**北部湾城市群综合科技服务平台**

**资源池管理系统**

**开**

**发**

**文**

**档**

**海南傲为智慧产业有限公司**

**2020年11月11日**

**一、业务需求**

北部湾城市群科技服务资源既有数字化资源，又有实体服务资源，分散在北部湾不同城市、不同行业、不同系统平台中，建立典型科技服务资源池，不仅需要解决实体服务资源虚拟化的问题，还需解决跨平台、跨区域的资源组织问题。

《北部湾城市群综合科技服务平台研发与应用示范项目实施方案》提出的建立科技服务资源池的关键技术路线分为以下三步：

1. **资源建模：**从资源入手，通过知识抽取、元数据处理、自组织映射、知识表示等处理过程，研究科技服务资源属性和功能，分析其共性特征和个性化差异，解决不同服务资源之间的异构性问题，制定带语义的资源多层粒度统一描述和建模标准规范、服务资源的分类标准规范、资源集成标准规范等。
2. **资源集合：**收集、整理以高端装备制造、冶金石化、电子信息等产业为主的科技服务需求和科技服务资源，分析科技服务内在关系，解决服务资源跨平台、跨区域的问题，整合科技服务资源，实现科技服务资源虚拟化。
3. **资源服务：**研究领域智能科技服务开发技术，构建研究开发、检验检测认证、知识产权、科技金融和创业孵化等5个科技服务资源池，并围绕提高科技资源的利用效率、提升科技服务质量、满足不同企业个性化需求，开展科技资源服务技术研究。

针对以上业务需求，项目开发了资源池管理系统，旨在为科技服务资源池提供设计建模、资源运营、服务接口等配套设施与工具，实现科技服务资源数据的统一管理，解决多源、跨域科技服务资源池开发与利用问题。

**二、系统介绍**

资源池管理系统是北部湾城市群综合科技服务平台的重要组成部分，系统支持文本、富文本、Markdown、图片、文件、关联等多种基础类型的可视化建模能力。系统支持基于自定义角色的资源级权限管理，满足企业级资源管理需求，系统支持 Webhook 触发，可以方便地发布服务接口，与外部系统集成。

资源池管理系统需要处理资源建模、集合、服务等复杂场景应用，因此我们引入了一些抽象的、通用的概念来表述系统的能力。

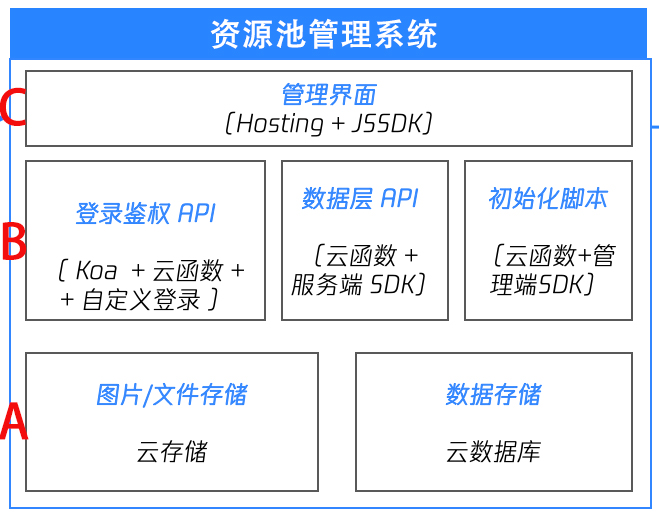
| **概念** | **定义** |
| --- | --- |
| ID | 通常表示一件事物的唯一标志，如公民的身份证号。 |
| 资源池 | 资源池管理系统以<资源池>为维度管理内容文档，不同资源池间的内容文档是彼此隔离的。 |
| 内容模型 | 对内容文档的结构定义。 |
| 内容集合 | 由一类内容文档组成的数据集，表现为表格的形式，表格中的每一列代表一个内容文档。 |
| 服务接口 | 服务接口是用户定义的接口回调，服务接口通常被资源池的某些事件激活，如内容修改。 |

**三、技术架构**

资源池管理系统是基于云平台基础能力，提供科技服务资源池的设计建模、运营管理、服务发布等功能，实现北部湾城市群综合科技服务平台Web门户、小程序、APP、服务端等多端统一管理。



资源池管理系统采用基于 Restful 标准的三层架构设计。



**A．**资源池管理系统底层由云存储和云数据库提供图片/文件存储、数据存储能力。云存储实现资源池管理系统上传的图片、文件的存储。云数据库系统级模型包括：

* rpms-projects 集合：资源池管理系统项目数据；
* rpms-schemas 集合：资源池管理系统内容配置数据，所有的系统内容类型配置、字段配置等信息都存储在该集合内；
* rpms-users 集合：资源池管理系统用户数据，存储系统的用户信息，包括管理员和运营者的账号信息，包括角色信息等；
* rpms-webhooks 集合： 资源池管理系统 webhook 集合，存储系统的回调接口配置，系统数据的变更可以通过回调来进行同步；
* rpms-user-roles 集合：资源池管理系统用户角色配置集合，存储系统的自定义用户角色信息；
* rpms-settings 集合：资源池管理系统配置集合，存储系统的设置；

**B．**资源池管理系统中间层由云函数分别提供登陆鉴权 API、数据层 API和初始化脚本，包括：

* rpms-service 该函数提供登录鉴权功能，用户在资源池系统管理界面通过用户名和密码来进行登录时，会通过 HTTP 来请求该函数；提供 API 接口功能，所有对内容的操作和管理都会经过此函数调用，内容操作会根据用户权限来进行数据库操作；
* rpms-init 提供初始化应用功能，安装系统后，会通过该函数来进行静态资源的部署和密码的生成和设置，修改账号密码或者部署路径等扩展参数都会再次执行该函数来进行更新；

