



浙江机电职业技术学院  
ZHEJIANG INSTITUTE OF MECHANICAL & ELECTRICAL ENGINEERING

# 毕业设计

课题名称： 小学生学习英语网站系统前端开发

学生姓名 韩亮亮

学 号 150302153209

所在系部 信息技术系

专 业 计算机应用技术

班 级 计算机 1533

指导教师 林逢升

起讫时间： 2017 年 12 月 25 日～2018 年 04 月 01 日

## 小学生学习英语网站系统前端开发

### 摘 要

针对如今社会学习英语已经不是初中后才开始,逐渐向更年轻的小学发展。我们开发的这款软件主要是为了辅助小学生记忆单词,并培养他们对英语的兴趣。本软件产品的用户主要是刚开始学英语以及有一定基础的小学生,帮助他们学习英语,不仅仅局限于教室,在任何地方,任何时间都可以去学习英语,提高学习的兴趣。

**关键词:** C/S 结构, 小学生学习英语网站系统前端开发, 数据库, 前端框架;

### Abstract

For today's society to learn English is not the beginning of junior high school, and gradually to the younger primary school. We developed this software mainly to help pupils remember words and develop their interest in English. Users of this software product are mainly primary school students who have just started to learn English and have a certain basis to help them learn English. They are not limited to the classroom. They can learn English at any time and improve their interest in learning.

Key words: C / S structure, Front end development of primary school students learning English Web site system, database, frontend framework.

# 目 录

摘 要 .....	II
Abstract .....	II
目 录 .....	i
第 1 章 绪论 .....	1
1.1 引言 .....	1
1.2 项目背景 .....	1
1.3 项目目标及意义 .....	2
1.4 项目所用技术介绍 .....	2
第 2 章 需求分析 .....	3
2.1 可行性分析 .....	3
2.2 需求分析 .....	3
2.2.1 需求分析概述 .....	3
2.2.2 功能分析 .....	4
2.2.3 性能分析 .....	5
2.2.4 开发及运行环境 .....	5
第 3 章 系统设计 .....	6
3.1 功能模块设计 .....	6
3.2 操作流程设计 .....	7
3.3 数据库设计 .....	7
3.3.1 数据库概念设计 .....	7
3.3.2 数据库存储结构 .....	9
3.4 接口设计 .....	15
第 4 章 系统实现 .....	37
4.1 首页 .....	37
4.2 登录 .....	39
4.3 注册 .....	41
4.4 个人中心 .....	43
4.5 学习中心 .....	45
4.6 后台用户管理 .....	46
4.7 数据中心管理 .....	49
4.7.1 年级和单元 .....	49
4.7.2 单词 .....	52
4.7.3 短语 .....	54
4.7.4 语法 .....	55
4.7.5 课文 .....	55
4.7.6 练习 .....	56

4.7.7 听力 .....	56
<b>第5章 系统安装及部署 .....</b>	<b>57</b>
5.1 软件运行环境 .....	57
5.2 安装步骤 .....	57
<b>第6章 结论 .....</b>	<b>60</b>
<b>参 考 文 献 .....</b>	<b>61</b>
<b>致 谢 .....</b>	<b>62</b>

# 第 1 章 绪论

## 1.1 引言

随着 Internet 和网络技术的发展, Web 的应用也广泛起来, 而且 Web 应用以其特殊的特点, 更方便的在生活中运用。对于开发 Web 应用程序进行系统的研究具有很大的意义。想做好一个满意的网站, 是需要运用新的技术, 使用框架来做出优秀的网页。一个令人欣赏的网站, 首先网站要优美, 二要性能稳定, 这就对于开发者来说的考验是大的。在此基础上, 我们不得不继续学习新的技术来补充在开发上缺陷。目前最流行的前端开发框架是 Bootstrap, ElementUI, AntDesign, NG-ZORRO 等等。运用这些框架做出优秀的网站前端是具有很大价值的。

## 1.2 项目背景

结合本项目软件系统的名称为小学生学习英语网站系统前端开发, 该软件在互联网的支持下主要用户为需要学习英语的小学生。英语学习, 一直以来都是大部分人的头疼的问题, 而英语单词, 更是整个英语学习过程中的重要环节, 而且在英语学习的过程中, 不同的人群所学的领域也不仅相同, 而此网站是专门针对小学生来设计的, 用于生动灵活的方式去一步一步的引领学生学习英语单词, 慢慢培养他们学习英语的兴趣。面对当前生活压力和快节奏的生活规律, 学习是个非常快的, 一天就有不同的变化, 面对这样的压力, 从小开始培养孩子学习英语是个非常有必要的计划。更重要的是, 现在是全面发展的时候, 孩子每天都有不同功课要做, 那么怎么才能利用平时空余的时间, 随时随地的解决单词学习问题, 这就是一个需要考虑的问题了。还有兴趣是最好的老师, 如何一步一步的带领着孩子进入学习英语的环境, 这就显得尤为重要了。

随着移动互联网时代的到来, 手机, 电脑等电子产品是人们生活中必不可少的, 根据《第 41 次中国互联网络发展状况统计报告》发布的数据显示, 截至 2017 年 12 月, 我国网民规模达 7.72 亿, 普及率达到 55.8%, 超过全球平均水平(51.7%)4.1 个百分点, 超过亚洲平均水平(46.7%)9.1 个百分点。因此开发 Web 应用是必然的选择, 目前应用市场上的在线学习网站有很多, 如何选择一款是需要我们认真去考虑的事, 针对这样, 开发一款面向小学生学习英语的网站是

个不错的选择，这需要深入去了解小学的英语和他们学习上遇到的困难，针对他们的问题来设计有趣的类似游戏的学习。当然英语最重要的是单词，目前要做的是针对学习单词来深入研究，以后再丰富其内容。

### 1.3 项目目标及意义

此项目可以帮助学生更好的使用英语资源，合理的利用时间来学习单词，短语和语法知识，并结合练习来帮助学生更好的学习英语，来提高对英语学习的兴趣。将可以在线练习英语测试和检测学习效果，后期可以做出一个习惯养成的计划，来更好培养学生对英语的学习。

### 1.4 项目所用技术介绍

本项目采用前端框架 Bootstrap，和 css,html, jQuery, Ajax, JavaScript 技术。Bootstrap 是流行的响应式前端框架，可以很和的兼容浏览器，在不同的设备上有不同效果，不会丢失主要的样式。利用 jQuery 和 Ajax 以及 JavaScript 技术可以使得项目的前后端分离，将数据更合理的展现在界面上，使得开发更加灵活方便。

## 第2章 需求分析

通过对系统的需求分析，可以做出针对小学生学习英语网站系统的前端开发，做出以下的：

### 2.1 可行性分析

（1）在技术方面，应用前端对好用的 Bootstrap，是目前很受欢迎的前端框架。Bootstrap 是基于 HTML、CSS、JavaScript 的，它简洁灵活，使得 Web 开发更加快捷，它由 Twitter 的设计师 Mark Otto 和 Jacob Thornton 合作开发，是一个 CSS/HTML 框架。Bootstrap 提供了优雅的 HTML 和 CSS 规范，它即是由动态 CSS 语言 Less 写成。Bootstrap 一经推出后颇受欢迎，一直是 GitHub 上的热门开源项目，包括 NASA 的 MSNBC（微软全国广播公司）的 Breaking News 都使用了该项目。国内一些移动开发者较为熟悉的框架，如 WeX5 前端开源框架等，也是基于 Bootstrap 源码进行性能优化而来。因此使用 Bootstrap 框架可以快捷有效的做出网站。给人一种全新的视觉感官。

（2）在操作使用方面上，和大多数网站一样，可以很快的适应，对用户没有任何技能上的要求，在手机或电脑上等一切能与互联网相连的电子产品都可以通过网站地址来进行在线的学习，随时随地的学习，更好的帮助学生学习英语单词，进一步可以提高对英语的学习兴趣，更好的培养学生利用空闲时间去学习的自控力。

（3）从经济的方面考虑，自己设计开发的网站可以不计成本，将知识化为产品，这正是知识体现的价值，所以我认为还是可行的。

综上所述，本系统的开发和设计从任何方面都具有很强的可行性。

### 2.2 需求分析

#### 2.2.1 需求分析概述

在网站系统设计和开发之前，要对小学英语进行深入的研究，做出详细的分析，从而可以更好的去设计网站所需要的功能模块。更好的达到设计的的初衷，将学习英语单词更好的融入到生活中。贴近生活，用易懂的方式，一步一

步的引领学生进入英语的学习，加上在学校老师的教导，更好的完成复习功课。最重要的是，网站的风格要符合学习的氛围，给人产生要学习的感觉，还要方便灵活的操作，性能要稳定。还要有本网站的独特方面，相比较市面上的在线学习网站有自己的独特功能。要达到界面设计友好，数据准确安全，保证知识的准确，还要加大维护网站的时间，根据时间的变化，制定一系列的学习方法和技巧，比如坚持训练的时间，规律的学习时间。做到系统响应的时间要符合使用者最低等待的时间节点。在对数据进行存储时，采用更加人性化的方式，保证一次存储不出错。

