1. Giới thiệu

Như đã giới thiệu sơ qua về thuật ngữ *code coverage* ở phần trước (Istanbul), ở phần này sẽ nói rõ hơn về thuật ngữ này cũng như công cụ để báo cáo mức độ bao phủ mã nguồn (codecov.io)

* 1. Code coverage

Là một sự đo lường được dùng để cho thấy những dòng nào trong mã nguồn được thực thi bởi bộ mô tả kiểm thử (test suite). Có ba thuật ngữ được dùng để mô tả sự thực thi của mỗi dòng:

* Hit: cho biết đoạn mã được thực thi
* Partial: cho biết đoạn mã không hoàn toàn được thực thi và vẫn còn nhánh (branch) không được thực thi
* Miss: cho biết đoạn mã không được thực thi

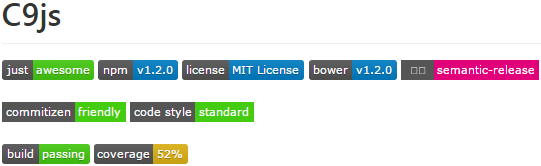
Độ bao phủ được tính bằng công thức: *total hits / (total of hit + partial + miss)*.

Theo thuật ngữ của Layman: code coverage cung cấp sự đo lường trực quan của mã nguồn được thực thi bởi bộ kiểm thử. Thông tin này cho lập trình viên biết họ cần viết những test mới ở đâu để đạt được mức độ bao phủ cao hơn.

Code coverage thúc đẩy lập trình viên tăng mức độ bao phủ, và có thể tìm ra các lỗi mới, các vấn đề về cú pháp trong mã nguồn trong quá trình phát triển, để họ có thể giải quyết trước khi đưa ứng dụng ra cộng đồng.

* 1. Codecov.io

Là công cụ hỗ trợ tạo báo cáo về mức độ bao phủ mã nguồn của codecov.io và có thêm một nút *coverage* trong README.



Codecov.io đem vấn đề *bao phủ mã nguồn* lên một tầm cao mới. Không giống như những công cụ coverage khác, các công cụ mã nguồn mở hay tính phí. Codecov tập trung vào vấn đề tích hợp (integration) và thúc đẩy các pull request (các yêu cầu được ghép mã vào nhánh chính – master) một cách lành mạnh. Công cụ Codecov cho phép các số liệu bao phủ (coverage) được thể hiện trực tiếp trong quá trình làm việc (workflow) để thúc đẩy nhiều hơn việc tăng cao mức độ bao phủ, đặc biệt là ở các pull request, nơi mà ở trong những quy trình làm việc hiện đại ngày nay, các tính năng mới và các vấn đề sửa lỗi xảy ra phổ biến.

1. Tính năng

Những tính năng mạnh mẽ của codecov.io:

* Browser Extension: hỗ trợ xuất báo cáo với các đo đạc, thông số trên trình duyệt. Dễ dàng theo dõi và có cái nhìn tổng quan
* Auto-Merging Builds: hỗ trợ hầu hết các ngôn ngữ, kết hợp với tích hợp liên tục và xây dựng trong một báo cáo
* Notifications: gửi thông báo qua nhiều kênh giao tiếp như Slack, Gitter, Hipchat, hỗ trợ tạo Webhook,…
* Coverage Compare: so sánh sự khác biệt trong sự thay đổi mã nguồn nhờ *git diff*
* Pull Request Comments: là một bản mô tả chi tiết của một pull request. Nó cung cấp thông tin chi tiết về các thay đổi trong sự bao phủ (coverage) của pull request để giúp tăng nhanh tốc độ xem lại mã nguồn và đảm bảo các pull request đều đã được kiểm tra
* Commit Status: trạng thái dự án trong thư mục codecov sẽ đo đạc độ bao phủ của dự án đó và so sánh nó với pull request hoặc commit cha (parent commit)

1. Ứng dụng

C9js sử dụng codecov.io để cho biết mức độ bao phủ của mã nguồn thư viện trực tiếp trên các kênh như github và codecov.io:

<https://github.com/csethanhcong/C9js> (README)

<https://codecov.io/gh/csethanhcong/C9js>

Qua các báo cáo mà codecov.io đem lại như tính chính xác của mã nguồn (thông qua các test đã đạt hay không đạt), số lượng các câu lệnh, dòng, nhánh, hàm đã được bao phủ (cover),… đã góp phần làm cho mã nguồn thêm sự ổn định và tin cậy.