OMSE 555/556 Software Engineering Practicum I & II

Software Architecture Document

Distributed Development Monitoring and Mining

Tom Mooney, Shailesh Shimpi, Ahmed Osman, Isaac Pendergrass

Version 1.0

2019.10.2

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description of Versions / Changes** | **Responsible Party** | **Date** |
| 1.0 | 1.前言 | 陈铮 | 2019.10.20 |
|  | 2.视图含义 | 陈庆忠 | 2019.10.20 |
|  | 3.系统结构目标和条件 | 陈庆忠 | 2019.10.20 |
|  | 4.用例图4.1.1-4.1.5 | 陈铮 | 2019.10.20 |
|  | 4.用例图4.1.6-4.1.10 | 陈庆忠 | 2019.10.20 |
|  |  |  |  |

**Approval Block**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Comments** | **Responsible Party** | **Date** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Introduction 1](#_Toc353470033)

[1.1. Purpose 1](#_Toc353470034)

[1.2. Scope 1](#_Toc353470035)

[1.3. Definitions, Acronyms, and Abbreviations 2](#_Toc353470036)

[1.4. References 2](#_Toc353470037)

[1.5. Overview 2](#_Toc353470038)

[2. Architectural Representation 3](#_Toc353470039)

[3. Architectural Goals and Constraints 4](#_Toc353470040)

[4. Use-Case View 4](#_Toc353470041)

[4.1. Actors 4](#_Toc353470042)

[4.2. Use-Case Realizations 5](#_Toc353470043)

[4.2.1. Login 5](#_Toc353470044)

[4.2.2. Request Analysis (Get Report) 7](#_Toc353470045)

[4.2.3. Retrieve Last Report 9](#_Toc353470046)

[4.2.4. Print Report 10](#_Toc353470047)

[4.2.5. Email Report 11](#_Toc353470048)

[5. Logical View 12](#_Toc353470049)

[5.1. Overview 12](#_Toc353470050)

[5.2. Interface Definitions 14](#_Toc353470051)

[6. Data View 23](#_Toc353470052)

[7. Deployment View 25](#_Toc353470053)

# 前言

本文提供了一个高层次的概述，并解释了教学综合管理系统的体系结构。

该文档定义了体系结构的目标、系统支持的用例、体系结构样式和已选择的组件。该文档提供了从概念概念到实现的架构和设计决策的基本原理。

## 目标

软件体系结构文档（sad）提供了教学综合管理系统的全面体系结构概述。它提供了许多不同的架构视图来描述系统的不同方面。

为了尽可能准确地描述软件，本文的结构基于Philippe Kruchten的“4+1”架构模型视图[Kruchten]。



“4+1”视图模型允许不同的涉众在软件架构中找到他们需要的东西。

## 范围

本sad的范围是解释教学综合管理系统的体系结构。

本文档描述了教学综合管理系统设计的各个方面，这些方面被认为在架构上很重要。这些要素和行为对于指导教学综合管理系统的构建和对整个项目的理解是至关重要的。鼓励需要对教学综合系统有技术了解的利益相关者首先阅读为该系统开发的软件需求规格说明书[SRS]。

## 定义、同义词与缩写

* **Apache** – Web Server
* **ASP.NET** - Microsoft web platform
* **HTTP** – Hypertext Transfer Protocol
* [**Mono**](http://www.mono-project.com/ASP.NET)– open source implementation of Microsoft’s Common Language Infrastructure
* **WWW** – World Wide Web
* **SAD -** Software Architecture Document
* **UML** – Unified Modeling Language
* **User -** This is any user who is registered on the website

## 参考文献

[SRS]: Software Requirements Specification

[MedBiquitous]: Sample SAD, <http://medbiq.org/std_specs/techguidelines/softwarearchitecture.pdf>

[Kruchten]: The “4+1” view model of software architecture, Philippe Kruchten, November 1995, <http://www3.software.ibm.com/ibmdl/pub/software/rational/web/whitepapers/2003/Pbk4p1.pdf>

# 视图含义

用于记录本系统的视图有:

用例视图

受众:系统的所有涉众，包括最终用户。

描述一组场景或用例，它们代表系统的一些重要的、中心的功能。

描述系统的参与者和用例，此视图表示用户的需求，并在设计级别进一步详细说明，以更详细地描述离散流和约束。

相关工件：用例模型，用例文档

逻辑视图

观众:架构师。

领域:功能需求:描述设计的对象模型。

还描述了系统最重要的用例实现和业务需求。

相关工件：设计模型

# 系统结构目标和条件

有一些关键的需求和系统约束对体系结构有重要的影响，如下：

1.该系统是一个概念的证明，未来将建立一个更完整的项目预测系统。因此，本文档和整个系统中的主要涉众之一是未来的架构师和设计人员，而不一定是通常情况下的用户。因此，本文档的一个目标是对未来的架构师和设计师有用。

