

金融开放平台

总体架构设计方案

2019年04月18日

**目 录**

[1. 项目背景 2](#_Toc6509100)

[2. 总体目标 3](#_Toc6509101)

[3. 功能分布 4](#_Toc6509102)

[3.1 开发者社区 5](#_Toc6509103)

[2.2.1 功能分布 5](#_Toc6509108)

[2.2.2 功能点描述 5](#_Toc6509109)

[2.2.3 技术栈选型 7](#_Toc6509110)

[3.2 API网关 7](#_Toc6509111)

[2.2.1 功能分布 7](#_Toc6509112)

[2.2.2 功能点描述 8](#_Toc6509113)

[2.2.3 技术栈选型 9](#_Toc6509114)

[3.3 运营中心 10](#_Toc6509115)

[2.1.1 功能分布 10](#_Toc6509117)

[2.1.2 功能点描述 10](#_Toc6509118)

[2.1.1 技术栈选型 11](#_Toc6509120)

[3.4 微服务平台对接 11](#_Toc6509121)

[3.5 统一开发平台 12](#_Toc6509122)

[3.6 互联网账户体系对外输出 12](#_Toc6509123)

[3.7 统一日志管理平台对接 12](#_Toc6509124)

[3.8 集中监控系统对接 12](#_Toc6509125)

[3.9 消息平台对接 12](#_Toc6509126)

[3.10 文件安全检测平台对接 12](#_Toc6509127)

[3.11 影像平台对接 12](#_Toc6509128)

[3.12 核心系统 12](#_Toc6509129)

[3.13 在线支付系统 12](#_Toc6509130)

[3.14 企业网银系统 13](#_Toc6509131)

[3.15 非功能需求 13](#_Toc6509132)

# 项目背景

银行金融服务模式经历了从网点经济、APP经济到API经济的演进过程。在金融科技的赋能下，开放银行将推动银行金融服务渗透到生活的方方面面，具有**服务场景化、业务扁平化、参与多元化、能力综合化**的特点。~~面对开放银行浪潮，中小金融机构可以组成金融科技联盟，联合投资金融科技的研发单元，集中科技资源，统一研发、统一维护、共享成果、共担成本。~~

互联网环境下的企业和电商平台开放内容的深度和广度不断加大，各利益相关方由竞争转向相互开放、合作与竞争并存的行业形态，银行开放平台在这一环境下应运而生。**通过开放平台打造面向移动互联网的生态圈，并逐渐从流量导向朝流量变现过渡。**开放平台即依托于银行的金融服务、大数据服务能力，把物理银行及银行互联网平台的服务和数据向互联网开放。开放平台通过建设统一用户中心为开放平台实现银行现有用户和互联网用户的信息共享、减少用户信息壁垒，达到提升客户体验及用户信息整合的目的。开放平台将基于一套标准化协议及接口对第三方提供技术支持，充分带动吸引内外部开发资源。**基于开放平台所提供的接口和能力，开发者将根据客户个性化需求自行定制相关应用并发布，全力打造包括账户管理、金融咨询、投资理财、贷款服务、移动支付、民生缴费、生活服务等**金融和非金融功能的“混血”服务，最终实现“技术、产品、服务、合作”的全面开放。

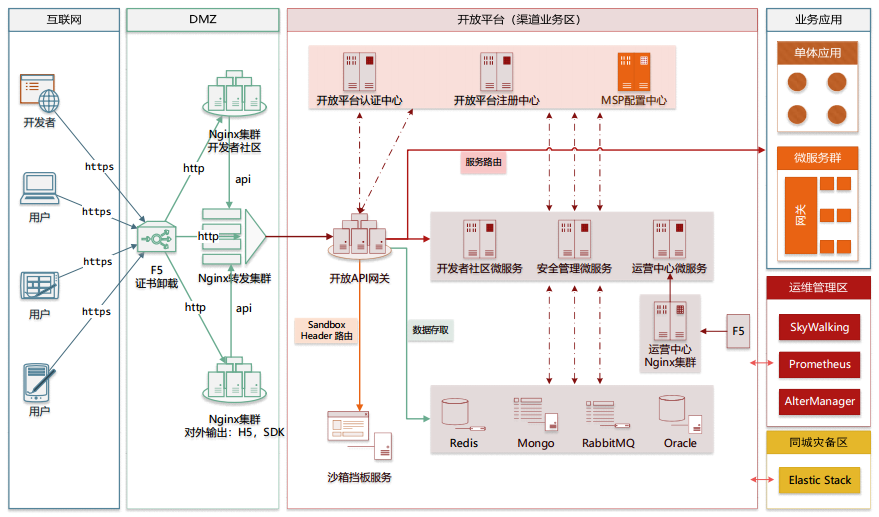
基于开放平台的统一标准体系，外部开发者可通过获得平台有关资源、信息、数据、安全等方面的专业化支持，开发自己的应用。随着开放环境的不断成熟，外部开发者可借助开放服务提供商的能力，实现自身应用价值。同时，也会有越来越多的外部开发者开发出基于开放服务提供商自身业务生态的衍生产品，为开放服务提供商本身的客户提供更好的服务。开放平台旨在为开放服务提供商和外部开发者提供一个规范、合作、创新、实现各自价值的开放社区平台。

