# Continuous Integration Report

## a) Summarise your continuous integration method(s) and approach(es), explaining why these are appropriate for the project.

For our project, we have decided to utilise GitHub to collaboratively work on the project without negatively impacting our colleague’s work, due to GitHub’s version control features.
In our development process, we automatically build and test our project on each pull request and on each merge with the master branch. We do this so that we can test each pull request before it gets merged, but also to guarantee that if someone were to push to master, that would also get tested, which our change management policy strictly forbids. Additionally, it assists us in maintaining a high standard of code, as we will be less likely to merge something that fails to compile. It is important to note that we do not deploy our builds, we only test them. This is because we also utilise manual testing, which can’t be done by continuous integration methods. As such, we would still need to do manual tests on release.

## b) Give a brief report on the actual continuous integration infrastructure you have set up for your project. (5 marks, ≤ 1 page)

The infrastructure that we have set up for our project is GitHub Actions. We created a grade.yml file that is read by GitHub Actions. In this file, we wrote it so that on any pushes or on any pull requests to the master branch, the code will tell the CI to trigger, which compiles the project and runs tests on the aforementioned project.We used the CI to determine whether pull requests should be merged or not. The following is the code we used to set up and integrate continuous integration into our project.

```
on:
       push:
         branches: [
      master ]
       pull_request:
         branches: [
      master ]
```