

Genex 券交易平台 - 技术架构开发需求文档

本文档基于《Genex 券交易平台 - 软件需求规格说明书 v4.1》提炼的技术架构设计与开发指导

1. 文档概述

1.1 目的

为开发团队提供系统实现的技术规范和指导，确保技术架构与SRS v4.1保持完全一致。

1.2 权威文档

本文档的所有设计决策均以《券金融平台-软件需求规格 v4.1》为权威来源。如有冲突，以SRS为准。

1.3 核心设计原则

- 区块链是基础设施，不是用户界面**：用户看到支付宝/Zelle，底层跑区块链
- 合约清算保护企业客户**：消费环节平台不介入，客户数据不经过平台
- 自建链 = 完全掌控**：Gas、性能、合规、升级全部自主可控
- Utility Track优先**：MVP阶段只开放消费型券，规避SEC证券合规风险

2. 系统架构

2.1 整体架构模式

采用 **三层五域** 架构设计：

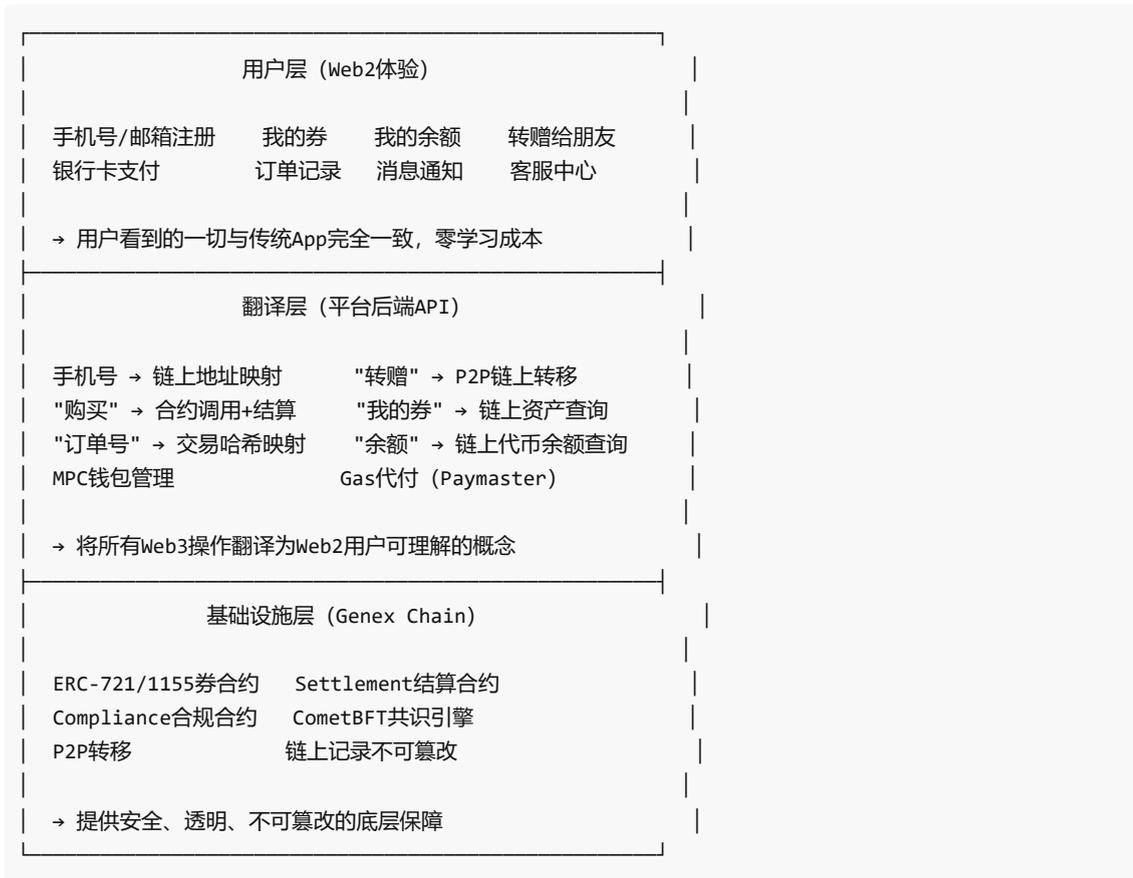
三层架构

层级	名称	职责
L1	应用层 (Application Layer)	Web2用户交互、多端适配、UX翻译层
L2	服务层 (Service Layer)	业务逻辑、微服务编排、API网关、AI/ML引擎
L3	基础设施层 (Infrastructure Layer)	Genex Chain、数据库、消息队列、缓存

