Headless CMS

小微项目的业务中台解决方案

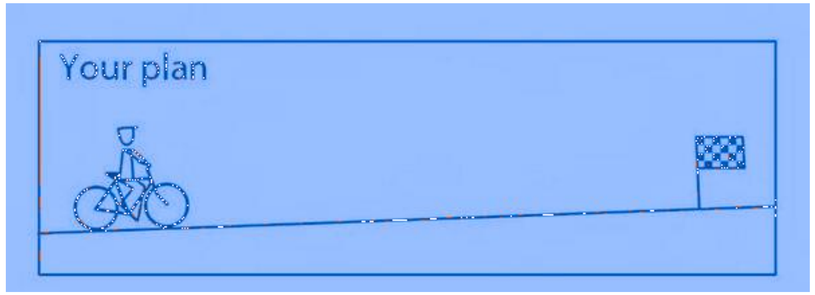
——by 张云龙

一个程序员的成长之路

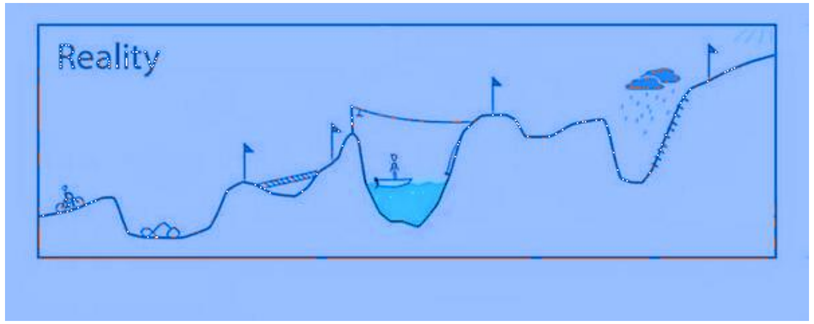
曾为『独立做事』做的那些准备

- 高效的前端集成解决方案
- 带过百人规模团队的管理经验
- 处理百万并发在线的架构设计
- 真正的全栈，从开发到运维，从前端到后端


不要给我说什么
底层原理、框架内核！
老夫敲代码就是
一把 梭 ！
复制！
粘贴！
拿起键盘就是
