

# 金融服务平台——数据区块链存储

## 软件开发计划书

目录

概述(Develop Plan Summary) .....1

一、引言 (Introduction).....1

    1.1 编写目的 (Purpose).....1

    1.2 开发背景 (Background).....1

    1.3 参考资料 (References) .....1

    1.4 相关文档 (Related Documents) .....1

二、项目概述 (Project Summary).....3

    2.1 工作内容 (Project content) .....3

    2.2 开发环境 (Develop Environment).....3

    2.3 开发工具 (Develop Tools).....3

    2.4 人员分工 (Division of Work) .....3

三、进度安排及验收标准 (Schedule & Test Standards) .....4

    3.1 项目进度规划表 (Schedule).....4

    3.2 项目验收标准 (Test Standards).....4

四、风险管理计划 (Risk Control Plan).....6

    4.1 风险列表 (Risk Lists) .....6

    4.2 风险规避 (Risk Avoidance).....6

附录 (Appendix).....7

## 概述(Develop Plan Summary)

本项目使用 Hyperledger Fabric 技术，实现了基于区块链技术的去中心化金融支付交易服务平台的储存系统。该系统在具有可靠的数据安全性和系统运行稳定性的同时拥有较为出色的储存与查询效率。

## 一、引言 (Introduction)

### 1.1 编写目的 (Purpose)

本计划书用于指导金融服务平台中区块链数据存储模块的设计与开发，并且最终取得合格的产品，本计划面向项目组的全体成员。

本计划书阐述了项目组的工作内容、明确了组员的职责、指导组员协作、保障质量把控等工作，力求项目能够有条不紊地进行，增强开发的目的性，降低开发风险，减少软件开发中的时间、资源、人力上的损耗和浪费。

### 1.2 开发背景 (Background)

随着区块链概念和产业的迅速发展，区块链技术近年来收到了越来越多的关注，也产生了众多衍生的概念，去中心化存储正是其中之一。区块链是按时间顺序将数据区块以顺序相连的方式组合成的一种链式数据结构，并拥有由密码学原理保证的不可篡改和不可伪造的特性。区块链技术以其去中心化、开放性、信息不可篡改性、匿名性等优秀特性，被广泛地应用在金融行业中。

### 1.3 参考资料 (References)

- [1] 《GB8567—88 计算机软件产品开发文件编制指南》
- [2] 吕云翔.软件工程实用教程.北京：清华大学出版社，2015.
- [3] 《需求规格说明书》

### 1.4 相关文档 (Related Documents)

- [1] 《需求规格说明书》
- [2] 《软件设计说明书》

## 二、项目概述 (Project Summary)

### 2.1 工作内容 (Project content)

本项目实现用户充值、转账、提现信息的入链与查询功能。用户进行充值、提现、转账等操作后，相应的信息会被逐条地记录在区块链储存系统。同时系统向上层的综合运营平台提供相应查询接口，以支持其完成有效的查询操作。

### 2.2 开发环境 (Develop Environment)

本项目采用 PC 开发，区块链使用 go、shell 等语言，Java 作为主要开发语言，在 Hyperledger Fabric 的基础上进行封装，其中建立的区块链网络需要在支持 Shell 系统的 Docker 上运行。WEB 服务器选用 Tomcat。

