中间件实现技术

实践案例

**知识管理**

**Wiki系统**

专业：软件工程

学号：2016160191

姓名：陈永强

工作单位：明基逐鹿软件（苏州）有限公司

**版权说明：**

本案例仅限于教学实践案例使用，案例中不对任何相关软件进行评判。

## 摘要

本案例主要对利用开源软件WordPress进行知识管理Wiki系统搭建进行说明，包含系统说明、功能设计、组件选择、搭建配置及系统扩展性进行说明，着眼于目前知识管理平台的现状需求，搭建可用于进行有效信息挖掘，人工智能学习测试的平台。

此实践系统基于简单易用开源的WordPress博客系统，采用Flat-Sky主题设计，使用YasaWiki插件进行wiki功能页面管理，使用Pst Type Switcher插件实现文章内容转wiki页面功能，使用User Role Editor插件实现账户权限管理。

整体实现用户注册、发表博客文章、讨论、博主贡献文章，博文转为Wiki页面、Wiki管理及编辑的完整知识转化体系。

关键词：博客文章、知识管理、Wiki、账户权限管理

# 知识管理Wiki系统

目前互联网技术日新月异，对于编程开发人员的要求也越来越高，为弥补技术差异，一方面公司不断从外界招募有新技术资格的员工入职，另一方面也亟待提高内部技术的进化发展速度。

### 一、平台搭建

因此搭建简单、易用的轻量级知识管理Wiki系统已经成为非常必要的任务，由于最近几年开源软件和中间件技术的日新月异导致目前这个任务变得非常容易。由于不是专业性平台，所以我们在选择平台组件时选择了：Winserver2008R2 64位系统+IIS7这样架构；考虑到费用及便宜开发的目的，在WEB开发上选择了PHP5.6，在后台数据库上选择MYSQL这样的组合，另外为了更好的在IIS上管理PHP，使用了PHPManagerForIIS组件，需要注意的是该组件只找到IIS7平台上的版本。

在代码管理上，由于其简单已用且便于集成开源环境，选取了GitHub进行代码管理。而整个站点使用了WordPress作为CMS管理平台来使用，主要是WordPress的易扩展性和相比国内其他一些CMS软件而言结构比较简单。

整体环境安装部署次序是：

首先安装系统及更新，添加IIS7

安装并配置PHP5.6

安装并配置MYSQL数据库

在本地环境下载并安装GITHUB客户端，创建项目

在本地将下载到的最新版本WordPress站点并解压放置到项目文件夹内，创建项目，并将代码推送到GITHUB仓库

在服务器上安装GITHUB客户端，将项目代码CLONE到服务器

在IIS中将该目录配置为站点，安装配置WordPress及数据库链接，启动网站

将最新配置情况Push到项目中

至此整个平台就已经搭建完毕，我们可以在WEB端进行代码配置调试，或者在程序员开发环境下调试，通过github推送到服务器。

### 二、站点搭建

#### 1．Wiki功能

知识管理Wiki系统主要包含四方面功能文章编辑、文章讨论及修订、wiki编辑、文章转化为wiki。其中文章编辑、文章讨论及修订为WordPress自带功能。我们主要需要增加wiki编辑、文章转化为wiki的功能，本着尽量不要重复造轮子的原则，我们使用WordPress中的Yada Wiki插件，选择该插件主要是该插件简洁易用，便于我们进行二次开发。

通过后台的插件安装功能安装该插件（如图1）后我们就可以在后台看到Wiki管理功能（如图2），包括Wiki页面的新增、编辑及类别添加。

图1

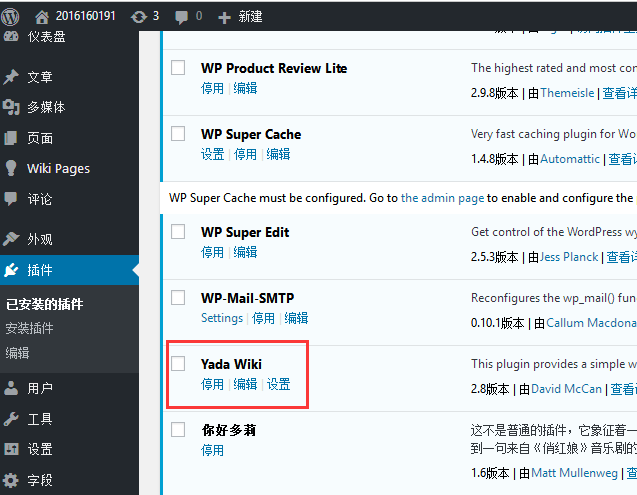


图2



由于覆盖主题后插件不能和主题页面编辑很好的吻合，导致只能通过工具栏就进行wiki页面的编辑，我们在对应主题的页面body元素中增加编辑链接，如图3所示（该处实际应该是php动态代码前半段是get\_bloginfo,后面是动态获取postid然后进行串联）。因此用户可以通过页面左上角的EditWiki链接跳转到编辑页面进行Wiki编辑，如图4所示。