干！



理想很美好，现实很骨干




成本
收益

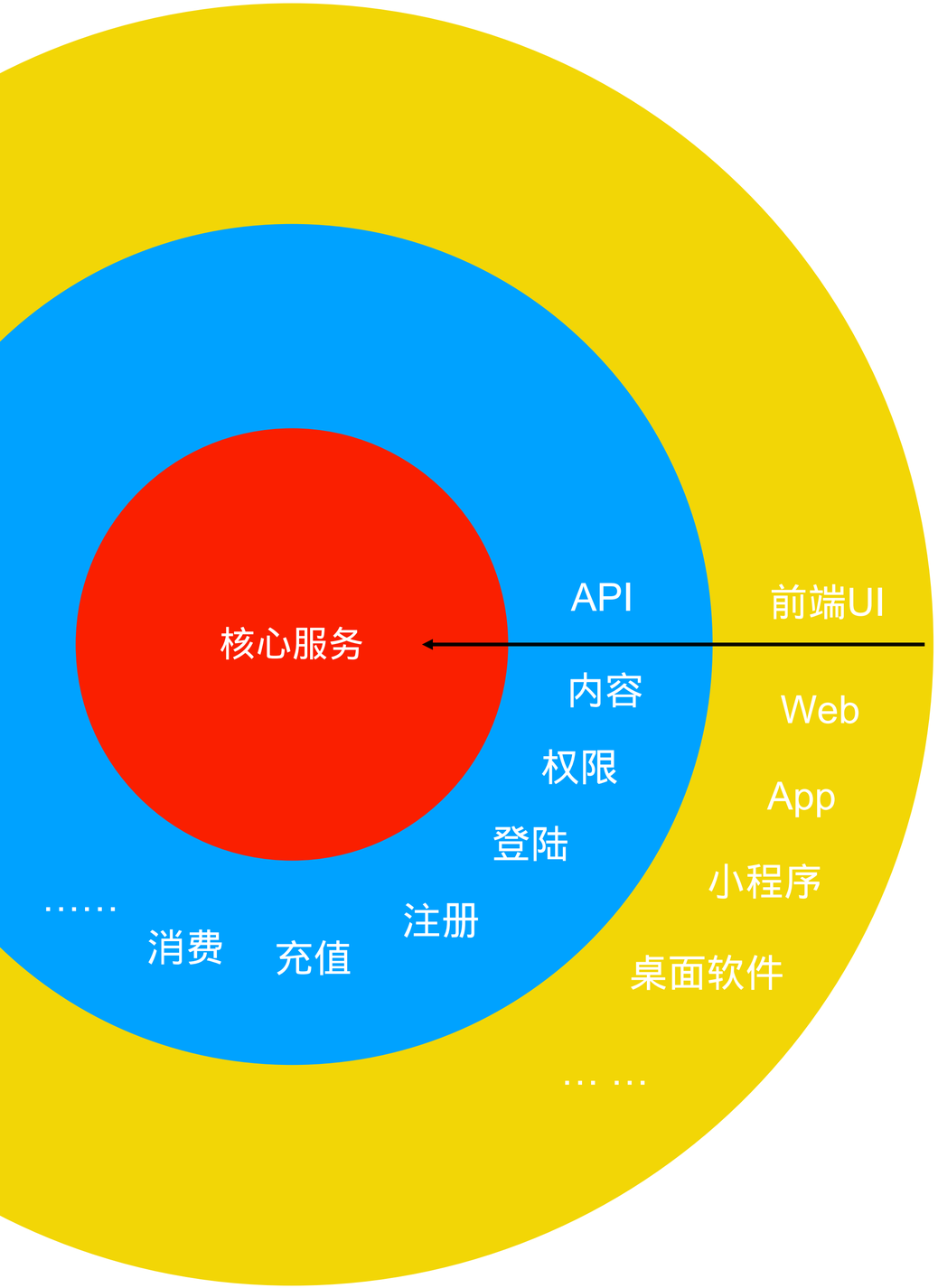
找到行业缺口，用最低的成本和最快速的速度验证市场


天下武功，无坚不破，唯快不破


活着

研发每款产品都需要经历的流程

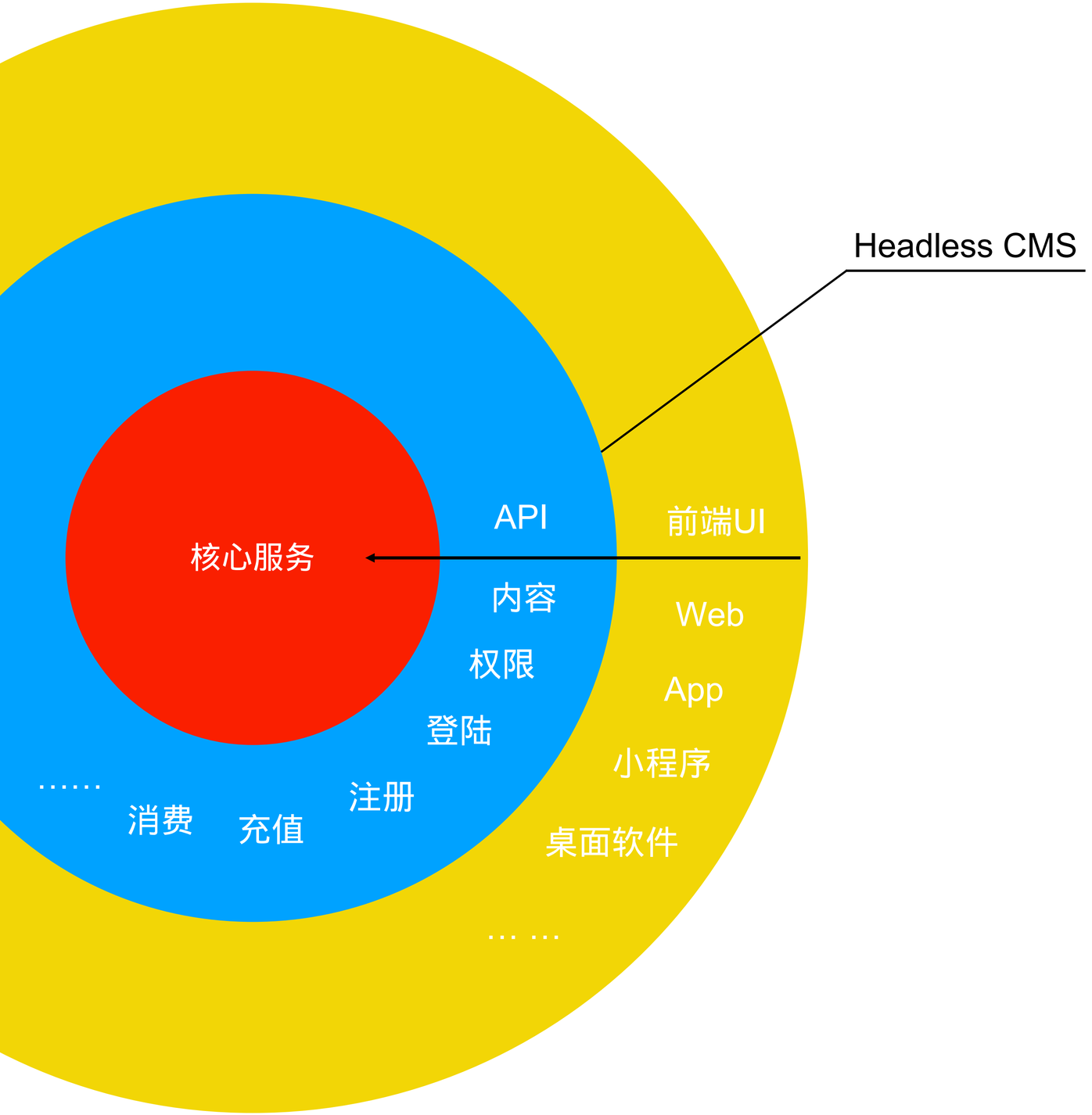
- 数据库设计
- 编写Model层
- 封装Service
- 编写API
- 定义Router
- 访问权限
- 注册/登陆
- 充值/消费
- 前端联调
- 上线运营



