

# Build a Serverless Github Bot in GCP

Franklin E. Diaz

[frank378@gmail.com](mailto:frank378@gmail.com)

October 3, 2022

## Abstract

Did you ever wonder how the cool kids get their bots going to manage pull requests in Github? The bots that can comment on Pull Requests, label things, perform other actions that are helpful to human developers? Well so did I, so I assembled one a while back that I want to share with you.

## 1 Build a Serverless Github Bot in GCP

This workshop is meant to be a fun way to learn more about some modern software development technologies. You don't need to have a deep understanding of all parts. Rather, you can follow along from end to end and choose to focus more deeply on any part that holds your attention.

A high level overview of the learning path is as follows:

1. Review the Python source for the bot.
2. Configure Google Cloud account, service user, etc.
3. Set up GitHub with a bot account, dev token, and webhook.
4. Configure Terraform and deploy the bot.
5. Test it out.
6. Explore possibilities for extending the functionality.

## 2 Preparation

There are some requirements that must be met to successfully complete this workshop.

### 2.1 Google Cloud Account

## 3 GNU Autotools

GNU Autotools have been included in the code base for the workshop. Autotools are a well-maintained set of Open Source tools with a gentle learning curve and are included in the distribution of many Open Source packages that we all rely on daily, at least indirectly, and often unknowingly.

The list of tools for using this Autotools configuration paradigm is show in Table ??.

Tool	Description
autoconf	Generates a configure script from configure.ac
automake	Generates a system-specific Makefile based on Makefile.am template
make	X

Table 1: Tools used in this project

The image shown in Figure ?? is from [?]. It illustrates the relationship between components mentioned in Table ??.

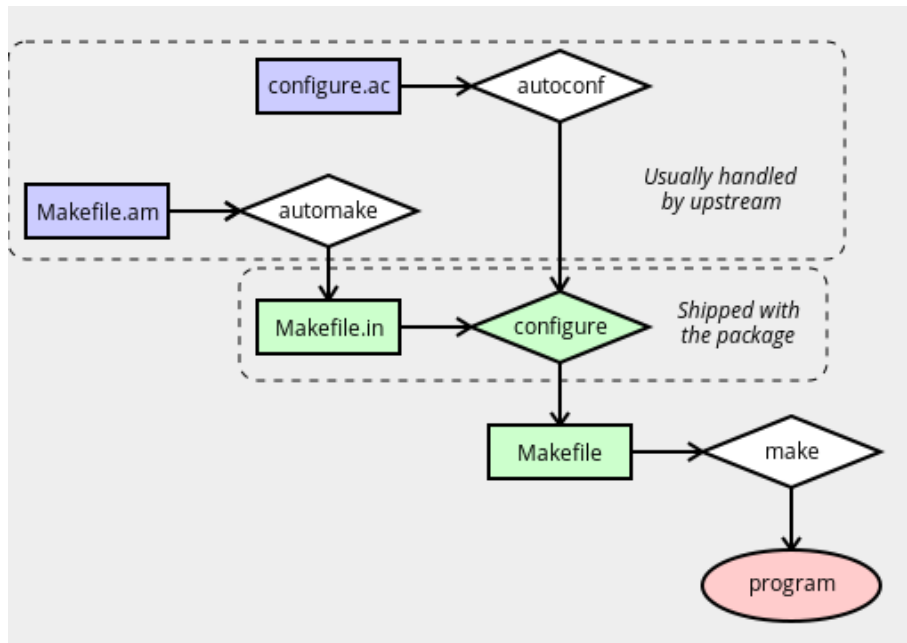


Figure 1: A basic overview of how the main Autotools components fit together.

## 4 Python

The Python code is meant to run as a “Cloud Function” in GCP. This is a cost-effective way to run code without deploying cloud infrastructure such as dedicated host instances or other cloud infrastructure that must be maintained. We get to run our code and let Google take care of the rest.

## 5 Terraform

## Revision History

Revision	Date	Author(s)	Description
v0.1	October 2nd, 2022	Franklin Diaz	Initial Draft