2.系统必须与多个第三方api、程序和谷歌浏览器进行通信。定义系统如何与这些第三方系统接口是体系结构的主要关注点。

3.软件需求规范概述了随着时间的推移，应用程序可能面临的一些预期的变化。系统体系结构的主要目标之一是通过最小化实现这些更改所需修改的代码量来最小化这些更改的影响。体系结构试图通过使用模块化和信息隐藏来隔离可能从系统其他部分更改的组件。

# 用例图

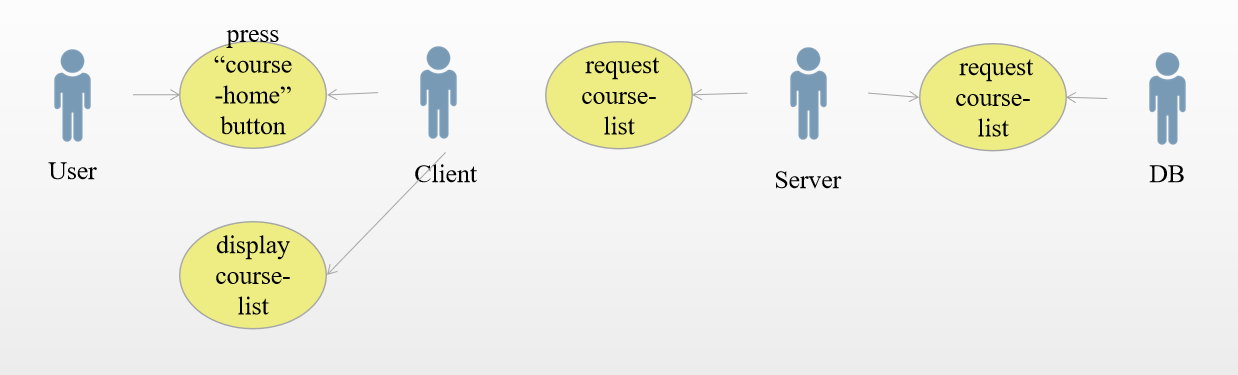
用例视图的目的是为系统的使用及其组件之间的交互提供额外的上下文。在本文档中，每个组件都被视为一个用例参与者。使用UML用例图概述和说明了最常见的用例。

