Building a JavaScript Development Environment

YOU NEED A STARTER KIT

Cory House

@housecor bitnative.com

Do you like to fail fast?

Do you like rapid feedback?

Want to know immediately
when a test is failing?

Overwhelmed by
all the options?

So. Many. Decisions.

- Editor
  - Which one?
  - Which plugins?
  - Use built in terminal?
  - Editor config
- Module format
  - ES6 Modules, CommonJS...
- HTML generation
  - Minify?
  - Use plugin?
  - Inject prod only concerns?
  - Templating language?
- Transpiling
  - Native ES or diff language?
  - Use experimental features?
  - Which plugins?
  - Production vs dev config
- Bundler
  - Webpack, Browserify, Rollup...
- Linting
  - Which linter?
  - Enable which rules?
  - Warning or error?
  - Which plugins?
  - Use a preset?
- Testing
  - Framework?
  - Assertion Library?
  - Helpers?
  - Test file location?
  - File naming?
  - What environment?
  - Mocking?
  - Code Coverage
  - Continuous Integration
- Project structure
  - By file type or feature?
  - Centralize API?
  - Allow Inline JS?
  - Extract to POJOs?
- HTTP
  - Library
  - Mock schema format
  - Mock data generation
  - Mock server
- Production build
  - Minification
  - Sourcemaps
  - Bundle splitting
  - Cache busting
  - Error logging

Overwhelmed yet?

Make ”best practices”
automatic.

Your team *needs* a JavaScript starter kit.

Starter kits speed development, increase quality and reduce fatigue.

Why a Starter Kit?

Codifies

- Decisions
- Best practices
- Lessons learned

Encourages consistency

Avoids forgetting important details

Increases quality

- Doing the “right” thing is the easy thing

Avoids repeating work

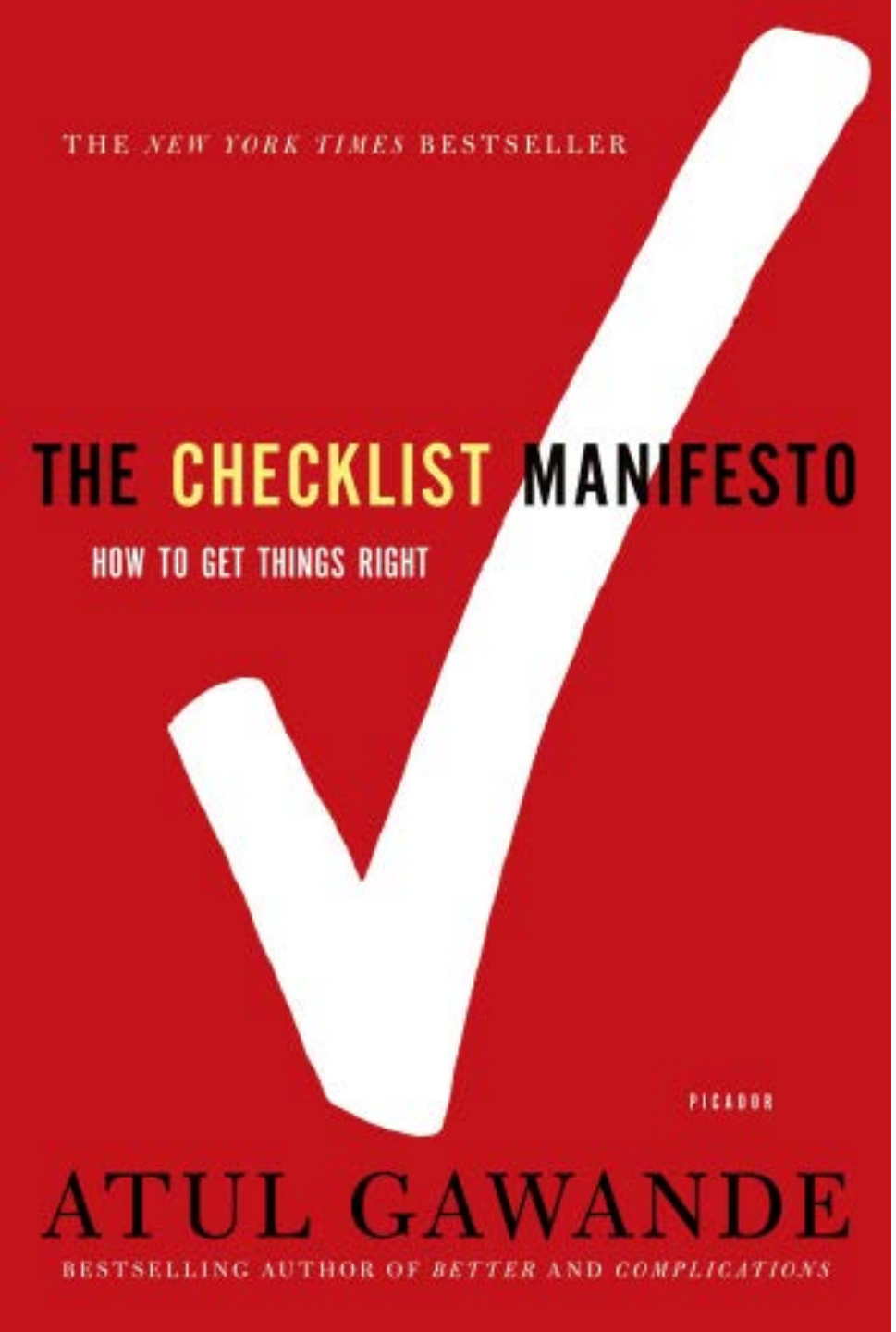
Today's list of "best practices" is too long for any individual to track.