**C．**资源池管理系统顶层是基于 React 开发的系统管理界面，应用云对象存储在 CDN 实现静态托管，通过 https 标准协议通信访问资源池管理系统的函数、数据库和存储等资源。

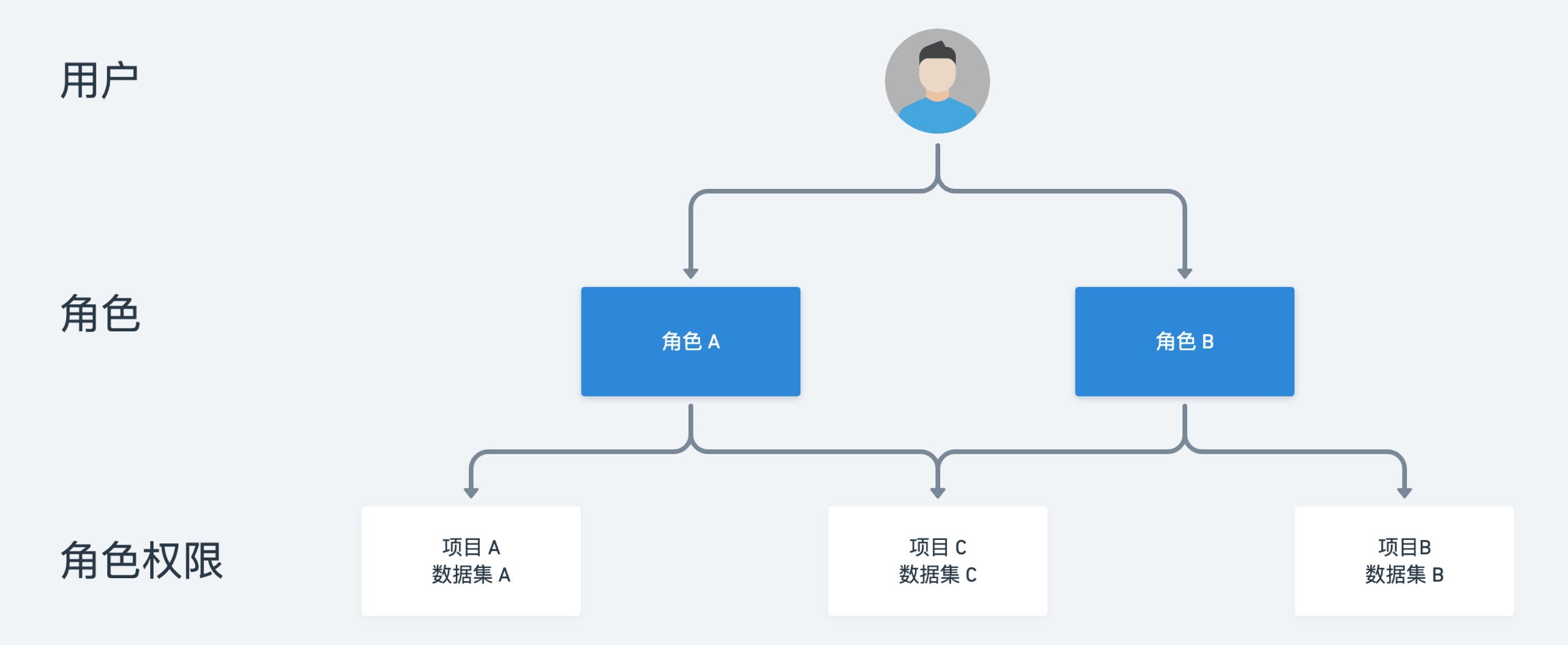
**四、功能特性**

| **特性** | **介绍** |
| --- | --- |
| 免开发 | 基于资源模型配置生成数据管理界面，无须编写代码 |
| 功能丰富 | 支持文本、图片、文件、枚举等多种类型的可视化建模，并且支持资源关联 |
| 权限控制 | 基于自定义角色的资源级权限管理，满足企业级需求 |
| 服务集成 | 支持 Webhook 触发，可以方便的与外部系统集成 |
| 数据源兼容 | 支持管理已有的资源池数据，也可以在资源池管理系统创建新的内容和数据集合 |
| 部署简单 | 支持容器化一键部署和升级，也可通过项目提供的脚本自动部署 |

