

一、需求

我们不知道项目需求，我们根本无从下手。

如果不知道需求，就像要造一样女神，只能想象女神的样子。而造出来的女神，又不是用户所想要的。

可以这么讲，将需求弄明白了。就搞明白了，在没完成之前，我们就知道未来开发完后的东西是什么样的样子。

当然需求的采集、产生、策划、定制... ..涉及到太多跨学科的知识，不是我们今天这本书里面涵盖的内容。我们直接跨过需求的采集部分，为大家还原一整套需求，从已确定的需求中开始项目的开发。

1.论坛定位

本论坛的定位和开发是根据大家之前所学的知的基础上，将基础知识和内容进行整合而开发出来的教学项目。

大家通过之前的学习的组合，认真学，认真写后会发现自己掌握了很多知识，可是为什么写不出一个中小型的项目呢？

因为，在互联网的业务中需要非常明确自己要做什么。再根据做的东西，每一步每一步把业务需求转换成为代码。

之前的知识学好后，绝大多数人在转换的过程当中也会略有一些阵痛。怎么解决呢？

解决办法：通常不知道需求是什么，不能根据需求设计出来流程和数据表结构和规范。本章会帮你解决好核心的业务流程，提供数据库，并提供全面的代码，你只需要够静下心来一个一个功能写出来。本章需要你付出一定的心智去解决一些或大或小的隐藏任务，以达到真正理解什么是论坛系统(BBS)和如何运用python独立开发论坛系统的目的。

采用面向过程的方式做完一个项目。在面向对象的过程中，你需要解决的只是了解知识点。

在以后，你学习的更加强大，你会发现面向过程和面向对象没有多大的区别。

你学习了面向对象的知识后，不需要再经历业务转化代码的过程。只需要将代码以面向对象的方式写出来而已。

不用再经历复杂的需求变代码的过程。

2.前台、后台；前端、后端

了解了论坛的定位，我们再讲几个基本概念：前台、后台；前端、后端。

前台：就是用户看到的那些东西，提供给所有用户使用。用户包括注册用户和普通游客，二者所拥有的功能不同。系统前台主要包括用户登录、用户注册、发表帖子、论坛浏览、回复内容浏览等功能模块。

后台：有时也称为网站管理后台，通常需要帐号及密码等信息的登陆验证，登陆信息正确则验证然后进入网站后台的管理界面进行相关的一系列操作：可对论坛中相关信息进行管理、维护，普通用户无权使用。后台管理主要包括用户管理、版块管理、帖子管理、回复管理、管理员注销等功能模块。

前端：手机端我们认为是前端，只展现和操作界面，而数据处理通常在远程服务器上存储着。它直接给用户展示内容，我们认为直接给用户展示，我们认为是前端。网页，我们认为是前端。前端主要是指html、css、javascript写出来的页面给用户直接看到的展现。

后端：用户的登陆、用户密码的处理都需要在服务器端处理。用户的银行卡余额等都存在后端服务器中，我们将服务器处理、存储数据和业务逻辑的部份存为后端。

3.总体业务流程介绍

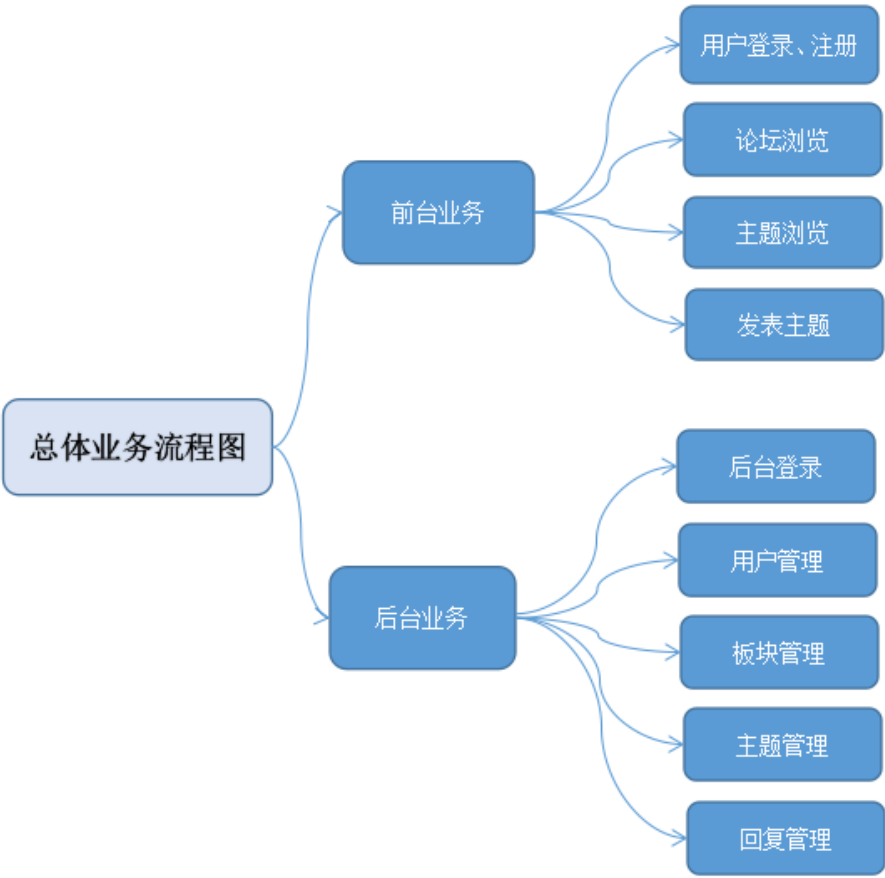
前台业务流程主要包括：用户注册、用户登录；论坛浏览：显示论坛名称、显示论坛创建时间；主题浏览：回复文章作者信息、作者相关信息浏览、原文浏览、回复文章浏览；发表主题：主题回复、发表新主题。

后台业务流程主要包括：后台登录、管理员注销；用户管理：删除用户、查询用户、编辑用户；栏目管理：更换版主、新增论坛、删除栏目；主题管理：主题查询、主题删除；回复管理：回复文章查询、回复文章删除。

前端：手机端我们认为是前端，只展现和操作界面，而数据处理通常在远程服务器上存储着。它直接给用户展示内容，我们认为直接给用户展示，我们认为是前端。网页，我们认为是前端。前端主要是指html、css、javascript写出来的页面给用户直接看到的展现。

后端：用户的登陆、用户密码的处理都需要在服务器端处理。用户的银行卡余额等都存在后端服务器中，我们将服务器处理、存储数据和业务逻辑的部份存为后端。

其处理流程如下图所示：



4.内容规划

4.1 角色设置

4.1.1 管理员

管理员权限，可进入后台，可进行回帖的管理、版块管理和用户管理等操作；

4.1.2 版主

版主权限，在前台管理帖子内容，包括将帖子设置为置顶、精华等操作。管理回帖内容，可屏蔽用户的回帖内容；

4.1.3 注册用户

在前台浏览帖子(付费帖需要消费对应的积分才能浏览), 发表帖子、回复;

4.1.4 访客

与注册用户的权限类似, 可浏览免费帖子和回复内容, 无发帖和回帖功能;

4.2 功能体现

4.2.1 前台功能主要包括:

用户注册、用户登录; 论坛浏览: 显示论坛名称、显示论坛创建时间; 帖子浏览: 回复的作者信息、作者相关信息浏览、原文浏览、回复内容浏览; 发表帖子: 帖子回复、发表新帖子。

