

背单词网站

KOWORD

设计文档

Date : 2018-06-20

目录

一、引言.....	4
1.1 编写目的.....	4
1.2 项目背景.....	4
二、总体设计.....	5
2.1 需求规定.....	5
2.1.1 基础用户信息模块——users:	5
2.1.2 英语单词书学习模块——books:	5
2.1.3 英语单词学习模块——learn:	6
2.1.4 英语单词复习模块——review:	6
2.1.5 英语单词考核模块——exam:	7
2.2 运行环境.....	8
硬件客户端.....	8
软件服务器端:	8
软件客户端.....	8
三、CRC Cards.....	8
3.1 基础用户信息模块——users	8
3.2 英语单词书学习模块——books:	9
3.3 英语单词学习模块——learn:	9
3.4 英语单词复习模块——Review:	9
3.5 英语单词考核模块——Exam:	10
四、接口设计.....	11
4.1 用户接口.....	11
4.1.1 登录界面.....	11
4.1.2 注册界面.....	11
4.1.3 英语单词书介绍界面.....	12
4.1.4 英语单词学习界面.....	12
4.1.5 英语单词学习记录界面.....	13
4.1.6 英语单词复习界面.....	13
4.1.7 英语单词复习记录界面.....	14
4.1.8 英语单词考核界面.....	14
4.2 外部接口.....	14
4.2.1 软件接口.....	14
4.2.2 硬件接口.....	14
4.3 内部接口.....	15
五、运行设计.....	15
5.1 运行模块组合.....	15
六、功能设计.....	15
6.1 基础用户信息模块——users:	15
6.1.1 学生注册:	15
6.1.2 学生登录:	19

6.1.3 单词书介绍——books:	20
6.1.4 用户单词学习界面——learn:	20
6.1.5 用户单词学习记录界面——Learn Record:	22
6.1.6 用户单词复习界面——Review:	22
6.1.7 用户单词复习记录界面——Review Record:	24
6.1.8 用户单词考核界面——Exam:	24
七、系统数据结构设计	26
7.1 逻辑结构设计要点	26
7.2 物理结构设计要点	27
10.3 后端系统设计	29
八、小结	31

一、引言

1.1 编写目的

现代用户对于英语单词学习的需求是非常高的，尤其是学校里数以千计的学生都需要英语单词进行学习。对于这种需求，采用在线单词学习网站是最为合适的。用户可以很方便地对自己的学习进度和学习效果进行管理，用户可以更加方便的进行自定义。这样数量庞大的用户是非常非常多的，所以，在线单词学习网站显得尤为重要。

为了满足用户需求和提高用户体验，需要开发这款背单词网站。它可以为用户提供便捷、高效并且安全的服务，可以满足高效的需求。本说明书的预期读者包括：

- 用户
- 项目开发人员
- 软件维护人员

1.2 项目背景

- 软件系统名称：背单词网站 KOWORD
- 任务提出者：浙江大学软件工程任课老师：胡晓军
- 开发者：来舒悦
- 用户：英语学习者

二、总体设计

2.1 需求规定

实现背单词网站的相关业务逻辑。开发用户界面，包括用户注册，登陆，查看单词书，选择单词书学习，查看学习进度，选择单词书复习，查看复习进度，测试。具体针对用户的需求如下：

2.1.1 基础用户信息模块——users:

基础用户信息管理子模块实现背单词网站 KOWORD 的相关业务逻辑。开发用户界面，包括用户客户端注册，登陆。

具体针对用户的需求如下：

- 为用户提供工作界面——新用户注册

新用户需要注册账户方可使用本客户端。注册账号需要提供的信息如下：

用户名：可作为登录账号 需检测是否已经被注册

邮箱：可作为登录账号，需检测是否符合邮箱格式以及是否已经被注册

密码：用户自定义密码，需满足一定的格式要求以保证安全性

确认密码：用户自定义密码，需满足一定的格式要求以保证安全性

- 学生用户登录

用户需要登录账户方可使用本客户端。登录账号需要提供的信息如下：

用户名：可作为登录账号

邮箱：可作为登录账号

密码： 需要与账户密码对应

2.1.2 英语单词书学习模块——books:

- 为用户提供英语单词书介绍界面

用户登录后，可以通过点击左侧 **Books** 按钮查看本网站提供的所有单词书以及对应的信息。

单词书 **Name**：显示网站提供的所有单词书书名

单元 **Units**：显示对应书的单元数量

作者 **Author**：显示对应书的作者名/上传者用户名

单词书信息 Information: 单词书简介

2.1.3 英语单词学习模块——learn:

- 为用户提供英语单词书学习内容选择界面

当用户登录后，系统将提供选择学习和维护自定义单词。用户主要通过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本选择：列表显示所有可选单词书，通过点击右侧 **Learn Now** 按钮完成

单元选择：列表显示所有可选单词书的单元，通过点击对应单元的按钮完成

- 为用户提供英语单词书学习界面

当用户选择完毕后，系统将提供所对应选择学习的单词和解释。用户主要通过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本单元：表头显示当前所学书本与单元

单词 **Word**：列表显示所有当前单元的单词

解释 **Meaning**：列表显示所有当前单元的单词对应的解释

- 为用户提供英语单词书学习记录界面

当用户选择完毕后，用户点击左侧 **Learn Record** 按钮，查看用户对应的学习记录。具体功能如下：

单词书 **Name**：显示网站提供的所有单词书书名

单元 **Units**：显示对应书的单元数量

已学单元 **Learned Units**：显示对应书的已学单元数量

未学单元 **Not Learned Units**：显示对应书的未学单元数量

2.1.4 英语单词复习模块——review:

- 为用户提供英语单词书复习内容选择界面

当用户登录后，系统将提供选择复习和维护自定义单词。用户主要通过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本选择：列表显示所有可选单词书，通过点击右侧 **Review Now** 按钮完成

单元选择：列表显示所有可选单词书的单元，通过点击对应单元的按钮完成

- 为用户提供英语单词书复习界面

当用户选择完毕后，系统将提供所对应选择复习的单词和解释。用户主要通

过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本单元：表头显示当前所学书本与单元

单词 **Word**：列表显示所有当前单元的单词

解释 **Meaning**：列表显示所有当前单元的单词对应的解释

- 为用户提供英语单词书学习复习界面

当用户选择完毕后，用户点击左侧 **Review Record** 按钮，查看用户对应的复习记录。具体功能如下：

单词书 **Name**：显示网站提供的所有单词书书名

单元 **Units**：显示对应书的单元数量

已学单元 **Reviewed Units**：显示对应书的已复习单元数量

未学单元 **Not Reviewed Units**：显示对应书的未复习单元数量

2.1.5 英语单词考核模块——**exam**：

- 为用户提供英语单词书考核内容选择界面

当用户登录后，系统将提供选择考核和维护自定义单词。用户主要通过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本选择：列表显示所有可选单词书，通过点击右侧 **Exam Now** 按钮完成

单元选择：列表显示所有可选单词书的单元，通过点击对应单元的按钮完成

- 为用户提供英语单词书考核界面

当用户选择完毕后，系统将提供所对应选择考核的单词。用户主要通过选择单词书并选择该书本的对应单元来完成。具体功能如下：

书本单元：表头显示当前所学书本与单元

单词 **Word**：列表显示所有当前单元的单词

已知按钮 **Know**：列表显示所有当前单元的单词对应的已知按钮

未知按钮 **Don't Know**：列表显示所有当前单元的单词对应的未知按钮

查看答案按钮 **See Answers Now**：点击后将显示该单元列表单词对应的解释

2.2 运行环境

硬件客户端

- 处理器：主流 Intel 及 AMD 处理器
- 内存：128M 以上
- 网卡：支持浙江大学校园网（L2TP）

软件服务器端：

- 操作系统：Ubuntu16.04
- 数据库软件：Sqlite3
- 系统框架：Django

软件客户端

- 操作系统：Windows、Linux、Mac 等均可
- 浏览器：Chrome、Firefox、Safari 等均可

三、CRC Cards

3.1 基础用户信息模块——users

类：用户账户信息	
说明：	
访问和设置用户账号有关的相关操作	
职责：	协作者：用户
定义了属性：处理用户注册和登录	
检测用户个人信息	
添加用户个人信息	

3.2 英语单词书学习模块——books:

类：单词书（单本）	
说明：	
设置和添加单词书中单词的相关信息操作	
职责：	协作者：管理员
添加单词书的单元有关信息	
获取单词书的单元有关信息	
添加单词书的单词有关信息	
获取单词书的单词有关信息	

类：单词书（所有）	
说明：	
设置和添加单词书有关的操作	
职责：	协作者：用户
添加单词书有关信息	
获取单词书有关数据	

3.3 英语单词学习模块——learn:

类：学习记录	
说明：	
添加和获取单词书中单词的相关信息操作	
职责：	协作者：用户
添加学习记录有关信息	
获取学习记录有关信息	

3.4 英语单词复习模块——Review:

类：复习记录

说明:	
添加和获取单词书中单词的相关信息操作	
职责:	协作者: 用户
添加复习记录有关信息	
获取复习记录有关信息	

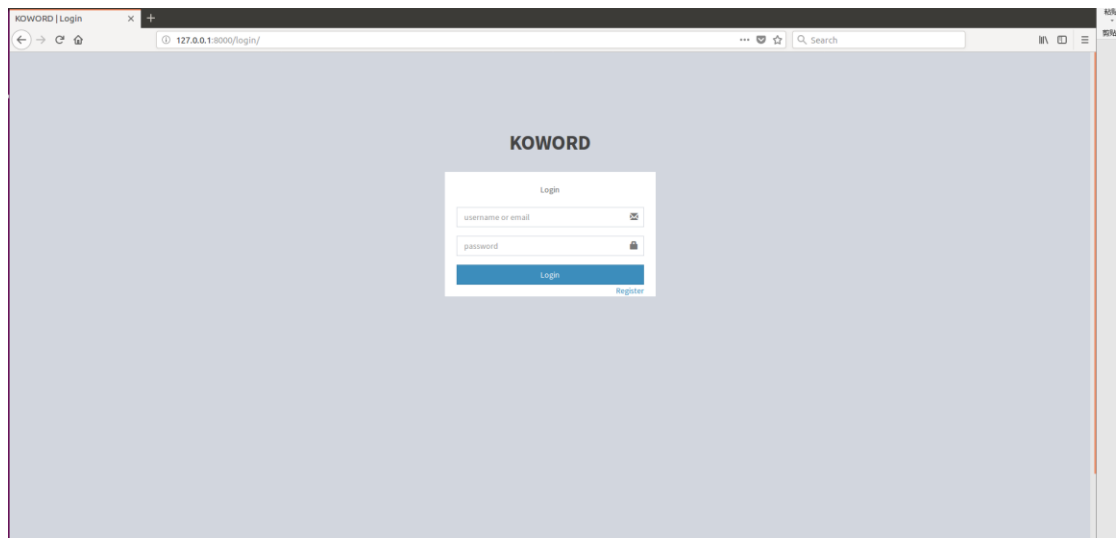
3.5 英语单词考核模块——Exam:

类: 考核	
说明:	
对考核题目进行管理	
职责:	协作者: 用户
选择考核内容并完成考核相关信息的操作	
显示考核内容	
添加考核记录等有关信息	
提供正确答案	