### 2.2.2 功能分析

经过很大的需求分析，多次修改，大致设计了网站系统的功能为：

- (1) 单词学习
- (2) 短语学习
- (3) 语法知识
- (4) 课文学习
- (5) 听力训练
- (6) 练习测试
- (7) 单词复习

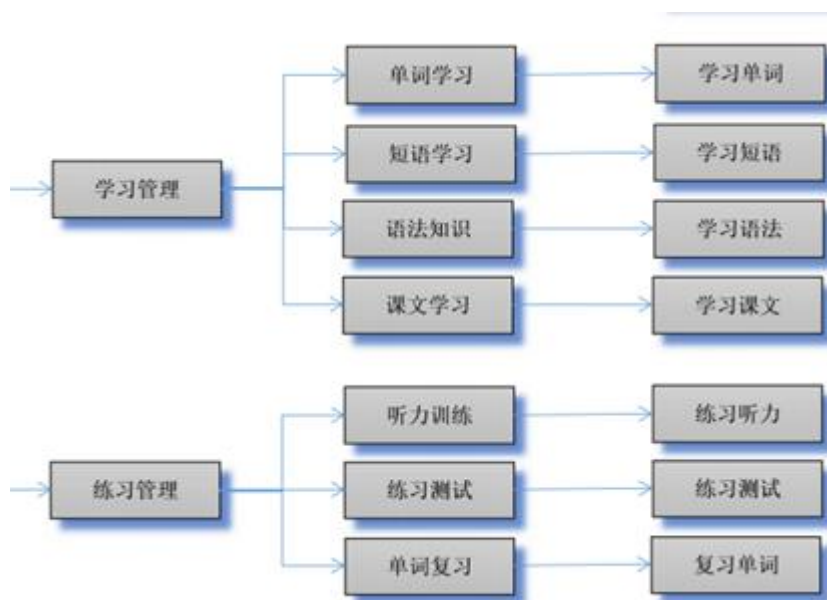


图 2- 1 小学生学习英语网站系统前端开发功能总体需求



### 2.2.3 性能分析

- (1) 要按照严格的数据格式输入，对符合数据格式要求的输入进行提示。
  - (2) 客户端一般响应时间不超过（除数据上传和下载）1 秒；
  - (3) 支持 5000 名用户并发使用，并保证性能不受影响；
  - (4) 支持同时上传数据 1000 条的一次性导入，导入时间不超过 1 分钟。
  - (5) 尽量从使用对象的角度出发，以方便使用本产品；
  - (6) 能够对必须录入的信息进行控制，使用户能够确保信息录入的完整；
  - (7) 容错能力。系统具有一定的容错和抗干扰能力，在非硬件故障或通讯故障（8）时，系统能够保证正常运行，并有足够的提示信息帮助用户有效正确的完成；操作完成时有统一的规范提示信息。
  - (9) 运行环境的变化：软件的设计和实现需要考虑其与运行环境的变化，比能（10）对不同的运行环境提供支持，软件应支持 Windows\_XP 及以上版本的操作系统，支持 Android4 版本以上的安卓手机。
  - (11) 软件具有足够的灵活性，以适应将来可能出现的需求更改和增加。
- 对一些重要的数据按一定的算法进行加密，对数据进行备份和恢复。

### 2.2.4 开发及运行环境

设备：各类 android 手机或平板；  
CPU 类型：多核处理器；  
内存容量：2G 及以上；  
外存容量：16G 及以上。  
操作系统：Android 操作系统;Window10 系统  
操作系统版本：6.0 及以上；  
编译程序：IntelliJ IDEA；  
数据库支持：Mysql；  
数据通信协议采用 HTTP 协议。

## 第3章 系统设计

### 3.1 功能模块设计

表 3-1 小学生学习英语网站系统前端开发功能模块总表

功能名称	子功能名称	功能介绍
用户账户管理	用户注册	在系统中注册个人信息
	用户登录/登出	用户登录/登出到本系统
	修改注册信息	修改个人的注册信息
	修改密码	已注册用户修改账户密码
选择年级	查询年级	查询所有年级名称
	选择年级	确定所属年级
选择单元	查询单元	查询该年级下所有单元
	选择单元	确定学习第几单元
单词学习	学习单词	学习所属单元的单词
短语学习	学习短语	学习所属单元的短语
语法知识	学习语法	学习所属单元的语法
课文学习	学习课文	学习课文中的知识
听力训练	练习听力	练习学生的听力能力
练习测试	练习测试	巩固学生的学习
单词复习	复习单词	复习上次所学过的生词
已学词汇	生词	把生词加入到已学词汇生词表中
	简单词汇	把简单单词加入到已学词汇简单单词表中
错题集	错题	记录做错的题目

## 3.2 操作流程设计

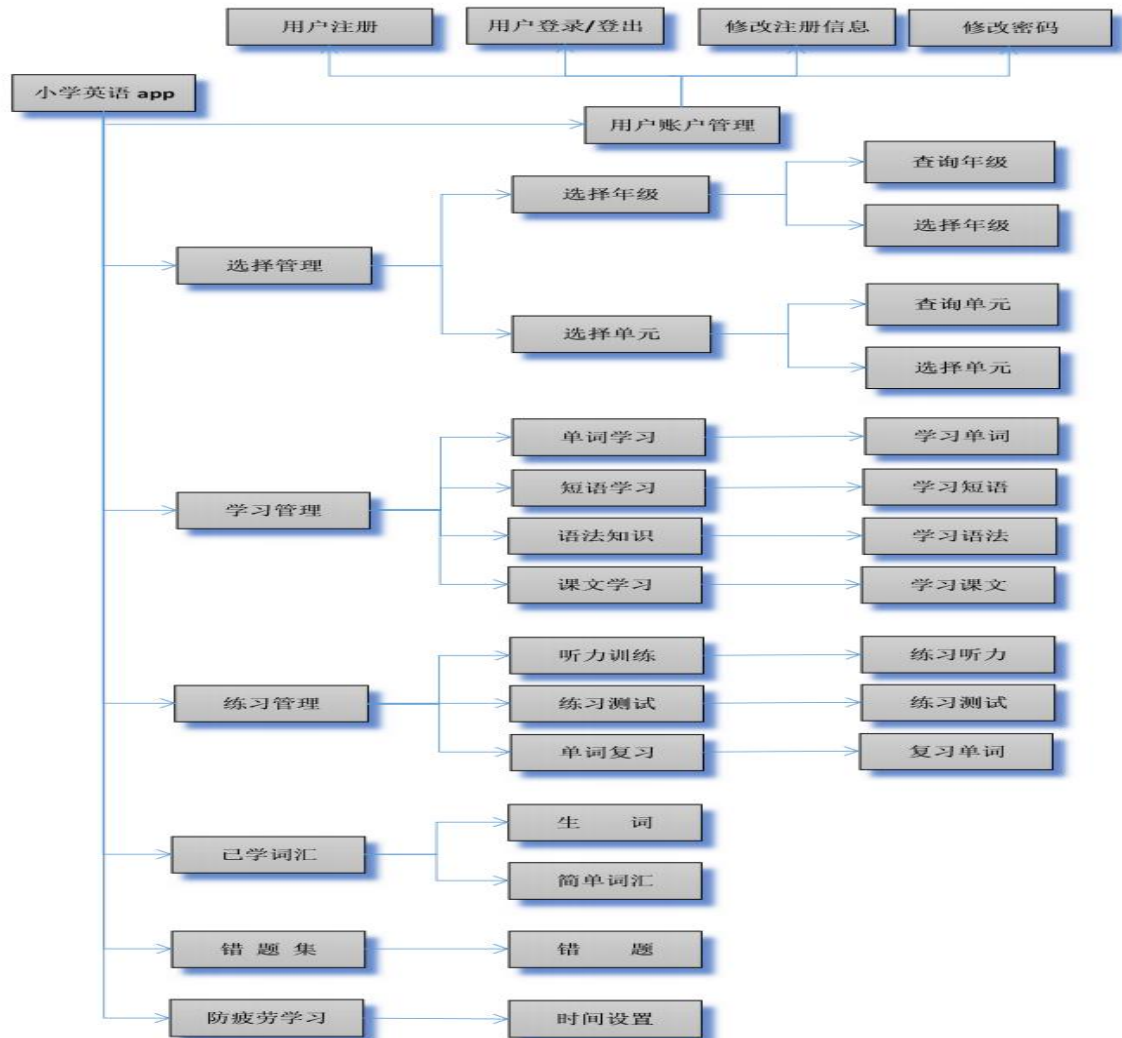


图 3-1 小学生学习英语网站系统前端开发操作流程图

## 3.3 数据库设计

### 3.3.1 数据库概念设计

E-R 图是实体联系图，E-R 模型是对现实世界的一种抽象，其组成部分主要包括实体，联系和属性。通过这三种成分，我们可以很好的抽象出现实世界的个体，并通过彼此之间的联系来抽象整个世界。本网站系统创建了如下表：

表 3-2 小学生学习英语网站系统前端开发数据库总表

表名	用途
user（用户表）	用于存放用户信息
grade（年级表）	存放年级名
element（单元表）	用于存放每个年级的单元名
word（单词表）	用于存放每个单元的单词
phrase（短语表）	用于存放每个单元的短语
grammar（语法表）	用于存放每个单元的语法
text（课文表）	用于存放每个单元的课文内容
hearing（听力表）	用于存放每个单元的听力内容
new_word（生词表）	用于存放每个单元生疏的词汇
simple_words（简单词汇表）	用于存放每个单元的简单词汇
exercise（练习表）	用于存放每个单元的练习
mistakes（错题集表）	用于存放做错的题目

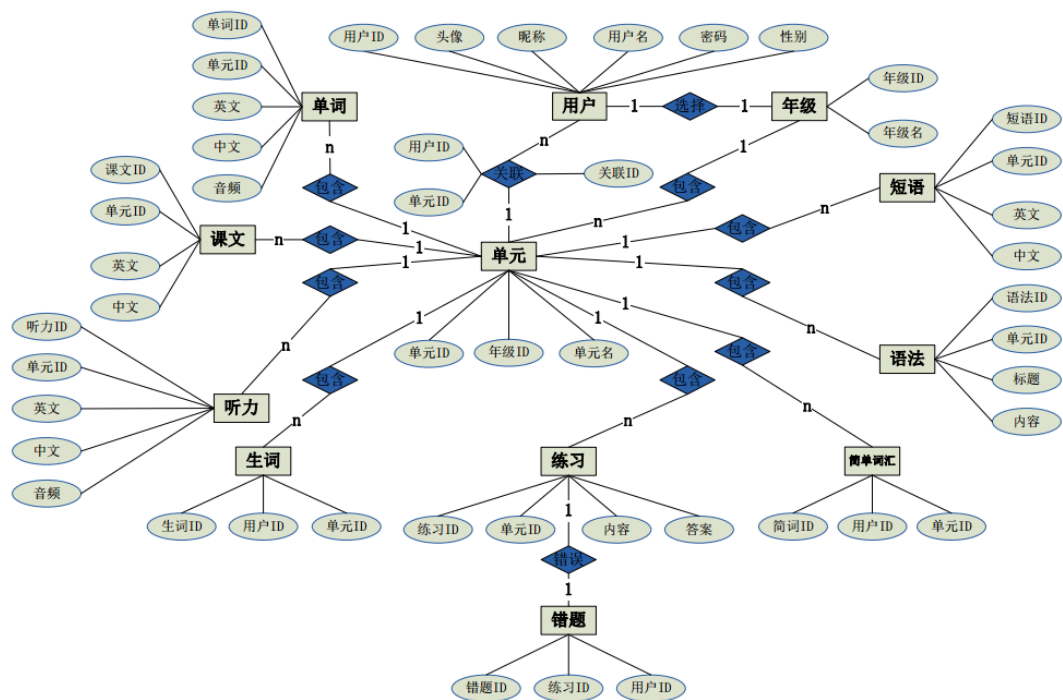


图 3-2 小学生学习英语网站系统前端开发数据库 E-R 图

### 3.3.2 数据库存储结构

数据库名: psenglish

表名: user

表 3-3 小学生学习英语网站系统前端开发数据库-用户表

字段名	意义	数据类型	约束
userID	用户 ID	int	主键, 自增
head_portrait	头像	varchar(255)	
nickname	昵称	varchar(255)	唯一、非空
username	用户名	varchar(255)	唯一、非空
password	密码	varchar(255)	非空