4.2.2 后台功能主要包括:

后台登录、管理员注销; 用户管理: 删除用户、查询用户、编辑用户; 版块管理: 更换版主、新增版块、删除版块; 帖子管理: 帖子查询、帖子删除; 回复管理: 回复的查询、回复的删除。

4.3 奖励机制

为鼓励用户积极参与, 设置如下奖励机制:

注册: 赠送50积分; 登陆: 每天首次登陆赠送2积分; 发帖: 赠送2积分;
回帖: 赠送1积分;

4.4 论坛版主的权利及义务

版主负责的版面需及时回复网友的问题及为论坛制造气氛, 并能及时删除含有非法内容的贴(例如含广告、色情、反动等)

二、核心业务流程

1. 用户注册流程

- 注册功能说明 用户注册是网上最最常见的一个功能, 也是我们开始体验一个网

站服务的开始。

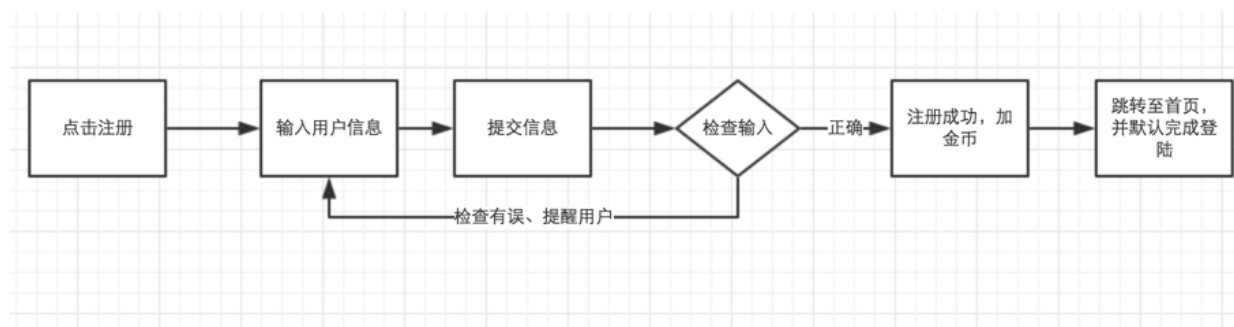
例如：我们在京东、淘宝等网站注册了才能发贴。我们在微信、QQ等社交工具注册了才能够开始加好友开始聊天。

可以这么来讲，如果网站需要我们有个人信息，那么注册是我们体验一个互联网服务的开始的第一步。

而我们在开发过程当中，也会设计按照自己业务的注册流程。

注册流程可以不同，也可以相同。我们现在将本论坛的注册流程用流程图来表示：

- 注册功能流程图



注意：大家可以在《16.6 核心功能说明》中下载安装后，进一步体验每一步流程。

- 用图分解注册流程演示

第一步：点击注册按钮进入注册界面



第二步：在注册页输入注册信息

*用户名:

*密码:

*确认密码:

*Email:

*验证码:


[提交](#)

第三步：判断是否用户输入错误，如果输入错误显示错误信息



用户名长度错误：用户名由 **3** 到 **12** 个字符组成
密码长度错误：由 **3** 到 **12** 个字符组成
错误：邮箱不合法
验证码输入错误，请稍候.....
[如果你的浏览器没有自动跳转，请点击此链接](#)

第四步：如果输入信息无误，提示用户注册成功，并完成跳转。



感谢您的注册，现在将以会员身份登录站点，请稍候.....
[如果你的浏览器没有自动跳转，请点击此链接](#)

注：请参考《数据库表设计》听用户表结构设计。对着流程完成用户注册代码的实现。

2.普通用户和管理员登陆流程

- 普通用户和管理员登陆的区别

我们在玩游戏的时候，会听到一个非常厉害的词:GM(游戏管理员)。游戏管理员，有一个专门的系统可以做到：

- 1. 删除用户
- 2. 封掉一个用户
- 3. 加一个新的游戏地图
- 4. 查看某些用户的聊天违规行为给出警告

... ..

大家会发现，GM在游戏里面有。她与普通用户有一些区别，她有更高的操作权限。

在我们的论坛、办公管理系统、小说网站等系统中也会设计这一类特殊的用户。

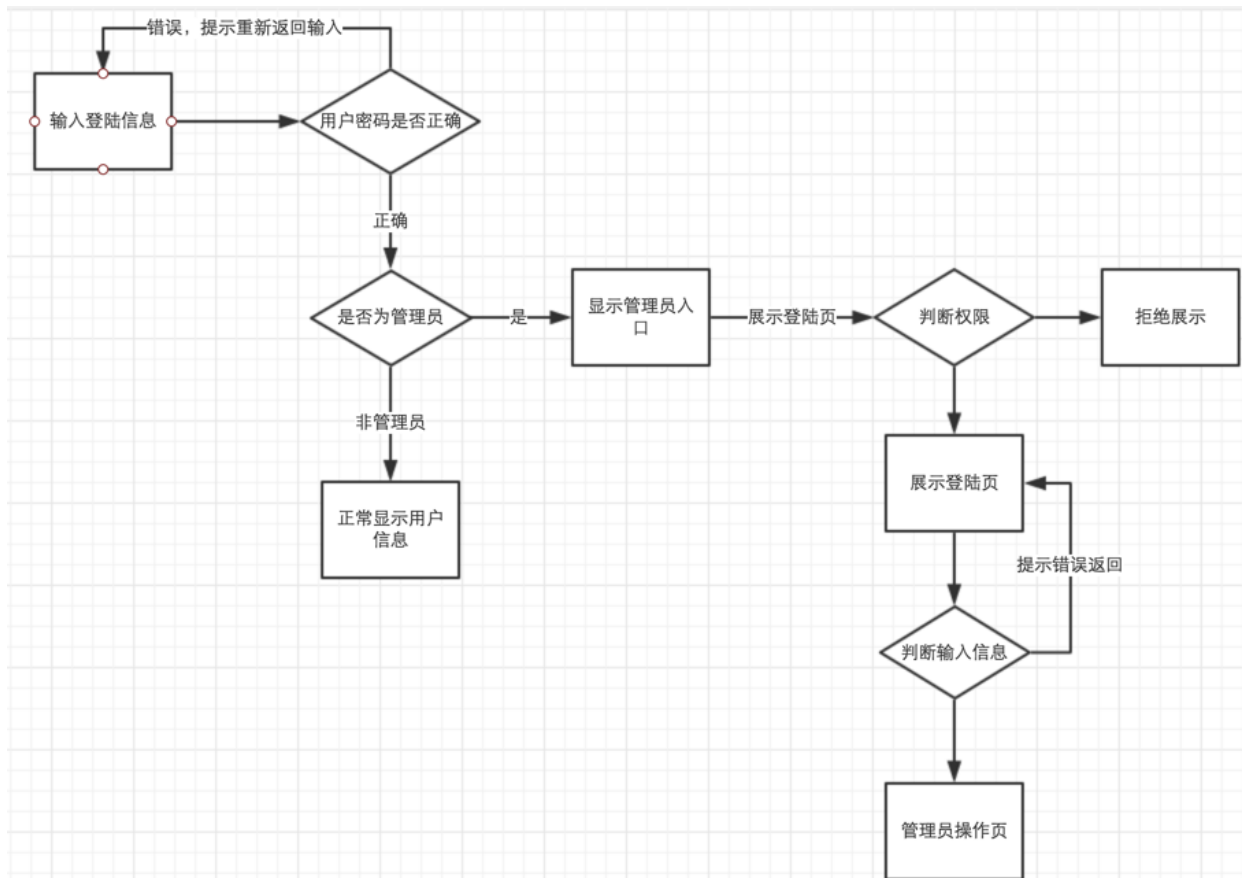
这类特殊用户我们统一称之为：管理员。

论坛系统中普通用户和管理员的区别：

权限	普通用户	管理员用户
注册	有	有
登陆	有	有
发贴	有	有
回贴	有	有
删自己贴	有	有
删他人贴	无	有
删除用户	版主有，用户没有	有
添加版主	无	有
禁止用户	无	有
添加版本	无	有
修改网站名等基本信息	无	有
添加友情链接	无	有