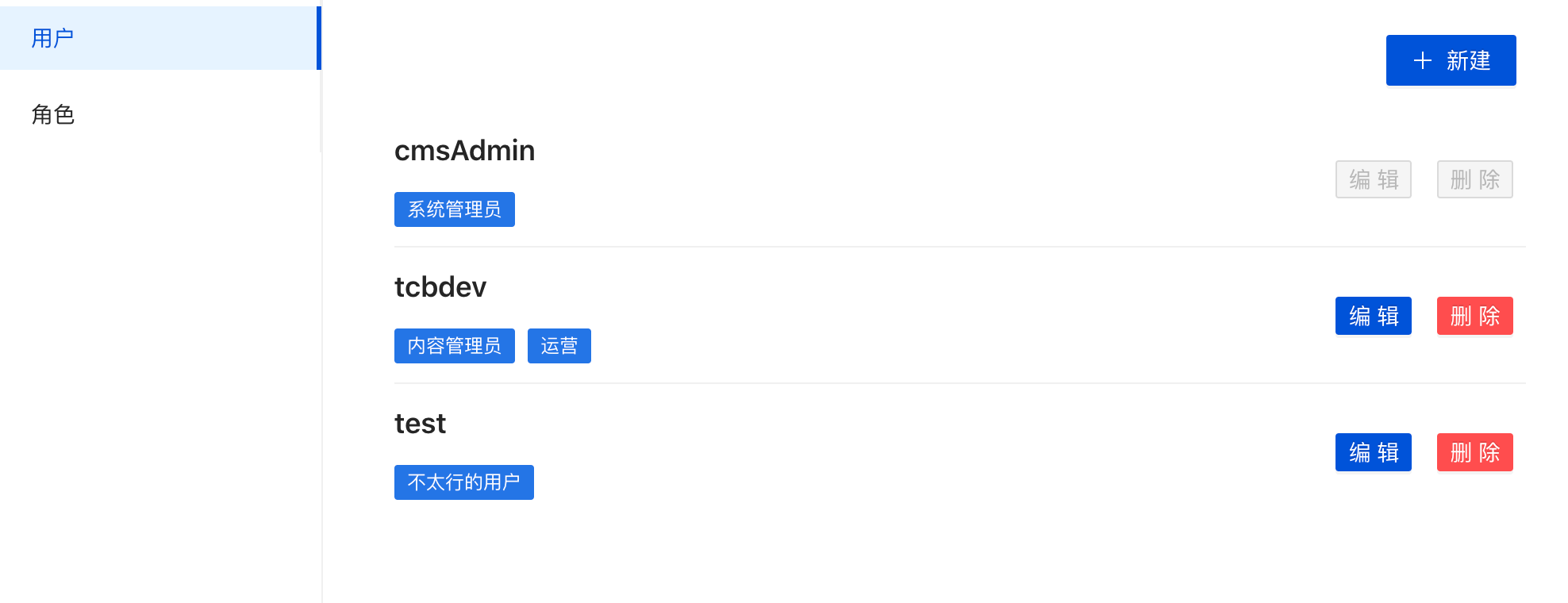
**五、核心功能**

**（一）权限控制**

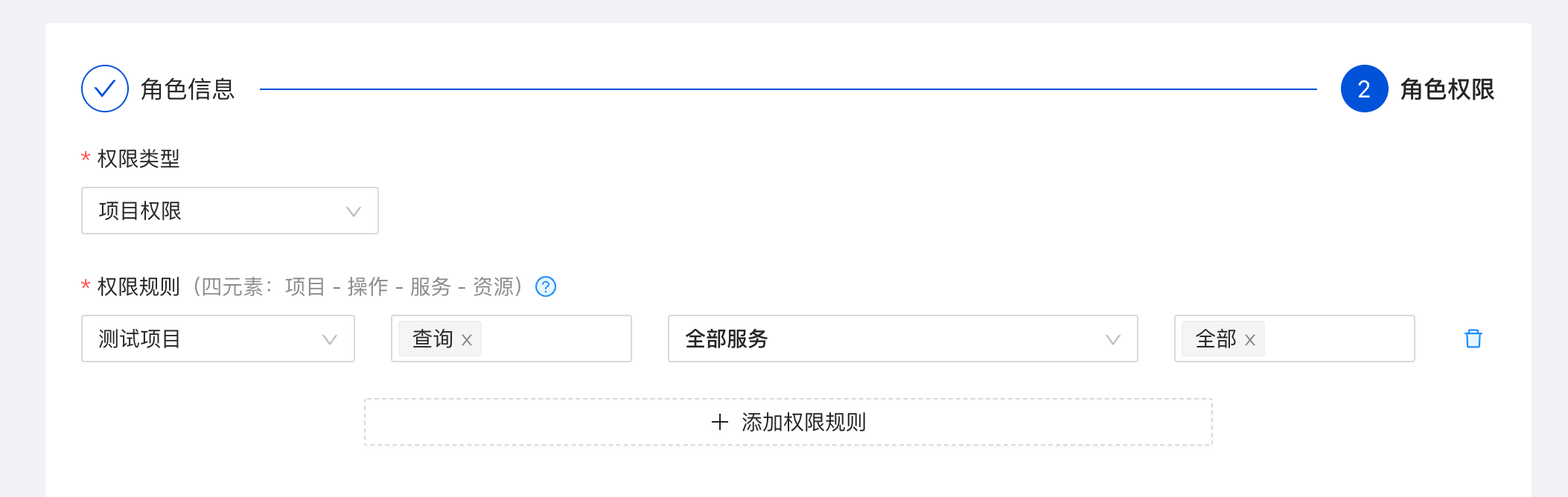
资源池管理系统的权限控制体系，引入了基于角色/资源的多维度权限校验体系，以控制用户的访问权限。



