PROJECT – PART 2

SOFTWARE ENGINEERING II

P3

GROUP:

MASTERCODE

INTEGRANTS:

MISHELL EUGENIA ANGULO VILLACIS

HAROLD ARTURO YEPEZ NIETO

OSCAR DARWIN GUAMAN PACALLA

JONATHAN FERNANDO SANTANA QUISPILLO

JONATHAN ANDRÉS VIVAS AVILÉS

INDEX

[Git Repository (e.g. Bitbucket, GitHub) for Code and Documents 3](#_Toc80037320)

[Use of a Build Automation Tool for the Project (e.g. Maven) 3](#_Toc80037321)

[Evidence of adherence to SCRUM [1]: backlogs, planning, roles definition, etc. Adequate Use of a Teamwork Management Tool (e.g. Clickup) 4](#_Toc80037322)

[Coding Standard Documentation. Detecting and Enforcing Coding Standards 5](#_Toc80037323)

[Preemptive Error Detection (e.g. PMD) 7](#_Toc80037324)

[Definition and Execution of Test Cases (e.g. JUnit) 8](#_Toc80037325)

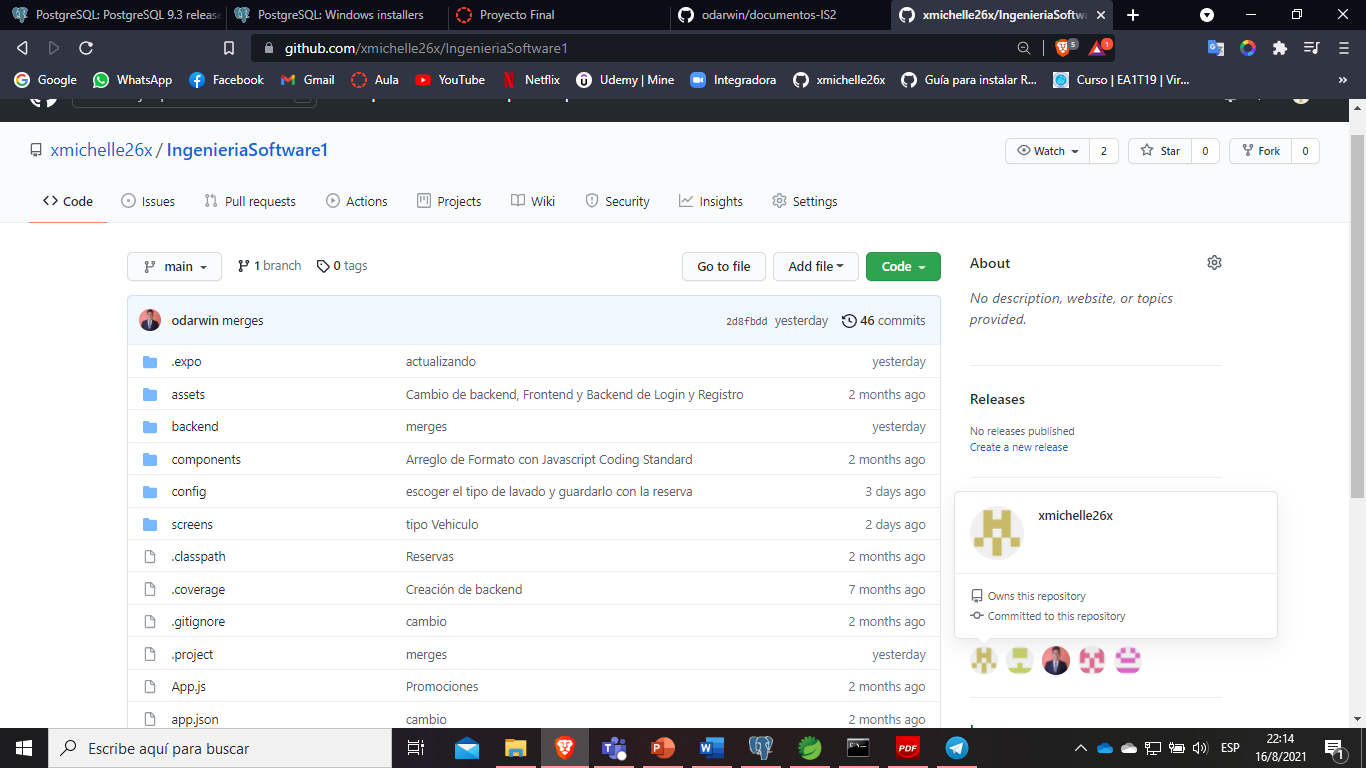
[System Development Guide 10](#_Toc80037326)

