ALDARUNNER

Report

Submitted by Dave Hoevenaars

Fontys Hogeschool Techniek en Logistiek

Reuver, 22-06-2021

# Management Summary

In this project sequence, the possibilities of automating the testing and the grading of student code through GitHub will be analysed and advised upon. The final goal is to have a clear cut advice which will describe how the teachers can automate grading and testing through GitHub.

This will be the first project sequence, therefore there is no previous work to be referred to, as there will most likely be a follow up of the project sequence, I will suggest a few thing how the project sequence could continue.

As this project is part of a sequence, it is important to clearly define the scope, so that its easier to see what was and was not achieved in this project sequence.

Table of contents

[Management Summary 2](#_Toc76949287)

[1. Introduction 4](#_Toc76949288)

[2. Project Justification 5](#_Toc76949289)

[2.1 Project objectives 5](#_Toc76949290)

[2.2 Project requirements 5](#_Toc76949291)

[2.3 Project scope 5](#_Toc76949292)

[3. Project products 6](#_Toc76949293)

[3.1 Customer quality expectations 6](#_Toc76949294)

[3.2 Customer acceptance criteria 6](#_Toc76949295)

[4. Solution & Results 7](#_Toc76949296)

[4.1 Analysis 7](#_Toc76949297)

[5. Conclusion & Recommendations 17](#_Toc76949298)

[Literature List 18](#_Toc76949299)

[Resources 19](#_Toc76949300)

[Questions 21](#_Toc76949301)

# 1. Introduction

My assignment is consisting of researching and realising a prototype of an automated assignment handling system, which will remove the need for labour intensive testing and also reduce the complexity and possibly the vulnerability level of the current system in place. And document my findings about this in such a way that it can be easily picked up by whoever continues within this project sequence.

# Project Justification

## Project objectives

The goal of this project is to give the teachers advice on how to utilise the GitHub platform in order to conduct automated testing and grading. In order to do that I will be focussing on GitHub Actions and Classrooms. Once the current situation has been analysed and the to be has been designed, I will test the prototype in order to give advice on how to best implement the prototype, taking into account scalability, maintainability and security.

## Project requirements

To finish the project, we need some resources. These will be provided by PSV, Fontys or by ourselves. The things we need are:

- Java assignments

- GitHub Actions

- GitHub Classrooms

- Maven knowledge

- JaCoCo knowledge

## Project scope

The scope describes what will be done during this project. All involved parties will know what they can expect during this project. We also describe the exclusions. This makes clear what won’t be done during this project.

In scope

- Analysis of the current situation for building and testing assignments

- Design of the situation to be for building and testing assignments and or exams

- Prototype of building process for exams and assignments

- Advice on how to keep your GitHub actions private

- Advice on how automated correcting of assignments and exams could take place at Fontys

Out of scope

- How to connect Canvas with GitHub classrooms

- How to Run Coverage report

# Project products

In this chapter the quality of the products will be discussed. Spread over the 18 weeks that are set out for the project there will be a couple of different products that are going to be delivered to the customer. This chapter addresses the quality, the expectations of the customer and the acceptance criteria for the different products.

## Customer quality expectations

In this part the customer’s quality expectations are explained. This topic describes the project deliverables that the customer, in this case Fontys teacher – Pieter Hombergh, has on the project.

In various meetings it was shared that the expectations of the project are as follows:

- Remain relevant to the time constraints of the project;

- Well documented project, with listed resources for technologies that were applied.

- The advice on how to automated testing and building, must include security from student attempted fraud.

## Customer acceptance criteria

The project deliverables and acceptance criteria regarding quality and intended use are described in detail in order to serve as an official agreement between the project stakeholders. This part describes the sufficiently added value of the project to the customer in order to consider this project to be successful.

|  |  |  |  |
| --- | --- | --- | --- |
| Deliverable | Acceptance criteria | Product goal | Intended use |
| As is analysis | Fact-based, transparent | Creating clarity of the current system in place | To be used a reference for developing the To Be Design |
| To be Design | Every feature from the requirement document has been met | Providing the arguments of investment justification to the Fontys board | To create a clear overview of the new assignment handling solution |
| Implementation Advise | Step by step approach | Guideline of the assignment handling solution implementation. Described in steps | This plan is going to serve as the guideline of the next stage in the assignment handling solution of the customer |

Table 1 products

# Solution & Results

In this section the progress that has been made will be discussed such as analysis artefacts and working prototypes.

