**节能低碳技术文档**

**（第一版）**

**编写者：陈志昊**

1. **引言**

**为了本响应我国《区块链赋能“碳达峰碳中和”白皮书（2023版）》和《可信数据空间发展行动计划（2024—2028年）》的号召，本项目旨在为全社会提供全面碳中和碳普惠的解决方案，助力用户积极参与节能低碳环保实践，同时推动全社会向节能低碳可持续发展转型。**

1. **安装依赖：**

**meta-cloud: mvn install**

**meta-web: pnpm install**

**meta-miniprogram: npm install**

**其他py和npm均需云服务和训练集**

**因本项目全部代码调用本人云服务（非docker）和隐私外部依赖**

**存在运行失败的可能**

**三、代码清单：**

**1.管理后台和h5的前端代码**

**2.后台接口的源代码**

**3.小程序的源代码**

**4.业务智能合约代码（14个）**

**5.人工智能和自动化代码**

**6.联盟链管理后台官网技术文档**

**webase.front.version=v3.1.1**

**<https://webasedoc.readthedocs.io/zh-cn/latest/docs/WeBASE-Front>**

**7.IPFS密文的索引管理代码**

**8.wecross1.4和weidentity智能合约（12个）**

**9.收购方能源设备物联网管理系统**

**<https://energy.sunwoda.com/client-system/#/login>**

**GZLshuixuleng**

**Zhny@1744**

**<https://energy.sunwoda.com/energy-board-ui/#/login-gzl>**

**Gangzhonglv123**

**zhonglv@123**

1. **接口设计：**
2. 注册登录接口

接收用户注册信息（微信号ID），验证通过后生成登录令牌（Token），返回给前端，前端在后续请求中携带 Token 进行身份验证。

1. 节能低碳叶子接口

提供历史支付查询接口，根据用户 ID 和时间范围查询并返回相应的数据库数据。

1. 联盟链各功能接口

提供创建联盟链用户账号密钥对，智能合约调用等数据返回或错误提示。

4.兑换商城接口

记录购买信息，更新用户的剩余金豆值，并返回信息或错误提示。

5.商品查询接口

根据用户的查询条件（如商品类别、关键词、价格范围等）从数据库中查询商品信息，返回商品列表，包括商品名称、图片、价格、简介等。

6.购物车接口

允许用户添加商品到购物车、删除商品等操作，接口更新购物车数据并返回操作结果。

7.订单接口

接收用户提交的订单信息（购物车商品列表、收货地址、支付方式等），进行订单处理，包括生成订单号、扣减库存、记录订单状态等，返回订单处理结果和支付链接。

1. **测试优化与运维：**
2. **前端性能优化**

对图片进行压缩处理，减少图片文件大小，提高页面加载速度。采用代码分包技术，将小程序代码按照功能模块拆分成多个子包，用户在首次打开小程序时只加载核心功能代码包，其他功能包在用户使用到相应功能时再进行加载，减少初始加载时间。合理使用缓存机制，将常用的数据（如 节能低碳指南信息、用户基本信息等）存储在本地缓存中，减少数据请求次数，提高页面响应速度。

1. **后端性能优化**

对数据库查询语句进行优化，建立合适的索引，提高数据查询效率。 采用缓存技术，如 Redis 缓存，将频繁访问的数据（如碳排放因子数据、热门商品信息等）缓存到内存中，减少数据库查询压力。 对服务器进行性能监控和负载均衡，根据业务量动态调整服务器资源配置，确保小程序在高并发情况下的稳定运行。

安全措施

3.**用户数据安全**

对用户密码进行加密存储，采用安全的哈希算法（如 bcrypt）将用户密码加密后存储到数据库中，防止密码泄露。

在数据传输过程中，采用 HTTPS 协议对数据进行加密传输，防止数据被窃取或篡改。

定期对数据库进行备份，将备份数据存储在安全的位置，防止数据丢失。

4.**系统安全**

对小程序进行代码安全审计，检查代码中是否存在安全漏洞（如 SQL 注入漏洞、XSS 跨站脚本漏洞等），及时修复漏洞。 对用户输入数据进行严格的合法性验证和过滤，防止恶意数据输入导致系统故障或安全问题。 - 限制用户操作权限，根据用户角色（普通用户、管理员等）分配不同的操作权限，防止用户越权操作。

测试与维护

5.**测试计划**

功能测试

对小程序的各个功能模块进行详细测试，包括碳足迹计算的准确性、 节能低碳生活指南的展示完整性、打卡功能的正常运作、社区互动功能的流畅性、商城购物流程的正确性等，确保每个功能都能按照设计要求正常运行。 - 采用黑盒测试方法，设计各种测试用例，覆盖不同的用户操作场景和数据输入情况，检查功能输出结果是否符合预期。

