

# Scrum + BDD + DDD

[@HelderDOliveira](#)

# Who am I?

[@HelderDOliveira](#)

Father, husband, brother, son,  
Venezuelan, Portuguese, Informatic  
Engineer, CSM, CSPO, CSD, CSP,  
Java Certified, Scrum Master, Software  
Architect, QA, aspiring Software  
Craftsman, wannabe guitar player,  
painter, anarchist, adustriano, ucabista,  
magallanero, berniquista, colchonero,  
Golf fan, gamer, friki, et. al.


DEVELOPER







SM

A person is seen from behind, walking up a rocky mountain trail. They are carrying a large, heavy load on their back, which includes a woven basket, several red gas cylinders, and other equipment. The person is wearing a blue jacket and red pants with a green stripe. The background features steep, snow-covered mountain peaks under a blue sky with some clouds.

Scrum Master, they say...  
Servile Leader, they say...

...

...

(cabrones!)



**ENTER  
AT  
YOUR  
OWN  
RISK**



# Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

# Agile Manifesto

We are uncovering better ways of **developing software** by doing it and helping others do it.

<http://www.agilemanifesto.org/>



# Agile Manifesto

Welcome **changing requirements, even late in development**. Agile processes harness change for the customer's competitive advantage.

<http://www.agilemanifesto.org/principles.html>

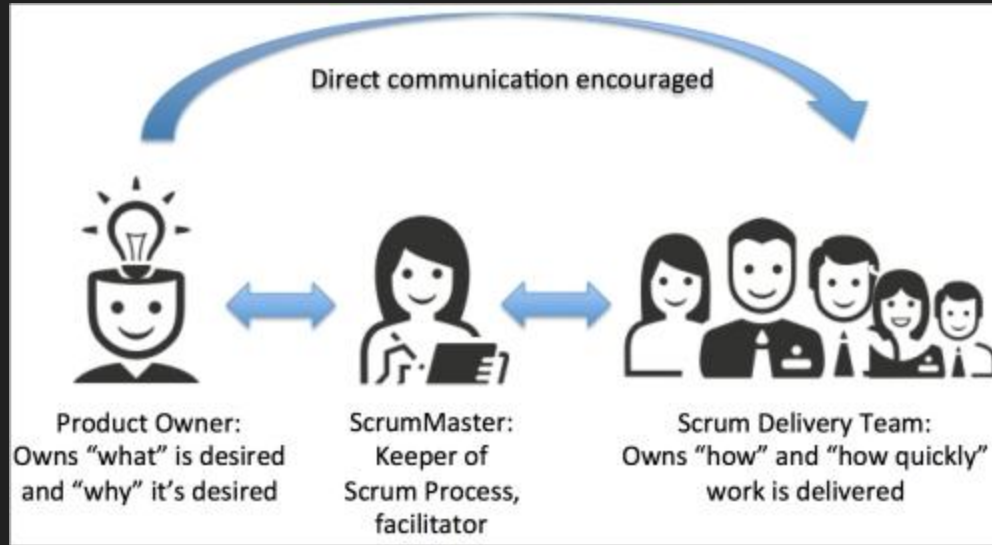
# Scrum

## Definition of Scrum

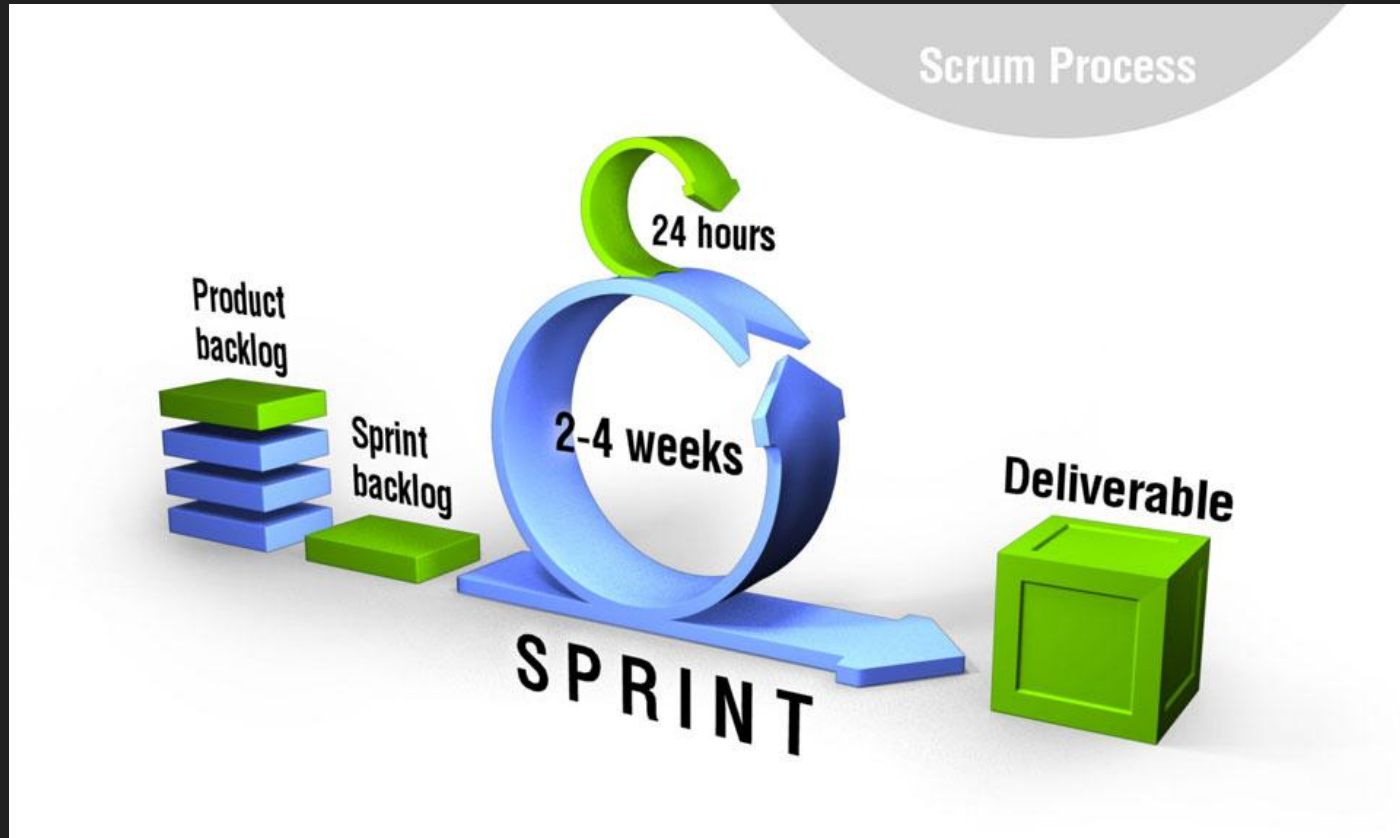
Scrum (n): A **framework** within which people can address **complex** adaptive problems, while productively and creatively delivering products of the **highest possible value**.

