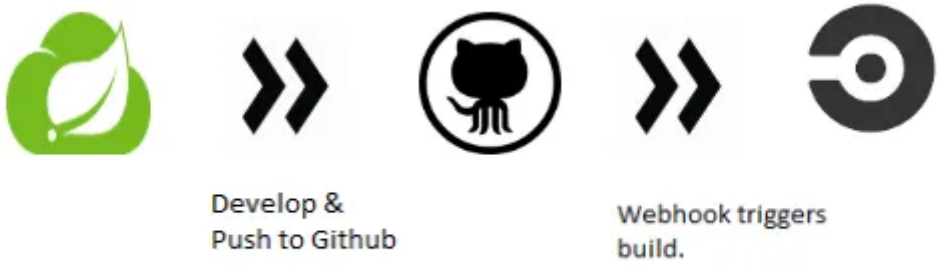
How to enable CircleCI builds for a Spring Boot Application

Xebia Engineers · Follow
Published in Xebia Engineering Blog
4 min read · Sep 12, 2019

By Amit Verma, Himank Batra, and Sugandha Sapra



In this post, we will look at integrating Spring Boot Application with CircleCI.

CircleCI is the continuous integration and delivery platform for software builds, tests & deployments. Under the hood, it uses Docker containers to run CI builds. It provisions a fresh environment to build our project. Every push to GitHub (on the master branch) triggers a build on CircleCI. If the build fails, CircleCI notifies us via an email. We believe in keeping our builds green so build failure emails helps us to be on top of any build failure.

Prerequisite :

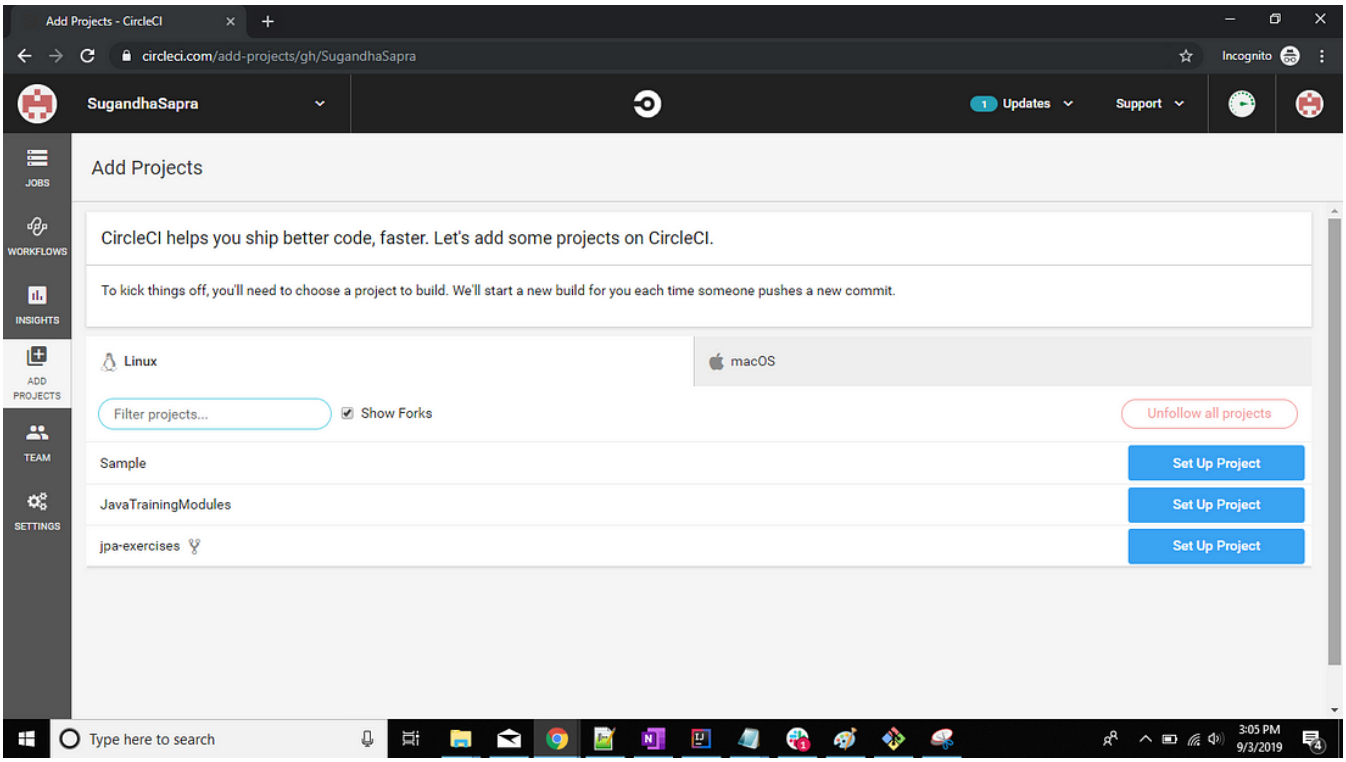
We assume you already have a Spring Boot application. In case you don't have one, we request you to use this link.

