1. Tóm tắt các quy tắc

Tất cả các plugin video.js thông thường phải:

- Có npm packages

- Có tự động hoá thông qua npm scripts

- Được viết bằng ES6 và pass videojs-standard linting

- Có bài test

2. Đóng gói và các phần phụ thuộc

Tất cả các video.js plugin thông thường phải là npm packages

Các plugin phải càng khép kín càng tốt, do đó, phần phụ thuộc duy nhất theo mặc định là video.js và global - cung cấp khả năng truy cập vào các biến global.

Các phần phụ thuộc (Development dependencies) sẽ bao gồm nhiều gói liên quan đến: developing, building và testing 1 plugin video.js.

3. Kiến trúc 1 plugin

| **Folder/Filename** | **Optional** | **Generated** | **Description** |
| --- | --- | --- | --- |
| dist/ |  |  | Build output cho trình duyệt/ script, sẽ bị bỏ qua bởi Git |
| docs/ | ✓ |  | Bất kỳ documentation nào ngoài README.md. |
| lang/ | ✓ | ? | Chứa các tệp ngôn ngữ JSON cho plugin |
| scripts/ |  | ✓ | Script được sử dụng bởi tooling, không phải là 1 phần của source code |
| src/ |  | ✓ | All source code. |
| src/js/ | ✓ |  | JavaScript source code. |
| src/plugin.js |  | ✓ | Plugin entry point. |
| test/ |  | ✓ | Unit tests. |
| scripts/karma.conf.js |  | ✓ | Karma configuration file. |
| test/plugin.test.js |  | ✓ | Default plugin test module. |
| .editorconfig |  | ✓ |  |
| .gitignore |  | ✓ |  |
| .npmignore |  | ✓ |  |
| .nvmrc |  | ✓ |  |
| CHANGELOG.md | ✓ | ✓ |  |
| CONTRIBUTING.md | ✓ | ✓ | Tài liệu về cách các nhà phát triển có thể làm việc với plugin |
| index.html | ✓ | ✓ | Một ví dụ về cách sử dụng plugin |
| LICENSE | ✓ | ? | Defaults to MIT. |
| package.json |  | ✓ |  |
| README.md |  | ✓ | Ghi lại các chức năng của plugin và cách sử dụng plugin với tư cách là người sử dụng thông thường. |

Optional: file/directory không bắt buộc phải đáp ứng các quy tắc trong tài liệu này.

Generated?

Khi chúng ta chạy generator nhiều lần, generator sẽ nhắc ta về các tệp xung đột tồn tại trước khi ghi đè

- Dấu tích: file/directory luôn được tạo bởi generator

- Dấu hỏi: file/directory có thể được tạo, phụ thuộc vào options

- Không có tích: file sẽ không baoh được tạo

4. Tự động hoá

Các tính năng tự động hoá cho 1 plugin video.js thông thường phải có sẵn thông qua npm scripts.

Generator cung cấp script automation cho tất cả các khía cạnh của plugin development. Không cần công cụ xây dựng của bên thứ ba.

Tuy nhiên, 1 số developers có thể thích triển khai bằng cách sử dụng các tool như Gulp hoặc Grunt và đây là 1 phương pháp được chấp nhận với điều kiện tập lợp các core npm script được đặt bí danh (Alias) đại diện cho các script command mà ta dùng để build tool. Ví dụ, nếu ta sử dụng grunt, file package.json phải như thế này:

"scripts": {

      "build": "grunt",

      "build:js": "grunt js"

    }

Có nhiều lý do để ta chuẩn hoá các npm script:

- npm cung cấp 1 giao diện thống nhất trong khi vẫn giữ các lựa chọn triển khai tuỳ thuộc vào từng nhà phát triển

- npm quá phổ biến rồi (phổ biến hơn nhiều so với các build tool). Vì vậy, CI Servers, contributors và các tool khác có thể sử dụng npm dễ dàng mà không cần lo lắng về các tool phía sau nó.

5. npm Core Scripts

| **npm Script** | **Optional** | **Description** |
| --- | --- | --- |
| build |  | Runs all build sub-tasks. |
| clean |  | Cleans up all build artifacts. |
| docs | ✓ | Performs documentation tasks. |
| lint |  | Lints all .js ES6 source file(s) using videojs-standard. |
| start |  | Starts a development server at port 9999 with automatic background builds (see watch). |
| test |  | Runs lint, builds tests, and runs tests in available browsers. |
| watch |  | Watches .js and (optional) .scss sources and rebuilds on demand. |

6. Coding style

Tất cả các video.js plugin phải pass videojs-standard linting

7. Testing

Tất cả các plugin video.js phải có phần test

Testing là 1 yếu tố quan trọng của bất kỳ dự án phần mềm nào và nó phải được thực hiện trong một môi trường càng giống với môi trường triển khai trong thực tế càng tốt. Vì vậy, các video.js plugin test cần được chạy trong trình duyệt.

Quá trình kiểm thử được thực hiện bởi QUnit và Karma: QUnit là testing framework, Karma là runner

7.1. Viết 1 script test

Tất cả các scripts chứa các bài test phải được đặt tên theo định dạng: \*.test.js

Entry point của test-generator là test/plugin.test.js (Khớp với src/plugin.js). Đối với các plugin đơn giản, điều này là đủ.