

Configuring Jenkins with Github

OW

Prerequisites

- **Prerequisites**
- You must have a Github account and a repository on which we can test Jenkins.
- You must have a configured Jenkins Server. If you do not have, go and first [set-up Jenkins Server](#).

gi

Enter an item name

» Required field



Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



Multi-configuration project

OK

Use for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds,

Add scm

Repository URL ?

https://github.com/amitopenwriteup/my_proj.git

Credentials ?

amitopenwriteup/*****

+ Add

Advanced...

Add Repository

Branches to build ?

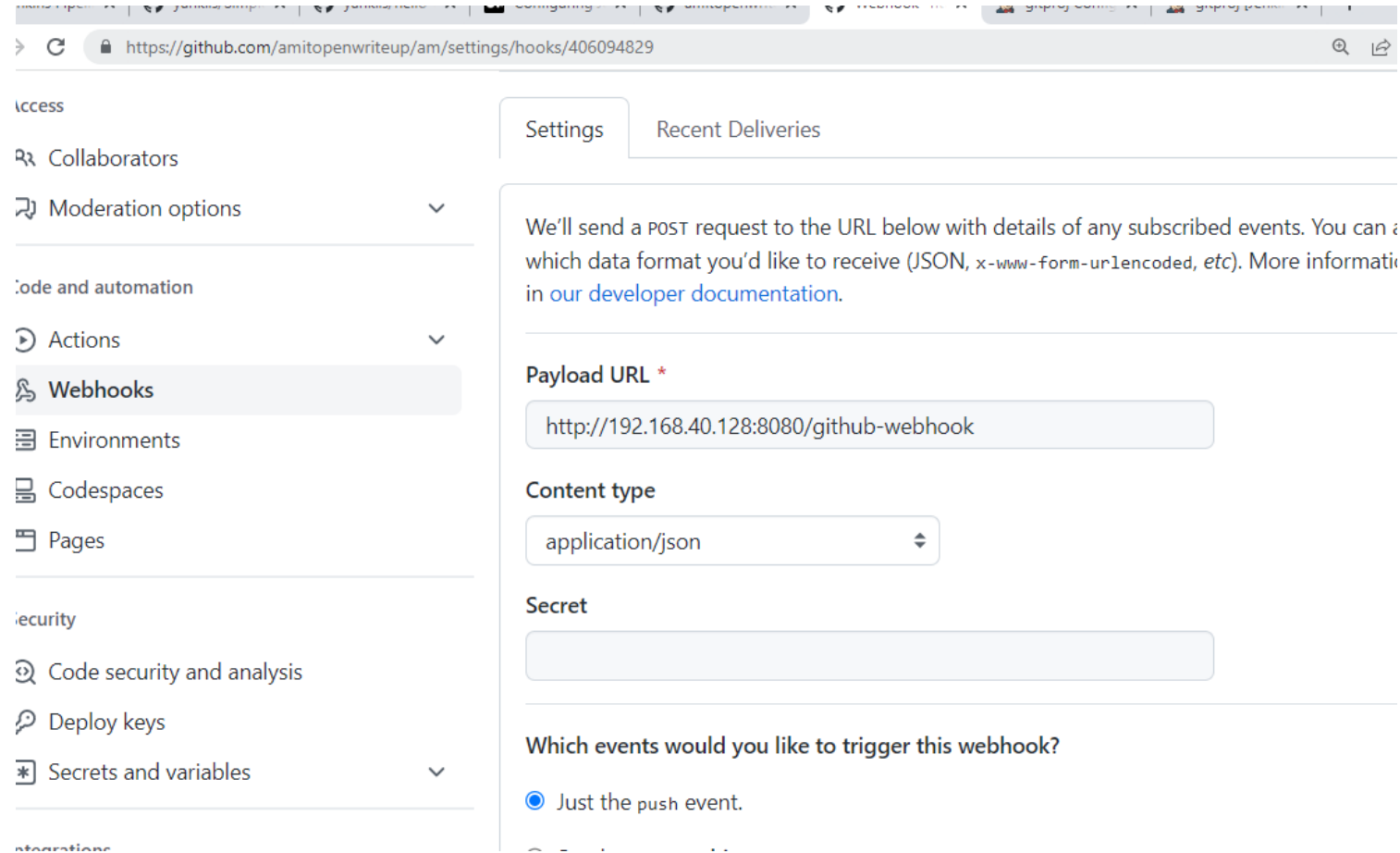
Build Triggers

- Trigger builds remotely (e.g., from scripts) ?
- Build after other projects are built ?
- Build periodically ?
- GitHub hook trigger for GITScm polling ?
- Poll SCM ?

