实验室管理系统

王子轩 王一帆 刘佳明 招丽莹 李若璇 马润琪

Web项目建议书

侧重业务建议，所属领域、目标、解决核心问题、带来效益等方面

Web项目建议书

- **项目背景**
- 实验室作为实践教学中的重要手段，在教学中扮演了重要的角色，实验室的课程和成绩在老师与学生中的对接也是需要解决的难题，对西安电子科技大学日益增多的实验教学需求，古老的人工管理方式和人工对接方式已显得力不从心，因此提出了更加简便、清晰、规范的实验室管理系统的需求。

- **术语定义**
- Spring Boot：Java平台开源应用框架
- JDBC：Java数据库连接
- MySQL：关系型数据库管理系统
- HTML：超文本标记语言

Web项目建议书

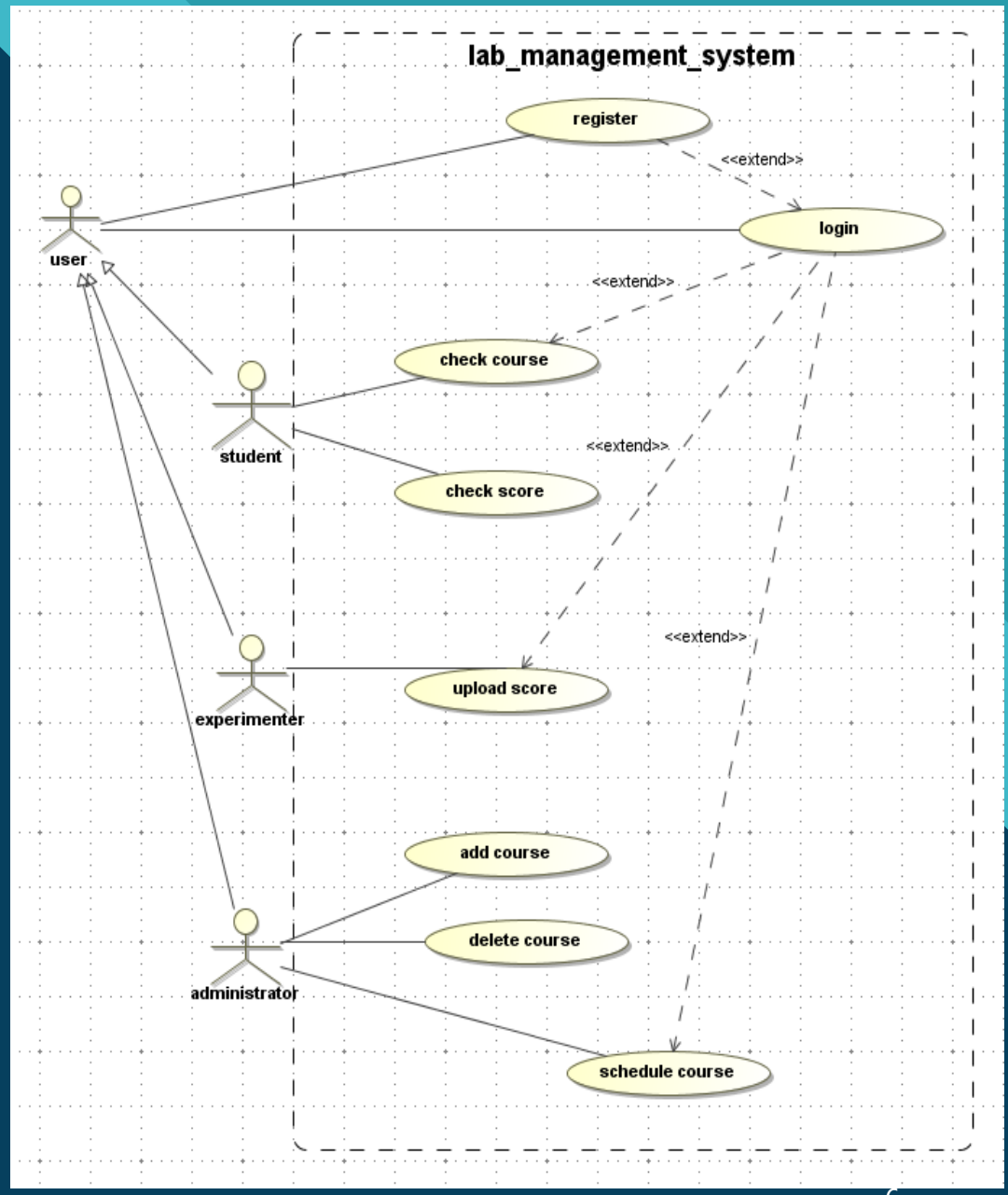
- **系统概述**
- 实验室管理系统为实验室的管理提供快捷方便的服务，并且集数据查询、统计为一体。它包括了实验课程的管理、实验成绩的录入、给某个班级安排课程、课程查询、课程成绩查询等。
- 通过使用实验室管理系统实现高校实验室、与在校师生之间管理的规范化、信息化；提高实验教学特别是开放实验教学的管理水平与服务水平；为实验室评估、实验室建设及实验教学质量管理等决策提供数据支持；协助高校轻而易举完成数据上报工作。运用计算机技术，特别是现代网络技术，为实验室管理、实验教学管理、实验室评估与评教等相关事务进行网络化的规范管理。