## 用例实现

### 查看课程列表

用户想要查看课程列表，客户端向服务器发送请求，然后将结果在页面上展现出来。

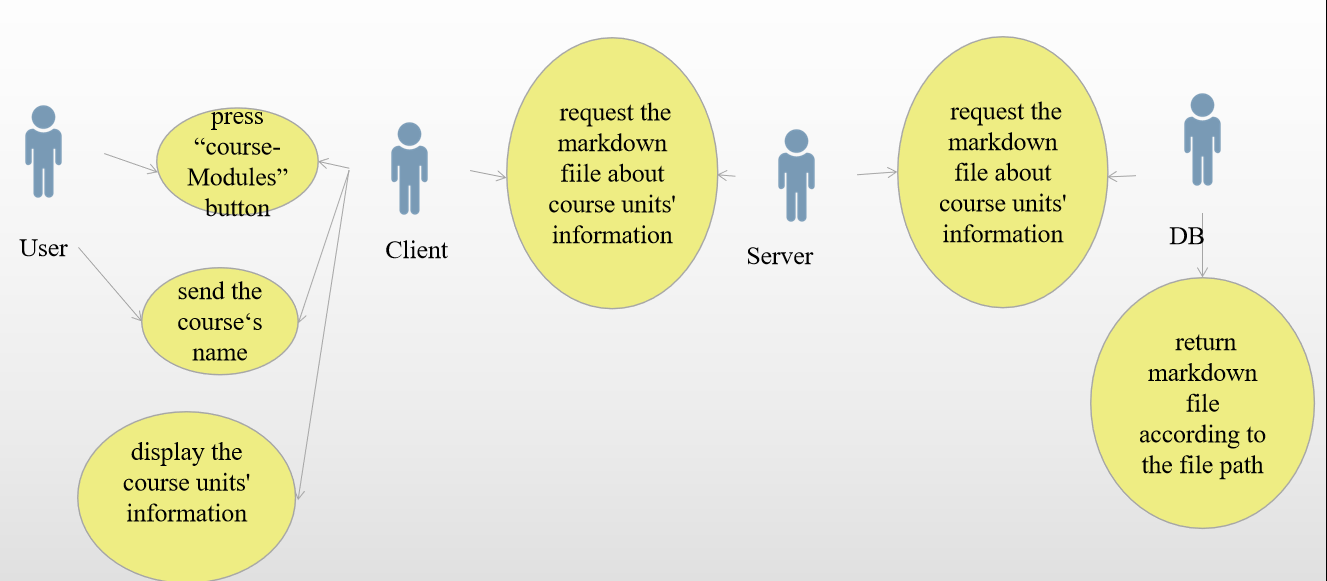
**图 4.1** 查看课程列表的用例图



### 查看单元信息

用户向客户端发送请求，客户端向服务器请求课程的单元信息markdown文件，将结果展现在页面上。