Create a starter kit that does it for you.

A Starter Kit Is an Automated Checklist

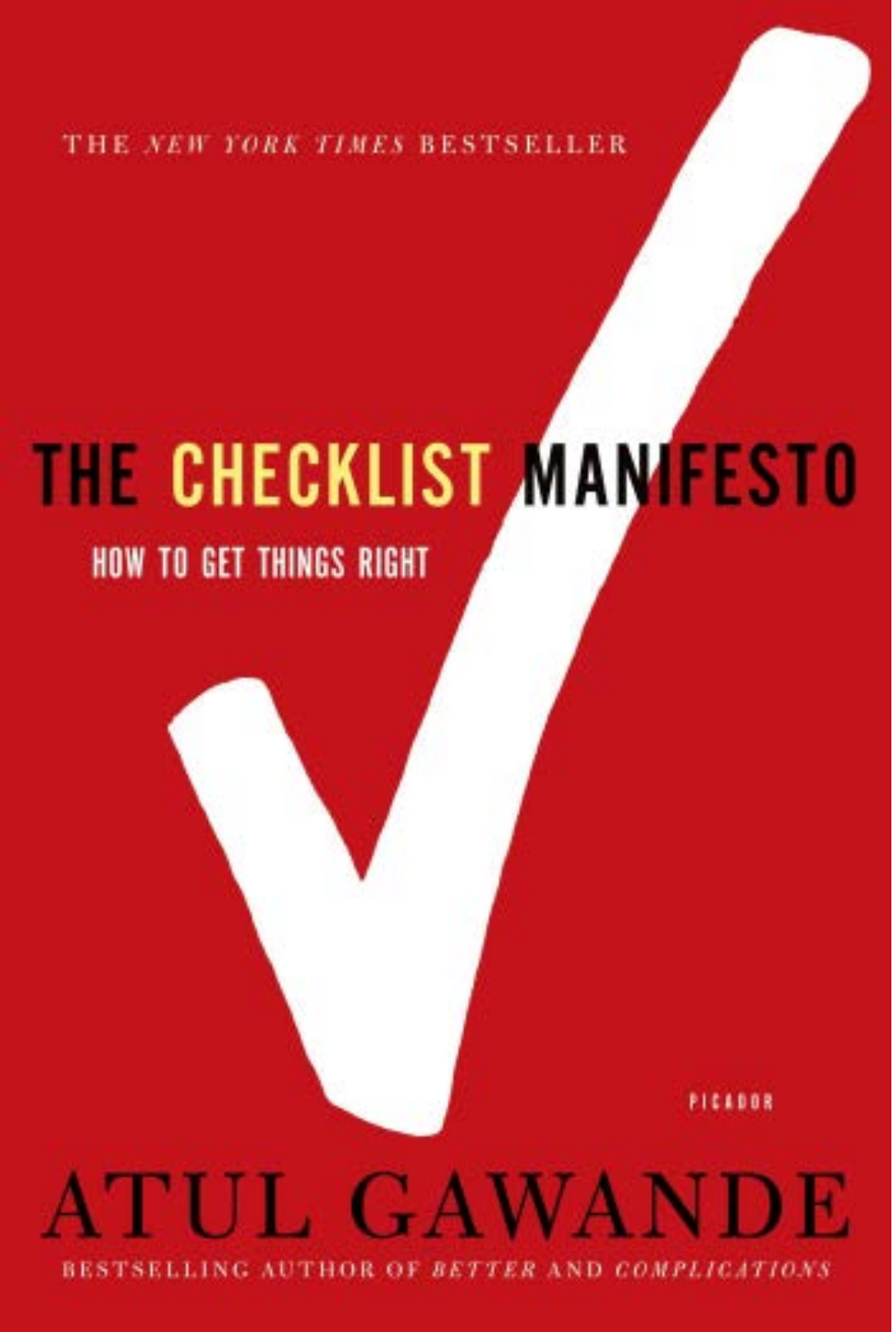
Line Checklist

1. Wash their hands with soap
2. Clean patient's skin with chlorhexidine
3. Put sterile drapes over entire patient
4. Wear sterile mask, hat, gown & gloves
5. Put sterile dressing over catheter



We think we can remember all the steps.