研发每款产品都需要经历的流程

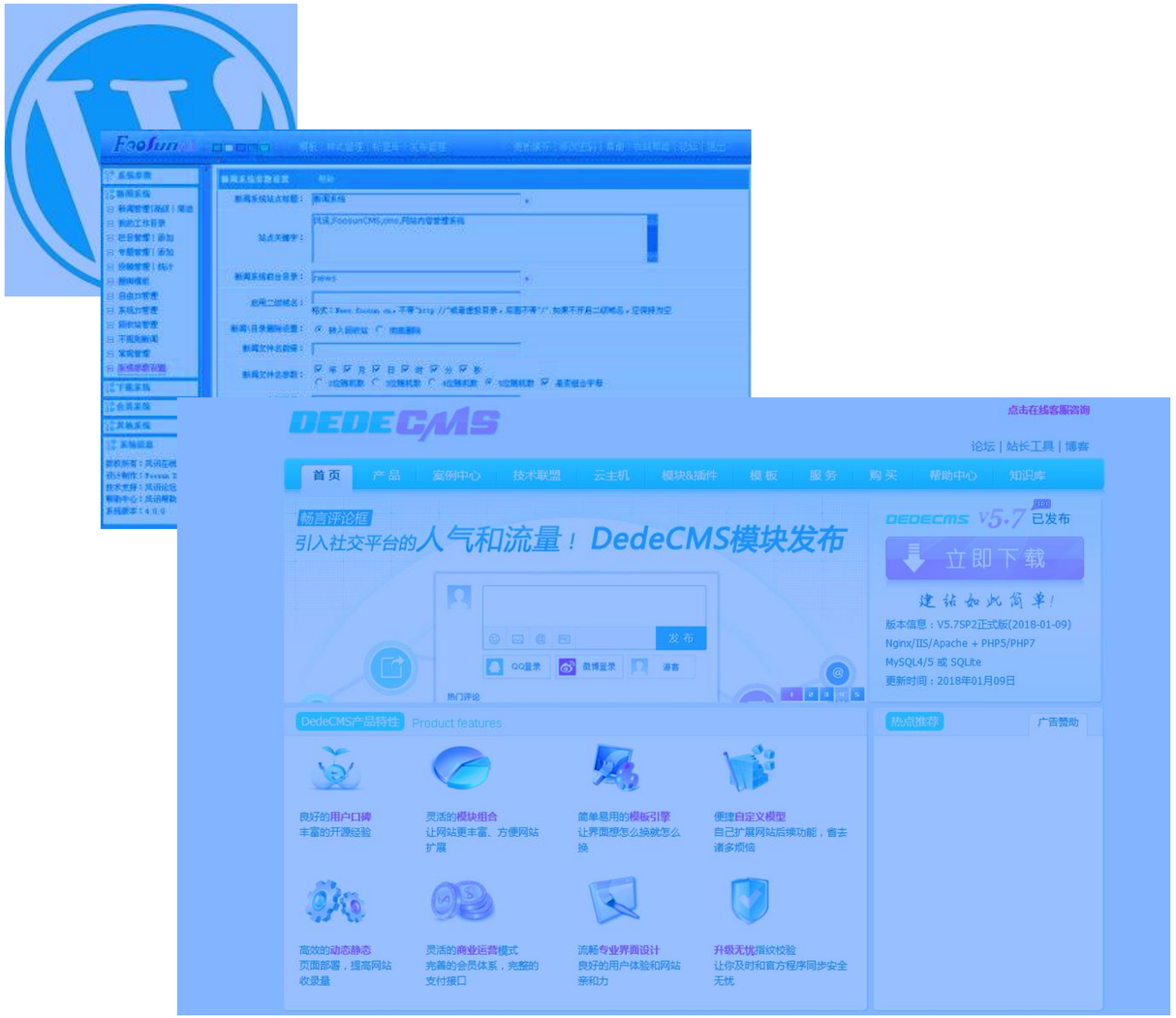
- 数据库设计
- 编写Model层
- 封装Service
- 编写API
- 定义Router
- 访问权限
- 注册/登陆
- 充值/消费
- 前端联调
- 上线运营


这都9102年了，
难道每次都要从0开始？



研发每款产品都需要经历的流程

- 数据库设计
- 编写Model层
- 封装Service
- 编写API
- 定义Router
- 访问权限
- 注册/登陆
- 充值/消费
- 前端联调
- 上线运营


DEDECMS
点击在线客服咨询
论坛 | 站长工具 | 博客
首 页
产 品
案例中心
技术联盟
云主机
模块&插件
模 板
服 务
购 买
帮助中心
知识库
畅言评论框
引入社交平台的人气和流量！ DedeCMS模块发布
发 布
QQ登录
微博登录
游客
热门评论
DEDECMS v5.7 已发布
立即下载
建站如此简单！
版本信息：V5.7SP2正式版(2018-01-09)
Nginx/IIS/Apache + PHP5/PHP7
MySQL4/5 或 SQLite
更新时间：2018年01月09日
DedeCMS产品特性 Product features
良好的用户口碑
丰富的开源经验
灵活的模块组合
让网站更丰富、方便网站扩展
简单易用的模板引擎
让界面想怎么换就怎么换
便捷自定义模型
自己扩展网站后续功能，省去诸多烦恼
高效的动态静态
页面部署，提高网站收录量
灵活的商业运营模式
完善的会员体系，完整的支付接口
流畅专业界面设计
良好的用户体验和网站亲和力
升级无忧指纹校验
让你及时和官方程序同步安全无忧
热点推荐
广告赞助


DEDECMS


这都9102年了，还在研究CMS？

我需要什么

- 一套数据到API的自动生成系统
- 为产品运营人员提供数据管理后台
- 自带权限管理，适应大部分业务场景
- 自带用户体系，灵活的注册、登陆方式
- 面向多终端，快速接入，减少开发
- 方便扩展，没有太多约束

The “head” in “headless CMS” refers to the frontend. A headless content management system consists primarily of an API as well as the backend technology required to store and deliver content. The headless approach allows developers to provide content as a service, abbreviated as CaaS, which simply means that content storage and delivery are handled by separate software.

–Cody Arsenault

~~Headless CMS~~ → API First CMS

strapi

https://strapi.io/

Keep control. Deliver faster.

Building self-hosted, customizable and performant Content API has never been easier.

Open source

Free and open source, forever. The entire codebase is available on GitHub and is maintained by hundreds of contributors.

Customizable

Each project requires specific requirements. Easily customize the admin panel as well as the API.

100% JavaScript

One language to rule them all. Use JavaScript everywhere: both for your front-end and your Headless CMS.

Self-hosted

Security is crucial for companies. Host your data safely, on your own servers. GDPR compliant.

RESTful or GraphQL

Consume the API from any client (React, Vue, Angular), mobile apps or even IoT, using REST or GraphQL.

Extensible by design

Plugins system included. Install auth system, content management, custom plugins, and more, in seconds.