<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf>

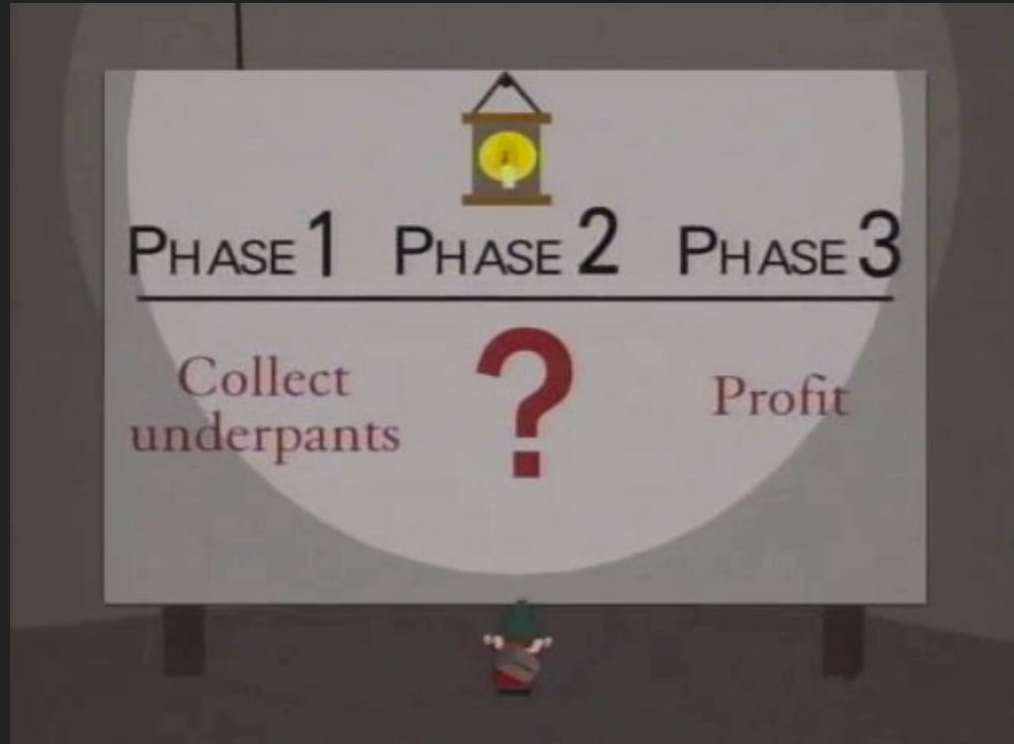
# Scrum



# Scrum



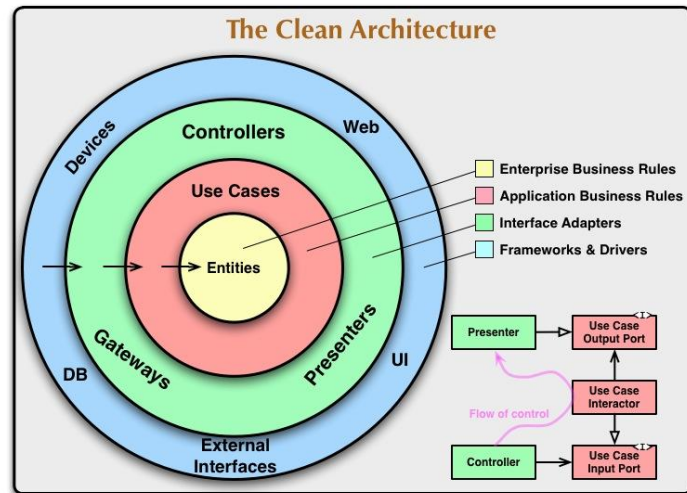
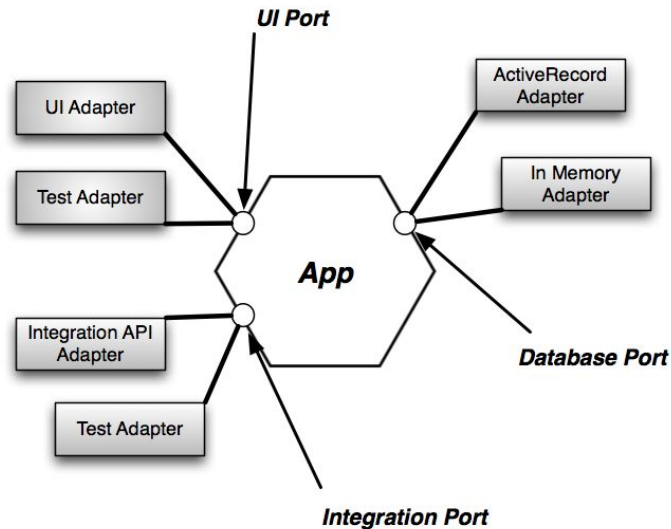
# Where is the 'how'?





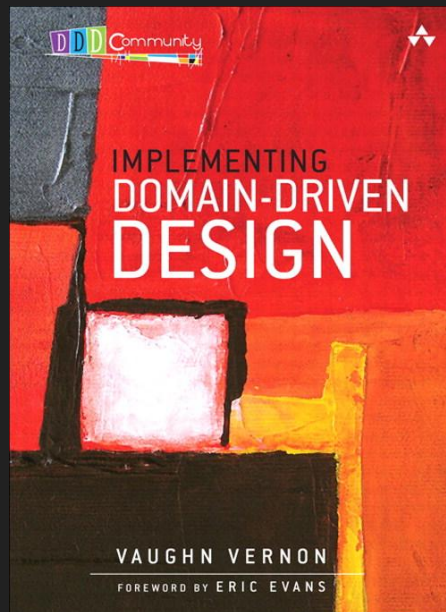
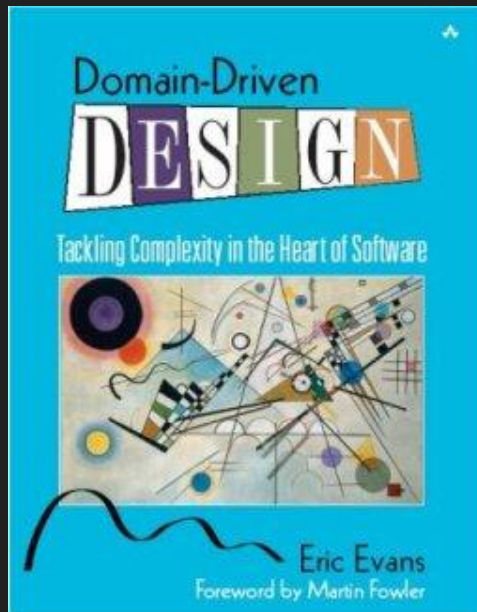
# Alternatives

- Hexagonal architecture
- Clean Architecture
- Domain-Driven Design
- et. al.



# Domain-Driven Design (DDD)

Is an approach to developing software for complex needs by deeply connecting the implementation to an **evolving model of the core business concepts**.



# DDD

- Place the project's primary focus on the **core domain and domain logic**
- Base **complex designs** on a model
- Initiate a creative **collaboration between technical and domain experts** to iteratively cut ever closer to the conceptual heart of the problem.

# DDD

Domain model:

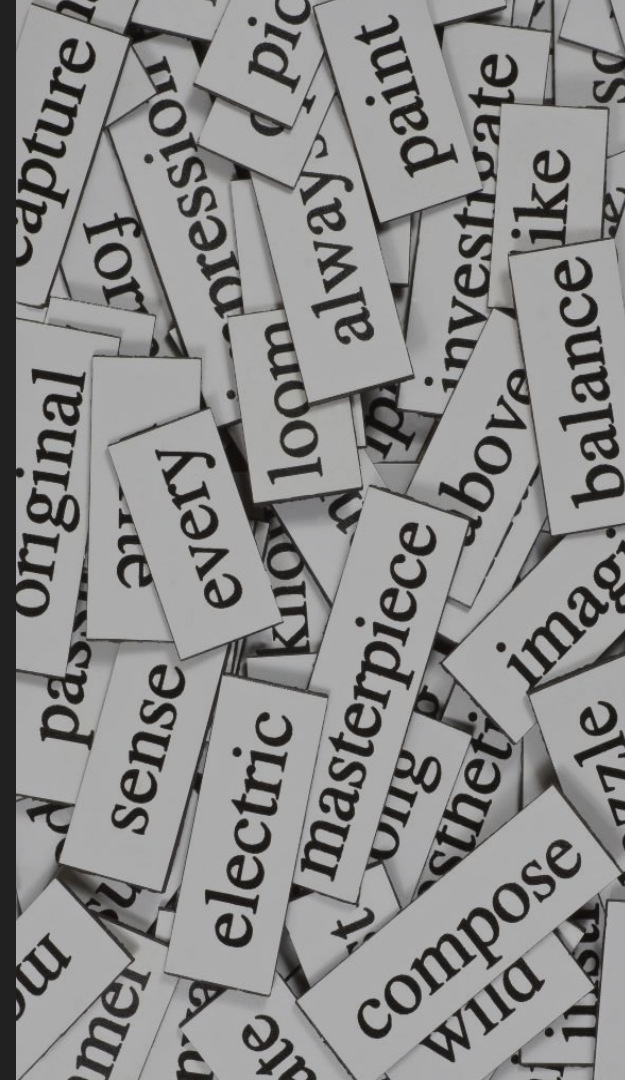
- A software model based specifically in the **domain of the business** you are working with
- It's **usually build as an object model**, where objects have both data and behaviour with accurate **business meaning**
- Since we are going to create a software solution for a problem in the domain, we need a form of **simplification of that domain** to base the solution on.

# DDD

# Ubiquitous Language

The practice of building up a common, **rigorous language between developers and users**. This language should be based on the Domain Model used in the software - hence the need for it to be rigorous, since software doesn't cope well with ambiguity.

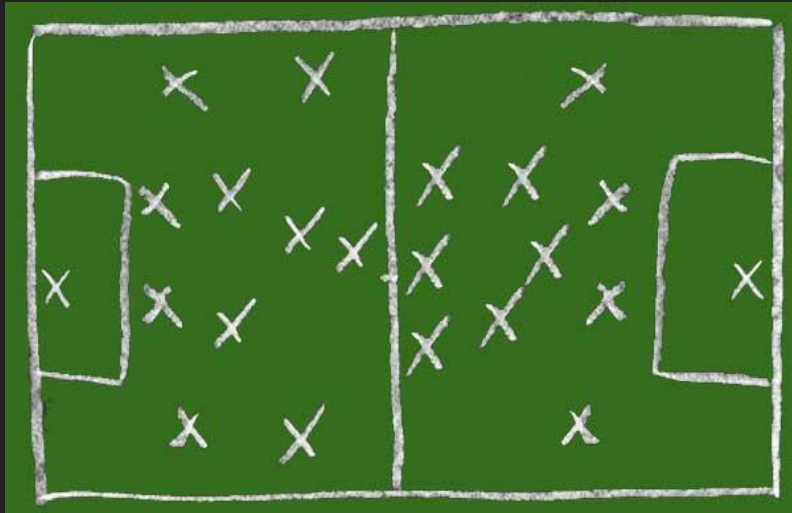
<http://martinfowler.com/bliki/UbiquitousLanguage.html>



