

Domain-Driven Design in PHP

Full day workshop with Mathias Verraes



Software design can have a very **slow feedback loop**: by the time you find out if the architectural and design decisions you made were the right ones, it may be too late for drastic changes. As developers, we tend to phrase all problems as technical problems, which can be sufficiently challenging in their own right. We tend to **lose sight of the underlying business concerns**.

Domain-Driven Design helps us to tie our domain model to the business. Deep models have great value: they help us build systems that are more relevant. They communicate the meaning to the developers and the stakeholders. DDD can be quite opaque though. The patterns and practices often appear to be nothing but theory. In this session, Mathias will complement the theory with a practical approach. You'll learn how to discover and evolve your domain model, and how design patterns can help you translate it to code.

Learn all about

- Domain-Driven Design basics
- Discovering the domain and the Ubiquitous Language
- Structural and behavioral modelling
- Applying tactical design patterns: value objects, entities, aggregates, repositories, commands, events, specifications, strategies, factories, services...
- And more!

Attendees should have a good working knowledge of PHP, Object-Oriented Programming, and basic SOLID principles.

About the instructor

Mathias Verraes is a programmer, consultant, blogger, speaker, and trainer. He advises companies on building enterprise applications for complex business domains, and on dealing with legacy systems. He has advised for clients in Automotive, Logistics, Finance, E-Commerce, Web, and more. He's the organizer of the Domain-Driven Design Belgium community. When he's not working, he's at home in Kortrijk, Belgium, helping his two sons build crazy Lego train tracks.

<http://verraes.net>