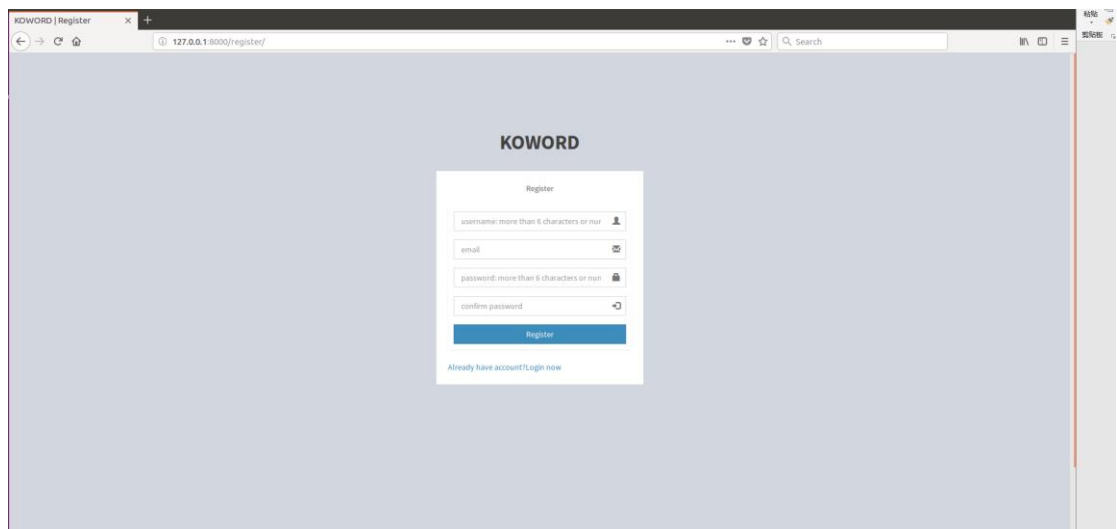
四、接口设计

4.1 用户接口

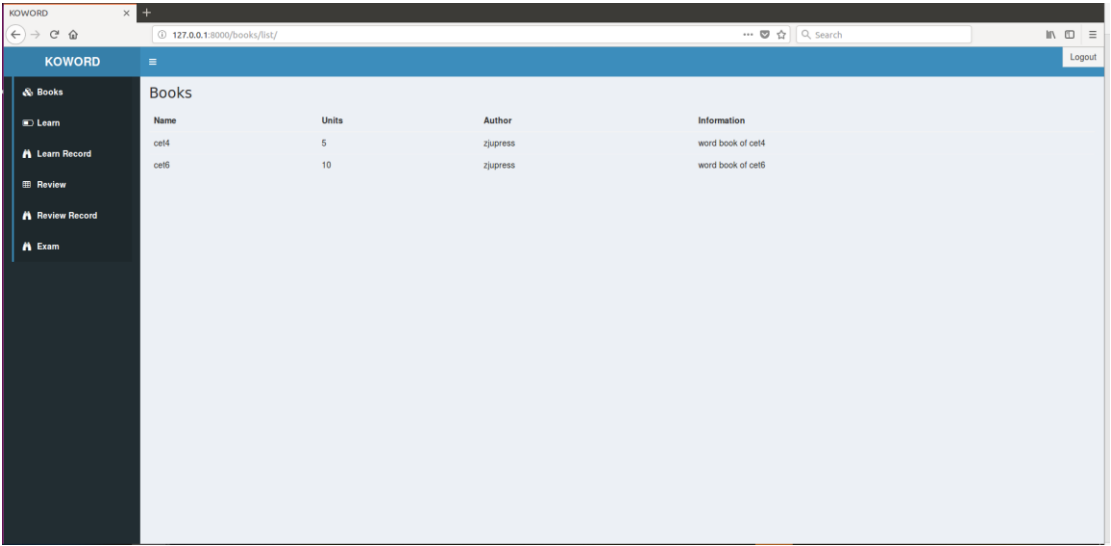
4.1.1 登录界面



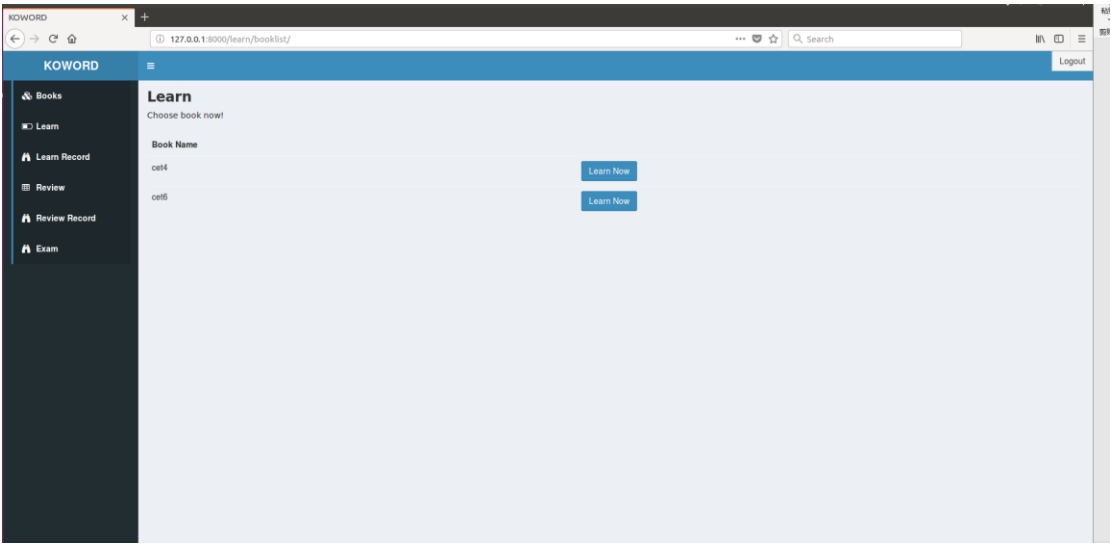
4.1.2 注册界面



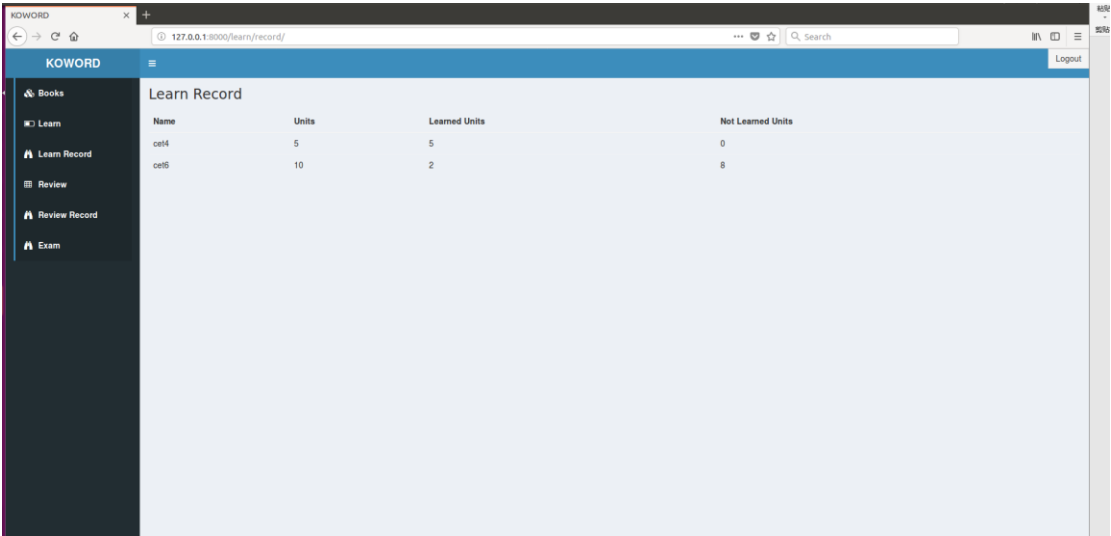
4.1.3 英语单词书介绍界面



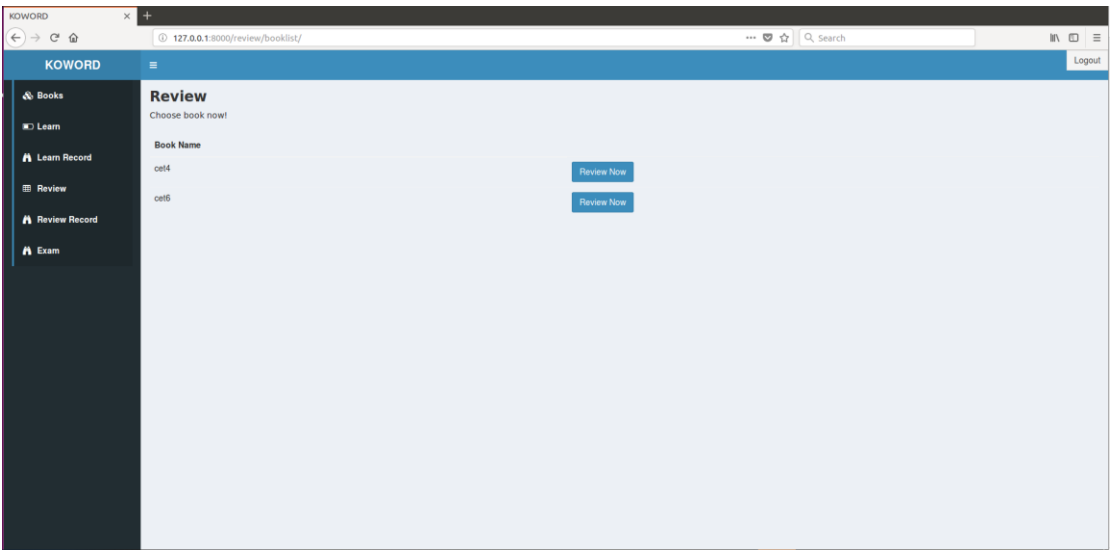
4.1.4 英语单词学习界面



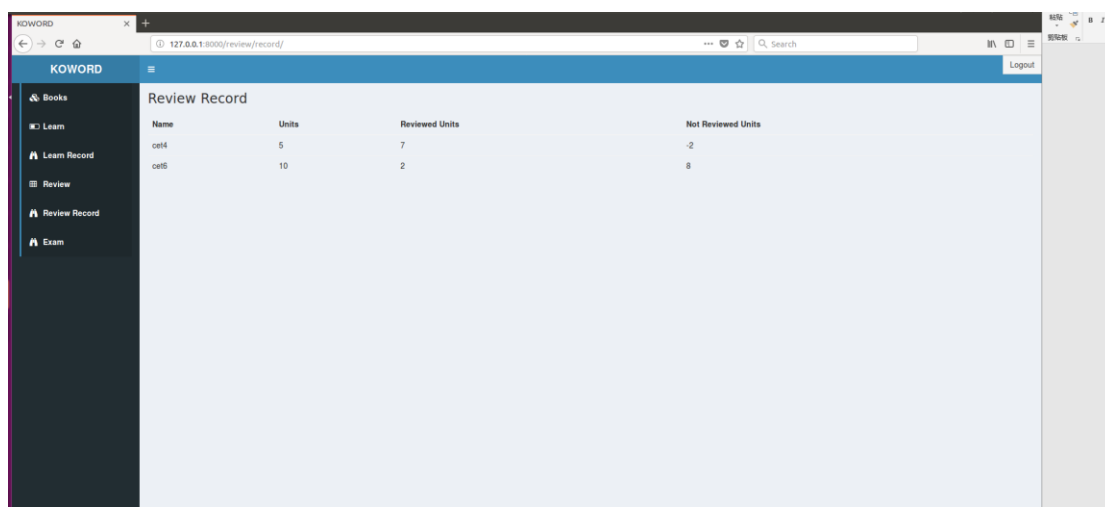
4.1.5 英语单词学习记录界面



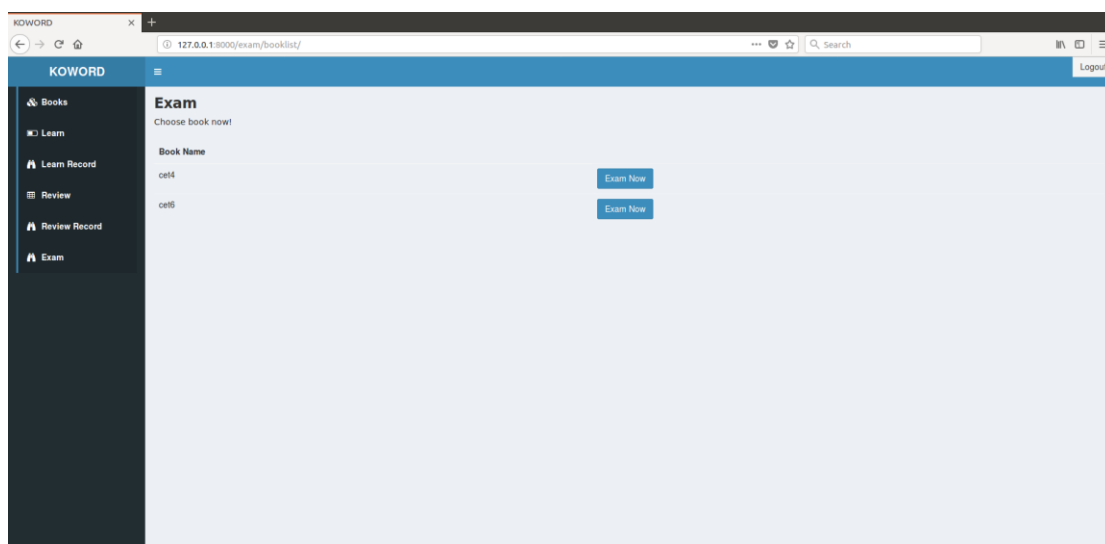
4.1.6 英语单词复习界面



4.1.7 英语单词复习记录界面



4.1.8 英语单词考核界面



4.2 外部接口

4.2.1 软件接口

程序采用 python Django 框架提供的 Sqlite 接口来实现对数据库的访问操作。
在前端数据的提交方面，主要采取 post, get 数据传输。

4.2.2 硬件接口

用户通过使用键盘、鼠标等标准输入设备来完成与浏览器的相关交互，通

过显示器来获取相关应用的输出。

4.3 内部接口

各子系统之间互相调用。

五、运行设计

5.1 运行模块组合

本网站主要是在 Sqlite3 数据库服务中运行。运行过程中，将以 Django 作为后端服务和控制程序，根据请求返回对应数据控制前端 Html 输出，并且根据请求控制底层数据库操作。

数据库主要是用于记录用户账号信息、学习记录等信息。

六、功能设计

实现背单词网站的相关操作，对于用户开发出最便捷的使用界面。

用户可以注册登录，学习英语单词并查看自己的学习记录，复习英语单词并查看自己的复习记录并考核自己的学习和复习情况。


6.1 基础用户信息模块——users:


6.1.1 学生注册:


新用户需要注册账户方可使用本背单词网站。用户可自定义用户名，输入有效邮箱信息以及自定义密码并满足一定的格式要求后可完成新用户注册。


KOWORD

Register

username: more than 6 characters or nur 

email 

password: more than 6 characters or nun 

confirm password 


Register

[Already have account?Login now](#)