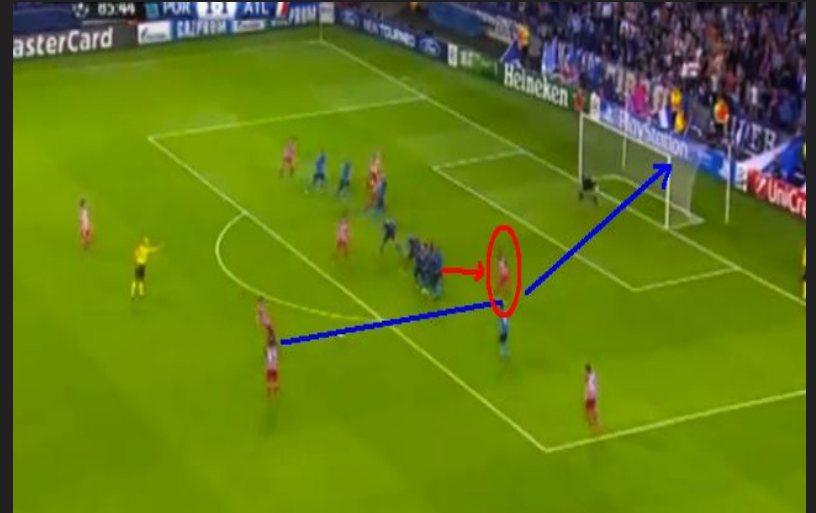


# DDD

Strategic model



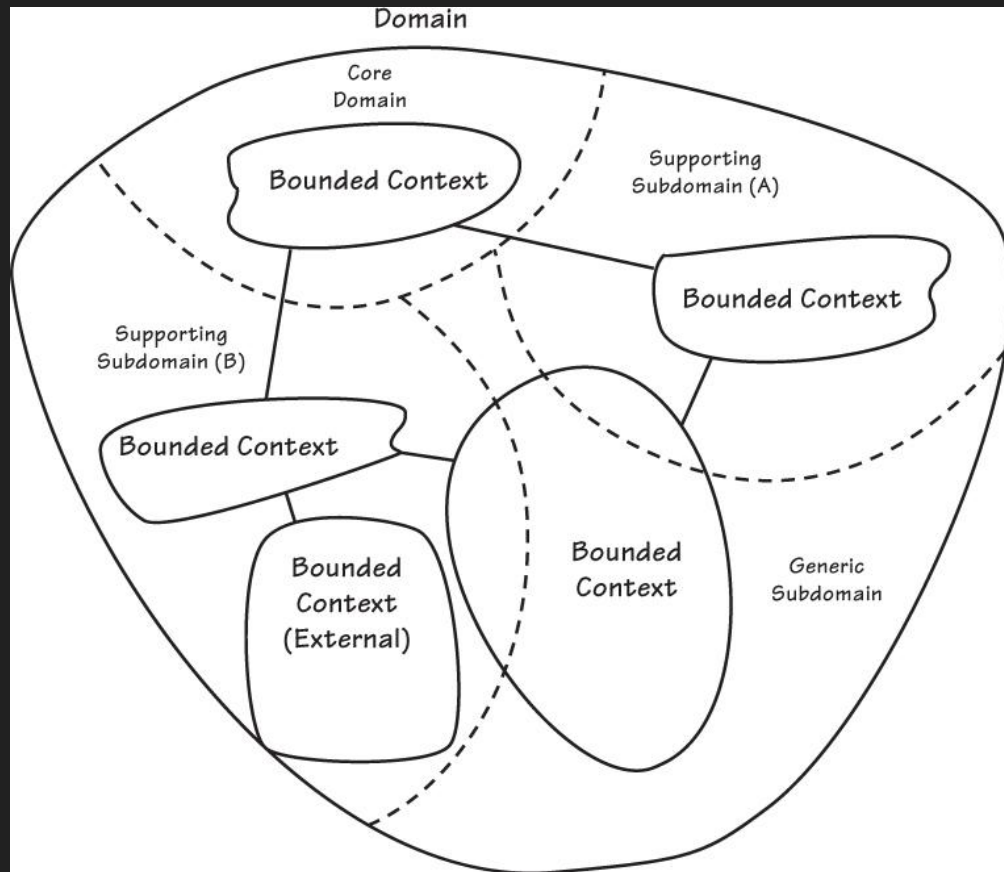
Tactic model



# DDD

Strategic model:

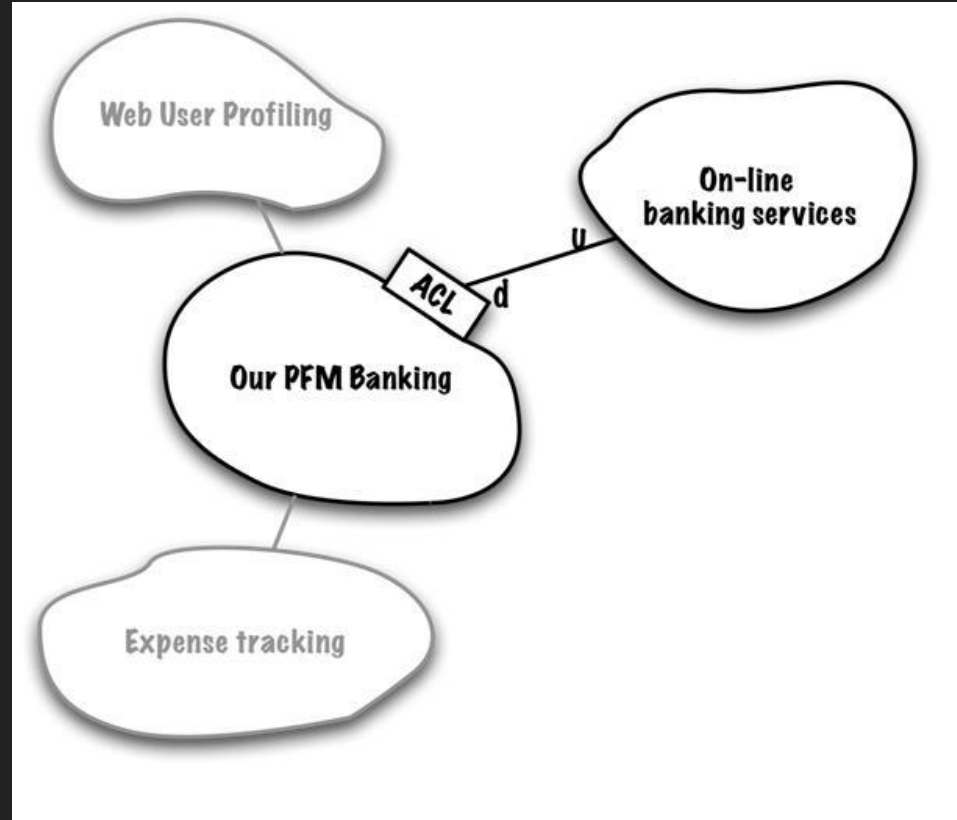
- Domain
- Subdomain
- Core domain
- Context



# DDD

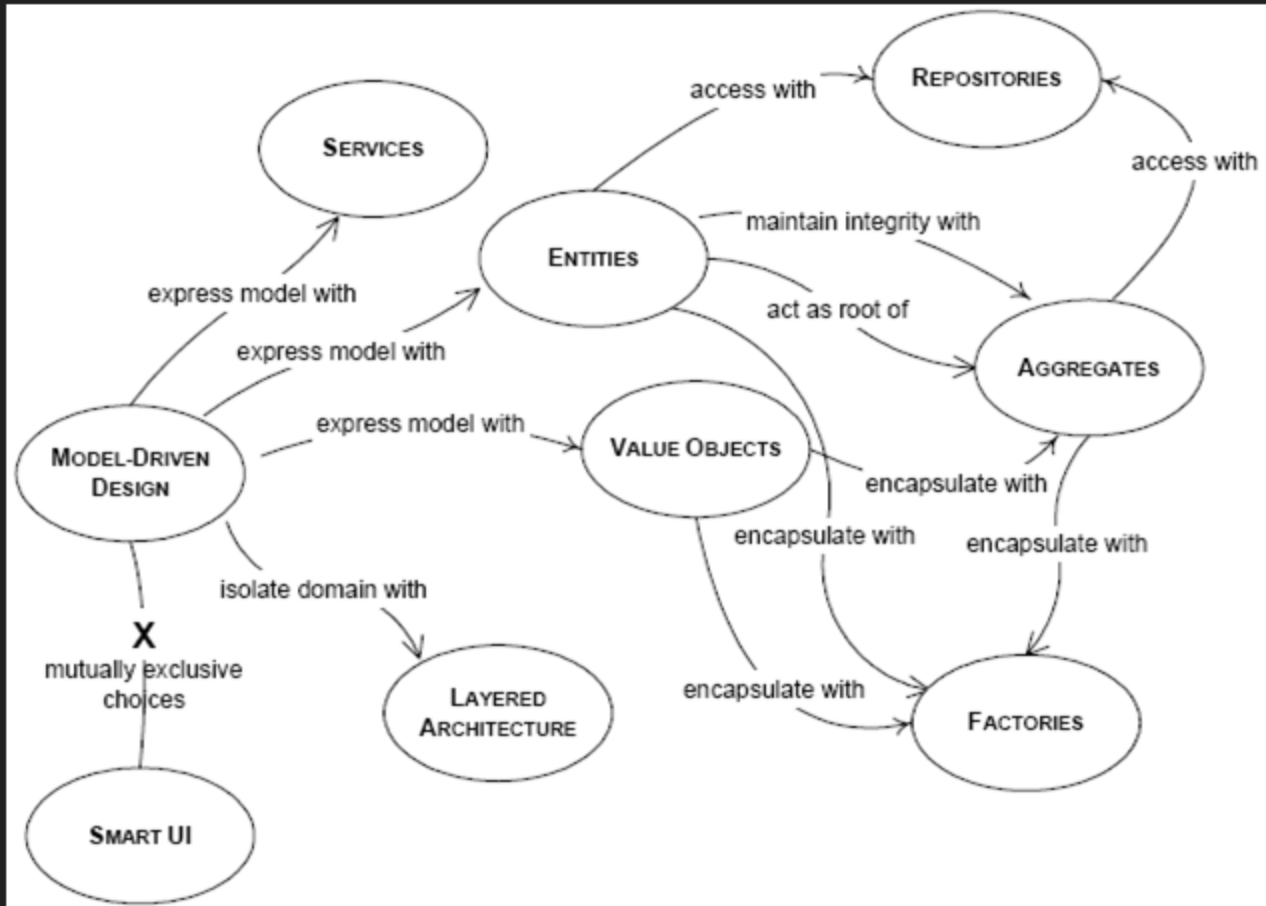
Strategic model patterns:

- anticorruption layer
- open host service
- published language



# DDD

## Tactic model



# Scrum + DDD

- DDD and Scrum needs a good involvement of product owners to make it a success.
- DDD and Scrum works with stories:
  - DDD uses stories to create and verify the model.
  - Scrum uses stories to determine the scope for each sprint, write tests and running software.
- About tests:
  - Scrum uses acceptance criteria to verify if the story is implemented
  - DDD uses tests to create a good API.



