

# Secure and Quick Software Development by Agile Techniques and Continuous Deployment Pipeline

Pritha Krishnan<sup>1\*</sup>, C. Anand<sup>2</sup>, M. Somu<sup>3</sup>, N. Vasuki<sup>4</sup>

<sup>1</sup>Masters Student, Department of Computer Science and Engineering, KSR College of Engineering, Tiruchengode, India

<sup>2,3</sup>Assistant Professor, Department of Computer Science and Engineering, KSR College of Engineering, Tiruchengode, India

<sup>4</sup>Assistant Professor, Department of Computer Science and Engineering, Government Engineering College, Erode, India

**Abstract:** Software is developed in numerous ways. What is the best way to develop software? There is no single answer to this question due to the present of numerous tools and techniques. But there are proven best practices which give best results. The Art of Developing Web Software can be beautified by following the suitable Agile Methodologies, right choice of tools, suitable modern technologies, CICD and Automating the Web Security Testing. My research is about the best practices attributes and their impact in Software Development. We will look into these in greater detail.

**Keywords:** Agile, automation, best practices, CICD, continuous deployment pipeline, DevOps, DevSecOps, developing software, Gitlab, life cycle, pipeline, quick software development, security, secure automation, secure agile techniques, sprint, test case.

## 1. Introduction

The Best Quality Software Development approach will find the right way with least time consumption, least cost and minimum manpower. Quality attributes such as Automation, CICD Pipeline, Agile Techniques, and Security must have been used in that approach. The next problem is the selection of the Right tools to get used in the development process. We have considered Gitlab for DevOps due to its trending nature.

## 2. Why Gitlab for CI/CD Pipeline?

It is an open source code repository and collaborative software development platform for large DevOps and DevSecOps projects.

## 3. Proposed Software Development Process

- Customer Requirements (Objective, Features)
- Technology Selection
- Technical Design
- Development

### A. Agile

- i. To divide the requirements into a list of features.
- ii. To conduct weekly sprint meetings to select the

- features for the development in the current sprint.
- iii. To create the initial version of code checked into Gitlab.
- iv. To create the CICD pipeline which has Build, Test and Deploy jobs.
- v. To add test cases for Functional and security testing.
- vi. To send the developed Features to the customer to get their feedback.
- vii. To repeat the above iterations to implement the features one by one and update the CICD pipeline with new functional test cases.
- viii. To deploy the Software by running the CICD pipelines deploy job.

## 4. What Is GitLab CI/CD?

Gitlab has an excellent CI/CD offering for projects hosted on Gitlab as well as other git providers. Using GitLab CI/CD we can incorporate all of the three stages that we discussed i.e. Continuous integration, Continuous Delivery and Continuous Deployment.

What makes GitLab CI/CD powerful is the fact that it allows us to host the Git repository to any of the other Git providers such as GitHub, and we can still harness its CI/CD system. we don't even have to change the Git provider in order to use Gitlab CI/CD. The only requirement to run CI/CD is the presence of a special GitLab CI YAML configuration file. GitLab CI YAML file contains all the instructions and data required to run different CI/CD pipelines.

There are plenty of customization options available to shape the pipelines according to custom needs. Another key thing to note is that .gitlab-ci.yml is version-controlled and placed in the repository. This allows even the old versions of the repository to build successfully, making it easier for our team to adopt the CI practice. Reason being, if the GitLab CI YAML is placed in the repository itself, it means we have now put the logic of CI/CD into our repository. Making us free from the worries that our CI/CD system might fail and we might lose the data. Now, wherever a code lives, our CI/CD is present there, making it

\*Corresponding author: prithaamie@gmail.com

simpler to shift from one hosting environment to the another, as long as it suppose the same pipeline. That way, our team can easily make use of CI branches as special different pipelines and jobs and we have a single source of truth for all CI CD pipelines.