## Analysis

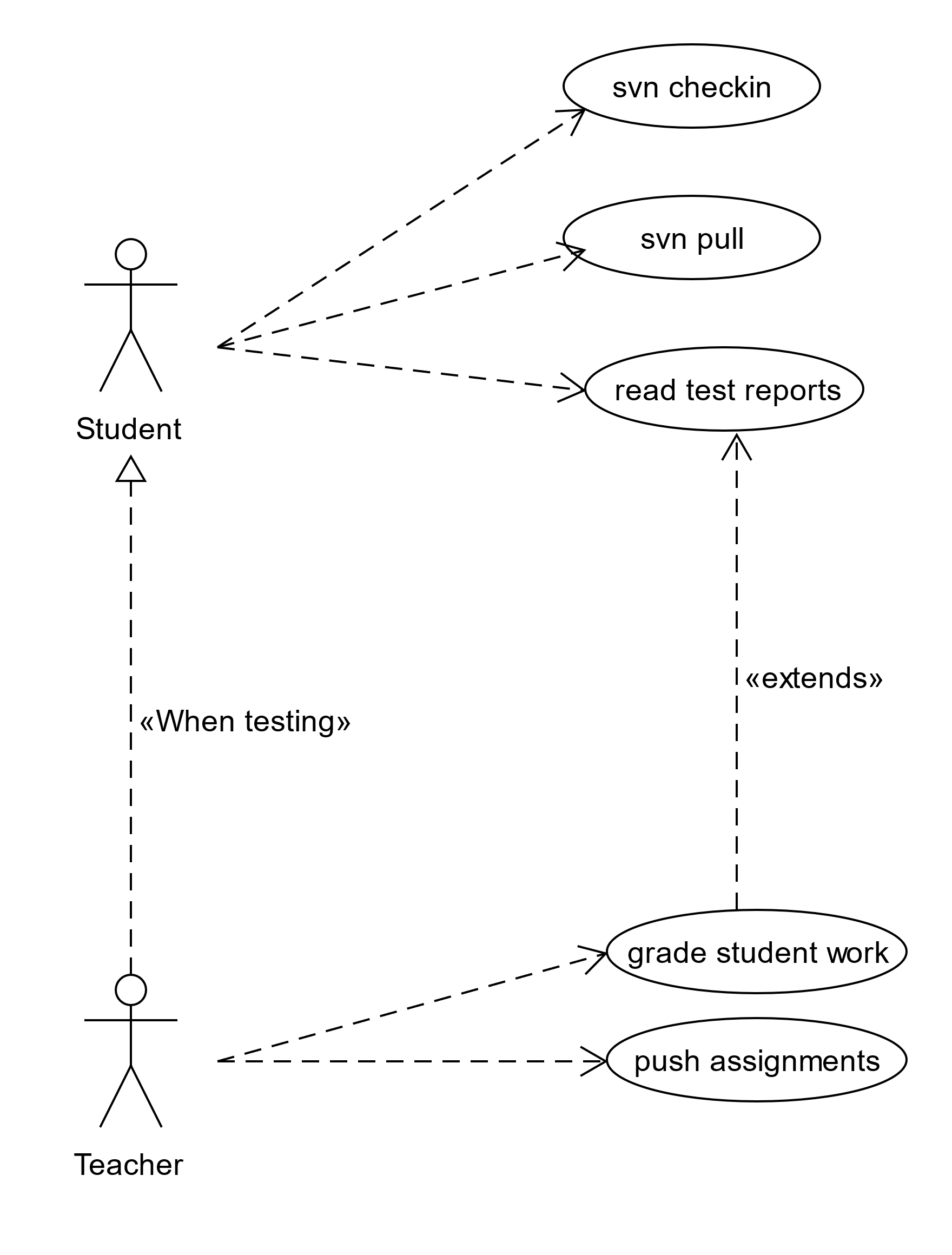


Figure 1 Usecase diagram

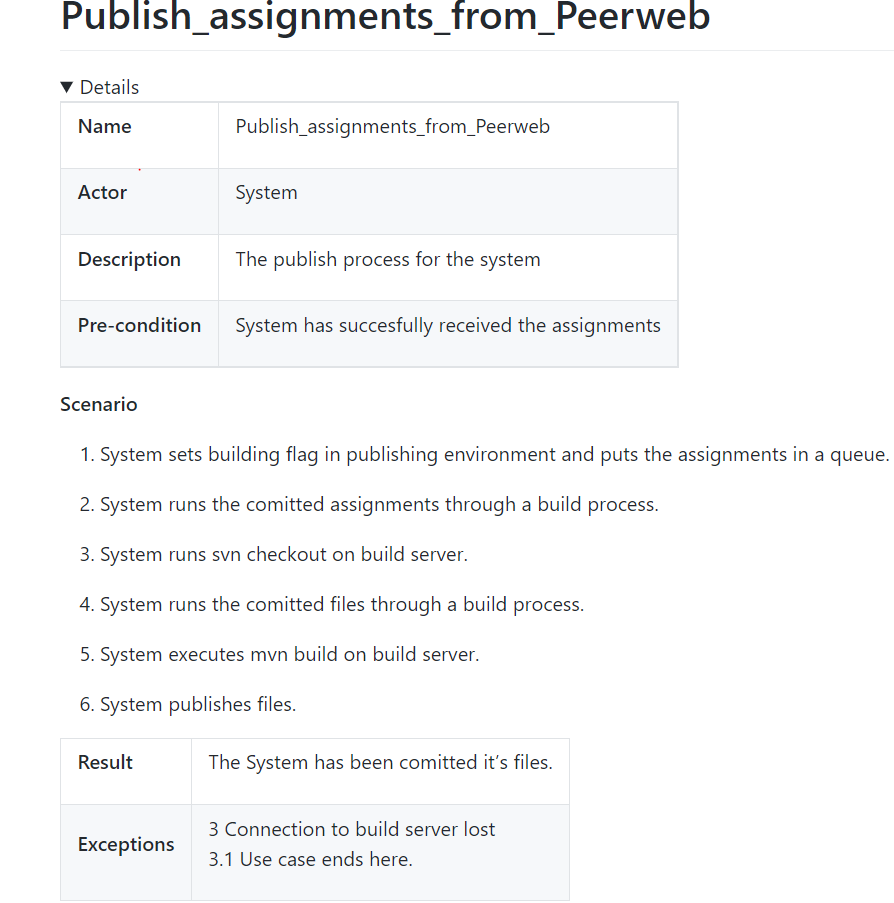