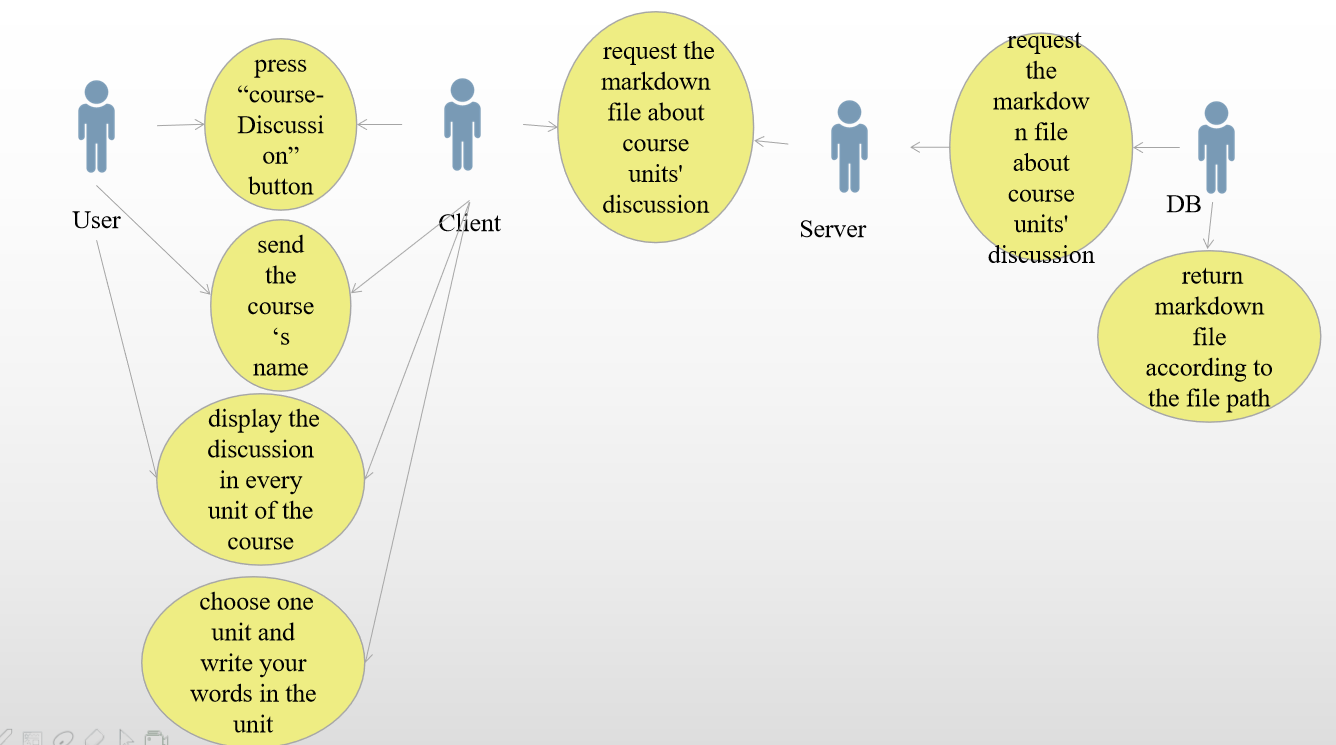
**图 4.2** 查看单元信息的用例图



### 讨论

用户请求查看课程单元的讨论、选择一个单元并在这个单元里写下自己的话。

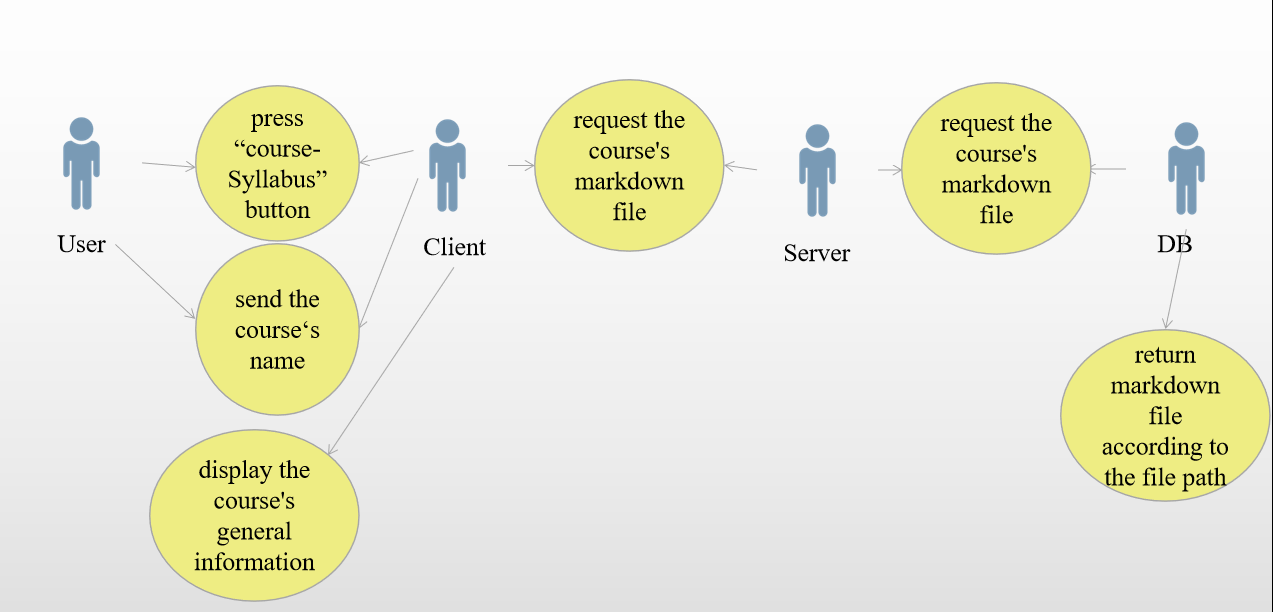
**图 4.3** 讨论用例图



### 查看课程概要

用户向客户端发送请求，客户端根据用户选择的课程名从数据库里调用课程的markdown文件在页面上进行呈现。