注意：所在板块的普通用户中，有一种特殊用户叫版主用户。版主与普通用户的区别是：版主这种普通用户，能够管理自己所在板块的贴子。删除贴子、屏蔽贴子等。

##普通用户、管理员登陆流程



流程说明：

1. 用户输入信息后，显示用户名和密码是否正确，不正确返回重新输入
2. 用户名和密码正确则激活用户登陆的cookie
3. 用户如果是管理员展示管理员入口；否则不显示，按普通用户操作
4. 管理员用户可进入管理员登陆页，进入前判断是否有进入该页的权限，如果没有则拒绝。
5. 输入管理员的帐号密码，进入验证通过则放行，否则拒绝通过。

##图解登陆流程

1.输入用户名和密码界面

用户名	<input type="text" value="liwenkai"/>	<input type="checkbox"/> 自动登录
密码	<input type="password"/>	<input type="button" value="登录"/>

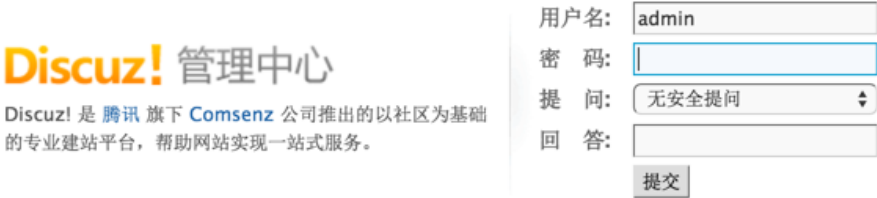
2.普通用户登陆成功展示



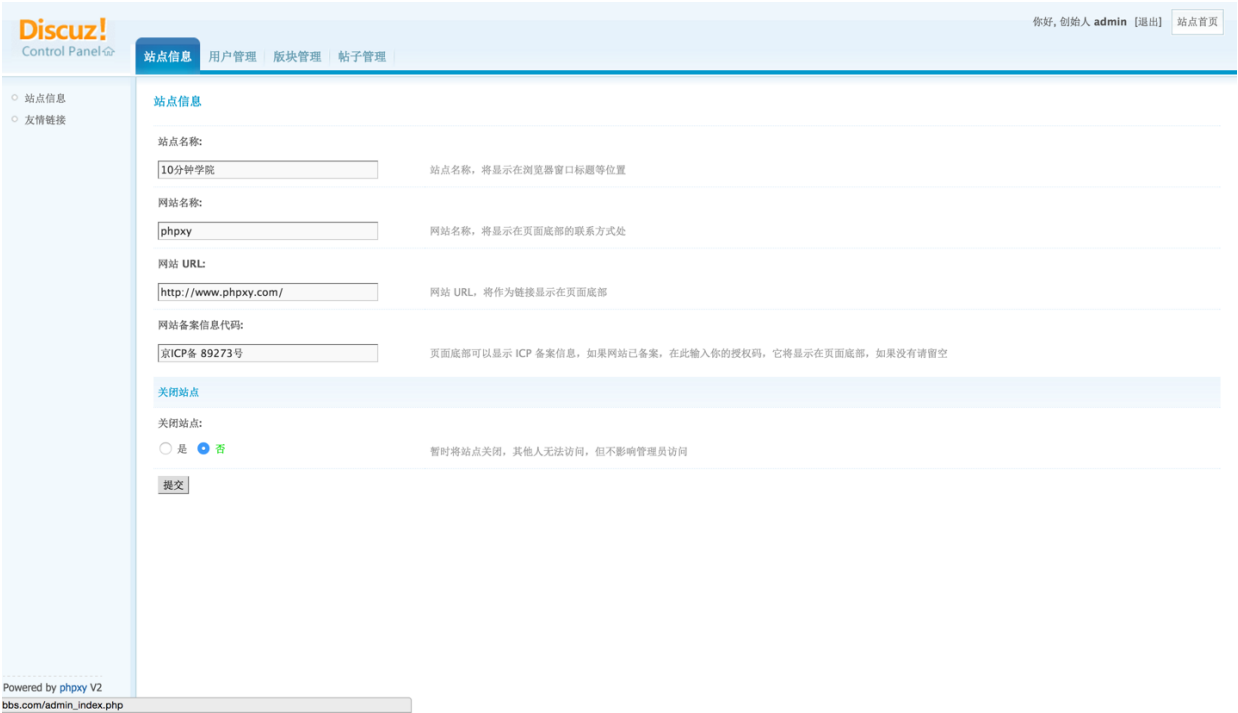
3.管理员登陆成功展示



4.管理员点击管理中心进入后台展示页



5.后台界面展示



3.发贴流程

- 发贴说明

某个用户登陆成功，就可以使用某个用户发表贴子。

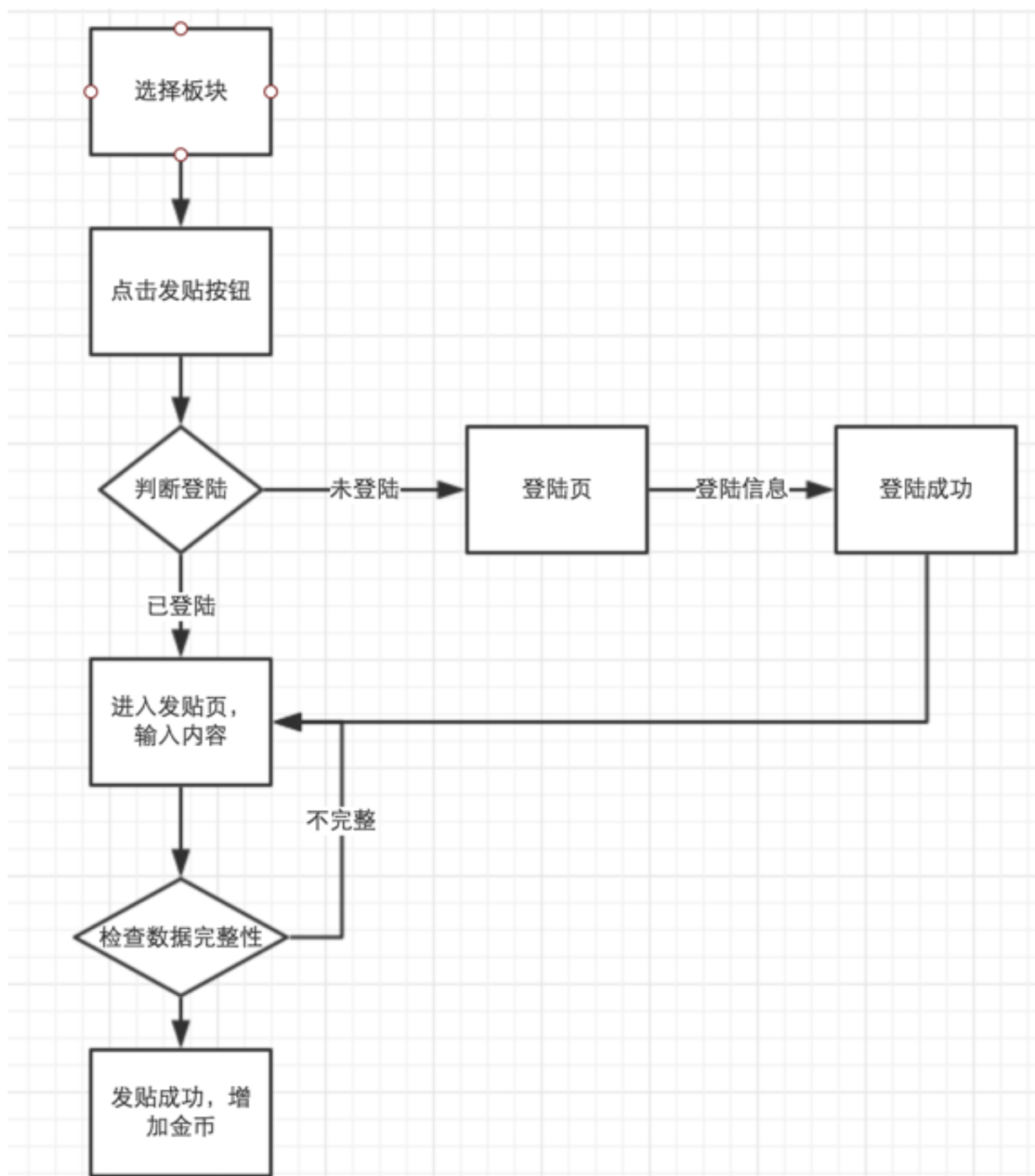
我们将贴子分为了普通贴和金币贴。

它们的区别：

1. 普通贴不需要花钱就可以看
2. 金币贴必须要使用注册时、每天登陆、回贴时获得的金币购买贴子后才能查看。

在发贴前，我们先要选择对应的版本，在所在贴块中点击发贴，写入对应的内容点击发表，发表成功。

流程图



说明：

1. 用户选择对应的版块后，点击发贴时传入了板块的ID
2. 在用户的发贴界面，将板块的ID放入了隐藏的表单中
3. 用户点发贴时，其实是相对应的

- 图解发贴流程

1. 登陆成功后选择论坛中的相应版块，如：选择内核源码

PHP技术交流		
内核源码 版主: test001, admin	10 / 9	本论坛超管的账号、密码都为admin 2015-10-08 10:49:55 admin
PHP框架 版主: test002, admin	0 / 0	从未
开源产品 版主: admin	0 / 0	从未
进阶讨论 版主: admin	0 / 0	从未
程序人生		
求职招聘 版主: admin	0 / 0	从未
