

使用 Nodejs+Mongodb+Vue+Express 创建重写个人博客

一、数据库结构

1. Users 数据表 // 存放用户信息及管理多用户
2. Posts 数据表 // 存放发表的博客
3. Comments 数据表 // 存放评论
4. Options 数据表 // 存放站点信息,及一些自定义信息

Users:结构

_id:由 mongodb 自动生成
Username:用户名字段
Password:密码字段
Nick_name:网名
Signature:个性签名
Website:个人网站
QQ:用户 qq
Wechat:用户微信
Email:用户邮箱
Authority:用户权限(阅读,发表,修改,root)
Create_at:创建时间
Update_at:信息更新时间
Logged_at:最近登陆时间

Posts 结构

_id:由 mongodb 生成
Post_title:博文标题
Post_author:博文作者(用户表_id 字段)
Post_content:博文正文
Post_summary:博文摘要
Post_category:博文分类
Post_tags:博文标签
Post_status:博文状态(已发布,草稿,回收站,隐藏,置顶)
Post_comments_status:博客评论状态(开启/关闭)
Create_at:博文创建时间

Update_at:博文更新时间

Comments 结构

_id:由 mongodb 自动生成
Comments_post:被评论的博文(posts 表_id)
Comments_content:评论内容
Comments_author:评论人(users 表_id)
Comments_email:评论者邮箱(适用于未登录评论者)
Create_at:评论创建时间

Options 结构

_id:由 mongodb 自动生成
Option_name:自定义项名称
Option_content:自定义项内容
Option_chage:是否可修改
Create_at:创建时间
Update_at:更新时间

二、开发文档结构

1. modules 文件夹 // 数据模型,对数据库的所有操作
2. Node_modules // 开发依赖的模块
 - i. Express
 - ii. Mongoose
 - iii. Formidable
 - iv. Express-session
3. routers 文件夹 // 路由处理,负责输出对应的 json
4. public 文件夹 // 静态资源文件,js,css,images
 - i. Views // vue 页面呈现
5. Tets 文件夹 // 测试文件
6. App.js // 启动文件
 - i. Crypto // 原生加密模块
7. Package.json // 依赖

路由展示设计:(全站 ajax)(get)

前台:

首页 /
文章页 /post/_id
登录页 /login
注册页 /sign

后台:

首页 /admin
用户信息 /admin/_id
文章发表页 /admin/post
文章编辑页: /admin/post/_id
评论列表页: /admin/comments_list
文章列表页 /admin/posts_list

路由处理设计:(全站 ajax)

注册逻辑 post /sign_check
登录逻辑 post /login_check
发表文章 /post_publish
更新文章 /post_update
删除文章 /post_delete
隐藏/显示文章 /post_hidden

处理文章到回收站 /post_recycle
文章列表 /posts_list
评论发表 /comments_publish
评论列表 /comments_list
更新个人信息 /admin/user_update
站点等自定义信息 /admin/option_update

三 前端页面布局

四 上线测试

1. 服务器安装配置 mongodb /nginx /node
2. Git 代码到服务器
3. Nginx 配置反向代理到 443 ssl 端口
4. 测试连接