资源池管理系统支持创建自定义用户、自定义角色，实现自由的系统访问权限控制。自定义用户可以绑定任意角色，并获取角色的权限集，实现用户权限的授予。

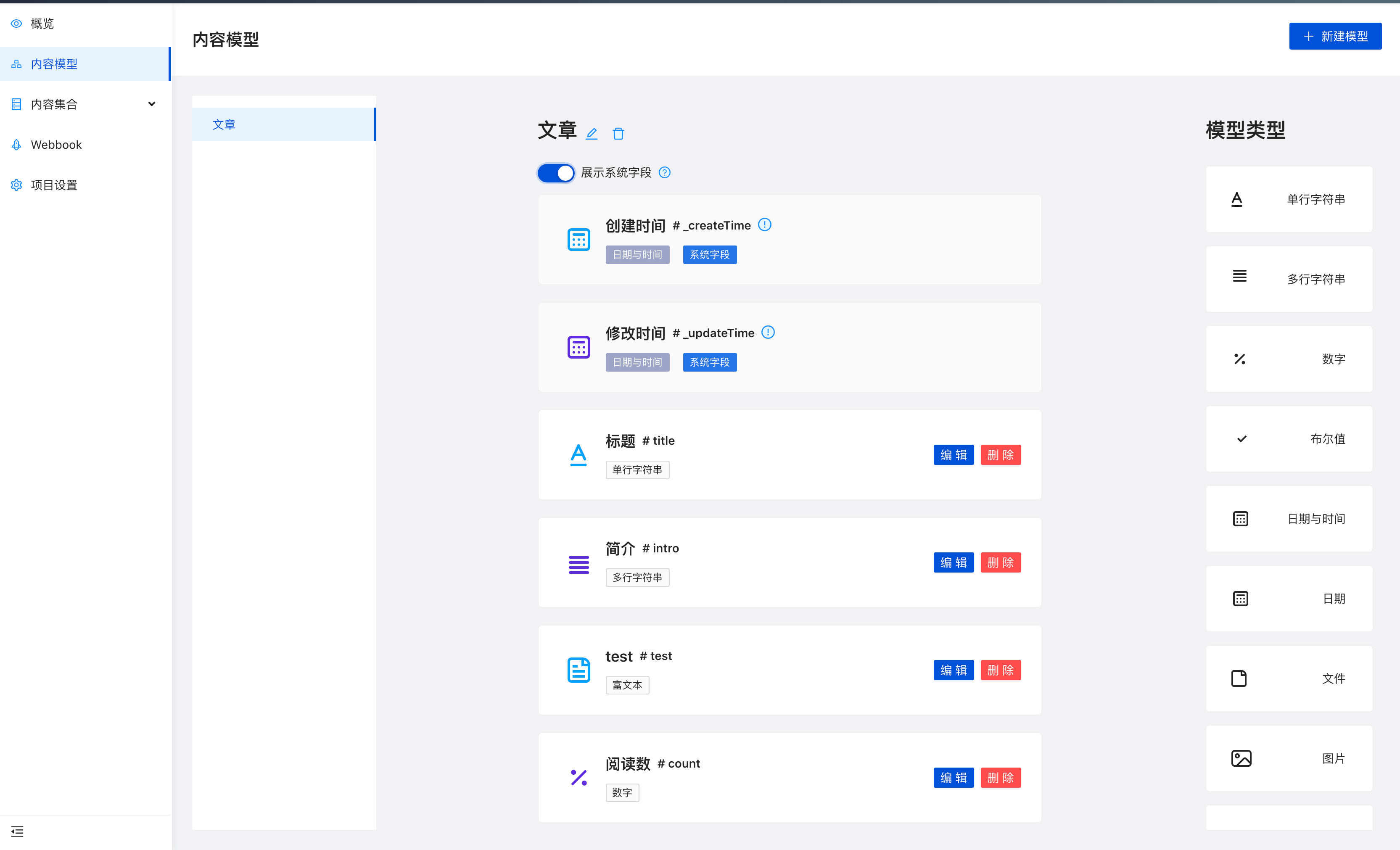


角色是资源池管理系统资源可访问权限的载体，可以自由组合多种不同的系统资源访问权限。资源池管理系统支持用户自定义系统中的角色，通过复合的权限规则，指定角色拥有的资源访问权限，如下所示



**（二）资源建模**

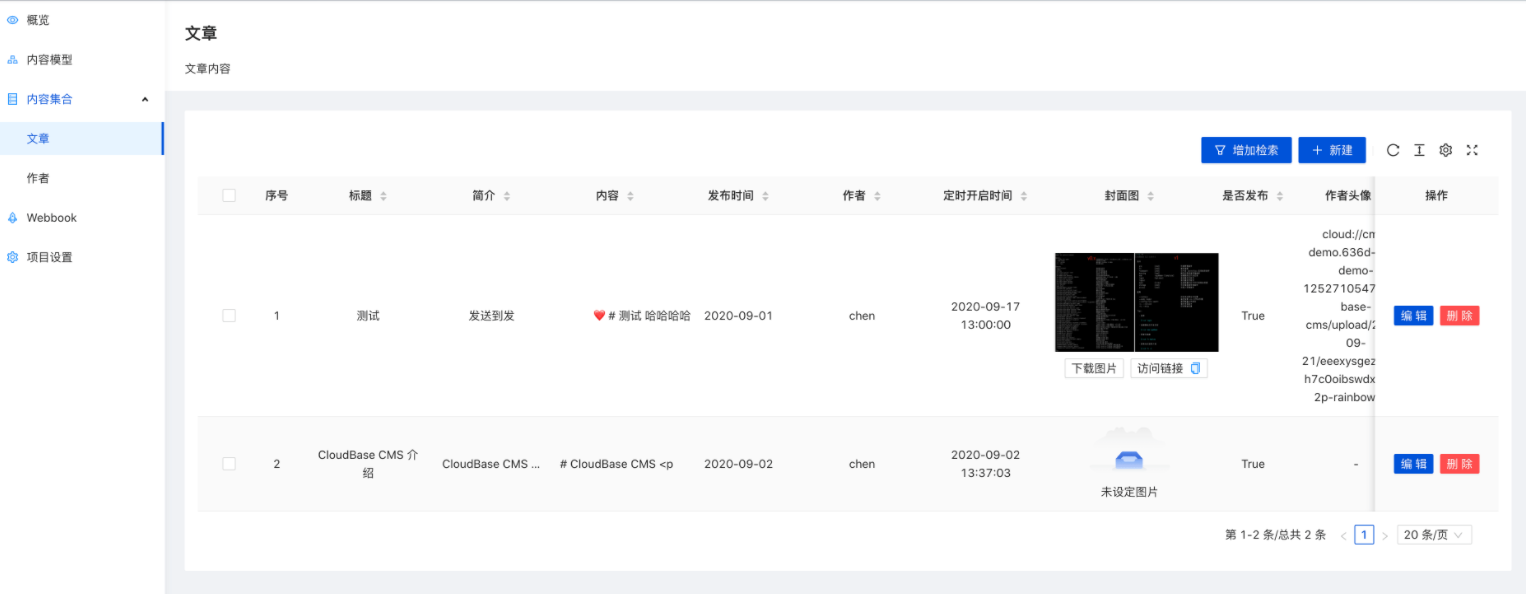
在资源池管理系统中，我们设计了资源模型的可视化编辑方式，实现了无代码建模，如下图所示：