See all features

“是时候表演真正的技术了。

5分钟上手

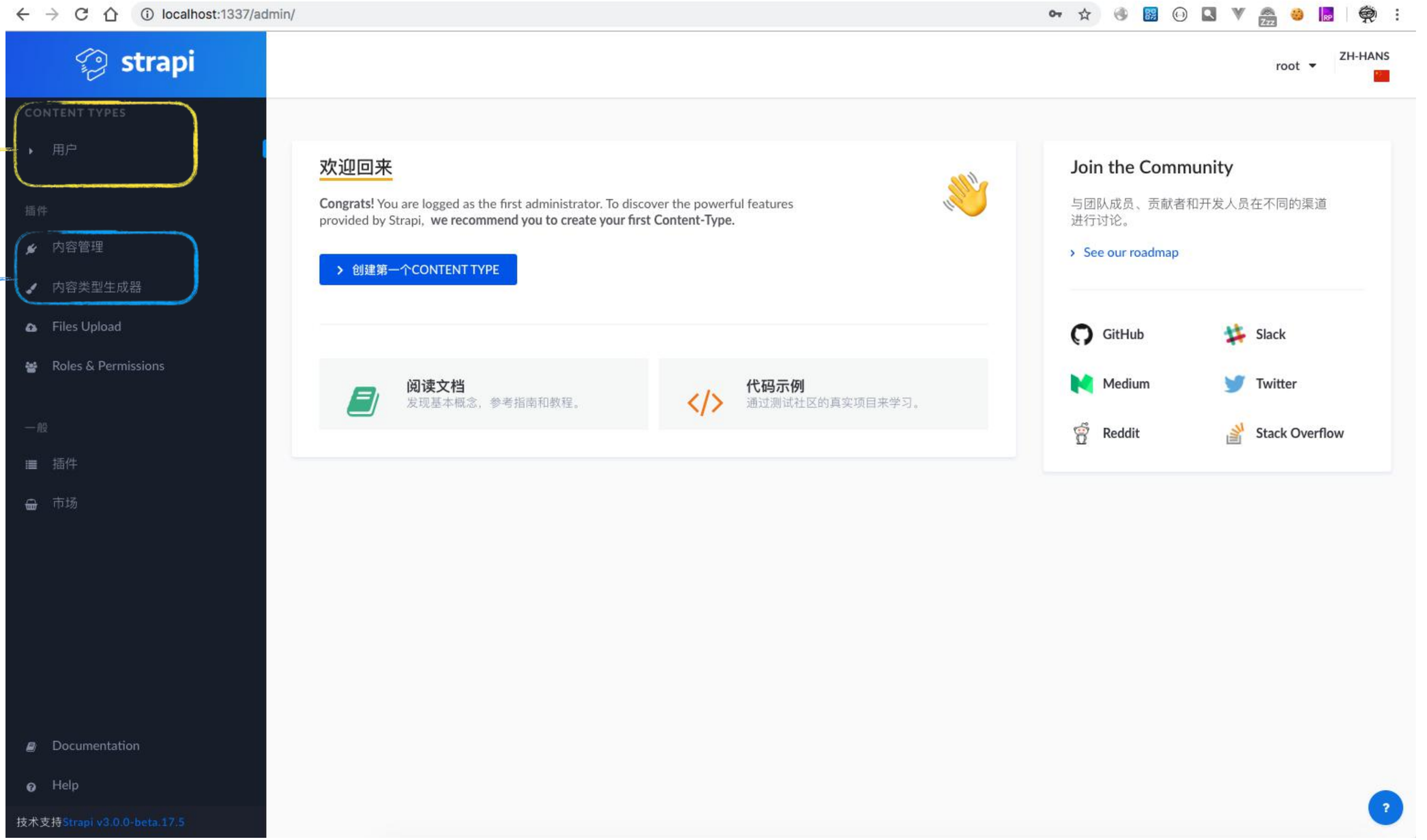
第一步：安装

```
yarn create strapi-app my-project --quickstart
# or use npx
npx create-strapi-app my-project --quickstart
# start
cd my-project
yarn develop
```