Also, create an account on CircleCI if you do not have it already. Signing up for CircleCI is quite easy. All you need is a GitHub, BitBucket, or a Google account.

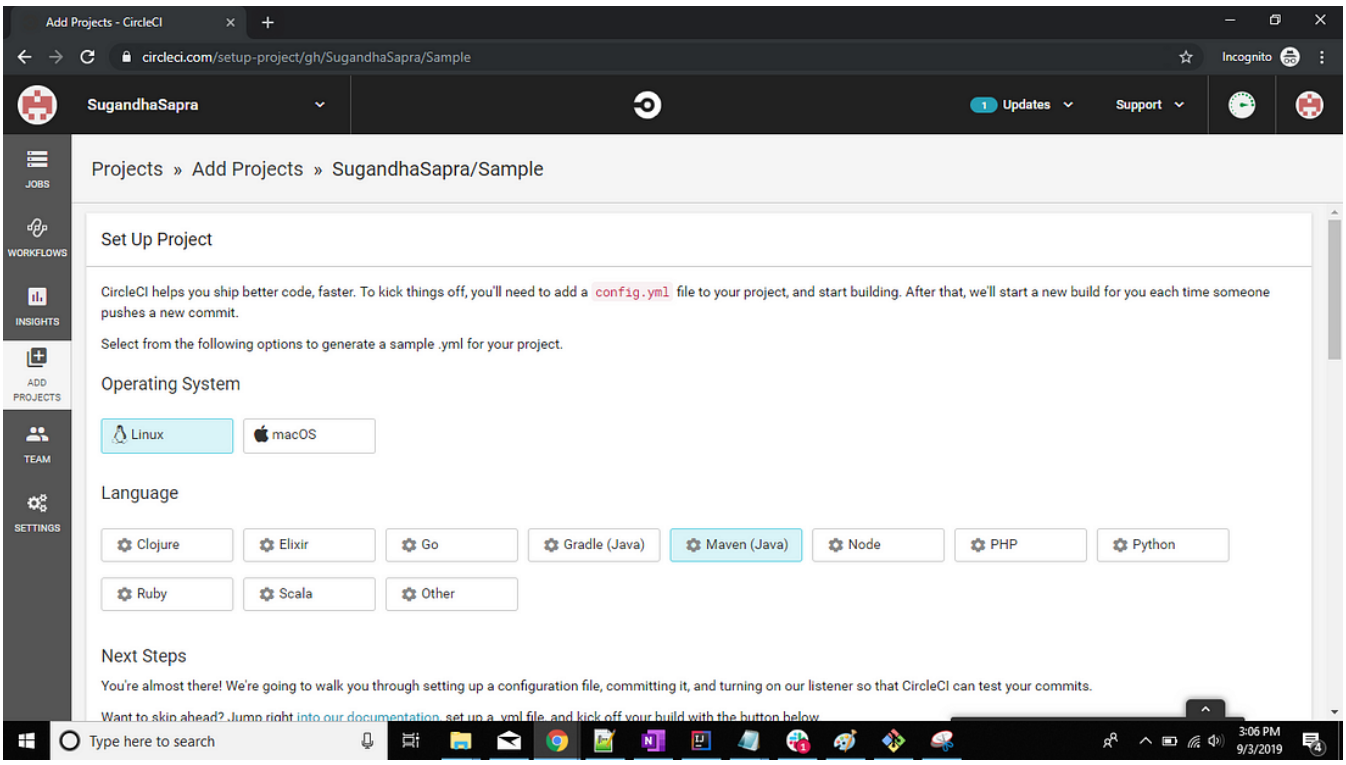
Configuring CircleCI to build project:

This application is a basic example of the Spring Boot project generated through Spring initializer. The focus of this post is on building the application using CircleCI rather than explaining you about Spring Boot.