性能测试

对小程序的性能进行压力测试，模拟大量用户并发访问小程序的情况，测试页面加载时间、接口响应时间、服务器资源利用率等性能指标，确保小程序在高并发情况下能够稳定运行，满足用户的使用需求。采用性能测试工具（如 Apache JMeter）进行性能测试，根据测试结果对性能瓶颈进行分析和优化。利用安全测试，进行安全漏洞扫描，检查小程序是否存在安全漏洞，如 SQL 注入漏洞、XSS 跨站脚本漏洞、文件上传漏洞等。

对用户认证和授权机制进行测试，确保只有合法用户能够访问相应的功能和数据，防止非法用户入侵和数据泄露。

6.**维护计划**

定期更新

根据用户反馈和业务发展需求，定期对小程序进行功能更新和优化，如添加新的节能低碳行动项目、更新节能低碳生活指南内容、改进碳足迹计算模型等。

定期更新数据库中的数据，如商品信息等，确保数据的准确性和时效性。

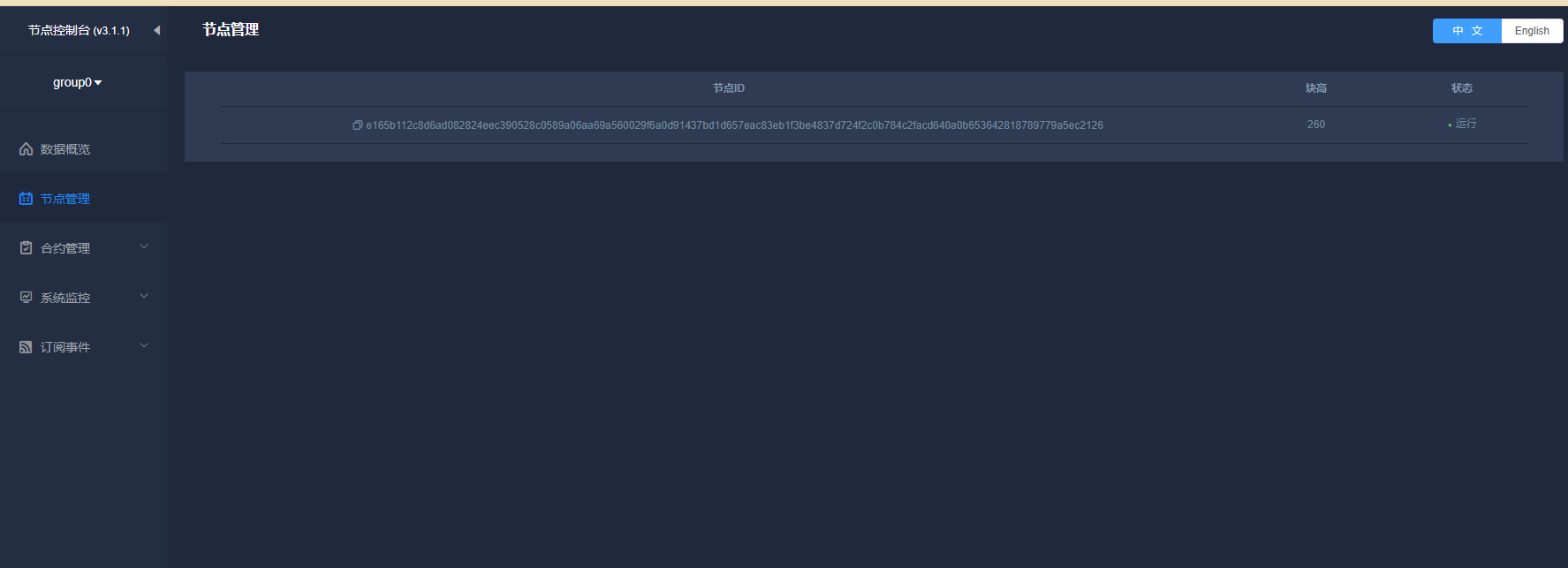
故障处理

建立完善的故障监控和报警机制，实时监控小程序的运行状态，能够及时发现并发出报警信息。制定故障处理流程，运维人员能够迅速响应，进行故障排查和修复，确保小程序尽快恢复正常运行。

7.**兼容性维护**

跟随微信小程序平台的不断更新和手机操作系统的升级，定期对小程序进行兼容性测试，确保小程序在不同版本的微信客户端和手机操作系统上都能正常运行，及时修复因兼容性问题导致的故障。

1. **运行结果展示：**

****