# Scrum + DDD

- DDD and Scrum needs a good involvement of product owners to make it a success.
- DDD and Scrum works with stories:
  - DDD uses stories to create and verify the model.
  - Scrum uses stories to determine the scope for each sprint, write tests and running software.
- About tests:
  - Scrum uses acceptance criteria to verify if the story is implemented
  - DDD uses tests to create a good API.

Where is the glue?



# Alternatives

- ATDD:  
Acceptance Test Driven Development
- TDD:  
Test Driven Development  
(outside in / classicist)
- BDD:  
Behaviour-Driven Development



# Behavior-Driven Development (BDD)

A software development process that **emerged from TDD**.

BDD provide software development and management teams with shared tools and a shared process to **collaborate on software development**.



# Corey Haines - BDD in 5 minutes



<https://vimeo.com/23061155>



# BDD

- Slice into problem domains
- Each domain has its own stakeholders
- Each domain has its own language
- Test the behaviors of each domain
- Isolate tests using test doubles
- Stub other domains and external systems



# BDD

**Story:** Returns go to stock

**In order to** keep track of stock

**As a** store owner

**I want to** add items back to stock when they're returned

**Scenario 1:** Refunded items should be returned to stock

**Given** a customer previously bought a black sweater from me

**And** I currently have three black sweaters left in stock

**When** he returns the sweater for a refund

**Then** I should have four black sweaters in stock

**Scenario 2:** Replaced items should be returned to stock

**Given** that a customer buys a blue garment

**And** I have two blue garments in stock

**And** three black garments in stock.

**When** he returns the garment for a replacement in black,

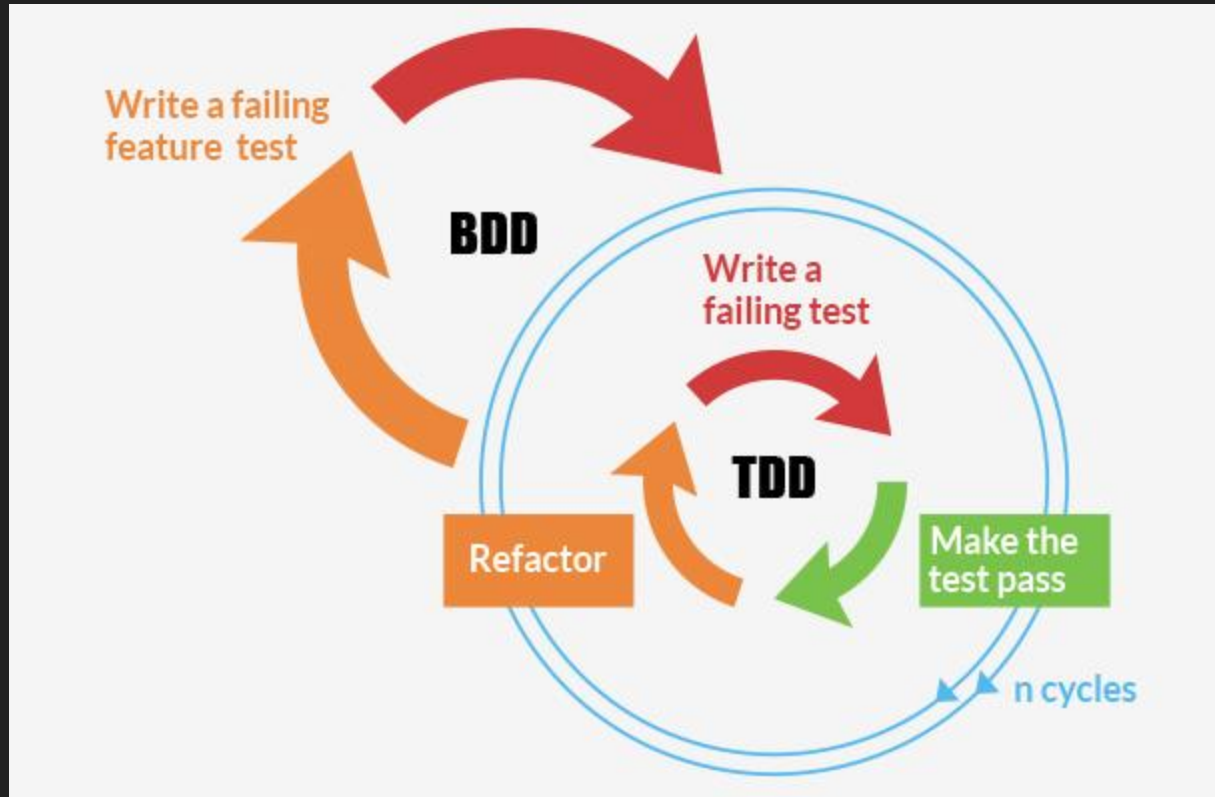
**Then** I should have three blue garments in stock

**And** two black garments in stock

## Gherkin



# BDD



# Example: Recruitment Process





# Recruitment Process



## RECRUITMENT PROCESS OUTSOURCING (RPO)

SOURCE

RECRUIT

HIRE

ON-BOARD

CUSTOMER  
NEEDS

FIND  
CANDIDATES

QUALIFY CANDIDATES

HIRE  
EMPLOYEES

ON-BOARD  
EMPLOYEES

SMW  
ACTIVITIES  
DURING  
EACH STEP

- Requisition Management
- Career Portal
- Applicant Intake
- Resume Intake
- Internet Sourcing

- Pre-Screening
- Screening
- Assessments and Testing
- Interviewing & Interview Coordination
- Employment Branding

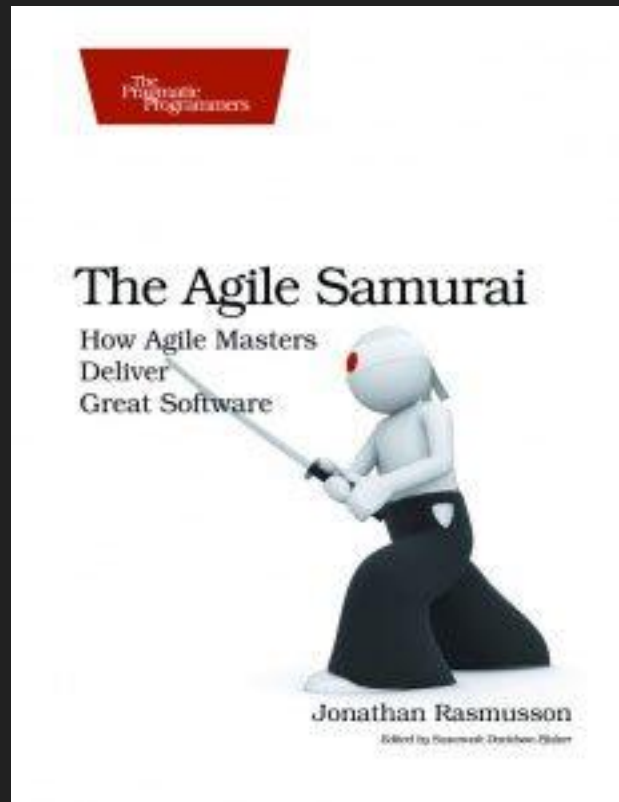
- Final Interview Management
- Offer Management
- Reference Checks
- Pre-Hire Assessments and Testing

- Background Checks & Drug Screens
- I-9 Verification
- WOTC Management
- Record/ Paperwork Management
- Third-Party Vendor Management

# Inception

## 1. Why are we here?

We need a recruitment system in order to track respectfully all the events with our candidates