(1) 用户名需要满足大于 6 位，否则将报错。

KOWORD


Register

1 


- Ensure this value has at least 6 characters (it has 1).

email 

- This field is required.

password: more than 6 characters or nun 

- This field is required.

confirm password 

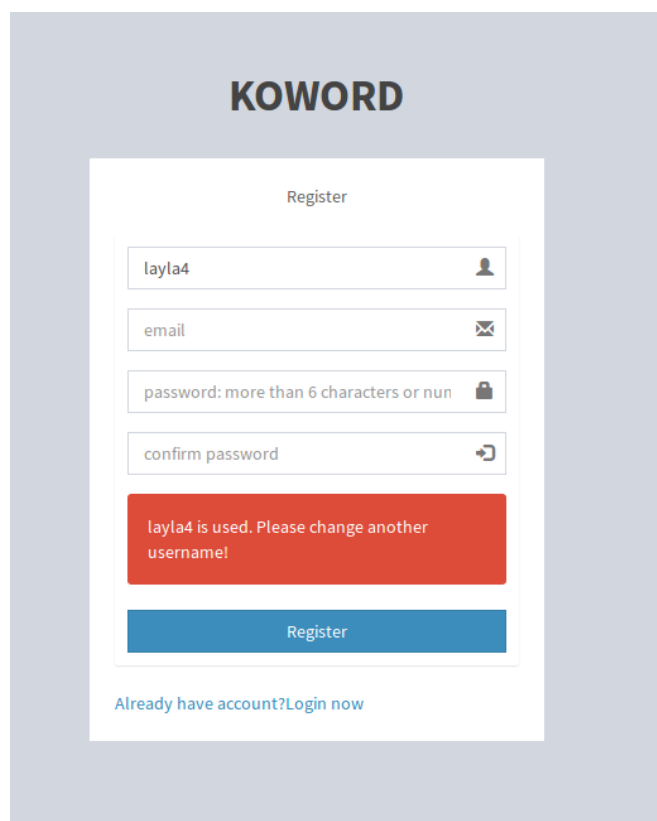
- This field is required.

Fail to register!

Register

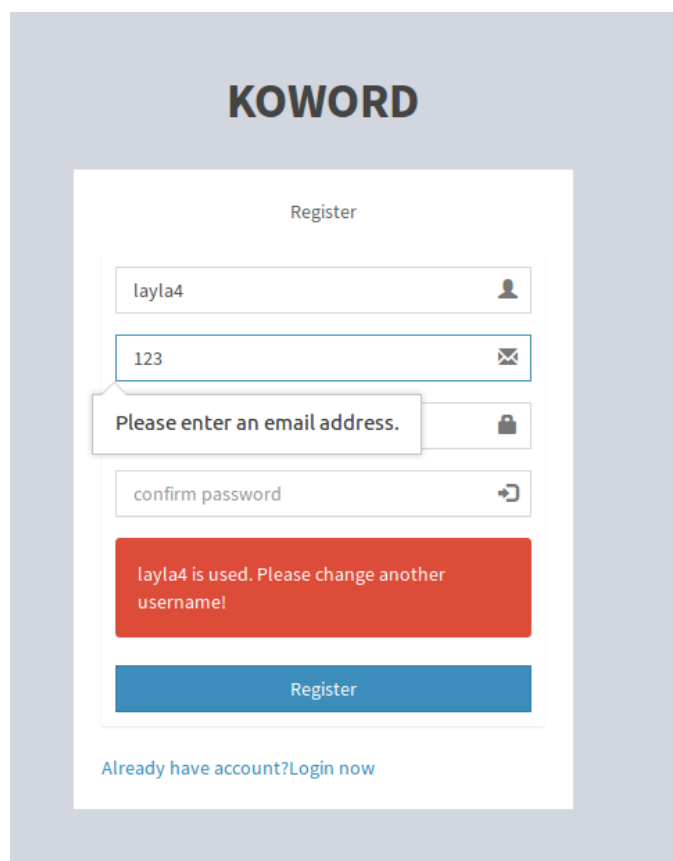
[Already have account?Login now](#)

(2) 用户名如果已经被注册，将报错。



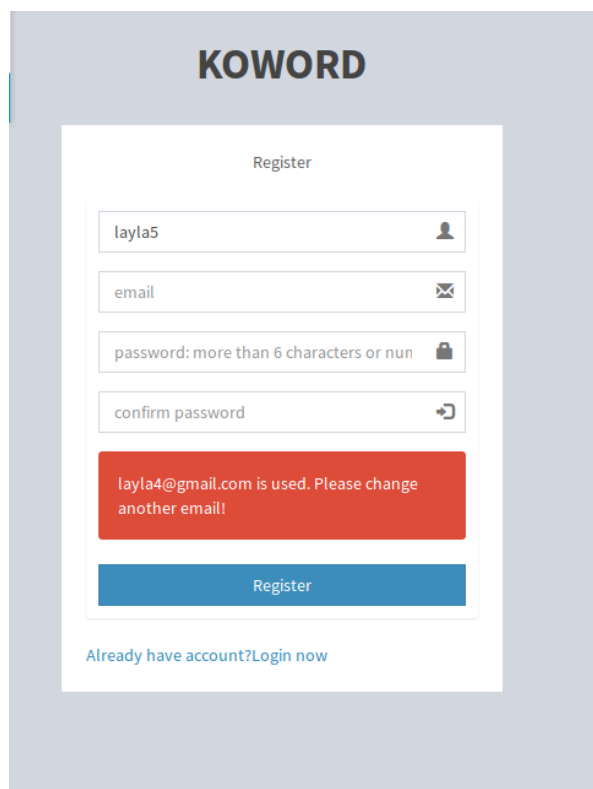
The image shows a web registration form titled "KOWORD" with a sub-header "Register". It contains four input fields: "username" (with a person icon), "email" (with an envelope icon), "password" (with a lock icon and a hint "password: more than 6 characters or num"), and "confirm password" (with a refresh icon). A red error message box states "layla4 is used. Please change another username!". Below the fields is a blue "Register" button and a link "Already have account? Login now".

(3) 邮箱输入需要满足邮箱格式，否则将报错。



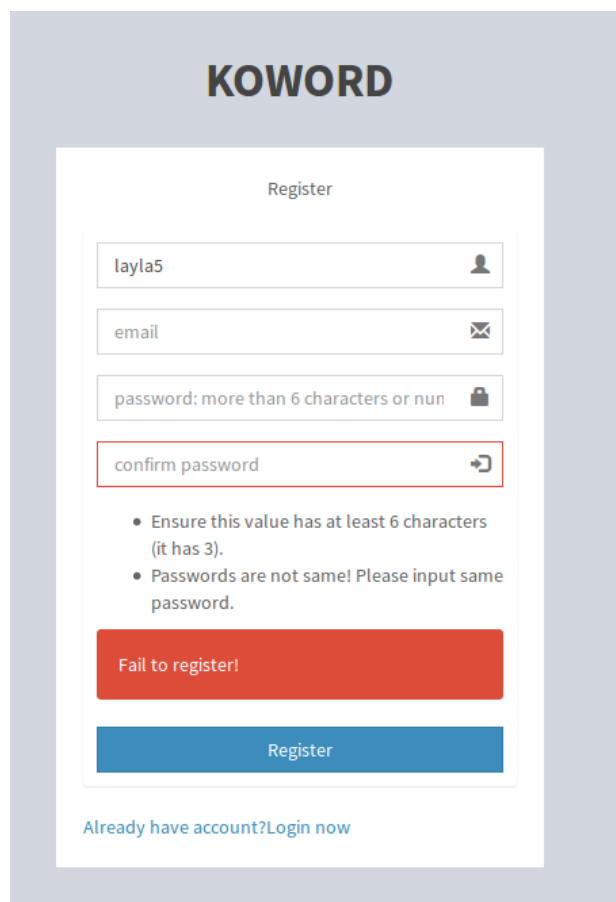
The image shows the same "KOWORD" registration form, but the "email" field now contains "123". A tooltip message "Please enter an email address." is displayed over the email field. The "password" field is empty. The "confirm password" field is empty. The red error message box "layla4 is used. Please change another username!" is still present. The blue "Register" button and the "Already have account? Login now" link are also visible.