[Extra: Use of an Authentication and Access-Control Framework (e.g. Spring Security) 10](#_Toc80037327)

[Video showing the sprint review process 11](#_Toc80037328)

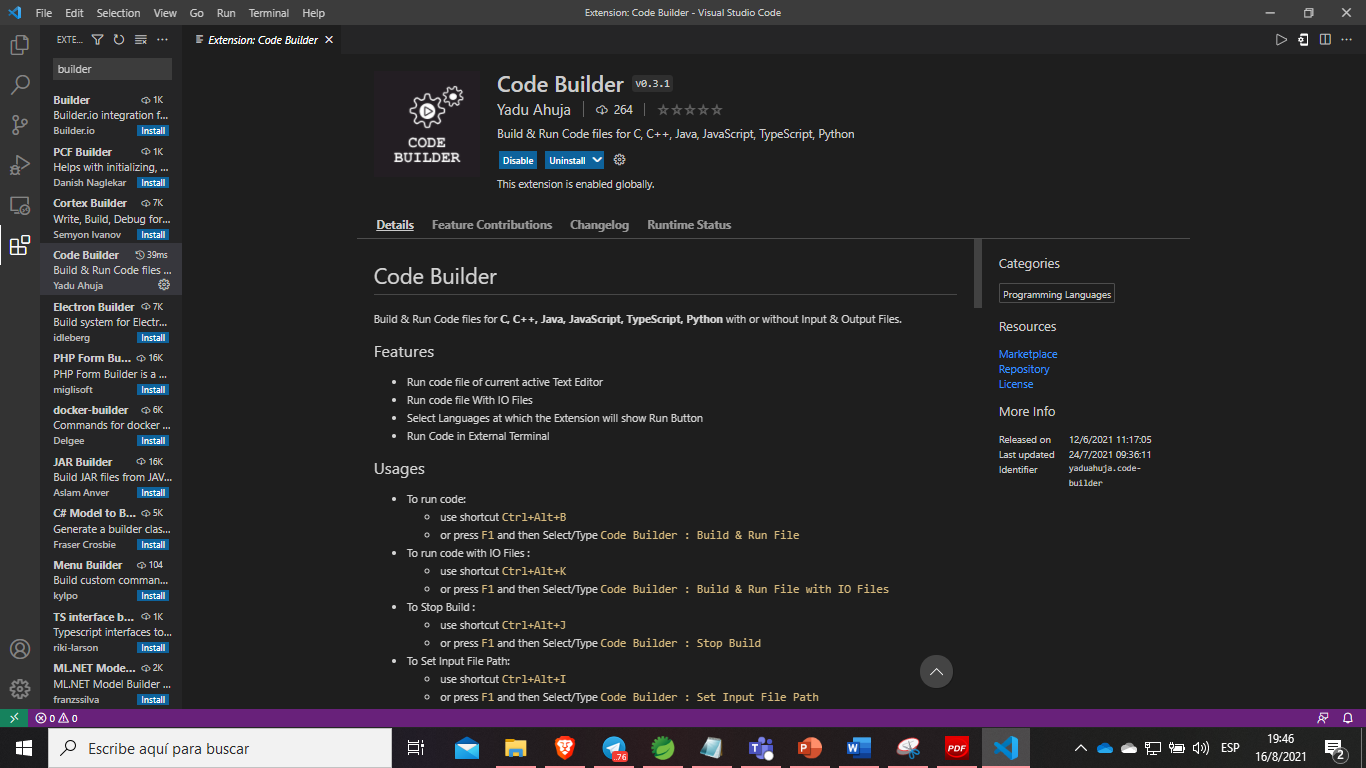
# Git Repository (e.g. Bitbucket, GitHub) for Code and Documents