五大业务域

域	名称	核心功能
D1	发行域 (Issuer Domain)	发行方入驻、券种管理、发行审批、可配置规则
D2	交易域 (Trading Domain)	挂单撮合、价格发现、订单管理、做市商
D3	清算域 (Clearing Domain)	链上原子交换结算、兑付清算、退款处理
D4	合规域 (Compliance Domain)	KYC/AML、OFAC筛查、Travel Rule、券类型防火墙
D5	AI与数据域 (AI & Data Domain)	信用评估、动态定价、异常检测、风控预警

2.2 UX架构 (核心创新)



翻译层核心功能

功能	Web2用户操作	翻译层处理	Web3实际执行
注册	手机号/邮箱注册	创建MPC托管钱包, 绑定映射	链上地址生成
购买券	点击"立即购买", 银行卡支付	法币→稳定币转换, 调用合约	Settlement合约原子交换
转赠	输入朋友手机号, 点"转赠"	手机号→地址解析, Gas代付	链上P2P转移 (含合规路由)
消费	出示券码/扫码核销	调用Redemption合约	合约清算+券销毁
查看余额	打开"我的余额"	聚合链上+链下余额	查询链上代币余额
查看记录	打开"交易记录"	交易哈希→订单号映射	读取链上事件日志
发行券 (B端)	填写券模板, 点"发布"	调用CouponFactory合约铸造	链上ERC-721/1155铸造
核销 (B端)	扫码/输入券码, 点"核销"	调用Redemption合约	合约清算+券销毁

术语映射表 (全平台执行)

所有面向用户的界面 (App、网站、邮件、客服话术) 统一使用左列术语, 禁止使用右列术语

用户界面术语	底层技术术语 (仅内部/Pro模式可见)
我的账户	链上钱包地址
我的券	ERC-721/1155 NFT资产
我的余额	链上稳定币 (USDC) 余额
转赠给朋友	P2P链上转移 (Transfer)
购买	链上原子交换 (Atomic Swap)
核销/使用	合约兑付 (Redemption)
订单号	交易哈希 (TX Hash)
平台积分	GNX代币
安全验证	链上签名 (由MPC钱包后台执行)

3. Genex Chain 技术架构

3.1 链架构总览

平台自建一条EVM兼容的应用链，承载所有券业务与合规逻辑。

组件	技术选型	说明
链框架	Cosmos SDK (最新版本)	200+生产链验证，模块化、可定制
共识引擎	CometBFT (原Tendermint)	拜占庭容错共识， 即时终结性 (1区块=最终确认)
EVM模块	cosmos/evm (官方Cosmos EVM)	Apache 2.0，替代已弃用的Ethereum
跨链通信	IBC (Inter-Blockchain Communication)	Cosmos生态原生跨链协议
跨链桥	Axelar (主) / Wormhole (备)	稳定币从Ethereum桥入Genex Chain
撮合架构	链下内存订单簿 + 链上结算	参考dYdX v4

3.2 链设计参数

参数	目标
共识机制	CometBFT PoS (初期平台运营验证节点，后期开放质押)
EVM兼容	完全兼容EVM (Solidity、Hardhat、MetaMask全套工具链)
出块时间	≤ 1秒 (CometBFT即时终结性)
TPS	≥ 5,000 (Block-STM并行执行)
Gas策略	平台前期全额补贴，用户零Gas体验
原生代币	GNX (用于Gas、治理；前期Gas补贴，用户不接触)

节点运营	平台自营验证节点 + 未来开放合格机构节点
跨链能力	IBC连接Cosmos生态 + Axelar桥连接Ethereum生态
区块浏览器	自建 (EVM兼容, 支持合约验证和交易查询)
监管节点	为监管机构提供只读全节点/专属API