(4) 邮箱如果已经被注册，将报错。



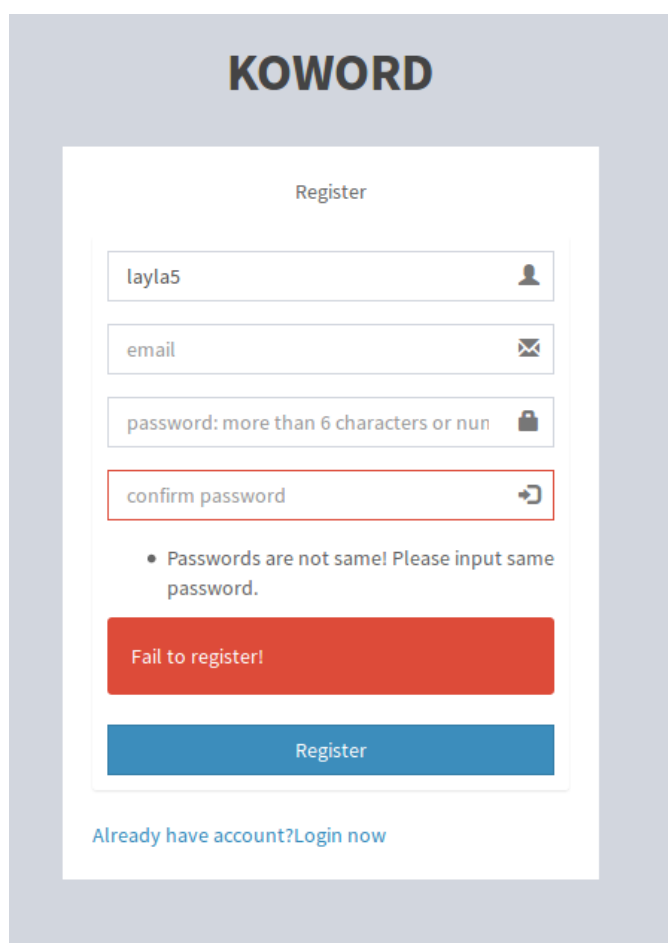
The image shows a web registration form for 'KOWORD'. The form is titled 'Register' and contains four input fields: a username field with 'layla5', an email field with 'email', a password field with the placeholder 'password: more than 6 characters or nun', and a confirm password field. A red error message box is displayed below the email field, stating 'layla4@gmail.com is used. Please change another email!'. Below the error message is a blue 'Register' button. At the bottom of the form, there is a link that says 'Already have account? Login now'.

(5) 密码输入需要符合要求，否则将报错。



The image shows the same 'KOWORD Register' form as in the previous screenshot. In this case, the 'confirm password' field is highlighted with a red border. Below this field, there are two bullet points indicating password requirements: 'Ensure this value has at least 6 characters (it has 3).' and 'Passwords are not same! Please input same password.'. A red error message box is displayed below the bullet points, stating 'Fail to register!'. The 'Register' button is still present at the bottom of the form, along with the 'Already have account? Login now' link.

(6) 两次密码输入需要相同，否则将报错。

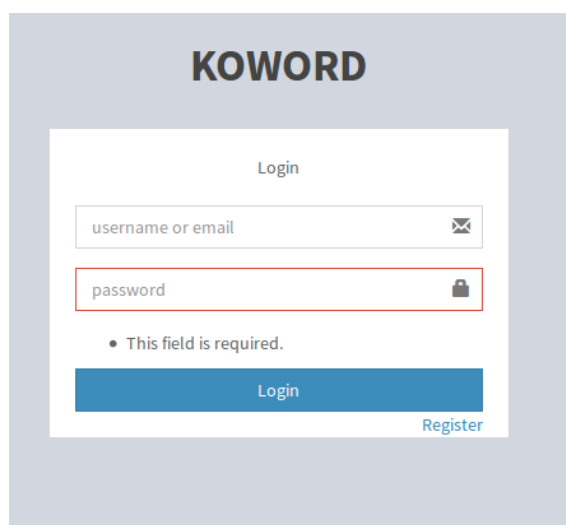


The image shows a web form titled "KOWORD" with a "Register" section. It contains four input fields: a username field with "layla5", an email field, a password field with a hint "password: more than 6 characters or nun", and a confirm password field. A red border highlights the confirm password field. Below the fields, a message states: "• Passwords are not same! Please input same password." A red banner below this message says "Fail to register!". At the bottom is a blue "Register" button and a link "Already have account? Login now".

6.1.2 学生登录:

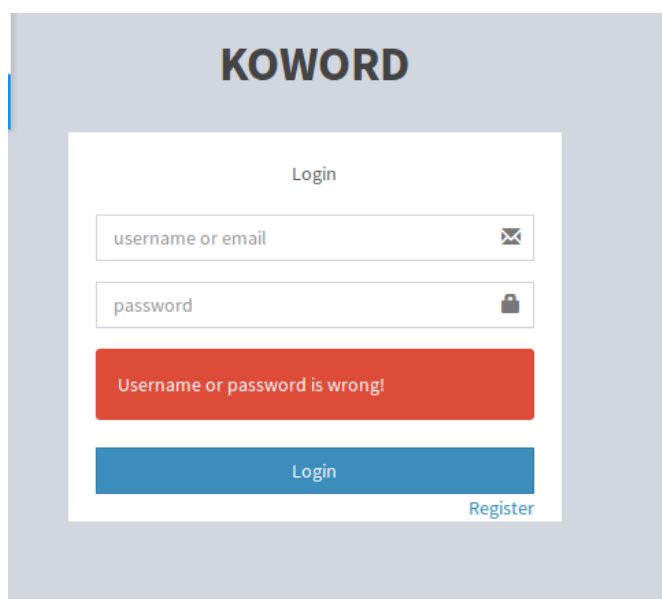
新用户需要登录账户方可使用本背单词网站，本网站支持用户名/邮箱登录。

(1) 需要填入用户名/邮箱作为账号以及密码，否则将报错。



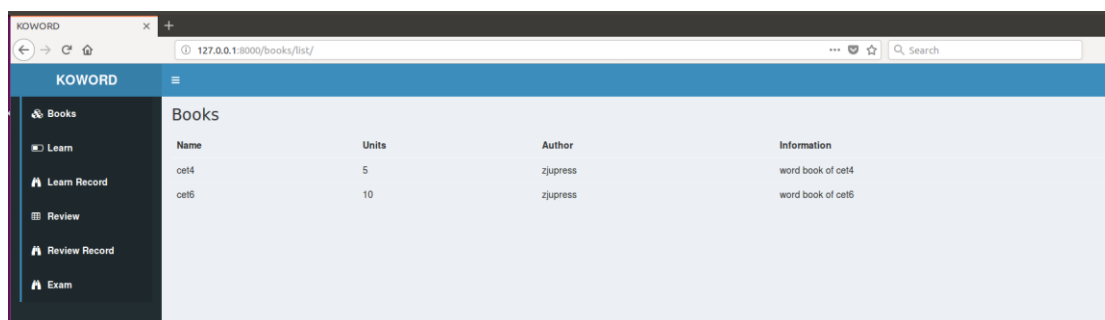
The image shows a web form titled "KOWORD" with a "Login" section. It contains two input fields: a "username or email" field and a "password" field. A red border highlights the password field. Below the fields, a message states: "• This field is required." At the bottom is a blue "Login" button and a link "Register".

(2) 输入的用户名/邮箱和密码需要匹配，否则将报错。



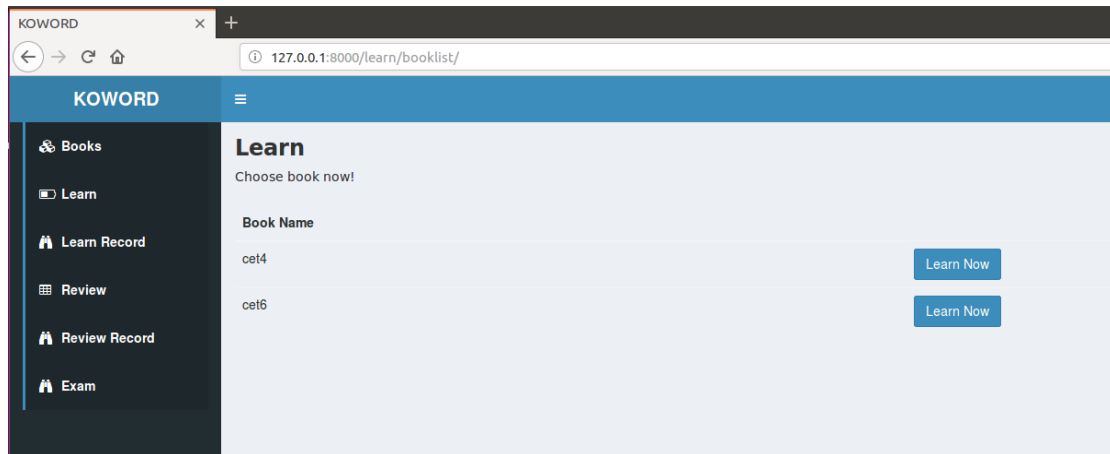
6.1.3 单词书介绍——books:

用户登录后点击左侧 Books 按钮即可查看本网站所有支持的单词书以及其相关信息。在本系统中会自动读取数据库的相关信息，并根据信息排列。

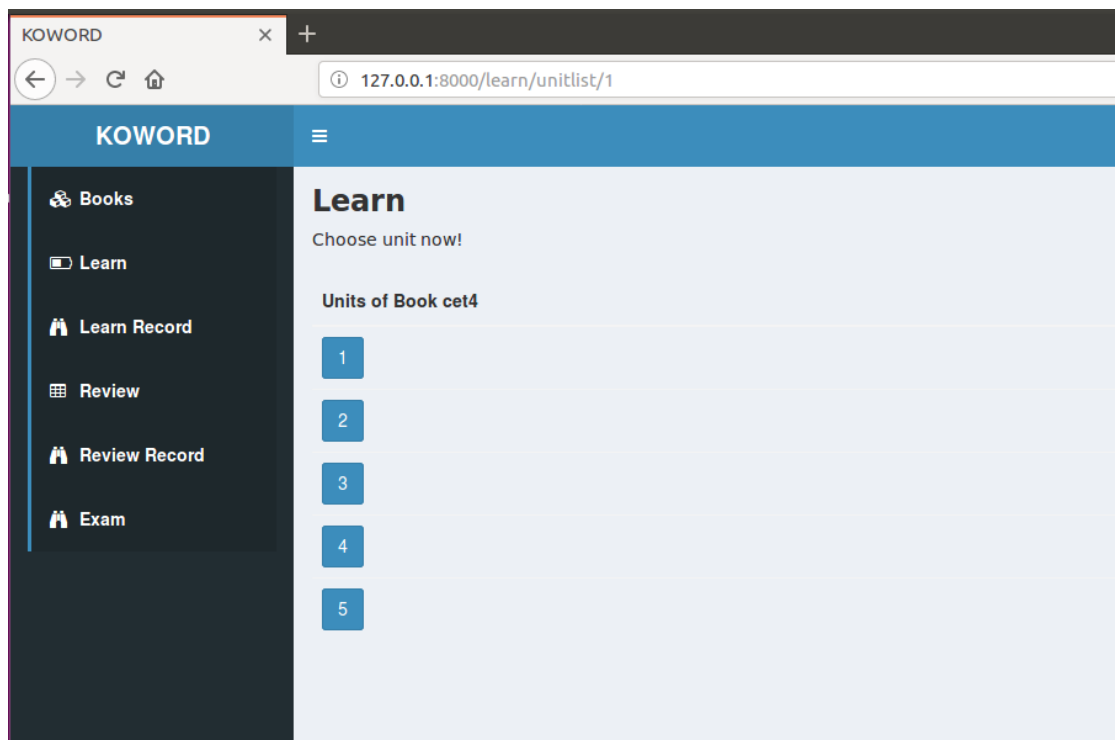


6.1.4 用户单词学习界面——learn:

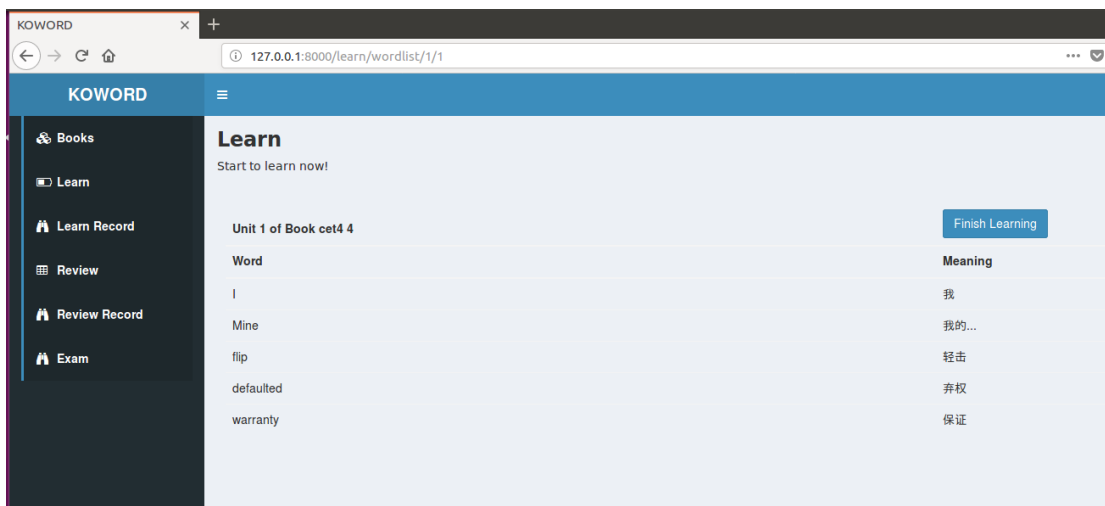
(1) 用户登录后点击左侧 Learn 按钮即可查看本网站所有支持的单词书并用户可以进行选择。



(2) 用户登录后点击希望学习的书本右侧的 Learn Now 按钮即可查看本单词书所有支持的单元并用户可以进行选择。



(3) 用户登录后点击希望学习的单元按钮即可查看本单元所有单词并用户可以进行选择。



(4) 用户学习完成后点击右上侧的 Finish Learning 按钮即可返回学习主页面重新开始选择继续学习的书本。

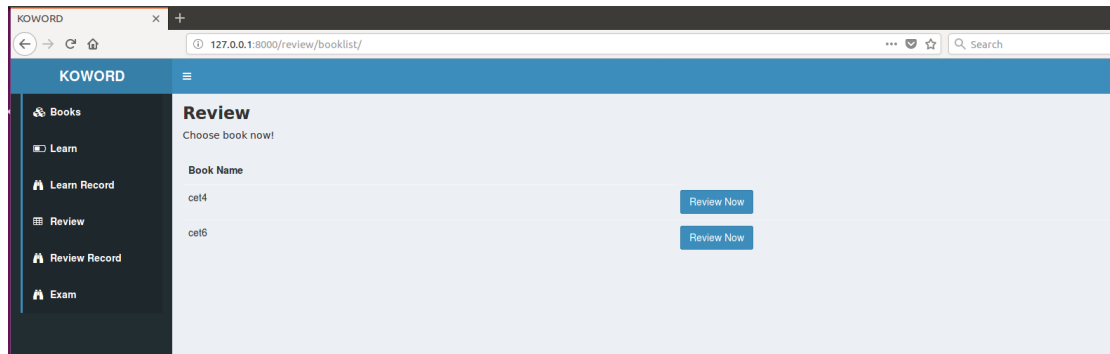
6.1.5 用户单词学习记录界面——Learn Record:

(1) 用户登录后点击左侧 Learn Record 按钮即可查看本人每本单词书的学习情况。

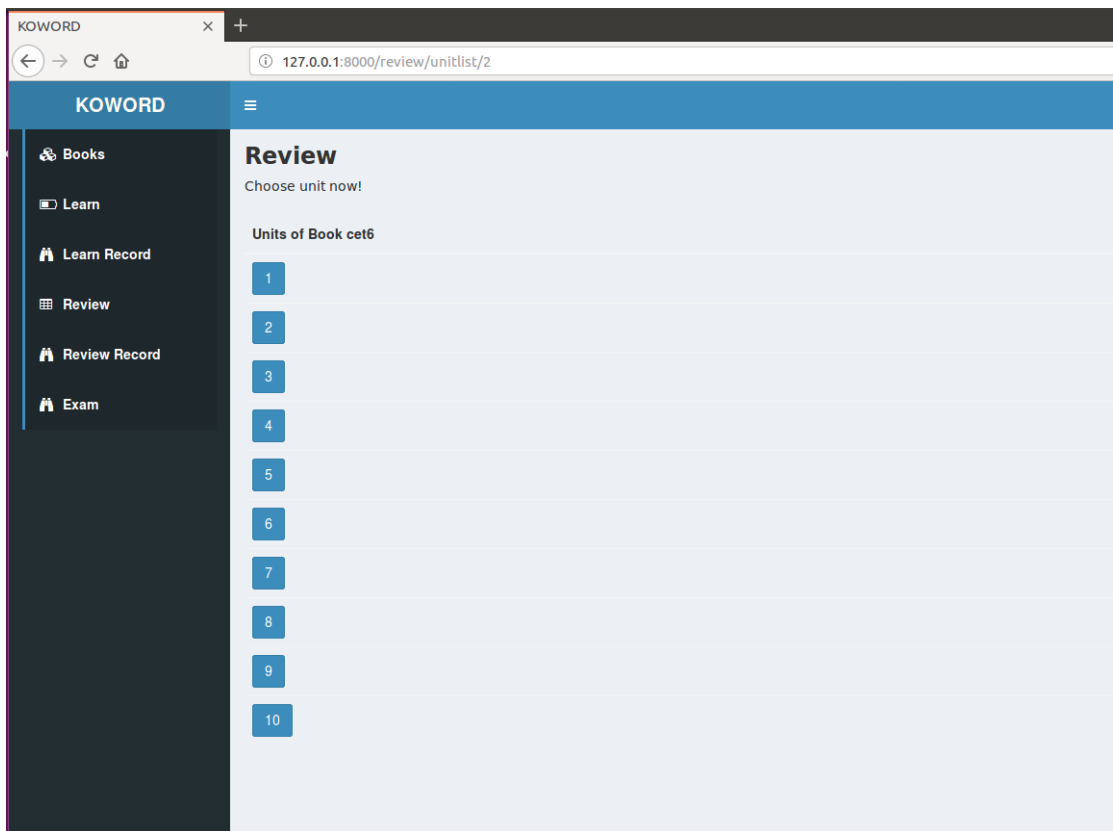
Name	Units	Learned Units	Not Learned Units
cet4	5	5	0
cet6	10	2	8

