Anti patterns

* Golden Hamer  
  try to solve all problems with one trick
* excessive code  
  implementing not needed functionality
* String obsession

try to fit everything in a String

* art of complexity  
  >< KISS (keep it simple, stupid)
* test allergy  
  >< Software testing is needed to verify that your new feature works as intended and doesn’t break some other (forgotten) feature.
* bad naming

‘piep’  
>< naming conventions & ...

* double data, double code
* being too chatty with the database  
  n + 1 problem (JPA course)
* fragile solutions  
  eg. a new requirement makes that you have to start all over  
  >< design for change
* too long methods
* communication problems  
  in a team, between team and stakeholders, users, ...
* …