

PHP Team Meeting Agenda – 1 Month Plan

Week 1 — Modern PHP Features (PHP 8.2/8.3)

Goal: Sharpen understanding of new PHP features and when to use them.

Mini-talk Topics:

- readonly classes & properties
- enum improvements
- Intersection & disjunctive normal form types
- true type, json_validate(), mb_str_pad() (PHP 8.3)

Exercise:

Take an existing DTO or config class and refactor it to use a readonly class.

Discussion Prompts:

- Which features should we adopt immediately?
- Do we update coding standards to require them?

Week 2 — Testing & Code Quality

Goal: Strengthen confidence in automated testing.

Mini-talk Topics:

- PHPUnit vs Pest
- Writing feature tests in Laravel/Symfony
- Static analysis with PHPStan/Psalm

Exercise:

Write a test for a function that calculates invoice totals. Then run phpstan analyse on the same code.

Discussion Prompts:

- Should we enforce minimum coverage or static analysis levels in CI?
- Share one recent bug that tests/static analysis could've caught earlier.

Week 3 — Architecture & Scalability

Goal: Explore patterns for maintainable, scalable PHP apps.

Mini-talk Topics:

- Repository pattern: pros/cons with modern ORMs
- Event-driven architecture with Laravel Events/Observers
- When to introduce queues (Redis/Horizon)

Exercise:

Refactor a simple service to dispatch an event (UserRegistered) and handle it asynchronously with a queued listener.

Discussion Prompts:

- Where could we apply events in our current projects?
- Are we overusing repositories or could we simplify?

Week 4 — Security & Secure Practices

Goal: Refresh secure coding habits.

Mini-talk Topics:

- Top 5 PHP security pitfalls: SQL injection, XSS, CSRF, insecure file uploads, weak password storage
- Importance of prepared statements & Laravel's Query Builder

Exercise:

Show two snippets: one vulnerable login query using string interpolation, one secure using prepared statements. Team identifies the risk.

Discussion Prompts:

- Which security rules should be in our code review checklist?
- Do we need extra tooling (e.g., security linters, dependency scanners)?