- Default database client: sqlite
- Filename: .tmp/data.db

数据管理

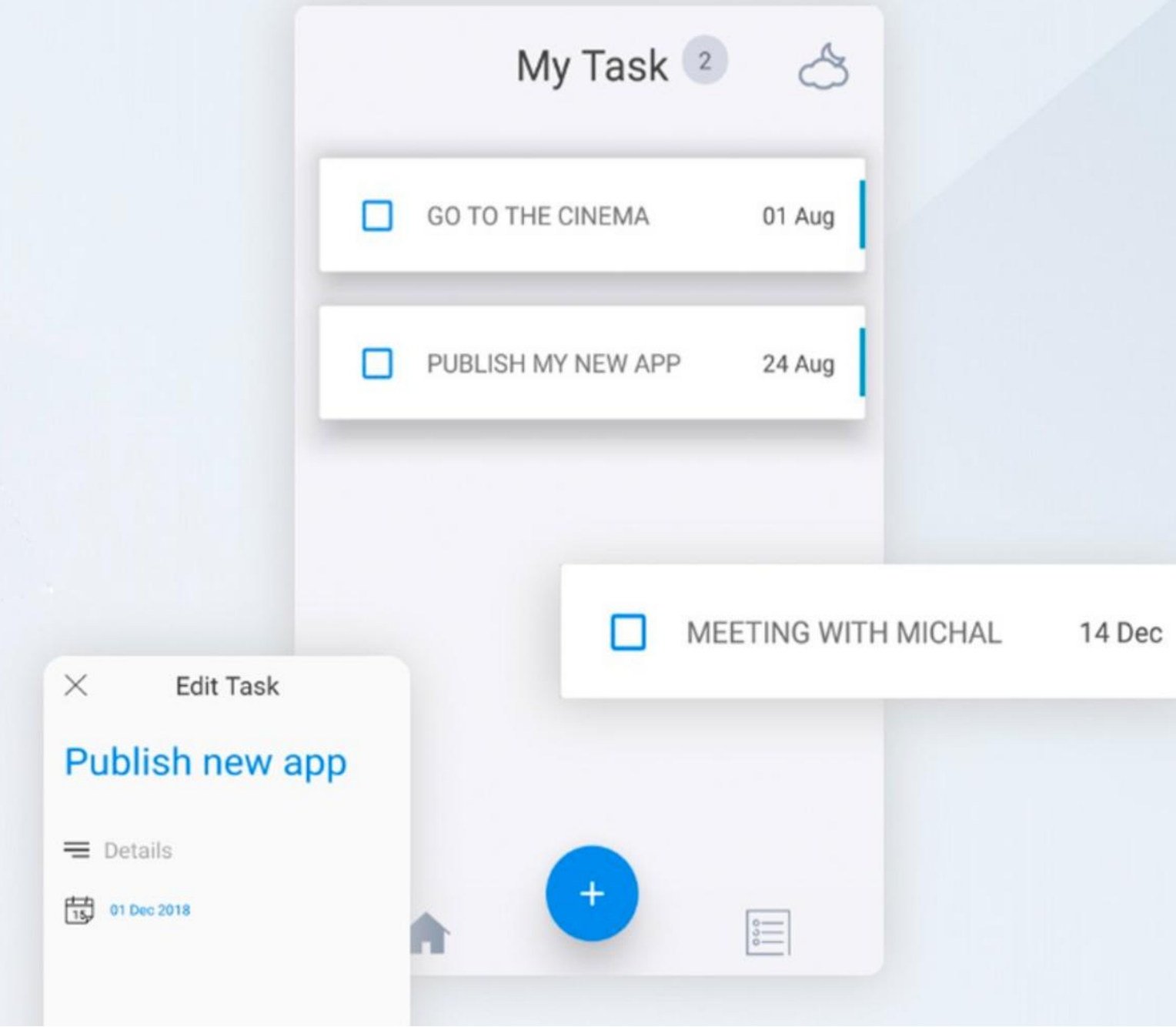
数据类型定制


localhost:1337/admin/
strapi
root
ZH-HANS
CONTENT TYPES
用户
插件
内容管理
内容类型生成器
Files Upload
Roles & Permissions
一般
插件
市场
Documentation
Help
技术支持Strapi v3.0.0-beta.17.5
欢迎回来
Congrats! You are logged as the first administrator. To discover the powerful features provided by Strapi, we recommend you to create your first Content-Type.
创建第一个CONTENT TYPE
阅读文档
发现基本概念，参考指南和教程。
代码示例
通过测试社区的真实项目来学习。
Join the Community
与团队成员、贡献者和开发人员在不同的渠道进行讨论。
See our roadmap
GitHub
Slack
Medium
Twitter
Reddit
Stack Overflow

目录结构


资源管理器
MY-PROJECT
.cache
.tmp
api
.gitkeep
build
config
extensions
node_modules
public
.editorconfig
.eslintignore
.eslintrc
.gitignore
favicon.ico
package.json
README.md
yarn.lock
业务相关代码
配置目录
扩展目录
静态资源目录

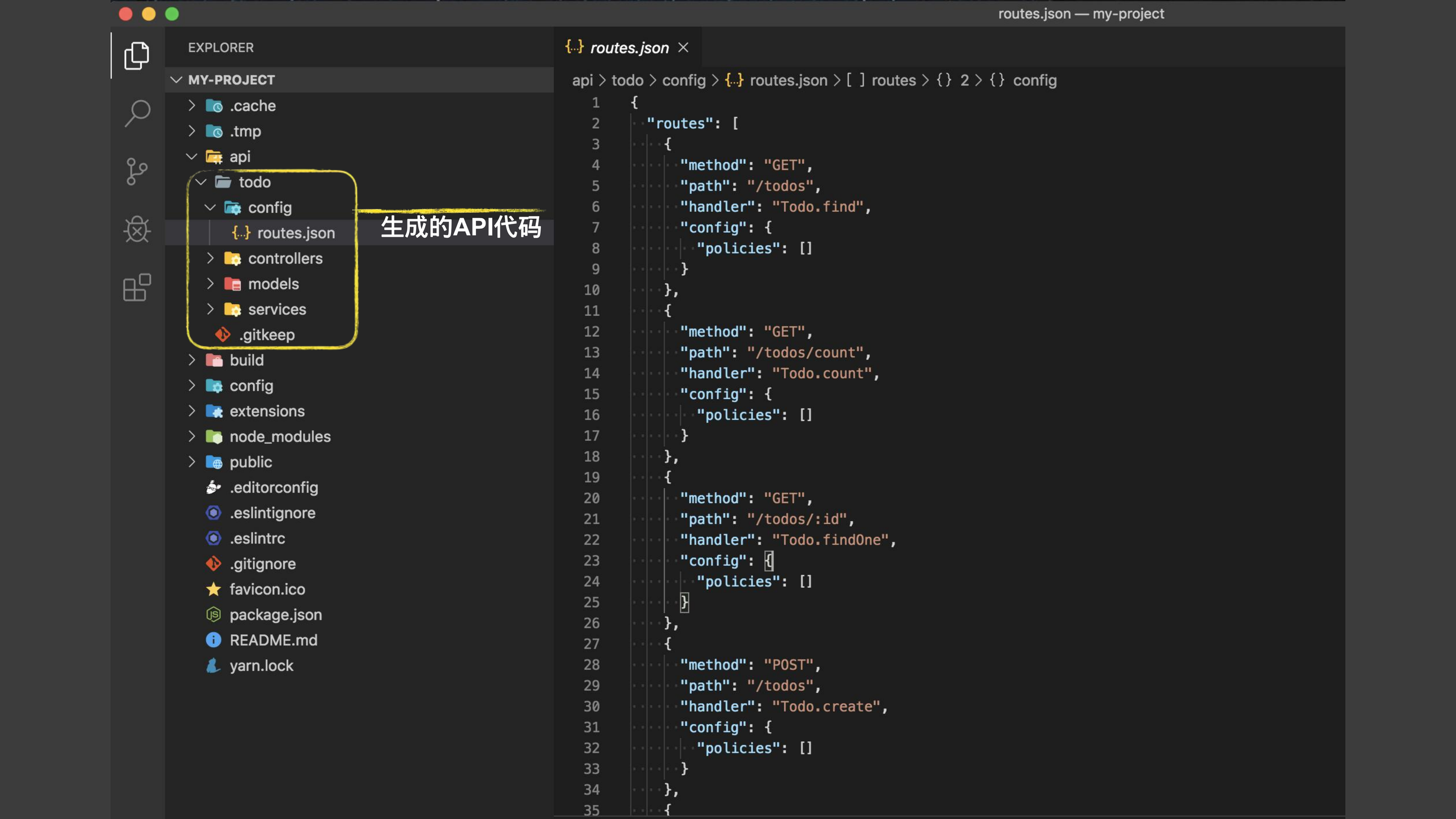
DEMO
Todo List



Last login: Sat Nov 16 01:45:49 on ttys002

FBI WARNING

Federal Law provides severe civil and criminal penalties for
the unauthorized reproduction, distribution, or exhibition of
copyrighted motion pictures (Title 17, United States Code,
Sections 501 and 508). The Federal Bureau of Investigation
investigates allegations of criminal copyright infringement
(Title 17, United States Code, Section 506).


