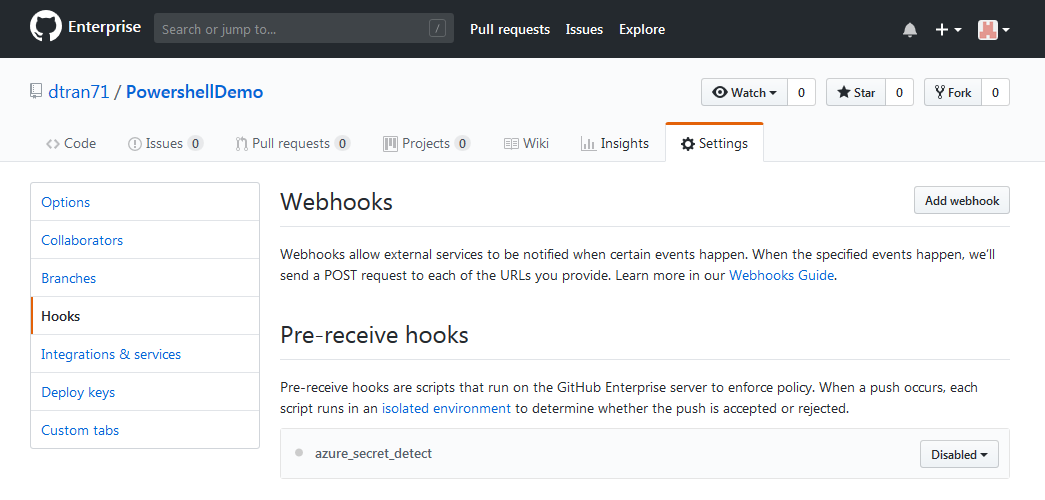
Sau khi thiết lập Azure VM agent để tạo máy từ xa , chúng em sẽ bắt đầu tạo build trên Jenkins cho việc triển khai code. Build khi chạy sẽ kích hoạt plugin Azure VM agent để tạo một máy ảo trên tài khoản Azure đã được cấu hình sẵn và sẽ chạy code và các tùy chỉnh của Build trên máy ảo ấy. Để có được code chạy cho build , chúng em làm việc đó thông qua tới một thứ có thể giúp cho Jenkins đi tới Github để có thể kéo các code nguồn về để chạy build , thứ đó gọi là Github webhook.

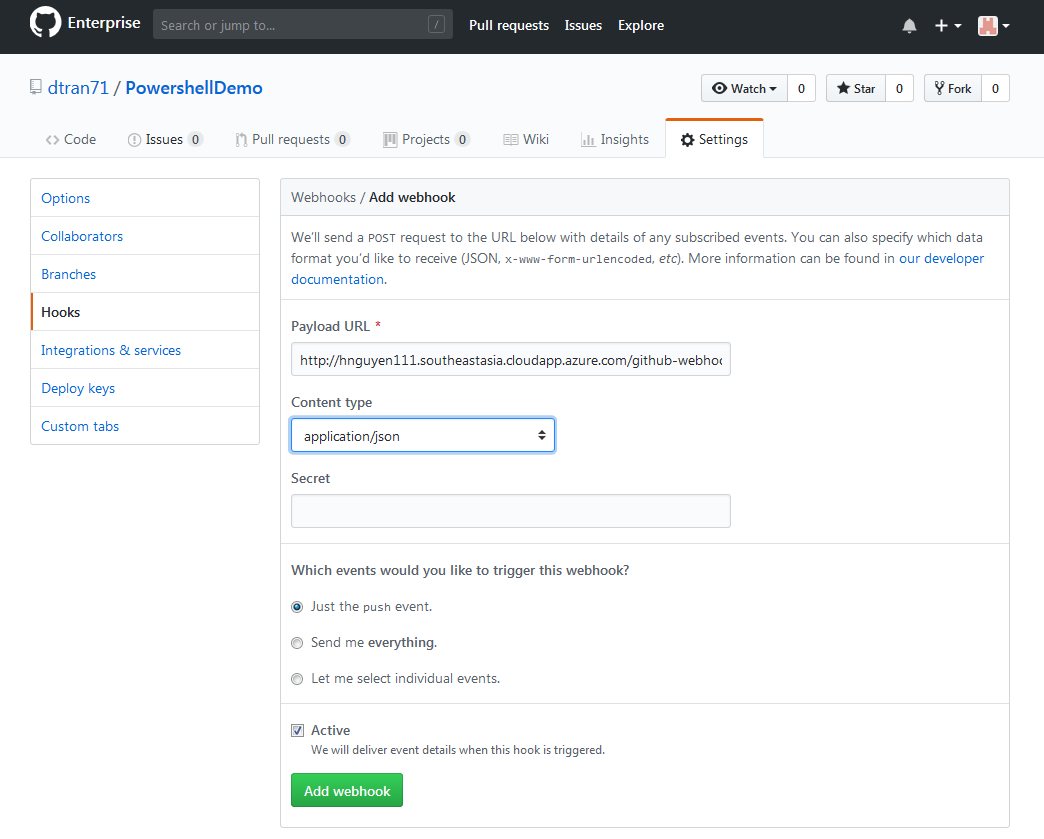
Github webhook hoạt động như sau : Khi có một sự kiện nào đó xảy ra trên kho lưu trữ từ xa ( Merge , push , fork ,.. ) , Github webhook sẽ gửi một payload HTTPS POST đến địa chỉ URL đã được cấu hình trên WebHook. Việc này sẽ giúp kích hoạt build của Jenkins dựa trên Github. Github webhook có thể được cấu hình như sau :