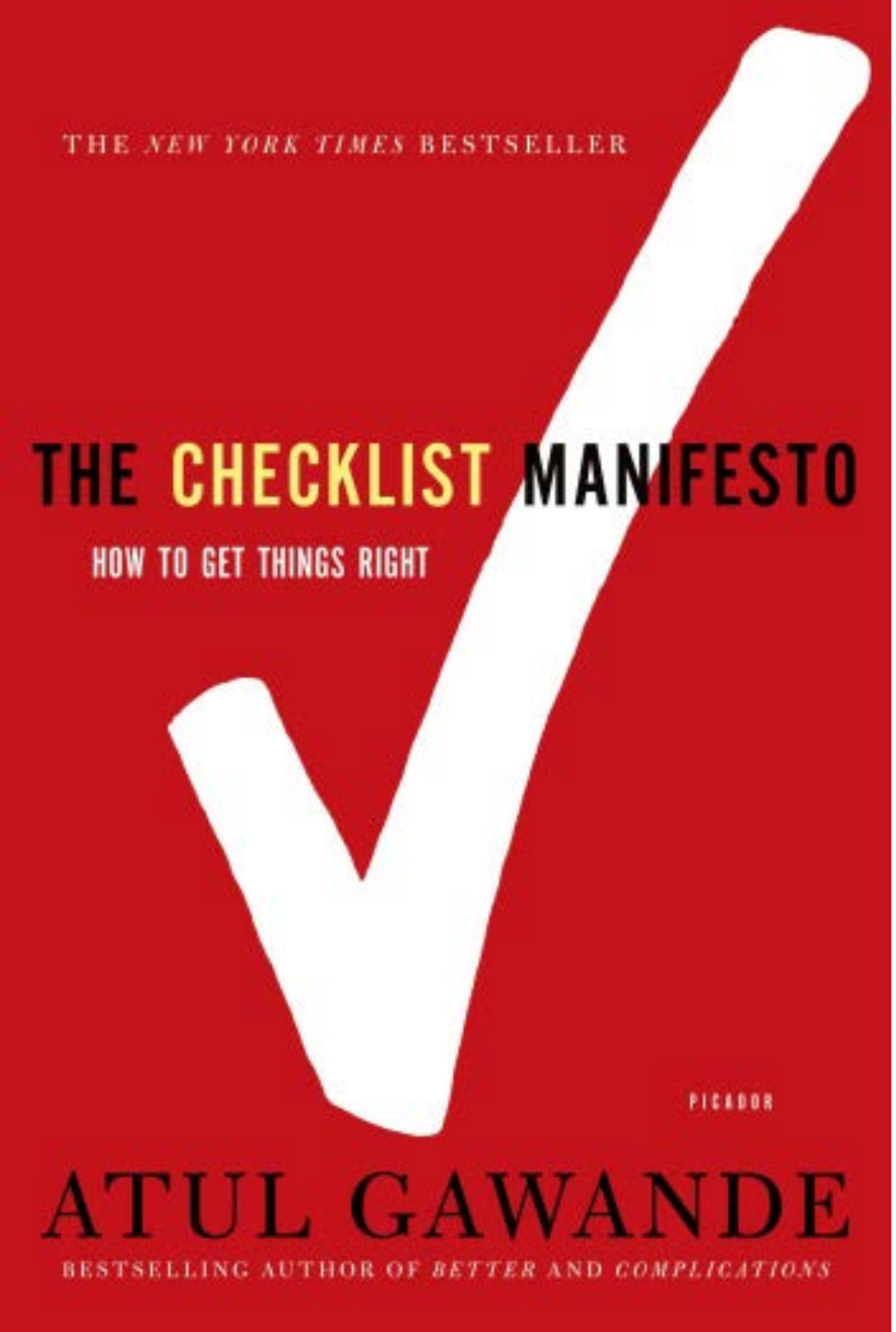
We can’t.