在资源建模过程中，我们引入了系统字段的概念，即通过资源池管理系统创建模型时，系统会默认添加一些字段，如创建、修改时间，未来可能会有更多的系统字段出现。在资源池管理系统中，所有存储的时间类型都会被格式化为 Unix Timestamp 存储。

**（三）资源集合**

资源池管理系统采用最新的设计规范，提供了更加简洁、优化的 UI 交互界面，优化了内容数据的展示效果，根据字段的类型不同，都做了对应的展示优化，以提供更好的使用体验。同时，我们也对字段较多的集合进行了优化，支持表格的滚动查看，满足运营人员日常管理资源池数据集合的业务需求，如下所示：



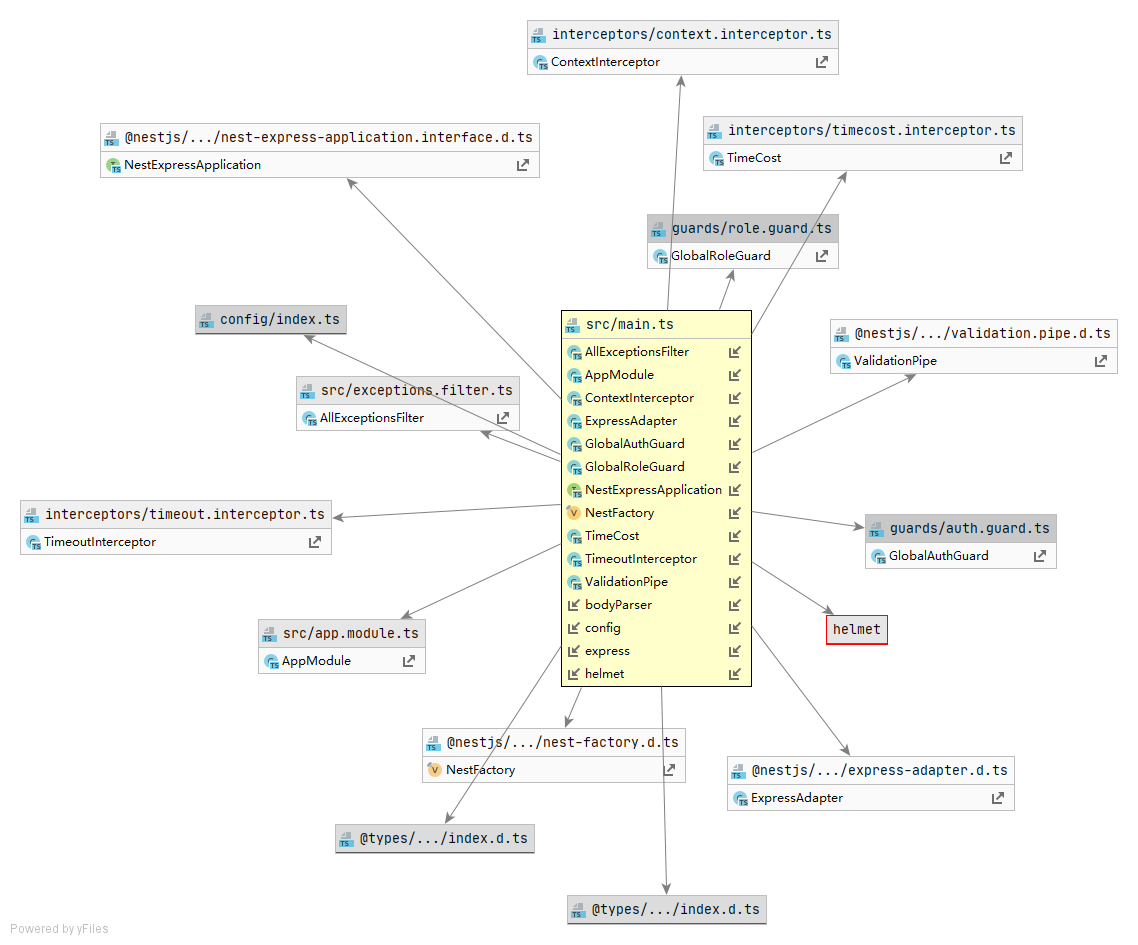
**六、详细设计**

**（一）主入口类**

资源池管理系统是在TypeScript和JavaScript (ES6、ES7、ES8)之上构建高效、可伸缩的企业级服务器端应用程序。系统基于良好的 Node 中间件生态，采用一个层与层直接的耦合度极小、抽象化极高的一个架构体系。

健壮的服务端程序须实现安全控制、鉴权授权、异常处理等关键功能，资源池管理系统采用渐进式的中间件架构方案，将主体业务逻辑与通用功能拆分，降低系统耦合度，主入口类注入中间件包括：