Figure 2 Publish assignment from peerweb

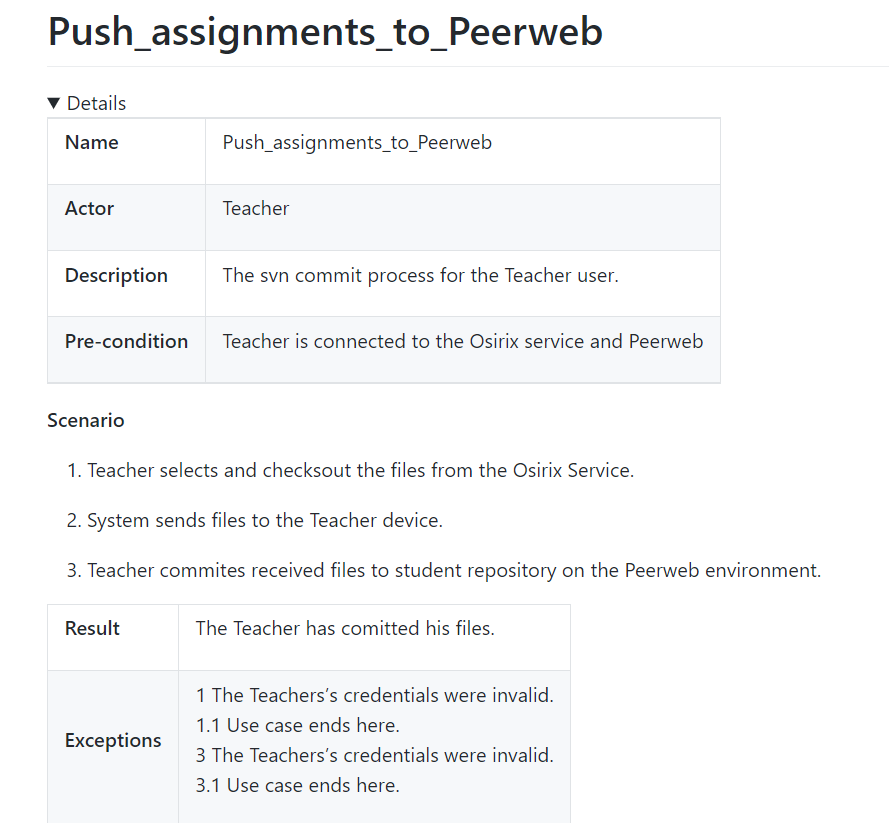


Figure 3 push assignment to peerweb

