《数据库系统原理》课程设计

系统实现报告

题目名称：　libilibi 电影论坛网站

学号及姓名：__15231100__曹强港__

__15231089__王冠楠__

2017 年　　12 月　　22 日

一. 系统功能需求分析

一. 用户相关功能

1. 登录：在登录界面输入用户名和密码，在数据库中完成查询，如果存在该用户，则登陆成功，进入到网站的主界面；如果不存在，则登录失败，提示用户名/密码错误。
2. 注册：前端输入用户名，昵称，密码，确认密码，邮箱，手机号。如果用户名可用，密码与确认密码相同，则注册成功，提示注册成功，并跳转到登录界面。用户名已存在，或者密码与确认密码不同则提示错误，重新填表。
3. 注销：当用户注销后，跳转到登录界面，注销登录状态。
4. 查询个人信息：在主界面中可以进入个人资料页面，显示用户当前的资料，包括有用户的 id，用户的用户名，用户的昵称，用户的电话，用户的邮箱。
5. 更改个人信息：在个人资料页面，允许用户更改一下信息：昵称，密码，邮箱，手机号。更改密码需要与确认密码相同。
6. 设置用户偏好：在个人资料页面，有喜好设定部分，分别有电影题材的选项，导演的选项，演员的选项。用户勾选自己喜爱的选项并提交即可设置。在用户查看时，之前选好的选项应该高亮显示。
7. 更改个人偏好：同样地，直接通过勾选或者取消勾选，并提交，即可更改当前的喜好信息。

二. 电影浏览功能

1. 展示：当需要显示电影的基本信息时，每个电影显示的内容有：电影名称、演员、上映时间、简介这几种信息。
2. 搜索：在主页页面顶部有搜索栏，可根据电影名称、导演、演员、题材进行搜索，并展示搜索结果。
3. 详细信息：用户点击电影名称的超链接之后，进入介绍电影的详细界面，展示相关的基本信息等。用户也可点击演员名的超链接，进入介绍演员信息的页面。

三. 影评功能

1. 浏览影评：显示电影信息时，应在电影信息的下面展示用户的影评。在影评部分的最开始展示此电影评论的总个数。之后展示的每一条影评展示以下信息：昵称、评价内容、评价时间。
2. 发布影评：在浏览影评的下方，用户可以发布自己的评论。用户只填写自己的影评内容，提交即可。发布成功之后，回到刚才浏览的电影界面，此时应该能够看到用户发布的新影评。

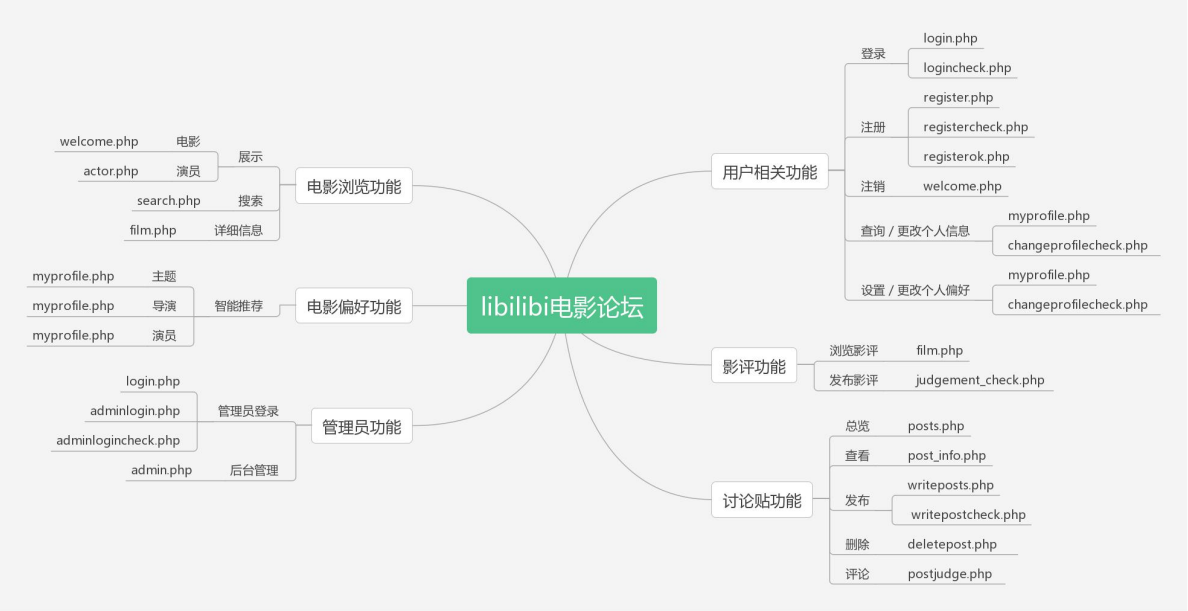
四. 电影偏好功能

1. 智能推荐：用户登录之后，在主界面为用户推荐用户喜欢的电影，分别根据用户设置的题材、导演、演员偏好进行推荐。
2. 设置与更改偏好：在用户个人信息界面可设置更改。