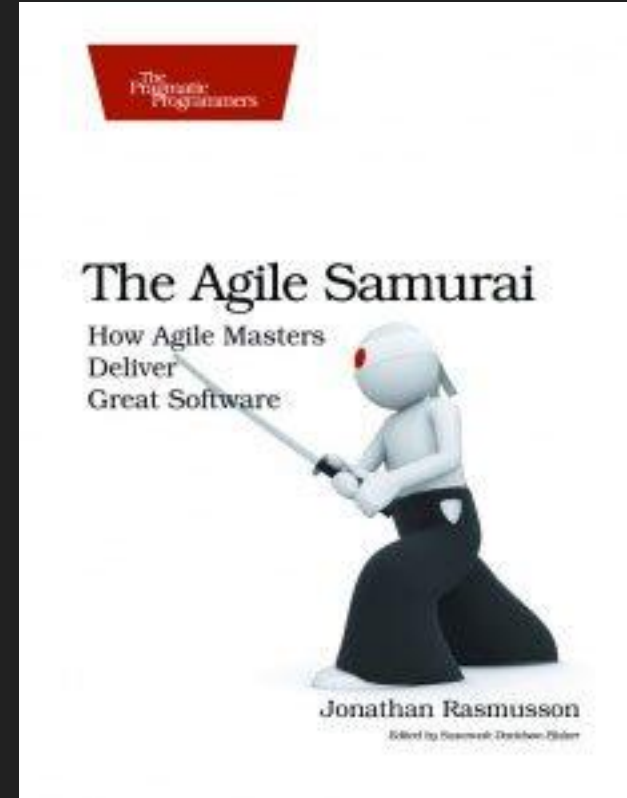


# Inception

## 2. Elevator pitch

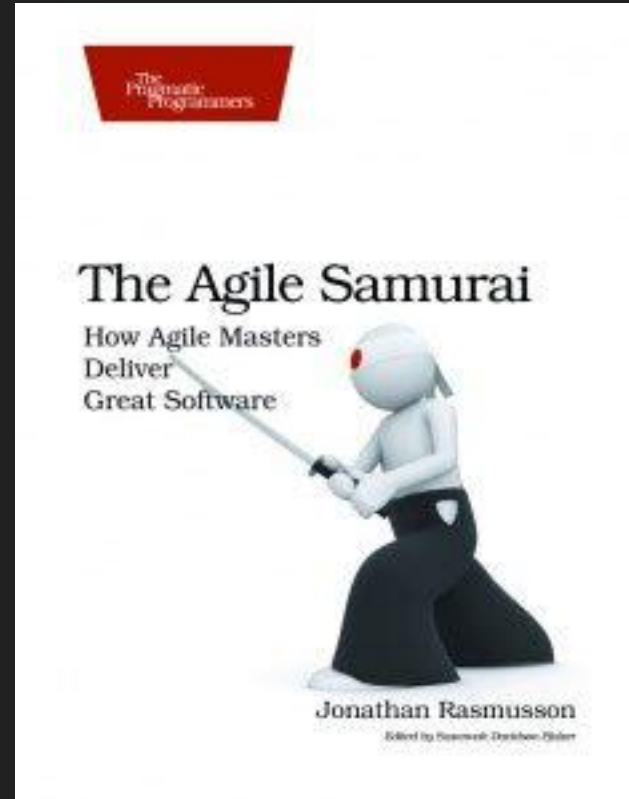
For recruiters  
who interact with many candidates  
the recruiting system (RS)  
is a communication tool  
that informs the recruiters the status of any  
candidate.

Unlike the current paper process  
our RS is web based so can be accessed  
anywhere anytime



# Inception

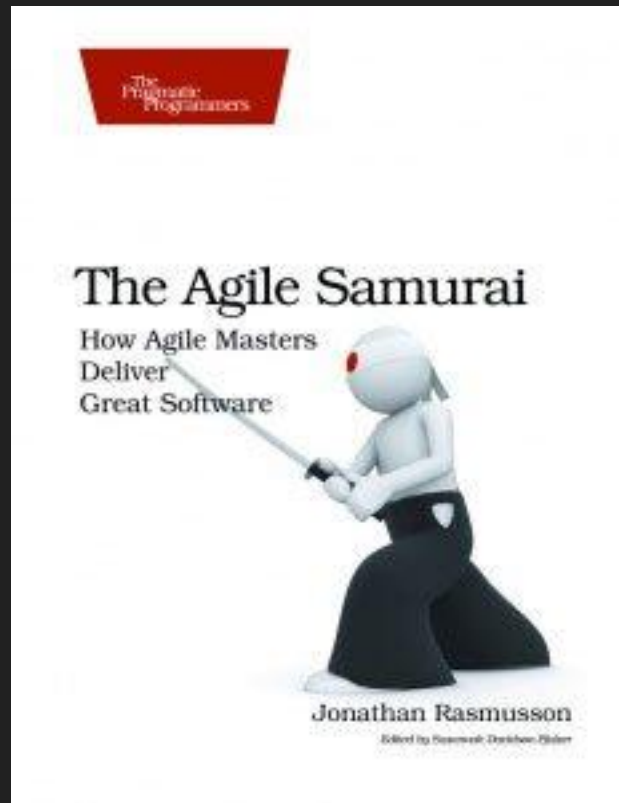
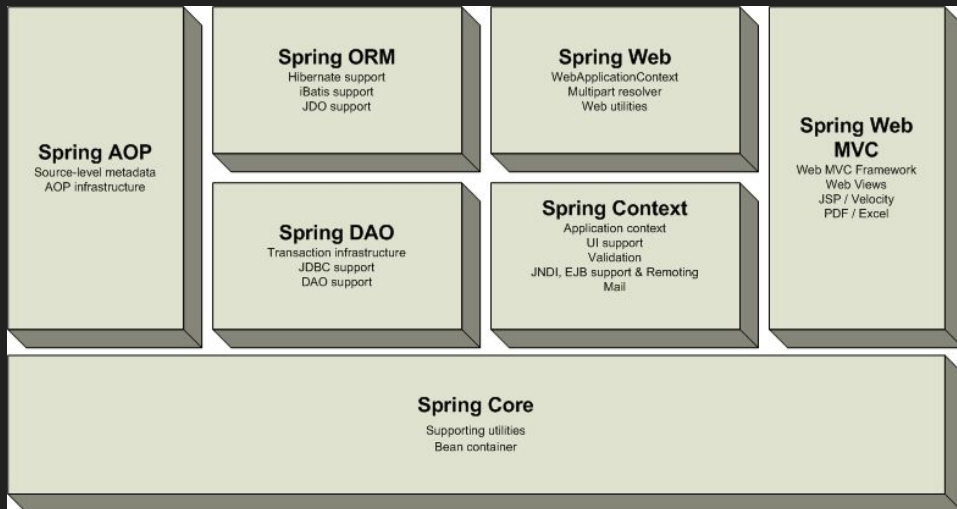
... and so on, until ...