Github Webhook nằm ở mục hook trong Setting của kho lưu trữ từ xa trên giao diện web :



Hình 2.?? Github Webhook

Khi nhấp vào mục “Add webhook” , Github sẽ đưa người dùng tới trang cấu hình webhook cho kho lưu trữ. Trang này sẽ cho chúng ta cấu hình nhiều nhứ cho webhook như địa chỉ URL mà webhook sẽ gửi Payload , Secret token để tăng cường bảo mật , loại event để kích hoạt webhook,..

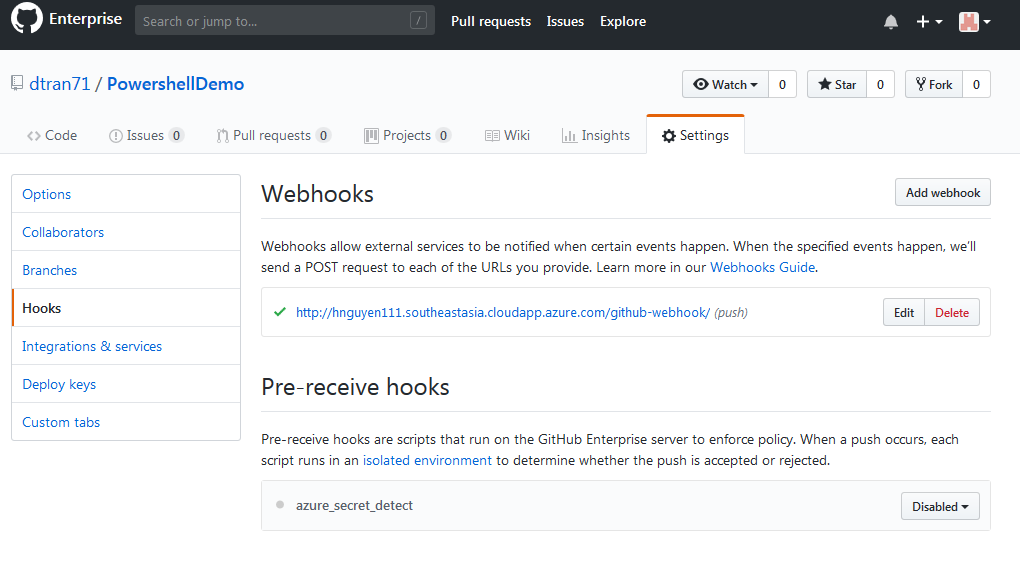


Hình 2.?? Cấu hình Github webhook

Trong đó :

* **Payload URL :** Đây là địa chỉ URL mà Github webhook sẽ gửi payload tới . Chúng em sẽ để URL của Jenkins.
* **Content Type :** Kiểu thông điệp của payload . Github webhook hỗ trợ application/x-www-form-urlencoded và application/json.
* **Secret :** Người dùng có thể nhập vào đây một đoạn chữ để có thể làm secret token cho webhook .
* **Which events would you like to trigger this webhook ?:** Loại sự kiện xảy ra trên kho lưu trữ để kích hoạt webhook. Có rất nhiều loại sự kiện có thể dùng để kích hoạt webhook nhưng chúng em tập trung vào chỉ sự kiện push.
* Mục cuối cùng sẽ cho phép người dùng khởi động hay ngưng webhook tùy ý muốn.