3.3 参考链与架构借鉴

参考链	价值	关键参考点
Cronos (Crypto.com)	Cosmos SDK + EVM最成熟实现	<1s出块、30K TPS、Block-STM并行执行
dYdX v4	交易平台专用链标杆	链下订单簿+链上结算、自定义Cosmos模块
Injective	金融应用Layer-1	链上CLOB、RWA支持、机构级合规框架

注意：不使用Ethernint (已停止维护) 和Kava (已转向DeAI方向)。

3.4 为什么自建链

维度	公链/L2	自建链 (Genex Chain)
Gas控制	受市场波动	平台完全控制, 可设为零
合规执行	链层面无法强制	链级OFAC/Travel Rule内置
性能调优	共享资源	独享资源, 针对券业务优化
升级自主	依赖链治理	平台自主决定升级
数据主权	完全公开	可控可见性
监管对接	难以定制	专属节点/API

3.5 Gas费策略

前期策略 (平台全额补贴) :

操作	Gas承担方	说明
券发行 (铸造)	平台补贴	降低发行方入驻门槛
一级/二级市场交易	平台补贴	交易零额外摩擦
P2P转移	平台补贴	鼓励券流通
券兑付 (消费)	平台补贴	消费环节零成本

自建链Gas = 平台运营成本 (服务器/节点), 非公链ETH支付。

技术实现:

- 创世配置Gas Price = 0 (或极低值)
- 可选: ERC-4337 Paymaster作为Gas代付标准化接口

- Gas参数通过Governance合约动态调整（无需硬分叉）

3.6 链级合规能力（自建链独有优势）

- 链级OFAC过滤：验证节点拒绝处理制裁地址的交易
- 链级Travel Rule：大额转移必须携带身份哈希，否则节点拒绝打包
- 链级交易监控：节点内置异常交易检测模块
- 监管API：监管机构通过专属API实时查询链上数据
- 紧急冻结：治理合约多签冻结涉案地址资产

4. 智能合约架构

4.1 合约体系（7合约系统）

合约	功能	核心逻辑
CouponFactory	券发行工厂	铸造新券，标记Utility/Security类型，写入发行方配置规则
Coupon	数字券合约（ERC-721/1155）	管理所有权与转移，执行转售次数限制、不可转让等规则
Settlement	交易结算合约	链下撮合后链上原子交换（券↔稳定币），Utility Track价格上限=面值
Redemption	兑付合约	消费者直接与发行方结算，验证并销毁已使用券（平台不介入）
Treasury	资金托管合约	管理交易资金流，退款原子反向交换
Compliance	合规合约	OFAC地址过滤、Travel Rule数据哈希记录、紧急冻结
Governance	治理合约	Gas参数调整、紧急冻结多签授权、合约升级管理

撮合引擎不上链，采用链下内存订单簿（参考dYdX v4），成交后调用Settlement合约链上结算。

4.2 券类型防火墙（合约级实现）

规则	消费型券（Utility Track）	投资型券（Securities Track）
二级市场价格	Settlement合约强制 ≤ 面值	无限制
转售次数	链上计数器，默认2-3次	无限制
有效期	强制 ≤ 12个月	可更长
券类型标记	CouponFactory铸造时写入，不可变	同上

MVP阶段只开放Utility Track。Securities Track待法律意见书和牌照后再开放。

4.3 合约代码架构

```
// CouponFactory - 券发行工厂合约
contract CouponFactory {
```

```

enum CouponType { Utility, Security }

struct CouponConfig {
    CouponType couponType;    // 券类型 (不可更改)
    bool transferable;        // 是否可转让
    uint8 maxResaleCount;     // 最大转售次数
    uint256 maxPrice;         // 价格上限 (Utility = 面值)
    uint256 expiryDate;      // 到期日
    uint256 minPurchase;     // 最低消费金额
    bool stackable;          // 是否可叠加使用
    bytes32[] allowedStores;  // 限定门店列表
}

function mintBatch(
    address issuer,
    uint256 faceValue,
    uint256 quantity,
    CouponConfig calldata config
) external returns (uint256[] memory tokenIds);
}

// Settlement - 交易结算合约
contract Settlement {
    // 原子交换: 券 ↔ 稳定币 同时转移
    function executeSwap(
        uint256 tokenId,
        address buyer,
        address seller,
        uint256 price
    ) external {
        // Utility Track: require(price <= coupon.faceValue)
        // 原子性: 要么全部成功, 要么全部回滚
    }

    // 退款: 反向原子交换
    function executeRefund(
        uint256 tokenId,
        address buyer,
        address seller,
        uint256 refundAmount
    ) external;
}

// Compliance - 合规合约
contract Compliance {
    // OFAC筛查
    function isBlacklisted(address account) external view returns (bool);

    // Travel Rule (≥$3,000强制)
    function recordTravelRule(
        address sender,
        address receiver,