6.1.6 用户单词复习界面——Review:

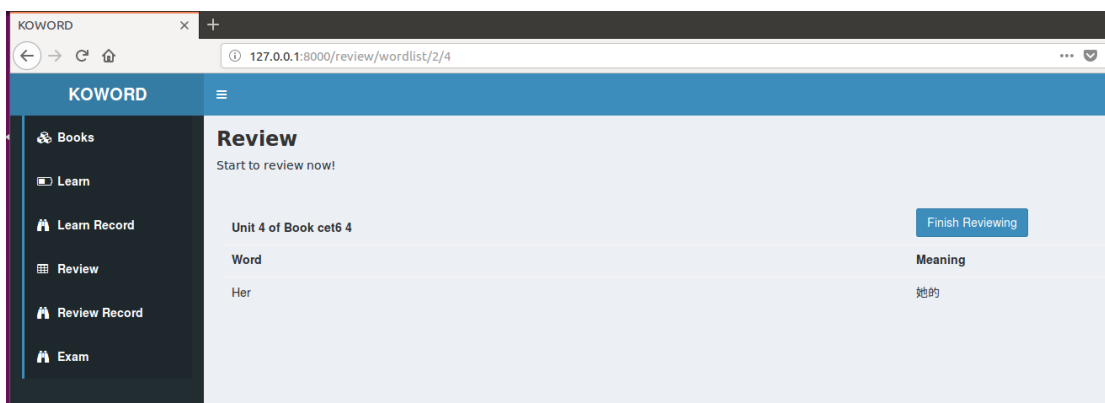
(1) 用户登录后点击左侧 Review 按钮即可查看本网站所有支持的单词书并用户可以进行选择。



(2) 用户登录后点击希望学习的书本右侧的 Review Now 按钮即可查看本单词书所有支持的单元并用户可以进行选择。



(3) 用户登录后点击希望复习的单元按钮即可查看本单元所有单词并用户可以进行选择。



(4) 用户学习完成后点击右上侧的 Finish Reviewing 按钮即可返回复习主页面重新开始选择继续学习的书本。

6.1.7 用户单词复习记录界面——Review Record:

用户登录后点击左侧 Learn Record 按钮即可查看本人每本单词书的学习情况。

Name	Units	Reviewed Units	Not Reviewed Units
cet4	5	1	4
cet6	10	1	9

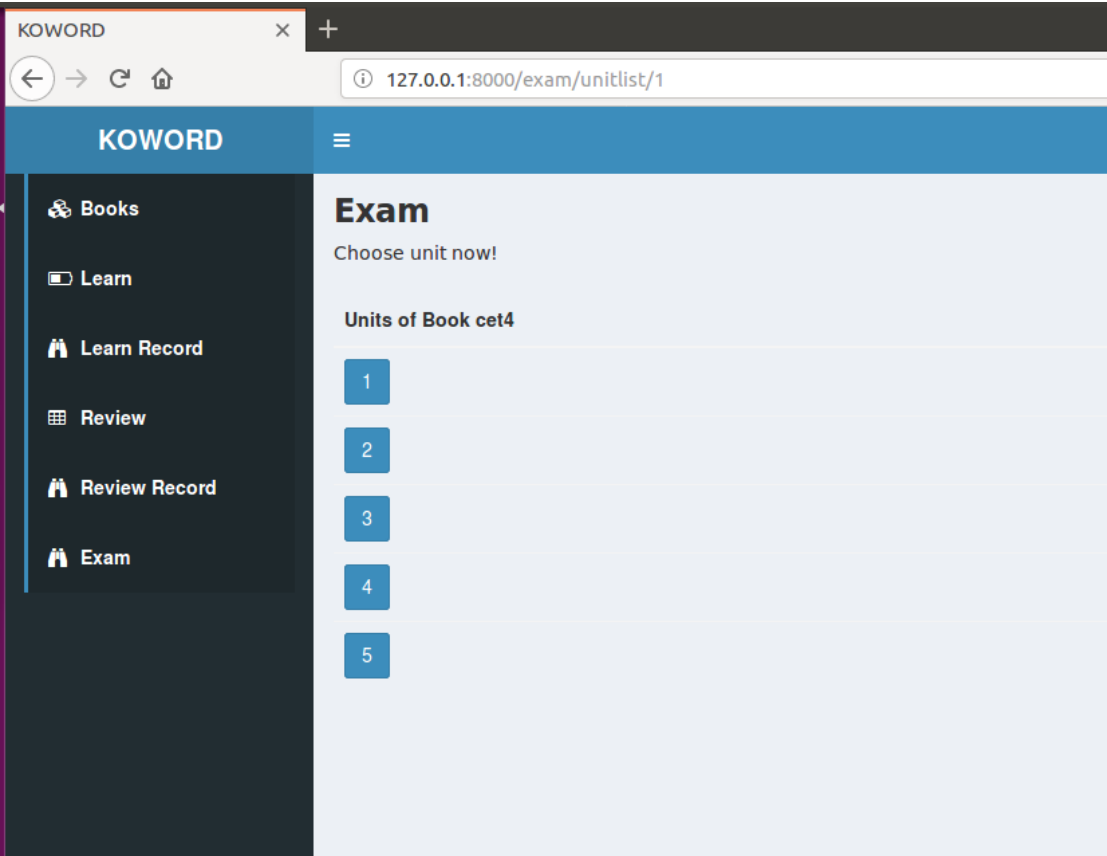
6.1.8 用户单词考核界面——Exam:

(1) 用户登录后点击左侧 Exam 按钮即可查看本网站所有支持的单词书并用户可以进行选择。

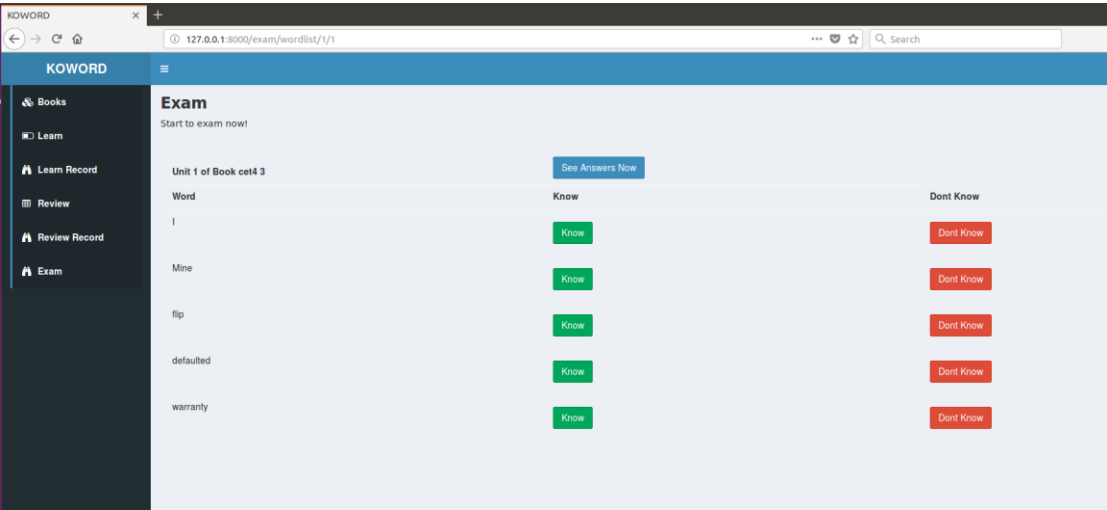
Book Name	Exam Now
cet4	Exam Now
cet6	Exam Now

