



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks @

# Spring Boot Integration Test

/thoughtworks

# Annotation used in Integration Test

@SpringBootTest

@AutoConfigureMockMvc

@Autowired



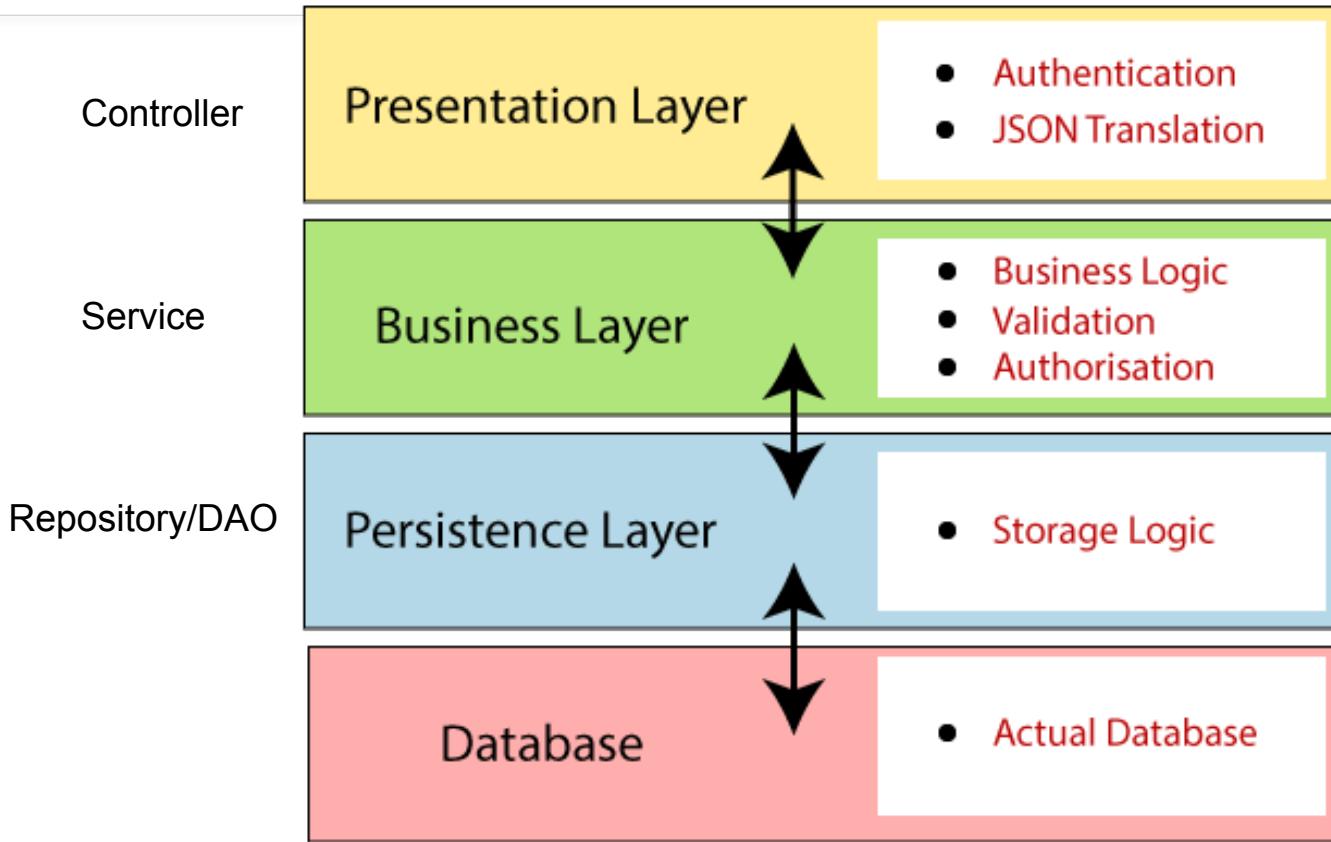
AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks @

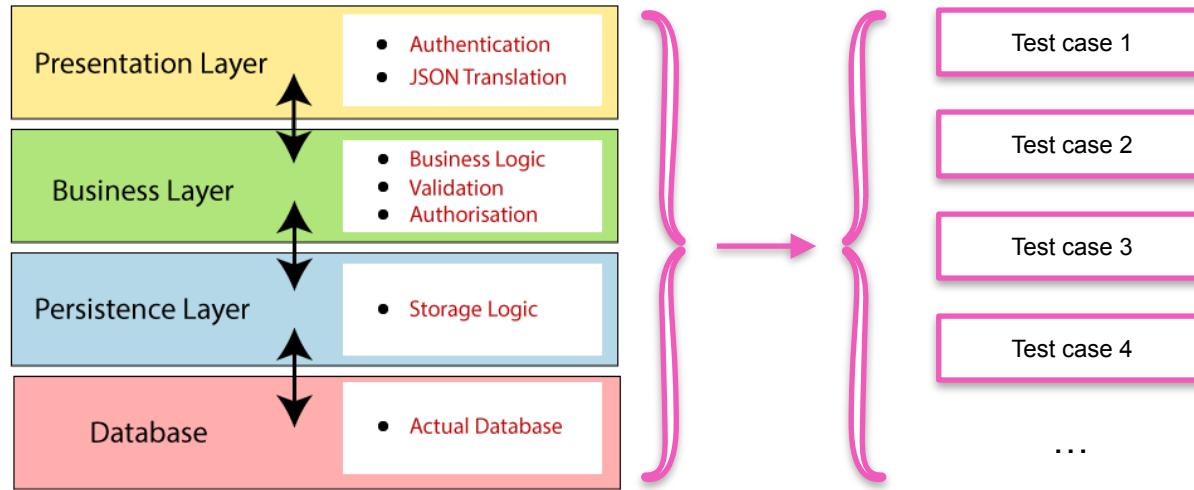
# Spring Boot Architecture

/thoughtworks

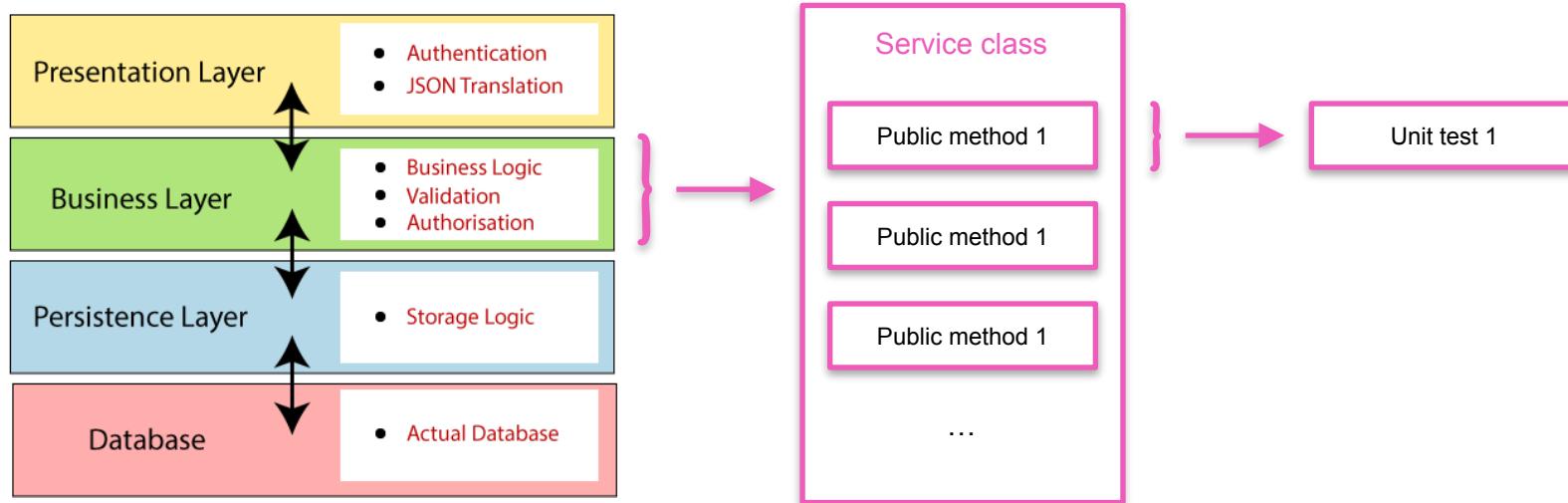