Web项目需求与应用建模

项目需求规格说明书

Web项目需求与应用建模

- **系统角色：**
- 本系统主要用于以下的几类人员：（实验室管理系统-示例）
- 管理员，管理课程（包括增删课程）和课程安排。
- 实验员，录入自己负责的学生的成绩。
- 学生，查询自己的课程，查找自己的成绩（如果未录入就查不到）。



Web项目需求与应用建模

- **运行环境**
- （1）客户端：
- 操作系统：Windows
- 浏览器：Chrome等
- （2）应用服务器端：
- 操作系统：Ubuntu
- 应用服务器：Ubuntu
- 数据库访问：JDBC
- （3）数据库服务器端：
- 操作系统：ubuntu
- 数据库系统：ubuntu

- **假设与依赖**
- 用户浏览器内核版本不兼容，提示用户使用其他浏览器；
- 用户输入密码错误，提供找回密码链接；
- 用户电脑未连接上网络，不能进入网站 。

Web项目需求与应用建模

- **具体功能**
- 在实验室管理系统中，主要有用户登陆、学生课程查询、学生成绩查询、实验员录入成绩、课程信息管理、课程安排等功能，功能分析如下:
- （1）用户登陆
- 网站采用邮箱及密码验证模式，进入实验室管理网站前，用户必须在登陆页面输入邮箱及密码，只有验证通过的用户方可进入实验室管理网站操作主页面，登录账号会自动识别身份，有对应的权限可操作。
- （2）学生课程管理
- 学生登录网站后可以查询自己班级所安排的课程。
- （3）学生成绩查询
- 学生登录网站后可以查询自己课程的成绩。
- （4）录入成绩
- 实验员登录网站后，可以为相应课程录入每个学生的成绩。
- （5）课程信息管理
- 包括两个功能模块：添加课程、删除课程。
- 管理员登录网站后，可以添加课程，也可以对课程进行删除。
- （6）课程安排
- 管理员登录网站后，可以选择系统中班级安排相应的课程。