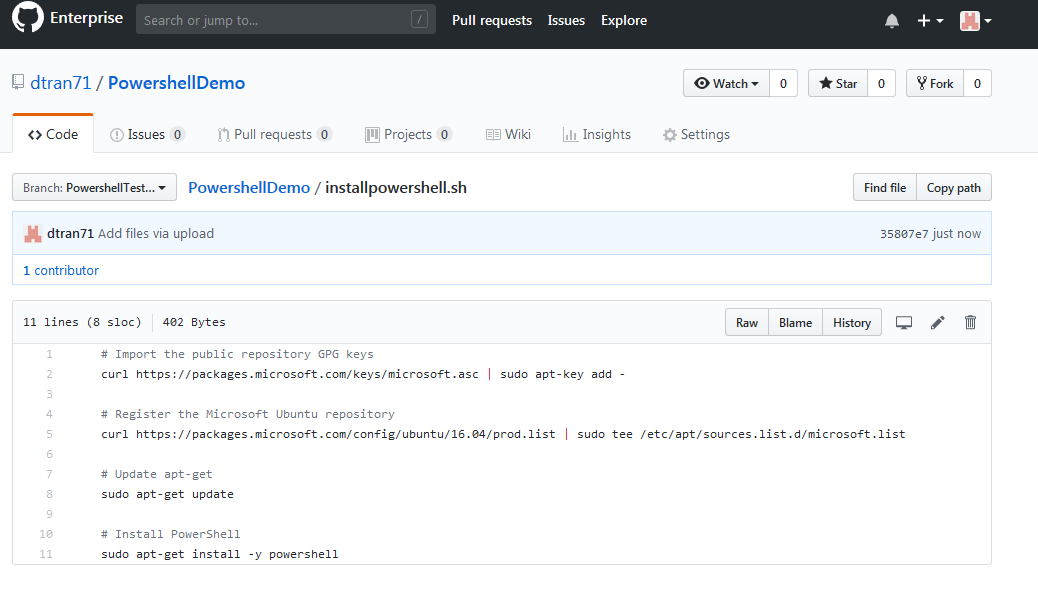
Sau khi nhấn Add webhook , webhook mới được tạo sẽ xuất hiện ở trang Hook.



Hình 2.??? Webhook đã được tạo

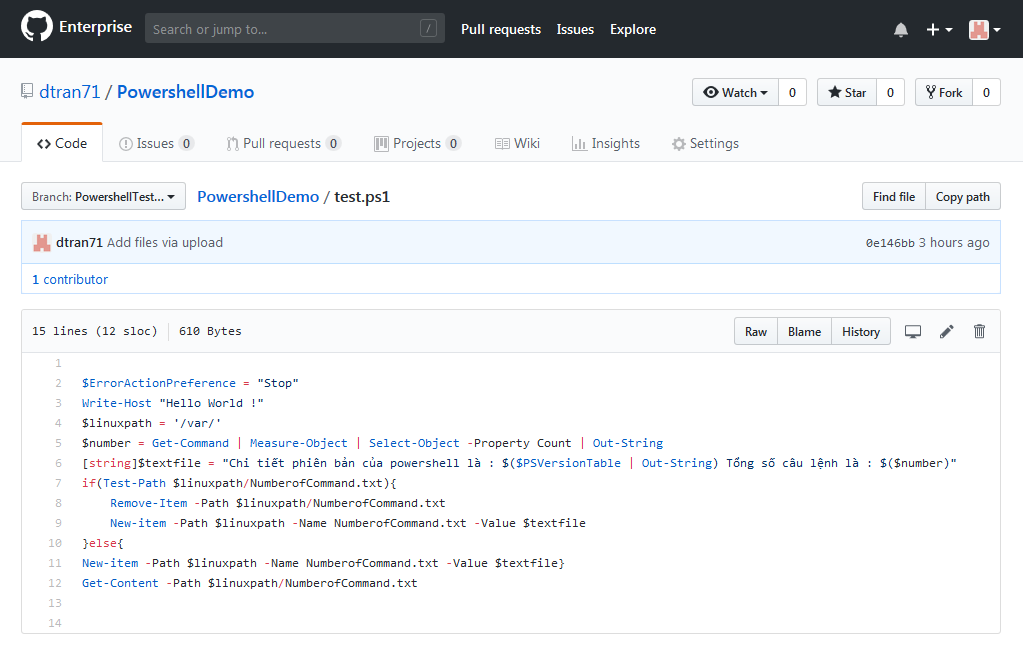
Việc cấu hình build sẽ hoàn toàn dựa vào người dùng sẽ chạy code gì trên build đó. Ở đây chúng em sẽ chạy một đoạn code powershell trên máy ảo với hệ điều hành Ubuntu 16.04 đã được cấu hình trên Azure VM agent plugin. Build của chúng em sẽ tập trung vào 2 file chính có trên kho lưu trữ PowershellDemo nhánh PowershellTestBranch đó là installpowershell.sh và test.ps1.

* **installpowershell.sh :** Không như hệ điều hành Window , các hệ điều hành linux như Ubuntu không có sẵn powershell để chạy code powershell. Thay vào đó , người dùng phải cài đặt thủ công powershell qua các câu lệnh. installpowershell.sh là một file shell script có nhiệm vụ là cài đặt powershell cho máy ảo của Azure VM agent thay vì phải cài đặt thủ công powershell trên máy ảo.



Hình 2.?? : Đoạn code của installpowershell.sh

* **test.ps1 :** test.ps1 là file script được viết bằng ngôn ngữ powershell. Đuôi ps1 là mặc định cho các file script của powershell . Script của test.ps1 được chạy như sau : Trước tiên nó sẽ viết ra dòng chữ “Hello World !” và lấy thông tin về chi tiết phiên bản powershell hiện cùng với tổng số câu lệnh , lưu chúng vào một biến dạng string , sau đó tạo một file mới ở dường dẫn định sẵn rồi cuối cùng là lấy thông tin của file đó.



Hình 2.?? : Đoạn code của test.ps1