# Spring Boot Architecture



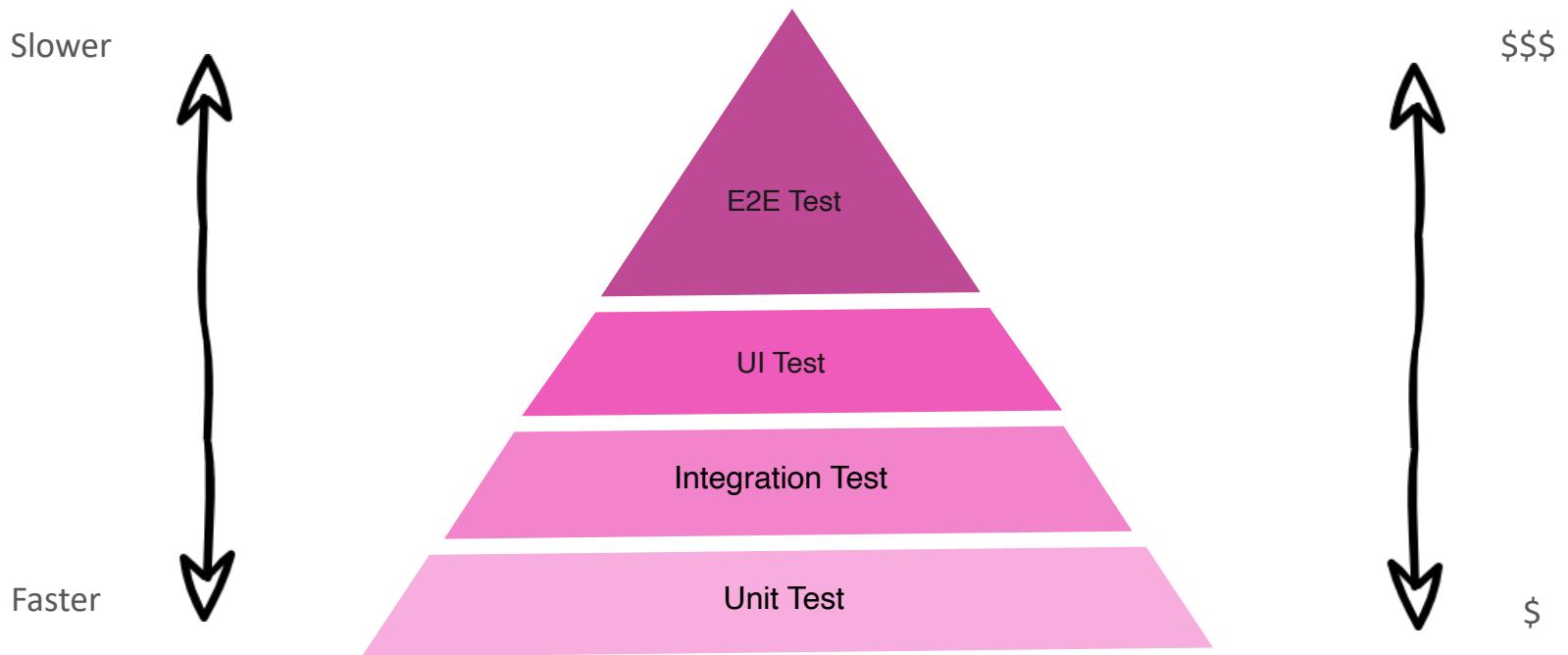
# Integration Test



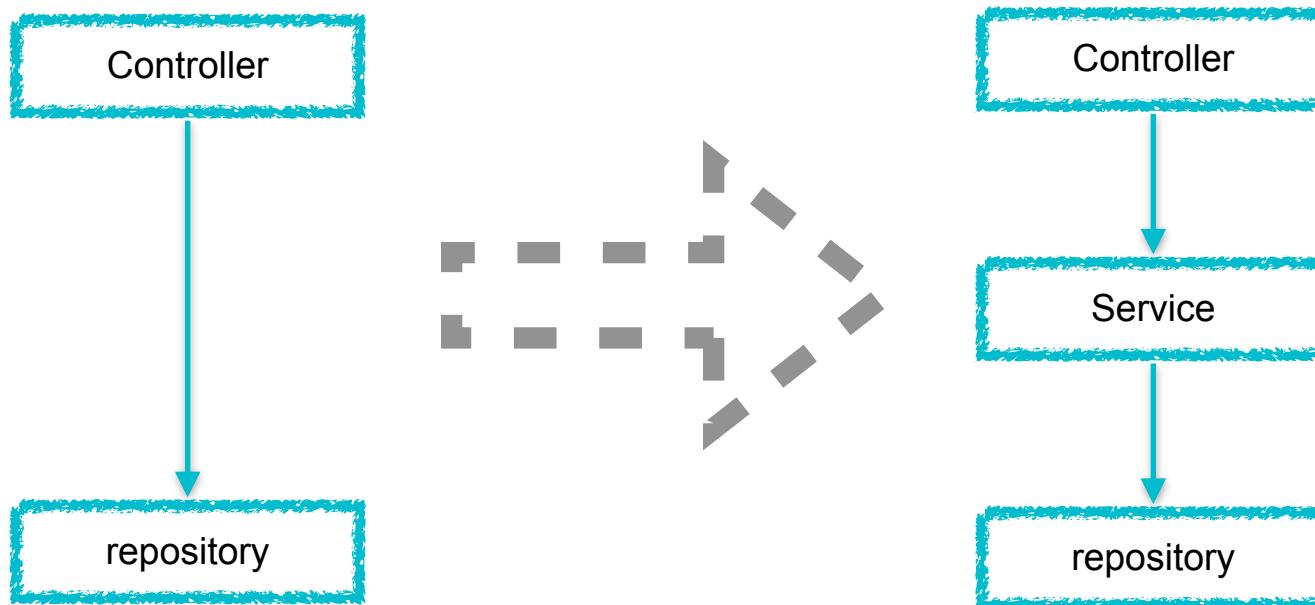