行方将建成微服务平台和统一开放平台，可以快速应对变化，敏捷响应。另一方面，行方亟需一个互联网开放平台，并且已经具备了建设和运营平台的技术能力，开放行方的金融科技能力，为中小金融机构和企业合作伙伴或个人开发者提供金融 API聚合服务。**开发者们可以通过行方提供的金融API来完善其应用和业务，行方将获得流量转化和资金沉淀**，实现双赢。

# 总体目标

* 基于 Java技术栈开发
* 采用微服务架构，以SpringCloud体系为主
* 具有统一的服务治理方案
* 具有统一的可热更新的配置管理方案
* 具有可靠的故障隔离技术，单个服务/线程的故障不应对其他服务/线程产生影响
* 支持CI/CD，蓝绿发布/灰度发布， 0-downtime 提供服务
* 支持集成容器云环境，支持Docker镜像打包，支持Kubernates部署
* 开放源码，可进行代码修改、重构、编译、发布。

# 功能分布

金融开放平台整体结构如下图：

用户和开发者通过互联网以https形式访问，经过DMZ区F5时进行证书卸载转换为 http协议，将请求转发至开发者社区Nginx集群、对外输出H5 Nginx集群、Nginx转发集群，转发集群根据请求域名、路径等条件进行服务分发：将访问网关的请求按照域名和路径分为生产请求和sandbox请求，为sandbox 请求添加 Header，然后将请求转发至API Gateway。

API Gateway存在一个全局过滤器，优先顺序最高，存在sandbox Header的请求转发到挡板服务，其他请求根据路由规则进行路由。开发者社区页面和对外输出页面访问后台微服务的均需要经过转发集群和API Gateway。

开放平台自己独自的认证中心，对外部请求进行授权和认证，技术架构上采用同MSP同样的架构。开放平台需要部署独自的注册中心，采用MSP同构，用于开放平台自身微服务及将来业务服务聚合微服务的注册。配置中心复用微服务平台渠道业务区的Apollo配置中心。开发者社区，运营中心，安全管理等服务均采用微服务架构，对接注册中心、认证中心、配置中心。MongoDB用于微服务网关存储路由信息；Redis用于微服务认证中心缓存Token、密码，另外网关限流功能也需要使用Redis；RabbitMQ为异步消息使用；Oracle为OLTP数据库。