		55)	
gender	性别	char(2)	非空

表名: grade

表 3-4 小学生学习英语网站系统前端开发数据库-年级表

字段名	意义	数据类型	约束
gradeID	年级 ID	int	主键, 自增
Gradenam e	年级	varchar(2 55)	唯一、非空

表名: unit

表 3-5 小学生学习英语网站系统前端开发数据库-单元表

字段名	意义	数据类型	约束
unitID	单元 ID	int	主键, 自增
gradeID	年级 ID	int	外键
unitname	单元	varchar(2 55)	非空

表名: relevance

表 3-6 小学生学习英语网站系统前端开发数据库-关联表

字段名	意义	数据类型	约束
relevanc eID	关联 ID	int	主键, 自增
userID	用户 ID	int	外键
unitID	单元 ID	int	外键

表名: word

表 3-7 小学生学习英语网站系统前端开发数据库-单词表

字段名	意义	数据类型	约束
wordID	单词 ID	int	主键, 自

			增
unitID	单元 ID	int	外键
english	英文	varchar(255)	非空
chinese	中文	varchar(255)	非空
Soundmark	音标	varchar(255)	
Frequency	音频	varchar(255)	

表名: phrase

表 3-8 小学生学习英语网站系统前端开发数据库-短语表

字段名	意义	数据类型	约束
phraseID	短语 ID	int	主键, 自增
unitID	单元 ID	int	外键
english	英文	varchar(1024)	非空
chinese	中文	varchar(1024)	非空

表名: grammar

表 3-9 小学生学习英语网站系统前端开发数据库-语法表

字段名	意义	数据类型	约束
grammarID	语法 ID	int	主键, 自增
unitID	单元 ID	int	外键
title	标题	varchar(255)	

content	内容	varchar(102 40)	非空
---------	----	--------------------	----

表名: text

表 3-10 小学生学习英语网站系统前端开发数据库-课文表

字段名	意义	数据类型	约束
textID	课文 ID	int	主键, 自增
unitID	单元 ID	int	外键
english	英文	varchar(102 40)	非空
chinese	中文	varchar(102 40)	非空

表名: hearing

表 3-11 小学生学习英语网站系统前端开发数据库-听力表

字段名	意义	数据类型	约束
hearingID	听力 ID	int	主键, 自增
unitID	单元 ID	int	外键
english	英文	varchar(102 40)	非空
chinese	中文	varchar(102 40)	非空
Frequency	音频	varchar(255 )	

表名: new word

表 3-12 小学生学习英语网站系统前端开发数据库-生词表

字段名	意义	数据类型	约束
-----	----	------	----



nwID	生词 ID	int	主键，自增
userID	用户 ID	int	外键
wordID	单词 ID	int	外键

表名: simple words

表 3-13 小学生学习英语网站系统前端开发数据库-简单词汇表

字段名	意义	数据类型	约束
swID	简单词汇 ID	int	主键，自增
user ID	用户 ID	int	外键
word ID	单词 ID	int	外键

表名: exercise

表 3-14 小学生学习英语网站系统前端开发数据库-练习表

字段名	意义	数据类型	约束
exercise ID	练习 ID	int	主键，自增
unitID	单元 ID	int	外键
content	内容	varchar(10240)	非空
answer	答案	varchar(10240)	非空

表名: mistakes

表 3-15 小学生学习英语网站系统前端开发数据库-错题表

字段名	意义	数据类型	约束
mistakesID	错题集 ID	int	主键，自增

userID	用户 ID	int	外键
exerciseID	练习 ID	int	外键

### 3.4 接口设计

请求数据 URL: `http://IP:8081/data/request=?`

页面请求 URL: `http://IP:8081/psenglish/(页面名)`

request 的值为每个端口对应的编号

数据返回格式均为 json 格式, 有三个属性, 分别为:

返回码 code;

返回信息 msg;

返回数据 data。

返回码 code=0 代表成功, msg 为“操作成功”, 其他取值含义另行说明。

#### (1) 注册 1.01

表 3-16 小学生学习英语网站系统前端开发接口设计-注册

类型	请求参数	响应参数
POST	nickname username password gender	{ "code": "0", "msg": "操作成功", "data": { "userID": 18, "head_portrait": "\\", "nickname": 妮可, "password": 5555, "gender": 男 } }

#### (2) 登录 1.02

表 3-17 小学生学习英语网站系统前端开发接口设计-登录

类型	请求参数	响应参数
----	------	------

POST	Username password	<pre>{ "code": "0",   "msg": "登录成功",   "data":{     "userID":18,     "head_portrait":\\,     "nickname":妮可,     "password":5555,     "gender":男   } }</pre>
------	----------------------	---

## (3) 通过用户名查询 1.03

表 3-18 小学生学习英语网站系统前端开发接口设计-用户名查询

类型	请求参数	响应参数
GET	username	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "userID":18,     "head_portrait":\\,     "nickname":妮可,     "password":5555,     "gender":男   } }</pre>

## (4) 通过昵称查询 1.04

表 3-19 小学生学习英语网站系统前端开发接口设计-昵称查询

类型	请求参数	响应参数
----	------	------

GET	nickname	<pre>{ "code": "0",   "msg": "登录成功",   "data":{     "userID":18,     "head_portrait":\\,     "nickname":妮可,     "password":5555,     "gender":男   } }</pre>
-----	----------	---

## (5) 查询所有用户 1.05

表 3-20 小学生学习英语网站系统前端开发接口设计-查询用户

类型	请求参数	响应参数
GET	无	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "userID":17,     "head_portrait":\\,     "nickname":测试,     "password":5555,     "gender":男   },   {     "userID":18,     "head_portrait":\\,     "nickname":妮可,     "password":5555,     "gender":男   } }</pre>

		} }
--	--	--------

## (6) 更新用户信息 1.06

表 3-21 小学生学习英语网站系统前端开发接口设计-修改用户

类型	请求参数	响应参数
PUT	Nickname gender	<pre>{ "code": "0",   "msg": "操作成功",   "data": {     "userID": 18,     "head_portrait": "\\",     "nickname": "妮可",     "password": "5555",     "gender": "男"   } }</pre>

## (7) 修改密码 1.07

表 3-22 小学生学习英语网站系统前端开发接口设计-修改密码

类型	请求参数	响应参数
PUT	password newpassword	<pre>{ "code": "0",   "msg": "操作成功",   "data": {     "userID": 18,     "head_portrait": "\\",     "nickname": "妮可",     "password": "5555",     "gender": "男"   } }</pre>

## (8) 删除用户 1.08

表 3-23 小学生学习英语网站系统前端开发接口设计-删除用户

类型	请求参数	响应参数
DELETE	userID	{ "code": "0", "msg": "操作成功", "data":null }

## (9) 退出登录 1.09

表 3-24 小学生学习英语网站系统前端开发接口设计-退出登录

类型	请求参数	响应参数
GET	无	{ "code": "0", "msg": "已成功退出", "data":null }

## (10) 修改头像 1.10

表 3-25 小学生学习英语网站系统前端开发接口设计-修改头像

类型	请求参数	响应参数
POST	file(file)	{ "code": "0", "msg": "操作成功", "data": "userID":18, "head_portrait":\\, "nickname":妮可修改, "password":55555, "gender":女 }

## (11) 新增年级 2.01

表 3-26 小学生学习英语网站系统前端开发接口设计-增加年级

类型	请求参数	响应参数
----	------	------

GET	gradename	{ "code": "0", "msg": "操作成功", "data": "gradeID":5, "gradename":三年级上册 }
-----	-----------	--

## (12) 通过年级查询 2.02

表 3-27 小学生学习英语网站系统前端开发接口设计-查询年级名

类型	请求参数	响应参数
GET	gradename	{ "code": "0", "msg": "操作成功", "data": { "gradeID":5, "gradename":三年级上册 } }

## (13) 查询所有年级 2.03

表 3-28 小学生学习英语网站系统前端开发接口设计-增加年级

类型	请求参数	响应参数
GET	gradename	{ "code": "0", "msg": "操作成功", "data": [ { "gradeID":5, "gradename":三年级上册 }, { "gradeID":5, "gradename":三年级下册 } ] }

## (14) 更改年级名 2.04

表 3-29 小学生学习英语网站系统前端开发接口设计-更改年级名



类型	请求参数	响应参数
PUT	gradeID gradename	{ "code": "0", "msg": "操作成功", "data":{ "gradeID":5, "gradename":四年级上册 }, }

## (15) 删除年级

表 3-30 小学生学习英语网站系统前端开发接口设计-删除年级

类型	请求参数	响应参数
DELETE	gradeID gradename	{ "code": "0", "msg": "操作成功", "data":null }

## (16) 新增单元 3.01

表 3-32 小学生学习英语网站系统前端开发接口设计-增加单元

类型	请求参数	响应参数
POST	gradeID unitname	{ "code": "0", "msg": "操作成功", "data":{ "unitID":2, "gradeID":5, "unitname":单元一 } }

## (17) 通过单元 ID 查询 3.02

表 3-33 小学生学习英语网站系统前端开发接口设计-单元 ID 查询

类型	请求参数	响应参数
GET	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "unitID":2,     "gradeID":5,     "unitname":单元一   } }</pre>

(18) 通过年级 ID 查询 3.03

表 3-34 小学生学习英语网站系统前端开发接口设计-年级 ID 查询

类型	请求参数	响应参数
GET	gradeID	<pre>{ "code": "0",   "msg": "操作成功",   "data":[{"unitID":2,     "gradeID":5,     "unitname":单元一   },   {     "unitID":3,     "gradeID":5,     "unitname":单元二   } ] }</pre>

(19) 更新单元名 3.04

表 3-35 小学生学习英语网站系统前端开发接口设计-修改单元

类型	请求参数	响应参数
PUT	gradeID unitID unitname	{ "code": "0", "msg": "操作成功", "data":{ "unitID":2, "gradeID":5, "unitname":单元二 } }