图3

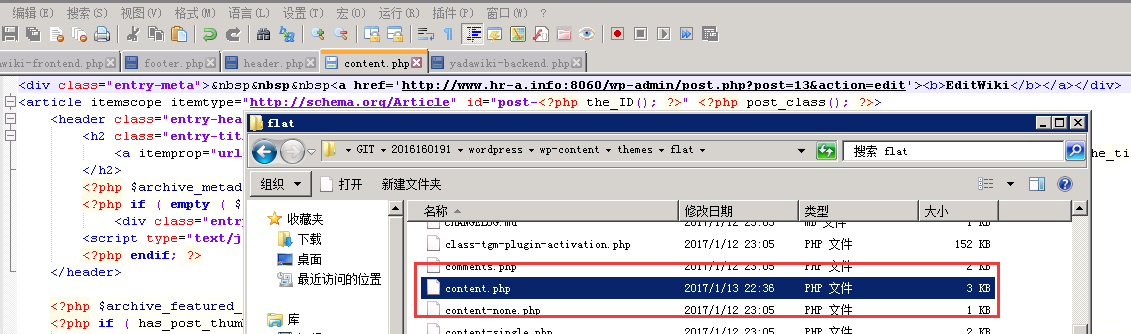
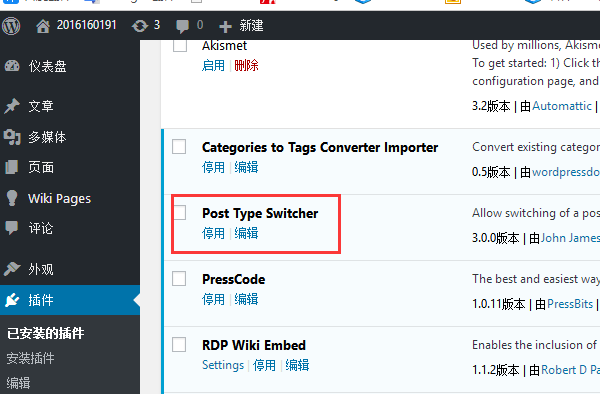


图4



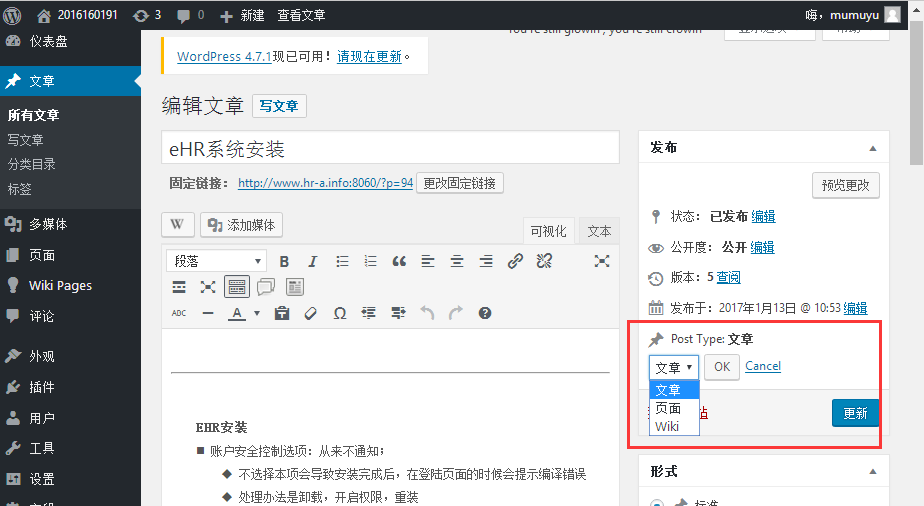
第二是页面转化功能，系统设定为文章博客为作者私有，不能被公共编辑，所以我们增加了PostTypeSwitcher插件来允许作者将文章分享成Wiki供大家编辑。如图5所示，PostTypeSwitcher功能原用于转换文章与页面，而wiki页面属于页面的一种，所以我们可以使用该功能。

图5



在文章/wiki编辑页面上，作业可以选择变更页面的类型，将文章共享为wiki，需要注意的是，由于文章与wiki的分类目录结构不一致，更替目录时需要先更新类别，保存更新页面后才能选择对应的分类目录。

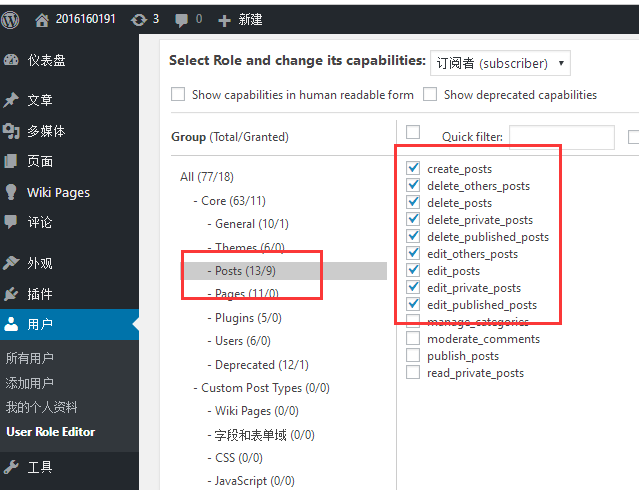
图6



#### 2、用户权限管理

完成Wiki功能后，我们发现权限分级管理与标准模板有了出入，我们目前有两类用户：访客、会员，访客无需拥有编辑权限，但会员应当在原有权限的基础上增加wiki编辑功能。所以我们增加了UserRoleEditor插件进行权限管理。

