Ref:

<http://about.travis-ci.org/docs/user/getting-started/>

<http://stackoverflow.com/questions/10591599/rake-dbmigration-not-working-on-travis-ci-build>

### Sign in

To get started with Travis CI, sign in through Github. Travis CI and follow sign in link at the top.

### Activate GitHub Service Hook

Now once you’re signed in go to your [profile page](http://travis-ci.org/profile). You will see list of repositories. Flip the on/off switch for each repository that you want to hook up on Travis CI.

Now go to the [Github service hook page](https://help.github.com/articles/post-receive-hooks) and under admin section click on service hooks button and search for Travis under ‘Available service hooks’. And make sure

1. It should be green.

2. It should have Github username and Travis token into the settings. or if not prefilled you can update settings.

### Add .travis.yml file to your repository

language: ruby

rvm:

- 1.9.3

env:

- DB=postgresql

branches:

only:

- 'master'

script:

- bundle exec rake --trace db:create:all

- bundle exec rake db:migrate

- bundle exec rake db:test:prepare

- bundle exec rspec spec/

before\_script:

- psql -c 'create database railsforcharity\_test' -U postgres

- cp config/database.travis.yml config/database.yml

bundler\_args: "--without development"

In order for Travis to build your project, you need to tell the system a little bit about it. To do so, add .travis.yml to the root of your repository. Learn more about [.travis.yml options for Ruby projects](http://about.travis-ci.org/docs/user/languages/ruby/)

#### Validate Your .travis.yml

Put travis-lint gem inside your Gemfile to check .travis.yml is valid or not.

$ travis-lint

=> Hooray, /Users/vintimaheshwari/Projects/railsforcharity/.travis.yml seems to be solid!

### Trigger Your First Build With a Git Push

git add .travis.yml

git commit -m “adding .travis.yml”

git push

Once GitHub hook is set up, push your commit that adds .travis.yml to your repository. That should add a build into one of the queues on [Travis CI](http://travis-ci.org/)

To start a build you can either commit and push something to your repository.

### Tweaking your build configuration

Check your builds in [travis-ci.org](http://travis-ci.org)

Chances are, your project requires some customization to the build process: maybe you need to create a database before running your tests or you use build tools different from what Travis defaults are. Worry not: Travis lets you override almost everything. See [Build Configuration](http://about.travis-ci.org/docs/user/build-configuration/) to learn more.

After making some changes to the .travis.yml, don't forget to check that it is [valid YAML](http://yaml-online-parser.appspot.com/) and run travis-lint to validate it.

### More Information

A Travis worker comes with a good amount of services you might depend on, including MySQL, PostgreSQL, MongoDB, Redis, CouchDB, RabbitMQ, memcached and others.

See [Database setup](http://about.travis-ci.org/docs/user/database-setup/) to learn how to configure a database connection for your test suite. More information about our test environment can be found [in a separate guide](http://about.travis-ci.org/docs/user/ci-environment/).