Connecting your github repository with jenkins

- First, we have to setup our Github repository so that whenever we make a commit in any branch of this repository, a web hook will be triggered and tell our Jenkins server that its time to start a build.
 - Open your Github repository and click on Settings Tab.
 - Then click on Integrations & Services and click Add Service. Here start typing jenkins and install Jenkins plugin for Github.

Github add webhook



The screenshot shows the GitHub repository settings page for a user named 'amitopenwriteup'. The page is titled 'Hooks' and has two tabs: 'Settings' and 'Recent Deliveries'. The left sidebar contains a navigation menu with categories: 'Access', 'Code and automation', 'Security', and 'Integrations'. Under 'Code and automation', 'Webhooks' is selected. The main content area contains the following information:

We'll send a POST request to the URL below with details of any subscribed events. You can choose which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information in [our developer documentation](#).

Payload URL *

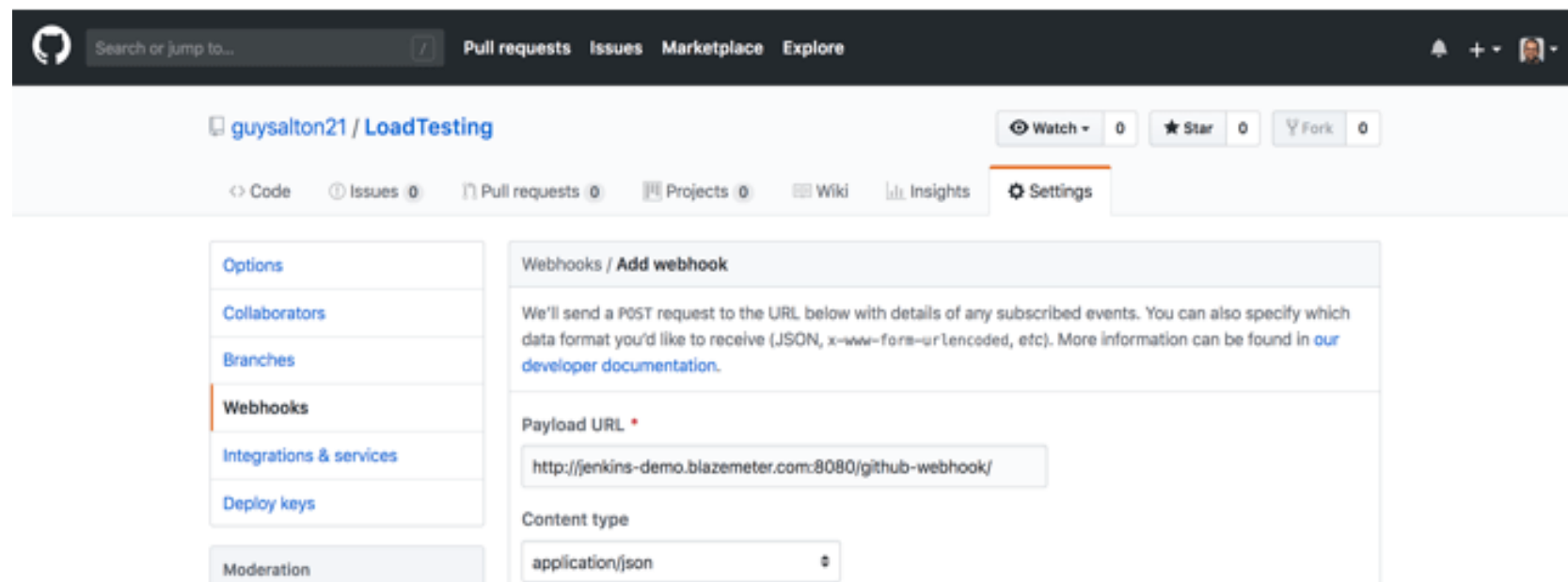
Content type

Secret

Which events would you like to trigger this webhook?

Just the push event.

Step 3: In the **'Payload URL'** field, paste your Jenkins environment URL. At the end of this URL add `/github-webhook/`. In the **'Content type'** select: `'application/json'` and leave the **'Secret'** field empty.



The screenshot shows the GitHub interface for the repository 'guysalton21 / LoadTesting'. The 'Settings' tab is selected, and the 'Webhooks / Add webhook' section is active. The 'Payload URL' field contains the text 'http://jenkins-demo.blazemeter.com:8080/github-webhook/'. The 'Content type' dropdown menu is set to 'application/json'. The left sidebar contains navigation links for Options, Collaborators, Branches, Webhooks, Integrations & services, Deploy keys, and Moderation. The top navigation bar includes a search bar and links for Pull requests, Issues, Marketplace, and Explore.

Trigger

Content type

Secret

Which events would you like to trigger this webhook?

- Just the `push` event.
 - Send me **everything**.
 - Let me select individual events.
-