调用链监控完全复用微服务平台 SkyWalking技术体系，指标监控使用微服务平台

Prometheus技术体系，告警使用AlertManager，日志使用EFK/ELK收集。

金融开放平台主要包含三大新子系统：**开发者社区、API网关、运营中心**。

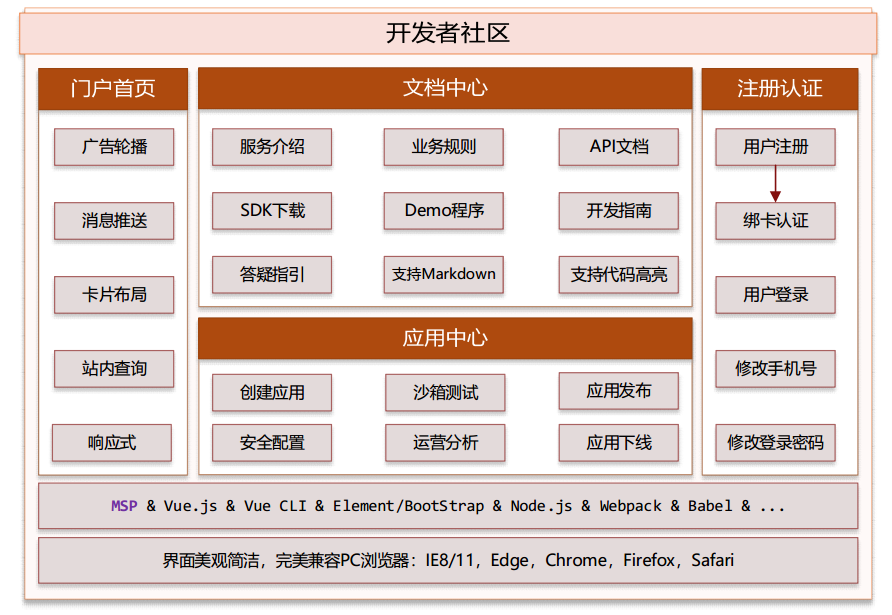
## 开发者社区



### 功能分布

开发者社区面向互联网，对企业开发者和个人开发者，开放注册、实名认证、开发者相关信息维护、应用创建、应用发布、SDK 及其说明、开发者集成流程及指引、产品信息展示、广告和新闻发布、业务场景介绍、金融产品发布、知识库等功能。

开发者社区功能结构如下图：



### 功能点描述

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | **用户注册登陆** | （1） 提供用户自助注册功能。个人用户在注册时需要提供五要素（证件类型、证件号码、姓名、我行银行卡号、预留手机号），且个人用户必须是我行 I 类账户  的拥有者；企业用户在提供证明材料的基础上必须洽谈面签方可完成实名认证，另外需要提供对公打款验证等方式，线上验证企业用户身份，规避业务风险。只有完成以上实名认证和验证，并通过审核的用户可以创建应用和使用 API。  （2） 支持通过社交账号直接登录，首次登陆后要求继续提供用户信息（与个人用户在注册时需要提供的内容一致） 进行注册，注册后关联社交账号，再次登陆可  直接登陆。  （3）已审核的用户可以临时或永久禁用，禁用后的用户不能再登录系统和使用API。  （4） 提供防止暴力注册的校验，防止使用机器人进行暴力注册攻击 |
| 2 | **图像存储** | 采用合理的方案来**临时**存储用户上传的证照等非结构化数据， DMZ 区不要允许使用 NAS，应使用分布式 DFS 文件系统。  其次用户上传的非结构化数据需要对接文件安全检测平台进行文件查毒等安全检查。最终证照数据永久存储到影像平台中，后续从影像平台查询展示。 |
| 3 | **用户中心** | 用户中心展示用户相关信息，提供应用的快速入口，提供修改手机号，登陆密码等安全设置功能。 |
| 4 | **应用创建** | （1） 提供应用创建功能，创建应用时分配 APPID，上传用户公钥，设置通讯密码，安全回调域名，应用描述，申请 API 以及配额等功能  （2）只有实名后并审核过的用户可以创建应用。  （3）应用可以临时或永久禁用，禁用后应用对应的 API 不能使用。 |
| 5 | **沙箱测试** | 应用创建后，可以申请沙箱环境进行沙箱测试 |
| 6 | **应用发布** | 测试通过后可以申请发布应用 |
| 7 | **安全配置** | 可以对应用的安全配置信息进行修改 |
| 8 | **运营分析** | 提供应用运营数据报表图表 |
| 9 | **应用下线** | 可以手动下线应用，下线后对应APPID失去API访问权限；且下线后的应用，重新审核通过后，可再重新恢复上线。 |
| 10 | **轮播通知/广告** | 支持在首页展示轮播广告，广告可以在运营中心进行配 |
| 11 | **消息通知** | 支持服务端实时的消息推送，并在前端展示推送消息 |
| 12 | **服务介绍** | 提供开放平台的服务介绍，业务规则介绍等资料的网页展示 |
| 13 | **API文档** | 提供结构清晰，逻辑严谨，美观简洁的API文档网页展示 |
| 14 | **开发指南** | 提供完整和详尽的开发者指南，开发者可以直接据此进行开发工作 |
| 15 | **SDK下载** | 提供Java、iOS、Android、 JS等平台SDK下载，SDK下载支持选择版本 |
| 16 | **Demo 程序** | 提供 Demo 程序的在线展示和下载，支持代码高亮 |
| 17 | **答疑指引** | 提供答疑和指引模块，开发者可以按照问题进行查找解决方案，提供留言功能，留言可以在运营中心进行显示，审核和答复。留言支持富文本和 markdown 格式 |
|  |  |  |