# Unit Test



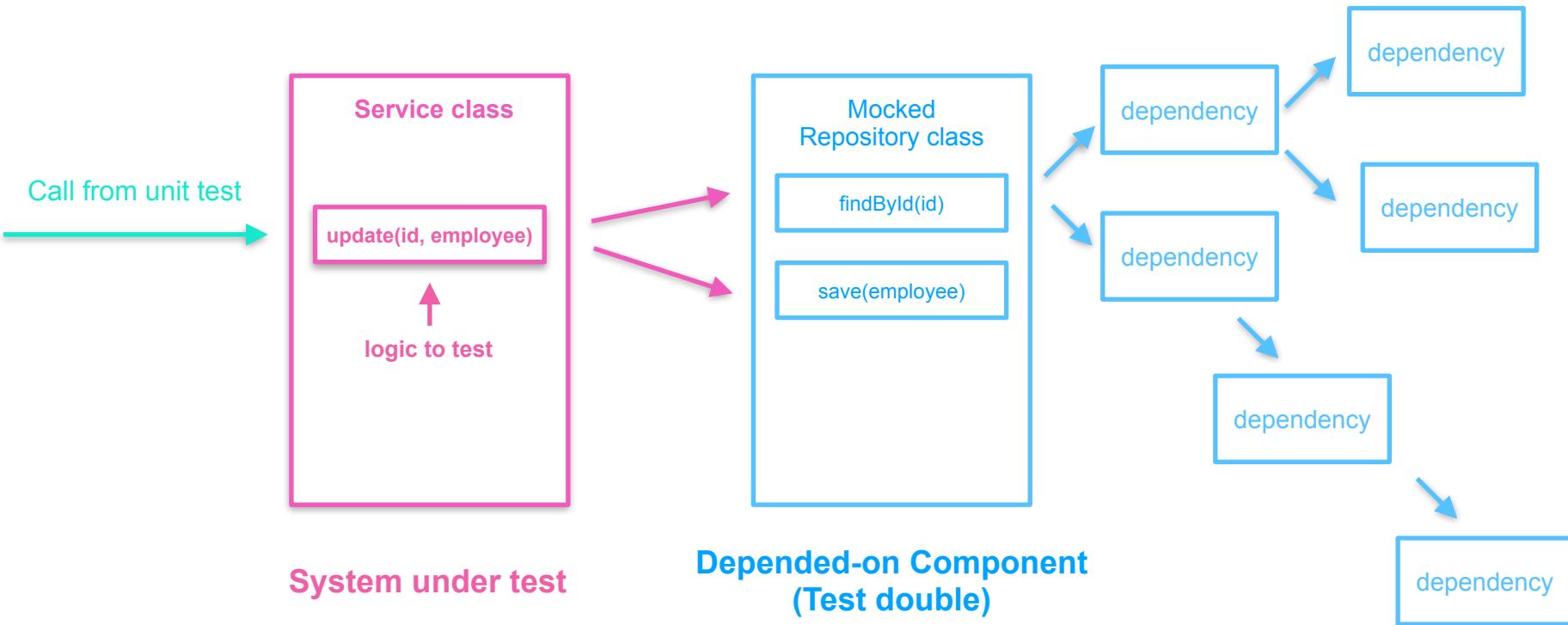
# Test Pyramid



# Employees API Practice

