FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

SOFTWARE ENGINEERING II

CONTINUOUS INTEGRATION WORKSHOP - I TERM 2021

LAB REPORT

GROUP#3:

- Cajas Correa Rogwi Cajas
- Vivas Avilez Jonathan Andrés
- Vera García Pedro Gabriel

1. INTRODUCTION

In the following practice we work in what is called Continuous Integration (CI) using tools like Jenkins to build, test and deploy software and Ngrok to expose a local web server to the Internet.

Continuous Integration is the practice of automate the integration of code changes of various contributors in a central repository where they can run compilations and tests. This practice is important to verify if the new code achieves code quality standards.

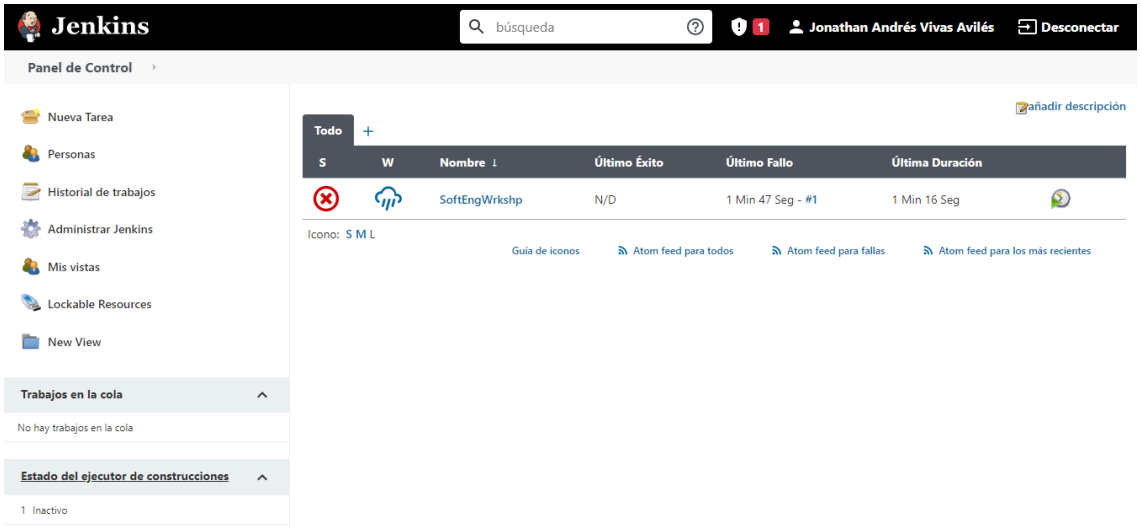
The importance of Continuous Integration is notorious when you can see the problems that appear when it´s not used. Without CI the developers have to coordinate and communicate manually every time a code change is made. This problem is visible not only in development team but others departments of the organization. The synchronization task become complex and confuse, the code publications become slower and with more failures.

The problems of not using CI increase when the team and projects size is bigger. And that's why CI is recommended to be used in very large projects when the final product is complex and robust.