五. 讨论帖功能

1. 总览帖子：用户通过主页面进入讨论贴页面，分条显示每一条讨论贴，每条帖子显示以下信息：标题和发布时间。用户点击标题的超链接进入帖子详细界面。
2. 查看帖子：进入帖子的详细界面之后，展示讨论贴的标题、内容、发布时间。如果没有评论，提示当前帖子没有评论；如果有评论，则分条显示这些评论，每一条评论显示评论的用户名、回复内容、回复时间。
3. 发布帖子：在发表帖子的界面，用户填写讨论贴的标题和讨论贴的内容，发布成功后回到讨论贴总览的界面，此时能够看到钢材发布的新的讨论贴。
4. 删除帖子：在讨论贴总览的界面，对于用户自己发布的帖子，标题旁边会有一个删除按钮，当用户点击之后，会删除该讨论贴，提示删除成功，回到帖子总览界面，此时可以看到刚才的帖子已经删除了。
5. 评论帖子：在帖子的详细界面的最下方，用户可以添加自己的评论，用户只需要填写自己的评论内容即可。评论成功后，回到详细界面，此时可以看到刚才新加的评论。

二．系统功能结构设计


libilibi电影论坛
电影浏览功能
展示
电影
welcome.php
演员
actor.php
搜索
search.php
详细信息
film.php
电影偏好功能
智能推荐
主题
myprofile.php
导演
myprofile.php
演员
myprofile.php
管理员功能
管理员登录
login.php
adminlogin.php
adminlogincheck.php
后台管理
admin.php
用户相关功能
登录
login.php
logincheck.php
注册
register.php
registercheck.php
registerok.php
注销
welcome.php
查询 / 更改个人信息
myprofile.php
changeprofilecheck.php
设置 / 更改个人偏好
myprofile.php
changeprofilecheck.php
影评功能
浏览影评
film.php
发布影评
judgement_check.php
讨论贴功能
总览
posts.php
查看
post_info.php
发布
writeposts.php
writepostcheck.php
删除
deletepost.php
评论
postjudge.php

三、数据库基本表的定义

共定义了 14 个基本表。

表名:actor

用于记录演员的基本信息

列名	类型	描述	是否允许为空
actor_id	int	演员的 id，自增主键	
actor_name	varchar(20)	演员姓名，utf8 编码	
info	varcahr(300)	关于该演员的介绍，utf8 编码	允许
picture	varchar(20)	存储了该演员照片的路径	允许

表名:Admin

用于记录网站管理员的登陆账户和口令

列名	类型	描述	是否允许为空
name	varchar(10)	管理员的登陆账户	
pwd	varchar(20)	管理员的登陆口令	

表名:director

用于记录网站管理员的登陆账户和口令

列名	类型	描述	是否允许为空
director_id	int	导演的 id，自增主键	
director_Name	varchar(20)	导演的姓名，utf8 编码	

表名:film_info

用于记录电影的详细信息

列名	类型	描述	是否允许为空
film_id	int	影片的 id，自增主键	
film_name	char(20)	影片名，utf8 编码	
film_date	date	影片的发布日期	允许
film_info	varchar(600)	影片内容的详细描述	允许
director_Id	int	影片的导演的 id	允许
picture	varchar(30)	存储了该电影照片的路径	允许

表名:film_actor

用于记录不同电影的参演演员

列名	类型	描述	是否允许为空
film_id	int	影片的 id	
actor_id	int	演员的 id	
is_leading	tinyint	是否为主演，主演值为 1	允许，默认值为 0

表名:film_theme

用于记录不同电影的所属类型

列名	类型	描述	是否允许为空
film_id	int	影片的 id	
theme_id	int	影片所述的类型的 id，一个电影的所属类型可能为多个	

表名:judgement

用于记录电影的评论信息

列名	类型	描述	是否允许为空
judge_id	int	评论的 id，自增主键	
film_id	int	评论的影片的 id	
judge_time	datetime	评论的日期时间	允许
info	varchar(600)	评论内容，utf8 编码	允许
user_id	int	发布此评论的用户的 id	

表名:posts

用于记录用户发布的讨论帖的部分信息

列名	类型	描述	是否允许为空
posts_id	int	讨论帖的 id，自增主键	
user_id	int	发布此讨论帖的用户的 id	
topic	char(40)	讨论帖的标题，utf8 编码	
info	varchar(200)	发布讨论帖时填写的内容，utf8 编码	允许
release_time	datetime	讨论帖发布的日期时间	

表名:posts_info

用于记录讨论帖中用户回复的相关信息

列名	类型	描述	是否允许为空
post_id	int	讨论帖的 id	
user_id	int	回复此讨论帖的用户的 id	
respond_idx	int	回复的 id(编号)，每一条回复都有不同的 id	
respond_info	varchar(120)	回复的内容	允许
respond_time	datetime	回复的日期时间	允许