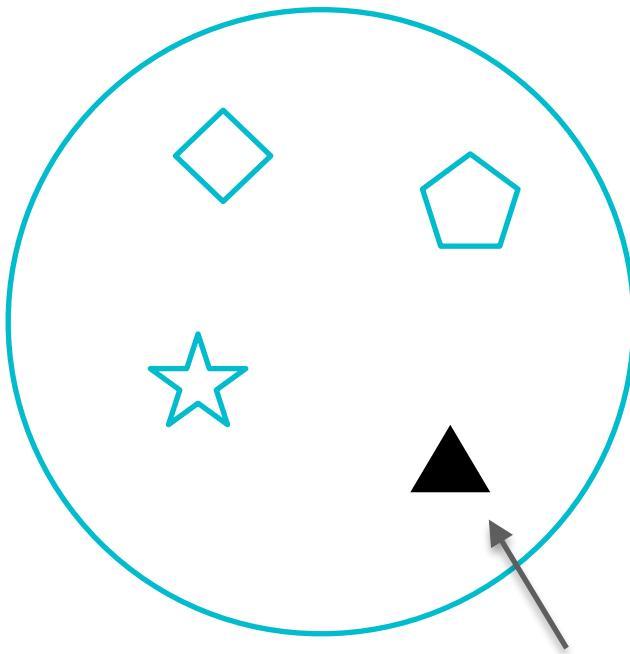


# Unit Test



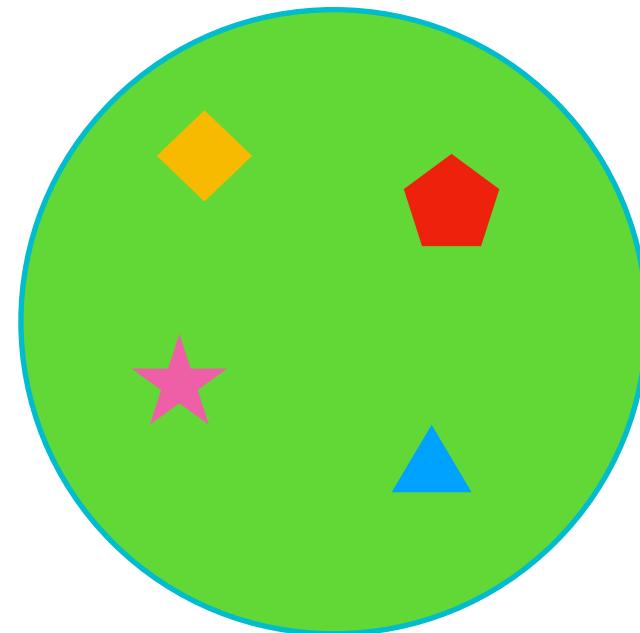
# Unit Test

A mocked