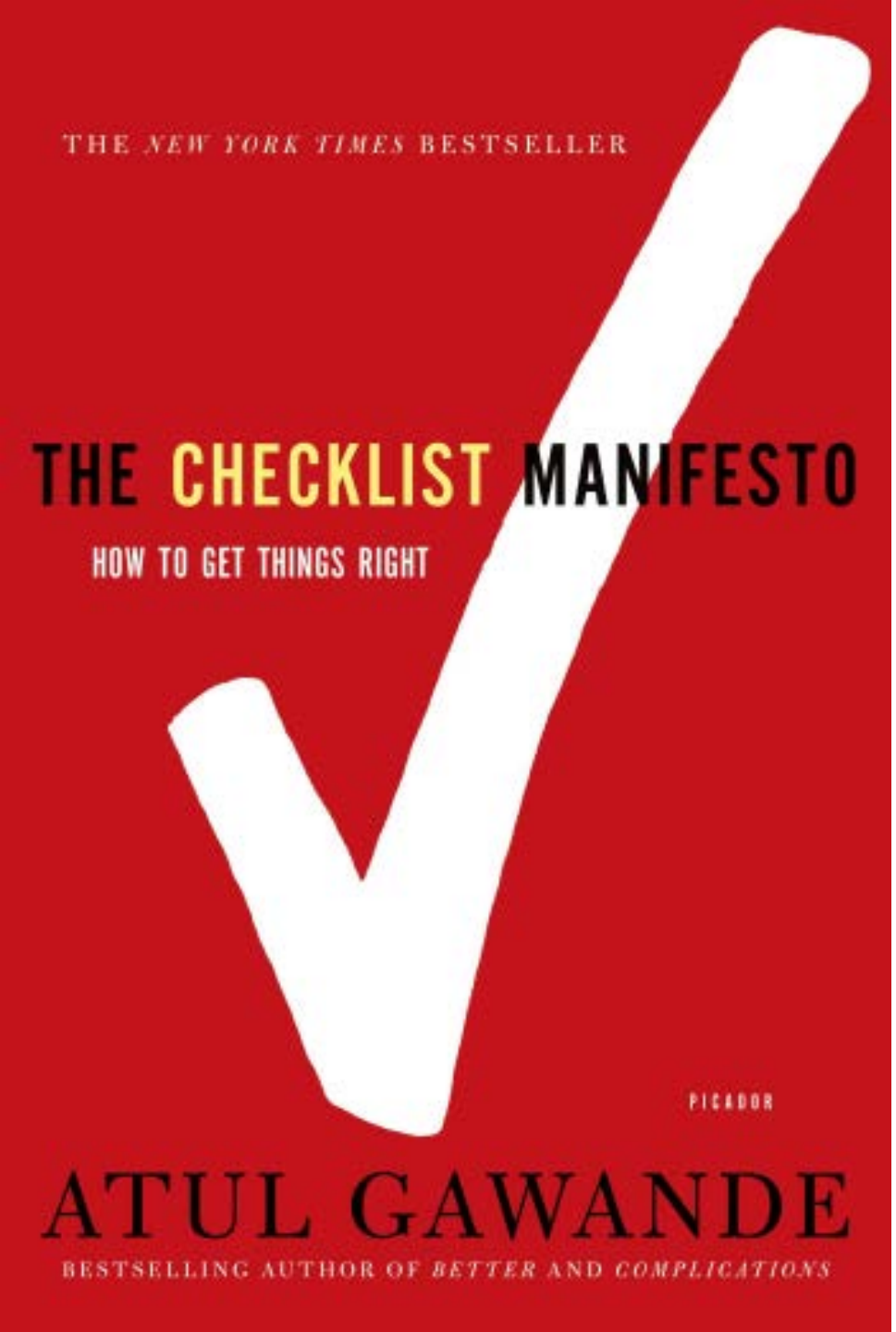
Dramatic Results

10 day line-infection rate
Fell from 11% to 0%

Checklist prevented:
43 infections
8 deaths
$2 million



Doctors know what to do,
but it's easy to overlook a step.



Developers

~~Doctors~~ know what to do,
but it's easy to overlook a step.

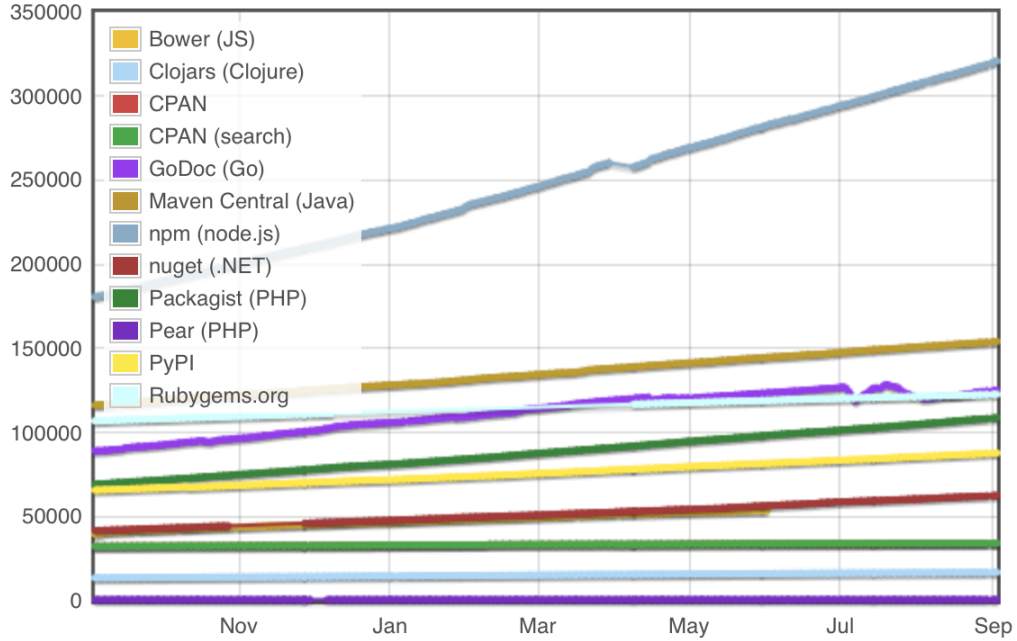
A starter kit is a living, automated, and interactive checklist.

Who Is This Course For?

"Any application that *can* be written in JavaScript, *will* eventually be written in JavaScript."

Atwood's Law

Module Counts



Include

- [x] Bower (JS)
- [x] Clojars (Clojure)
- [x] CPAN
- [x] CPAN (search)
- [ ] CRAN (R)
- [ ] Crates.io (Rust)
- [ ] Drupal (php)
- [ ] DUB (dlang)
- [x] GoDoc (Go)
- [ ] Hackage (Haskell)
- [ ] Hex.pm (Elixir/Erlang)
- [ ] LuaRocks (Lua)
- [x] Maven Central (Java)
- [ ] MELPA (Emacs)
- [x] npm (node.js)
- [x] nuget (.NET)
- [x] Packagist (PHP)
- [x] Pear (PHP)
- [ ] Perl 6 Ecosystem (perl 6)
- [x] PyPI
- [x] Rubygems.org

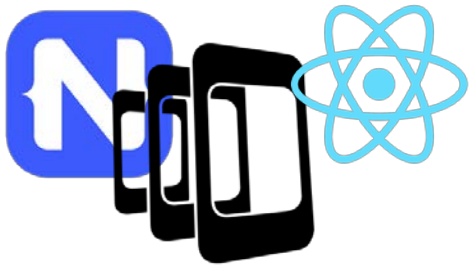
time period ○ all time ◉ last year ○ last 90 days ○ last 30 days ○ last 7 days