表名:theme

用于记录网站所拥有的电影类型

列名	类型	描述	是否允许为空
theme_id	int	电影类型的 id	
theme_name	char(5)	电影类型的中文名，utf8 编码	

表名:user_actor

用于记录用户对某些演员的偏好

列名	类型	描述	是否允许为空
user_id	int	用户 id	
actor_id	int	用户喜爱的演员的 id，一个用户可能喜爱多个演员	

表名:user_director

用于记录用户对某些导演的偏好

列名	类型	描述	是否允许为空
user_id	int	用户 id	
director_id	int	用户喜爱的导演的 id，一个用户可能喜爱多个导演	

表名:user_theme

用于记录用户对某些电影类型的偏好

列名	类型	描述	是否允许为空
user_id	int	用户 id	
theme_id	int	用户喜爱的电影类型的 id，一个用户可能喜爱多个电影类型	

表名:user_list

用于记录用户对某些电影类型的偏好

列名	类型	描述	是否允许为空
user_id	int	用户 id，自增主键	
phone_num	decimal(11,0)	用户的手机号	允许
pwd	char(20)	登陆时的用户密码	
user_name	varchar(11)	登录时的用户名	
nick_name	char(20)	用户的昵称	
mail	char(30)	用户的邮箱，必须符合邮箱的地址格式	

DDL 语言为：

```
CREATE TABLE `actor` (
  `actor_id` int(11) NOT NULL,
```

```
  `actor_name` char(20) NOT NULL,
  `info` varchar(300) DEFAULT NULL,
  `picture` varchar(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `Admin` (
  `name` varchar(10) NOT NULL,
  `pwd` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `director` (
  `director_Id` int(11) NOT NULL,
  `director_Name` char(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;


CREATE TABLE `film_actor` (
  `film_id` int(11) NOT NULL,
  `actor_id` int(11) NOT NULL,
  `is_leading` tinyint(4) DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `film_info` (
  `film_id` int(11) NOT NULL,
  `film_name` char(20) NOT NULL,
  `film_date` date DEFAULT NULL,
  `info` varchar(600) DEFAULT NULL,
  `director_Id` int(11) DEFAULT NULL,
  `picture` varchar(30) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `film_theme` (
  `film_id` int(11) NOT NULL,
  `theme_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `judgement` (
  `judge_id` int(11) NOT NULL,
  `film_id` int(11) NOT NULL,
  `judge_time` datetime DEFAULT NULL,
  `info` varchar(240) CHARACTER SET utf8mb4 DEFAULT NULL,
  `user_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `posts` (
  `posts_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `topic` char(40) NOT NULL,
  `info` varchar(200) DEFAULT NULL,
  `release_time` datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `posts_info` (
  `post_id` int(11) NOT NULL,
  `respond_idx` int(11) NOT NULL,
  `respond_time` datetime DEFAULT NULL,
  `user_id` int(11) NOT NULL,
  `respond_info` char(120) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `theme` (
  `theme_id` int(11) NOT NULL,
  `theme_name` char(5) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `user_actor` (
  `user_id` int(11) NOT NULL,
  `actor_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `user_director` (
  `user_id` int(11) NOT NULL,
  `director_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `user_list` (
  `user_id` int(11) NOT NULL,
  `phone_num` decimal(11,0) DEFAULT NULL,
  `pwd` char(20) NOT NULL,
  `nick_name` char(20) DEFAULT NULL,
  `mail` char(30) DEFAULT NULL,
  `user_name` varchar(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `user_theme` (
  `user_id` int(11) NOT NULL,
  `theme_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

四、触发器的定义与实现

触发器 D_Post_info

代码：

触发器功能：当删除某个讨论帖时，需要将这个讨论帖的回复内容也从数据库中删除

```
CREATE TRIGGER D_Post_info BEFORE DELETE ON posts FOR EACH ROW

BEGIN

DELETE FROM posts_info WHERE posts_info.post_id = old.posts_id;

end
```

触发器 D_user_info

触发器功能：当删除某个用户时，需要将这个用户的所有信息从数据库中删除，如这个用户的偏好、的电影发表评论。

代码：

```
CREATE TRIGGER D_user_info BEFORE DELETE ON user_list FOR EACH ROW

BEGIN

DELETE FROM judgement WHERE judgement.user_id = old.user_id;

DELETE FROM posts WHERE posts.user_id = old.user_id;

DELETE FROM posts_info WHERE posts_info.user_id = old.user_id;

DELETE FROM user_actor WHERE user_actor.user_id = old.user_id;

DELETE FROM user_director WHERE user_director.user_id = old.user_id;