### 5. Methodology

The following modules are present in the project:

- 1. Gitlab
- 2. CICD PIPELINE
- 3. Quality Automation
- 4. Agile process

### 6. How to create a CICD Pipeline in Gitlab?

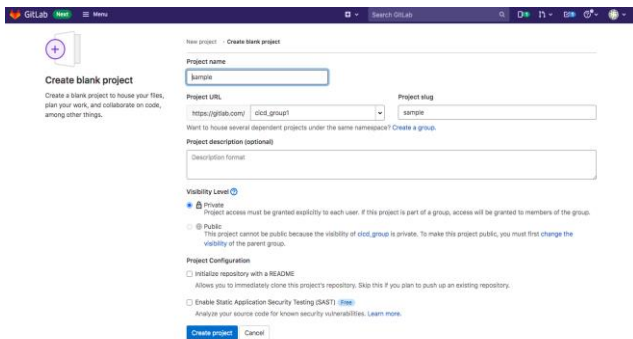


Fig. 1. Create blank project



Fig. 2.

- Enter the Project name
- Choose the private or public which one we need
- Create a new project
- The repository for this project is empty (select the new file)
- Save the file name (.gitlab-ci.yml)
- Then start the coding in the files portion in the repository
- After finishing the coding, click the commit
- Next the pipeline will start

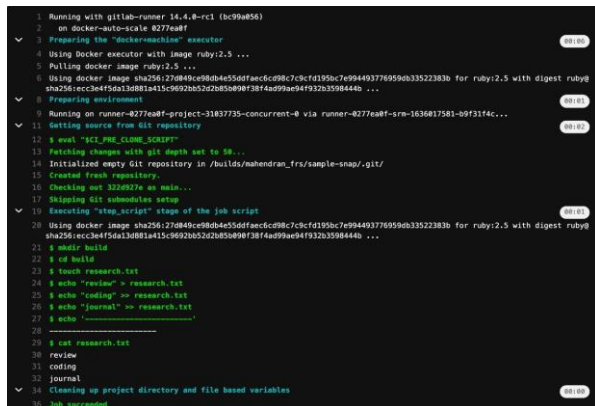


Fig. 3.

### 7. To Reap the Benefits of Agile

Software Development has evolved greatly in the past 3 decades. Agile methodologies have a greater role to play in this evolution. So, a disciplined approach to Iterative Software Development with Periodic Customer Feedback and better Sprint Planning, milestone meeting. To convert different features into epics will improve the overall efficiency of the Software Development. Gitlab has the facility to have a Kanban Board to categorise based on the status of the issues to Open, To-Do, Doing, Done. Its screenshot is attached as follows:

### 8. Test Cases Automation?

- Focused
- Informative
- Adding value
- Trustworthy
- Repeatable

### 9. YAML Features

It is a human -friendly data serialization standard for all programming languages.

YAML: Ain't Markup Language

- 1. Understand existing pipelines configurations from a YAML point of view.
- 2. Learn important YAML language features relevant to Gitlab CI.

### 10. Security Testing in CICD Job

The included template creates SAST (Static Application Security Testing) jobs in the CI/CD pipeline and scans the project's source code for possible vulnerabilities.

```
include:
- template: Security/SAST.gitlab-ci.yml
```

Fig. 4.

The included template creates a DAST (Dynamic Application Security Testing) job in the CI/CD pipeline and scans the project's running application for possible vulnerabilities.

```
stages:
- build
- dast

include:
- template: DAST.gitlab-ci.yml

# Deploys the container to the GitLab container registry
deploy:
services:
- name: docker:dind
  alias: dind
image: docker:19.03.5
stage: build
script:
- docker login -u gitlab-ci-token -p $CI_JOB_TOKEN $CI_REGISTRY
- docker pull $CI_REGISTRY_IMAGE:latest || true
- docker build --tag $CI_REGISTRY_IMAGE:$CI_COMMIT_SHA --tag $CI
- docker push $CI_REGISTRY_IMAGE:$CI_COMMIT_SHA
- docker push $CI_REGISTRY_IMAGE:latest

services: # use services to link your app container to the dast job
- name: $CI_REGISTRY_IMAGE:$CI_COMMIT_SHA
  alias: yourapp

variables:
DAST_FULL_SCAN_ENABLED: "true" # do a full scan
DAST_BAP_USE_AJAX_SPIDER: "true" # use the ajax spider
```

Fig. 5.