```

```

        bytes32 senderInfoHash,
        bytes32 receiverInfoHash
    ) external;

    // 紧急冻结 (需多签)
    function freezeAccount(address account) external onlyGovernance;
}

// Redemption - 兑付合约 (消费者直接与发行方结算)
contract Redemption {
    function redeem(
        uint256 tokenId,
        bytes32 storeId // 门店ID (验证限定门店规则)
    ) external {
        // 验证券有效性、门店限定、叠加规则
        // 销毁券 (burn)
        // 通知发行方 (event)
        // 平台不介入消费数据
    }
}

```

4.4 合约升级策略

模式: **Transparent Proxy (透明代理)**

用户/前端 → Proxy合约 (地址不变, 存储不变) → Implementation合约 (可替换)

升级治理流程:

发起升级提案 → 多签审批 (3/5多签) → 时间锁等待 (48小时) → 自动执行
 ↓
 等待期内可紧急取消 (安全委员会)

- 核心合约均采用Transparent Proxy部署
- 升级需Governance合约3/5多签 (核心成员+独立安全审计师)
- 时间锁48小时 (紧急通道缩短至4小时, 需4/5多签)
- 升级前必须通过第三方安全审计
- 升级日志链上记录 (SOX审计追踪)
- 保留前一版合约地址, 支持回滚

不可升级的合约逻辑 (安全红线):

- 券类型标记逻辑 (Utility/Security) ——类型不可修改
- 所有权记录——不可被升级操作篡改
- 链上转售计数器——防止绕过Utility Track限制

4.5 数字券标准

- 基于ERC-721 (单张券唯一) / ERC-1155 (批量同类券高效)
- 券元数据: 面值、发行方、有效期、使用条件、券类型、配置规则
- 券状态管理: 可流通、已兑付、已过期

- 批量发行优化 (gas效率)

5. 技术栈

5.1 前端技术

端	技术选型	说明
Web端 (消费者)	React + Next.js	SSR支持、SEO优化
移动端 (消费者)	Flutter 3.x	跨平台、一套代码多端
管理后台 (发行方B端)	Ant Design Pro	企业级中后台框架
商户核销App	Flutter (共享组件)	门店收银员专用, 扫码即核销

5.2 后端技术

层级	技术选型	说明
API网关	Kong / APISIX	限流、认证、路由
业务服务	Go 微服务架构	高性能、强类型
翻译层服务	Go + Redis	地址映射、Gas代付、术语翻译
消息队列	Kafka	事件驱动、链上事件监听
缓存	Redis Cluster	热点数据、会话管理、订单簿缓存
搜索引擎	Elasticsearch	全文检索、日志分析

5.3 数据层技术

类型	技术选型	用途
关系数据库	PostgreSQL 15+	核心业务数据、用户数据、映射表
时序数据库	TimescaleDB	行情数据、监控指标
搜索引擎	Elasticsearch	券搜索、日志分析
区块链	Genex Chain (Cosmos SDK + cosmos/evm)	券资产、交易记录、合规存证
链下缓存	Redis	订单簿、撮合引擎状态

5.4 AI/ML技术

模块	功能	框架/算法	部署
信用评分引擎	发行方与券信用评估	LightGBM + Scikit-Learn	FastAPI微服务
价格预测模型	券折价与市场波动预测	Prophet + LSTM	批量训练+在线推理
异常检测系统	洗钱、欺诈、异常交易	Isolation Forest + DeepSVDD	实时流处理 (Kafka消费)

