

[PRD] Project: Mole

1. 프로젝트 개요 (Project Overview)

- 비전: 개발자가 브라우저로 화면을 전환하지 않고 터미널 내에서 ngrok의 모든 트래픽을 모니터링하고 제어할 수 있는 최적의 DX(Developer Experience) 제공.
- 핵심 가치: Context Switching 최소화, 키보드 중심 제어, 신속한 API 디버깅.
- 슬로건: "The missing TUI for ngrok."

2. 타겟 유저 (Target Audience)

- 로컬 개발 환경에서 외부 연동(Webhook, Mobile App, Frontend)을 위해 ngrok을 빈번하게 사용하는 개발자.
- GUI보다 CLI/TUI 환경을 선호하며, 생산성 도구(lazygit 등)에 익숙한 유저.

3. 사용자 스토리 (User Stories)

- 조회: "나는 터미널에서 현재 활성화된 ngrok 터널 목록과 상태를 한눈에 보고 싶다."
- 모니터링: "나는 실시간으로 들어오는 HTTP 요청의 메서드, 경로, 상태 코드를 확인하고 싶다."
- 상세 분석: "나는 특정 요청의 Header와 Body(JSON 등)를 터미널 안에서 포맷팅된 형태로 보고 싶다."
- 재현: "나는 디버깅을 위해 이전에 성공/실패했던 요청을 단축키 하나로 즉시 재전송(Replay)하고 싶다."

4. 기능 요구사항 (Functional Requirements)

4.1. MVP (Phase 1)

| 구분 | 기능 | 상세 설명 |
|--------------|-------------|---|
| Dashboard | 터널 상태 표시 | 활성화된 터널의 URL(Public/Local) 및 프로토콜 표시 |
| Request List | 실시간 로그 스트리밍 | ngrok 로컬 API를 폴링하여 요청 리스트를 최신순으로 나열 |
| Inspector | 상세 정보 뷰어 | 선택된 요청의 Request/Response Header 및 Body 출력 |

| | | |
|------------|----------------|--|
| Formatting | JSON 가독성 개선 | Body 데이터가 JSON일 경우 Pretty-print 및 Syntax Highlighting |
| Action | Replay 기능 | 특정 요청을 선택하고 <code>r</code> 키를 누르면 ngrok API를 통해 해당 요청 재전송 |

4.2. Advanced (Phase 2)

- Search/Filter: 특정 Path나 Status Code(예: 500번대만 보기)로 리스트 필터링.
- Copy as cURL: 특정 요청을 cURL 명령어로 즉시 복사.
- Multiple Providers: Cloudflare Tunnel(argo), Localtunnel 등으로 지원 범위 확대.
- Configuration: 사용자 정의 단축키 및 테마(Color) 설정.

5. 비기능 요구사항 (Non-Functional Requirements)

- 성능: 대량의 트래픽이 발생해도 UI가 프리징되지 않아야 함 (Asynchronous API Call).
- 사용성: Vim-like 단축키(`j`, `k`, `g`, `G`) 지원으로 학습 곡선 최소화.
- 배포: `brew`, `go install` 등을 통한 간편한 설치 지원.

6. UI/UX 컨셉 (TUI Layout)

TUI는 화면 공간 활용이 핵심입니다. `bubbletea`와 `lipgloss`를 활용한 표준 3분할 구조를 제안합니다.

- Left Sidebar (Tunnels): 현재 실행 중인 터널 목록.
- Top Main (Logs): HTTP 요청 이력 (Method, Path, Status, Latency).
- Bottom/Right (Detail): 선택된 요청의 상세 정보 (Header/Body).
- Footer: 현재 사용 가능한 단축키 가이드 (e.g., `r`: replay, `c`: copy, `q`: quit).

7. 기술 스택 (Tech Stack)

- Language: Go 1.21+
- TUI Framework: [Charmbracelet Bubbletea](#) (The Elm Architecture)
- Styling: [Lipgloss](#)
- Components: [Bubbles](#) (List, Viewport, Spinner)
- API Client: Standard `net/http` (ngrok Local API 연동)

8. 성공 지표 (Success Metrics)

- GitHub Stars 및 Forks 수.
- 패키지 매니저(Homebrew 등)를 통한 다운로드 수.
- ngrok 외의 다른 터널링 도구에 대한 커뮤니티의 추가 지원 Issue 발생 여부.

