

Why Testing Is Important And Where Do I Start?

Nolan Erck

github.com/nolanerck
why-testing-is-important-preso

ABOUT ME

- Owner @ South of Shasta
- Developer Consultancy
- VP of Engineering @ e-Mission Control
- Organizer @ Sac Interactive Tech Meetup (formerly Sacramento CFUG)
- Recovering Video Game Developer (Grim Fandango, SimPark, Star Wars Phantom Menace, etc)
- Music Junkie





I'M REAL PROUD OF MY
RECORD COLLECTION



Should I test my web app?

**Yes.
No exceptions.**

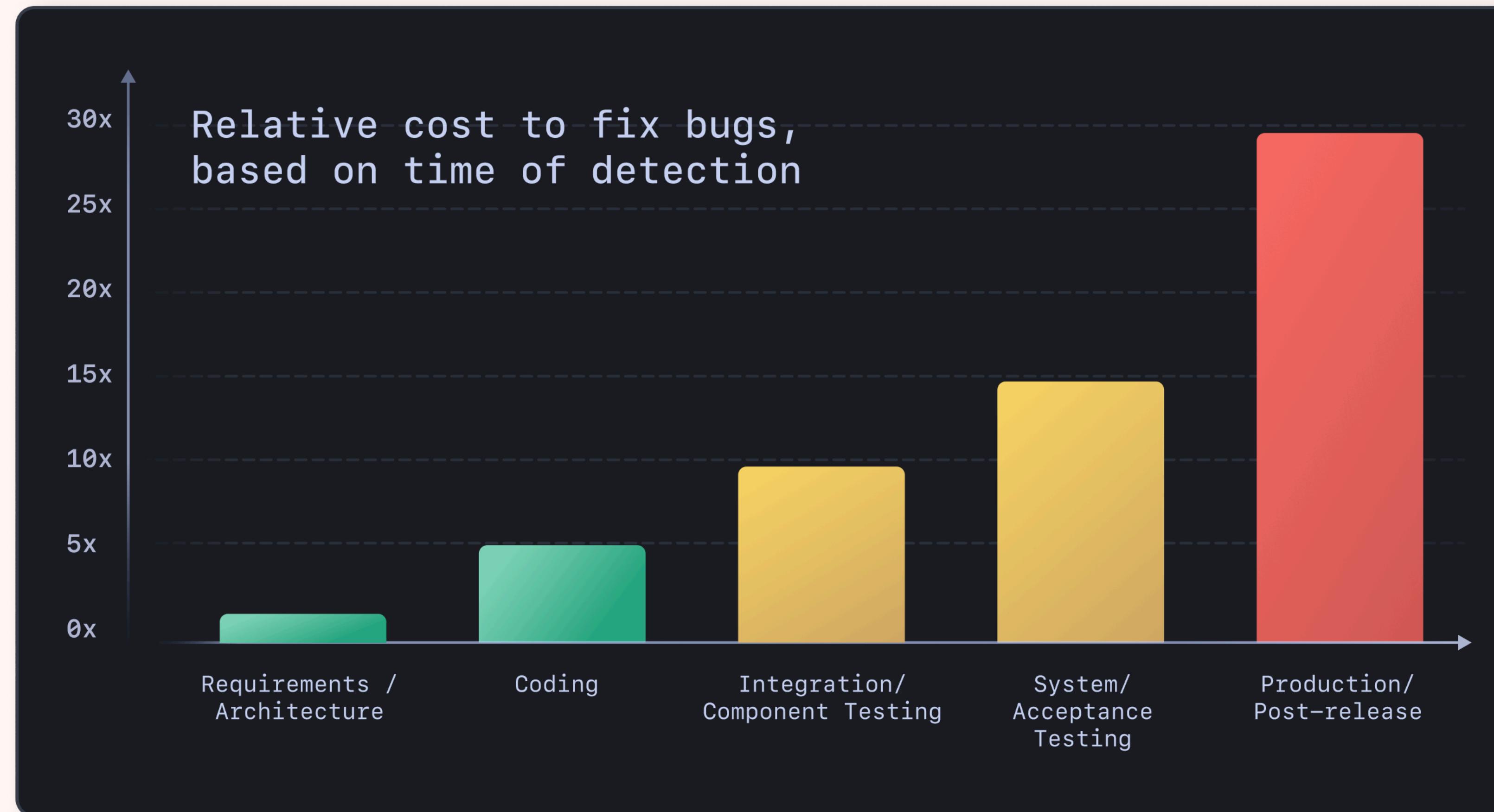
The difference between a Senior
and a Junior Developer is...