风控决策引擎	实时风险预警与自动冻结	规则引擎 + ML模型混合	FastAPI + Redis Queue
智能推荐	个性化券推荐	协同过滤 + 内容推荐	P2优先级

5.5 基础设施

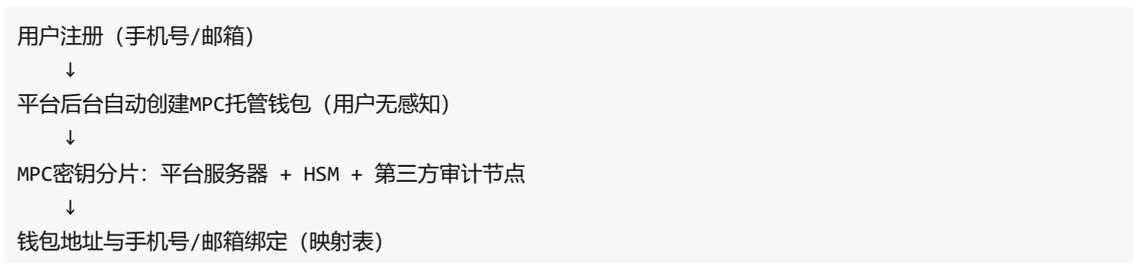
组件	技术选型	说明
容器编排	Kubernetes 1.28+	服务部署、弹性伸缩
服务网格	Istio	流量管理、可观测性
CI/CD	GitLab CI + ArgoCD	持续集成、GitOps
监控	Prometheus + Grafana	指标采集、可视化
日志	ELK Stack	日志收集、分析
链路追踪	Jaeger	分布式链路追踪
密钥管理	HSM + MPC	钱包私钥、加密密钥

6. 资产托管架构

6.1 三级资产控制模型

级别	用户画像	区块链可见性	平台控制力	消费隐私
标准模式 (默认)	99%普通用户	完全不可见	平台MPC托管	政策保证+ 审计保证
提取到外部钱包	想自持资产的用户	部分可见	平台无关	技术保证
Pro模式	加密原生用户	完全可见	用户完全自控	技术保证

6.2 MPC托管钱包架构



映射表安全方案 (防篡改核心) :

- MPC多方签名**: 映射记录创建/修改需多方共同签名
- Append-only审计日志**: 所有操作可追溯、可检测
- 映射哈希链上锚定**: Merkle Root定期写入Genex Chain
- 加密存储**: AES-256, 密钥由HSM管理
- 最小权限访问控制**: 读写权限严格分离

6.3 法币通道

场景	资金流向	说明
用户购买券	法币（银行卡）→ 法币通道 → USDC → Settlement合约	翻译层自动转换
用户卖出券	Settlement合约 → USDC → 法币通道 → 银行卡	翻译层自动转换
提现	平台账户余额 → 法币通道 → 银行卡	T+1（银行侧延迟）

7. 撮合引擎架构

7.1 链下撮合 + 链上结算



7.2 做市商接口

- 做市商准入：KYC L3 + 最低资金要求
- 低延迟专用API（10,000 req/min限额）
- 批量挂单/撤单
- Maker-Taker手续费模型（Maker费率低于Taker）
- 做市义务：维持最小挂单深度、最大价差限制
- Spoofing/Layering检测

8. 信用评级与定价模型

8.1 企业信用评级模型

评分公式：

$$\text{CreditScore} = 0.35 \times Rr + 0.25 \times (1 - Br) + 0.20 \times \log(Mt+1)/\log(36+1) + 0.20 \times Us$$

其中：

Rr = Redemption Rate（核销率） 权重：35%

Br = Breakage Ratio (沉淀率) 权重: 25%
 Mt = Market Tenure (市场存续月数) 权重: 20%
 Us = User Satisfaction (用户满意度) 权重: 20%

信用等级:

等级	分数范围	发行额度	监控级别
AAA	90-100	最高额度	常规
AA	80-89	高额度	常规
A	70-79	标准额度	标准
BBB	60-69	受限于额度	加强
BB及以下	<60	暂停发行	重点

动态更新触发:

触发事件	频率	影响
券核销	实时	+Rr
券过期	每日	+Br
用户投诉	实时	-Us
季度审计	季度	综合调整
违约事件	实时	直接降级

8.2 券定价模型

三因子公式:

$$P = F \times (1 - dt - rc - lp)$$

其中:

F = 券面值
 dt = Time Discount (时间折扣) 取值范围: 0%-15%
 rc = Credit Risk (信用风险溢价) 取值范围: 0%-20%
 lp = Liquidity Premium (流动性溢价) 取值范围: -5%-10%

因子计算规则:

因子	计算方式	取值范围
dt	线性衰减: (总有效期-剩余天数)/总有效期 × 最大折扣	0%-15%
rc	信用等级映射: AAA=0%, AA=3%, A=6%, BBB=10%, BB+=15%	0%-20%
lp	供需比: (卖单量-买单量)/总单量 × 调节系数	-5%-10%

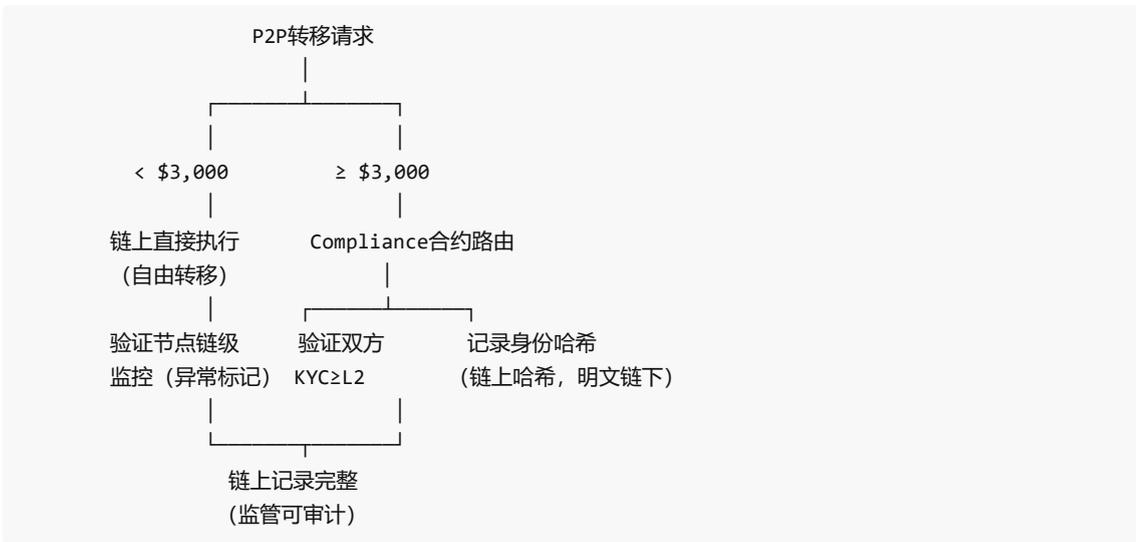
Utility Track 价格上限 = 面值。定价公式仅用于建议价。

AI辅助定价:

功能	算法	输入	输出
价格预测	LSTM	历史价格、成交量、时间、信用变动	价格走势与置信区间
最优挂单	强化学习	市场状态、库存、目标收益	建议挂单价格
异常检测	Isolation Forest	价格序列、成交模式	异常交易标记

9. P2P流通与合规路由

9.1 分层合规架构

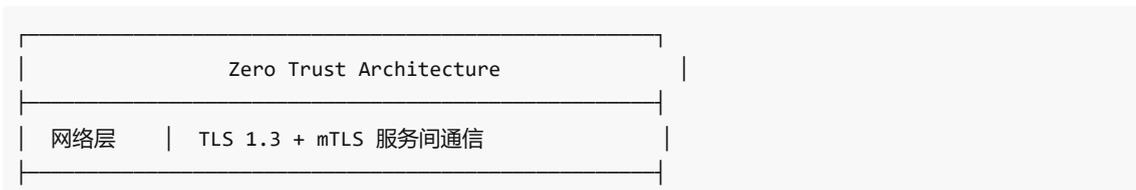


9.2 与"合约清算保护"不矛盾

场景	Travel Rule	平台介入	隐私
消费兑付	不适用	不介入	完全保护
小额P2P	不适用	不介入 (链级监控)	完全保护
大额P2P	适用	合规合约自动路由	链上仅哈希
二级市场交易	平台已知双方	撮合+结算	平台已知