### 技术栈选型

**选型原则：**优先考虑社区活跃、使用率高的开源产品，选择当前社区的较新稳定版本，优先考虑社区版，在社区版不满足需求的前提下，可酌情考虑企业版本。另外，因为开发者社区需要在外网发布，因此要求具有较好的浏览器兼容性。目标选型如下：

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 开源组件 |
| 1 | **JS框架** | Vue.js |
| 2 | **UI框架** | Bootstrap or Element |
| 3 | **前端基础框架**  **前端组件管理工具** | Node.js & NPM |
| 4 | **前端构建工具** | Webpack |
| 5 | **接口文档显示** | Swagger |
| 6 | **Java环境** | Oracle JDK |
| 7 | **Java框架** | SpringBoot |
| 8 | **分布式缓存** | Redis |
| 9 | **声明式服务调用** | Spring Cloud Feign |
| 10 | **客户端负载均衡** | Spring Cloud Ribbon |
| 11 | **服务容错保护** | Spring Cloud Hystrix |
| 12 | **服务限流** | Google Guava RateLimit + Redis |
| 13 | **链路跟踪** | SkyWalking |
| 14 | **监控指标收集** | Prometheus |

## API网关

### 功能分布

API 网关是金融开放平台核心基础功能。主要包含服务发布、安全控制、服务路由、过滤器等技术集成等功能，为 API 的调度提供安全、稳定、可靠的运行环境。同时还提供流量控制、熔断降级、存储转发、故障隔离等一系列平台功能，具有安全可靠高效稳定的特点，采用主流开发平台和语言，提供良好的二次开发接口。提供完善的管理功能，对运行平台上的相关功能进行参数的配置和管理。

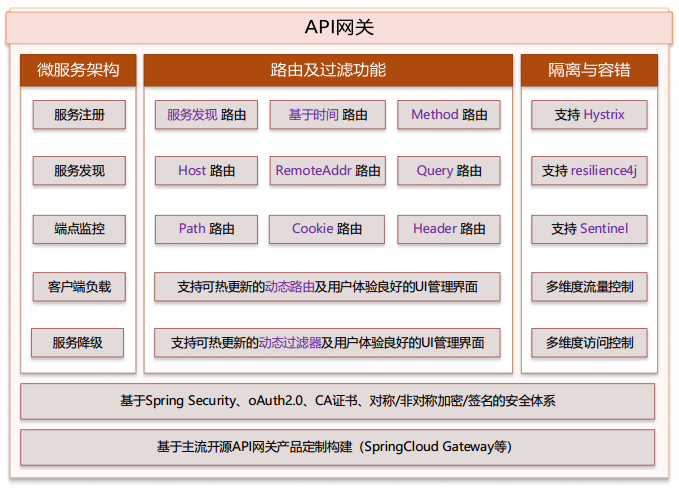
API 网关的核心要求是：安全、高效、稳定、标准。 需满足以下要求：

1. 标准服务发布采用 Restful API，采用 https/json 格式进行数据交换；

2. API访问授权机制采用 OAuth2.0体系，API访问均需受控；

3. 提供服务路由、负载均衡、故障隔离、熔断限流、流量统计、访问控制、异常处理等基础服务；

4. 针对特殊的场景，提供协议/报文转换功能。



### 功能点描述

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | **技术框架** | 基于 SpringBoot微服务技术开发，可以方便的集成 Eureka/Consul 服务发现，Apollo配置中心，SkyWalking链路监控，Prometheus指标监控。 |
| 2 | **路由** | 提供丰富的灵活的路由功能，可以实现：基于服务发现的路由规则、基于时间的路由规则、基于 Method、Host、RemoteAddr、Query、 Path、Header、Cookie 的路由规则；可以使用以上路由规则进行组合实现灰度发布、蓝绿发布、负载测试等功能；提供动态路由配置功能，并支持配置持久化，提供简洁美观的路由配置 UI 界面；动态路由可热发布；支持路由权重配置 |
| 3 | **过滤器** | 提供配置化，免开发的灵活过滤器功能，过滤器可以热发布。  可通过过滤器实现：修改 Path、Header、RequestBody、ResponseBody、重定向等功能。提供基于过滤器的安全认证、 身份识别、 token 校验功能、流量控制等功能。 |
| 4 | **服务注册发现** | 网关可以作为一个微服务注册到注册中心，支持Eureka、consul |
| 5 | **端点监控** | 提供标准的监控端点，可以对接 Prometheus 等监控平台。  可以通过控制端点对网关进行控制。 |
| 6 | **负载均衡** | 支持客户端软负载 Ribbon 集成，对可用服务采用多种可配置的负载策略进行请求分发。 |
| 7 | **服务降级** | 路由的服务不可用时，提供个性化的降级服务 |
| 8 | **熔断隔离** | 可以集成 Hystrix、Resilience4j、Sentinel来实现熔断和隔离 |
| 9 | **多维流量控制** | 提供基于多维度（客户端 IP， APPID，用户 ID， API 等）流量控制方案 |
| 10 | **访问控制** | 提供基于多维度（客户端 IP，APPID，用户ID，API等）进行访问权限控制，可以对API进行配额（如可以配置某APPID每天可以访问某个API上限为5万次）。 |
| 11 | **多协议支持**  **报文转换** | 支持HTTP，TCP，RPC协议，支持协议转换、报文转换 |
| 12 | **安全访问控制** | API访问受控，提供身份认证，鉴权，AccessToken访问等方案保护 API；提供对禁用的用户下的所有API和禁用的应用下的API将无法使用；提供对已经下线的API将无法使用；提供对于Token 的加密，报文的全加密； |
| 13 |  |  |
| 14 |  |  |
| 15 |  |  |
| 16 |  |  |
| 17 |  |  |
|  |  |  |