A Senior Developer
tests their code.

WHY TESTING IS IMPORTANT

- **Find and fix bugs**
 - **The sooner the better, and cheaper!**
 - **Customers finding bugs in Production = bad PR, bad reviews, bad everything**
 - **Want to find and fix issues *before* customers see them**
-

WHY TESTING IS IMPORTANT



Source: <https://deepsource.com/blog/exponential-cost-of-fixing-bugs>

So how / where do I start testing my app?

Just start.

JUST START

- **Manual testing can (and should) be done immediately**
 - **Run your code, click the button, does it work?**
 - **Don't worry about all the options, analysis paralysis**
 - **TestBox? Cypress? Codecept? Chai? Mocha? MX Unit?**
 - **Build the testing process over time, update as needed**
-

MAKESHIFT TEST SUITE

- **Don't have to use a testing library to start**
 - **[demo 1]**
 - **Pros: quick to start using, nothing to install**
 - **Cons: Tedious, exception = no preceding tests run, only output is primitive HTML (no PDF, CSV, summary data, etc)**
-

MAKESHIFT TEST SUITE

- **Want to do that, but more, better, without tons of developer overhead.**
 - **Run *all* the tests — if 1 fails, the rest still run**
 - **Also, run specific sub-groups of tests together**
 - **Summary stats - X tests passed, Y tests failed**
 - **Execute the tests via browser, command line, via DevOps tools**
 - **For future planning, need something better (even if we don't get fancy with it right away)**
-

TestBox

TESTBOX

- **Can be used to test ANY cfml application!**
 - **Not just ColdBox apps!**
 - **Better structured codebases yield better quality test results**
 - **CFincludes — poor**
 - **Custom Tags — medium**
 - **Functions and CFC — better**
-

STARTING WITH TESTBOX

- **box install testbox**
 - **That's it!**
 - **/testbox — the TestBox app**
 - **/testbox/runner.cfm — easiest way to use TestBox**
 - **/testbox/tests --- default test location**
 - **[demo 2]**
-

TESTING CFINCLUDES

- **CFincludes are a poor choice for modularity**
 - **Leak data**
 - **Not self-documenting**
 - **Unclear what the inputs/outputs are**
 - **But are still testable (kind of)**
 - **Your mileage may vary, depends on the code quality**
 - **[demo 3]**
-

TESTING COLDBOX

- **box coldbox create app**
 - **ColdBox installs w/ TestBox ready to go!**
 - **Exact same setup**
 - **/testbox/tests/runner.cfm**
 - **CFCs can be generic things like we saw before**
 - **Or they can be ColdBox related**
 - **Run an event that uses WireBox, etc**
-

TESTING CFC WITH TESTBOX

```
component extends="textbox.system.BaseSpec" appMapping="/" {  
  
    function run(){  
  
        describe( "Facility Model Suite", function(){  
  
            it( "should initialize properly", function(){  
  
                var objFacility = new models.Facility();  
  
                expect( !isNull( objFacility ) );  
                expect( objFacility )  
                    .toBeTypeOf( 'component' )  
                    .toBeInstanceOf( 'models.Facility' );  
            });  
        });  
    };  
}
```