- After you log in, navigate to Add Projects tab on the left-hand side and set up the project. To illustrate this, the ***Sample*** java project has being used below.
- Click on the "Setup Project" to begin.



- CircleCI will detect the programming language used in your project.It will create a sample config.yml.The config.yml is the configuration file used by CircleCI to build your project.



- In the root directory of your local project, create a hidden folder called .circleci and inside that directory, initialize config.yml file.

```
mkdir .circleci # this will create a folder with name .circleci
touch .circleci/config.yml # this will create a blank config.yml in
folder.circleci
```

- For our sample application, we need the configurations mentioned below. This will cover the minimum requirements needed to build spring boot application but you may try other things as well.Comments are also provided in each lines for better understanding.

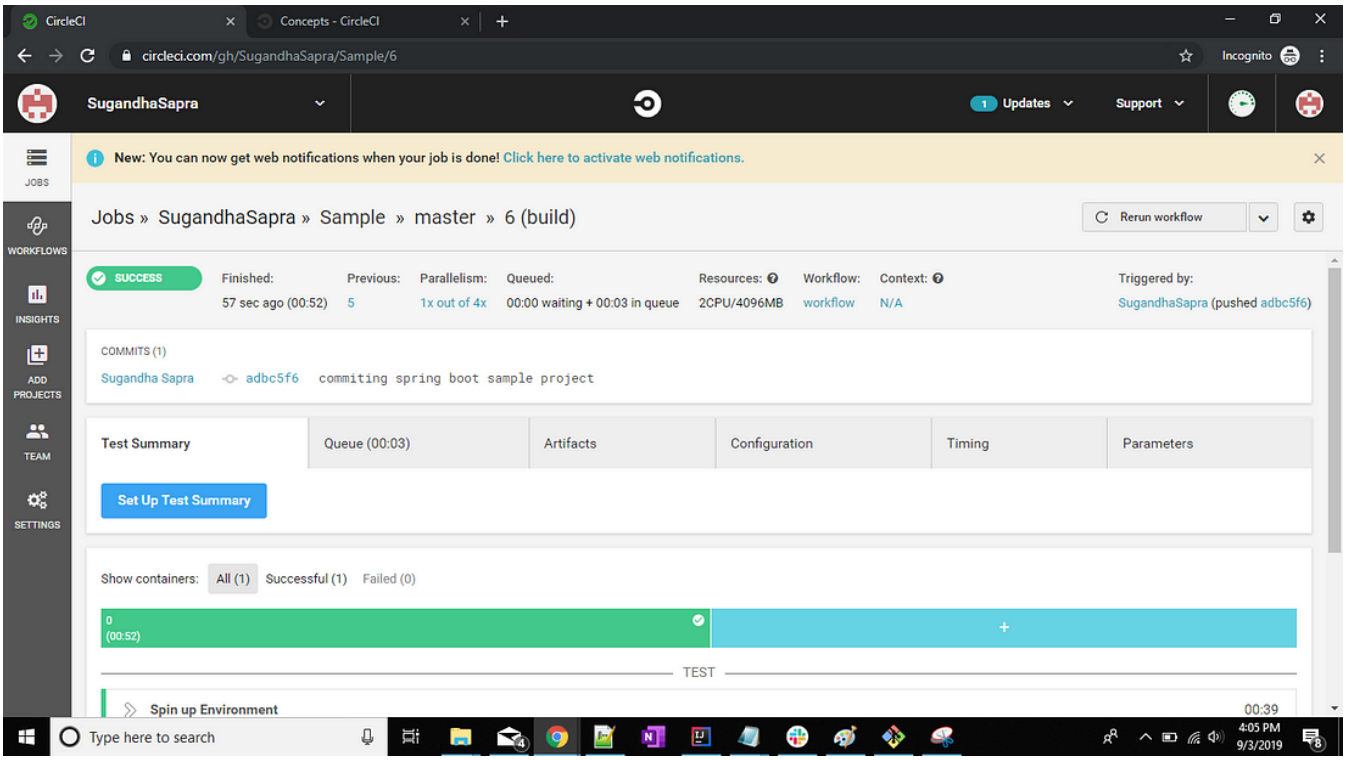
```
version: 2 # use CircleCI 2.0 #
jobs: # a collection of steps
 build: # runs not using Workflows must have a `build` job as entry
point

 docker: # run the steps with Docker
 – image: circleci/openjdk:8-jdk-stretch # …with this image as the
primary container; this is where all `steps` will run

 steps: # a collection of executable commands
 – checkout # Checkout the code from Github
 – run: ./mvnw clean install # mvn clean install tells Maven to do
```

```
the clean phase in each module before running the install phase for
each module.
```