DELETE FROM user_theme WHERE user_theme.user_id = old.user_id;

end
```

触发器 picture_default

触发器功能：添加电影图片时，如果没有照片，需要将照片的路径设为一张默认的图片路径。

代码：

```
CREATE TRIGGER `picture_default` AFTER INSERT ON `film_info`

 FOR EACH ROW BEGIN

  UPDATE film_info SET picture = './default.jpg' WHERE picture = NULL;

end
```

触发器 picture_default2

触发器功能：添加演员图片时，如果没有照片，需要将照片的路径设为一张默认的图片路径。

代码：

```
CREATE TRIGGER `picture_default2` AFTER INSERT ON `actor`

 FOR EACH ROW BEGIN

  UPDATE actor SET picture = './default.jpg' WHERE picture = NULL;

end
```

五、存储过程的定义与实现

存储过程：C_user(用户注册)

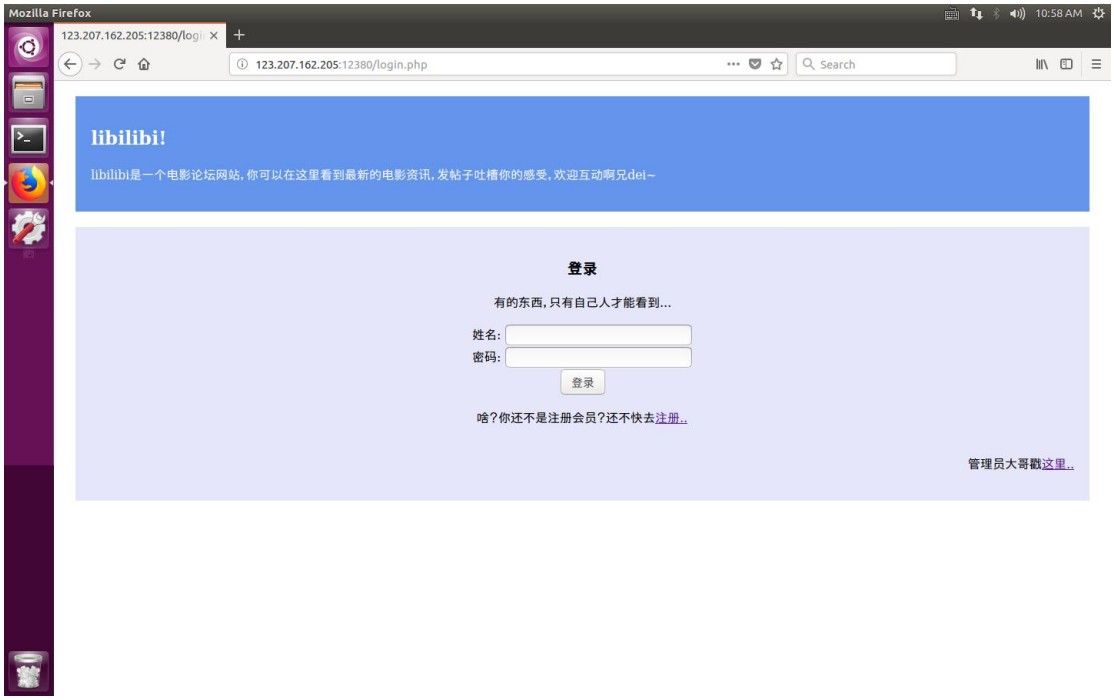
描述：在用户注册时，填写的内容较多，可以采用 procedure 的方式使前端代码更简单清晰。

代码：

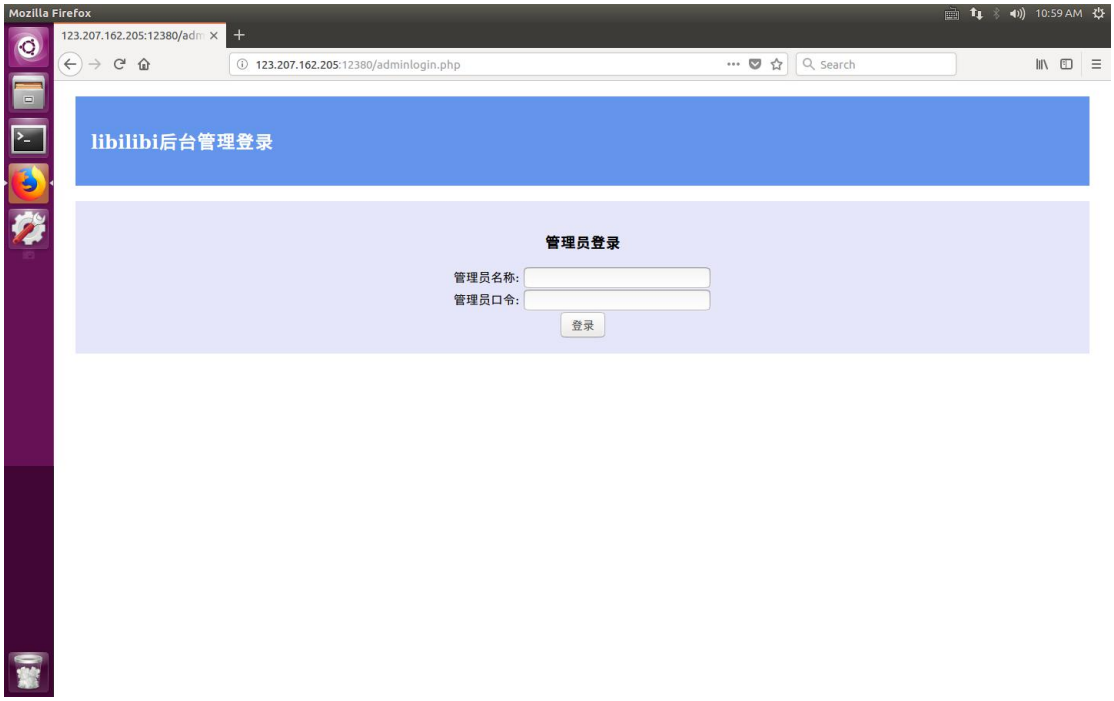
```
CREATE PROCEDURE C_user(IN u_name CHAR(11), n_name CHAR(20), paswd CHAR(20), mail CHAR(30), ph_num decimal(11,0))