10. 安全架构

10.1 零信任安全架构



应用层	OAuth 2.1 + MFA + RBAC
数据层	AES-256加密 + 字段级脱敏
审计层	全链路日志 + 链上不可篡改存证

10.2 安全要求

类别	需求	标准
传输安全	全链路HTTPS、服务间mTLS	TLS 1.3
数据加密	敏感数据加密存储	AES-256
身份认证	多因素认证	OAuth 2.1
访问控制	基于角色的权限管理	RBAC
审计追踪	操作日志、变更记录	不可篡改
渗透测试	定期安全评估	季度执行
智能合约审计	第三方安全审计	每次升级前
密钥管理	MPC/HSM	金融级

10.3 安全事件响应 (IR)

事件分级:

级别	定义	响应时限	示例
P0	资产被盗/合约漏洞被利用	15分钟内启动	MPC密钥泄露、合约攻击
P1	数据泄露/系统被入侵	1小时内响应	用户数据泄露、映射表篡改
P2	局部服务异常	4小时内响应	API被攻击、异常登录
P3	潜在风险	24小时内评估	依赖库漏洞、安全扫描告警

Bug Bounty计划:

- 严重漏洞奖励\$10K-\$100K
- 与Immunefi/HackerOne合作

10.4 合规标准

- **ISO 27001**: 信息安全管理体
- **PCI-DSS**: 支付卡行业数据安全标准
- **SOX**: Sarbanes-Oxley (Nasdaq上市要求)

11. 部署架构

11.1 多区域部署

部署架构		
AWS 美国 (主)	AWS 新加坡	香港节点
<ul style="list-style-type: none"> 核心业务服务 主数据库集群 Genex Chain 主验证节点 	<ul style="list-style-type: none"> 亚太用户服务 灾备数据库 CDN加速 区域验证节点 	<ul style="list-style-type: none"> 监管节点 审计接口 RegNode

11.2 灾难恢复

指标	目标
RPO	< 1分钟 (数据库实时同步至备用区域)
RTO	< 15分钟 (备用系统接管)
链上数据	天然灾备 (区块链分布式存储)
链下数据	定期快照 + 增量备份

11.3 部署要求

需求	描述
容器化	所有服务Docker镜像化
编排	Kubernetes + Helm Charts
配置管理	ConfigMap + Vault密钥管理
蓝绿部署	支持零停机发布
自动扩缩	HPA基于CPU/内存/自定义指标

12. GCFN全球清算节点架构

基于Genex Chain的BC跨链能力和多区域部署，构建全球清算节点网络。

Global Coupon Financial Network (GCFN)		
ROOT NODE (US / Global Hub) 全球路由 跨境清算 合规协调		
ASIA-PACIFIC	EMEA	AMERICAS
<ul style="list-style-type: none"> 新加坡(MAS) 香港(SFC) 	<ul style="list-style-type: none"> 伦敦(FCA) 法兰克福 	<ul style="list-style-type: none"> 美国(FinCEN)

• 日本(FSA)	(BaFin)	
-----------	---------	--

节点类型:

节点类型	部署位置	职责	数据存储
根节点	美国	全球路由、跨境清算、主验证节点	全局索引
区域节点	各大区	区域清算、监管报送、本地化	区域数据
本地节点	各国家	用户服务、本地支付、法规遵从	本地数据
监管节点	监管机构	只读审计、实时监控	审计日志

13. API架构

13.1 API设计规范

API规范:

风格: RESTful

版本控制: URL路径版本 (/api/v1/)

认证方式: OAuth 2.1 + JWT

限流策略:

普通用户: 100 req/min

企业用户: 1,000 req/min

做市商: 10,000 req/min

内部服务: 无限制

响应格式: JSON

错误处理: RFC 7807 Problem Details

13.2 核心API域

API域	路径前缀	主要接口
用户服务	/api/v1/users	注册、登录、KYC、资产查询
券服务	/api/v1/coupons	发行、查询、转让、核销
交易服务	/api/v1/trades	挂单、撤单、成交、历史
支付服务	/api/v1/payments	充值、提现、法币转换
企业服务	/api/v1/issuers	入驻、审核、数据统计
合规服务	/api/v1/compliance	KYC、AML、审计日志
翻译层	/api/v1/translate	地址映射、Gas代付、术语映射

13.3 SDK支持计划

SDK	语言/平台	功能范围	优先级
-----	-------	------	-----

genex-js	JavaScript/Node.js	全功能	P0
genex-java	Java/Android	全功能	P0
genex-python	Python	数据分析、批量操作	P1
genex-swift	Swift/iOS	移动端功能	P1
genex-go	Go	高性能服务端、做市商	P2

14. 性能需求

14.1 性能指标

指标	目标值	说明
Genex Chain TPS	≥ 5,000	链上交易处理
出块时间	≤ 1秒	CometBFT即时终结性
撮合延迟	< 10ms	链下订单簿撮合
API响应时间	P99 < 200ms	核心交易接口
翻译层延迟	< 50ms	UX翻译层额外耗时
系统可用性	> 99.9%	年度SLA

14.2 容量规划

阶段	用户规模	日交易量	Genex Chain节点	基础设施
Phase 1 (MVP)	10万	50万笔	3自有验证节点	单区域K8s
Phase 2 (商业化)	100万	500万笔	5+ 节点	双区域K8s+热备
Phase 3 (金融化)	1,000万	5,000万笔	10+节点	多区域+GCFN

15. 开发里程碑

Phase 1: 基础平台 (MVP, 仅Utility Track)

- Genex Chain主网上线 (Cosmos SDK + cosmos/evm)
- 手机号/邮箱注册系统 + MPC托管钱包
- UX翻译层 (地址映射、Gas代付、订单号映射)
- KYC分级认证 (L0/L1/L2/L3)
- 发行方入驻与审核 (零保证金, 信用成长)
- 消费型券发行与上架 (Utility Track, 链上铸造)
- 一级市场购买 (银行卡支付 → 链上结算)
- 基础二级市场 (价格上限=面值)
- 手机号P2P转赠 (含Travel Rule合规路由)

- 链上原子交换结算
- 基础风控与AML (OFAC筛查、链级监控)
- 商户核销App + 离线核销能力

Phase 2: 体验优化

- AI定价建议、信用评分体系
- 搜索/筛选优化、消息通知
- 完整数据分析看板
- 做市商接入

Phase 3: 发行方端增强

- 批量发券、高级数据分析
- POS SDK集成
- 多门店管理

Phase 4: Securities Track与扩展

- 取得法律意见书
- 开放Securities Track
- GCFN跨境节点部署
- 券资产证券化

16. 附录

16.1 术语表

术语	说明
Genex Chain	平台自建的EVM兼容应用链 (Cosmos SDK + cosmos/evm + CometBFT)
CometBFT	Cosmos SDK的共识引擎, 提供即时终结性
cosmos/evm	官方Cosmos EVM模块, 替代已弃用的Ethermint
IBC	Inter-Blockchain Communication, Cosmos生态跨链协议
MPC	Multi-Party Computation, 多方计算密钥管理
Utility Track	消费型券轨道, 价格≤面值, 规避SEC证券合规
Securities Track	投资型券轨道, 需完整证券合规 (Broker-Dealer等)
GNX	Genex Chain原生代币 (Gas + 治理)
GCFN	Global Coupon Financial Network, 全球券金融网络
Breakage	券过期未使用比例, 发行方潜在收益

16.2 参考文档

- 《Genex 券交易平台 - 软件需求规格说明书 v4.1》(权威源)
- 《券的金融本质与短期资金募集机制白皮书》

- Cosmos SDK 官方文档
- cosmos/evm 官方文档
- CometBFT 官方文档
- dYdX v4 架构文档
- Cronos 技术白皮书

文档版本: v3.0 创建日期: 2026-02-09 最后更新: 2026-02-09 基于: Genex 券交易平台 - 软件需求规格说明书 v4.1