## (20) 删除单元 3.05

表 3-36 小学生学习英语网站系统前端开发接口设计-删除单元

类型	请求参数	响应参数
DELETE	unitID	{ "code": "0", "msg": "操作成功", "data":null }

## (21) 用户绑定单元 4.01

表 3-37 小学生学习英语网站系统前端开发接口设计-用户绑定单元

类型	请求参数	响应参数
POST	unitID	{ "code": "0", "msg": "操作成功", "data":{ "relevanceID":1, "userID":22, "unitID":2 } }

## (22) 通过用户 ID 查询 4.02

表 3-38 小学生学习英语网站系统前端开发接口设计-用户 ID 查询单元

类型	请求参数	响应参数
GET	无	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "relevanceID":1,     "userID":22,     "unitID":2   } }</pre>

## (23) 更新用户所在单元 4.03

表 3-39 小学生学习英语网站系统前端开发接口设计-更新用户单元

类型	请求参数	响应参数
PUT	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "relevanceID":1,     "userID":22,     "unitID":3   } }</pre>

## (24) 增加单词 5.01

表 3-40 小学生学习英语网站系统前端开发接口设计-增加单词

类型	请求参数	响应参数
POST	unitID english chinese soundmark file(file)	<pre>{ "code": "0",   "msg": "操作成功",   "data":[{     "wordID":1,     "unitID":2,</pre>

		<pre> "english":"easy", "chinese":"容易的", "soundmark":"[izi]", "frenquency":"grade\\三年 级\\单元一\\word\\easy.mp3" } ] } </pre>
--	--	--

## (25) 查询相应单元下的所有单词 5.02

表 3-40 小学生学习英语网站系统前端开发接口设计-查询单元单词

类型	请求参数	响应参数
GET	unitID	<pre> { "code": "0",   "msg": "操作成功",   "data": [{     "wordID": 1,     "unitID": 2,     "english": "easy",     "chinese": "容易的",     "soundmark": "[izi]",     "frenquency": "grade\\三年 级\\单元一\\word\\easy.mp3"   } ] } </pre>

## (26) 通过单词名查询 5.03

表 3-41 小学生学习英语网站系统前端开发接口设计-查询单词名

类型	请求参数	响应参数
----	------	------

GET	english	<pre>{ "code": "0",   "msg": "操作成功",   "data": [{     "wordID": 1,     "unitID": 2,     "english": "easy",     "chinese": "容易的",     "soundmark": "[izi]",     "frequency": "grade\\三年级\\单元一\\word\\easy.mp3"   }] }</pre>
-----	---------	--

(27) 修改单词 5.04

表 3-42 小学生学习英语网站系统前端开发接口设计-修改单词

类型	请求参数	响应参数
POST	wordID unitID english chinese soundmark file(file)	<pre>{ "code": "0",   "msg": "操作成功",   "data": [{     "wordID": 1,     "unitID": 2,     "english": "easy",     "chinese": "容易的",     "soundmark": "[izi]",     "frequency": "grade\\三年级\\单元一\\word\\easy.mp3"   }] }</pre>

		}
--	--	---

## (28) 删除单词 5.05

表 3-43 小学生学习英语网站系统前端开发接口设计-删除单词

类型	请求参数	响应参数
DELETE	wordID	{ "code": "0", "msg": "操作成功", "data":null }

## (29) 增加短语 6.01

表 3-44 小学生学习英语网站系统前端开发接口设计-增加短语

类型	请求参数	响应参数
POST	unitID english chinese	{ "code": "0", "msg": "操作成功", "data":{ "phraseID":1, "unitID":2, "english":"go to", "chinese":"转到" } }

## (30) 查询相应单元的所有短语 6.02

表 3-45 小学生学习英语网站系统前端开发接口设计-查询单元短语

类型	请求参数	响应参数
----	------	------

GET	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":[     {       "phraseID":1,       "unitID":2,       "english":"go to",       "chinese":"转到"     },     {     }   ] }</pre>
-----	--------	--

## (31) 通过短语 ID 查询 6.03

表 3-46 小学生学习英语网站系统前端开发接口设计-短语 ID 查询

类型	请求参数	响应参数
GET	phraseID	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "phraseID":1,     "unitID":2,     "english":"go to",     "chinese":"转到"   } }</pre>

## (32) 修改短语 6.04

表 3-47 小学生学习英语网站系统前端开发接口设计-修改短语

类型	请求参数	响应参数
----	------	------



PUT	<b>phraseID</b> <b>unitID</b> <b>english</b> <b>chinese</b>	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "phraseID":1,     "unitID":2,     "english":"be suited to",     "chinese":"适合于"   } }</pre>
-----	--	---

## (33) 删除短语 6.05

表 3-48 小学生学习英语网站系统前端开发接口设计-删除短语

类型	请求参数	响应参数
DELETE	<b>phraseID</b>	<pre>{ "code": "0",   "msg": "操作成功",   "data":null }</pre>

## (34) 增加语法 7.01

表 3-49 小学生学习英语网站系统前端开发接口设计-查询语法

类型	请求参数	响应参数
POST	<b>unitID</b> <b>title</b> <b>content</b>	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "gramerID":3,     "unitID":2,     "title":"时态",     "content":"时态是一种...."   } }</pre>

## (35) 查询相应单元下的语法 7.02

表 3-50 小学生学习英语网站系统前端开发接口设计-查询单元语法

类型	请求参数	响应参数
GET	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":[     {       "grammerID":2,       "unitID":2,       "title":"代词",       "content":"代词....."     },{       "gramerID":3,       "unitID":2,       "title":"时态",       "content":"时态是一种...."     }   ] }</pre>

## (36) 通过语法 ID 查询 7.03

表 3-51 小学生学习英语网站系统前端开发接口设计-语法 ID 查询

类型	请求参数	响应参数
GET	grammerID	<pre>{ "code": "0",   "msg": "操作成功",   "data":   {     "gramerID":3,</pre>

		<pre> "unitID":2, "title":"时态", "content":"时态是一种...." } </pre>
--	--	--

(37) 修改语法 7.04

表 3-52 小学生学习英语网站系统前端开发接口设计-修改语法

PUT	<pre> grammarID unitID title content </pre>	<pre> { "code": "0",   "msg": "操作成功",   "data":   {     "gramerID":3,     "unitID":2,     "title":"名词",     "content":"名词...."   } } </pre>
-----	---	---

(38) 删除语法 7.05

表 3-53 小学生学习英语网站系统前端开发接口设计-删除语法

类型	请求参数	响应参数
DELETE	grammarID	<pre> { "code": "0",   "msg": "操作成功",   "data":null } </pre>

(39) 增加课文内容 8.01

表 3-54 小学生学习英语网站系统前端开发接口设计-增加课文

类型	请求参数	响应参数
----	------	------

POST	unitID english chinese	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "textID":1,     "unitID":2,     "english":"Singapore may be.....",     "chinese":"新加坡...."   } }</pre>
------	------------------------------	--

(40) 查询相应单元下的课文 8.02

表 3-55 小学生学习英语网站系统前端开发接口设计-查询单元课文

类型	请求参数	响应参数
GET	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":[     {       "textID":1,       "unitID":2,       "english":"Singapore may be.....",       "chinese":"新加坡...."     },     {       "textID":2,       "unitID":2,       "english":"With.....",       "chinese":"由于...."     }   ] }</pre>

		<pre>         }       ]     } </pre>
--	--	--------------------------------------

## (41) 通过课文 ID 查询 8.03

表 3-56 小学生学习英语网站系统前端开发接口设计-课文 ID 查询

类型	请求参数	响应参数
GET	textID	<pre> { "code": "0",   "msg": "操作成功",   "data": {     "textID": 1,     "unitID": 2,     "english": "Singapore may               be.....",     "chinese": "新加坡...."   } } </pre>

## (42) 修改课文内容 8.04

表 3-57 小学生学习英语网站系统前端开发接口设计-修改课文

类型	请求参数	响应参数
PUT	unitID textID english chinese	<pre> { "code": "0",   "msg": "操作成功",   "data": {     "textID": 1,     "unitID": 2,     "english": "Singapore may               be.....",     "chinese": "新加坡...."   } } </pre>

		}
--	--	---

## (43) 删除课文 8.05

表 3-58 小学生学习英语网站系统前端开发接口设计-删除课文

类型	请求参数	响应参数
DELETE	textID	{ "code": "0", "msg": "操作成功", "data":null }

## (44) 增加听力内容 9.01

表 3-59 小学生学习英语网站系统前端开发接口设计-增加听力

类型	请求参数	响应参数
POST	unitID english chinese file(file)	{ "code": "0", "msg": "操作成功", "data":{ "hearingID":3, "unitID":2, "english":"Singapore may be.....", "chinese":"新加坡....", "frequency":"grade\\三年级 上\\单元一\\hearing\\Try.mp3", } }

## (45) 查询相应单元的听力内容 9.02

表 3-60 小学生学习英语网站系统前端开发接口设计-查询单元听力

类型	请求参数	响应参数
----	------	------

GET	unitID	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "hearingID":3,     "unitID":2,     "english":"Singapore may be.....",     "chinese":"新加坡....",     "frequency":"grade\\三年级 上\\单元一\\hearing\\Try.mp3",   } }</pre>
-----	--------	---

## (46) 通过听力 ID 查询

表 3-61 小学生学习英语网站系统前端开发接口设计-听力 ID 查询

类型	请求参数	响应参数
GET	hearingID	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "hearingID":3,     "unitID":2,     "english":"Singapore may be.....",     "chinese":"新加坡....",     "frequency":"grade\\三年级 上\\单元一\\hearing\\Try.mp3",   } }</pre>

## (47) 修改听力内容 9.04

表 3-62 小学生学习英语网站系统前端开发接口设计-查询听力

类型	请求参数	响应参数
POST	unitID hearingID english chinese file(file)	<pre>{ "code": "0",   "msg": "操作成功",   "data":{     "hearingID":3,     "unitID":2,     "english":"Singapore may     be.....",     "chinese":"新加坡....",     "frequency":"grade\\三年级     上\\单元一\\hearing\\Try.mp3",   } }</pre>

(48) 删除听力 9.05

表 3-63 小学生学习英语网站系统前端开发接口设计-删除听力

类型	请求参数	响应参数
DELETE	hearingID	<pre>{ "code": "0",   "msg": "操作成功",   "data":null   } }</pre>