repository instance

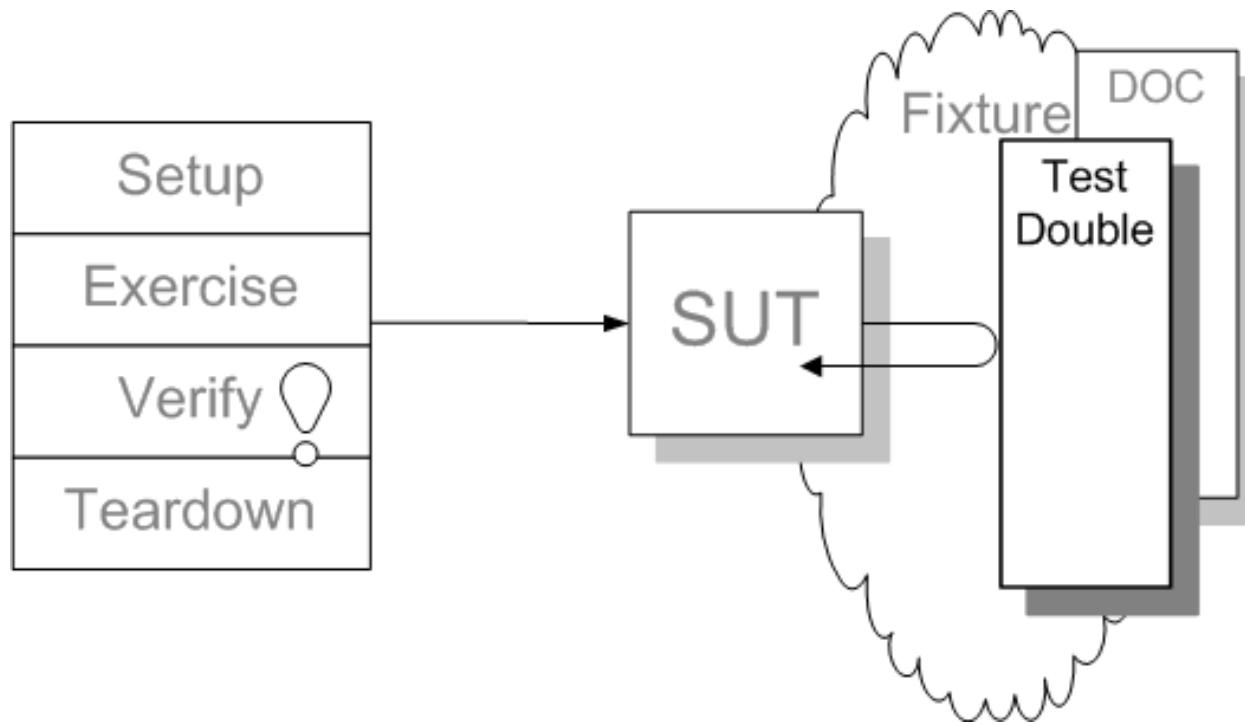


After given(repository. $\triangle$ ).willReturn(xx)