2. DEVELOPMENT

Repository Link: https://github.com/jvivas-rock/SoftIng.git

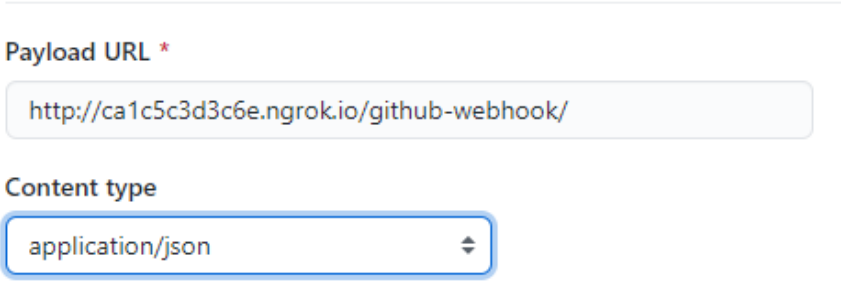
Step 1: Jenkins Installation



Step 2: Ngrok Installation

```
ngrok by @inconshreveable

Session Status                online
Account                       jvivas-rock (Plan: Free)
Version                       2.3.40
Region                        United States (us)
Web Interface                 http://127.0.0.1:4040
Forwarding                    http://ca1c5c3d3c6e.ngrok.io -> http://localhost:8080
Forwarding                    https://ca1c5c3d3c6e.ngrok.io -> http://localhost:8080

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

Step 3: Repository with the Webhook



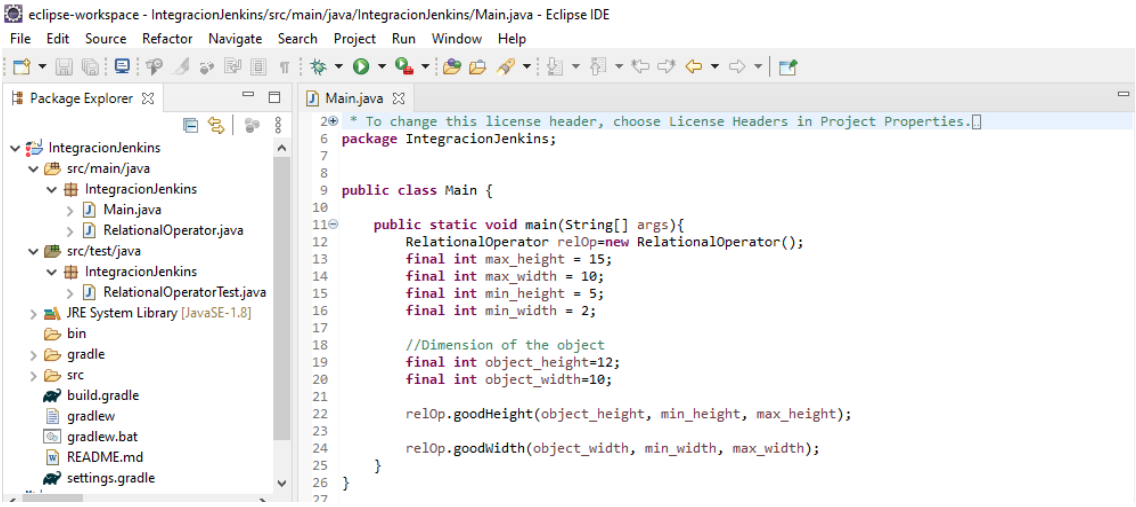
Step 4: Github-Ngrok connection

```
Forwarding                    http://ca1c5c3d3c6e.ngrok.io -> http://localhost:8080
Forwarding                    https://ca1c5c3d3c6e.ngrok.io -> http://localhost:8080

Connections                   ttl     opn     rt1     rt5     p50     p90
                              1       0       0.01    0.00    5.08    5.08

