

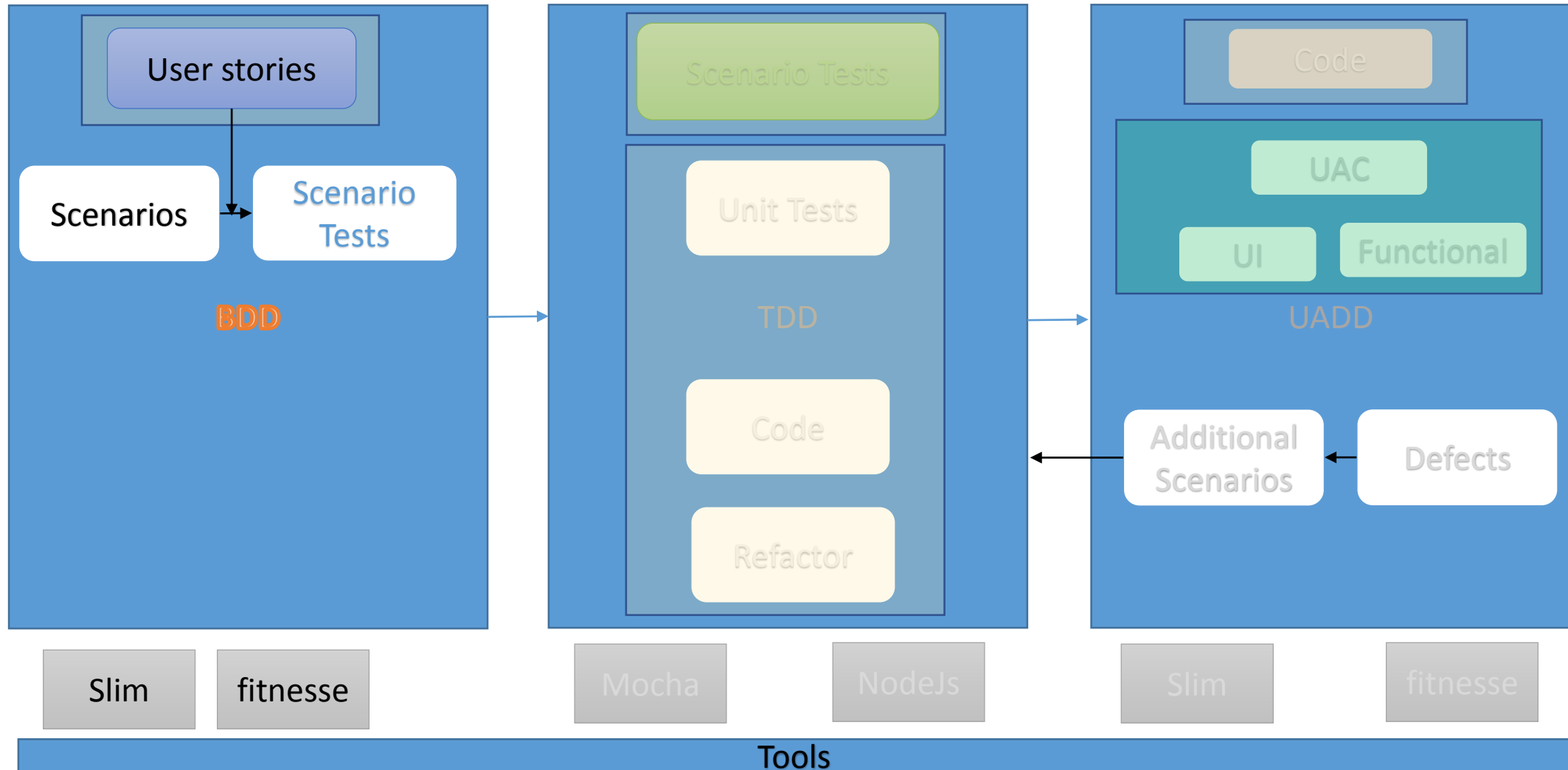
BDD with SlimJs

Presented

By

Ravichandran J V

BDD + TDD + UADD



Given, when, then ... the holy grail

- As a <role> The Context
- I want... The Feature
- So that... The Business Value



Given, when, then ... the holy grail

- Given a scenario Context
- When an input is made Event
- Then ensure some outcome Outcome

Given, when, then ... in action!

Requirement

- User types some English sentence that must comply to a supplied drill line of text.
- If user misses or adds any additional words from the drill text, then the words must be highlighted as errors for the user to correct.

Given, when, then ... in action!

Requirement

Given

- User types some English sentence that must comply to a supplied drill line of text.

when

then

- If user misses or adds any additional words from the drill text, then the words must be highlighted as errors for the user to correct.

Identify Scenarios (Behavior)

Scenario 1 – if user misses words

- If user misses or adds any additional words from the drill text,

Scenario 2 – if user adds words

Scenario testing

```
!define TEST_SYSTEM {slim}
!define COMMAND_PATTERN {node E:\fitnesse\node_modules\slimjs\src\SlimJS %p}
!path E:\fitnesse\slimtests

|import|
|test1|

!|scenario|Missed word| Scenario 1
!|scenario|Added word|

|script|test1.Test1|missing word|word|
|Missed word|
|check|missingwordscenariotest|missing| Scenario & method

|script|test1.Test1|added word|added word added|
|Added word|
|check|addedwordscenariotest|added|
```

import

test1

scenario Missed word

scenario Added word

script	test1.Test1	missing word	word
- Missed word			
scenario Missed word			
check	missingwordscenariotest	[missing]	expected [missing]

script	test1.Test1	added word	added word added
- Added word			
scenario Added word			
check	addedwordscenariotest	[added]	expected [added]

Scenario testing

- Remember, this is the first stage of development. The business rules and requirements are just in.
- The code contains just that much logic to run the Scenario test.
- The code will emerge once the next phase in the development lifecycle – TDD – begins.

Scenario testing

// Code – twoscenarios.js

```
function myApi(){};
myApi.prototype = {
  twoscenarios:function(drill,user){
    this.missingwordscenariotest=function(){
      return "missing";
    };

    this.addedwordscenariotest=function(){
      return "added";
    };
  }
};
module.exports=new myApi();
```

// Test

```
var app=require('./twoscenarios.js');
var assert=require('assert');

describe('#get missing word', function() {
  it('should return missing word',function(){
    var ob=new app.twoscenarios("missing
word","word");

    assert.equal("missing",ob.missingwordscenariotest());
  });
});
describe('#get added word', function() {
  it('should return added word',function(){
    var ob=new app.twoscenarios("word","word
added");

    assert.equal("added",ob.addedwordscenariotest());
  });
});
```

Thank you!