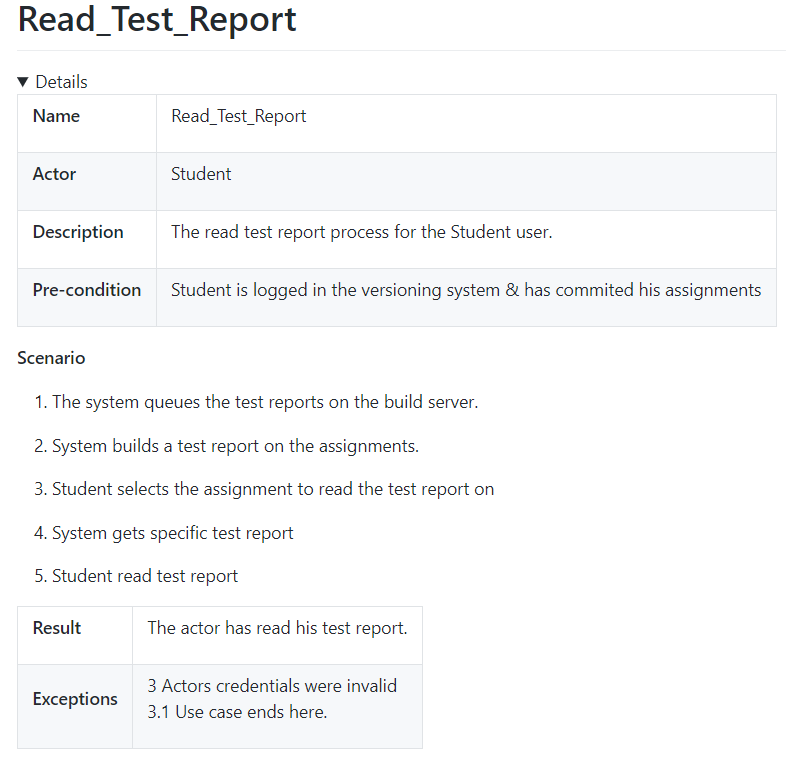


Figure 4 read test report

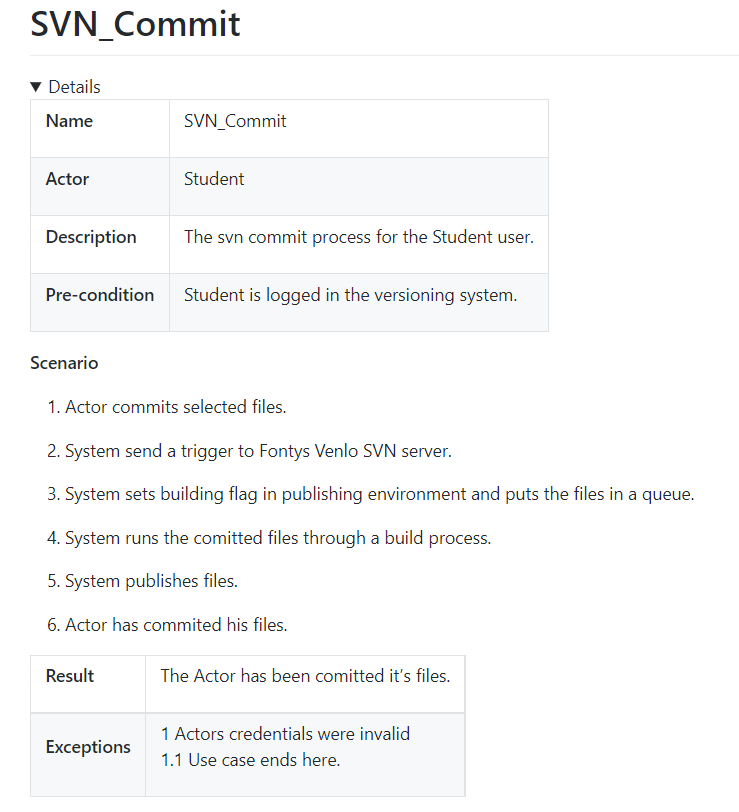