HTTP Requests
-------------

POST /github-webhook/         200 OK
```

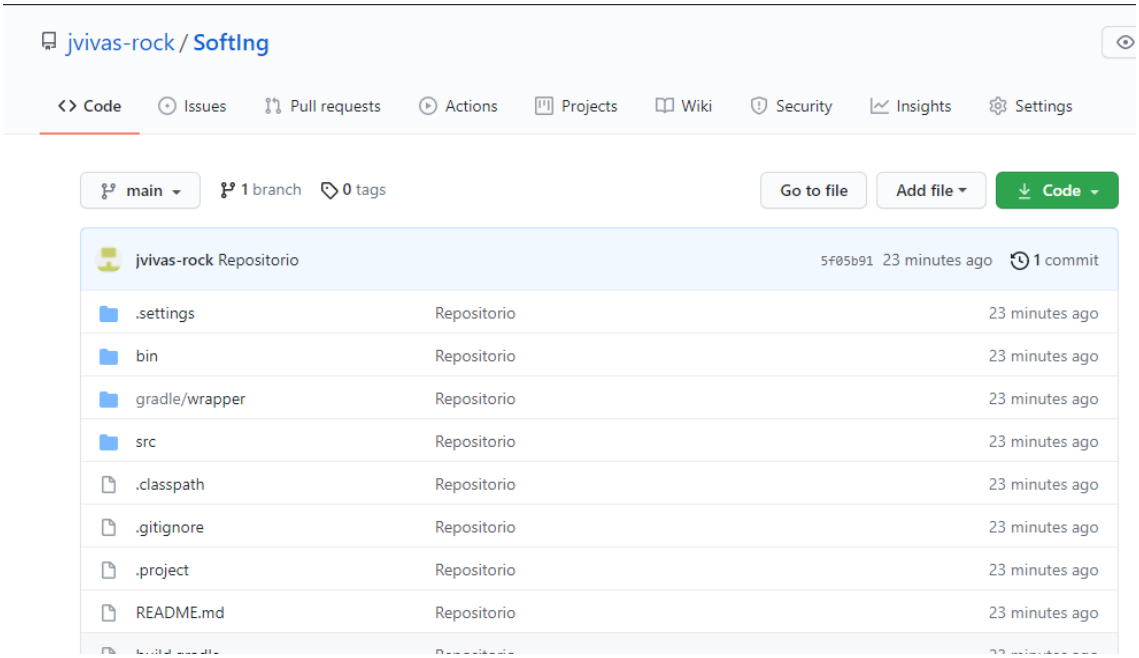
Step 5: “Test Results Analyzer” plugin instaled

Step 6: Jenkins project created

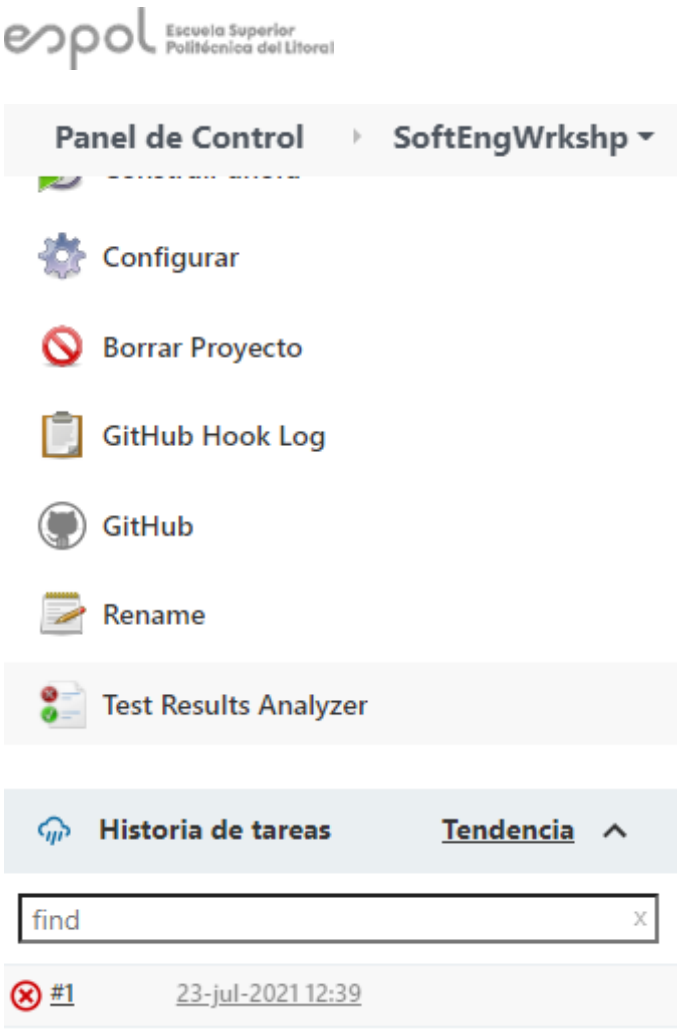
Step 7: Repository open in Eclipse IDE



Step 8: Pushing the practice code to the repository



Step 9: Jenkins executions

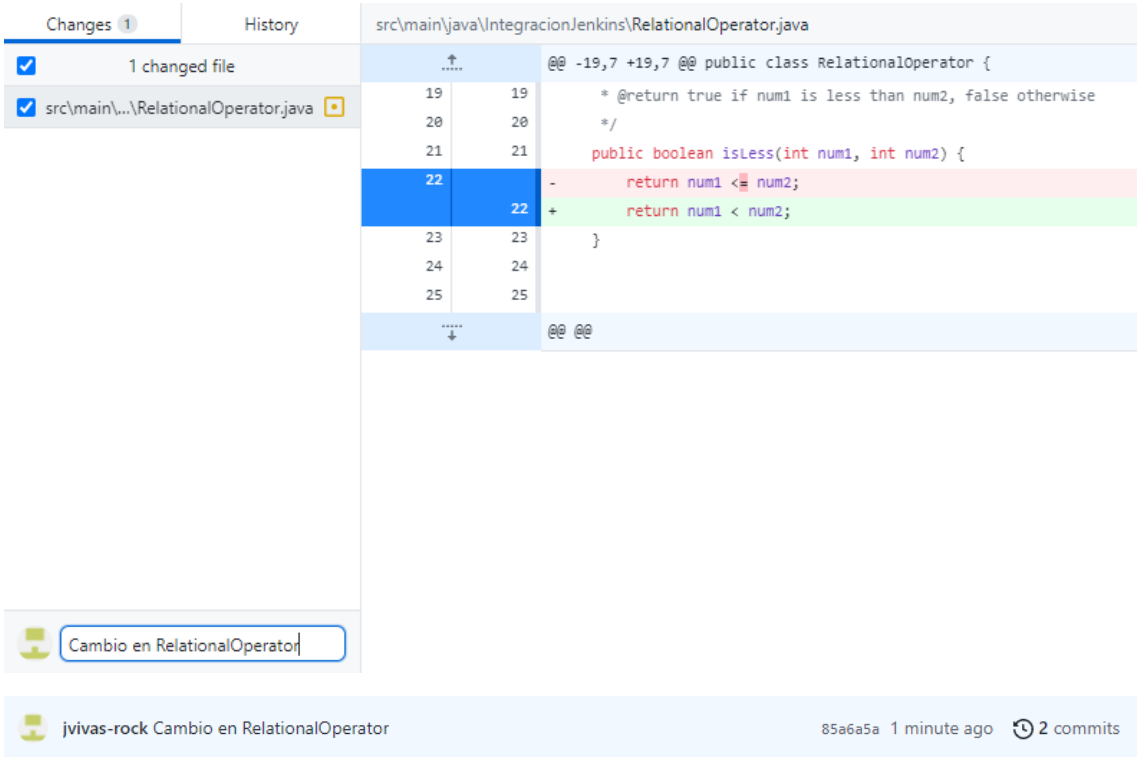


Step 10: Test Result Analyzer run results saved in CSV format

Step 11: Changes on the function

```
public boolean isLess(int num1, int num2) {
    return num1 < num2;
}
```

Step 12: Code changes pushed



3. CONCLUSION

- Continuous Integration is very useful with large projects when the development process is large and complex so the code changes have to be carefully tested. CI automate this process helping the development team to run tests and compilations of the code.

4. RECOMENDATIONS

- In the Github repository verify that you are workin on main branch to avoid problems because jenkins works on default with the main branch not the master one.
- Be Carefully with the port you use to run ngrok, it should be the same port that you use for the jenkins server. In this Case is recommended to use the 8080 port.

5. REFERENCES

[1] PubNub, <<What is Ngrok?>>. [En línea]. Available: https://www.pubnub.com/learn/glossary/what-is-ngrok/#:~:text=ngrok%20is%20a%20cross%2Dplatform,the%20local%20machine%20is%20needed.

[2] Atlassian, <<What is Continuous Integration?>>. [En Linea]. Available: https://www.atlassian.com/es/continuous-delivery/continuous-integration