经验分享 版主: admin	0 / 0	从未
名人故事 版主: admin	0 / 0	从未

2. 点击发贴按钮发表帖子

论坛 > PHP技术交流 > 内核源码

版块导航

PHP技术交流

内核源码

PHP框架

开源产品

进阶讨论

程序人生

求职招聘

经验分享

名人故事

内核源码 今日: 0 | 主题: 9

版主: test001, admin

发贴

返回

筛选: 全部 | 精华






	作者	回复/查看	最后发表
关于php你可能不知道的一php的事件驱动化设计 - [售价 10 金钱]	admin 2015-10-08	0 14	admin 2015-10-08 10:46:58
本论坛超管的账号、密码都为admin	admin 2015-10-08	0 5	admin 2015-10-08 10:49:55
树操作算法，非递归方法 - [售价 5 金钱]	admin 2015-10-08	0 2	admin 2015-10-08 10:46:07
【架构与优化】从facebook中集思广益，相互学习	admin 2015-10-08	10 21	test002 2015-10-08 10:59:19
深入理解zend sapi(zend sapi internals)	admin 2015-10-08	0 1	admin 2015-10-08 10:43:48


























3. 输入发贴内容、金币等


























发表帖子

还可输入 80 个字符

源码







B


I





























U


























abc

x₂

x²







样式




普通

字体

大小

A

A



body p

售价

售价: 金钱(最高 30)

4.贴子发表成功给用户虚拟金币奖励



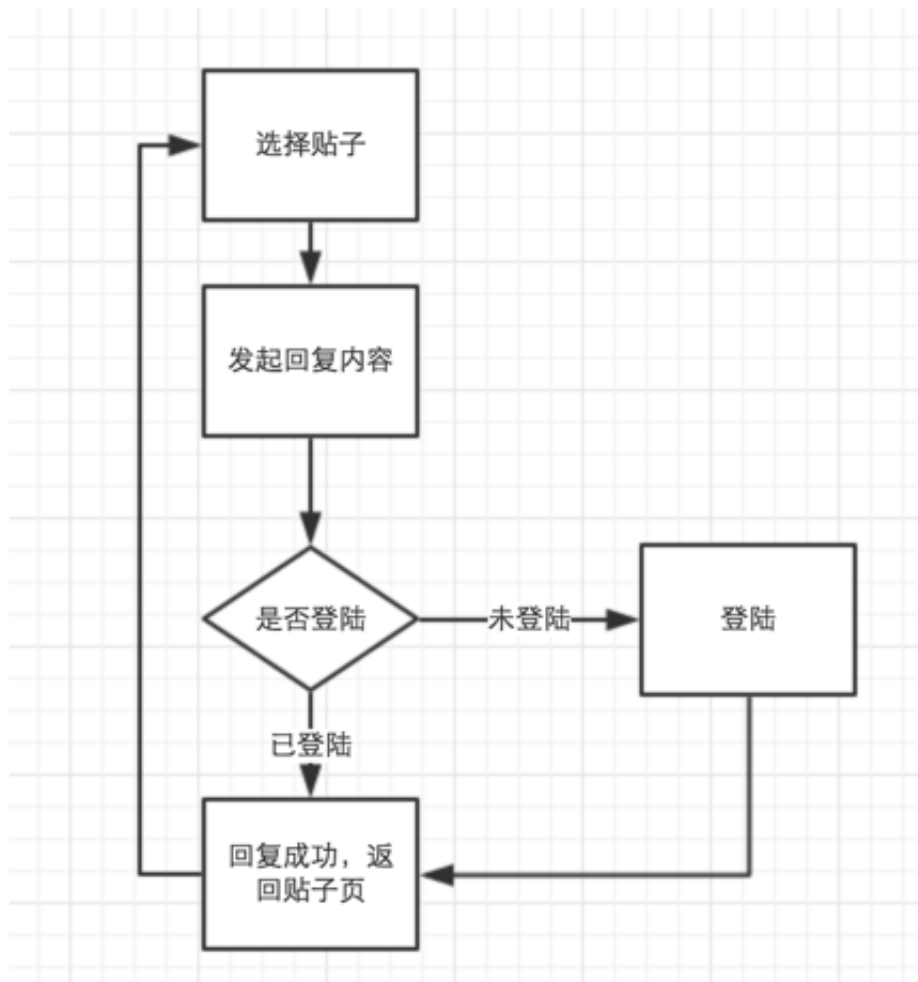
4.回复流程

● 回复流程说明

我们遇到自己感兴趣的贴子，我们在下面可以的输入框中再次发表自己的观点和看法。

回贴成功后，系统可以针对这个用户加上对应的金币。

回复流程图



注意：

1. 选择贴子发起回复，判断用户是否登陆
2. 如果用户已登陆则直接发表成功，在发表的隐藏表单中放入这个贴子的ID
3. 用户没登陆进入登陆页，登陆成功后返回至原贴地址重新发起回复