- Go ahead and copy the above code snippet in the config.yml.
- Now that we've finished setting up the project, let's continue by pushing our .circleci folder to our git repo. This should automatically kick off a build in your CircleCI dashboard. You can use the following git commands to push to your git repo :

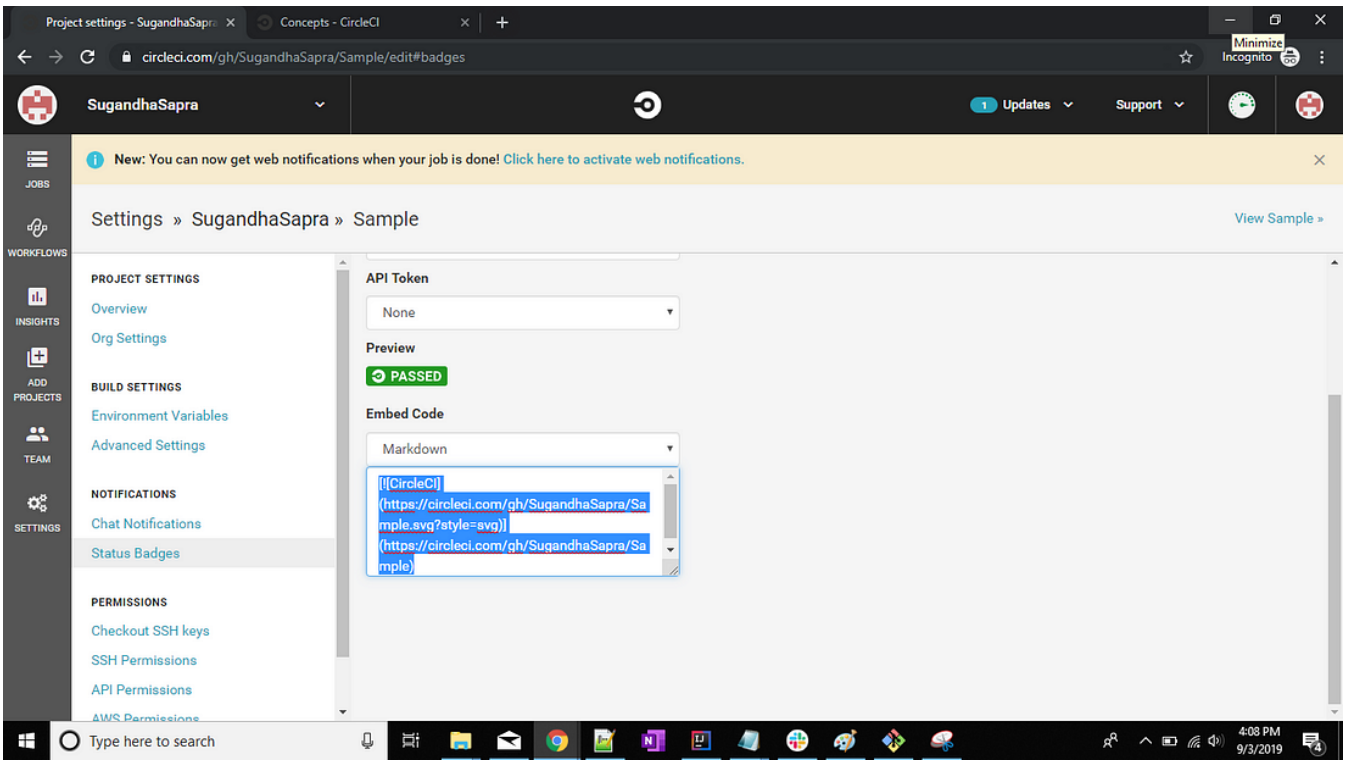
```
git add .
git commit -m "Committing Circle CI files"
git push
```