### 技术栈选型

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 开源组件 |
| 1 | **Java Runtime** | Oracle JDK 1.8.202 |
| 2 | **微服务框架** | SpringBoot、SpringCloud |
| 3 | **网关** | SpringCloud Gateway 或 Kong |
| 4 | **声明式服务调用** | Spring Cloud Feign |
| 5 | **客户端负载均衡** | Spring Cloud Ribbon |
| 6 | **服务容错保护** | Spring Cloud Hystrix |
| 7 | **微服务配置中心** | Apollo |
| 8 | **服务注册中心** | Eureka 或 Consul |
| 9 | **分布式缓存** | Redis |
| 10 | **网关管理数据库** | MongoDB |
| 11 | **数据库客户端** | Spring Data MongoDB Reactive |
| 12 | **链路跟踪** | SkyWalking |
| 13 | **监控指标收集** | Prometheus |
| 14 | **接口文档生成** | Swagger |

## 运营中心



### 功能分布

运营中心提供了强大的管理控制台，包括网关管理、内容管理、应用管理、用户管理四个主要模块。



### 功能点描述

|  |  |  |
| --- | --- | --- |
| 序号 | 功能名称 | 描述 |
| 1 | **路由管理** | 对 API 网关的路由策略、过滤器进行管理，提供简洁美观的 UI 管理界面，路由信息修改后可以实时发布，实时生效。 |
| 2 | **访问控制管理** | 管理网关的访问控制策略，可实时生效 |
| 3 | **流量控制管理** | 管理网关的流量控制策略，可实时生效 |
| 4 | **API管理** | API新增，修改，删除，可配置API的访问控制和流量控制策略，可配置API的不同版本的负载配额。 |
| 5 | **文档管理** | 提供 API 文档、开发者指南、 SDK 更新发布、答疑指引、服务介绍、业务规则等内容管理，文档支持在线编辑，支持 markdown 格式，支持离线 markdown 文档导入，支持富文本编辑器，支持附件上传。 |
| 6 | **轮播管理** | 可以对首页上的轮播通知或广告进行配置和管理 |
| 7 | **应用发布审核** | 对开发者提交的应用发布申请进行审核和发布；下线应用。 |
| 8 | **沙箱管理** | 可在线维护沙箱环境的挡板数据，新增 API 接口后应同步维护沙箱环境 |
| 9 | **配额管理** | 管理各 API 的默认配额（访问量/时间维度），可以针对不同的维度设置配额 |
| 10 | **运营报表** | 提供运营数据查询功能，提供多维度运营数据报表 |
| 11 | **开发者管理** | 对开发者的注册信息进行审核；开发者安全信息设置和重置；开发者禁用和恢复；开发者运维数据查询和报表查看； 可以代合作方进行用户注册，实名认证，应用注册。 |
| 12 | **平台用户权限**  **角色管理** | 运营中心的用户权限角色管理 |
| 13 | **审计日志** | 平台用户操作的日志 |
| 14 |  |  |
| 15 |  |  |
| 16 |  |  |
| 17 |  |  |



### 技术栈选型

同开发者社区，但对浏览器兼容性方面要去可以稍作降低，Chrome（含使用Chrome内核的主流国产浏览器，如360/QQ/Sogou）和Firefox 浏览器中可完美兼容即可。

