Verifying Behavior with a Mock

Nathan Taylor
SOFTWARE ENGINEER

@taylonr taylonr.com

What Is a Mock?

Mock

Mocks are fake methods (like spies) with pre-programmed behavior (like stubs) as well as pre-programmed expectations.

A mock will fail your test if it is not used as expected.

mock

Mocks are **fake methods (like spies)** with pre-programmed behavior (like stubs) as well as pre-programmed expectations.

A mock will fail your test if it is not used as expected.

mock

Mocks are fake methods (like spies) with **pre-programmed behavior (like stubs)** as well as pre-programmed expectations.

A mock will fail your test if it is not used as expected.

mock

Mocks are fake methods (like spies) with pre-programmed behavior (like stubs) as well as pre-programmed expectations.

A mock will fail your test **if it is not used as expected.**

Creating Mocks

```
const mock = sinon.mock(Wishlist);
```

Stub

```
{
  addItem : sinon.stub,
  removeItem : sinon.stub
}


{
  addItem : sinon.stub,
  removeItem : sinon.stub
}
```

Mock

```
{
  addItem : function(),
  removeItem : function()
}
{
  expects: function()
  invokeMethod: function()
  object: Object {addItem: }
  restore: function()
  verify: function()
}
```

Behind the scenes mocks are pretty different

Use mocks to verify, not just specify behavior

How to use mocks

Using Mocks

```
sinon.spy(model, 'create');

return controller.create(req, res).then(() => {
    return expect(model.create.called).to.be.true;
});
```

Assign expectations to mocks

Mock Expectations

```
const mockWishlist = sinon.mock(Wishlist);

mockWishlist.expects('addItem').atLeast(1);
```

Mock Expectations

```
const mockWishlist = sinon.mock(Wishlist);

mockWishlist.expects('addItem').atLeast(1);
```

Mock Expectations

```
const mockWishlist = sinon.mock(Wishlist);

mockWishlist.expects('addItem').atLeast(1);
```

Mock Expectations

```
const mockWishlist = sinon.mock(Wishlist);

mockWishlist.expects('addItem').atLeast(1);

Wishlist.addItem('test');
```

Call Count Expectations

atLeast

atMost

never

once

exactly

Mock Arguments

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').withArgs(1);
```

Mock Arguments

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').withArgs(1);

mockModel.expects('findAll').withExactArgs({where: {
    author = {
        $ilike: '%Piper%’
    }
}});
```

Mock Chaining

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);
```

Similar pattern to spies and stubs

Checking Mocks

Verify Mocks

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);

controller.getById(1);

mockModel.verify();
```

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);

controller.getById(2);

mockModel.verify();
```

Failing Verification

ExpectationError: Unexpected call: findById(2)

Expected findById(1[, ...]) once (never called)

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);

controller.getById(2);

mockModel.verify();
```

Failing Verification

ExpectationError: Unexpected call: findById(2)

Expected findById(1[, ...]) once (never called)

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);

controller.getById(2);

mockModel.verify();
```

Failing Verification

ExpectationError: Unexpected call: findById(2)

    Expected findById(1[, ...]) once (never called)

```
const mockModel = sinon.mock(model);


mockModel.expects('findById').once().withArgs(1);

controller.getById(1);
controller.getById(2);


mockModel.verify();
```

Failing Verification

ExpectationError: Unexpected call: findById(2)

Expectation met findById(1[, ...]) once

Relaxed Expectation

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').atLeast(1);
```

Additional Expectation

```
const mockModel = sinon.mock(model);

mockModel.expects('findById').once().withArgs(1);
mockModel.expects('findById').once().withArgs(2);
```

Mocks can be brittle

Verify Payment

Mocks vs. Spies vs. Stubs

Spies

Stubs

Mocks are enforcers

Getting better at matching