routes.json — my-project
EXPLORER
MY-PROJECT
.cache
.tmp
api
todo
config
routes.json
controllers
models
services
.gitkeep
build
config
extensions
node_modules
public
.editorconfig
.eslintignore
.eslintrc
.gitignore
favicon.ico
package.json
README.md
yarn.lock
生成的API代码
routes.json
api > todo > config > routes.json > [ ] routes > { } 2 > { } config
{
"routes": [
{
"method": "GET",
"path": "/todos",
"handler": "Todo.find",
"config": {
"policies": []
}
},
{
"method": "GET",
"path": "/todos/count",
"handler": "Todo.count",
"config": {
"policies": []
}
},
{
"method": "GET",
"path": "/todos/:id",
"handler": "Todo.findOne",
"config": {
"policies": []
}
},
{
"method": "POST",
"path": "/todos",
"handler": "Todo.create",
"config": {
"policies": []
}
},
{

HTTP请求	路径	示例	说明
GET	/{content-type}	/todos	获取todo数据的列表
GET	/{content-type}/count	/todos/count	获取todo数据的数量
GET	/{content-type}/:id	/todos/1	获取id为1的todo数据
POST	/{content-type}	/todos	添加一个todo数据
DELETE	/{content-type}/:id	/todos/2	删除id为2的todo数据
PUT	/{content-type}/:id	/todos/3	更新id为3的todo数据

RESTFul API

RESTFul API 参数 —— Filter

Filters

Filters are used as a suffix of a field name:

- Not suffix or `eq` : Equals
- `ne` : Not equals
- `lt` : Lower than
- `gt` : Greater than
- `lte` : Lower than or equal to
- `gte` : Greater than or equal to
- `in` : Included in an array of values
- `nin` : Isn't included in an array of values
- `contains` : Contains
- `ncontains` : Doesn't contain
- `containss` : Contains case sensitive
- `ncontainss` : Doesn't contain case sensitive
- `null` : Is null/Is not null

相等匹配（默认）

GET /todos?name=hello

相等匹配

GET /todos?name_eq=hello

大于等于

GET /students?score_gte=60

id in (3, 6, 8)

GET /todos?id_in=3&id_in=6&id_in=8

name 包含 pizza 或者 name 包含 giovanni

GET /restaurants?name_contains=pizza&name_contains=giovanni

RESTFul API 参数 —— Sort、Limit、Start

按id正序排序（默认）

GET /todos?_sort=id:ASC

按id倒序排序

GET /todos?_sort=id:DESC

组合排序

GET /users?_sort=email:asc,dateField:desc

组合排序（大小写不敏感）

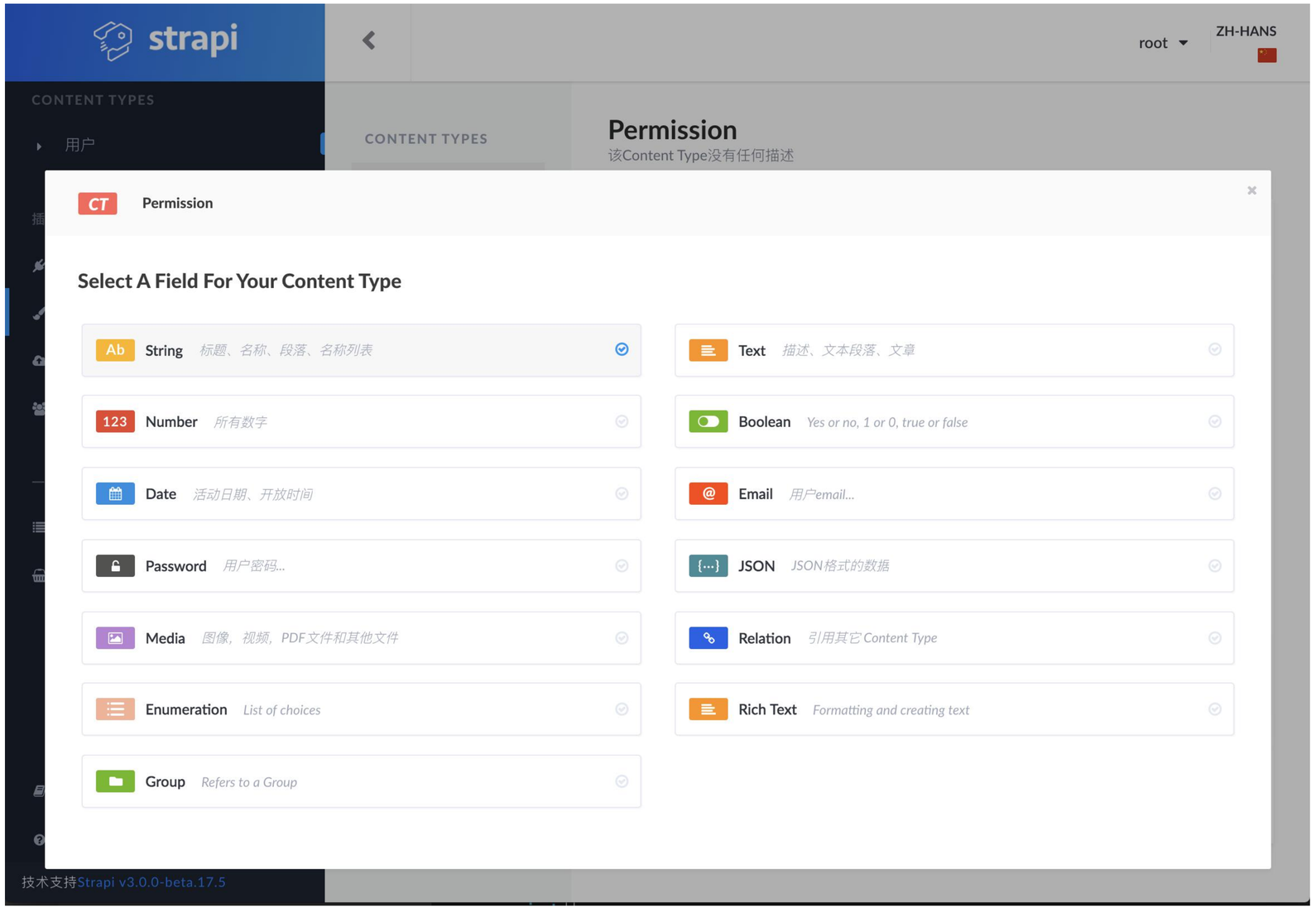
GET /users?_sort=email:DESC,username:ASC