## 第 4 章 系统实现

### 4.1 首页

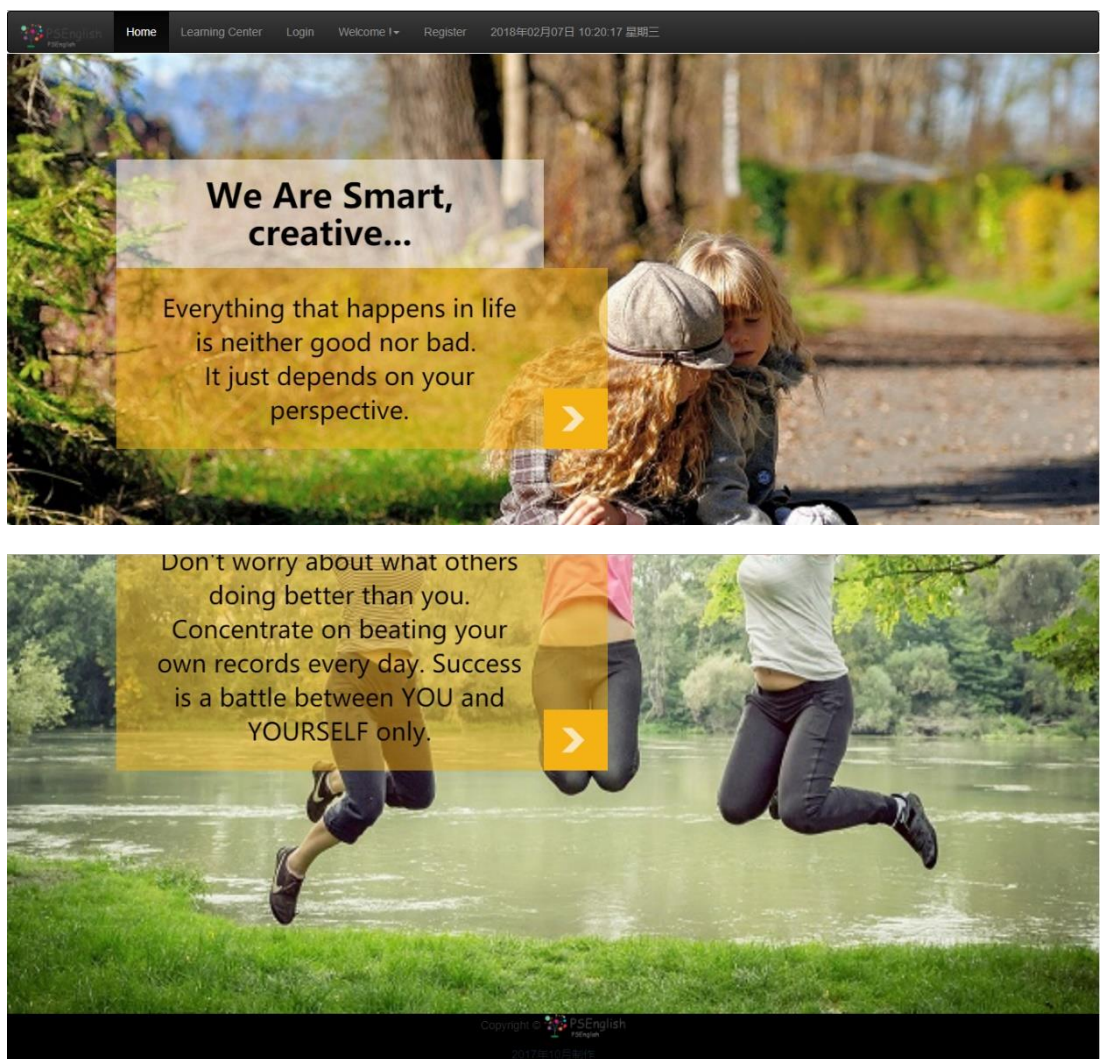


图 4- 1 首页

未登录之前，不可进入个人中心，学习中心，退出登录，登录后，可进入个人中心，学习中心，退出登录。

#### (1) 界面布局

<nav>导航栏</nav>

<div class="banner">图片展示</div>

<!—底部显示制作时间-->

```

<div>
  <div style="background-color:black">
    <p>Copyright &copy;
    <p><a href="index">2017 年 10 月制作</a></p>
  </div>
</div>

```

## (2) 获取及提交数据

```

获取用户名
${username}

```

## (3) 填充数据

```

用户名 ${username}
时间  var years = d.getFullYear();
      var month = add_zero(d.getMonth() + 1);
      var days = add_zero(d.getDate());
      var hours = add_zero(d.getHours());
      var minutes = add_zero(d.getMinutes());
      var seconds = add_zero(d.getSeconds());
      var ndate = years + "年" + month + "月" + days + "日 " + hours
+ ":"
      + minutes + ":" + seconds + " " + week;
      var divT = document.getElementById("logInfo");
      divT.innerHTML = ndate;

```

## (4) 输入的数据有效性验证

```

$(document).ready(function () {
  if ("${username}" != "") {
    $("#jumplogin").attr("class", "hidden");
    $("#jumpregister").attr("class", "hidden");
  }
});
//跳转到学习中心
function jumpCenter() {
  if ("${username}" == "") {
    Toast('您还未登录,请先去登录哟', 2000);
    //alert("您还未登录,请先去登录哟");
  } else {

```

```

        window.location.href = 'study';
    }
}

```

## 4.2 登录

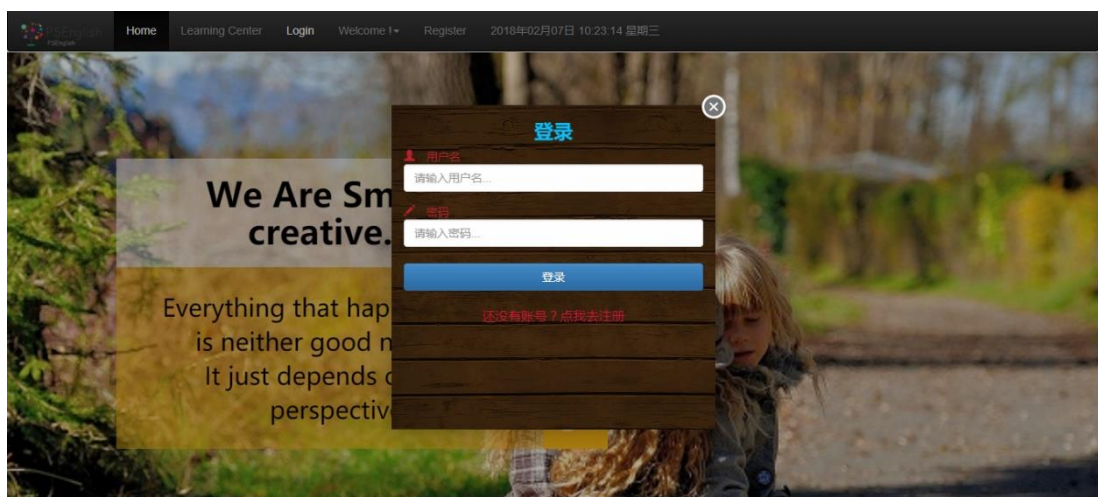


图 4- 2 登录

登录时，若无账号，提示先去注册，成功后方可进行登录。

### (1) 界面布局

```

<div class="modal-body">
<label class="fontsize">登录</label>
<input type="text" name="username" id="username" placeholder="请输入
用户名..." title="请输入用户名" />
<input type="password" name="password" id="password" placeholder="
请输入密码..." title="请输入密码"/>
<input id="btn-login" type="button" onclick="login()" title="登录"
value="登录"/>
<a href="register" title="点我去注册">还没有账号? 点我去注册</a>
</div>

```

## (2) 获取及提交数据

```
$.ajax({type: "POST",
        url:    "/data/request=1.02?username=" + username +
"&password=" + password,
        dataType: "json",
        success: function (data) {
            if (data.code == "1") {
                var message = data.msg;
                window.location.href = 'errortips';
            } else if (data.code == "0") {
                var message = data.msg;
                window.location.href = 'index';
                // 先获取窗口索引，才能关闭窗口
                var index = parent.layer.getFrameIndex(window.name);
                window.parent.location.reload();
                parent.layer.close(index);
            } else {
                var message = data.msg;
                window.location.href = 'errortips';
            }
        }
    });
```

## (3) 填充数据

此页面无填充数据

## (4) 输入的数据有效性验证

```
if (username == '' || username == undefined || username == null) {
    Toast('用户名为空', 2000);
    // alert("输入的用户名为空")
} else if (password == '' || password == undefined || password == null) {
    Toast('密码为空', 2000);
    //alert("输入的密码为空")
}
```

## 4.3 注册

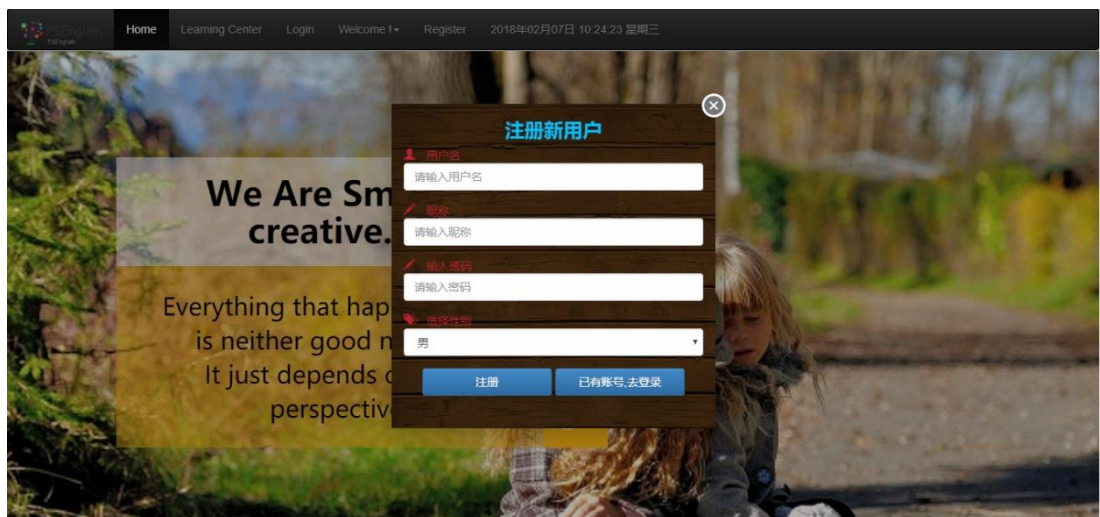


图 4- 3 注册

注册成功后跳转到登录窗口，用户注册之后可以使用学习资源，学习单词和语法知识。

### (1) 界面布局

```
<div class="modal-body">
<label class="fontsize">注册新用户</label>
<input type="text" name="username" id="username" />
<input type="text" name="nickname" id="nickname" />
<input type="password" name="password" id="password"/>
<select id="gender" class="form-control">
<option value="男">男</option>
<option value="女">女</option>
<input id="btn-register" type="button"
value="注册" title="注册"/>
<a href="login.htm">
<input type="button" value="已有账号, 去登录"/>
</div>
```