TESTING COLDBOX EVENTS

```
component extends="coldbox.system.testing.BaseTestCase"
{
    beforeEach( function( currentSpec ) {
        // Setup as a new ColdBox request, VERY IMPORTANT. ELSE EVERYTHING LOOKS LIKE THE SAME REQUEST.
        setup();
    } );

    function run() {

        describe( "Main Handler", function() {

            it( "can render the homepage", function() {
                var event = this.get( "main.index" );
                expect( event.getValue( name = "welcomemessage", private = true ) ).toBe( "Welcome to ColdBox!" );
            } );
        } );
    }
}
```

TESTBOX IN THE REAL WORLD

- Our apps are way more complex than this! Show us something real!
 - Unlimited tests, doing various complex things!
 - Does TestBox really handle big code bases?!
 - I'll literally show you my current app's test suite...
-

Front-end Testing

FRONT-END TESTING

- **Doesn't care if the server is CFML, Java, PHP, Python, etc.**
 - **Runs a client (browser) and clicks around on the site like a real user.**
 - **Several options:**
 - **Cypress, CodeCept, Mocha, Chai, etc.**
 - **They all do basically the same job, pick the one you like and...just start**
-

CYPRESS.JS

- **JavaScript based**
 - **Super easy to install**
 - **Great command line tools**
 - **Excellent YouTube channel**
 - **Constantly adding new features, well documented, active community**
-

CYPRESS.JS

```
describe( 'User login functionality', () => {

    it('Should log the user in successfully', () => {
        cy.visit( 'http://sitename/login' );

        cy.get( '#Email' )
            .type( 'nolan@sitename.com' );

        cy.get( '#Password' )
            .type( 'mySecretPassword' )
            .should( "have.value", 'mySecretPassword' );

        cy.get( "#btnSignIn" ).click();
    });

    it('Should log the user out successfully', () => {
        cy.visit( 'http://sitename/mainmenu' );
        cy.get( "#btnLogOut" ).click();
    });
});
```

CYPRESS.JS CLI

- **In the GUI**
 - **npx cypress open**
 - **Run them all from the CLI**
 - **npx cypress run**
 - **Let's look at these in a real app!**
-

CYPRESS.JS

- **Pros**
 - **Super easy to use**
 - **Great community and free support available**
 - **Cons**
 - **How good/bad is your HTML markup? That will effect selectors working correctly.
(This is true for all front-end testing tools, not specific to Cypress.)**
-

Automating Tests

AUTOMATED TESTS

- “I forgot to run the test suites” is common, accidents happen.
 - ...and developers are lazy. ;)
 - Can automate all of this
 - GitHub Actions, BitBucket Pipelines, etc.
 - cypress.yml
 - testbox.yml
-

CYPRESS.YML

```
name: Cypress
on:
  push:
    branches: [ 'main' ]
  pull_request:
    branches: [ 'main' ]
jobs:
  run-tests:
    runs-on: ubuntu-22.04
    steps:
      - name: Checkout
        uses: actions/checkout@v3
      - name : Start the servers and wait until up
        run: ./ci.sh
      - name: Checkout
        uses: actions/checkout@v3
      - name: Cypress run
        uses: cypress-io/github-action@v5
        with:
          browser: chrome
          headed: true
```

TESTBOX.YML

```
on:
  push:
    branches: [ 'main' ]
  pull_request:
    branches: [ 'main' ]
jobs:
  tests:
    name: Tests
    runs-on: ubuntu-latest
    steps:
      - name: Checkout
        uses: actions/checkout@v3

      - name : Start the servers and wait until up
        run: ./ci.sh

      - name: Checkout Repository
        uses: actions/checkout@v3
        with:
          fetch-depth: 0

      - name: Setup Java
        uses: actions/setup-java@v2
        with:
          distribution: "adopt"
          java-version: "11"

      - name: Setup CommandBox
        uses: Ortus-Solutions/setup-commandbox@main

      - name: Install dependencies
        run: box install --verbose

      - name: Start server
        run: box server start cfengine=lucee@5.3 --noSaveSettings

      - name: Run TestBox Tests
        run: box testbox run reporter=text
```

AUTOMATED TESTS

- **Not as crazy as that looked.**
 - **Lots of easy to follow tutorials and documentation on line.**
 - **I promise.**
-

Excuses

IT'S HARD TO CONFIGURE!