# Inception

## 6. Solution



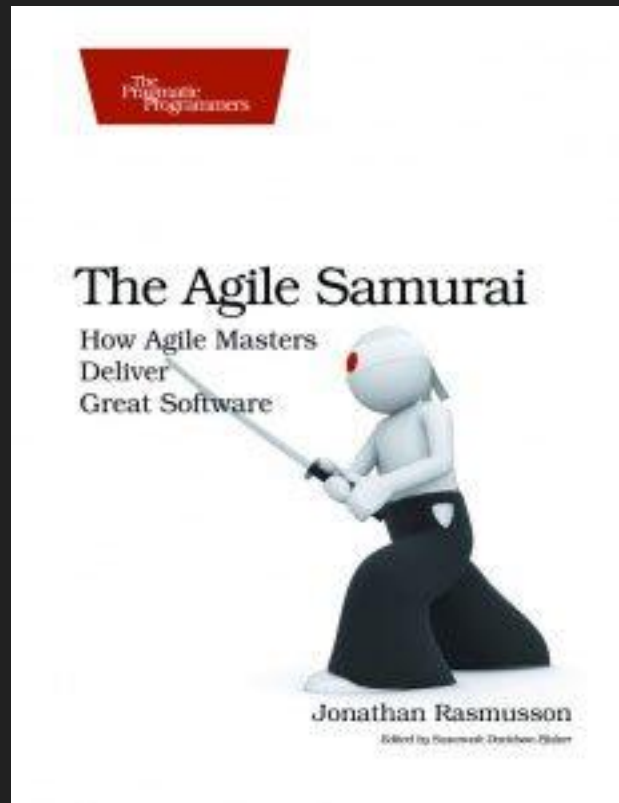
# Inception

## 6. Solution



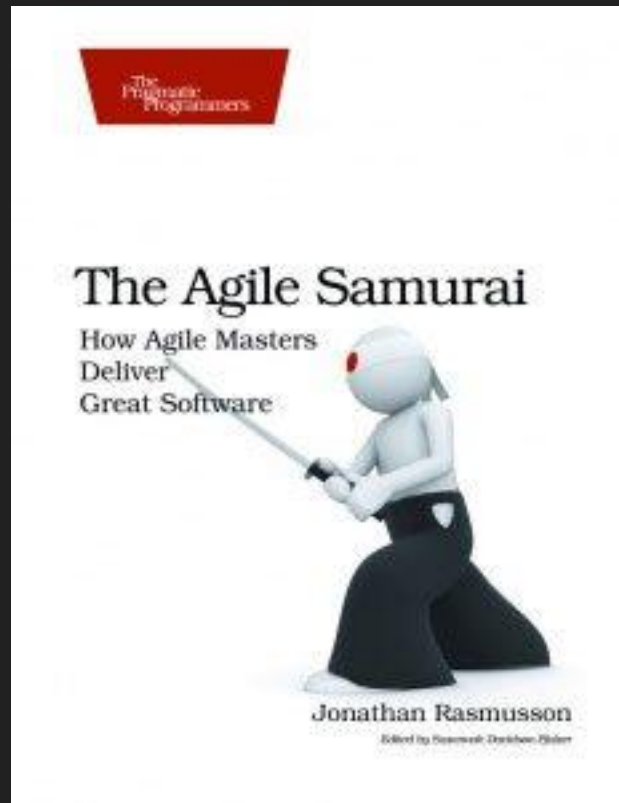
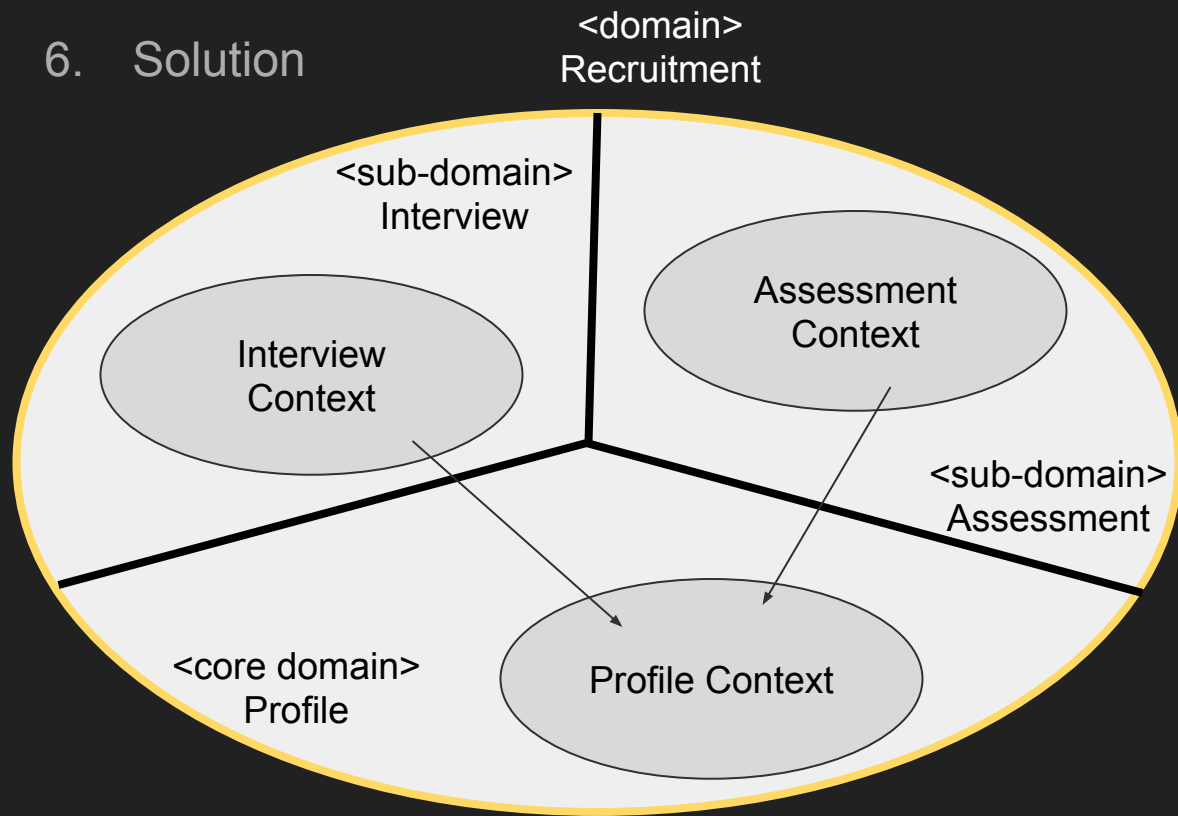
Please, watch this:  
Ruby Midwest 2011 - Keynote: Architecture the Lost  
Years by Robert Martin