**图 4.4** 查看课程概要的用例图

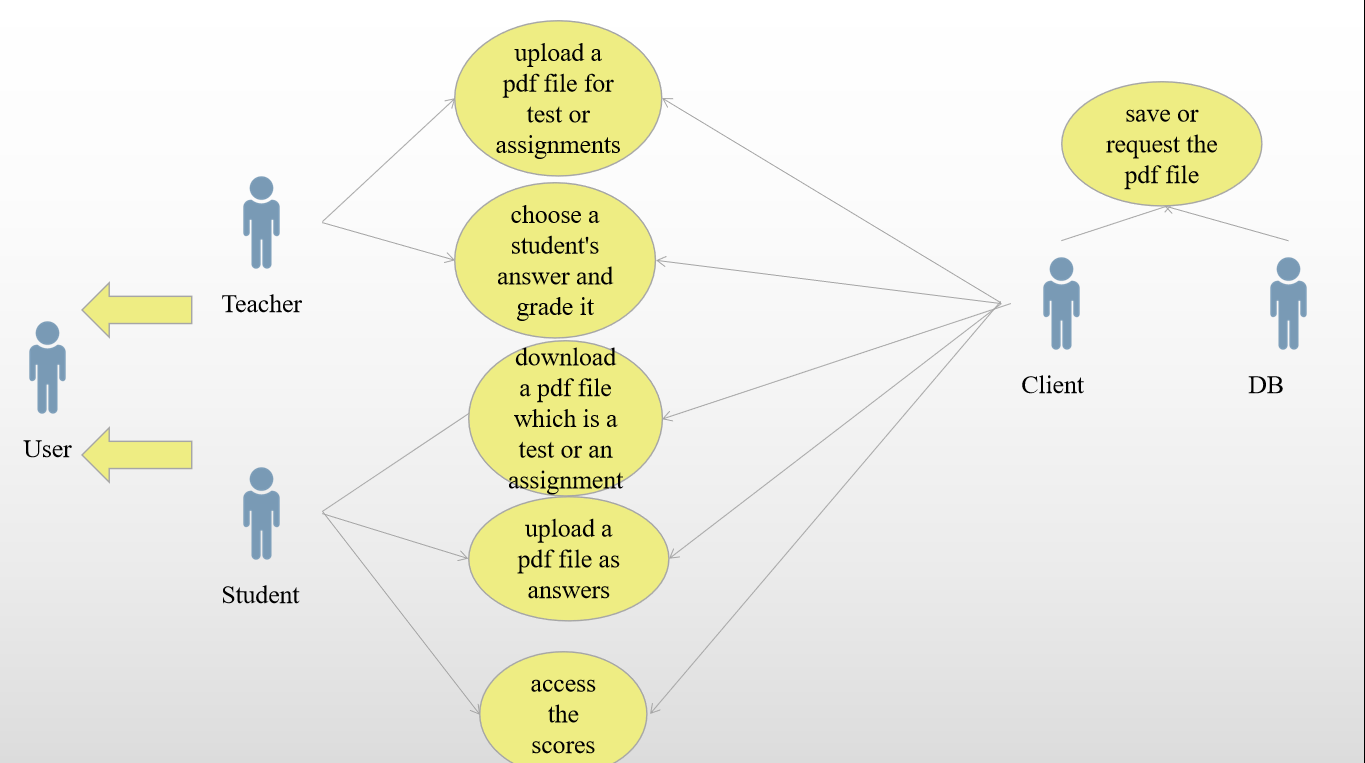


### 评分

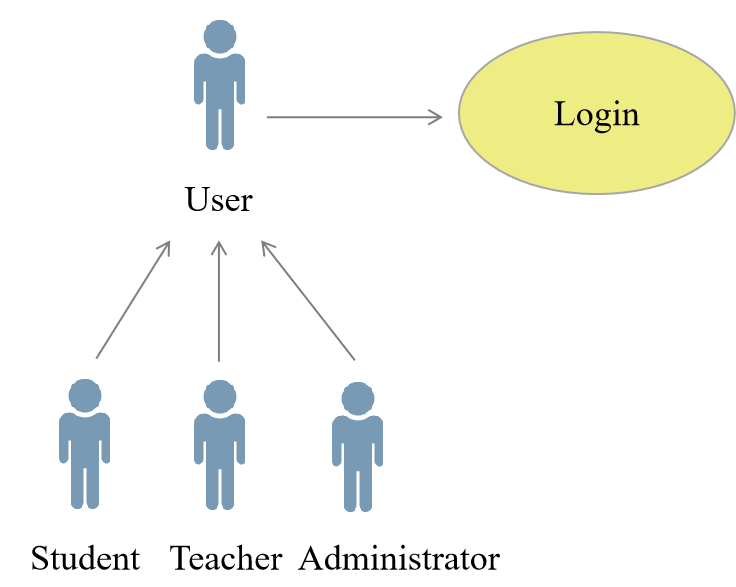
老师上传一个pdf文件作为测试或者作业。然后客户端就会保存这个pdf文件。老师可以从客户端上选择一个学生的答案进行下载并进行批改。

