# Starter Kit

## What Belongs in My Starter Kit

* Package Management
* Bundling
* Minification
* Sourcemaps
* Transpiling
* Dynamic HTML Generation
* Centralized HTTP requests
* Mock API framework
* Component libraries
* Development Webserver
* Linting
* Automated testing
* Continuous Integration
* Automated build
* Automated deployment
* Working example app (to convey how everything comes together including directory structure and file-naming practices).

# Editors and Configuration

## What to Look For in a JavaScript Editor

* It should show strong support for the latest JS features (ES 2017), but shouldn’t be too large.
  + To index JS files to provide **autocompletion**.
  + To **parse ES6 imports**
  + To **report unused imports**, typos and import statements.
  + To offer **automated refactoring** tools (rename, extract…)
* Built-in support for popular frameworks and libraries (Node, React, Angular…) [WebStorm, VSCode]
* Built-in terminal

## JS Editor Recommendations

* Atom
  + Free
  + Plug-in for EditorConfig
* WebStorm
  + 30-day free trial
* Brackets
  + Free
  + Plug-in for EditorConfig
* VSCode
  + Free
  + Fast
  + Git integration
  + Built-in terminal
  + Node debugging
  + Plenty of plug-ins (including EditorConfig)

## Automated Consistency via EditorConfig

Create a file named .editorconfig in the root of your project. Specify how your editor handles common settings (tabs/spaces, indent size, line feeds, char sets, trailing white space…)