<https://www.youtube.com/watch?v=WpkDN78P884>

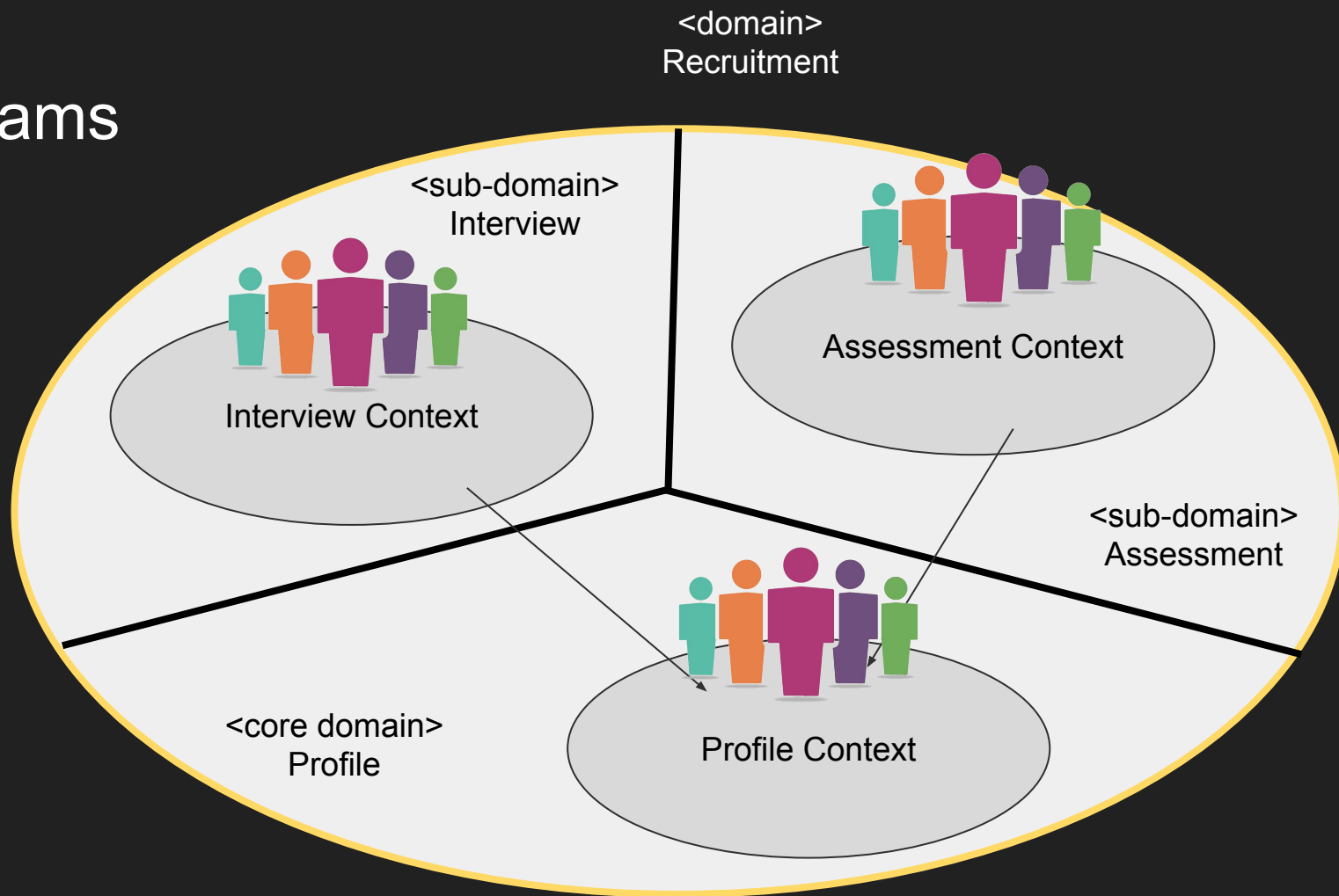


# Inception

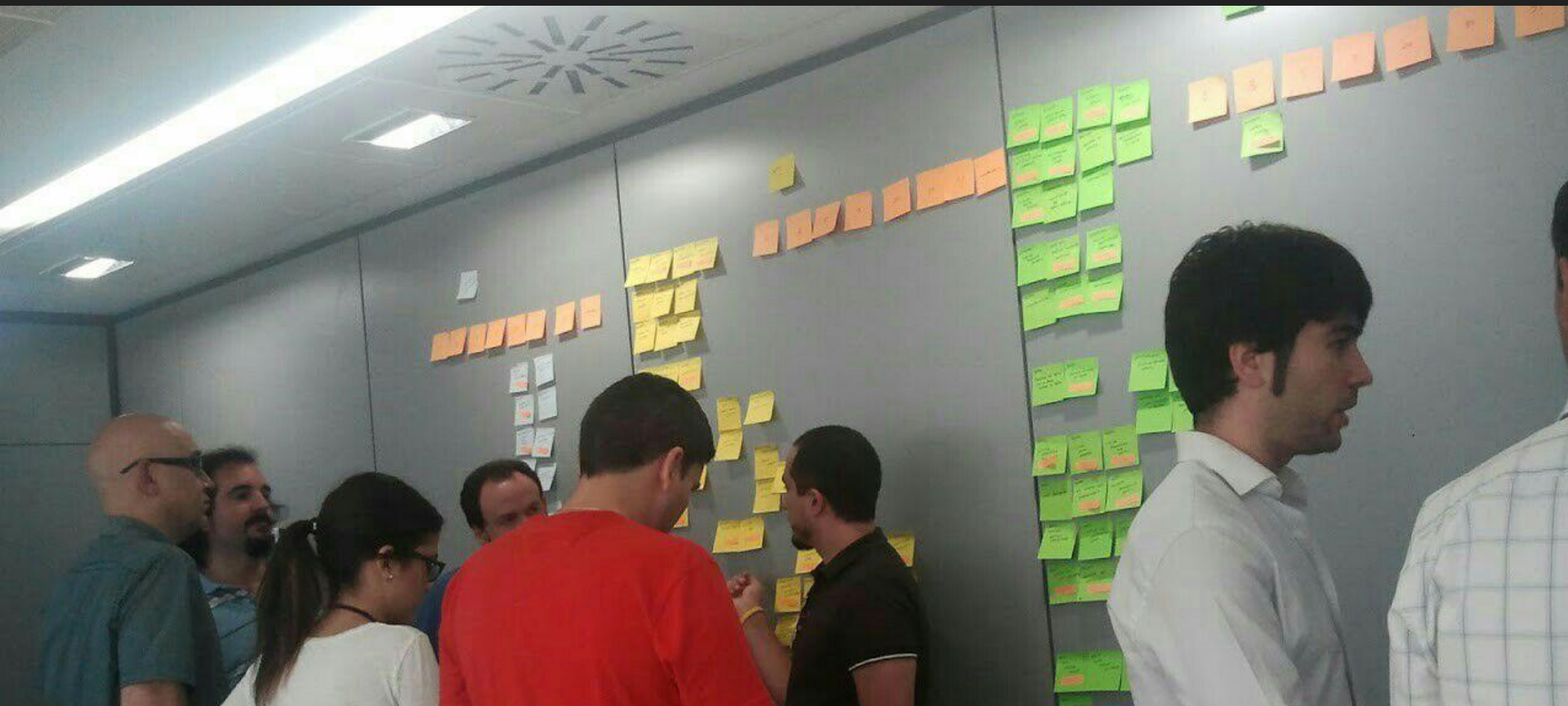
## 6. Solution



# Teams



Wall time!



# Profile context: user stories

As a recruiter

I want to create a new candidate profile

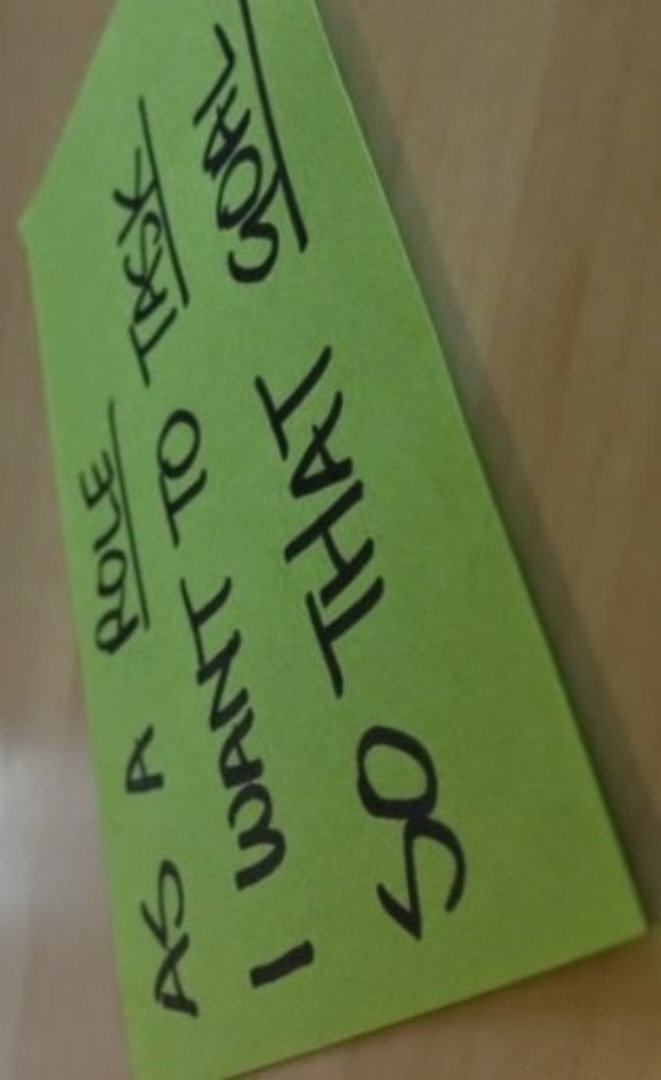
In order to track his recruitment process events

As a recruiter

I want to add a position to a candidate

In order to list his past jobs

...



# Profile context: acceptance test

As a recruiter

I want to create a new candidate profile

In order to track his recruitment process events

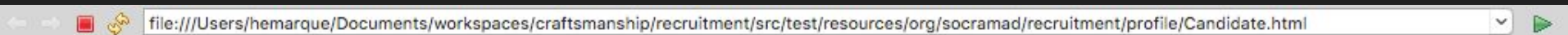
**Given** a candidate's name "Helder De Oliveira"  
and candidate's email "hdo@hdo.pt"  
and candidate's phone number "00351 234987654"

**When** I request to add the candidate

**Then** the candidate should be stored



# Profile context: acceptance test (HTML format)



## Profile context: Candidate

### User Story: create a new candidate profile

As a recruiter

I want to create a new candidate profile

In order to track his recruitment process events

### Scenario: request to add the candidate

Given a candidate's name "Helder De Oliveira"

and candidate's email "hdo@hdo.pt"

and candidate's phone number "00351 234987654"

When I request to add the candidate

Then the candidate should be stored

### User Story: create a new candidate profile

As a recruiter

I want to add a position to a candidate

In order to list his past jobs



# Profile context: acceptance test (HTML format)

```
<!DOCTYPE html>
<html xmlns:concordion="http://www.concordion.org/2007/concordion">
<head>
<meta charset="UTF-8"/>
<title>Profile context: Candidate</title>
</head>
<body>
<h1>Profile context: Candidate</h1>
<h2>User Story: create a new candidate profile</h2>
  <p>
    As a recruiter<br/>
    I want to create a new candidate profile<br/>
    In order to track his recruitment process events<br/>
  </p>
<h3>Scenario: request to add the candidate</h3>
  <p>
    Given a candidate's name "<span concordion:set="#name">Helder De Oliveira</span>"<br/>
    and candidate's email "<span concordion:set="#email">hdo@hdo.pt</span>"<br/>
    and candidate's phone number "<span concordion:set="#phone">00351 234987654</span>"<br/>
    When I request to add the candidate
    <span concordion:execute="#result = addCandidate(#name, #email, #phone)"></span><br/>
    Then the candidate should be
    <span concordion:assertTrue="#result">stored</span><br/>
  </p>
<h2>User Story: create a new candidate profile</h2>
  <p>
    As a recruiter<br/>
    I want to add a position to a candidate<br/>
    In order to list his past jobs<br/>
  </p>
```

# Profile context: fixture

Test / Fixture

```
package org.socramad.recruitment.profile;

import org.concordion.integration.junit4.ConcordionRunner;

@RunWith(ConcordionRunner.class)
public class CandidateTest {
    public boolean addCandidate(String name, String email, String phone){
        return false;
    }
}
```

# Profile context: acceptance test failing

file:///Users/hemarque/Documents/workspaces/craftsmanship/recruitment/target/concordion/org/socramad/recruitment/profile/Candidate.html

## User Story: create a new candidate profile

As a recruiter

I want to create a new candidate profile

In order to track his recruitment process events

## Scenario: request to add the candidate

Given a candidate's name "Helder De Oliveira"

and candidate's email "hdo@hdo.pt"

and candidate's phone number "00351 234987654"

When I request to add the candidate

Then the candidate should be `stored == false`

## User Story: create a new candidate profile

As a recruiter

I want to add a position to a candidate

In order to list his past jobs

# Profile context: fixture

```
package org.socramad.recruitment.profile;

import org.concordion.integration.junit4.ConcordionRunner;

@RunWith(ConcordionRunner.class)
public class CandidateTest {
    public boolean addCandidate(String name, String email, String phone){
        Candidate candidate = new Candidate(name, email, phone);
        CandidateApplicationService candidateApplicationService =
            ApplicationRegistry.candidateApplicationService();
        return candidateApplicationService.addCandidate(candidate);
    }
}
```

Test / Fixture

Registry

Application  
Service

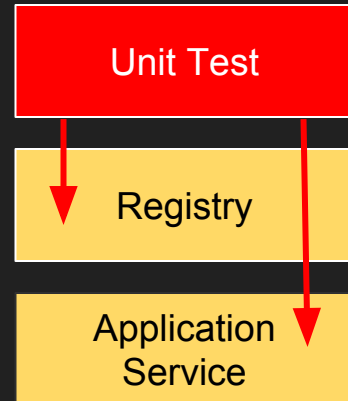
# Profile context: application service unit test failing

```
package org.socramad.recruitment.profile.application;

import static org.junit.Assert.*;

public class ApplicationServiceTest {

    @Test
    public void addCandidate() {
        Candidate candidate = new Candidate("Helder", "hdo@hdo.com", "123");
        CandidateApplicationService service =
            ApplicationRegistry.candidateApplicationService();
        assertTrue("Candidate not added", service.addCandidate(candidate));
    }
}
```



Console JUnit Markers Progress

ApplicationServiceTest

Runs: 1/1 Errors: 0 Failures: 1

org.socramad.recruitment.profile.application.ApplicationServiceTest [Runner: JUnit 4]

addCandidate (0.033 s)

Failure Trace

java.lang.AssertionError: Candidate not stored

at org.socramad.recruitment.profile.application.ApplicationService



# Profile context: repository access

```
import org.socramad.recruitment.profile.domain.model.Candidate;
import org.socramad.recruitment.profile.domain.model.CandidateRepository;

public class CandidateApplicationService {
    private CandidateRepository candidateRepository;

    void candidateRepository(CandidateRepository candidateRepository){
        this.candidateRepository = candidateRepository;
    }

    public boolean addCandidate(Candidate candidate) {
        return candidateRepository.add(candidate);
    }
}
```