学生下载一个作为测验或者作业的pdf文件，客户端根据请求，从数据库里下载下来，再给学生作为应答。学生还可以上传pdf文件作为答案、可以查看分数。

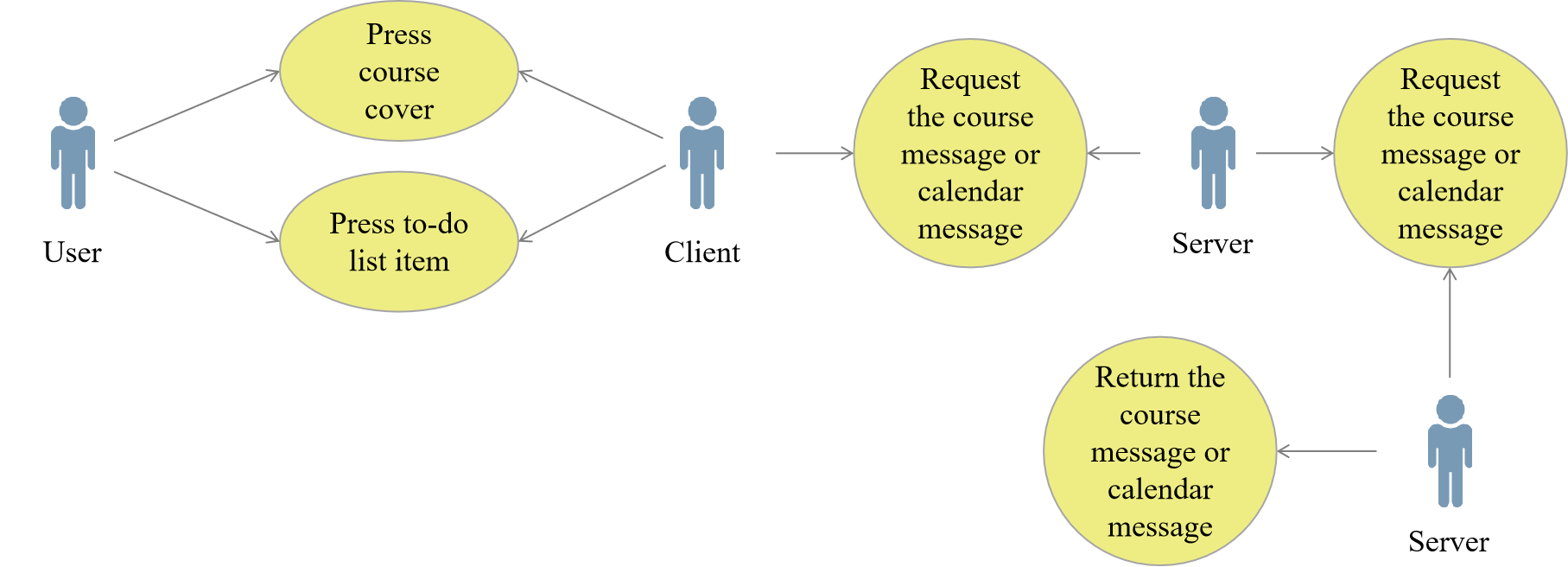
**图 4.5** 评分用例图



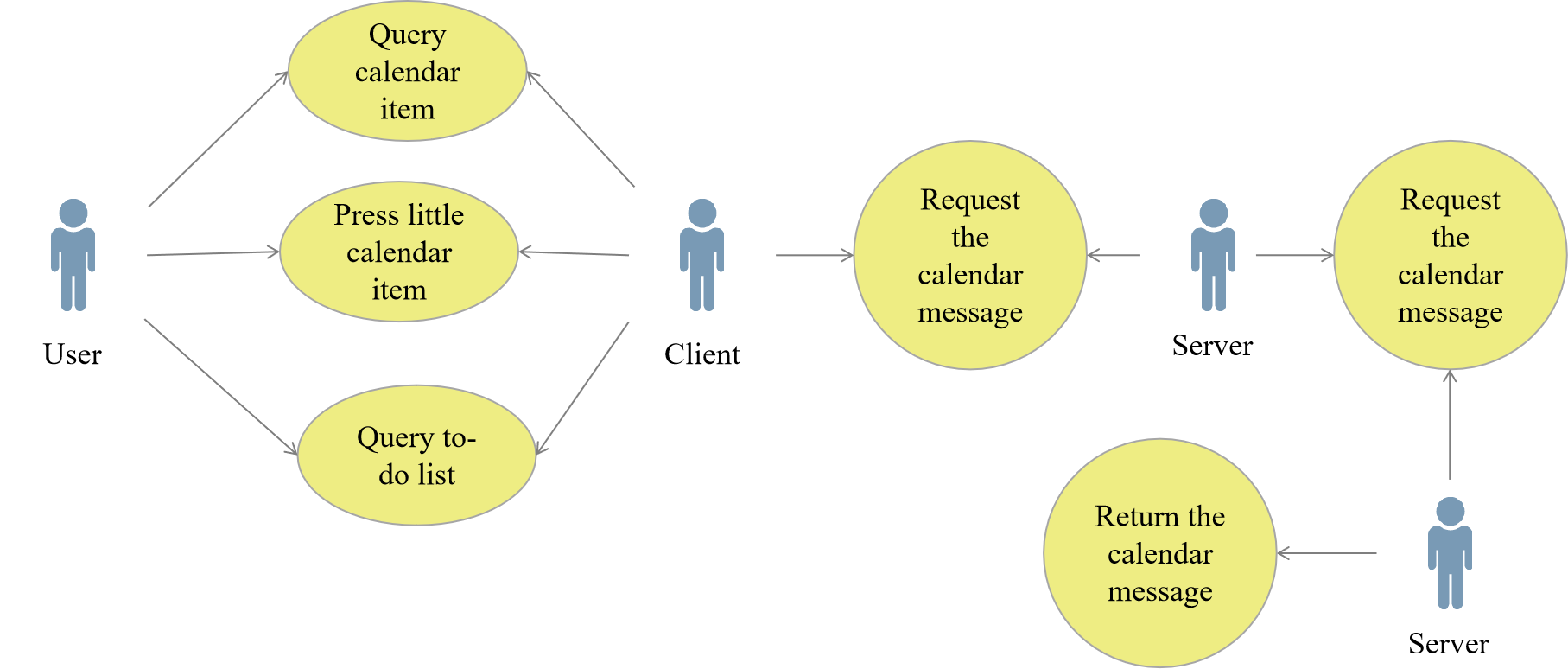
### 登录



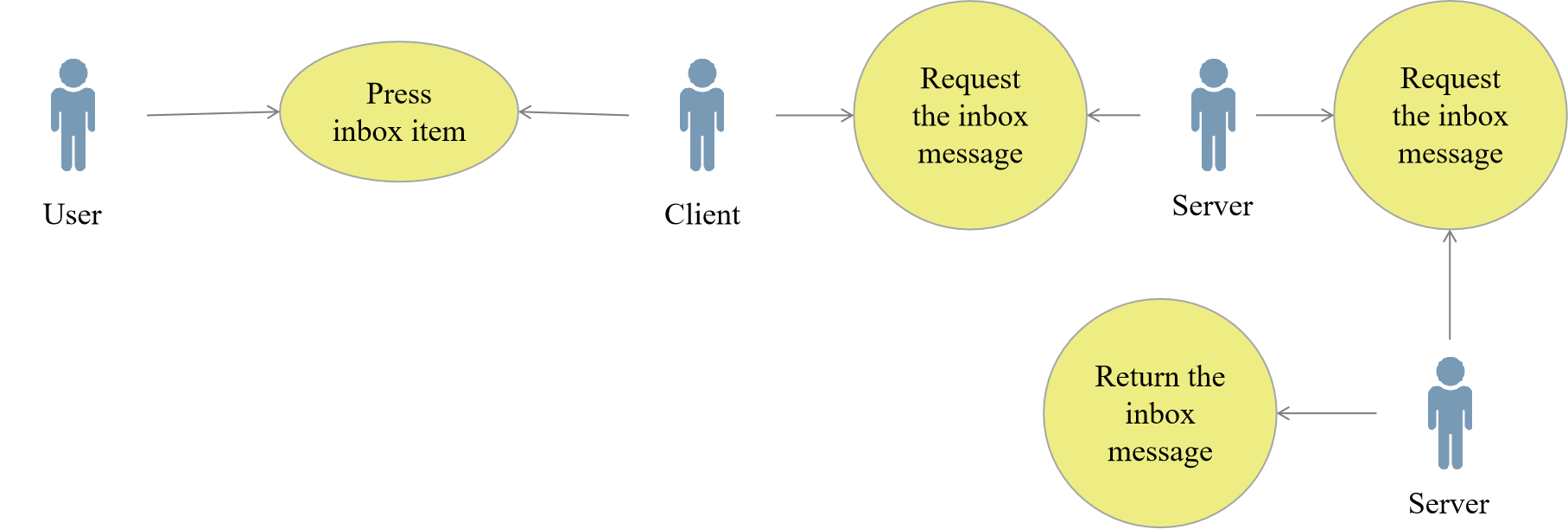