	Aug 27	Aug 28	Aug 29	Aug 30	Aug 31	Sep 1	Sep 2	Avg Growth
Bower (JS)								37/day
Clojars (Clojure)	16764	16769	16789		16821	16828	16837	12/day
CPAN	34109	34107	34108		34116	34119	34123	2/day
CPAN (search)	34109	34107	34108		34116	34120	34125	3/day
CRAN (R)	9034	9035	9044		9062	9069	9078	7/day
Crates.io (Rust)	5851	5862	5880		5902	5912	5920	11/day
Drupal (php)	35258	35261	35265		35281	35303	35316	10/day
DUB (dlang)	837	838	839		841	841	841	1/day
GoDoc (Go)	124396	124377	124497		124651	124969	125024	105/day
Hackage (Haskell)	10169	10171	10173		10177	10189	10191	4/day

JavaScript Is Everywhere

Web

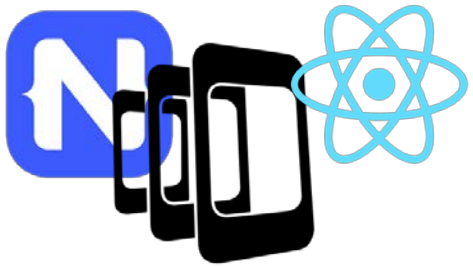
Server



Mobile

Desktop

This Course Is for JS Developers



Web

Server

Mobile

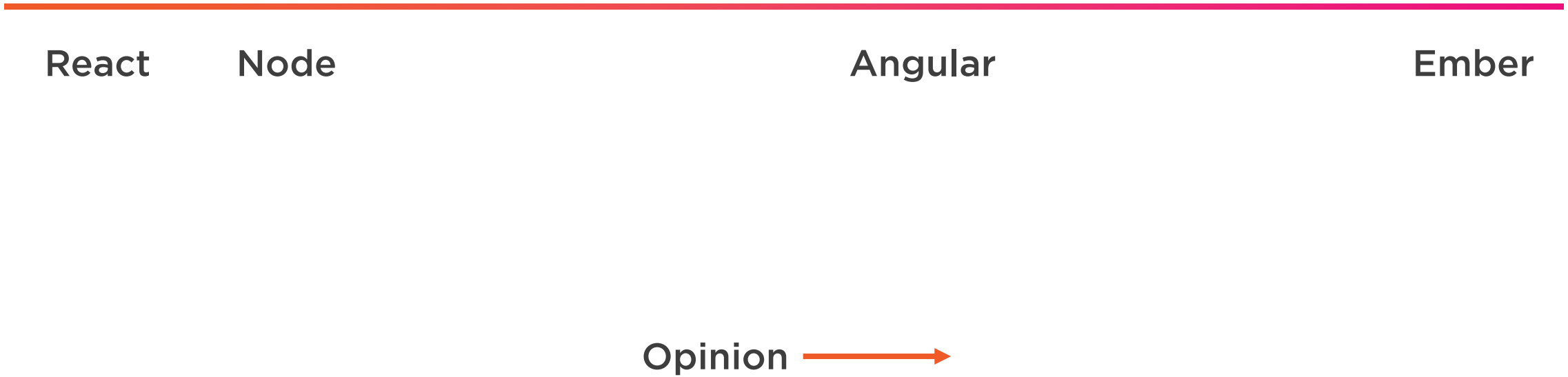
Desktop

Demo

Build tool & API

Who Doesn't Need This Course?

Highly opinionated and comprehensive frameworks


React
Node
Angular
Ember
Opinion

Boilerplate	Build your own
Save time	Freedom
Proven	Perfect fit for your team
Full-featured	No unnecessary complexity
	Understand it
	Easier to change

What Belongs in Your Starter Kit?

- Package Management
- Bundling
- Minification
- Sourcemaps
- Transpiling
- Dynamic HTML Generation
- Centralized HTTP
- Mock API framework
- Component libraries
- Development Webserver
- Linting
- Automated testing
- Continuous Integration
- Automated build
- Automated deployment
- Working example app

Here's the Plan...

GitHub

Why?

- Module 9: Continuous Integration
- Module 13: Automated Deploy

Git

Cory

https://git-scm.com

git --distributed-even-if-your-workflow-isnt

Search entire site...

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.

Learn Git in your browser for free with Try Git.

About

The advantages of Git compared to other source control systems.

Documentation

Command reference pages, Pro Git book content, videos and other material.