Console JUnit Markers Progress

Finished after 0.025 seconds

Runs: 1/1 Errors: 1 Failures: 0

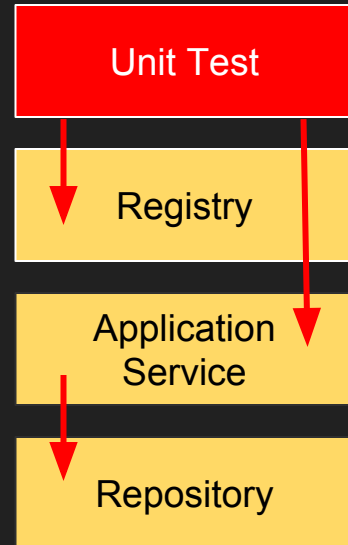
org.socramad.recruitment.profile.application.ApplicationServiceTest [Runner: JUnit 4]

addCandidate (0.000 s)

Failure Trace

sun.reflect.generics.reflectiveObjects.NotImplementedException

at org.socramad.recruitment.profile.domain.model.CandidateRepository.add(CandidateRepository.java:10)



# Profile context: repository unit test failing

```
package org.socramad.recruitment.profile.domain.model;

import static org.junit.Assert.*;

public class CandidateRepositoryTest {

    @Test
    public void addCandidate() {
        Candidate candidate = new Candidate("Helder", "hdo@hdo.com", "123");
        CandidateRepository repository = new CandidateRepository();
        assertTrue("Candidate not added", repository.add(candidate));
    }
}
```

Console JUnit Markers Progress

Finished after 0.022 seconds

Runs: 1/1 Errors: 0 Failures: 1

org.socramad.recruitment.profile.domain.model.CandidateRepositoryTest [Runner: JUnit4] addCandidate (0.000 s)

Failure Trace

```
java.lang.AssertionError: Candidate not added
    at org.socramad.recruitment.profile.domain.model.CandidateRepositoryTest.addCandidate(CandidateRepositoryTest.java:21)
```

Unit Test

Registry

Application  
Service

Repository

# Profile context: repository developed and test ok

```
package org.socramad.recruitment.profile.domain.model;

import java.util.ArrayList;
import java.util.List;

public class CandidateRepository {
    List<Candidate> candidates = new ArrayList<Candidate>();
    public boolean add(Candidate candidate) {
        return candidates.add(candidate);
    }
}
```

Console JUnit Markers Progress

Finished after 0.016 seconds

Runs: 1/1

Errors: 0

Failures: 0

org.socramad.recruitment.profile.domain.model.CandidateRepositoryTest [Runner: JUnit] Failure Trace

Unit Test

Registry

Application  
Service

Repository



# Profile context: application service and test ok

```
package org.socramad.recruitment.profile.application;

import static org.junit.Assert.*;

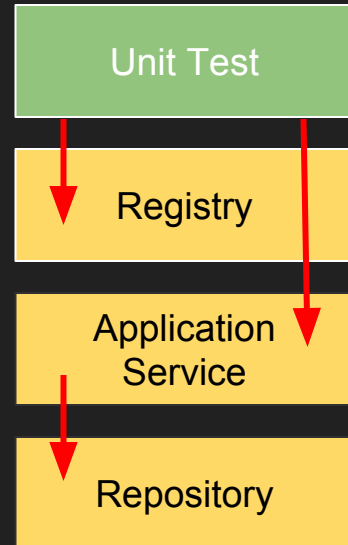
public class ApplicationServiceTest {
    @Test
    public void addCandidate() {
        Candidate candidate = new Candidate("Helder", "hdo@hdo.com", "123");
        CandidateApplicationService service =
            ApplicationRegistry.candidateApplicationService();
        assertTrue("Candidate not added", service.addCandidate(candidate));
    }
}
```

Console JUnit Markers Progress

Finished after 0.023 seconds

Runs: 1/1 Errors: 0 Failures: 0

org.socramad.recruitment.profile.application.ApplicationServiceTest [Runner: JUnit 4] Failure Trace



# Profile context: acceptance test ok

file:///Users/hemarque/Documents/workspaces/craftsmanship/recruitment/target/concordion/org/socramad/recruitment/profile/Candidate.html

## User Story: create a new candidate profile

As a recruiter

I want to create a new candidate profile

In order to track his recruitment process events

## Scenario: request to add the candidate

Given a candidate's name "Helder De Oliveira"

and candidate's email "hdo@hdo.pt"

and candidate's phone number "00351 234987654"

When I request to add the candidate

Then the candidate should be **stored**

## User Story: create a new candidate profile

As a recruiter

I want to add a position to a candidate

In order to list his past jobs

Console JUnit Markers Progress

<terminated> /Library/Java/JavaVirtualMachines/jdk1.8.0\_45.jdk/Contents/Home/bin/java (Dec 10, 2015, 5:30:04 PM)

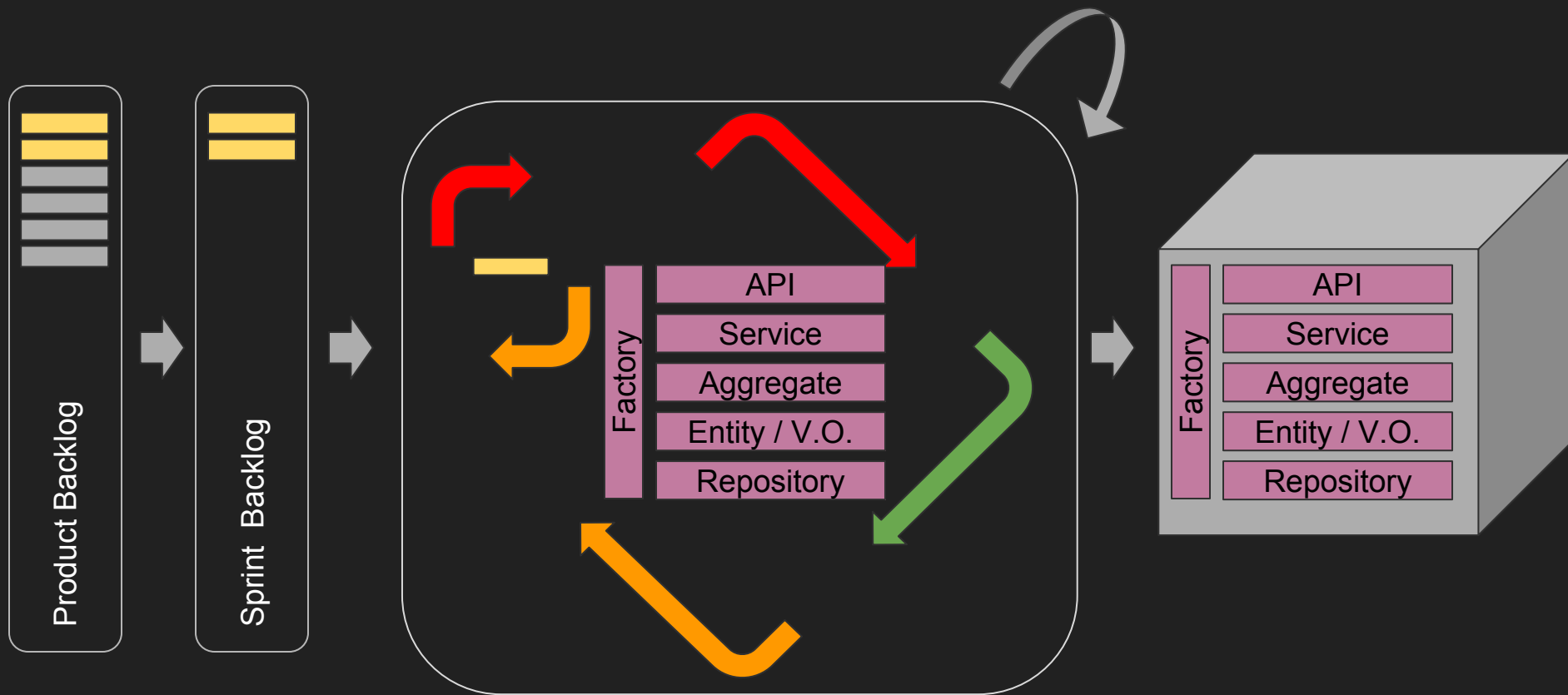
[FINE01



Warning: this code could smell better...



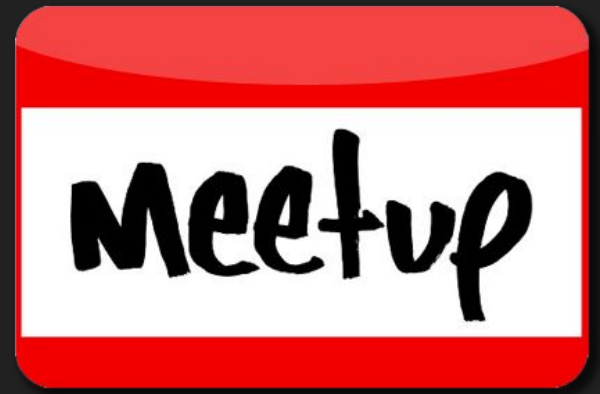
# Scrum + BDD + DDD





# Software Craftsmanship Madrid

<http://www.meetup.com/es/madswcraft/>



I HAVE  
QUESTIONS  
NOW!!!

@HelderDOliveira

