用户的并发访问，保证在 高峰时段用户能够顺利访问系统。
* 对于用户对系统的所有操作，系统都应在 3s 以内做出应答。
* 系统应该能及时检测并报告各种非正常情况，如设备的通信终端连接失败，无法连 接数据库服务器等，避免长时间等待。
* 每个页面一般情况下应在 3s 内加载完毕，高峰期应在 6s 内加载完毕。
* 系统保证在一个月内不超过一次的维护与重启。
* 实时记录学生跑步距离，如果学生手机中途退出系统，支持断点重连，维持退出系 统前的跑步距离。

4.2 安全性

安全性包括数据安全和系统安全，以下是一些具体细则：

* 只有系统管理员有权查看系统日志。
* 任何人都无权修改或删除日志。
* 对于用户位置信息，要进行严格保护。
* 每个学生只能查看自己的信息，包括打卡次数、项目成绩等。
* 教师有权查看教学班下的所有学生的相关考核信息，如项目成绩等。
* 系统应具备防御常见网络攻击的能力，如SQL注入、跨站脚本攻击或者其他外挂 行为等。
* 系统应定期进行安全漏洞扫描和渗透测试，及时发现并修复潜在的安全隐患。
* 允许系统管理员进行数据的备份和恢复，以防止数据的破坏和丢失。
* 建立数据库的安全保障体系，提供数据的错误恢复功能。

4.3 可维护性和可拓展性

* 代码应当模块化，将功能划分为独立的模块，降低耦合性，提高可拓展性。
* 系统代码应该遵循一致的编程规范和命名规定，便于代码的后续维护。
* 系统应当提供完整、准确的系统说明文档，包括系统设计、数据库系统构建、API 文档等。
* 代码应该有充分且规范的注释，解释变量、函数等关键逻辑，便于后续其他开发人 员理解。
* 提供详细的日志记录功能，包括系统运行日志、用户操作日志等，便于追踪和定位 问题。
* 系统设计应当充分考虑未来的拓展性需求，以便未来能够比较方便的添加新功能和 新模块。
* 提供单元测试和集成测试的工具或方法，确保代码的质量和正确性。
* 提供可扩展的API和接口，方便与其他系统进行集成或对接。
* 系统提供的接口应该保持稳定，避免频繁更改接口的定义和参数，减少对其他模块 和功能的影响。
* 系统开发文档、用户手册和技术支持文档应保证定期进行更新，与系统版本保持同 步。
* 系统应当提供用户管理和权限控制机制，方便管理员对用户进行添加、删除和修改 操作。
* 系统应该能处理更大规模的数据量，并保持高效的查询和处理功能。
* 避免过度依赖某个特定的技术或工具，保持系统的开放性和灵活性。
* 系统应提供性能监控和调优机制，能够根据实际情况调整系统参数和资源分配，优 化系统性能。

4.4 可靠性

* 系统应确保在任何情况下，数据的一致性和完整性都不会被破坏。这包括在并发操 作、网络中断或系统故障等情况下，数据仍能保持一致和完整。
* 系统故障发生后，系统应能迅速恢复到正常状态，确保服务的连续性和数据的完整 性。
* 系统应提供数据校验和验证机制，防止无效或错误的数据进入系统。
* 系统对于高并发、大数据量等场景进行充分的测试和优化，确保系统能够稳定地处 理各种复杂情况。
* 系统应提供持续的性能和可靠性监控功能，实时检测系统的运行状态和潜在问题。
* 系统应当设计并实施灾难恢复计划，包括数据备份、应急演练等，以应对可能的自 然灾害、硬件故障等不可抗力事件。
* 系统应当具备强大的安全机制，防止未经授权的访问、数据泄露和恶意攻击，确保 用户数据和系统的安全。
* 系统应该保持性能稳定，在各种负载条件下都应保持相对稳定的性能，不会出现性 能明显下降或崩溃的情况。
* 系统应提供回滚机制，以便在更新失败时能够迅速恢复到更新前的状态。

4.5 其他需求

* 系统界面设计应简洁明了，布局合理，避免过多的复杂元素和冗余信息。
* 系统应该有足够简化的操作步骤，避免冗余和不必要的操作，提高用户的使用效率。
* 系统应采用用户熟悉的界面元素和图标，确保用户能够迅速理解界面上的各项功能 和操作。
* 当用户操作不当或系统出现错误时，应提供清晰、具体的错误信息提示，指导用户 进行修正或解决问题。
* 系统应具备良好的兼容性，能够在不同的操作系统和设备上正常运行。
* 系统应当支持用户反馈功能，定期收集用户各种反馈，以便于总结需求，指导后续 的开发方向。