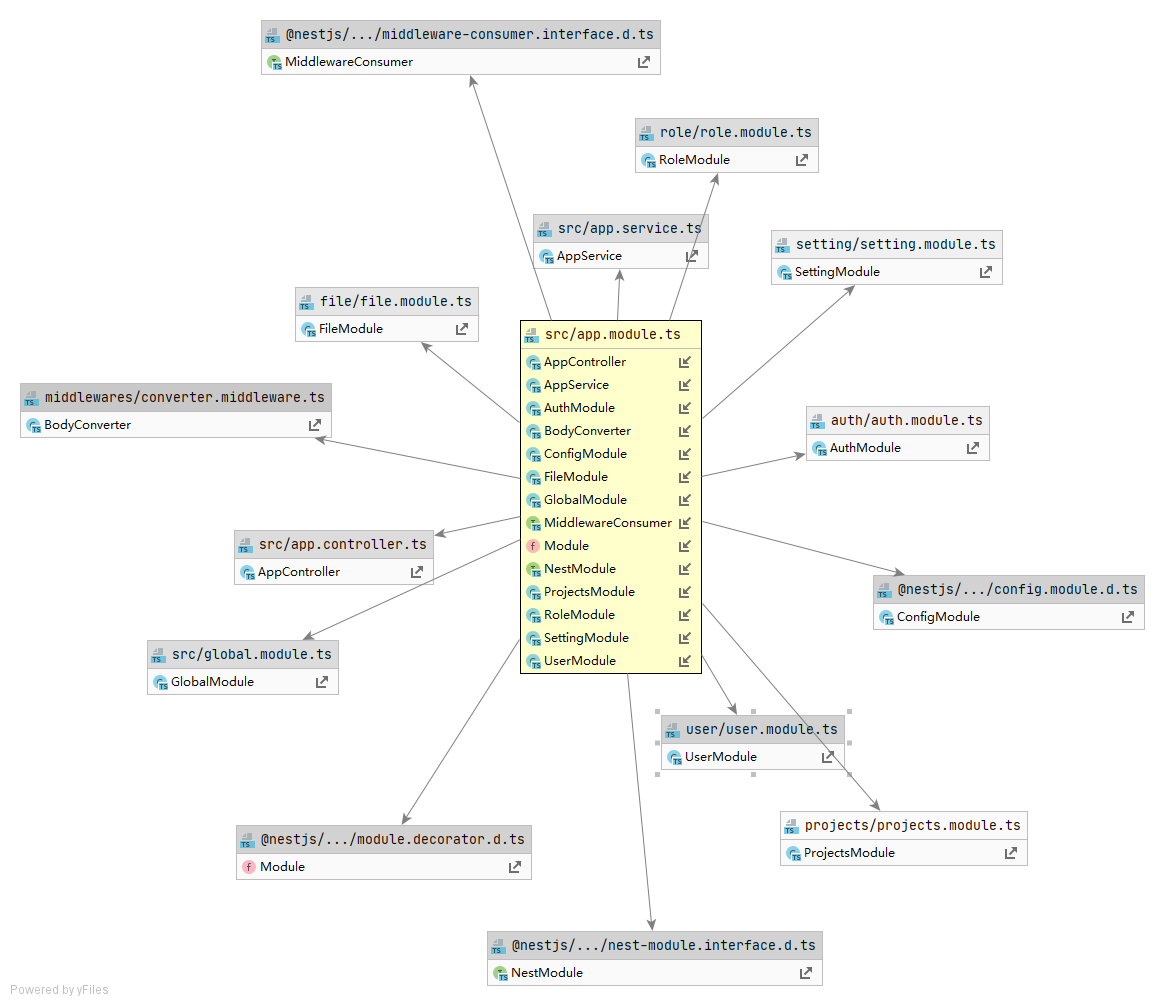
* Helmet：安全限制中间件
* ValidationPipe：参数校验中间件
* GlobalAuthGuard、GlobalRoleGuard：登陆校验、角色权限控制中间件
* bodyParser：请求body限制中间件
* TimeCost、TimeoutInterceptor：运行计时、超时处理中间件
* ContextInterceptor：上下文中间件
* AllExceptionsFilter：错误异常处理中间件



**（二）主模块类**

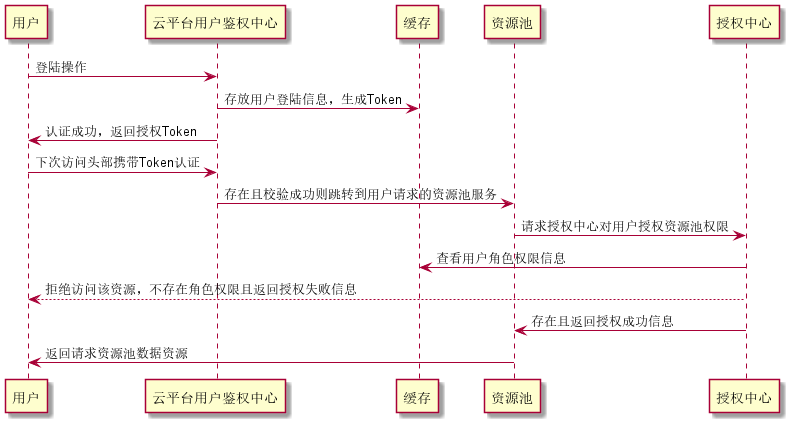
资源池管理系统通过模块化的方式构建业务应用，所有的业务模块都必须在主模块类 app.module.ts 中注册，系统的主体业务模块包括：

* Auth.module：登陆鉴权模块类
* Role.module：角色授权模块类
* User.module：用户管理模块类
* File.module：文件管理模块类
* Project.module：资源池设计建模、数据运营、服务发布模块类

****

**（三）基于用户角色的资源级权限控制**

资源池管理系统为满足不同级别、不同角色的管理人员对科技服务资源池数据运营需求，实现了基于用户角色的资源级权限控制功能。系统通过云平台用户管理功能提供统一的鉴权认证，系统授权中心满足自定义设置用户角色、资源池、操作、服务权限等功能。



**（四）资源池管理系统基础数据模型设计**

资源池管理系统提供科技服务资源可视化建模能力，支持基于自定义角色的资源级权限管理，支持发布 Webhook 服务接口等核心功能。根据业务需求，资源池管理系统的基础数据模型包括：