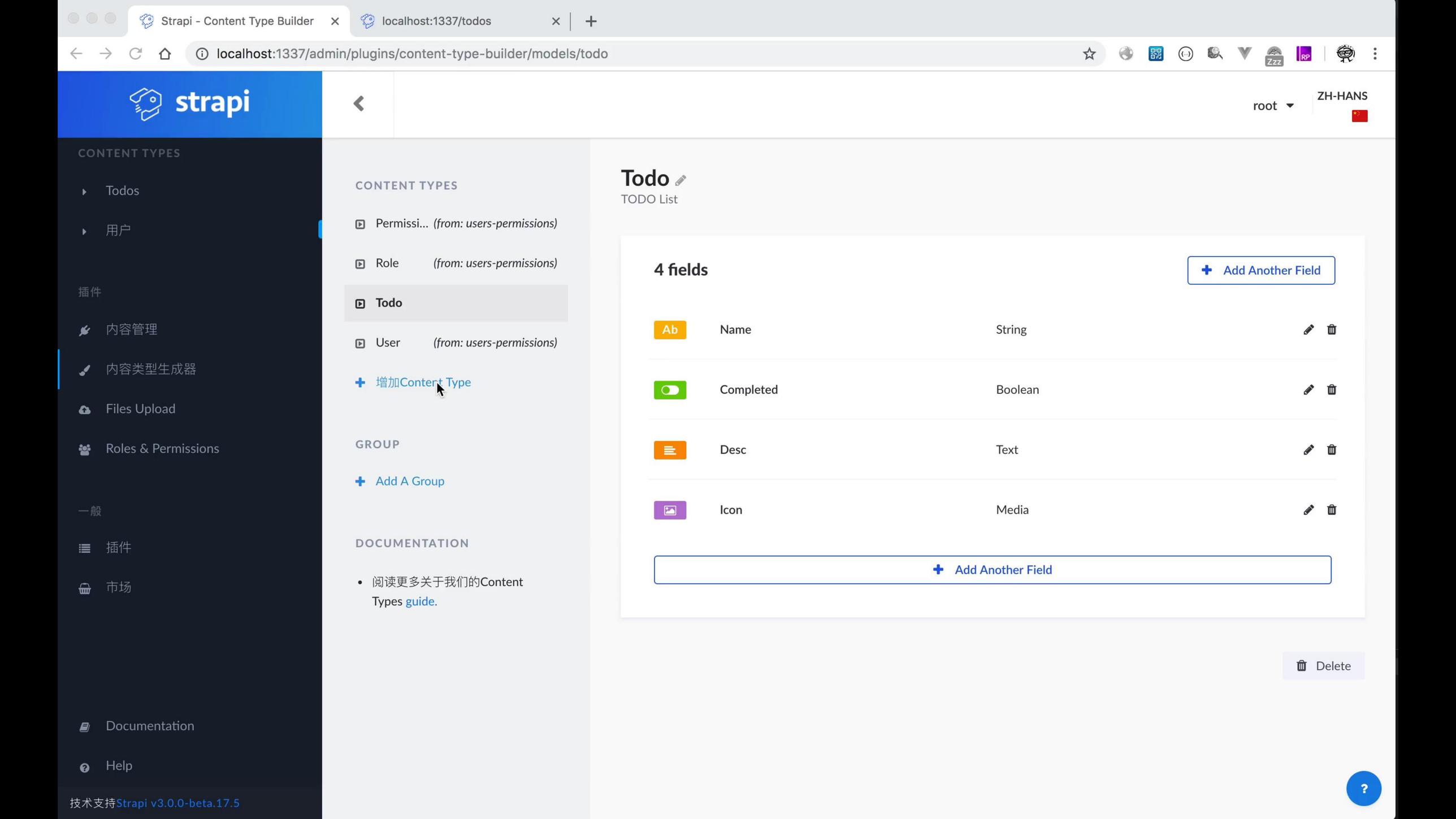
分页

GET /todos?_start=10&_limit=10

数据类型

- String
- Number
- Date
- Password
- Media
- Enumeration
- Group
- Text
- Boolean
- Email
- JSON
- **Relation**
- Rich Text(Markdown)




Strapi - Content Type Builder
localhost:1337/todos
localhost:1337/admin/plugins/content-type-builder/models/todo
strapi
root
ZH-HANS
CONTENT TYPES
Todos
用户
插件
内容管理
内容类型生成器
Files Upload
Roles & Permissions
一般
插件
市场
Documentation
Help
技术支持Strapi v3.0.0-beta.17.5
CONTENT TYPES
Permissi... (from: users-permissions)
Role (from: users-permissions)
Todo
User (from: users-permissions)
增加Content Type
GROUP
Add A Group
DOCUMENTATION
阅读更多关于我们的Content Types guide.
Todo
TODO List
4 fields
Add Another Field
Name
String
Completed
Boolean
Desc
Text
Icon
Media
Add Another Field
Delete

用户、分组与权限

用户注册

```js
import axios from 'axios';

// Request API.
// Add your own code here to customize or restrict how the public can register new users.
axios
  .post('http://localhost:1337/auth/local/register', {
    username: 'Strapi user',
    email: 'user@strapi.io',
    password: 'strapiPassword',
  })
  .then(response => {
    // Handle success.
    console.log('Well done!');
    console.log('User profile', response.data.user);
    console.log('User token', response.data.jwt);
  })
  .catch(error => {
    // Handle error.
    console.log('An error occurred:', error);
  });
```

用户登陆

```js
import axios from 'axios';

// Request API.
axios
  .post('http://localhost:1337/auth/local', {
    identifier: 'user@strapi.io',
    password: 'strapiPassword',
  })
  .then(response => {
    // Handle success.
    console.log('Well done!');
    console.log('User profile', response.data.user);
    console.log('User token', response.data.jwt);
  })
  .catch(error => {
    // Handle error.
    console.log('An error occurred:', error);
  });
```

用户请求（使用JWT）

```js
import axios from 'axios';

const token = 'YOUR_TOKEN_HERE';

// Request API.
axios
  .get('http://localhost:1337/posts', {
    headers: {
      Authorization: `Bearer ${token}`,
    },
  })
  .then(response => {
    // Handle success.
    console.log('Data: ', response.data);
  })
  .catch(error => {
    // Handle error.
    console.log('An error occurred:', error);
  });
```