begin

INSERT INTO `user_list` (`user_id`, `phone_num`, `pwd`, `nick_name`, `mail`, `user_name`) VALUES (NULL, ph_num, paswd, n_name, mail, u_name);

end
```

六、系统实现结果

用户登录：



管理员登录：



用户注册：


Mozilla Firefox
10:58 AM
123.207.162.205:12380/register.php
Search
libilibi!
libilibi是一个电影论坛网站，你可以在这里看到最新的电影资讯，发帖子吐槽你的感受，欢迎互动啊兄dei~
注册
简单几步就可以变成大爷啦.
用户名:
昵称:
密码:
确认密码:
邮箱:
手机号:
点击注册，开启新世界!
注册

主界面：


Mozilla Firefox
11:11 AM
123.207.162.205:12380/welcome.php
Search
libilibi!
libilibi是一个电影论坛网站，你可以在这里看到最新的电影资讯，发帖子吐槽你的感受，欢迎互动啊兄dei~
大爷来啦！keenan今儿来看点啥?
电影名/导演/演员/题材
搜索
讨论帖 | 我的资料 | 注销 | 打赏
根据你喜爱的题材提供的电影...
Kingsman
王牌特工2
电影日期:2017-10-20
电影简介:时光飞逝，一转眼，艾格西（塔伦·埃格顿 Taron Egerton 饰）已经成长为了一名出色而又可靠的特工，他和蒂尔德公主（汉娜·阿尔斯托姆 Hanna Alström 饰）之间的感情也愈演愈浓，两人眼看着就要携手步入婚姻的殿堂。就在这个节骨眼上，前特工查理（爱德华·霍尔克罗夫特 Edward Holcroft 饰）杀了回来，如今的他效力于一个名为“黄金圈”的贩毒组织，组织头目波比（朱丽安·摩尔 Julianne Moore 饰）是一个邪恶而又野心勃勃的女人。 查理查出了金士曼的所有据点，用导弹将它们全部摧毁。幸存的艾格西和梅林（马克·斯特朗 Mark Strong 饰）千里迢迢远赴美国，向潜伏在那里的联邦特工寻求帮助。波比种植了一种含有毒素的大麻，将它们输送往世界各地，当瘾君子们体内的毒素渐渐发作后，波比以此为筹

查看和修改资料：


Mozilla Firefox
11:13 AM
123.207.162.205:12380/myp
123.207.162.205:12380/
123.207.162.205:12380/myprofile.php
Search
libilibi!
libilibi是一个电影论坛网站，你可以在这里看到最新的电影资讯，发帖子吐槽你的感受，欢迎互动啊兄dei~
返回
我的档案
用户id: 5
用户名: keenan
昵称: keke
电话: 1234
邮箱: 1234
更改档案
爷也可以更改个人信息哦！
*注:除了用户名不能更改外，其他均可以更改..
昵称:
密码:
确认密码:
邮箱:
手机号:
更改

设置偏好：