- 图解回复流程

1.选择并查看贴子



查看: 22 | 回复: 10

admin

管理员
学士

【架构与优化】从facebook中集思广益，相互学习

发表于 2015-10-08 10:45:17

楼主 电梯直达 ☐ 

本人才学疏浅，提出这个问题只是为了抛砖引玉，希望大家能在自己擅长的领域里对facebook中可能采用的某个技术或者架构方案或优化方法，做个自己的论述。根据2008年9月1日，腾讯网对facebook的技术运营副总裁jonathan heiliger的访问，我们了解到：

- (1) facebook自称全球流量第四的网站，每天9000万活跃用户访问facebook的1万台服务器。
- (2) 25tb数据（08年12月为28tb），40万名外部开发者支持，开发了2.5万套应用软件。
- (3) 使用lamp（[linux](#)、[apache](#)、[mysql](#)、[php](#)）技术构架；数据库使用mysql；使用memcached缓存sql加速（全球最大的分布式memcached缓存，800多台服务器，光缓存在memcached中数据就达20多tb）；使用apc进行opcode编译缓存。关于php的编译执行如下图所示：

而facebook的缓存方案如下图所示：

- (4) 一些后台应用是用python、perl和java，以及一些gcc和boost。
- (5) 使用svn和git来进行代码管理，并且全部企业内部的软件部署都采用开源程序。

2. 可能会采用的mysql架构与优化

2.在贴子下面输入想要回复的内容

A screenshot of a web browser window. The address bar shows a URL starting with 'http://'. Below the address bar is a text input field containing the Chinese text '我可以选择一个内容进行回复' (I can choose a content to reply). The input field has a blue border and a light gray background. The browser's toolbar includes icons for back, forward, and search. The text 'body p' is visible in the bottom left corner of the browser window.

发表回复

3.回复成功并跳转回原贴地址



帖子回复成功,请稍候.....

如果你的浏览器没有自动跳转,请点击此链接

5.版块管理流程

- 版块管理

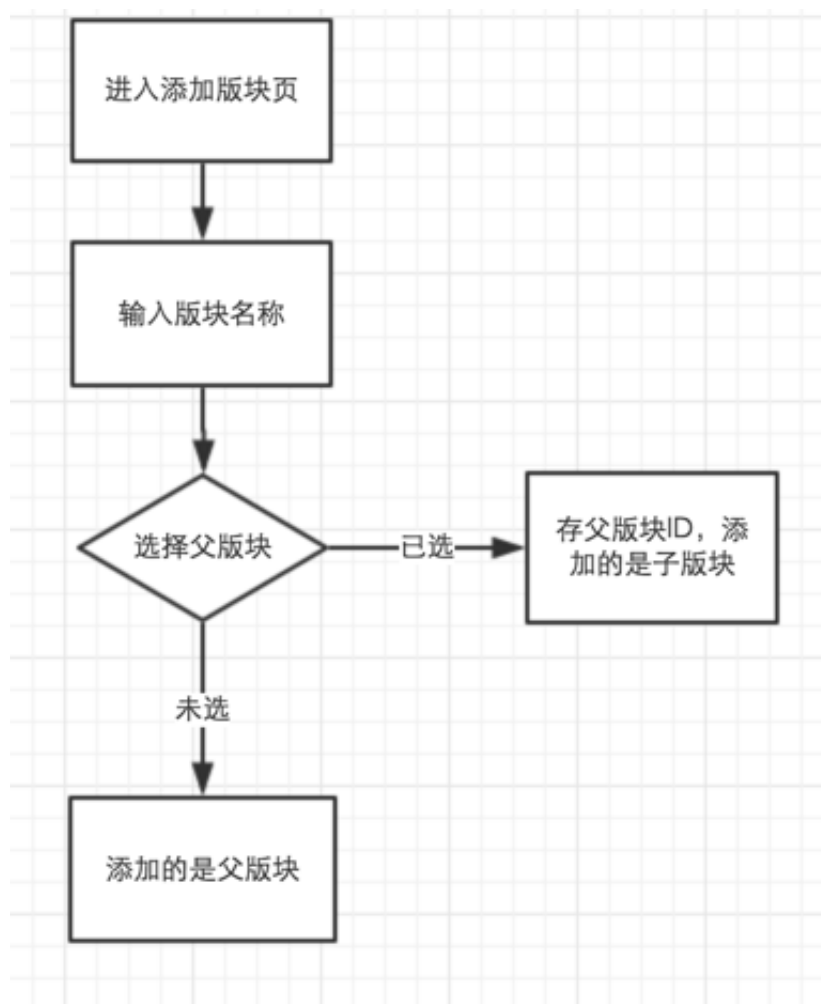
在百度贴吧中，我们可以创建不同的贴吧。不同的贴吧，在论坛中就是版块。

不同的版块（贴吧）讨论不同的内容。

论坛的版本在结构上与贴吧略有一些小的不同：

1. 贴吧是单个层级的，没有父板块
2. 论坛分为大版块和小版块。而小版本添加时有一个父版块，也就是小版本是属于哪个大版块的。例如电脑版块下，有电脑维修和软件故障。