User：用户实体，保存用户基本信息、角色信息；

UserRole：角色实体，保存角色的权限信息；

AuthRequest：用户缓存实体，保存当前请求缓存的用户信息；

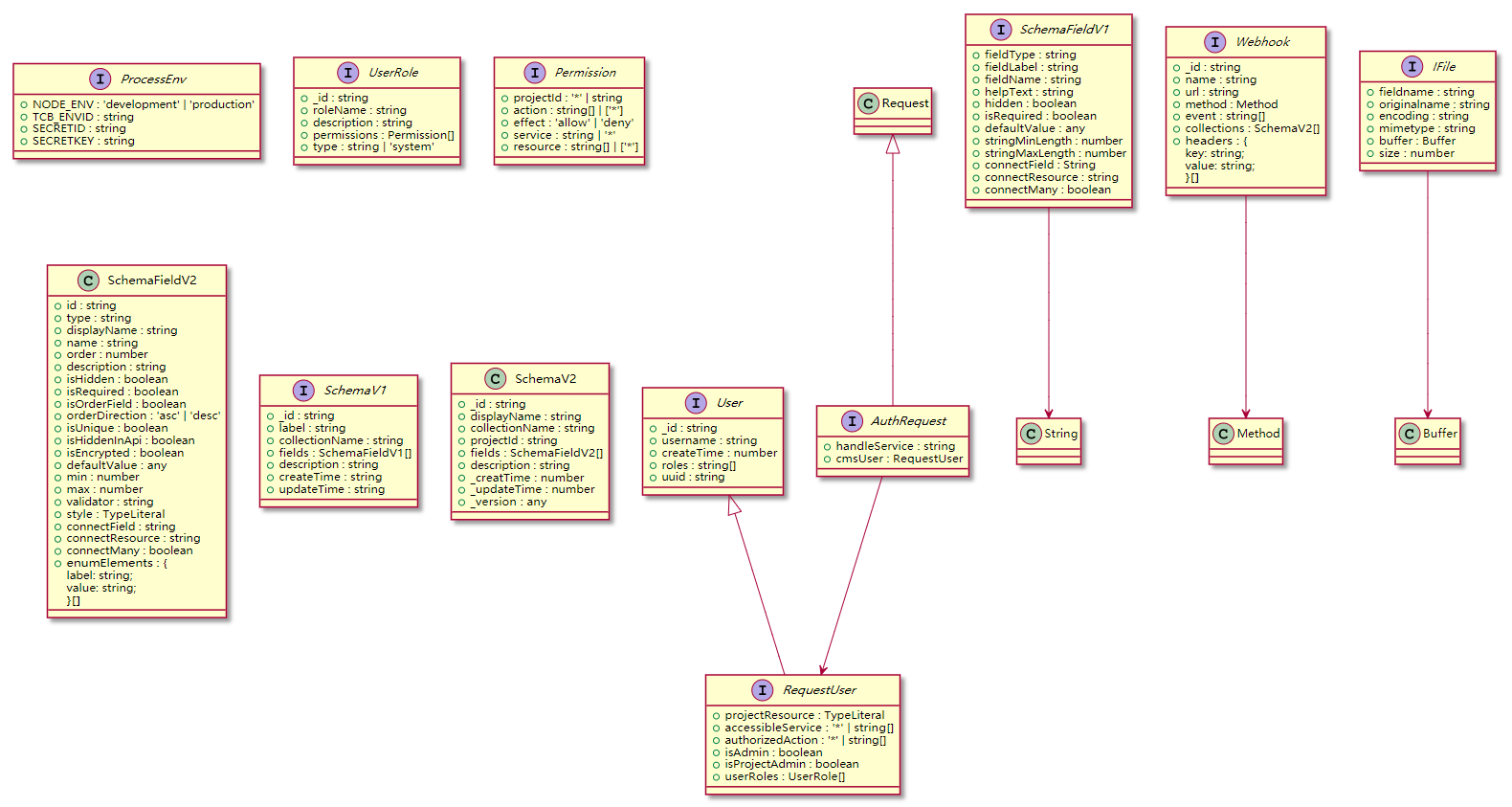
Permission：权限实体，保存资源池各类权限的操作信息；

SchemaFieldV1：数据模型实体 V1 版本，保存资源池可视化建模模型信息；

SchemaFieldV2：数据模型实体 V2 版本，保存资源池可视化建模模型信息；

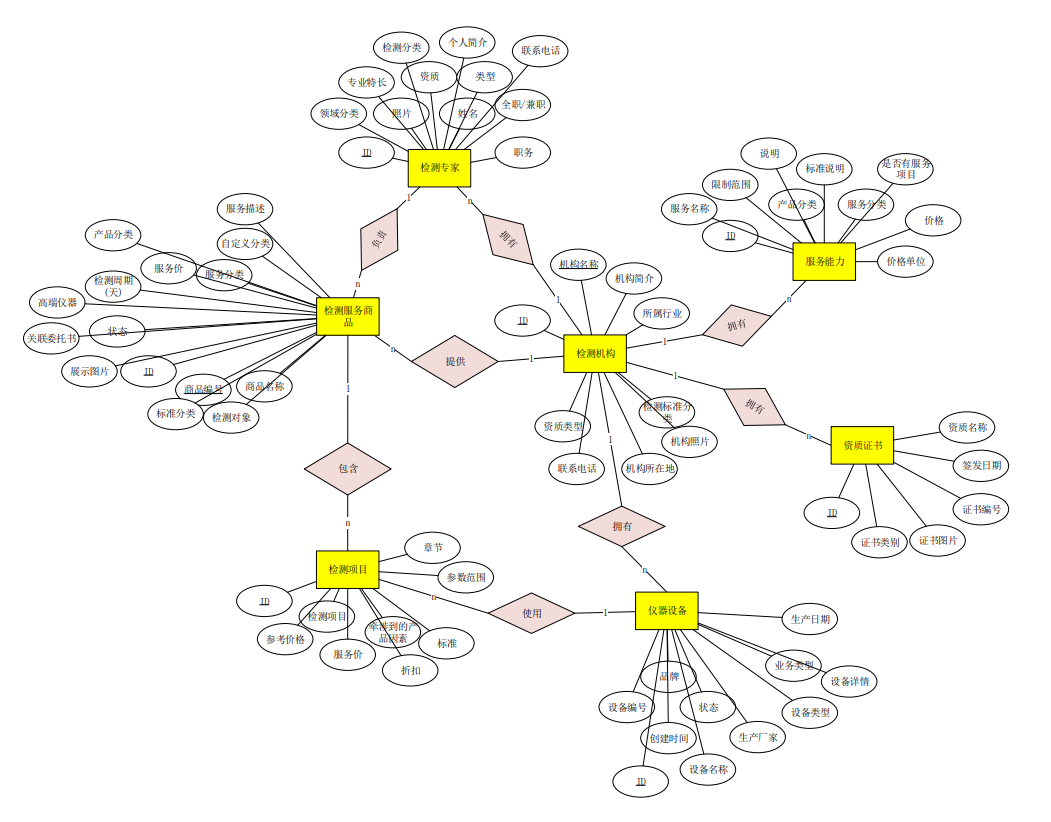
ProcessEnv：环境实体，保存当前系统版本号、云平台密钥等环境信息；

IFile：文件实体，保存文件信息；

****

**（五）资源池数据库设计**

资源池管理系统提供科技服务资源可视化建模工具，我们通过系统完成了研发服务、创业孵化、科技金融、知识产权、检验检测认证等资源池的设计建模，以检验检测资源池为例，数据模型包括检测专家、服务能力、检测机构、检测服务商品、检测项目、资质证书、仪器设备等实体。如下图所示：



**七、开发路线**

|  |  |  |
| --- | --- | --- |
| **功能** | **状态** | **发布版本** |
| 资源池管理、建模、数据运营、服务发布 | 已完成 | V1.0 |
| 模型导入、导出：支持模型数据的导入、导出 | 已完成 | V1.0 |