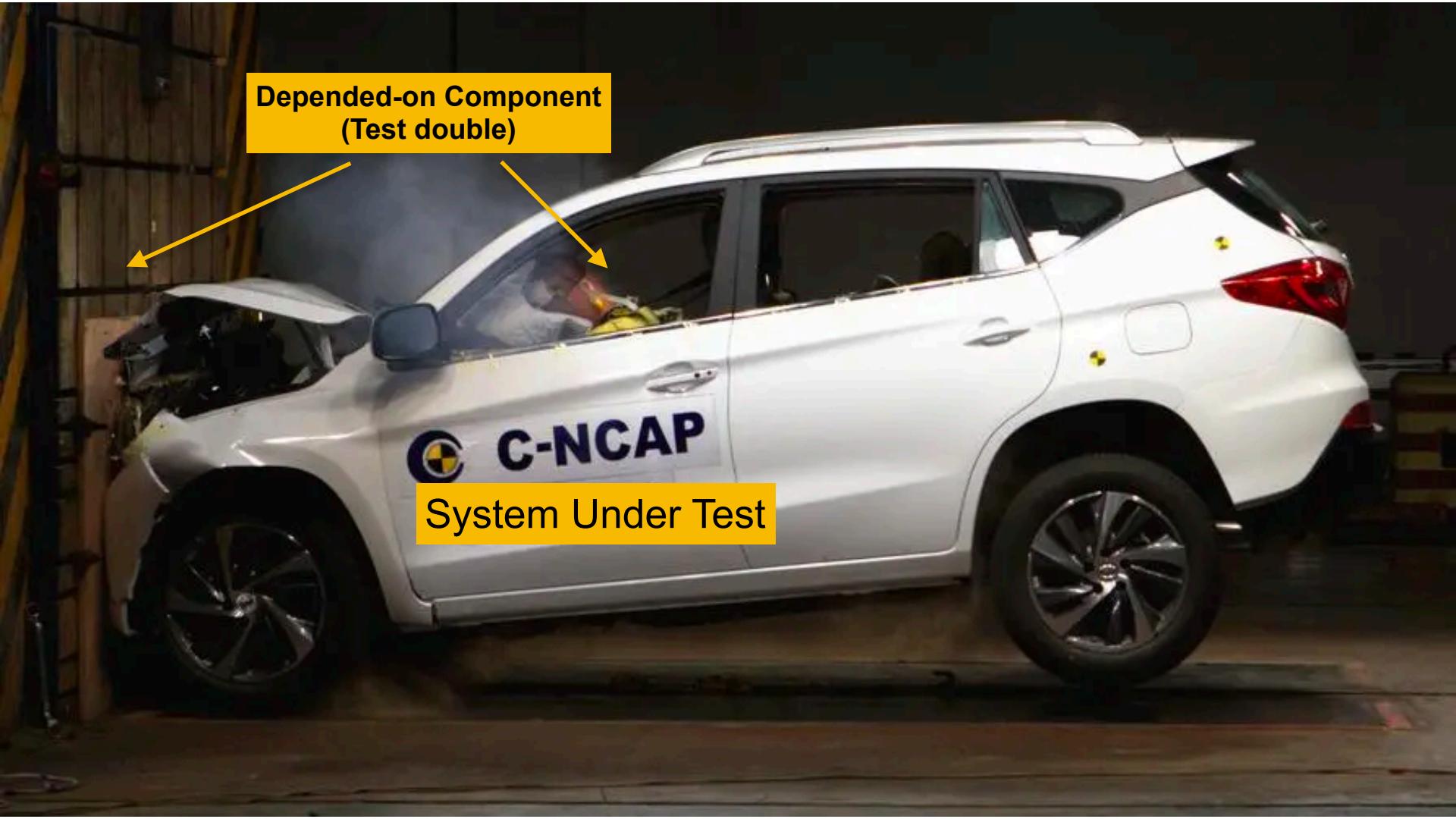
A real repository instance



# Unit Test



<http://xunitpatterns.com/Test%20Double.gif>

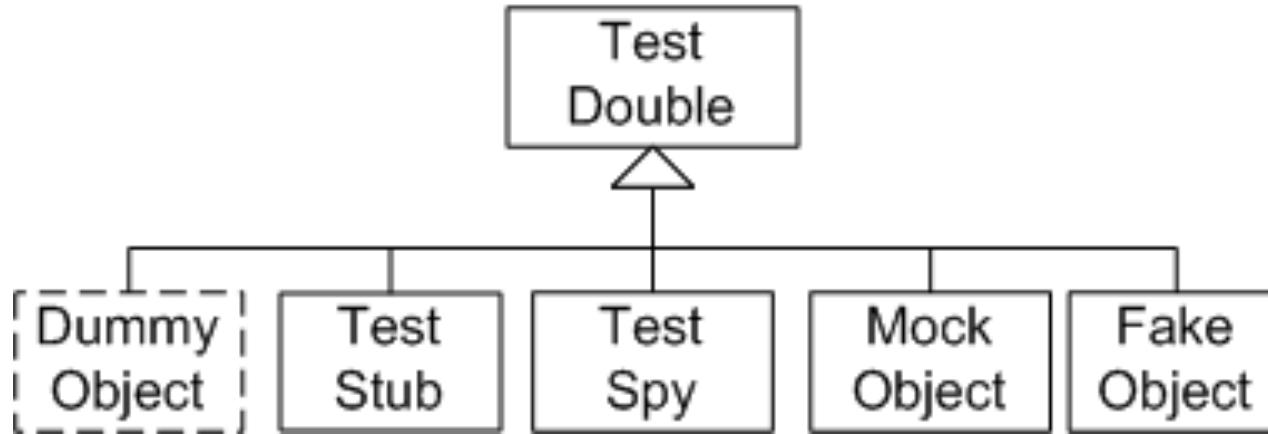


Depended-on Component  
(Test double)

C-NCAP

System Under Test

# Unit Test



<http://xunitpatterns.com/Types%20Of%20Test%20Doubles.gif>

- **Dummy:** objects are passed around but **never actually used**. Usually they are just used to **fill parameter lists**.
- **Stub:** provide **canned answers** to calls made **during the test**, usually not responding at all to anything outside what's programmed in for the test.
- **Spy:** are stubs that also **record** some information based on **how they were called**.
- **Mock:** are **pre-programmed with expectations** which form a specification of the calls they are expected to receive.
- **Fake:** objects actually have **working implementations**, but usually take some **shortcut**



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks @

# Error Handling

/thoughtworks

# Employees API error handling

```
GET      /employees/1          # get a specific employee by ID  
PUT      /employees/1          # update an employee  
DELETE   /employees/1          # delete an employee  
# when id doesn't exist, response status code 500 (Internal server error);
```

# Employees API error handling

```
GET      /employees/1                      # get a specific employee by ID  
PUT      /employees/1                      # update an employee  
DELETE   /employees/1                      # delete an employee  
# when id doesn't exist, response status code 500 (Internal server error);  
# when id doesn't exist, response status code 404 (Not found);
```

# Employees API error handling

## Solutions

- Try catch
- Define http status for Exception  
    @ResponseStatus
- Global Exception Handling  
    @RestControllerAdvice @ExceptionHandler @ResponseStatus



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks @

# Homework

/thoughtworks

# Homework



22:00pm

- Diary using ORID
- Finish spring boot practice Company & Employee:
  - Finish Integration Test on all APIs
  - Finish service test
  - Finish 3 layers with controller/service/repository with TDD
  - Finish ControllerAdvice (two exceptions: employeNotFound & companyNotFound)
- Install MongoDB and MongoDB Compass

## Requirement

1. Baby step commit with meaningful message
2. Format code before commit
3. Using meaningful names



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks @

# Thank You !

/thoughtworks