- 版块管理流程图



##图解版本管理

1.如果是添加大版本在选择大版块是就不填写，这样就产生了一个大版本。如果我们是指定了大版块添加一个新版块，那么新版块就属于这个大版块

Discuz!
Control Panel

站点信息 | 用户管理 | 版块管理 | 帖子管理

管理版块

添加版块

添加版块

技巧提示

添加时不选择大版块即为添加大版块

版块名称

选择大版块

-不选择-

提交

2.展示版块

Discuz!
Control Panel

站点信息 | 用户管理 | 版块管理 | 帖子管理

管理版块

添加版块

版块管理

显示顺序

版块名称

2

PHP技术交流

4

内核源码

3

PHP框架

2

开源产品

1

进阶讨论

1

程序人生

3

求职招聘

2

经验分享

1

名人故事

提交

6.版主业务流程

- 版主业务说明

在真实的项目中一个人很难将所有的工作全部完成。需要其他人来协助它。

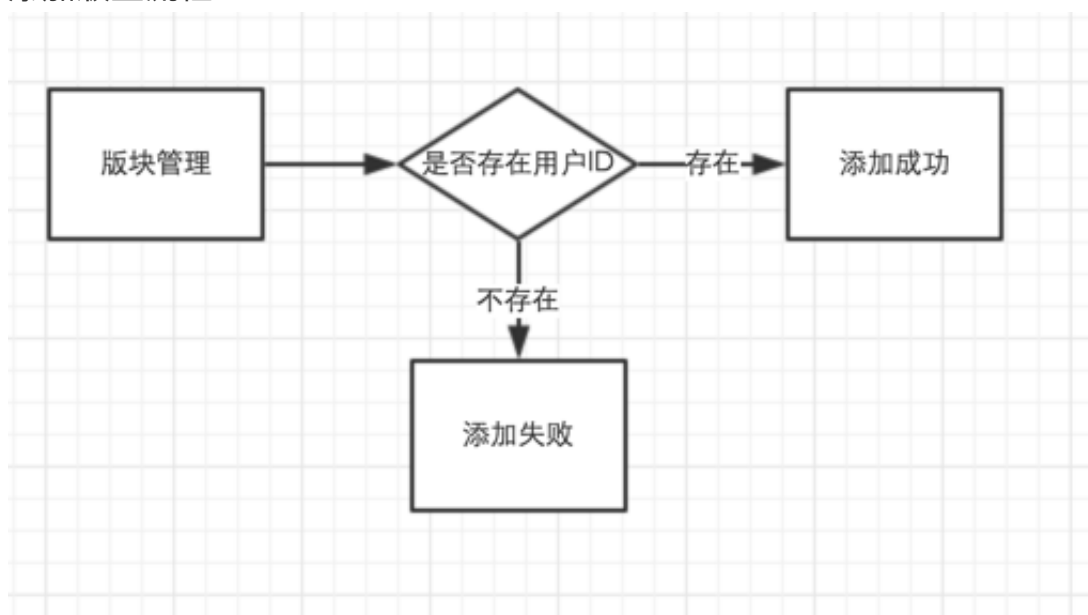
例如在贴吧中可以指定某个热心网友为某个贴吧的小吧主，协助完成某些工作。

论坛中也是，管理员不可能某个版块都能够有足够的精力管理。因此，它可以指定某个用户为版主。

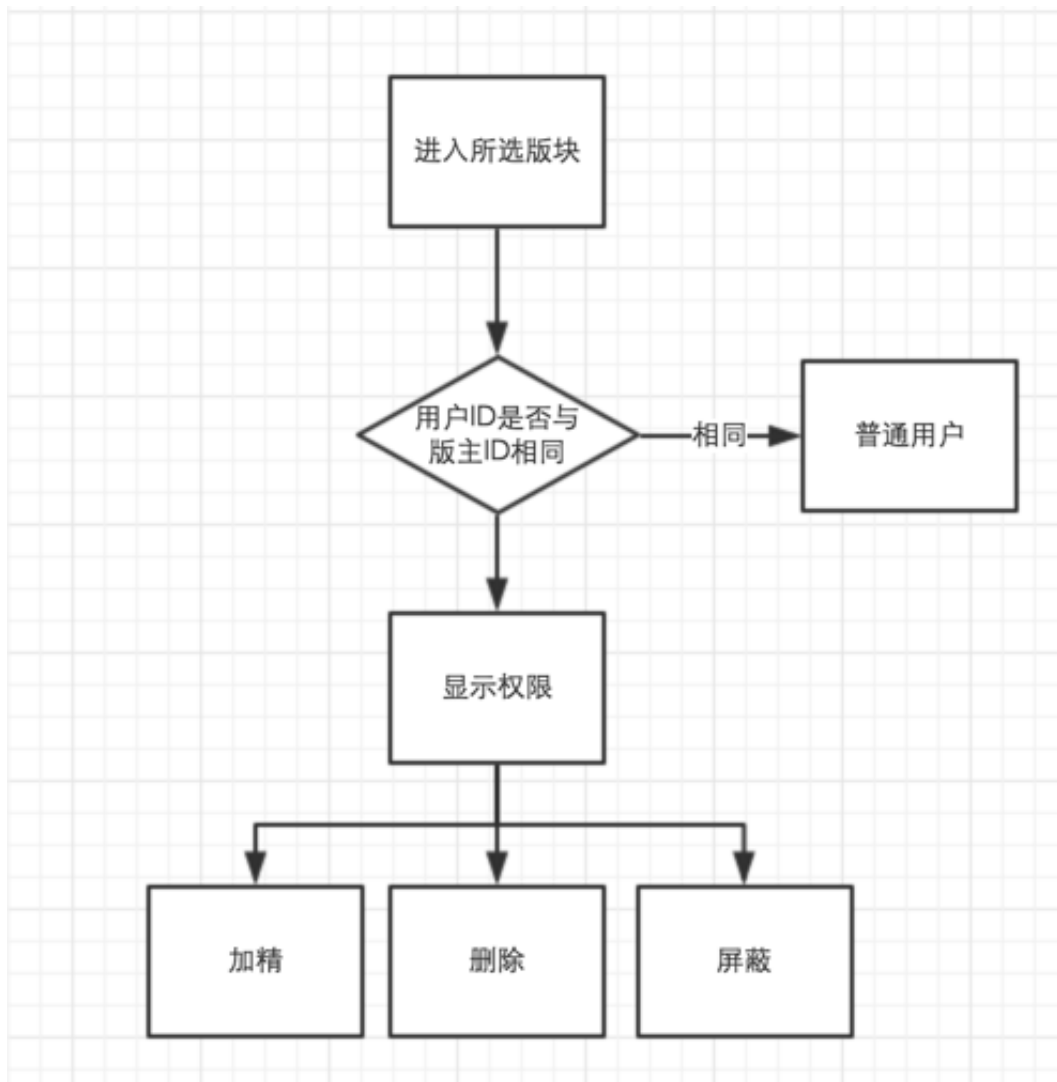
版主能够删掉这个版块的公告、违法信息屏蔽掉等功能。

因此，我们在这里也设计了一个版主功能。

- 添加版主流程



##版主管理流程



##图解版主管理

1.在后台管理界面中展示版本，在版主添写入写上用户的唯一ID

站点信息 | 用户管理 | **版块管理** | 帖子管理

版块管理

显示顺序	版块名称	版主
<input type="text" value="2"/>	<input type="text" value="PHP技术交流"/>	<input type="text" value="1,8"/>
<input type="text" value="4"/>	<input type="text" value="内核源码"/>	<input type="text" value="1,9"/>
<input type="text" value="3"/>	<input type="text" value="PHP框架"/>	<input type="text" value="1"/>
<input type="text" value="2"/>	<input type="text" value="开源产品"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="进阶讨论"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="程序人生"/>	<input type="text" value="1"/>
<input type="text" value="3"/>	<input type="text" value="求职招聘"/>	<input type="text" value="1"/>
<input type="text" value="2"/>	<input type="text" value="经验分享"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="名人故事"/>	<input type="text" value="1"/>

2.在前台展示这个用户为某个版块的版主

PHP技术交流		
内核源码 版主: test001, admin	11 / 10	已有业务换成pdo遇到的问题 1分钟前 admin
PHP框架 版主: test002, admin	0 / 0	从未
开源产品 版主: admin	0 / 0	从未
进阶讨论 版主: admin	0 / 0	从未
程序人生		

3.版主具有删除贴子等权限，而普通用户没有

论坛 > PHP技术交流 > 内核源码 > 已有业务换成pdo遇到的问题

[发帖](#)
[回复](#)

返回列表

查看: 4 | 回复: 1

admin

管理员
学士

已有业务换成pdo遇到的问题

发表于 1天前

楼主 电梯直达

公司的老项目从adodb改成pdo，产生了好多问题。升一个版本会遇到很多问题，有个预处理问题最严重。

pdo没有bigint的类型，预处理的时候会出现问题。

当pdo::attr_emulate_prepared这个参数 设置为false的时候才会出现这个问题。

`pdo::param_int (integer)`
represents the sql integer data type.