| 细粒度角色权限管理权限控制 | 已完成 | V1.0 |
| 支持枚举类型 | 已完成 | V1.0 |
| 登录使用云平台账号密码登录 | 已完成 | V1.0 |
| 富文本编辑器优化，支持上传图片 | 已完成 | V1.0 |
| 数据导入、导出：支持内容数据的导入、导出 | 进行中 |  |
| 支持复杂的对象，支持复杂数组类型，支持对象嵌套类型，JSON | 进行中 |  |
| 图片、文件数组支持 | 进行中 |  |
| 提供更简单的部署方法：CI 部署 | 规划中 |  |
| 定时任务 ⇒ 统计报表 | 规划中 |  |
| 提供资源池模板，支持从模板创建资源池 | 规划中 |  |

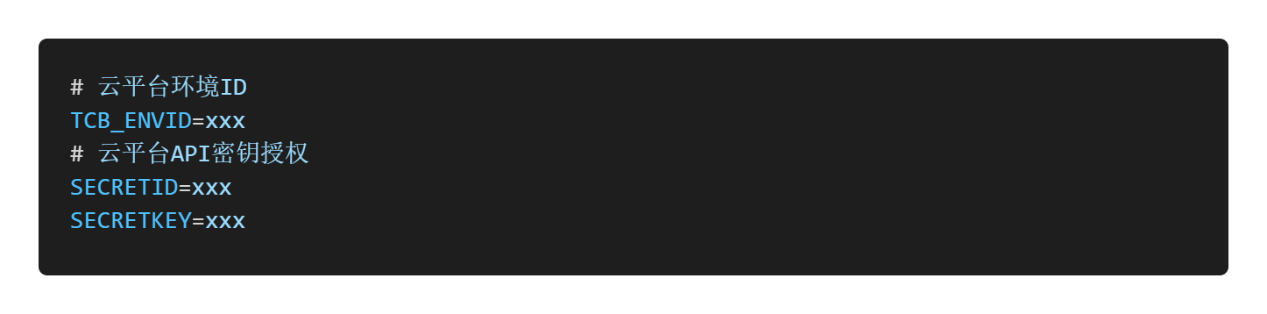
**八、开发指南**

**（一）基础环境**

安装 @cloudbase/cli，使用 Node 包管理器 NPM 安装最新版本的 CloudBase CLI 工具（1.0.0+）。

**（二）配置**

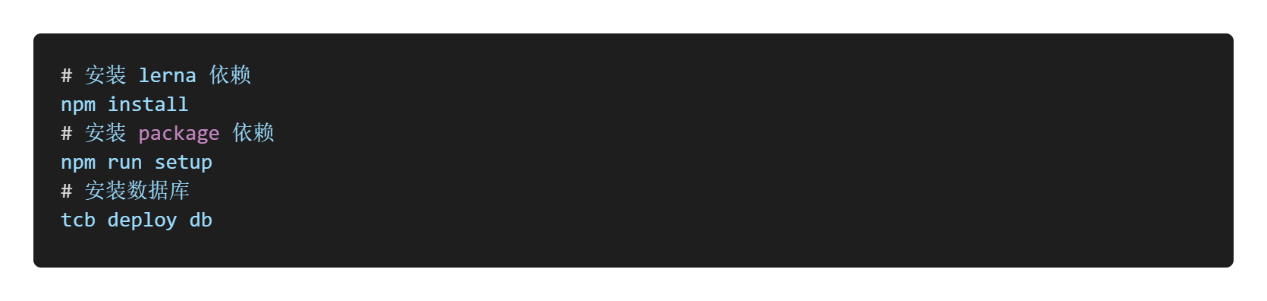
复制 packages/service/.env.example 为 .env.local，并配置。



复制 packages/admin/public/config.example.js 为 config.js，并配置。



**（三）安装依赖和数据库**

****

**（四）启动开发**