### (2) 获取及提交数据

```
$.ajax({
    type: "GET",
    url: "/data/request=1.04?nickname=" + nickname,
    dataType: "json",
    success: function (data) {
```

```

        if (data.code == "0") {
            Toast('昵称已存在', 2000);
        } else {
$.ajax({
    type: "POST",
    url: "/data/request=1.01?username=" + username + "&password="
+ password + "&nickname=" + nickname + "&gender=" + gender,
    dataType: "json",
    success: function (data) {
        if (data.code == "0") {
            var message = data.msg;
            setTimeout('window.location="login"', 3000);
            Toast('注册成功,耐心等待 3 秒钟,进入登录窗口', 3000);
        } else {
            var message = data.msg;
            alert(message);
            window.location.href = 'errortips';
        }
    }
});

```

## (3) 填充数据

此页面无填充数据

## (4) 输入的数据有效性验证

```

var username = $("#username").val();
var nickname = $("#nickname").val();
var password = $("#password").val();
var gender = $("#gender option:selected").val();
if (username == '' || username == undefined || username == null) {
    Toast('用户名为空', 2000);
} else if (nickname == '' || nickname == undefined || nickname == null)
    Toast('昵称为空', 2000);
else if (password == '' || password == undefined || password == null)
    Toast('密码为空', 2000);

```

## 4.4 个人中心

图 4- 4 个人中心

登录后才能进入个人中心，可以进行修改密码和更换头像操作，可查看自己的学习记录。

### (1) 界面布局

```
<div class="row centers">
  <div class="col-md-3">
    <div>
      <table class="autocenter"></table>
    </div>
  </div>
  <div class="col-md-9">
    <div style="width: 120px; float: left;">
      <ul class="list-group"> </ul>
    </div>
    <iframe style="margin-top: -18px; border: 1px solid blue;"
    src="basicdata" id="iframe-main" frameborder="no" order="0"
    allowtransparency="yes" width="100%"
    height="500px">
  </iframe>
</div>
</div>
</div>
```

### (2) 获取及提交数据

```
$.ajax({
  type: "GET",
```

```

url: "/data/request=1.03?username=" + ${username},
dataType: "json",
success: function (data) {
    if (data.code == "1") {
        var message = data.msg;
        alert(message);
    } else if (data.code == "0") {
        if (data.data.head_portrait == null) {
            $("#images").html('<div style="text-align: center">' + '' + '</div>');
        } else {
            $("#images").html('<div style="text-align: center" width="50px">' + '' + '</div>');
        }
    }
});

```

## (3) 填充数据

```

用户名 ${username}
头像    $("#images").html('<div style="text-align: center">' +
'' +
'</div>')

```

## (4) 输入的数据有效验证

此页面无需输入数据



## 4.5 学习中心



图 4- 5 学习中心

登录后才能进入学习中心，可以分单元的进行查看单词和短语，以及可以测试自己的学习成果。

### (1) 界面布局

轮播图片

```
<div class="carousel-inner">
  <div class="carousel-item active">
</div>
  <div class="carousel-item">
</div>
  <div class="carousel-item">
</div>
</div>
```

菜单栏

```
<div class="row">
  <div class="col-md-2 column"></div>
  <div class="col-md-3 column right">
  <div class="col-md-7 column right">
</div>
```

### (2) 获取及提交数据

无

### (3) 填充数据

无

### (4) 输入的数据有效验证

无

## 4.6 后台用户管理



图 4- 6 后台用户管理

管理用户的个人信息，查看使用人群，便于为后期开发功能做参考。可以做出数据的统计，一边开发相应的学习产品，更好的将学习融入到网站学习平台里。

### (1) 界面布局

#### 导航栏

```
<nav class="navbar navbar-inverse" role="navigation">
  <a class="navbar-brand" href="#">PSEnglish 后台管理系统</a>
  <ul class="nav navbar-nav navbar-right">
    <li>
      <a href="#"><span class="glyphicon glyphicon-user"></span>欢迎您:admin</a>
    </li>
    <li>
      <a href="#"><span class="glyphicon glyphicon-log-in"></span>退出登录</a>
    </li>
  </ul>
</nav>
```

#### 菜单栏

```
<div class="row conters">
  <div style="float: left;" class="col-md-2">
```

```

<ul class="list-group">
  <li class="list-group-item">
    用户管理</li>
  <li class="list-group-item">
    年级管理</li>
  <li class="list-group-item">
    单词管理</li>
  <li class="list-group-item">
    短语管理</li>
  <li class="list-group-item">
    练习管理</li>
  <li class="list-group-item">
    课文管理</li>
  <li class="list-group-item">
    语法管理</li>
  <li class="list-group-item">
    听力管理</h4></li>
</ul>
</div>
点击菜单进行切换
<div style="float: left; padding-top: 20px;" class="col-md-10">
  <iframe style="margin-top: -18px;" src="back_user"
id="iframe-main" frameborder="no"
  border="0" allowtransparency="yes" width="100%"
  height="500px"></iframe>
</div>
</div>

```

## (2) 获取及提交数据

```

$(document).ready(function () {
  $.ajax({
    type: "GET",
    url: "/data/request=1.05",
    dataType: "json",
    success: function (data) {
      if (data.code == "1") {
        var message = data.msg;
        alert(message);
      } else if (data.code == "0") {
        var dataObj = data.data;
        // alert(dataObj+"dataObj");
        $("#usertables tr:not(:first)").empty();
        if (data.data.length == 0) {

```

```

        $("#usertables").html(' <div
style="text-align: center">' + ' '
+ ' <h3>暂无用户数据' + ' </div>')
    } else {
        $.each(dataObj, function (index, item) {
            //alert(frequency);
            userID = item.userID;
            var $tr = $("<tr></tr>"); //表格的一行,
            在里面可以加各种属性;
            var $td = $("<td></td>"); //行中的元素

            $tr.append($td.clone().text(item.username)); //把数据填充到单元格
            中

            $tr.append($td.clone().text(item.nickname));

            $tr.append($td.clone().text(item.password));
            $tr.append($td.clone().html(' <div
style="text-align: center">' + ' <img src="" + item.head_portrait +
' " class="commentAvatarImage" width="100px" height="100px"/>' +
' </div>' ));

            $tr.append($td.clone().html(' <div><input type="button" value="删除
" class="btn btn-danger" onclick="jumpDel(' + userID + ');"/></div>' ));
            $("#users").after($tr); //最后要把内容
            放入要绑定的地方

        });
    }
}
});
});

```

## (3) 填充数据

```

$(document).ready(function () {
    $('.menu').click(function (e) {
        $("#iframe-main").attr("src", $(this).attr('link'));
    });
});

```

## (4) 输入的数据有效验证

无

## 4.7 数据中心管理

### 4.7.1 年级和单元



图 4-7 数据中心管理一年级和单元

管理年级和单元，对年级和单元进行添加

#### (1) 界面布局

```
<h1>添加年级</h1>
<div class="row">
  <div class="col-md-6 column">
    <div class="table-responsive">
      <table class="table table-hover table-condensed
table-bordered text-center" id="gradetables">
        <tr class="info text-center text-primary"></tr>
        <tr id="grade"></tr>
      </table>
    </div>
    <input type="button" value="增加年级" data-toggle="modal"
data-target="#addClass"/>
  </div>
  <div class="modal fade" id="addClass" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel"aria-hidden="true">
    <div class="modal-dialog modal-sm">
      <div class="modal-content">
```

```

    <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal"
aria-hidden="true">&times;</button>
        <h4 class="modal-title" id="myModalLabel">添加年级</h4>
    </div>
<div class="modal-body">
<label class="text-info">选择年级</label>
<select class="form-control" id="grades"></select>
<!-- /单元 -->
<div class="col-md-6 column">
<div class="table-responsive">
<select class="form-control" id="selectgrade">
<option value="-1">请选择年级查询单元</option>
</select>
<table class="table table-hover table-condensed table-bordered
text-center" id="unittable">
    <tr class="info text-center text-primary"></tr>
    <tr id="units"></tr>
</table>
<!-- 按钮触发模态框 -->
<input type="button" value="增加单元" data-toggle="modal"
data-target="#addUnit"/>
<!-- 模态框 (Modal) -->
<div class="modal fade" id="addUnit" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel"
aria-hidden="true">
<div class="modal-dialog modal-sm">
<div class="modal-content">
<h4 class="modal-title" id="myModalLabel">添加单元</h4>
</div>
<div class="modal-body">
<label class="text-info">选择年级</label>
<select class="form-control" id="unit_grade">

```

```

</select>
<label class="text-danger">添加单元</label>
<select class="form-control" id="unitname">
</select>

```

## (2) 获取及提交数据

```

$(document).ready(function () {
    $.ajax({
        type: "GET",
        url: "/data/request=2.03",
        dataType: "json",
        success: function (data) {
            // alert(JSON.stringify(data));
            if (data.code == "1") {
                var message = data.msg;
                alert(message);
            } else if (data.code == "0") {
                var dataObj = data.data;
                var i=dataObj.length+1;
                $("#unittable tr:not(:first)").empty();
                if (data.data.length == 0) {
                    $("#grade").html('<div style="text-align: center">' + '' + '<h3>暂无数据' + '</div>')
                } else {
                    $.each(dataObj, function (index, item) {

                    });
                }
            }
        }
    });

```

## (3) 填充数据

```
gradename = item.gradename;
```

```

gradeID = item.gradeID;
var $tr = $("<tr></tr>"); //表格的一行，在里面可以加各种属性;
var $td = $("<td></td>"); //行中的元素
$str.append($td.clone().text(i)); //把数据填充到单元格中
$str.append($td.clone().text(gradename));
$str.append($td.clone().html(' <input type="button" value=" 修改 "
onclick="jumpChangegrade(' + gradeID + ');" class="btn btn-danger
btn-xs" />' + ' &nbsp;  <input type="button" value=" 删除 "
onclick="jumpDelgrade(' + gradeID + ');" class="btn btn-primary
btn-xs" />'));
$("#grade").after($tr); //最后要把内容放入要绑定的地方
$("#unit_grade").append("<option value=' " + gradeID + "'>" +
gradename + "</option>");
$("#selectgrade").append("<option value=' " + gradeID + "'>" +
gradename + "</option>");