这个类型在mysql里面会超出int 整型所指定的范围。mysql存的是bigint，只能用param_str去代替。

这个事先没有意识到，紧急的处理是将bigint映射成pdo的param_str，不至于被转换成int。

回复

普通用户此处没有操作功能

admin

发表于 1天前

沙发

我可以选择一个内容进行回复

4.版主操作功能区展示



7.金币奖励和消耗流程

- 积分业务说明

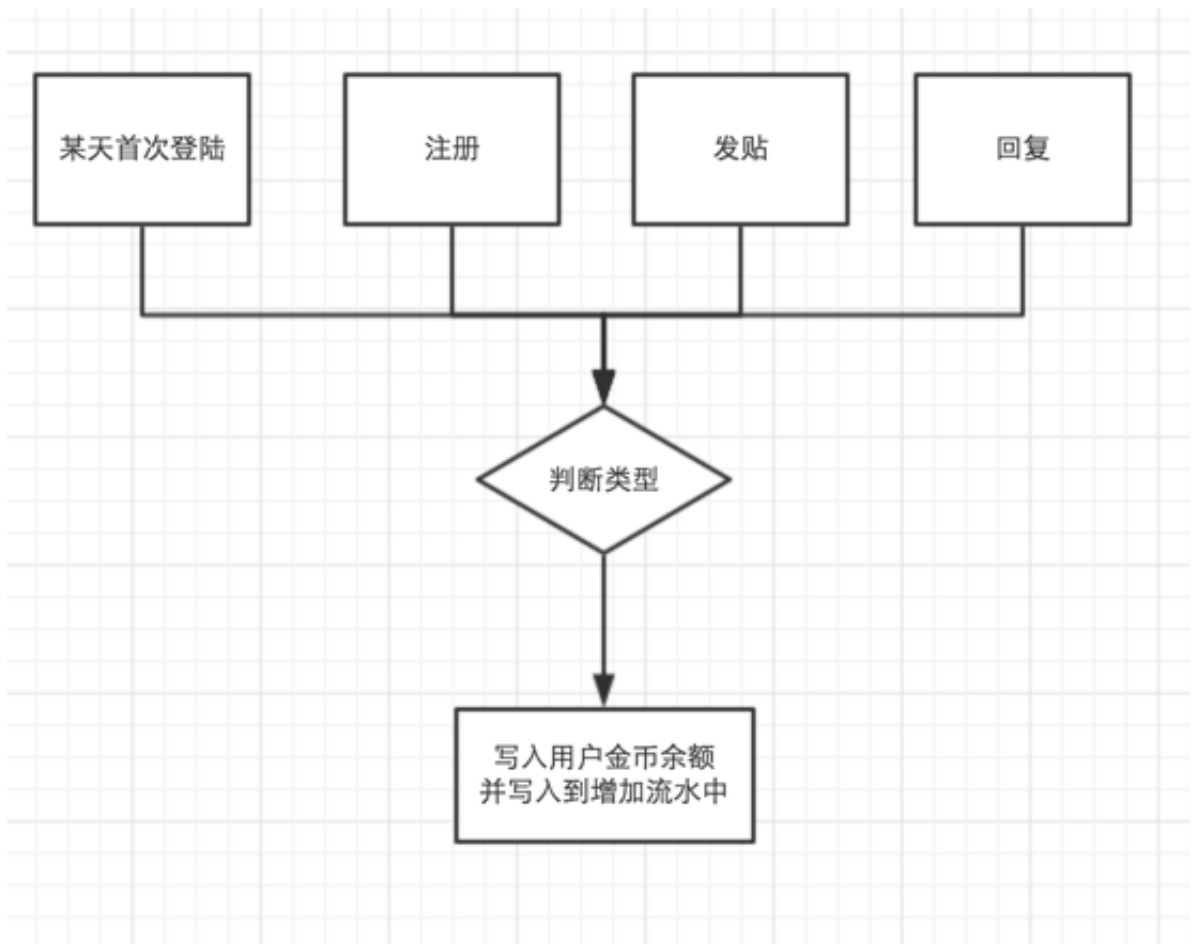
在需求部分我们讲解过积分（金币）的处理机制，机制如下：

注册：赠送50积分； 登陆：每天首次登陆赠送2积分； 发帖：赠送2积分；
回帖：赠送1积分；

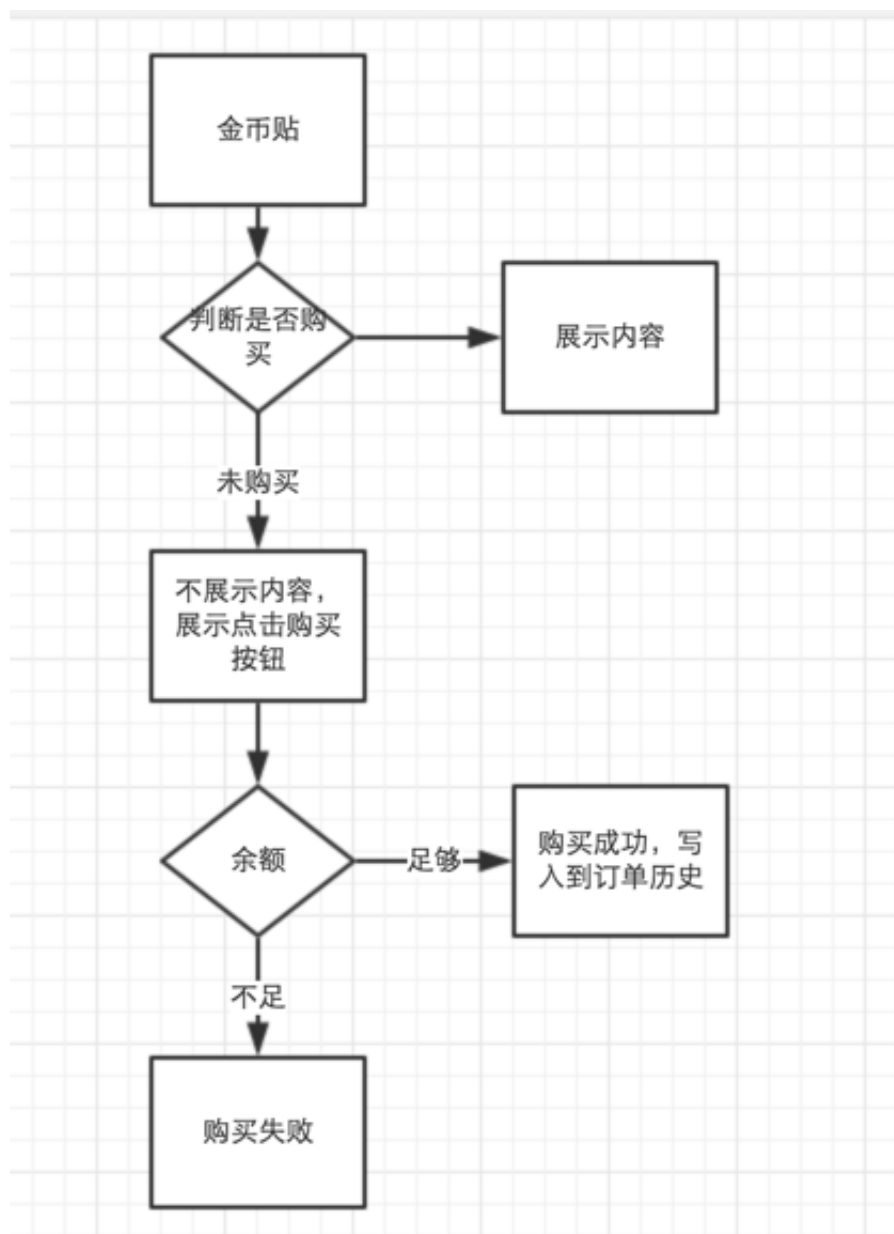
通过这种金币（积分）业务的处理，能够极大的丰富市场运营部门的活动的可用性，增强用户粘性，让用户觉得自己的发贴和回帖是有价值和意义的。