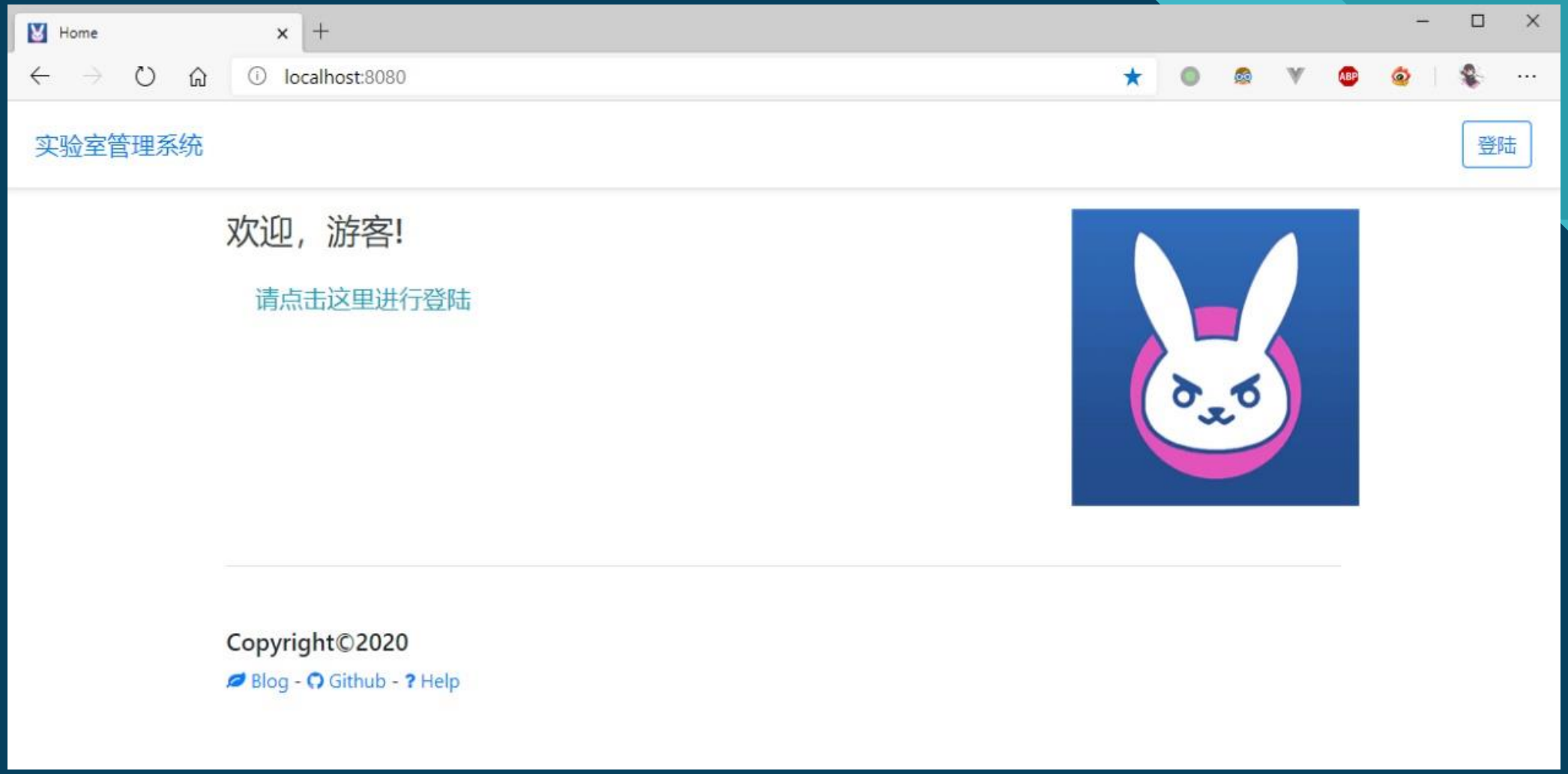
Web应用设计

界面样式、布局、导航设计、用户交互

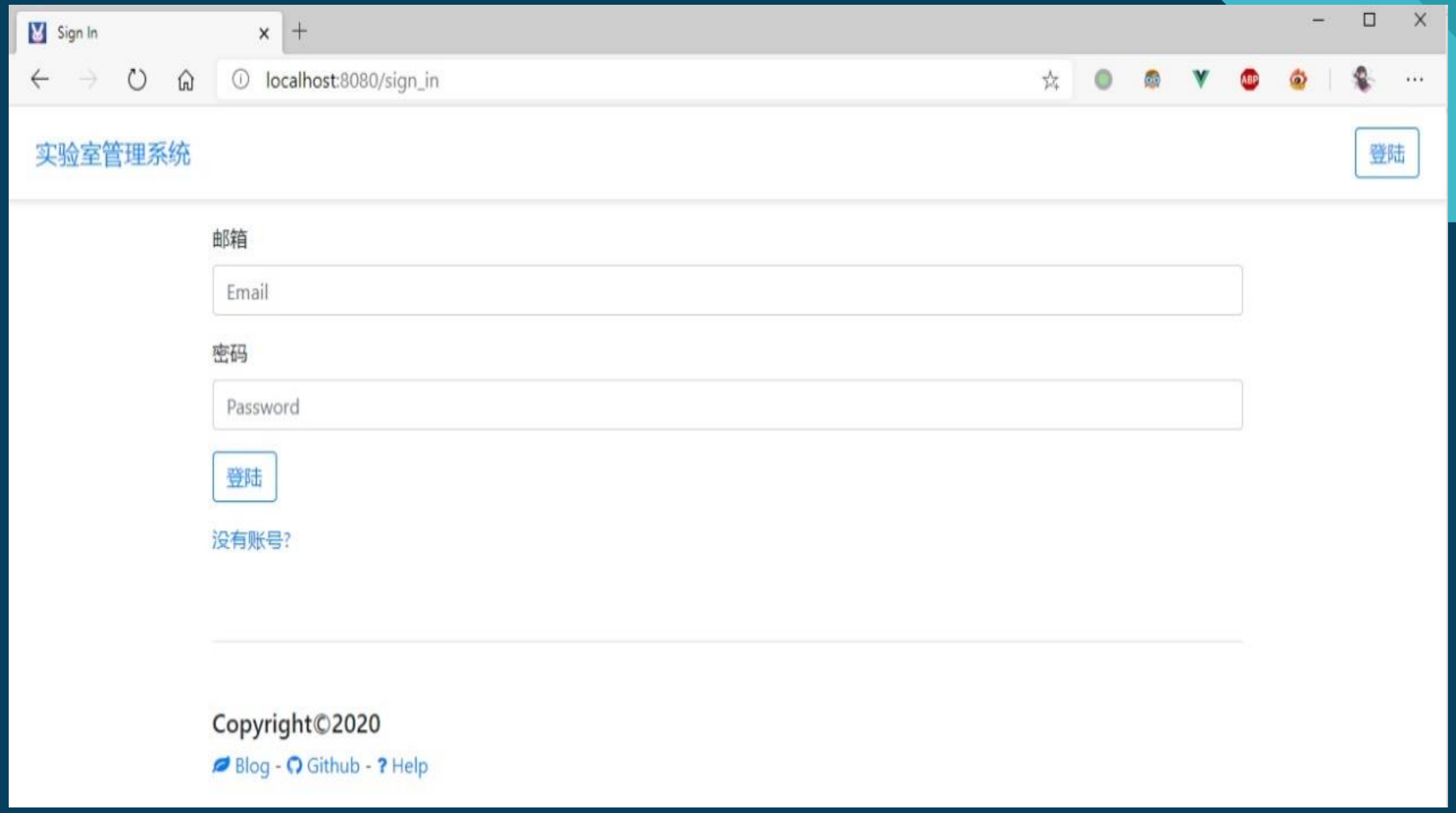
Web应用设计

- **用户界面**
- 本系统所有界面使用WEB界面；
- 用户界面的具体细节将在概要设计文档中描述。
- 文字采用黑色等线
- 页面框架为上下结构
- 屏幕分辨率应为1440x900；
- 颜色以白色和蓝色为主。