Figure 5 SVN Commit

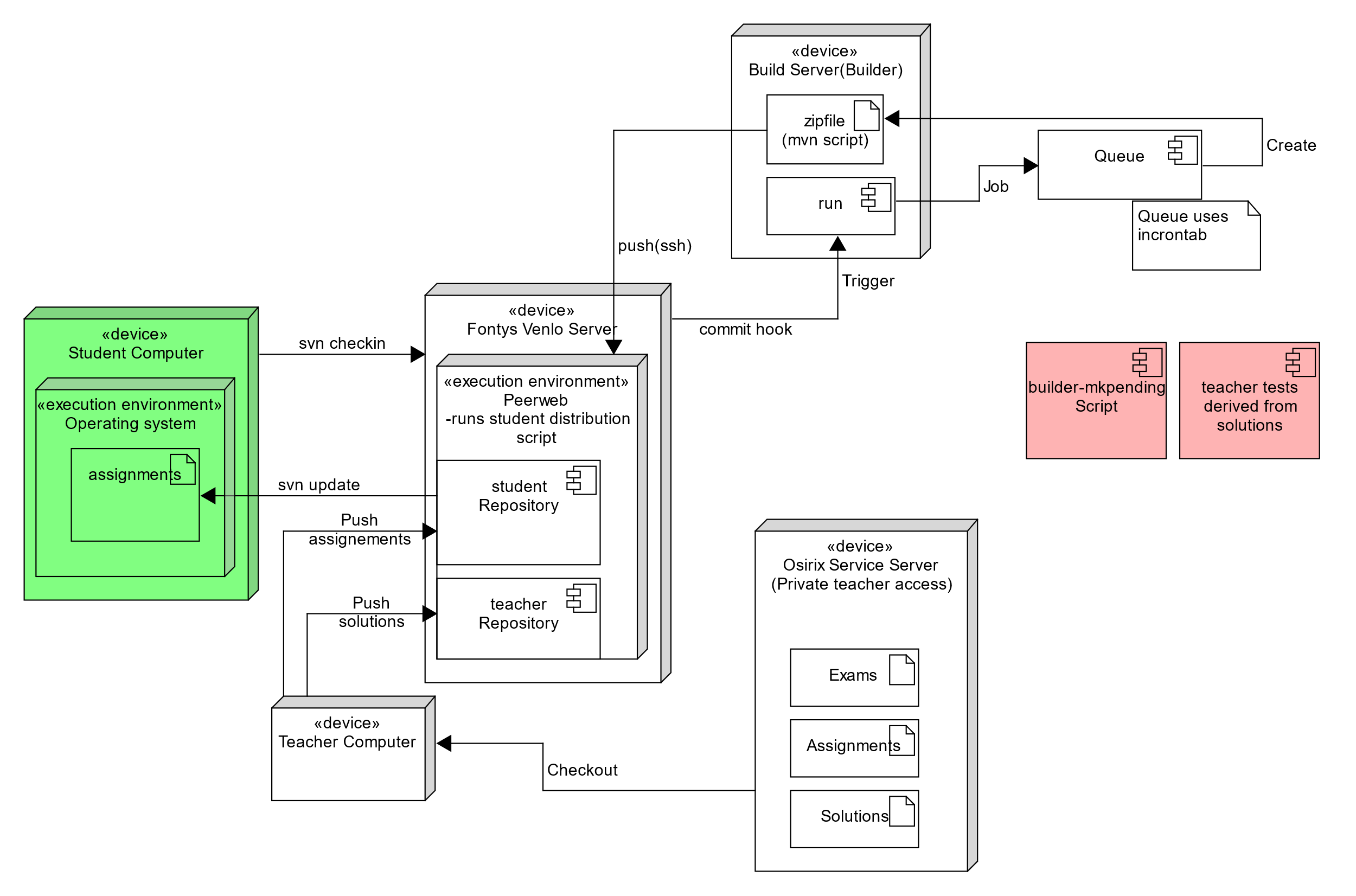


Figure 6 : Deployment Diagram(as is)

