

MoonModGuard 开源技术说明书

MoonBit 项目清单与供应链策略审计器

GitHub 主仓库	https://github.com/Noverberrain/MoonModGuard-MoonBit-
GitLink 镜像仓库	https://gitlink.org.cn/Wyc060514/moonmodguard
核心作者	wyc060514
开源许可证	Apache-2.0
参赛方向	MoonBit 国产基础软件开源生态项目

定位 面向 MoonBit 工程清单的发布前自检与策略审计工具。	对象 解析 moon.mod、moon.pkg, 提取元数据、 依赖和导入别名。
价值 补齐清单完整性、许可证策略和依赖透明度检查 能力。	交付 提供核心库、CLI 示例、测试、CI、README 与申报材料。

评审摘要

MoonModGuard 面向 MoonBit

项目的工程清单质量与供应链策略检查，提供轻量、可测试、可嵌入的审计内核。它解析 moon.mod 与 moon.pkg 文本，抽取项目元数据、包级依赖、导入别名和发布信息，并将这些信息统一转化为可复查的策略诊断报告。首版优先补齐发布前自检能力，而不是构建平台型服务。

选题价值

MoonBit 生态会持续积累包、模板和教学工程。随着项目数量增加，单个仓库是否具备可发布、可维护、可审计的基础条件，会直接影响生态复用效率。MoonModGuard 将人工阅读清单文件的经验沉淀为基础软件能力，适用于包发布、竞赛验收、课程规范检查和后续 CI 策略审计。

技术方案

系统采用清单解析层、项目建模层、策略评估层和报告生成层。解析层读取 moon.mod 与 moon.pkg 字符串；建模层合并模块级与包级信息；策略层检查必填元数据、许可证 allowlist、可信依赖前缀和重复依赖；报告层输出稳定 Markdown，供 CLI、CI 或文档系统复用。

实现边界

首版采用显式输入字符串与内存快照的方式，不做真实文件系统扫描、在线包仓库查询或完整 MoonBit 语法解析。这个边界降低实现风险，使核心逻辑更容易测试，也便于后续扩展真实工作区扫描器。

验证证据

当前验证覆盖 moon.mod 标量字段解析、keywords 数组解析、模块级 import 解析、包级 import 解析、缺失元数据诊断、非 allowlist 许可证诊断、未知依赖前缀诊断、重复依赖诊断和 Markdown 报告稳定输出。验证命令包括 moon info、moon fmt --check、moon test 和 moon run cmd/main。

开源计划

项目以 GitHub 仓库作为官方开源发布地址，满足赛事阶段一对有效 GitHub 仓库链接的要求；GitLink 仓库作为赛事平台镜像，用于同步提交和平台验收。后续迭代将优先保持核心 API 稳定、测试可复现、报告输出可比对。

核心 API 交付

parse_mod	解析模块清单中的名称、版本、许可证、README、仓库、关键词和依赖。
parse_pkg	解析包级 import 依赖与别名。
project_from	组合模块与包信息，构建统一项目模型。
evaluate_policy	依据策略生成元数据、许可证和依赖诊断。
render_markdown	输出稳定审计文本，便于提交、比对和 CI 集成。