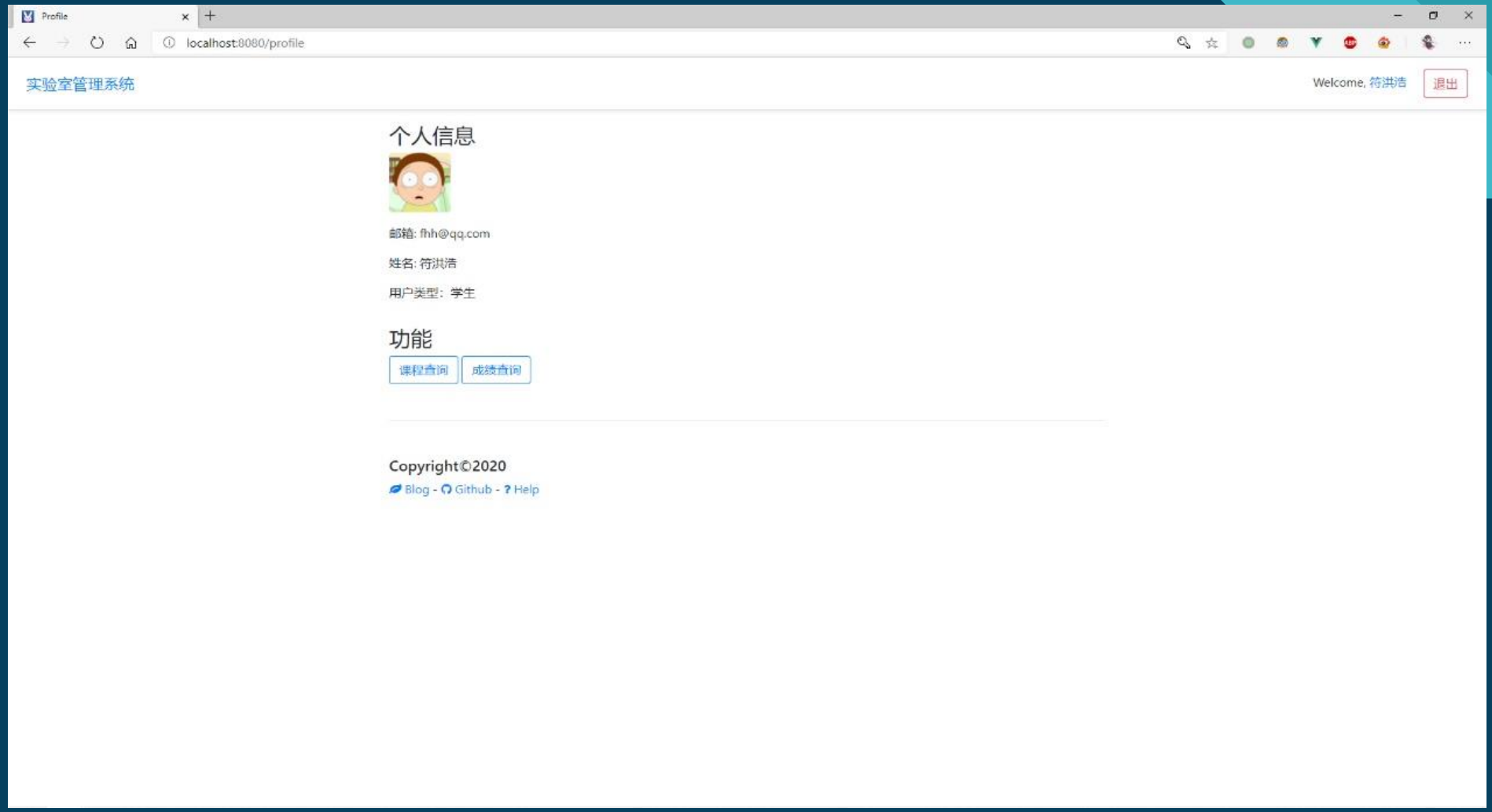
Web应用设计：主界面


Home
localhost:8080
实验室管理系统
登陆
欢迎，游客!
请点击这里进行登陆
Copyright©2020
Blog - Github - ? Help

Web应用设计：登录页面



Web应用设计：登录界面（不同类型的用户有不同的展示）



Web应用测试

- 测试的需求分析：

实验室管理系统为实验室的管理提供快捷方便的服务，并且集数据查询、统计为一体。它包括了实验课程的管理、实验成绩的录入、给某个班级安排课程、课程查询、课程成绩查询等。

整个系统的使用用户主要是学生与教师，整体的使用界面较为简洁明确

- 选用的测试方法和策略：

侧重于功能测试，兼容性测试，性能测试，安全性测试

以下是对于各部分的分析过程，及其测试后的结果分析

功能测试：

- 链接测试：

首先，所有的链接是否按指示的那样，确实链接到了该链接的指向的资源；其次，所链接的资源是否存在；最后是否所有的页面都能够被链接到，不存在鼓励页面。

我们对整个项目所涉及到的若干页面进行了统计。整个系统用户分为三类，学生，实验员和管理员。因此，利用三种账号进行页面的跳转检查，并未发现有未被覆盖的页面。

- 交互测试：

在链接测试中，不可避免的涉及到功能的交互。例如在学生账号角度，我们需要有加入课程，检查课程，以及最后的成绩查询等功能。在模拟整个系统运行的过程中，依次，重复的尝试使用了所有功能。并未检查出有功能上的缺失。每一项功能，都能在落实后，给予操作者直观反馈（指能有具体信息的展示）

- Cookies测试：

Cookies通常用来存储用户信息和用户在web应用中的操作。

（留白：检查cookies是否能正常工作。测试内容可包括cookies是否起作用，是否按预定的时间进行保存，刷新web页面对cookies是否有影响。如果保存了注册信息，请确认该cookies能正常工作，并有信息加密）

功能测试：

- 数据库测试：

数据库测试包括测试实际内容，及其完整性，以确保数据没有损坏且模式正常。

在各类账户下，使用了若干涉及数据库的功能，例如新建新的课程，添加学生成绩，修改学生成绩等。测试过程中，特意使用了较多的错误输入。对于这些错误输入，也都被拒绝参与数据库中数据的记录。

- 特定功能测试：

在这里就是特指各账户的功能测试，尝试三种用户可能进行的所有操作，这在前面的若干测试中也已被涉及。所有功能均能够实现，并达到预期效果。

兼容性测试：

- 平台测试：

Web应用的运行环境的可变性和不稳定性，是web工程面临挑战的重要因素。由于用户群的不同，所有客户端的硬件设备、网络连接、操作系统、服务端支持、浏览器等因素都有所不同。