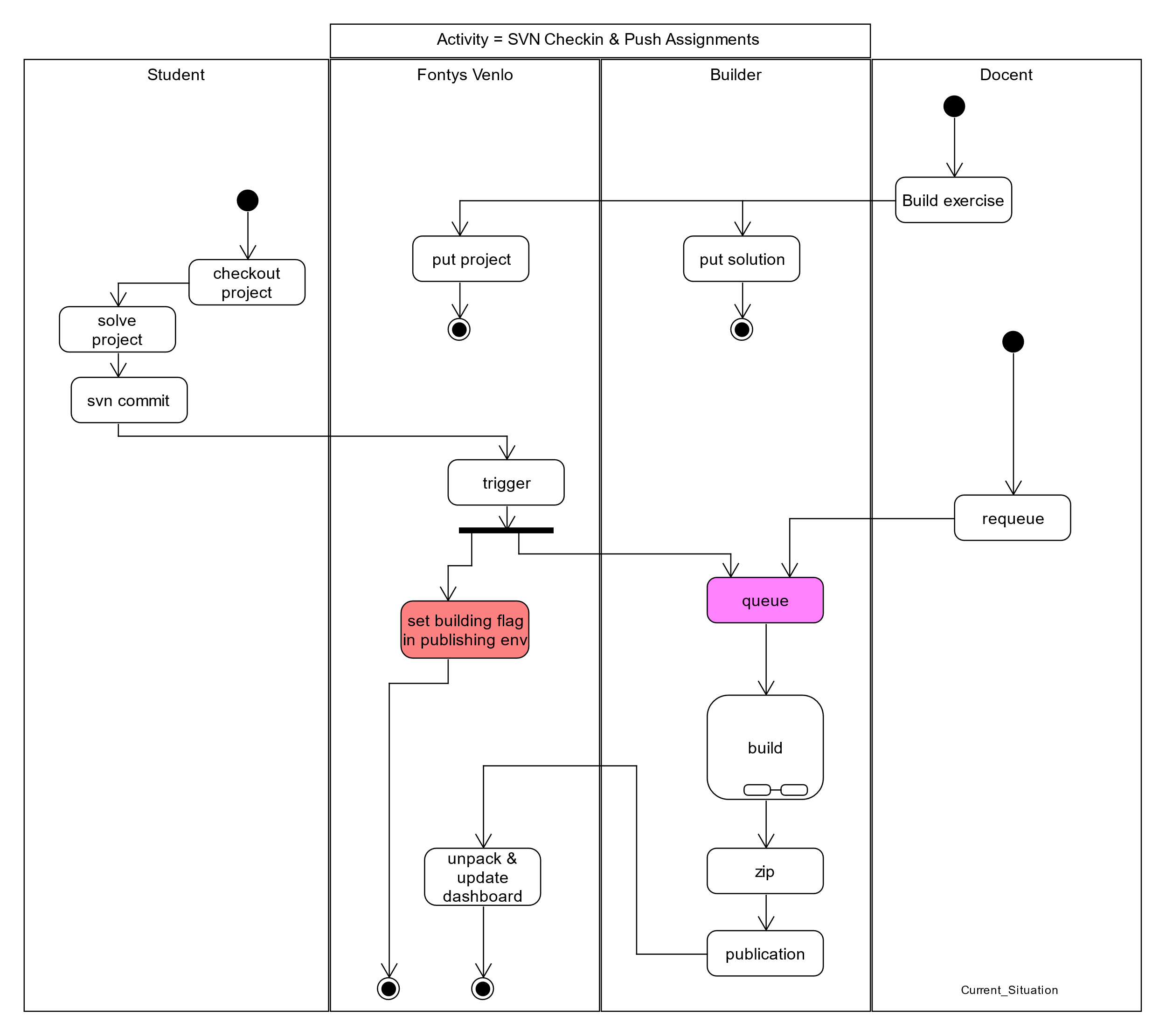


Figure 7 Activity Diagram: AsIs

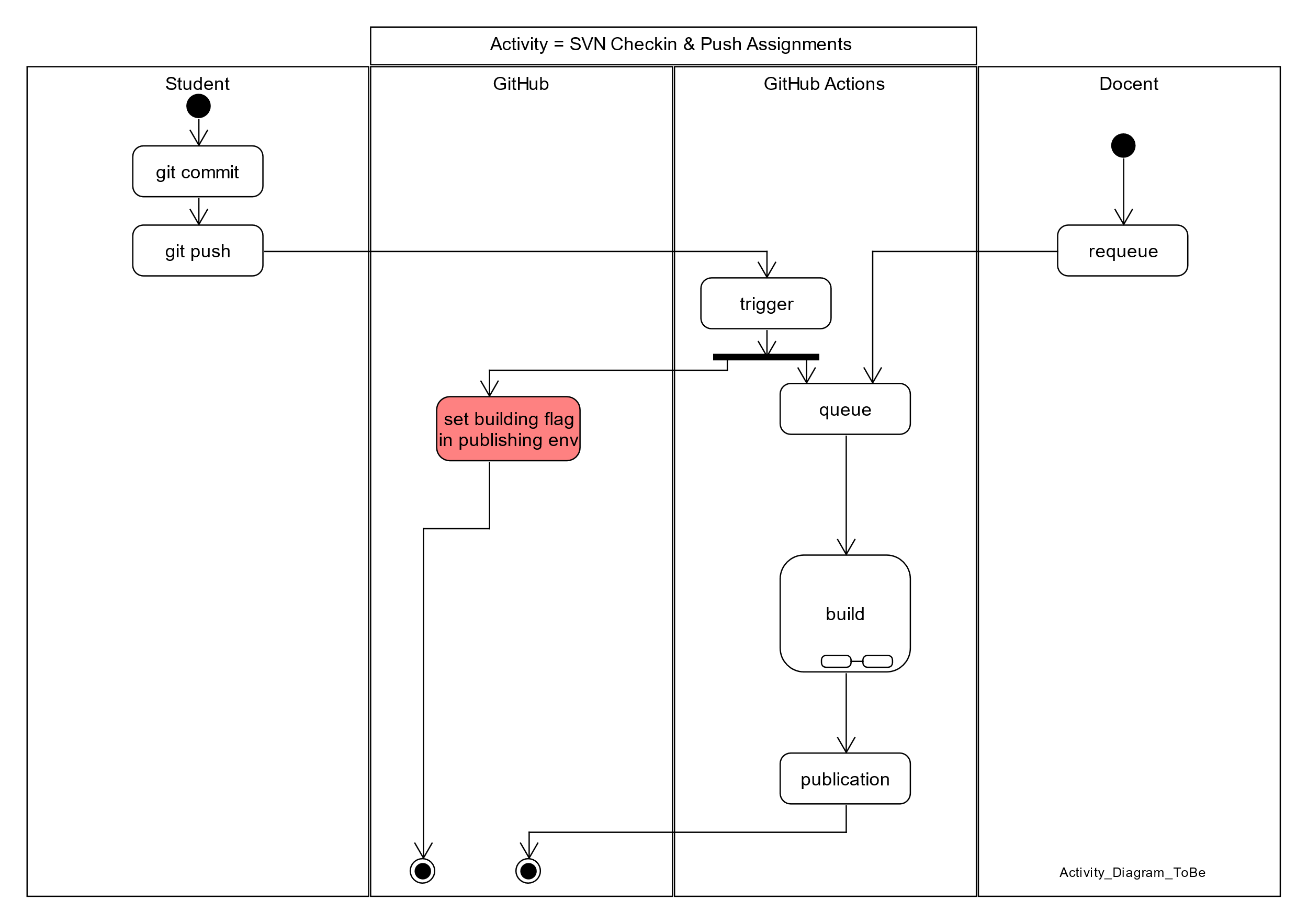


Figure 8 Activity Diagram: ToBe

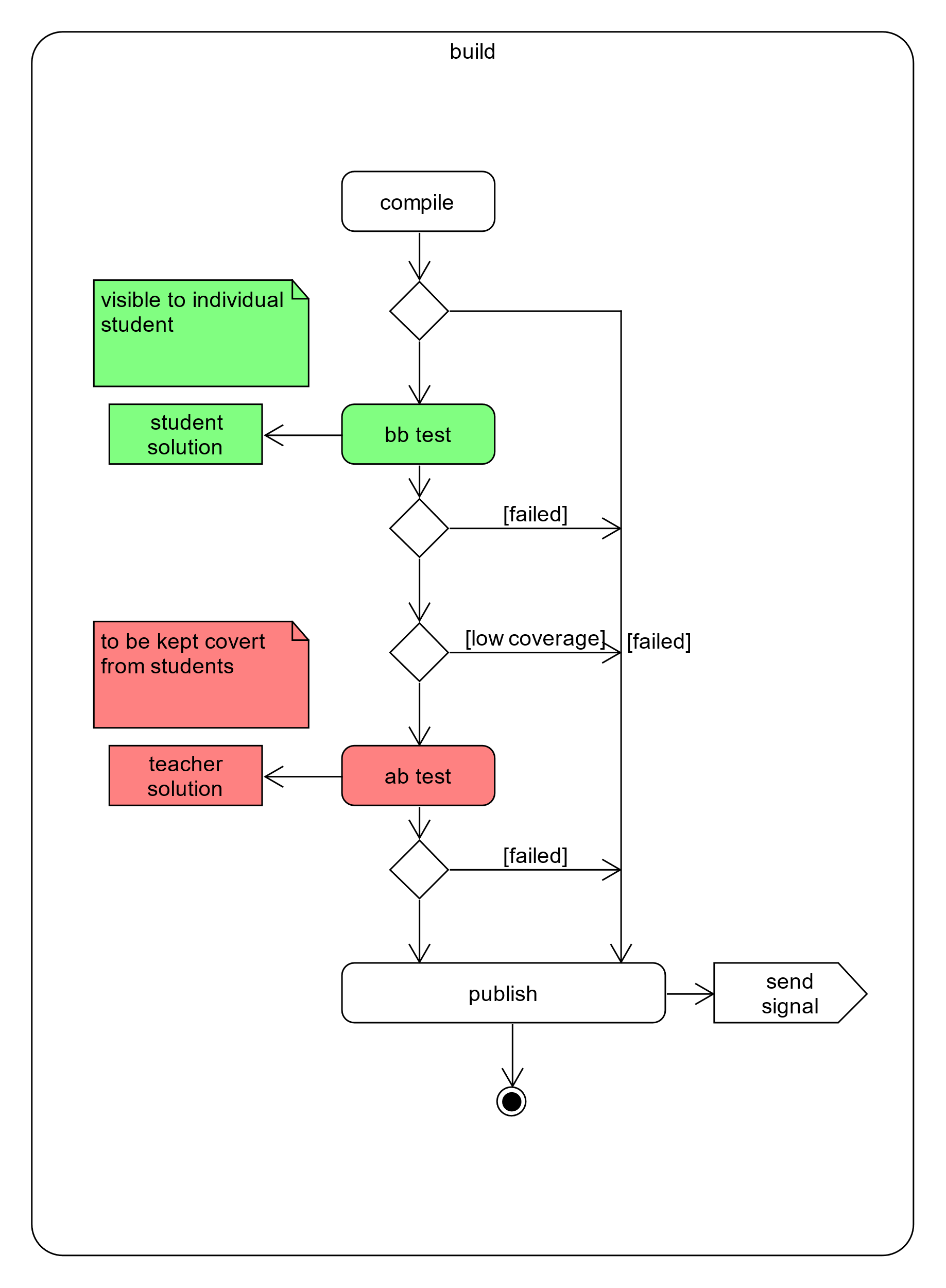


Figure 9 Build details

# Conclusion & Recommendations

In this section there will be argued why certain automation services would be better for this particular case.

Firstly, I would like to mention that the services provided by GitHub makes a good automation tool for distributing, testing and grading student work. Especially GitHub Actions will provide you with the flexibility and ease of use continuous integration, which as an excellent example for student to follow.

Secondly GitHub classrooms can work along side GitHub actions really well by taking advantage of the repositories where those actions are defined in, as well as being an effective and automatable Student administration tool.

In this project sequence I managed to construct a prototype which allows for automated software testing and (maven) building. I also was able to make this work in collaboration with GitHub Classrooms. And more importantly, I provided a starting point on which my successors can build on.

# Literature List

* GitHub Guides
* YouTube
* Mozilla

# Resources

GitHub Actions Tutorial:

<https://docs.github.com/en/actions>

GitHub Actions Demo:

<https://youtu.be/cP0I9w2coGU>

**Summary GitHub Actions:**

You can automate all the actions in a normal github work flow(main, branch) +