### 主页



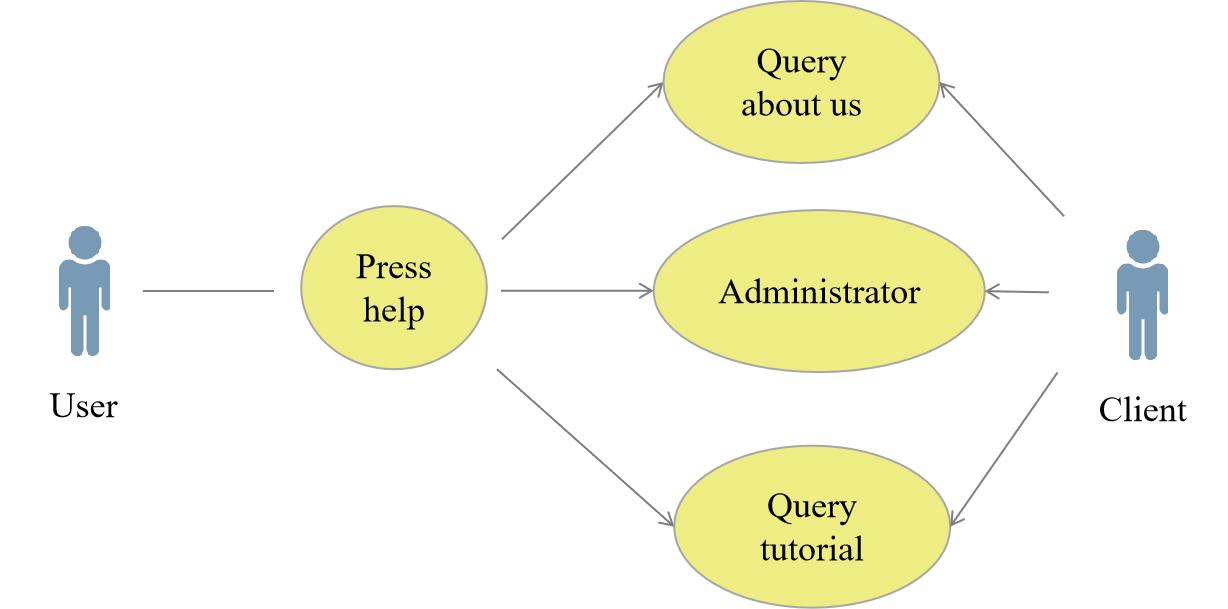
### 日历



### 站内消息



### 帮助



# 逻辑设计

## 总览

登录函数

Function login()

描述: 此函数用于用户登录 服务器接收用户输入的用户名和密码并进行检查

前提条件: 用户在浏览器上输入了网页的网址，并输入了用户和密码并点击了提交

后续事件: 若用户和密码配套，则跳转至用户页面，否则停在登录界面

浏览主页函数

Function dashboard()