当然UserRoleEditor插件与我们目前的Wiki插件并不完全吻合，如果只给Wiki编辑的功能，打开编辑页面的时候会报无权限错误。分析代码后我们发现，WordPress在判断页面编辑权限的时候检查的是Post（页面）属性的权限，因此我们将Post编辑和新建权限给到了订阅者（会员）角色上。至此能够实现整个Wiki管理系统全部逻辑。



### 三、站点完善

为完善站点，我们为站点增加了如下其他功能：

通过插件WP-Mail-SMTP增加邮件发送服务

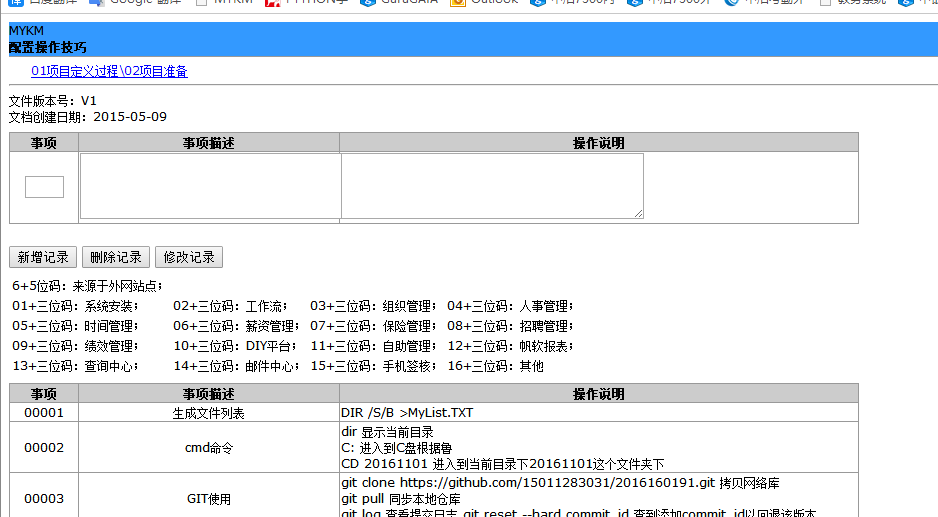
设置静态首页增加Wiki全功能导航图



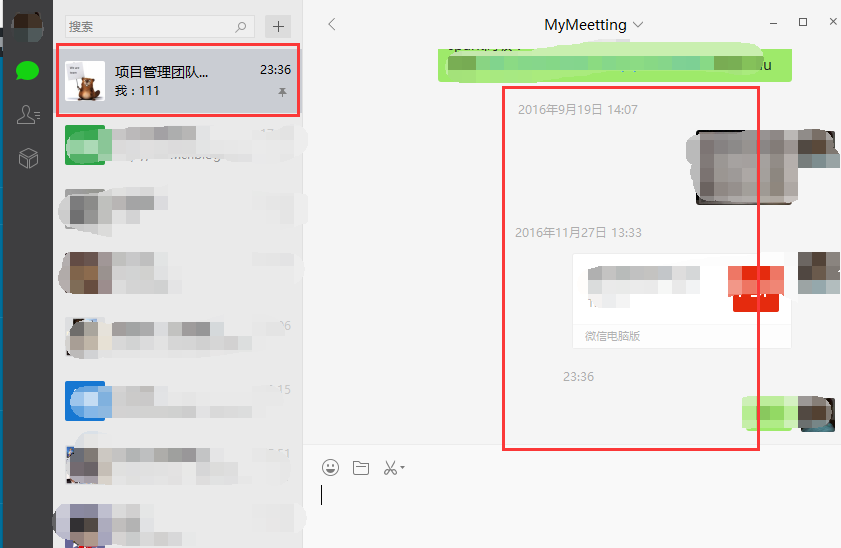
### 四、扩展功能

由于站点搭建时间比较仓促，且改思路从.net平台迁移到php上来，仍然有很多部分待完善，后续将持续进行改进：

1. 原页面添加是固定框架，以增加条目的形式进行页面编辑，将条目内容传入到数据库表内，便于进行内容拆分，持续改进内容首先增加就是这样的一个wiki模板页；



1. WIKI插件没有在权限管理插件中生成权限点，后续增加权限点，且需要在各个页面增加分列对wiki和常规页面的权限识别
2. 增加爬虫功能主动以相关主题从外站抓取相关知识并整体成wiki格式，管理员进行审核编辑后发布到wiki页面上
3. 与微信企业号进行绑定，对内容发布页面中的数据进行抓取，提取关键词如：“提交”关键词中相关内容到wiki审核页面，管理员进行审核编辑后发布到wiki页面上



### 五、相关附件

本案例相关附件为：WordPress站点源代码一份