因此，我们这里尝试使用了不同操作系统平台来运行我们的项目，具体有。发现都能正常运行，并且很好的兼容于此类操作系统。

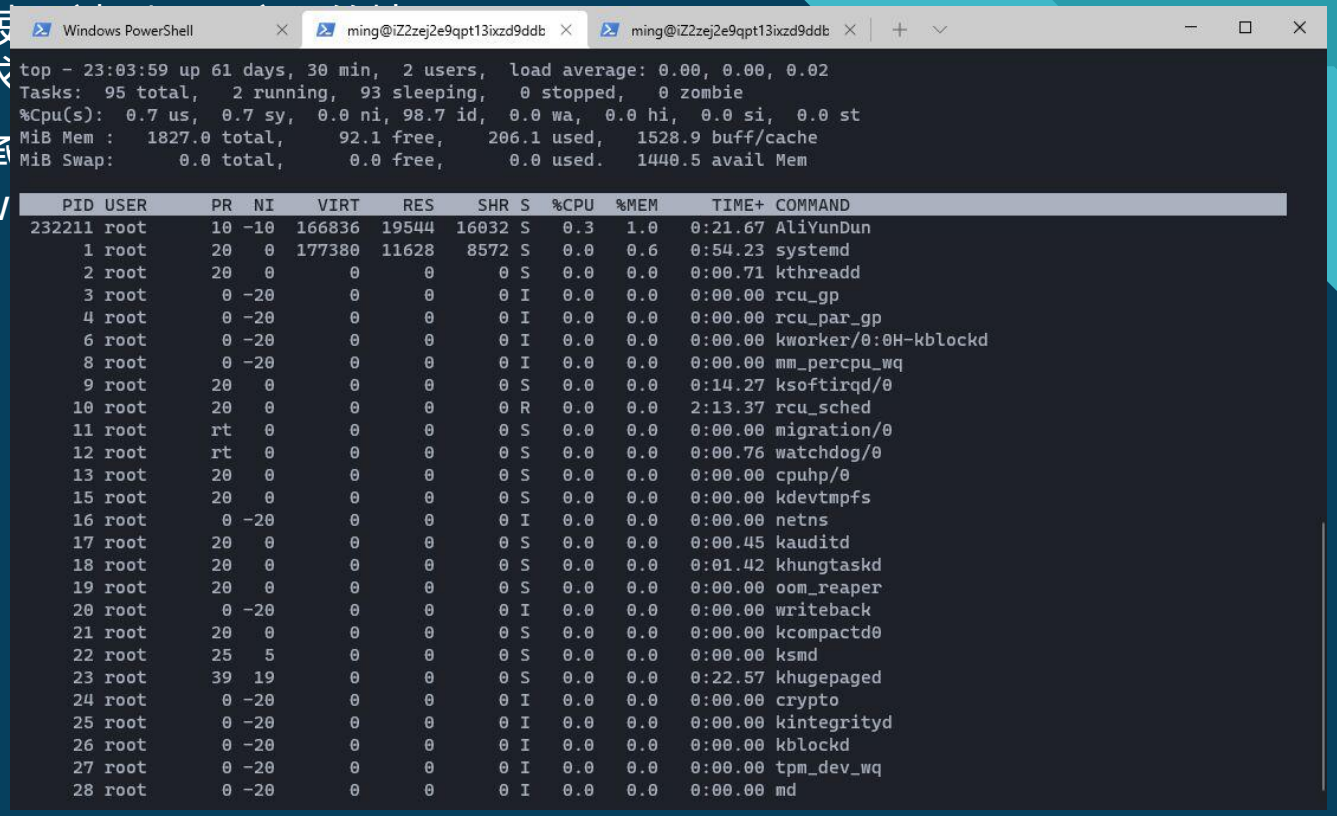
另一方面，我们也尝试了不同浏览器的测试。浏览器包括，google chrome,Microsoft Edge,Internet Explorer 等。在各类浏览器下，对于同一个页面的展现或者行为都完全一致。

（留白：是不是可以贴上两张不同浏览器的同一个页面的展示对比）

另外，我们也换用了不同分辨率，用以检查是否分辨率会影响页面整体的展示效果。

性能测试：

- Web应用性能测试用于评估web应用的响应时间，及可靠性如何受增长的用户数量和通信量以及功能复杂性的影响，找出与性能有关的web应用组件和特征，并确定性能下降如何影响web应用的整体需求和目标。Web应用性能测试是指在正常和疲劳驾驶使
况下，观察该web应用的性能是否满足性能要求
- 具体指标包括：用户可接受的响应时间，能够承
够同时处理的业务数目，以及不同负载情况下w
查瓶颈可能发生的位置。