描述: 界面跳转至用户主页

前提条件:用户已成功登录或返回至主页

后续事件:进行用户其它相关操作

添加课程

Function add\_course()

描述:管理员用户添加课程

前提条件:管理员在dashboard界面中点击了添加课程按钮

后续事件:设置课程的认可老师和名称

编辑课程 edit\_course\_info()

描述: 老师用户编辑课程信息

前提条件:前提条件:用户点击了某个课程的图标

后续事件:老师对课程信息进行编辑并上传

查看课程信息函数

Function course\_info()

描述:为学生用户提供特定课程的信息 (从数据库中读取)

前提条件:用户点击了某个课程的图标

后续事件:界面显示课程的相关信息

查看个人日历

function calendar()

描述:为用户显示个人的教学日历

前提条件:用户点击了日历图标

后续事件:页面显示用户的个人日历和总课表

留言板功能

Function discussion()

描述:为用户显示留言板内容

前提条件:用户点击了留言板按钮

后续事件:用户进行其他操作或写留言并提交

查看考试和作业列表

Function student\_quiz\_list()

描述:为用户显示个人的考试列表(从数据库中读取)

前提条件:用户点击了测试图标

后续事件:用户点击其它功能 或者点击考试和作业列表中的一项

进行考试和作业

Function quiz()

描述:为学生用户进行测试 显示单选题，主观题和主观题的文本框

前提条件:学生点击了作业和测试列表中的一项,且此项目未被提交过

后续事件：学生进行答题，并点击提交上传答案

自动批改

Function auto\_check()

描述:对学生提交的客观题进行批改

前提条件:学生在一个测试或作业项目中点击了提交按钮

后续事件:对学生的答案进行批改 并对数据库中数据进行更新

查看成绩

Function view\_grade()

描述:为学生用户显示测试和作业得分

前提条件:学生点击了作业和测试列表中的一项,且此项目被批改完

后续事件：学生进行其他操作

老师出题

Function teacher\_quiz\_list()

描述:老师用户查看或添加测试 屏幕显示测试列表

前提条件:老师点击了作业和测试按钮

后续事件:进行其他操作或点击添加测试,或点击已存在的测试,进行主观题批改

添加测试

Function add\_quiz()

描述:老师设置测试

前提条件:老师点击了添加测试按钮

后续事件:老师点击添加试题，或者提交测试

添加题目

Function add\_question()

描述:老师添加题目 对题目进行设置

前提条件:老师点击了添加题目的按键

后续事件:老师对题目进行设置并点击提交按钮

老师评分

Function teacher\_check()

描述:老师对主观题进行评分

前提条件:老师点击了测评列表中的一项

后续事件:老师对题目进行评分并点击上传

添加用户

Function add\_user()

描述：管理员添加学生或者老师用户

前提条件:管理员点击添加用户按钮

后续事件:管理员输入账户和密码和类型，并提交

## 5.2总览

### 5.2.1数据库接口

// login

{

"uid" 用户id

"psw" 密码

"user\_type" 账户类型

}

// course

{

"href" 课程链接

"src" 图片链接

"title" 课程标题

"text" 课程一句话简介

}

// todo

{

"text" 内容

"due" 截止日期

}

// coursehome

{

"href" md链接

}

// syllabus

{

"href" md链接

}

// models

{

"cid" 课程id

"title" 单元标题

"text" 单元内容

}

// discussion

{

"cid" 课程id

"uid" 用户id

"text" 对话内容

"href" 表情链接

}

// quiz

{

"cid" 课程id

"mid" 模块id

}

// quiz

{

"title" 选择题干

"choiceA" 选项A

"choiceB" 选项B

"choiceC" 选项C

"choiceD" 选项D

"correct" 正确选项

}

// calendar

{

"uid" 用户id

"date" 事件日期

"title" 事件标题

"text" 时间内容

}

登录

int status；

function login(string lid,string pwd){

return status;

}

status 0 为登录成功，1为账户不存在，2为密码错误

查看课程信息

function course\_info(int cid){

Return class;

}

查看个人日历

Struct calendar\_ouput{

int num;//该周有多少节课

class[num];

}

function calendar(int lid,int week){

return calendar\_ouput；

}

查看考试和作业列表

Struct quiz\_ouput{

int num;//有多少个考试

test[num];

}

function student\_quiz\_list(int lid)

return quiz\_ouput;

}

查看成绩

Function view\_grade (int cid){

Return class;

}

添加考试

Int status;

Function add\_quiz(int lid,string tname,int deadline,int total\_score){

Return status;

}

添加题目

int status；

function add\_question(int lid,int tid){

return status;

}