Mozilla Firefox
11:13 AM
123.207.162.205:12380/myp
123.207.162.205:12380/
123.207.162.205:12380/myprofile.php
Search
爷也可以更改个人信息哦！
*注:除了用户名不能更改外，其他均可以更改..
昵称:
密码:
确认密码:
邮箱:
手机号:
更改
喜好设定
选择自己的喜好，libi会根据你的喜好进行推荐
电影题材
冒险 剧情 动作 动画 历史 古装 喜剧 奇幻 恐怖 悬疑 战争 武侠 灾难 爱情 犯罪 科幻
导演
弗兰克·德拉邦特 马修·沃恩
演员
塔伦·埃格顿 科林·费尔斯 马克·斯特朗 朱丽安·摩尔 埃尔顿·约翰 塞缪尔·杰克逊 迈克尔·凯恩
选择

搜索结果：


Mozilla Firefox
11:17 AM
123.207.162.205:12380/search.php
Search
libilibi是一个电影论坛网站, 你可以在这里看到最新的电影资讯, 发帖子吐槽你的感受, 欢迎互动啊兄dei~
查询结果
Kingsman
王牌特工2
电影日期:2017-10-20
电影简介:时光飞逝, 一转眼, 艾格西(塔伦·埃格顿 Taron Egerton 饰)已经成长为了一名出色而又可靠的特工, 他和蒂尔德公主(汉娜·阿尔斯托姆 Hanna Alström 饰)之间的感情也愈演愈浓, 两人眼看着就要携手步入婚姻的殿堂。就在这个节骨眼上, 前特工查理(爱德华·霍尔克罗夫特 Edward Holcroft 饰)杀了回来, 如今的他效力于一个名为"黄金圈"的贩毒组织, 组织头目波比(朱丽安·摩尔 Julianne Moore 饰)是一个邪恶而又野心勃勃的女人。 查理查出了金士曼的所有据点, 用导弹将它们全部摧毁。幸存的艾格西和梅林(马克·斯特朗 Mark Strong 饰)千里迢迢远赴美国, 向潜伏在那里的联邦特工寻求帮助。波比种植了一种含有毒素的大麻, 将它们输送往世界各地, 当瘾君子们体内的毒素渐渐发作后, 波比以此为筹码, 正式向政府宣战。
返回

电影信息及评论：

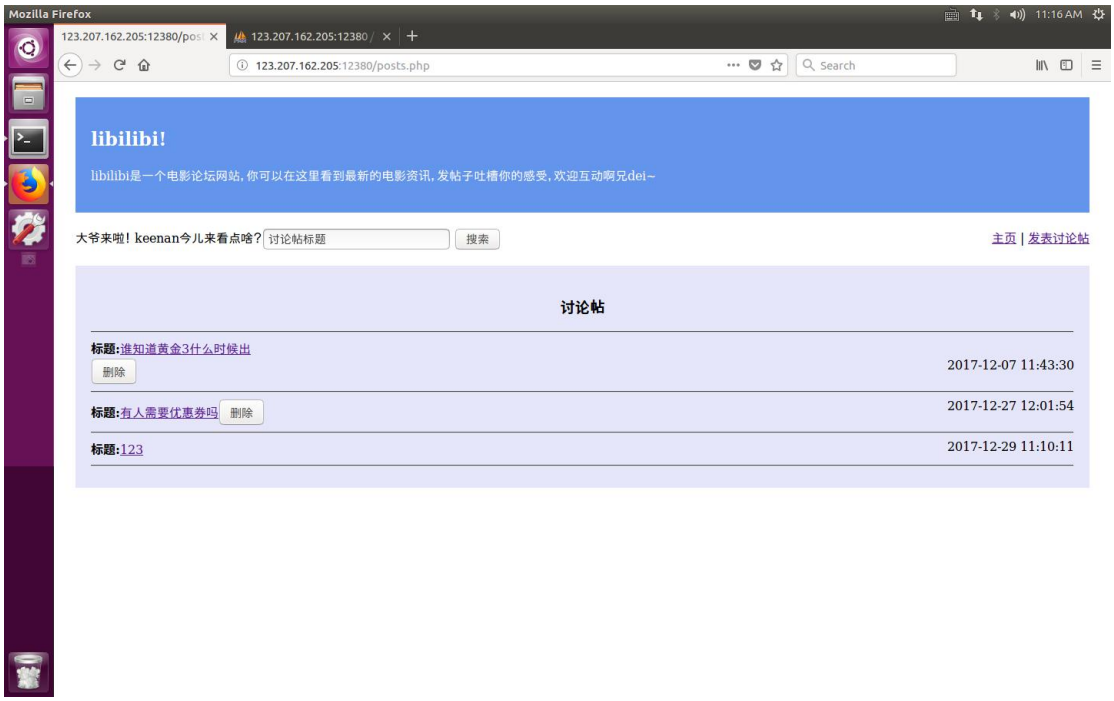

Mozilla Firefox
11:12 AM
123.207.162.205:12380/film.php?film=1
Search
大爷来啦! keenan今儿来看点啥?
电影名/导演/演员/题材
搜索
主页 | 讨论帖 | 我的资料 | 注销 | 打赏
详细信息
Kingsman
王牌特工2
演员:塔伦·埃格顿 科林·费尔斯 马克·斯特朗 朱丽安·摩尔 埃尔顿·约翰
电影日期:2017-10-20
电影简介:
时光飞逝, 一转眼, 艾格西(塔伦·埃格顿 Taron Egerton 饰)已经成长为了一名出色而又可靠的特工, 他和蒂尔德公主(汉娜·阿尔斯托姆 Hanna Alström 饰)之间的感情也愈演愈浓, 两人眼看着就要携手步入婚姻的殿堂。就在这个节骨眼上, 前特工查理(爱德华·霍尔克罗夫特 Edward Holcroft 饰)杀了回来, 如今的他效力于一个名为"黄金圈"的贩毒组织, 组织头目波比(朱丽安·摩尔 Julianne Moore 饰)是一个邪恶而又野心勃勃的女人。 查理查出了金士曼的所有据点, 用导弹将它们全部摧毁。幸存的艾格西和梅林(马克·斯特朗 Mark Strong 饰)千里迢迢远赴美国, 向潜伏在那里的联邦特工寻求帮助。波比种植了一种含有毒素的大麻, 将它们输送往世界各地, 当瘾君子们体内的毒素渐渐发作后, 波比以此为筹码, 正式向政府宣战。
用户评论


Mozilla Firefox
123.207.162.205:12380/film.php?film=2
王牌特工:特工学院
演员:塔伦·埃格顿 科林·费尔斯 马克·斯特朗 塞缪尔·杰克逊 迈克尔·凯恩
电影日期:2015-03-27
电影简介:
哈里(科林·费斯 Colin Firth 饰)是英国秘密特工组织金士曼中的一员,某次行动中,他的战友不幸牺牲,哈里将一枚徽章和一个电话号码交给了战友年幼的小儿子艾格西(亚历克斯·尼科洛夫 Alex Nikolov 饰),叮嘱他将来如果遇到了什么麻烦可以拨打这个号码,然而,这样的机会只能使用一次。 一晃眼十七年过去,破碎的家庭让艾格西(塔伦·埃格顿 Taron Egerton 饰)成长为了一个整日无所事事的小混混,某日,因为违反交通规则而遭到逮捕的艾格西使用了手中珍贵的号码,赶来的哈里在艾格西玩世不恭的外表之下发现了他善良的本质和极高的天赋,于是,哈里决定将艾格西培养成为新一代金士曼,他们需要共同面对的是强大而又邪恶的亿万富翁瓦伦丁(塞缪尔·杰克逊 Samuel L. Jackson 饰)。
用户评论
该电影还没有评论,去发表评论吧
发表你的评论
提交

演员信息：

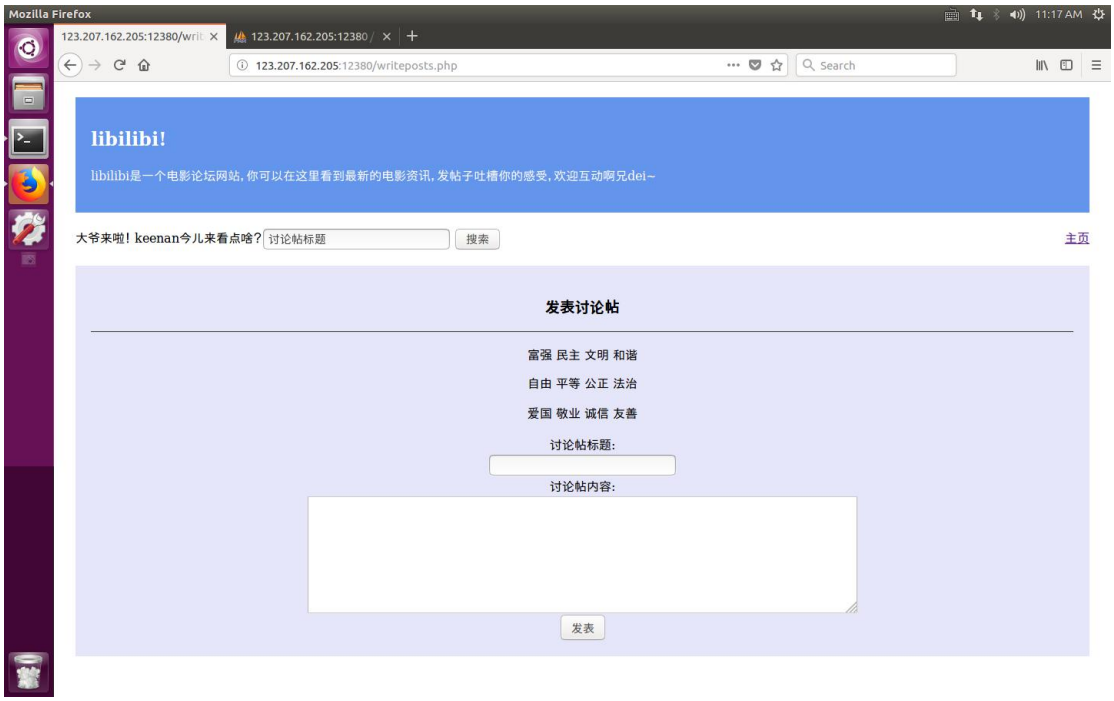

Mozilla Firefox
123.207.162.205:12380/actor.php?actor=1
libilibi!
libilibi是一个电影论坛网站,你可以在这里看到最新的电影资讯,发帖子吐槽你的感受,欢迎互动啊兄dei~
大爷来啦! keenan今儿来看点啥?
电影名/导演/演员/题材
搜索
主页 | 讨论帖 | 我的资料 | 注销 | 打赏
演员详细信息
演员名称:塔伦·埃格顿
演员信息:
塔伦·埃哲顿(Taron Egerton),1989年11月10日出生于英国威尔士,英国演员。2007年,在由唐·瑞德和 Bill Anderson执导的《督察刘易斯》中饰 Liam Jay 一角。2014年,在由马修·沃恩执导的《王牌特工:特工学院》中饰埃格西,该片2015年2月13日在美国上映。2015年1月,塔伦被评为英国《GQ》杂志50多位最佳西装穿搭英国男人之一。

讨论帖主界面：

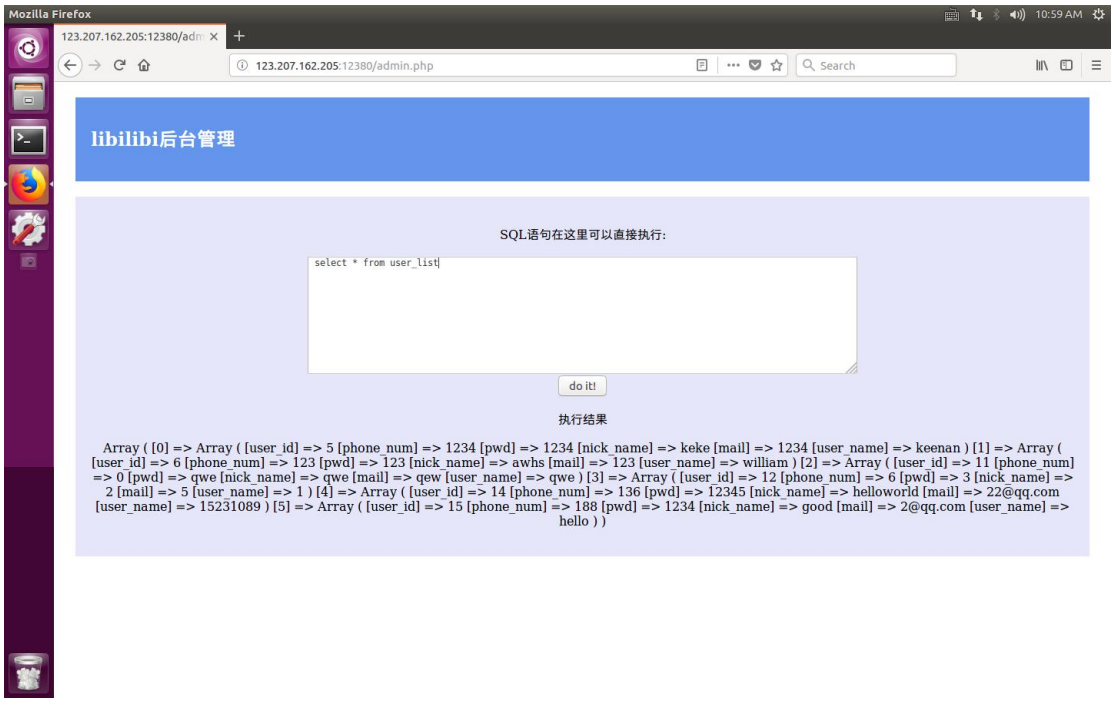


讨论帖详细信息及回复：

发布讨论帖：


Mozilla Firefox
123.207.162.205:12380/writeposts.php
libilibi!