安全性测试：

- 用户身份验证：

Web应用中，用户身份验证直接关系到用户权利和隐私的安全。

因此对于用户身份验证测试就尤为重要。

在我们的测试中主要针对：

1、无效用户名和密码登录过程

2、大小写敏感

- Session测试：

该项主要检查了web应用是否有超时的限制，包括但不限于 例如一段时间内没有任何操作，是否需要重新登陆才能正常使用 等情况

- 另外还涉及了sql的相关内容，这在后续的web应用的安全性中会再讲到

Web应用运维：

- Seo：seo是一种搜索引擎营销方式，通过优化web应用结构，web页面代码，和内容等方式，帮助搜索引擎的搜索程序找到含有最佳内容的web页面，提高其在搜索结果中的自然排名。

- 我们项目中所采用的seo策略主要包括两点：
- js的处理，以及页面性能的优化

Web应用维护

- **js处理：**
- 1. 尽量放到页面尾部
- Js的加载是阻塞页面的，没下载完毕后面的内容不会出来，所以尽量避免把JS放到页面头部，按照经验估计，整个页面中所用的JS逻辑，90%都是可以放到页面尾部。
- 2. 延迟加载（按需加载）
- 很多的业务逻辑并非每次都使用也不是要立即使用，首次加载过程中仅仅加载那些必须的，只有当必要的条件触发，才去加载请求必要的JS.比如说权限验证通过，加载管理模块。点击发表文章按钮，加载与发表文章有关的验证和处理函数。
- 如果写过C++的肯定会接触过动态库和静态库，这个与之类似，什么时候需要什么时候再加载，首次打开页面肯定会清净了许多，而且业务逻辑也由此分离开来，管理和维护也会方便很多，毕竟减少了那么多的耦合。
- 按照BBS项目经验估计，普通页面的所有业务逻辑中需要在首次请求中加载的不到50%，我们的JS又由此砍掉了一半。

Web应用维护：

- 3. 合并JS，减少请求
- 请求多个小文件的效率远小于请求一个大文件的效率，因为需要多次DNS解析，多次连接，浏览器和server端也需要进行多次开启进程、权限验证和预处理，以及 http请求在数据包传递上的一些问题。
- 所以尽量避免在页面中加载一堆的js 文件，需要先讲需要的小的JS合并成一个大的JS文件统一输出，页面因此被卡住的时间肯定会减少很多。
- 为了提高开发效率，合并建议不要每次都手动来进行，导致之后维护成本很大，相信些个XML配置文件，确定合并规则以及依赖关系后，用程序自动合并效率会高很多

- web页面性能优化：
- 对一个网站而言，CSS、javascript、logo、图标这些静态资源文件更新的频率都比较低，而这些文件又几乎是每次http请求都需要的，如果将这些文件缓存在浏览器中，可以极好的改善性能。通过设置http头中的cache-control和expires的属性，可设定浏览器缓存，缓存时间可以是数天，甚至是几个月。

- 在某些时候，静态资源文件变化需要及时应用到客户端浏览器，这种情况，可通过改变文件名实现，即更新javascript文件并不是更新javascript文件内容，而是生成一个新的JS文件并更新HTML文件中的引用。
- 使用浏览器缓存策略的网站在更新静态资源时，应采用逐量更新的方法，比如需要更新10个图标文件，不宜把10个文件一次全部更新，而是应该一个文件一个文件逐步更新，并有一定的间隔时间，以免用户浏览器忽然大量缓存失效，集中更新缓存，造成服务器负载骤增、网络堵塞的情况。

Web应用性能和可用性分析与调优：

- 对于数据库优化：
- 对于使用数据库的web应用来说，优化数据库能够大大地提高web应用的性能，往往一些性能问题，归根结底都是不合理的数据库表结构设计以及缺乏对于数据库内部构造的了解所导致的。
- 因此我们对于数据库的相关表重新进行了设计，包括不限于基于范式的表的建立，建立索引，关键字，视图，触发器等方式。

- 优化应用程序：特指sql语句优化

因为有对表的重新设计，基于新增加的视图等，因此我们也更进一步修改了若干sql语句，使其更为简洁方便。

Web应用性能和可用性分析与调优：

- 可用性：可用性是web应用在特定使用环境中为特定目标用户所使用的，从而快速、有效、满意地完成特定任务的程度。高可用性的用户界面使用户全神贯注于正在进行的工作，不用花很多心思考虑如何使用该web应用。
- 我们的项目就是基于实验室的管理系统，面向对象是教师，学生以及实验室的管理员。简单清晰的用户划分导致了，我们不需要繁杂花哨的界面展示，我们更需要的是展示并提供各类用户所需要的内容。这恰恰是web可用性的根源所在。
- 在后续的优化中，我们大大削减了页面的模块内容。在我们的页面上，可以清晰的识别出导航，标题，等用户需要的功能。

