

Maintenance Planning

Local User Login

The maintenance plan for the local user login module focuses on improving code maintainability and security by reducing duplicated authentication logic. During implementation, password verification logic was found to be repeated across multiple endpoints, increasing the risk of inconsistency and making future changes harder to manage. Refactoring is planned to centralize this logic and improve testability while keeping the functionality unchanged to support the MVP.

Maintenance & Refactoring

- **Smell:** Duplicate password verification logic across endpoints.
- **Fix:** Extract PasswordService; reuse; add unit tests.

Self-Service Password Reset

The self-service password reset feature requires maintenance to address duplicated token handling logic and improve reliability. Token creation, validation, and expiration checks are currently implemented in multiple places, which increases maintenance cost and the risk of security bugs. The maintenance approach aims to centralize token management and standardize the reset workflow without expanding the feature scope beyond the MVP.

Maintenance & Refactoring Strategy

- **Smell:** Token handling duplicated in reset and verify logic.
- **Fix:** Centralize in ResetTokenService; add unit tests.

Logging & Observability

Logging for authentication and password-related actions is minimal and inconsistent, limiting visibility when issues occur in production. Without structured logs, diagnosing failures or security incidents becomes difficult. Maintenance will improve observability without affecting user-facing features.

Maintenance & Refactoring Strategy

- **Smell:** Inconsistent and insufficient logging for authentication and reset workflows.
- **Fix:** Introduce structured logging with standardized log levels and messages for critical authentication events.

Test Coverage Improvement

Although core authentication flows are implemented, test coverage is uneven, particularly around edge cases and failure scenarios. This increases the risk of regressions during future refactoring. Maintenance planning prioritizes strengthening automated tests to support long-term stability.

Maintenance & Refactoring Strategy

- **Smell:** Limited test coverage for edge cases and failure paths in authentication features.
- **Fix:** Add unit and integration tests for invalid credentials, expired tokens, and boundary conditions.