## 微服务平台对接

开放平台复用微服务平台的以下技术组件：

* **注册中心**

独立部署一套注册中心，采用微服务平台 Eureka 技术组件。用于开放平台内部各微服务的注册，以及对接服务输出的聚合服务微服务的注册。

* **认证中心**

独立部署一套认证中心，采用微服务平台 oAuth2.0 技术组件。用于对外部合作企业进行认证和授权。

* **配置中心**

直接复用微服务平台渠道业务区的 Apollo 配置中心。

* **SkyWalking 调用链分析系统**

使用微服务平台的调用链分析 Agent，对接微服务平台渠道业务区的SkyWalking Collector。

* **Prometheus 指标监控系统**

采用微服务平台的监控埋点技术，对接微服务平台渠道业务区的 Prometheus集群。

## 统一开发平台

采用统一开发平台提供的公共技术组件和技术规范，后期进行微服务聚合服务开发、开发者社区、运营中心 UI 界面开发时使用 UDP 开发工具进行开发。

## 互联网账户体系对外输出

对接互联网账户体系，实现互联网账户的对外输出：包括 API 输出，SDK 输出，HTML

输出。

## 统一日志管理平台对接

通过在服务器安装统一日志管理平台的采集客户端，收集各微服务的日志，重新组织并统一保存，由统一的日志分析和检索工具进行处理，为运维、开发人员提供分析数据以及定位问题的依据。

## 集中监控系统对接

通过采集数据（网络流量、系统运行数据、服务节点数据、业务数据等信息），汇总保存，然后进行实时和定期分析的方式，得到开放平台系统的运行情况报告，并在监控指标超过阈值时，调用我行集中监控系统 API 接口，报警并通知指定的干系人。

## 消息平台对接

需要通过短信平台发送短信验证码；需要通过邮件系统发送激活链接邮件。

## 文件安全检测平台对接

用于检测图片等非结构化数据文件，保障文件安全，保障系统免于被病毒或木马程序攻击。

## 影像平台对接

用于传输和存储用户注册过程中上传的企业证照、身份证件、收款证明等影像资料。涉及到影像资料的上传、查询与更新。

## 核心系统

如果开发者为个人，且为我行存量客户，实名认证过程中需要通过核心系统查询客户的银行卡、三证及预留手机号，是否匹配，以校验客户的身份。

## 在线支付系统

如果开发者为个人，非我行存量客户，通过在线支付系统进行四要素鉴权，以验证

用户信息是否合法，是否是开发者本人，进而完成用户实名认证。

## 企业网银系统

如果开发者为企业客户，通过企业网银系统对企业客户的对公账户打随机金额款项，

然后客户提交收款证明，用以验证客户身份的真实性，完成企业开发者实名认证。

## 非功能需求

|  |  |  |
| --- | --- | --- |
| 类别 | 模块 | 说明 |
| 开源 | 开源软件 | 技术栈基于开源软件，厂商在开源软件上进行封装的代码需要开源，自研模块全部开源。在主流开源框架较新稳定版本上进行的二次开发，后端支持 SpringBoot，前后端分离，前端支持动态编译技术。 |
| 浏览器兼容性 | 外网 | 需要兼容主流浏览器： IE11， Edge， Chrome，  Firefox， Safari， PC 端完美兼容，移动端体验良好。 |
| 内网 | 需要兼容： IE11， Chrome，PC 端完美兼容体验良好 |
| 性能指标 | 开放网关 | 网关单个节点： 2000+ TPS |
| 开发者社区 | 支持 200 用户同时在线，支持500 QPS |
| 运营门户 | 支持 20 用户同时在线，支持50 QPS |
| 安全 | 互联网系统 | 符合行业安全规范要求，符合监管对金融开放平台（Openbanking）的安全规范要求。对于开发者社区个人用户和企业用户均要求强实名认证。API、SDK、HTML的输出，提供不同维度的安全防护功能，包括不限于：基于OAuth2授权框架的 AccessToken模式， JWT模式；基于证书和非对称算法的签名验签；基于对称算法报文AES加密；基于绑定对端IP或域名的访问控制等安全手段。 |
| 交付文档 | 使用规范 | 面向开放平台开发人员。应用对接与使用平台及功能模块的规范、技术指导及示例。 |
| 操作手册 | 面向开放平台运维人员。开放平台各模块的操作手册。 |
| 培训方案 |  | 对于使用该平台的应用开发人员，应对其进行15个工作日以上的平台使用及开发培训。 |
|  |  |  |
|  |  |  |