```

#### (4) 输入的数据有效验证

```
$("#selectgrade option:selected").val();
```

### 4.7.2 单词

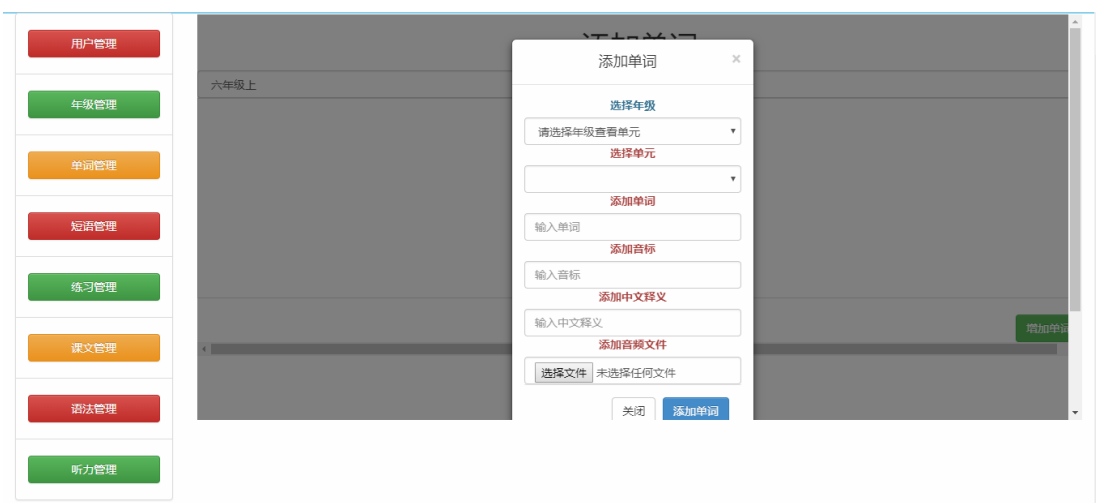


图 4-8 数据中心管理—单词

#### (1) 界面布局

```

<h1>添加单词</h1>
<div class="table-responsive">
  <div class="row">    </div>

```



```

<table id="wordtables">
  <thead class="text-center">
    <tr class="info text-center text-primary"></tr>
  </thead>
  <tr id="words">
  </tr>
</table>
<input type="button" value="增加单词" data-toggle="modal"
data-target="#addWord"/>
</div>
弹出框
<div class="modal fade" id="addWord" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">
      <div class="modal-header">
      </div>
    </div>
    <iframe id="id_iframe" name="id_iframe" height="100px;"></iframe>
  </div>

```

## (2) 获取及提交数据

```

$(document).ready(function () {
  //年级
  $.ajax({
    type: "GET",
    url: "/data/request=2.03",
    dataType: "json",
    success: function (data) {
    });

```

## (3) 填充数据

```

年级
$("#selectgrade").append("<option value=' " + gradeID + "'>" +
gradename + "</option>");
$("#modal_grade").append("<option value=' " + gradeID + "'>" +
gradename + "</option>");
单元
$("#selectunit").append("<option value=' " + unitID + "'>" + unitname
+ "</option>");
$str.append($td.clone().text(item.wordID)); //把数据填充到单元格中

```

```

$str.append($td.clone().text(selectGradename + selectUnitname));
$str.append($td.clone().text(item.english));
$str.append($td.clone().text(item.soundmark));
$str.append($td.clone().text(item.chinese));
$str.append($td.clone().html('<audio controls>' + '<source src="' +
item.frequency + '" type="audio/ogg">' + '</audio>'));
$str.append($td.clone().html('<input type="button" value=" 修改 "
class="btn btn-primary btn-xs" onclick="jumpEditword(' + delwordID
+ ');"/>&nbsp; <input type="button" value=" 删除 "
onclick="jumpDelwords(' + delwordID + ');" class="btn btn-primary
btn-xs" />'));
$("#words").after($tr);//最后要把内容放入要绑定的地方

```

(4) 输入的数据有效验证

```

$("#selectunit option:selected").val();

```

### 4.7.3 短语



图 4-9 数据中心管理—短语

注：此界面布局和上一界面布局相似，这里不再详细介绍。

## 4.7.4 语法



图 4-10 数据中性管理—语法

注：此界面布局和上一界面布局相似，这里不再详细介绍。

## 4.7.5 课文



图 4-11 数据中性管理—课文

注：此界面布局和上一界面布局相似，这里不再详细介绍。

## 4.7.6 练习

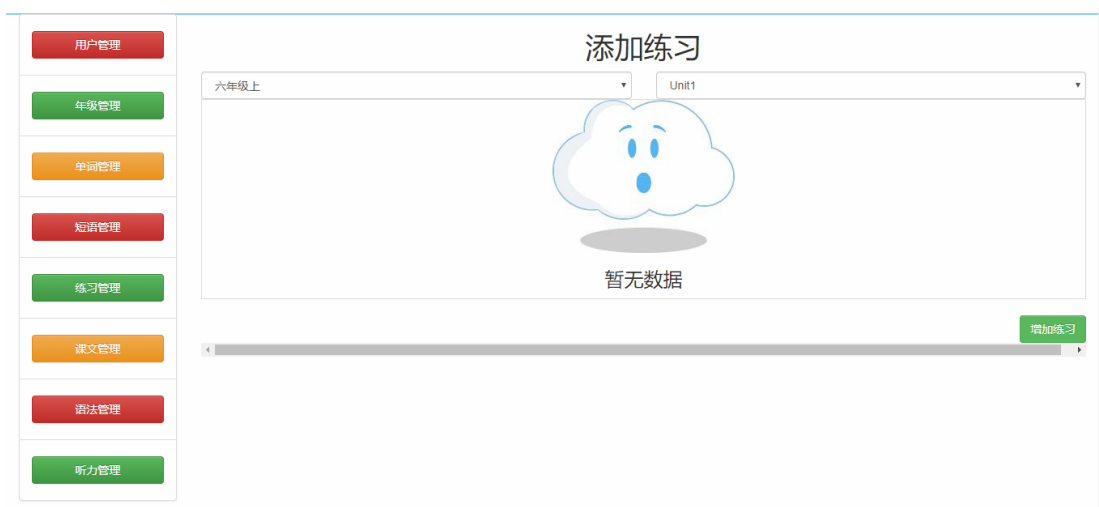


图 4-12 数据中心管理—练习

注：此界面布局和上一界面布局相似，这里不再详细介绍。

## 4.7.7 听力

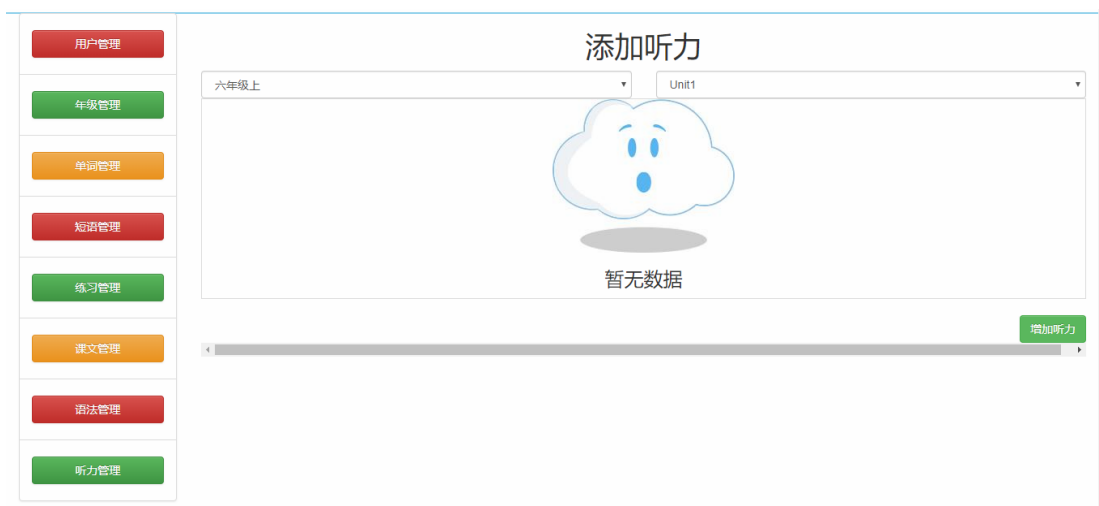


图 4-13 数据中心管理—听力

注：此界面布局和上一界面布局相似，这里不再详细介绍。

## 第 5 章 系统安装及部署

### 5.1 软件运行环境

- (1)操作系统: Windows2000/2003/2008 Server 或 Linux 系统
- (2)数据库服务器: Mysql5.6 及以上版本
- (3)web 服务器: Apache tomcat 6 及以上版本或 Jboss4.0 及以上
- (4)运行环境: Sun JDK 1.5 及以上版本
- (5)客户端浏览器: Chrome,Firefox 等

### 5.2 安装步骤

#### (1) 操作系统准备

安装 Linux 系统, Centos 7 版本



图 5-1 系统安装与部署-配置

#### (2) 数据库准备

在 Centos7 上安装 Mysql5.7 版本, 下载 MySQL 数据库版本, 并安装。

Red Hat Enterprise Linux 7 / Oracle Linux 7 (Architecture Independent), RPM Package (mysql57-community-release-el7-11.noarch.rpm)	25.1K	<a href="#">Download</a>
Red Hat Enterprise Linux 6 / Oracle Linux 6 (Architecture Independent), RPM Package (mysql57-community-release-el6-11.noarch.rpm)	25.1K	<a href="#">Download</a>
Red Hat Enterprise Linux 5 / Oracle Linux 5 (Architecture Independent), RPM Package (mysql57-community-release-el5-8.noarch.rpm)	23.8K	<a href="#">Download</a>
Fedora 27 (Architecture Independent), RPM Package (mysql57-community-release-fc27-10.noarch.rpm)	29.4K	<a href="#">Download</a>
Fedora 26 (Architecture Independent), RPM Package (mysql57-community-release-fc26-10.noarch.rpm)	29.1K	<a href="#">Download</a>
Fedora 25 (Architecture Independent), RPM Package (mysql57-community-release-fc25-10.noarch.rpm)	29.0K	<a href="#">Download</a>