### 2.3 开发工具 (Develop Tools)

**Hyper Ledger Fabric:** Hyper Ledger Fabric 是企业构建的领先的开源、通用区块链结构。

**Docker:** Docker 是一个开源的应用容器引擎，开发者可以利用它打包应用以及依赖包到一个可移植的容器中，然后发布到任何 Linux 机器上，也可以实现虚拟化。

**IntelliJ IDEA:** IntelliJ IDEA 是一个开发 Java 语言的集成环境。

**Tomcat:** 是一个免费的开放源代码的 Web 应用服务器，是 Apache 项目的扩展，技术先进、性能稳定。

### 2.4 人员分工 (Division of Work)

核心代码编写与测试：余志浩、余天予、田宏远；

项目设计与文档写作：邓坤权、马大慈。

另外，在代码开发的关键阶段，项目设计与文档制作组也参与到核心代码编写。

### 三、进度安排及验收标准 (Schedule & Test Standards)

#### 3.1 项目进度规划表 (Schedule)

时 间 名 称	进 度	持续天数（天）	开始时间	结束时间
环境配置		7	2019/3/25	2019/3/31
讨论需求、设计接口		21	2019/3/25	2019/4/14
需求分析（含完成文档）		35	2019/3/25	2019/4/28
程序框架设计（含设计说明书）		49	2019/3/25	2019/5/12
搭建底层节点(Peers)		7	2019/4/8	2019/4/14
实现业务逻辑链码(Chain code)		14	2019/4/15	2019/4/28
链码层基础上搭建应用		9	2019/4/29	2019/5/7
实现接口		5	2019/5/8	2019/5/12
包装接口为服务并注册到系统后端		5	2019/5/13	2019/5/17
实现上层查询、修改		5	2019/5/13	2019/5/17
与其他组信息传输		6	2019/5/19	2019/5/24
测试分析		7	2019/5/25	2019/5/31
部署文档		7	2019/5/25	2019/5/31
用户使用手册		7	2019/5/25	2019/5/31

#### 3.2 项目验收标准 (Test Standards)

项目验收表		
1	实现目标	数据区块链存储系统的功能完善及稳定性
	验收方法简要说明	使用系统测试工具，使用一段时间后是否稳定；检查综合运营平台调用的查询功能是否出现异常
2	实现目标	配置服务器服务支持
	验收方法简要说明	数据能够正常有效的写入区块链
3	实现目标	配置数据库区块链存储系统的安全性设置
	验收方法简要说明	使用攻击工具或者有意上传病毒来破坏区块链
4	实现目标	配置数据库区块链存储系统的运行服务器
	验收方法简要说明	使用外网进行各种条件的测试
5	实现目标	配置区块链的运行模式

	验收方法简要说明	安装必要的支持软件来为区块链的运行和恢复做好准备
6	实现目标	对系统的整体维护设计规划
	验收方法简要说明	针对区块链相关检查以及维护措施
7	实现目标	区块链备份恢复
	验收方法简要说明	查看备份时候产生日志中无异常错误，同时能够在模拟环境中恢复数据
8	实现目标	与上层运营平台的兼容程度
	验收方法简要说明	由运营平台的运维人员对储存系统的接口进行读取测试
9	实现目标	与项目组其他模块的兼容程度
	验收方法简要说明	与相关的项目组开发人员进行测试与确认

## 四、风险管理计划 (Risk Control Plan)

### 4.1 风险列表 (Risk Lists)

(1) 设计风险：在设计系统时可能会有疏漏不周之处，使系统存在先天的设计缺陷，导致系统无法正常地运行。

(2) 技术选型风险：尽管区块链技术的热度很高，但其仍然属于较为新兴的技术，支持文档和参考资料不够丰富。项目选择的框架也较为新颖，可能存在各种不可预期的问题。这两点给项目开发带来了一定的不确定性。

(3) 系统对接风险：本模块是金融服务平台的一个底层数据模块，同时还需要与综合运营平台对接。因此与系统其他模块的兼容性、与上层平台的对接，以及其他项目组的进度都会对本组的开发、测试和优化产生影响。

(4) 技能风险：本项目组的开发人员都是第一次接触类似的项目，对于开发所要求的技能和经验需要从头学习和积累。

(5) 沟通风险：项目组人数较多，其中沟通协调的成本不可忽视。

### 4.2 风险规避 (Risk Avoidance)

(1) 规避设计风险：在设计伊始，需求方、设计者和实现者应当进行全面充分的沟通，对可能出现的问题进行讨论，给出较为合理的设计思路，尽量减少失误。

(2) 规避技术、技能风险：代码编写人员应当对项目涉及到的技术的官方文档较为了解，同时对项目的总体结构和框架尽可能熟悉，尽量使用主流技术、主流方法实现功能，便于参考、测试等。

(3) 规避系统对接风险和沟通风险：项目组首先应该考虑到沟通成本，在规划开发时间表的时候预留足够的时间和资源进行沟通。同时在设计阶段、开发阶段、测试阶段等不同过程中尽量保持充分沟通。

(4) 使用版本管理工具，对项目进行完善的备份，便于修改、回滚。



附录 (Appendix)

项目进度甘特图