(2) 用户登录后点击希望学习的书本右侧的 Exam Now 按钮即可查看本单词

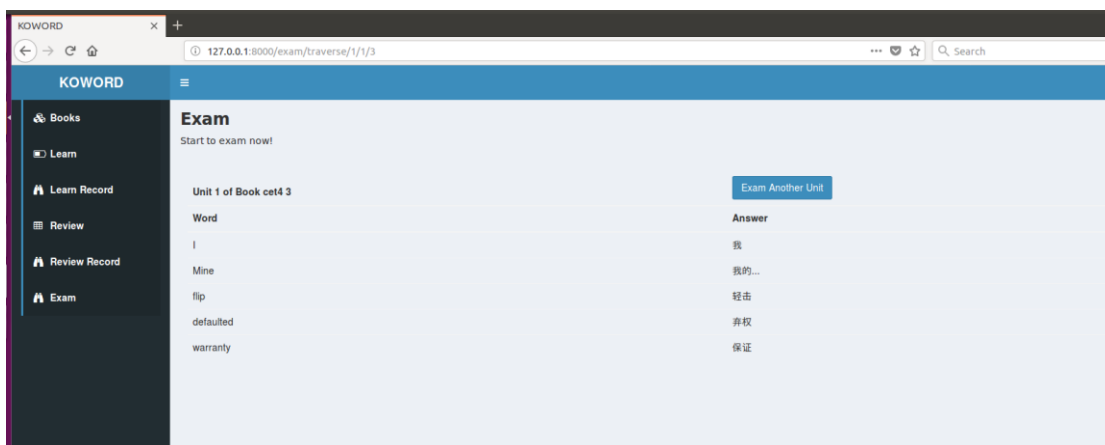
书所有支持的单元并用户可以进行选择。



(3) 用户登录后点击考的单元按钮即可查看本单元所有单词并用户可以
进行考核。其中每个单词的右侧都有两个对应的按钮，分别为已知和未知。



(4) 用户学习完成后点击右上侧的 See Answers Now 按钮即查看本单元所有
单词的解释。



七、系统数据结构设计

7.1 逻辑结构设计要点

设计的部分数据库的数据结构如下：

7.1.1 基础用户信息管理 users_userprofile

域名	注释	类型
id	用户 id（主键）	autoID
username	用户账户（主键）	varchar
email	邮箱	varchar
password	用户密码	varchar

7.1.2 单词书信息管理 books_list

域名	注释	类型
id	单词书 id（主键）	autoID
name	书名	varchar
unit	单元数	Integer
author	作者/上传者	varchar
Info	单词书简介	varchar

7.1.3 单词表 learn_word

域名	注释	类型
id	单词 id（主键）	autoID
Bookname	书名	varchar
unit	单元数	integer
word	单词	varchar
meaning	含义	varchar

7.1.4 学习记录 learn_record

域名	注释	类型
userid	用户 id	int
bookid	书本 id	int
unit	单元数	int
islearned	是否该单词已经学习	int
learntime	学习时间	Datetime

7.1.5 复习记录 review_record

域名	注释	类型
userid	用户 id	int
bookid	书本 id	int
unit	单元数	int
isreviewed	是否该单词已经复习	int
reviewtime	复习时间	Datetime

7.2 物理结构设计要点

物理结构设计为 models.py 中的类定义，与数据库表格的定义基本一致。

7.2.1 基础用户信息管理 users_userprofile

```

class UserProfile(AbstractUser):
    class Meta:
        verbose_name = "UserProfile"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.__unicode__()

    def __unicode__(self):
        return self.username

```

7.2.2 单词书信息管理 books_list

```

class List(models.Model):
    id = models.IntegerField(verbose_name='id', primary_key=True)
    name = models.CharField(blank=True, max_length=200, verbose_name="name")
    unit = models.IntegerField(verbose_name='unit')
    author = models.CharField(blank=True, max_length=200, verbose_name="author")
    info = models.CharField(blank=True, max_length=200, verbose_name="info")

    class Meta:
        verbose_name = "books_list"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.__unicode__()

    def __unicode__(self):
        return self.info

```

7.2.3 单词表 learn_word

```

class Word(models.Model):
    id = models.IntegerField(blank=True, primary_key=True, verbose_name="id")
    bookname = models.CharField(blank=True, max_length=50, verbose_name="bookname")
    unit = models.IntegerField(verbose_name='unit')
    word = models.CharField(blank=True, max_length=50, verbose_name="word")
    meaning = models.CharField(blank=True, max_length=50, verbose_name="meaning")

    class Meta:
        verbose_name = "learn_word"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.__unicode__()

    def __unicode__(self):
        return self.word

```

7.2.4 学习记录 learn_record

```
class Record(models.Model):
    userid = models.IntegerField(blank=True, verbose_name="userid")
    bookid = models.IntegerField(blank=True, verbose_name="bookid")
    unit = models.IntegerField(blank=True, verbose_name="unit")
    islearned = models.IntegerField(verbose_name="islearned", default=0)
    leartime = models.DateTimeField(default=datetime.now, verbose_name="l

    class Meta:
        verbose_name = "learn_record"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.__unicode__()

    def __unicode__(self):
        return self.id
```

7.2.5 复习记录 review_record

```
class Record(models.Model):
    userid = models.IntegerField(blank=True, verbose_name="userid")
    bookid = models.IntegerField(blank=True, verbose_name="bookid")
    unit = models.IntegerField(blank=True, verbose_name="unit")
    isreviewed = models.IntegerField(verbose_name="isreviewed", default=0)
    reviewtime = models.DateTimeField(default=datetime.now, verbose_name="

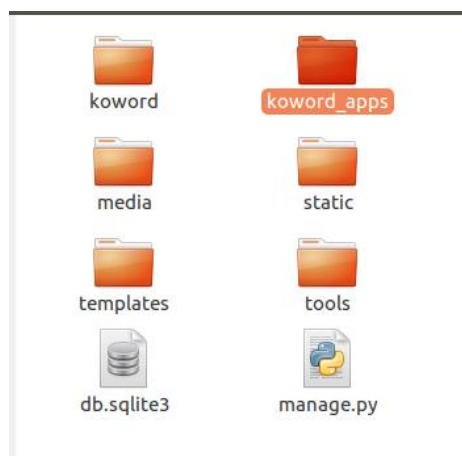
    class Meta:
        verbose_name = "review_record"
        verbose_name_plural = verbose_name

    def __str__(self):
        return self.__unicode__()

    def __unicode__(self):
        return self.id
```

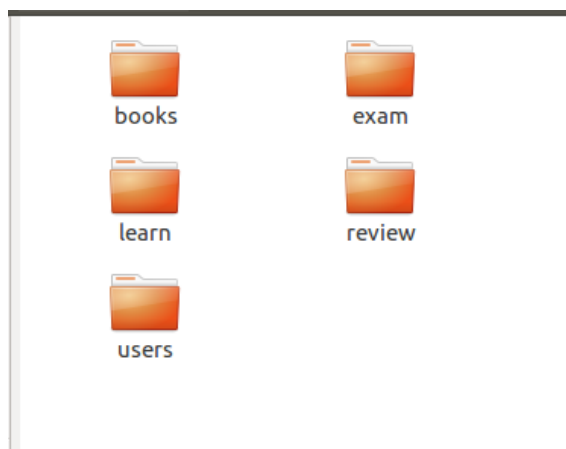
10.3 后端系统设计

后端使用了 django 框架：

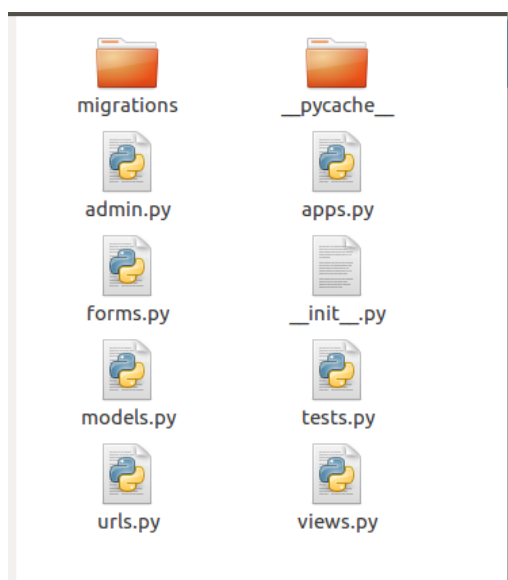


- db.sqlite3: 数据库，存储了所有上述表格与数据
- manage.py: 执行文件

- koword: 主框架，包含了 root_url 文件以及工程的 settings.py 文件
- koword_apps: 包含 5 个子模块



App 实现（以 user 为例）：



- urls.py: 包含所有本 app 中的 url 链接
- models.py: 与数据库连接从而进行相关操作
- views.py: web 前端逻辑实现
- forms.py: 检验 web 端输入是否符合规则

流程:

通过当前的 web url 在 urls.py 中找到对应的 views.py 中的 view，从而获得前端操作的逻辑以及静态 html 文件；web 端发生任何操作通过 views.py 来实现。如果有输入可以先通过 form.py 来检查输入的正确性，如果正确则会通过 models.py 来与数据库进行交互。

八、小结

这是我第一次独立完成整个网站的搭建，因此在前期的学习中遇到了较多的困难。我最终确定了使用 **django** 的框架来完成背单词网站 **KOWORD** 的设计。然后通过学习简单的 **demo** 与教程来了解整个网站搭建的前后端流程和技巧，并以此为基础完成了本项目。虽然在整个项目的实现过程中遇到了很多挑战，但最终 **KOWORD** 的实现让我对 **BS** 有了更加深入的了解和理解。