图 5-2 系统安装与部署-配置

### (3) JAVA 环境准备

在 Centos7 在线安装 Java

#### 1、搜索 JDK 有哪些可安装的版本

```
yum search jdk
```

#### 2、选择版本进行安装，这里选择 OpenJDK Runtime Environment，例如：

```
yum -y install java-1.8.0-openjdk.x86_64
```

#### 3、查看是否安装成功

```
java -version
```

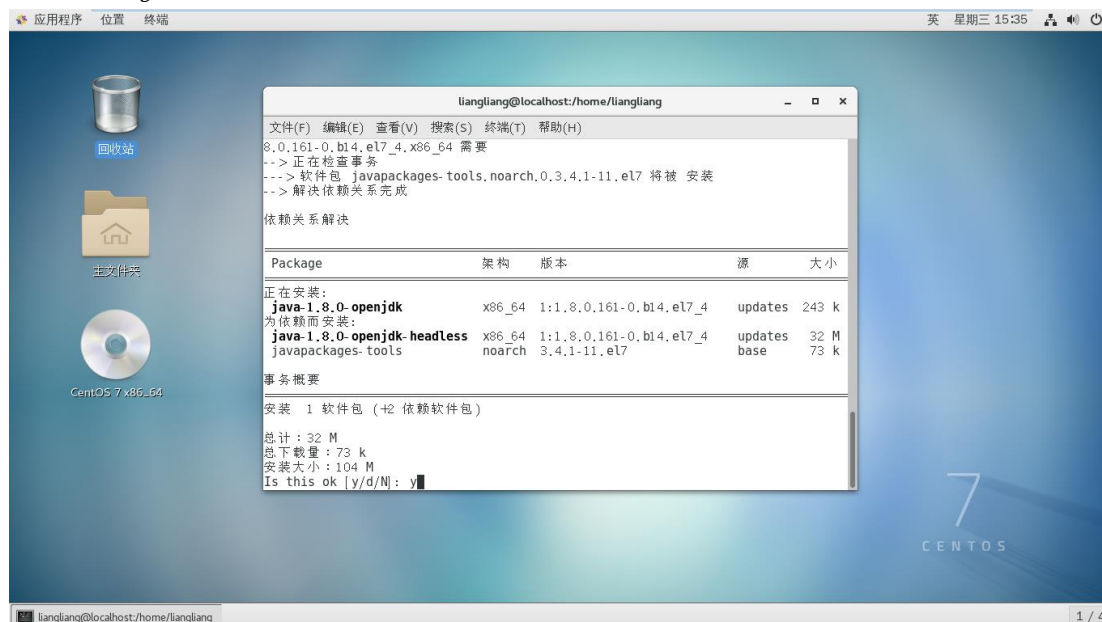


图 5-3 系统安装与部署-配置

### (4) 应用部署

安装 tomcat 服务器

#### 1、搜索 Tomcat 有哪些可安装的版本

```
yum search tomcat
```

2、选择版本进行安装，这里选择 tomcat.noarch:

```
yum -y install tomcat.noarch
```

3、启动 Tomcat

```
systemctl start tomcat
```

4、设置开机自启:

```
systemctl enable tomcat
```

5、检查状态:

```
systemctl status tomcat -l
```

(5) 项目部署

1、先修改 jdbc.properties 文件的 url 为

```
url=jdbc:mysql://IP:3306/psenglish?useUnicode=yes&characterE
ncoding=utf8
```

2、将数据库文件导入 MySQL 数据库

3、在 intellij idea 将项目打包成 war 文件

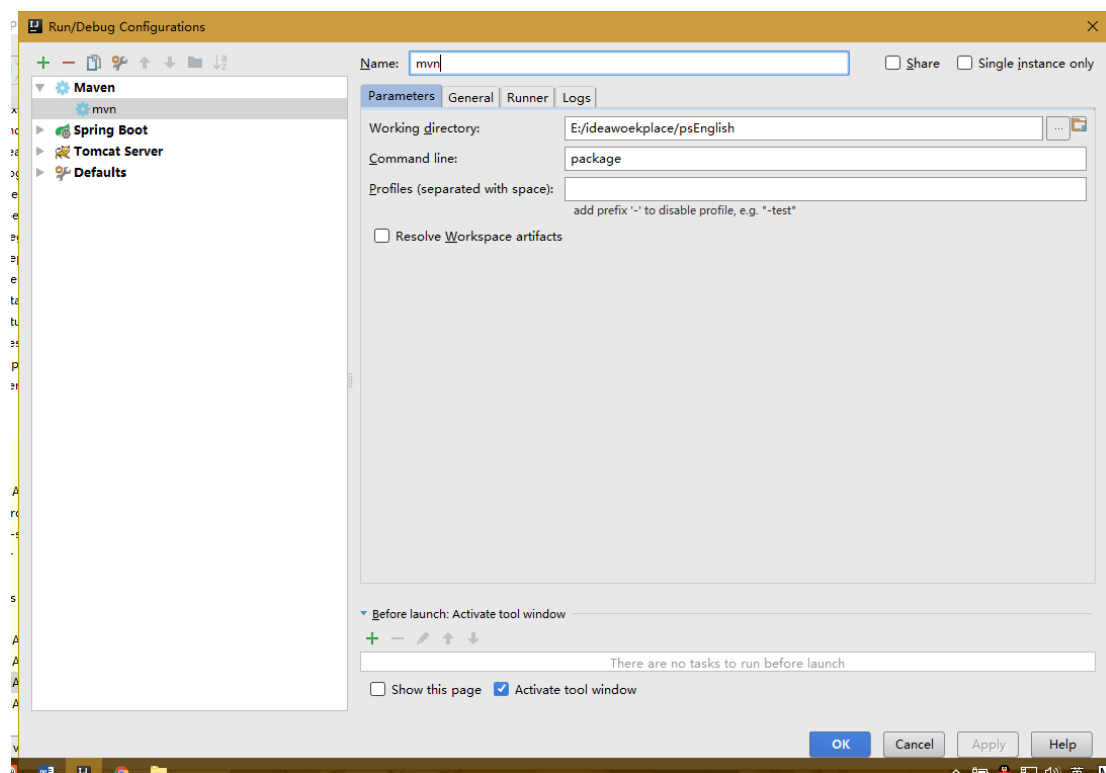


图 5-4 系统安装与部署-配置

4、将打包好的 war 文件上传到 tomcat 的 webapps 的文件里

(1) 先开启 tomcat 服务: `/usr/local/tomcat/bin/startup.sh`

(2) 然后进去 tomcat 的 webapps 目录: `cd /usr/local/tomcat/webapps`

(3) 输入下面命令，选择你刚刚上传的 war 包路径: `rz`

5、使用浏览器访问项目。

<http://IP:8081/psEnglish/index.htm>

## 第6章 结论

通过这次毕业设计的设计和编码，完成了任务的基本要求小学生学习英语网站系统的开发和设计主要内容和英语网站的后台的制作。网站主要面相的对象是小学生，帮助辅导他们学习英语，后台可以快速的添加每个年级的单词，语法，英语课文等内容。这样更方便的可以对数据进行操作。更便捷友好的操作数据，方便网站使用者可以快速的进行发布数据。

(1) 功能:用户管理，学习资源管理，数据进行合理的验证。

(2) 优点:在后台数据的操作管理上，可以更方便快捷安全的操作。

(3) 缺点:在用户体验方面上还存在着不足，还不能很满足很多人的需求。

(4) 解决办法:这在后期我会做个用户体验调查，根据他们的建议，进一步优化网站的体验度。在后台数据的操作管理上，可以更方便快捷安全的操作。进一步将网站的安全性做好，加强对数据的审核，做到对用户的体验满意度更高。未来，小学生学习英语网站系统会做出功能丰富，内容多，学习质量好的英语类型的网站系统。



## 参 考 文 献

- [1] 李立功,赵扬. MySQL 程序设计与数据库管理[M]. 北京:科学出版社 2001
- [2] 王保罗. Java 面向对象程序设计[M]
- [3] 王志刚,江友华. MySQL 高效编程 [M]. 北京:清华大学出版社 2012. 01
- [4] 缪勇. JSP 网络开发逐步深入[M]. 北京: 清华大学出版社 2010
- [5] 毛平;Bootstrap 方法及其应用[D];湘潭大学;2013 年
- [6] 余芳,林美,夏跃伟. JSP 动态网站开发案例指导 [M]. 北京:电子工业出版社
- [7] 崔松健. 响应式 Web 设计[J]. 信息与电脑(理论版), 2013, (10).
- [8] 欧勤坪, 余建桥. 基于 SpringMVC+ibATIS 框架的生物信息数据库的设计与实现[J]. 西南大学学报(自然科学版), 2008, (11).
- [9] 王星,潘郁. 基于 AJAX 技术的 Web 模型在网站开发中的应用研究[J]. 微计算机信息, 2006, (27):206-207, 241. doi:10.3969/j.issn.1008-0570.2006.27.072.
- [10]谢建华. 基于 Bootstrap 技术的企业网站设计与实现 [J]. 计算机时代, 2017, (8). doi:10.16644/j.cnki.cn33-1094/tp.2017.08.005.
- [11]李夏君. 使用 Bootstrap 实现响应式布局[M]. 信息与电脑, 计算机工程应用技术. tp393.092.  
广东省轻工职业技术学校, 广东 广州, 510300;2017.08.25
- [12]林子禹等. 基于 web 与组件技术的企业应用系统设计模型[J]. 计算机工程与应用, 2000:6

## 致 谢

在这次的网站系统的开发，从设计到开发，遇到很多的问题，从一开始的需求开始不明确，在林老师的指导下，一步一步的明确了网站系统开发的步骤。需求是关键，有了好的设计，才会做出优秀的系统，真正的把知识运用到生活中。感谢三年大学里的老师，是您们教会了我如何学习新的知识，如何操作软件，利用新的技术去开发软件。帮助我一步一步的学习，在这个互联网行业，动手实践检验自己的标准，只有自己去操作了，才会发觉自己哪里不足。当然，基础是最重要的，学好某一门语言都要从基础开始，有了扎实的基础，后面就好学多了。再次感谢在大学里的老师和同学们，帮助我学到了很多，不管是学习上的，还是生活上的，学无止境。