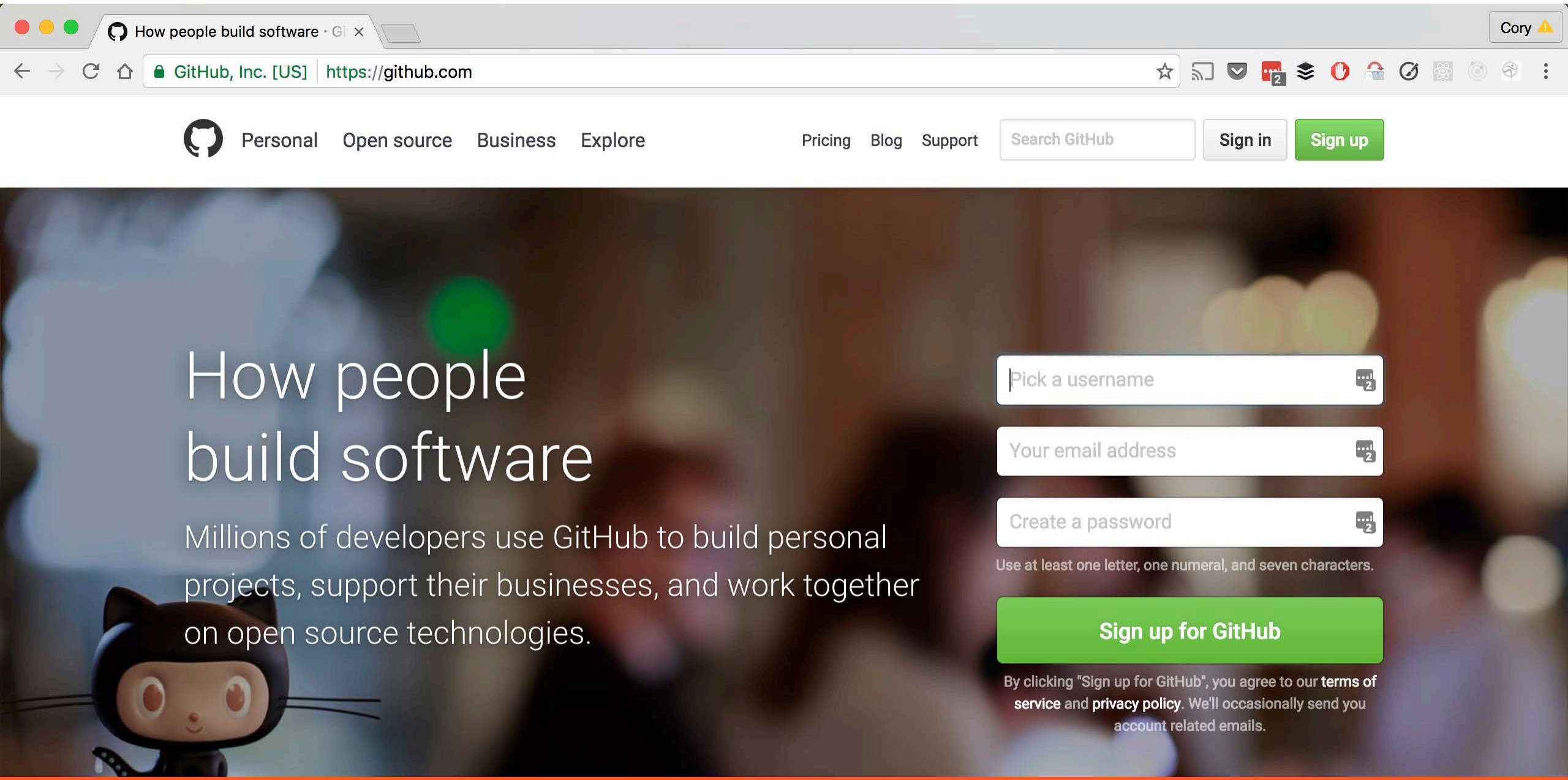
Latest source Release

2.10.1

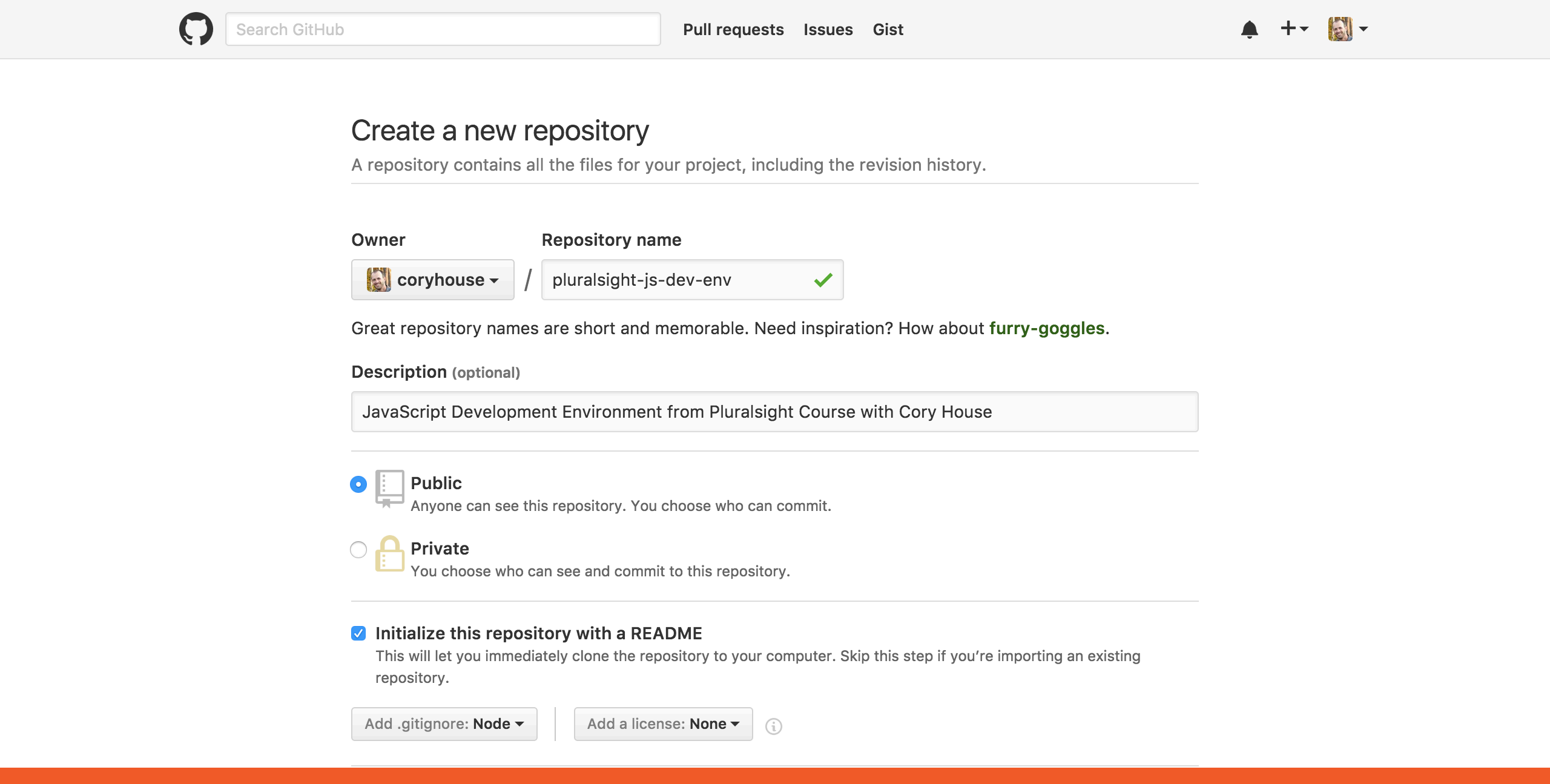
Release Notes (2016-10-03)

