

# HIP-12 Hachash 开发工作流程及代码库权限...

由于 Hachash 完全去中心化的目标和状态，没有预挖、没有ICO、没有开发抽税、没有资本投资、也没有基金会运作，开发工作没有任何资金来源，很长时间（近四年）开发组完全处于由兴趣驱动、免费义务为社区贡献代码的状态。开发工作完全遵循 [Hachash 白皮书](#) 的愿景，符合社区共识，做到机制的公平和安全。所有源码 100% 开源，并全部汇集在 <https://github.com/hachash> 仓库内。

从 Hachash 的 Github 提交日志上来看，截止到2023年2月，[Jojoin](#)（Hachash主要核心开发者）贡献了90%以上的代码，包括主网、钱包和SDK以及开发文档等。由于Hachash 发展缓慢，暂未取得主流市场的关注，没有更多的人参与到 Hachash 的开发工作，也没有形成一个具备规模的开发者社区。主网核心代码的合并权限目前也由 [Jojoin](#) 一个账号管理。

本提案的目标在于，参考比特币、以太坊等成熟加密项目的源代码管理机制，形成一套完备的 Hachash 开发工作流程，以及定义 Hachash 各个开发组成员的分类、参与资格等权限问题。

## 一、开发组架构

Hachash 的开发团队组织架构分为三个层级：

1. 核心组
2. 核心预备组
3. 应用组

应用组的数量和每组参与人数不设限制。应用组可以围绕不同的部分或主题，形成多个平级的组，比如钱包组、区块浏览器组、通道链钱包组等等。当单个应用组参与人数过多，或者开发工作量过大，影响到相互之间沟通和合作时，可以分裂成两个或多个应用组，一个开发者也可以同时参与到多个应用组。每个应用组的组长由核心组或核心预备组的成员来担任。

核心预备组的成员不超过25人。主要工作为以组长身份审阅各应用组成员提交的代码，检查潜在的安全漏洞或性能问题，参与公链核心的设计和支付、扩容协议层的设计和讨论，并为公链层提交试验性或单元测试代码，以及对提交到公链层的代码进行预先审阅。

核心组的成员不超过5人，拥有公链层代码主版本的合并权限，且对各架构方案、协议设计有直接的决策权力。对不改变核心价值共识的主网代码提交拥有直接合并权限，对公链的不兼容升级、协议底层修改和货币价值模型变更等重大事项，拥有投票权。涉及公链货币价值层和核心协议层重大升级，变更可获得通过的有效票数比例为：1/1 、 2/2 、 2/3 、 3/4 或 3/5 。弃权或超期不投票则视为反对票。

以上层级不代表代码贡献量的区别，而是源代码合并权力的区分。不成为核心组成员，也可以是十分重要的代码贡献者，只不过不承担代码审阅和源码管理职责。

要成为每个级别的组的成员，需要满足一些条件：

应用组：

无限制。合并代码权限由组长及副组长管理。

核心预备组：

0. 熟练掌握C++、Java、Golang或Rust其中一种编程语言。
1. 为 Hacash 应用层贡献6000行源代码，或者为hacash公链层贡献2000行源码以上
2. 独立开发过一个 100 star 以上的纯源代码开源项目（非资料整理或文档类）
3. 熟悉 Hacash 的技术路线、货币理论和最终目标，并初步了解 Hacash 公链底层架构

核心组：

0. 精通C++、Golang或Rust中至少两种编程语言
1. 为 Hacash 公链层贡献过1万行以上的代码，并且对公链底层主要模块进行过重大重构或升级。有充足的时间（每周不低于20个小时）负责 Hacash 的技术管理和重要决策工作
2. 开发和管理过2个以上200star的源代码开源项目。为5个或以上开源项目贡献过代码
3. 完全熟悉 Hacash 公链底层的基础架构和实现细节，并有能力在架构层面进行设计和重构，有参与过通信协议、支付接口或p2p网络协议的设计和开发最佳
4. 充分研究 Hacash 所有理论、技术资料，掌握 Hacash 的底层本质和核心价值来源，完全理解 Hacash 的去中心化和公平路线、技术方向和货币理论，了解 Hacash 各方面的权衡、限制和边界，赞同 Hacash 的最终使命
5. 在整个 Hacash 社区积累一定的社交信用，展现优良的品格和充足的工作能力。获得社区信任，始终保持完全公平和正直的思想，不代表任何单方面的立场，不为任意个人或团体谋取不公正的利益，不做任何直接或潜在损害项目长远发展的行为，不谋求为私利而推翻 Hacash 已经获得的价值共识

## 二、开发工作流程

Hacash 的开发工作分为四种类别，分别对应不同的权限和流程：

1. 应用层开发
2. 公链层 bug 修复
3. 公链接口层新功能开发
4. 公链核心价值层或协议层变更

第一类开发工作的权限将由个应用组的组长做最终的审阅和代码合并工作。核心预备组和核心组保持推翻和回退应用组提交的代码的权力。

第二类开发工作由核心预备组完成代码审阅，由核心组成员直接进行代码合并。

第三类开发工作，由核心预备组执行第一次代码审阅，并提交给核心组完成最终的代码审阅，然后由核心组成员经过简单讨论沟通后，执行代码合并。

第四类开发工作，需要整个 Hacash 社区的参与，并结合 Hacash 的 HIP 系统（Hacash Improvement Proposal），按照以下流程执行：

1. 由社区或开发组提议一个新的 HIP，在 Github、Discord、HacashTalk论坛或其它 Hacash 的公开社区进行话题讨论，获得整个社区的反馈。
2. 话题讨论获得一些确定性成果，由提出此 HIP 的相关人负责归纳和整理所有讨论资料，形成初步的参考文档。
3. 继续讨论该提案各方面的细节，提供测试或验证代码，在没有明显的重大争议或漏洞之后，形成一个较完备的 HIP 草案。
4. 将 HIP 草案提供给所有能触达Hacash社区的渠道，一段时间后，社区没有更多反对意见，提交给核心组成员进行投票。
5. 若核心开发组投票通过，则向社区公布此 HIP 获得通过，并安排相应开发工作。

以上 HIP 流程的推动，各个讨论、资料的整理，文档、草案的发布，由原提案人负责执行。