- Here is what the build looks like.

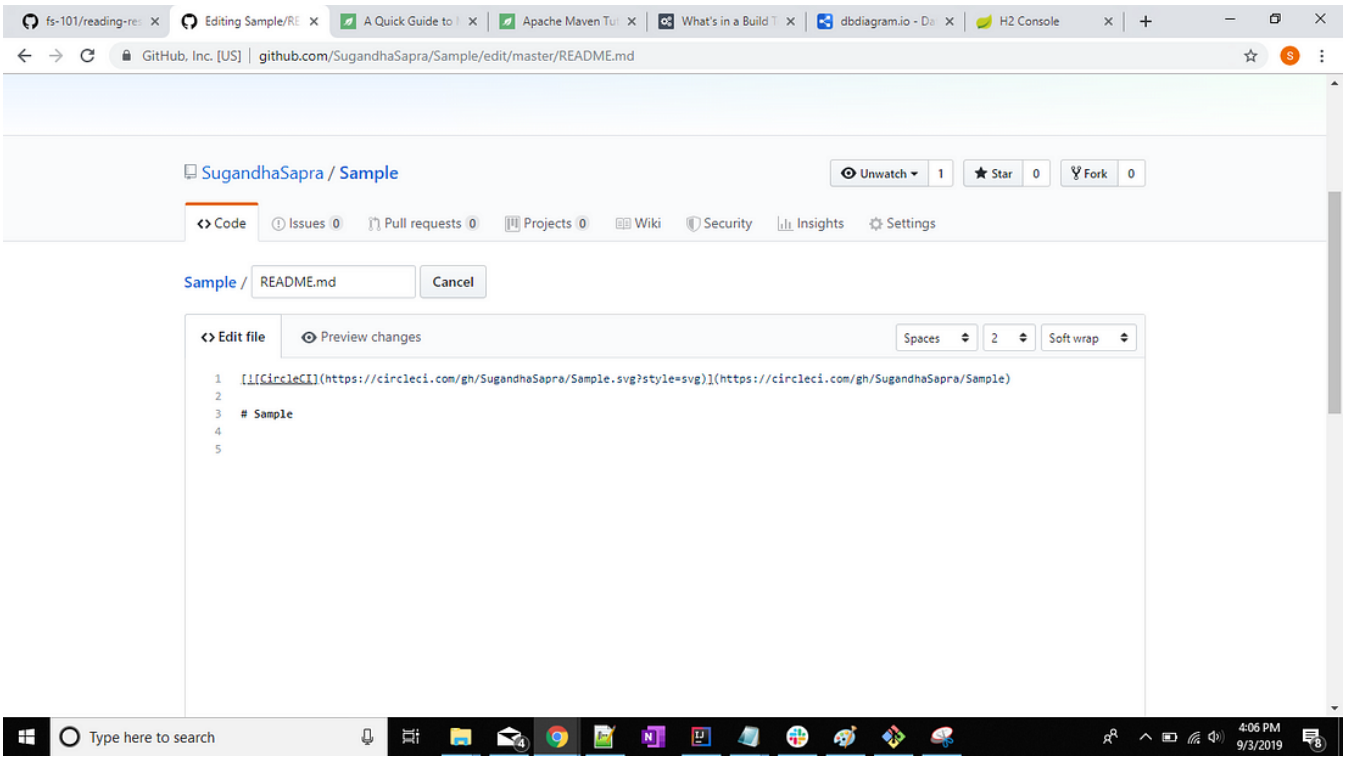


- For showing the CI status on our project Github repo, we will add CircleCI badge using the following steps:

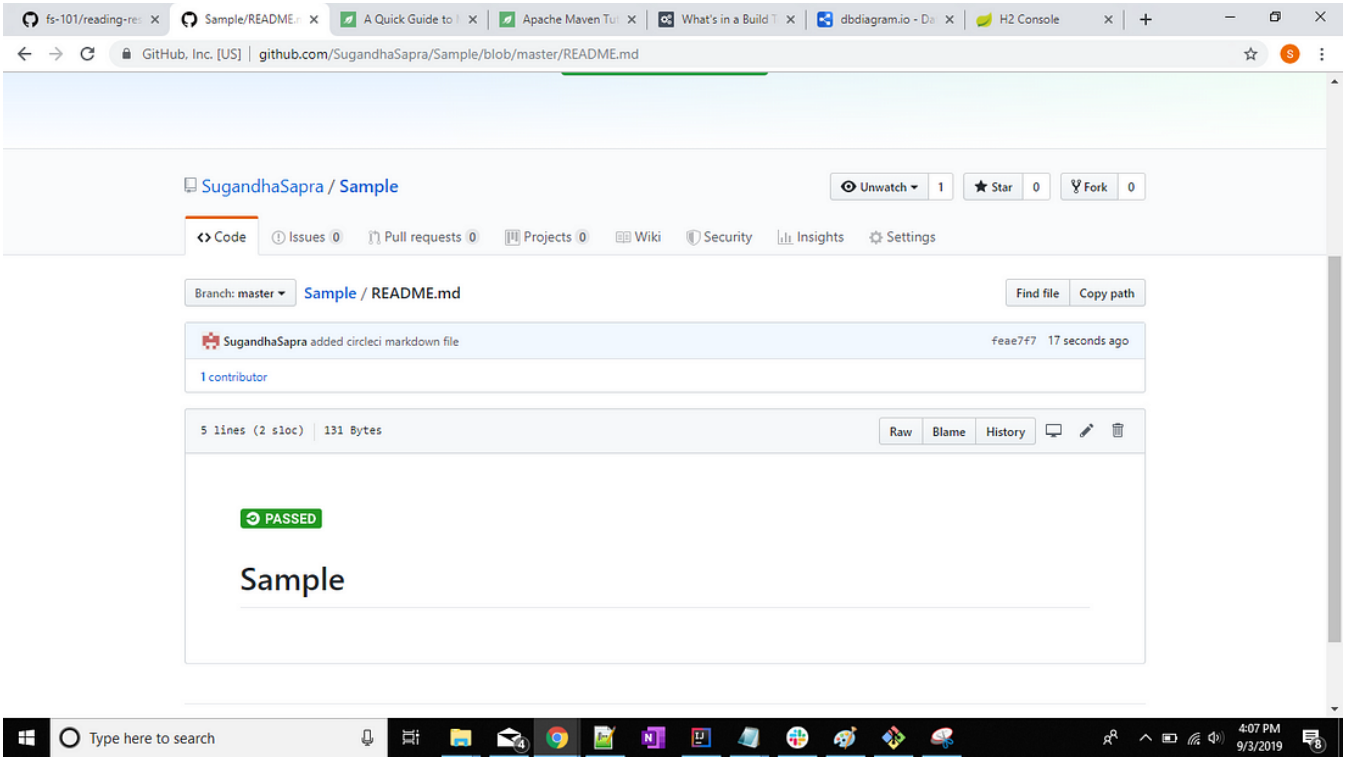
Click on settings of your project on the CircleCI dashboard and then proceed to click on Status Badge under Notification in the menu bar.



Copy the CircleCI URI of the project in the README.md file on Github.



Now check the build status of the project from Github repository.



CircleCI has very powerful build capabilities. We have not covered everything in this post. But, you can click here to explore all the features of CircleCI.

Github Circleci Continuous Integration Spring Boot

Follow

Written by Xebia Engineers

72 Followers · Writer for Xebia Engineering Blog

More from Xebia Engineers and Xebia Engineering Blog