WEST UNIVERSITY OF TIMIȘOARA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
STUDY PROGRAM: COMPUTER SCIENCE IN ENGLISH

BACHELOR THESIS

COORDINATOR:
Lect.Univ.Dr.Stelian Mihalaș

GRADUATE:
Nagy Gabriel Alexandru

TIMIŞOARA
2018

WEST UNIVERSITY OF TIMIȘOARA

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

STUDY PROGRAM: COMPUTER SCIENCE IN ENGLISH

CONTINUOUS INTEGRATION WITH BUILDBOT

COORDINATOR:
Lect.Univ.Dr.Stelian Mihalaș

GRADUATE:
Nagy Gabriel Alexandru

TIMIŞOARA

2018

Abstract

The purpose of this thesis is to shed light on Buildbot as a Continuous Integration tool. Continuous Integration (CI) is the process of automating the build and testing of code every time a team member commits changes to version control. We target to do this by automating most of the delivery process related tasks, freeing the developer to do other software development related tasks.

Using Buildbot, developers cand test their components for possible errors without the risk of committing to version control and breaking the code. Buildbot is an open source project, written in Python on top of the Twisted network programming framework.

In this thesis we will emphasize the extensibility of Buildbot, and how we can use Python and Angular to strengthen the capabilities of this tool, customizing it to best suit our project.

Contents

Chapter 1

Introduction

1.1 Motivation

1.2 Contribution

1.3 Structure

Chapter 2

Technologies used

2.1 Backend

2.1.1 Python

2.1.2 Twisted

2.1.3 SQLite

2.2 Frontend

2.2.1 Angular

2.2.2 Gulp

2.2.3 Less

2.2.4 CoffeeScript

2.2.5 pug

Chapter 3

Default implementations

3.1 Concepts and terminology

3.1.1 SourceStamps

3.1.2 Changes

3.1.3 BuildSets

3.1.4 BuildRequests

3.1.5 Builders

3.1.6 Schedulers

3.1.7 Builds

3.1.8 Master

3.1.9 Workers

3.1.10 Users

3.1.11 Data API

3.2 Web interface

3.2.1 Home page

3.2.2 Grid view

3.2.3 Waterfall view

3.2.4 Console view

3.2.5 Settings

Chapter 4

Custom implementations

4.1 Backend componentization

split master configuration into multiple modules to facilitate scalability

4.1.1 Email look-up using LDAP

4.1.2 Log parsing

4.1.3 Preferential build steps

4.2 Web interface development

4.2.1 Flask/WSGI dashboards

pros/cons

4.2.2 Angular dashboards

pros/cons

4.3 User scripts

user try scripts to send patches with uncomitted code to buildbot for testing

4.4 Extending the source code

extending buildbot source to allow multiple patchfiles and more API entrypoints maybe?

Chapter 5

Conclusion

Bibliography

[1] Autori, Titlu carte, Editura, An apariţie.

[2] Autori, Titlu articol, Nume jurnal Număr (An apariţie), pag. start - pag. final.

[3] Descriere resursă online, URL: `https://www.google.com`