But also add custom actions in between those normal actions (Custom software development lifecycle workflows directly in your GitHub repository. +

Github actions are event driven, meaning that you can run a series of commands after a specified event has occurred(like a pull request).

Github actions use YAML syntax to define the events, jobs and steps(this is stored in your repository @ .github/workflows

In this case push is the event that triggered the action, and uses steps to control the order in which actions are run which automate your software testing.

These steps are: GitHub actions checks out the pushed code, install the software dependencies, runs bats-v to output the software version

Furthermore you can build test and publish across multiple Operating systems and languages in one workflow.

Canvas API:

<https://developer.mozilla.org/nl/docs/Web/API/Canvas_API>

GitHub Actions Template Workflow:

<https://github.com/actions/starter-workflows/blob/055373ee0b531de9b779896c520d0555e7df48ae/ci/blank.yml>

GitHub Packages Explained:

<https://www.youtube.com/watch?v=N_-Cu9_2YAA>

**Summary GitHub Packages:**

Enabling sharing of packages that are to be trusted and that you can rely on

Fully integrated with github

Anyone with an account can publish registries

Works with npm, maven, docker, ruby registry protocols.

Unless you have a personal access token, you get not directly get packages available from github

ACCESS TOKENS can be given certain permissions.

GitHub Classroom Setup Tutorial:

<https://www.youtube.com/watch?v=KXWXg68KpTY>

Translating Test results to Canvas or other student administration systems:

<https://youtu.be/KXWXg68KpTY?t=702>

Creating Maven packages with GitHub:

<https://www.youtube.com/watch?v=MhzoxE7NdpI>

How to deploy a Maven Package:

<https://docs.github.com/en/actions/guides/building-and-testing-java-with-maven>

Sure Fire report Template:

<https://github.com/marketplace/actions/surefire-report>

JaCoCo Coverage:

<https://github.com/marketplace/actions/jacoco-report>

JaCoCo Badge:

<https://github.com/marketplace/actions/jacoco-badge-generator>

Conditional PIT Test with PR-Landmine:

<https://github.com/marketplace/actions/pr-landmine>

# Questions

1. Can GitHub do functionally the same as the current systems in place? With some adaptations, yes
2. If students have been imported from Canvas to GitHub Classrooms, can they still be added? Yes through a CSV file, if your GitHub Classroom is linked to a Canvas systems you’ll have to add the students there.
3. Can students access the GitHub actions? If they are defined in the assignment folder then yes, there is a solution in the work however in which the actions are called from a private repo in which cases the students won’t have access.
4. How to publish JaCoCo Coverage? Through Canvas or Email.
5. Where to the Maven build artefacts end up? In the target folder of your repository