### 11. How to Fix the Pipeline Failures?

While building CICD pipelines, things do not always work as expected. For this reason, it is very important to get familiar with the tools we are using and with understanding the console log, step by step. I cannot stress how important these skills are.

### 12. How can the Software Lifecycle be in Gitlab Usage?

- Requirement
- Technology Selection
- Design
- Development
- Test Cases Creation
- CICD Pipeline Creation
- Pipeline Execution
- Test case Automation

### 13. Conclusion

The development of Best Quality Software's by using Gitlab has been researched, and illustrated in this Paper. We look forward to the valuable feedback.

### References

- [1] Harshit Paul "Build an Automated Testing Pipeline with GitLab CI/CD & Selenium Grid" March 4, 2020.
- [2] <https://about.gitlab.com/stages-devops-lifecycle/>
- [3] Thomas Haver "How to Decide Which Types of Test Cases to Automate" Perfecto May 6, 2020.
- [4] [https://docs.gitlab.com/ee/user/application\\_security/dast/](https://docs.gitlab.com/ee/user/application_security/dast/)
- [5] P. Maragathavalli and M. Seshankumar "Automation Pipeline and Build Infrastructure using DevOps International Journal for Research in Applied Science & Engineering Technology," vol. 8, no. 11, Nov. 2020.
- [6] Christina Paule, André van Hoorn, Thomas F. Düllmann, "Vulnerabilities in Continuous Delivery Pipelines? A Case Study," Conference: Proceedings of the 2019 IEEE International Conference on Software Architecture (ICSA 2019).
- [7] Saad Turkey, "Creating Pipeline and Automated Testing on GitLab" Master's Thesis, March 2021
- [8] Fábio Freitas, "Application Security in Continuous Delivery," Masters Thesis, November 25, 2020.
- [9] Kalle Rindell, Jukka Ruohonen, Johannes Holvitie, Sami Hyrynsalmi, and Ville Leppänen, "Security in agile software development: A practitioner survey Information and Software Technology," Volume 131, March 2021.
- [10] Rosnani Shuib and Sa'adah Hassan, "Towards Adopting Software Quality Assurance in Agile Development Methodology," Turkish Journal of Computer and Mathematics Education, vol. 12, No. 3(2021), April 2021.
- [11] Karhapää, P., Behutiye, W., Rodríguez, P. et al. "Strategies to manage quality requirements in agile software development: a multiple case study." *Empir Software Eng* 26, 28 (2021).
- [12] Henni Endah Wahanani, Wahyu S.J. Saputra, Bari Hade Variant Wahono "VCS (Version Control System) Server Infrastructure Design with Git-based Gitlab," vol. 14, June 2019.
- [13] Dorota Celińska, "Coding together in a social network: collaboration among GitHub user's SM Society," 18, July 18–20, 2018, Copenhagen, Denmark" 2018 Association for Computing Machinery.
- [14] Carmine Vassallo, Sebastian Proksch, Anna Jancso, Harald C. Gall, and Massimiliano Di Penta, Configuration Smells in Continuous Delivery Pipelines: A Linter and a Six-Month Study on GitLab ESEC/FSE '20, November 8–13, 2020, Virtual Event, USA" 2020 Association for Computing Machinery.
- [15] Prithwiraj Choudhury, Kevin Crowston, Linus Dahlander, Marco S. Minervini and Sumita Raghuram, "GitLab: work where you want, when you want," *J Org Design*, (2020) 9:23.
- [16] [https://docs.gitlab.com/ee/user/application\\_security/sast/index.html](https://docs.gitlab.com/ee/user/application_security/sast/index.html)