Web应用的安全性：

- 安全性介绍：
- Web应用安全性是指一种能够识别和消除网络上对web应用的硬件、软件、数据等造成破坏的不安全因素的能力，凡是涉及网络上的信息保密性、完整性、可用性、真实性和可控性的相关技术，都是web应用安全的研究领域。

- 在我们的项目中，我们着重关注了一下两点：
- Sql注入
- 数据加密传输：

Sql注入：

- Sq注入l概念介绍：
- SQL注入即是指web应用程序对用户输入数据的合法性没有判断或过滤不严，攻击者可以在web应用程序中事先定义好的查询语句的结尾上添加额外的SQL语句，在管理员不知情的情况下实现非法操作，以此来实现欺骗数据库服务器执行非授权的任意查询，从而进一步得到相应的数据信息。
- 在我们的项目中，我们针对于登录界面，进行了防范sql注入的若干尝试。

Sql注入

- 1、 普通用户与系统管理员用户的权限要有严格的区分。
- 如果一个普通用户在使用查询语句中嵌入另一个Drop Table语句，那么是否允许执行呢?由于Drop语句关系到数据库的基本对象，故要操作这个语句用户必须有相关的权限。在权限设计中，对于终端用户，即应用软件的使用者，没有必要给他们数据库对象的建立、删除等权限。那么即使在他们使用SQL语句中带有嵌入式的恶意代码，由于其用户权限的限制，这些代码也将无法被执行。故应用程序在设计的时候，最好把系统管理员的用户与普通用户区分开来。如此可以最大限度的减少注入式攻击对数据库带来的危害。
- 2、 强迫使用参数化语句。
- 如果在编写SQL语句的时候，用户输入的变量不是直接嵌入到SQL语句。而是通过参数来传递这个变量的话，那么就可以有效的防治SQL注入式攻击。也就是说，用户的输入绝对不能够直接被嵌入到SQL语句中。与此相反，用户的输入的内容必须进行过滤，或者使用参数化的语句来传递用户输入的变量。参数化的语句使用参数而不是将用户输入变量嵌入到SQL语句中。采用这种措施，可以杜绝大部分的SQL注入式攻击。不过可惜的是，现在支持参数化语句的数据库引擎并不多。不过数据库工程师在开发产品的时候要尽量采用参数化语句。

Sql注入：

- 3、 加强对用户输入的验证。

- 总体来说，防治SQL注入式攻击可以采用两种方法，一是加强对用户输入内容的检查与验证;二是强迫使用参数化语句来传递用户输入的内容。在SQLServer数据库中，有比较多的用户输入内容验证工具，可以帮助管理员来对付SQL注入式攻击。测试字符串变量的内容，只接受所需的值。拒绝包含二进制数据、转义序列和注释字符的输入内容。这有助于防止脚本注入，防止某些缓冲区溢出攻击。测试用户输入内容的大小和数据类型，强制执行适当的限制与转换。这即有助于防止有意造成的缓冲区溢出，对于防治注入式攻击有比较明显的效果。
- 如可以使用存储过程来验证用户的输入。利用存储过程可以实现对用户输入变量的过滤，如拒绝一些特殊的符号。如以上那个恶意代码中，只要存储过程把那个分号过滤掉，那么这个恶意代码也就没有用武之地了。在执行SQL语句之前，可以通过数据库的存储过程，来拒绝接纳一些特殊的符号。在不影响数据库应用的前提下，应该让数据库拒绝包含以下字符的输入。如分号分隔符，它是SQL注入式攻击的主要帮凶。如注释分隔符。注释只有在数据设计的时候用的到。一般用户的查询语句中没有必要注释的内容，故可以直接把他拒绝掉，通常情况下这么做不会发生意外损失。把以上这些特殊符号拒绝掉，那么即使在SQL语句中嵌入了恶意代码，他们也将毫无作为。
- 故始终通过测试类型、长度、格式和范围来验证用户输入，过滤用户输入的内容。这是防止SQL注入式攻击的常见并且行之有效的措施。

Sql注入：

- 4、设置陷阱账号：

- 设置两个帐号，一个是普通管理员帐号，一个是防注入的帐号。将防注入的账号设置的很象管理员，如 admin，以制造假象吸引软件的检测，而密码是大于千字以上的中文字符，迫使软件分析账号的时候进入全负荷状态甚至资源耗尽而死机。

数据加密传输

谢谢