libilibi是一个电影论坛网站，你可以在这里看到最新的电影资讯，发帖子吐槽你的感受，欢迎互动啊兄dei~
大爷来啦！keenan今儿来看点啥？
讨论帖标题
搜索
主页
发表讨论帖
富强 民主 文明 和谐
自由 平等 公正 法治
爱国 敬业 诚信 友善
讨论帖标题:
讨论帖内容:
发表

后台管理：


Mozilla Firefox
123.207.162.205:12380/admin.php
libilibi后台管理
SQL语句在这里可以直接执行:
select * from user_list
do it!
执行结果
Array ([0] => Array ([user_id] => 5 [phone_num] => 1234 [pwd] => 1234 [nick_name] => keke [mail] => 1234 [user_name] => keenan) [1] => Array ([user_id] => 6 [phone_num] => 123 [pwd] => 123 [nick_name] => awhs [mail] => 123 [user_name] => william) [2] => Array ([user_id] => 11 [phone_num] => 0 [pwd] => qwe [nick_name] => qwe [mail] => qew [user_name] => qwe) [3] => Array ([user_id] => 12 [phone_num] => 6 [pwd] => 3 [nick_name] => 2 [mail] => 5 [user_name] => 1) [4] => Array ([user_id] => 14 [phone_num] => 136 [pwd] => 12345 [nick_name] => helloworld [mail] => 22@qq.com [user_name] => 15231089) [5] => Array ([user_id] => 15 [phone_num] => 188 [pwd] => 1234 [nick_name] => good [mail] => 2@qq.com [user_name] => hello))

七、总结

通过这学期的数据库课设实验，我们整体的开发了一个电影评价的网站。相比后端的数据库和 DBMS 的搭建工作，在前端的实现上占用了更多的开发时间。所以在整个项目的开发中，我们两人不仅把在数据库理论课上学到的知识成功运用到了我们的项目中，加深了我们对数据库知识的掌握程度。更让我们积累了一些开发前端的知识和经验。

总结整个网站搭建的过程，不得不说按照课设的既定流程来开发是很有效的，从一开始的需求分析到 E_R 图的绘制，再到最后整个模型的建立和功能的实现。正是因为按照既定的流程来完成整个课设，才使整个项目能够顺利高效的完成，没有出现为了完成某项工作花费了大量精力去添之前环节的“坑”的情况。

虽然因为时间问题，我们的网站并没有完全实现预期的效果，但是我们两人对数据库项目的理解和开发能力确实提高显著。