案例分享：微信公众号关注登陆——注册

```
117      // 新用户注册
118      const {
119        user
120      } = strapi.plugins['users-permissions'].services;
121
122      // 默认角色
123      const role = await strapi
124        .query('role', 'users-permissions')
125        .findOne({
126          type: 'authenticated'
127        }, []);
128
129      // 用户参数
130      const params = {
131        role: role.id,
132        username: nickname || openid,
133        password: openid,
134        provider: 'local',
135        blocked: false,
136        email: openid + '@qiaoz.net',
137        confirmed: true,
138      };
139      params.password = await user.hashPassword(params);
140
141      // 创建
142      const u = await strapi
143        .query('user', 'users-permissions')
144        .create(params);
145
146      // 注册微信对象，关联user
147      userInfo.user = u.id;
148      wxUser = await strapi.query('weixin').create(userInfo);
149      wxUser.isNew = true;
150    }
```

引用user的内置service

获取指定用户分组

准备user对象数据

写入新创建的user对象数据

案例分享：微信公众号关注登陆——扫码

```
74
75    async checkQrCode(ticket) {
76      if (!ticket) {
77        return {};
78      }
79      const key = `STR:WX:QR_CODE:${ticket}`;
80      const uid = await strapi.redis.get(key);
81      if (uid) {
82        const {
83          jwt
84        } = strapi.plugins['users-permissions'].services;
85        const token = jwt.issue({
86          id: uid
87        });
88        await strapi.redis.expire(key, 60);
89        return {
90          jwt: token
91        }
92      } else {
93        return {}
94      }
95    },
96
```

引入jwt相关service

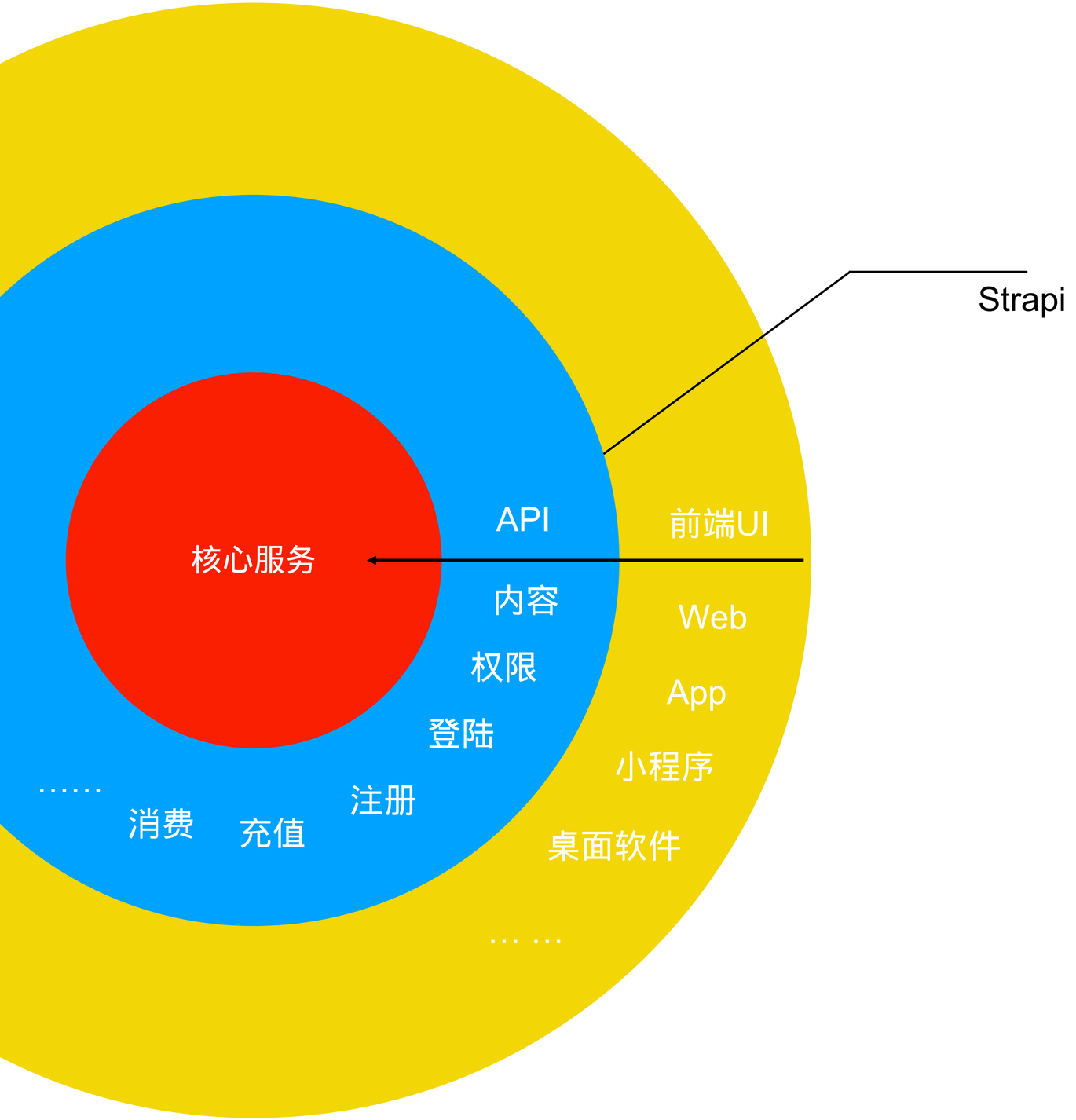
JWT存储的数据

安全策略

Strapi内置的安全策略

- CSP
- P3P
- HSTS
- X-Frame
- XSS
- CORS
- IP(黑白名单)

研发每款产品都需要经历的流程



- 数据库设计
- 编写Model层
- 封装Service
- 编写API
- 定义Router
- 访问权限
- 注册/登陆
- 充值/消费
- 前端联调
- 上线运营

一些坑

- 目前仍然是 v3.0-beta版
- 数据库使用MySQL时，开发环境变更，无法同步到生产环境（缺少migration）
- 单机部署重启问题（nginx反向代理+切换端口）


strapi