Downloads for Mac

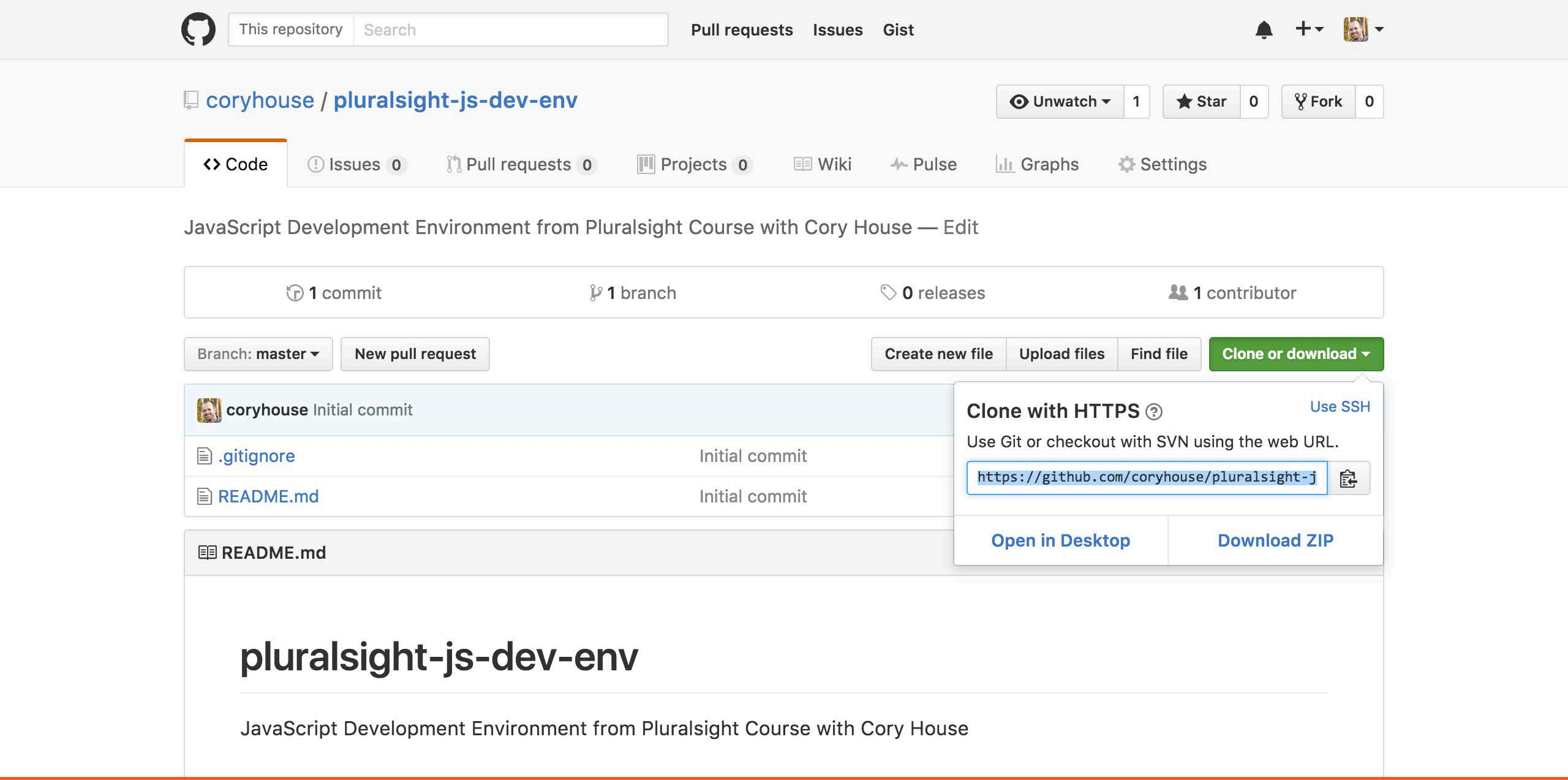
Step 1: Install Git



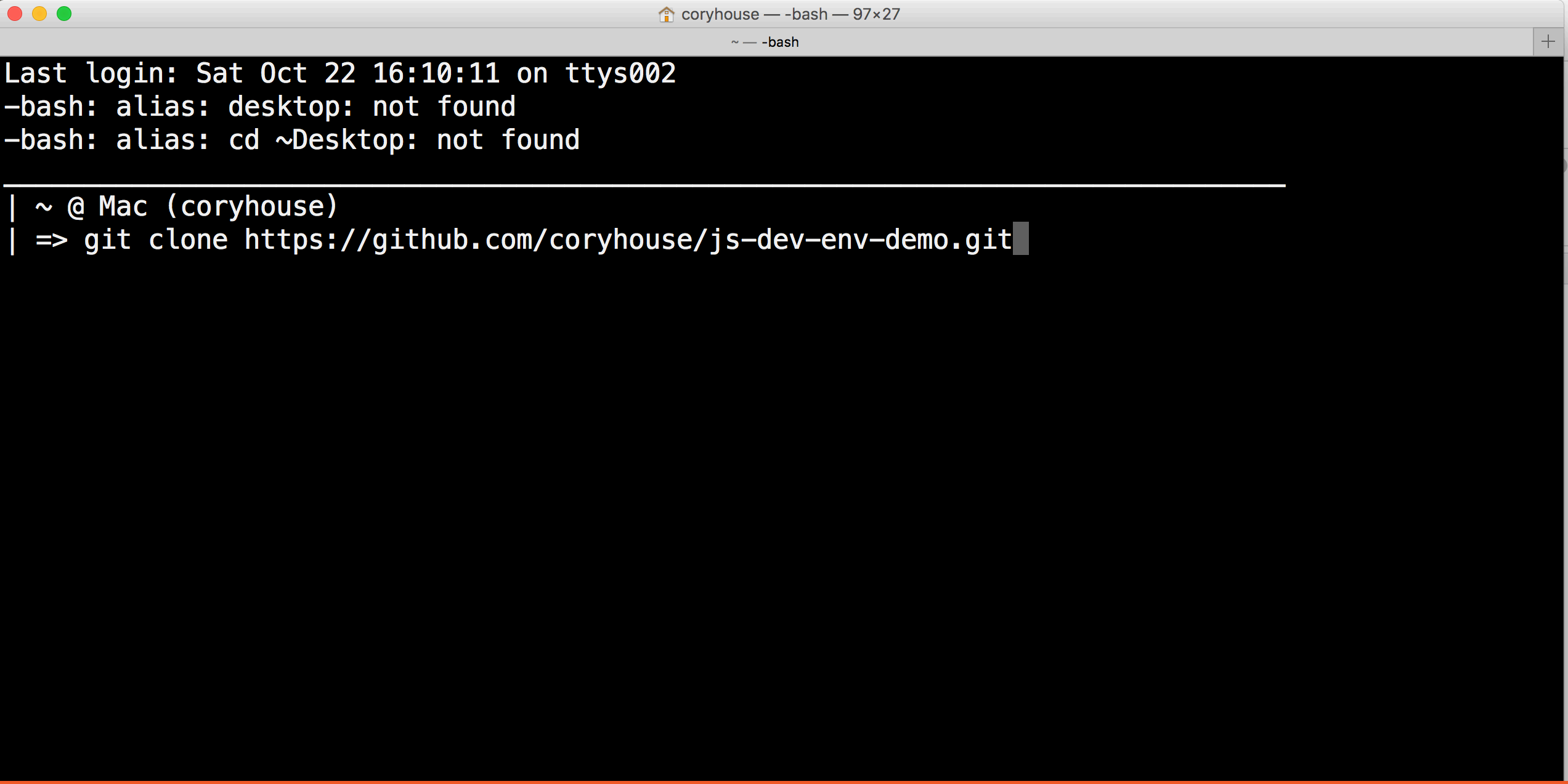
Step 2: Create GitHub Account



Step 3: Create GitHub Repository



Step 4: Clone Repository



Step 4: Clone Repository

1. Stage all the files you changed

```
git add .
```

2. Commit your changes locally

```
git commit -m “added new feature”
```

3. Push your work to Github

```
git push
```

Step 5: Enjoy Git!

Course Agenda

Synonyms

x == y

JavaScript Development Environment

Boilerplate

Starter Kit

Seed

Starter Project

The Rhythm

1

Options

2

Recommendation

3

Implement

Course Outline

1. Intro
2. Editors and configuration
3. Package management
4. Development webserver
5. Automation
6. Transpiling
7. Bundling
8. Linting
9. HTTP
10. Testing and CI
11. Project structure
12. Production build
13. Automated deployment

Wrap Up

You need a starter kit

- Reduce decision fatigue
- Right thing becomes automatic
- Codifies lessons learned
- Rapid feedback
- Automated checklist

Writing JS? This course is for you.

Next up: Editor and configuration