例如：每天首次登陆送2个积分，就像京东商城的京豆一样。每天登陆，我可以送用户金豆，让用户经常访问，增加消息和购物的冲动。

##增加金币流程图



- 消耗金币流程图



- 增加金币图例

在注册、回复成功和发贴成功等都会显示金币增加的相关图例。



帖子回复成功，请稍候.....

如果你的浏览器没有自动跳转，请点击此链接

回帖赠送 金钱+1



帖子发表成功，请稍候.....

如果你的浏览器没有自动跳转，请点击此链接

发帖赠送 金钱+2



感谢您的注册，现在将以会员身份登录站点，请稍候.....

如果你的浏览器没有自动跳转，请点击此链接

注册赠送 金钱+50

##展示金币贴图例

关于php你可能不知道的一php的事件驱动化设计

发表于 2015-10-08 10:46:58

本主题需向作者支付 **10 金钱** 才能浏览

购买主题

楼主 电梯直达

本主题已加入精华

回复

三、数据库表设计

1.数据库设计： E-R图

ER图概念化地构建实体间关系的模型，这使得它们区别于数据库模型图。ER图的理念是：项目所有参与者能理解ER图。ER图由不同实体类型、关系、特性和类型构成。实体是诸如用户的实际对象，有时更抽象，但必须有业务意义。特性用于描述实体，关系用于实体之间：

- (1) 实体：现实世界中的事物；
- (2) 属性：事物的特性；
- (3) 联系：现实世界中事物间的关系。

实体集的关系有一对一、一对多、多对多的联系。

2.数据库表及关系建立

1. 用户基本资料表 (bbs_user)

用户基本资料表 bbs_user						
字段名	数据类型	长度	主键	可空	默认值	备注
uid	int	11	是	否	无	AUTO_INCREMENT
username	char	16	否	否	无	账号
password	char	32	否	否	无	密码(md5加密)
email	char	30	否	否	无	邮箱
udertype	tinyint	2	否	否	无	账号类型(1=管理员, 0=用户)
regtime	int	12	否	否	无	注册时间
lasttime	int	12	否	否	无	最后登录时间
regip	int	12	否	否	无	注册IP
picture	varchar	255	否	否	public/images/avatar_blank.gif	头像
grade	int	10	否	是	0	积分
problem	varchar	200	否	是	NULL	找回密码问题
result	varchar	200	否	是	NULL	答案
realname	char	50	否	是	NULL	真实姓名
sex	tinyint	4	否	是	2	性别(1=男, 2=女)
birthday	varchar	20	否	是	NULL	生日
place	varchar	50	否	是	NULL	所在地
qq	char	13	否	是	NULL	QQ号码
autograph	varchar	500	否	是	NULL	个人签名
allowlogin	tinyint	2	否	否	0	是否允许登陆(0=允许, 1=不允许)

2.IP黑名单表 (bbs_closeip)

IP黑名单表 bbs_closeip						
字段名	数据类型	长度	主键	可空	默认值	备注
id	int	10	是	否	无	AUTO_INCREMENT
ip	int	10	否	否	无	IP地址(用到的函数: ip2long)
addtime	int	10	否	否	无	记录添加时间
overtime	int	10	否	是	NULL	IP限制的结束时间

3.友情链接表 (bbs_link)

友情链接表
bbs_link

字段名	数据类型	长度	主键	可空	默认值	备注
lid	smallint	6	是	否	无	AUTO_INCREMENT
displayorder	tinyint	2	否	否	0	排序(数字大排在前面)
name	varchar	30	否	否	无	名称
url	varchar	255	否	否	无	链接跳转地址
description	mediumtext	0	否	是	NULL	描述
logo	varchar	255	否	是	NULL	LOGO的URL地址
addtime	int	12	否	否	无	添加时间

4. 论坛版块表 (bbs_category)

论坛版块表
bbs_category

字段名	数据类型	长度	主键	可空	默认值	备注
cid	int	10	是	否	无	AUTO_INCREMENT
classname	varchar	60	否	否	无	版块名称
parentid	int	10	否	否	无	父级ID
classpath	char	20	否	是	NULL	关系
replycount	int	10	否	否	0	回帖数量
motifcount	int	10	否	否	0	帖子数量
compere	char	10	否	是	NULL	版主
classpic	varchar	255	否	否	public/images/forum.gif	版块ICON
description	mediumtext	0	否	是	NULL	版块描述
orderby	smallint	6	否	否	0	排序
lastpost	varchar	255	否	是	NULL	最后发表
namestyle	char	10	否	是	NULL	
ispass	tinyint	2	否	否	1	审核状态

5.帖子信息表 (bbs_details)

帖子信息表
bbs_details

字段名	数据类型	长度	主键	可空	默认值	备注
id	init	10	是	否	无	AUTO_INCREMENT
first	tinyint	1	否	否	0	帖子或回复 (1=帖子, 0=回复)
tid	int	10	否	否	无	帖子ID
authorid	int	10	否	否	无	发布人ID
title	varchar	600	否	否	无	帖子标题
content	mediumtext	0	否	否	无	帖子内容/回帖内容
addtime	int	12	否	否	无	发布时间
addip	int	12	否	否	无	发布人IP
classid	int	10	否	否	无	版块ID
replycount	int	12	否	否	0	帖子的回复数量
hits	int	12	否	否	0	浏览次数
istop	tinyint	2	否	否	0	是否置顶贴(1=是, 0=否)
elite	tinyint	2	否	否	0	是否精华贴(1=是, 0=否)
ishot	tinyint	2	否	否	0	是否热门贴(1=是, 0=否)
rate	smallint	3	否	否	0	帖子售价(积分)
attachment	smallint	3	否	是	NULL	帖子附件
isdel	int	2	否	否	0	是否放入回收站
style	char	10	否	是	NULL	帖子标题颜色(css样式)
isdisplay	int	2	否	否	0	是否屏蔽回复内容(1=是, 0=否)

6.帖子订单表 (bbs_order)

帖子订单表
bbs_order

字段名	数据类型	长度	主键	可空	默认值	备注
oid	int	10	是	否	无	AUTO_INCREMENT
uid	int	11	否	否	无	用户ID
tid	int	11	否	否	无	帖子ID
rate	int	11	否	否	无	价格
addtime	int	11	否	否	无	创建时间
ispay	tinyint	2	否	否	0	是否已支付(0=未支付, 1=已支付)