- **For TestBox, type “box install testbox” and wait 10 seconds...done**
 - **For Cypress, type “npm install cypress” and wait 10 seconds...done**
 - **Both of them work immediately**
 - **(You’ll probably want to adjust a couple things but it’s not required.)**
-

THAT'S A LOT OF TYPING!

- **This is an invalid excuse. :)**
 - **1. “Extra typing” is a 1-time pain point for 1 developer**
 - **2. Use the tools you already have installed**
 - **Testing libraries all have a CLI**
 - **Brand new VS Code Extension for TestBox**
-

I FEEL OVERWHELMED!

- **Relax**
 - **Start wherever you're comfortable starting**
 - **We did not build our entire test suite on day 1**
 - **There is no BAD place to add testing to your app**
 - **100% test coverage from day 1 is too much**
 - **Just start, add to it as you can**
 - **In Jira tickets: “And for this task write front-end tests for X, Y, and Z...”**
-

SO MUCH TO LEARN!

- **Just CFScript or JavaScript**
 - **All the front-end tools use similar JavaScript**
 - **A tiny bit of syntax**
 - **Mostly just a few function names**
 - **The test suites are really very similar to each other**
-

CYPRESS.JS

```
describe( 'User login functionality', () => {

    it('Should log the user in successfully', () => {
        cy.visit( 'http://sitename/login' );

        cy.get( '#Email' )
            .type( 'nolan@sitename.com' );

        cy.get( '#Password' )
            .type( 'mySecretPassword' )
            .should( "have.value", 'mySecretPassword' );

        cy.get( "#btnSignIn" ).click();
    });

    it('Should log the user out successfully', () => {
        cy.visit( 'http://sitename/mainmenu' );
        cy.get( "#btnLogOut" ).click();
    });
});
```

TESTBOX CFC

```
component extends="testbox.system.BaseSpec" appMapping="/" {  
  
    function run(){  
  
        describe( "Facility Model Suite", function(){  
  
            it( "should initialize properly", function(){  
  
                var objFacility = new models.Facility();  
  
                expect( !isNull( objFacility ) );  
                expect( objFacility )  
                    .toBeTypeOf( 'component' )  
                    .toBeInstanceOf( 'models.Facility' );  
            });  
        });  
    };  
}
```

ANGULAR

```
describe('hello-world App', () => {
  let page: AppPage;

  beforeEach(() => {
    page = new AppPage();
  });

  it('should display welcome message', () => {
    page.navigateTo();
    expect(page.getParagraphText()).toEqual('Welcome to app!');
  });
});
```

Just Start

Other Things That Help

OTHER THINGS THAT HELP

- **Use a source control branching strategy**
 - **EVERYONE on the team needs to follow the plan**
 - **Writing code in “main” = red flag, fix that asap**
 - **Write the tests when you write the code, not “some time in the future”**
 - **Test Driven Development makes this easier (sometimes)**
 - **No Deploy Fridays**
 - **Frameworks — MVC, DI, AOP, etc — all make code more easily testable**
-

Learning in 30 minutes a day

MORE RESOURCES

- **This conference!**
 - **Cypress YouTube Channel**
 - **CFCasts.com**
 - **Ortus TestBox Workshop**
 - **southofshasta.com**
 - **“Testing My Non ColdBox Site With TestBox” preso**
-

THANKS!

- You can find me here:
 - southofshasta.com
 - Twitter [@southofshasta](https://twitter.com/southofshasta), [@nolanerck](https://twitter.com/nolanerck)
 - nolan@southofshasta.com
 - e-missioncontrol.com
-