GitHub Repository Link: https://github.com/xmichelle26x/IngenieriaSoftware1



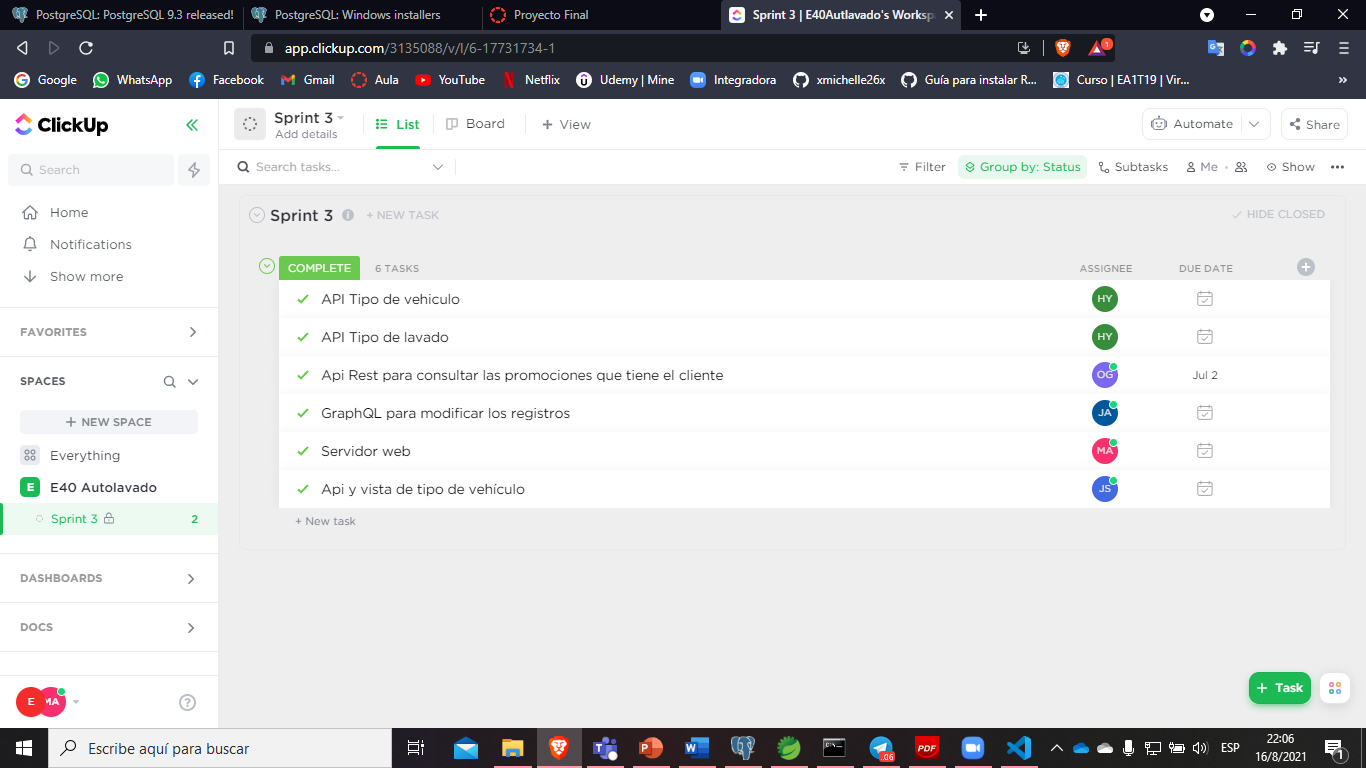
# Use of a Build Automation Tool for the Project (e.g. Maven)

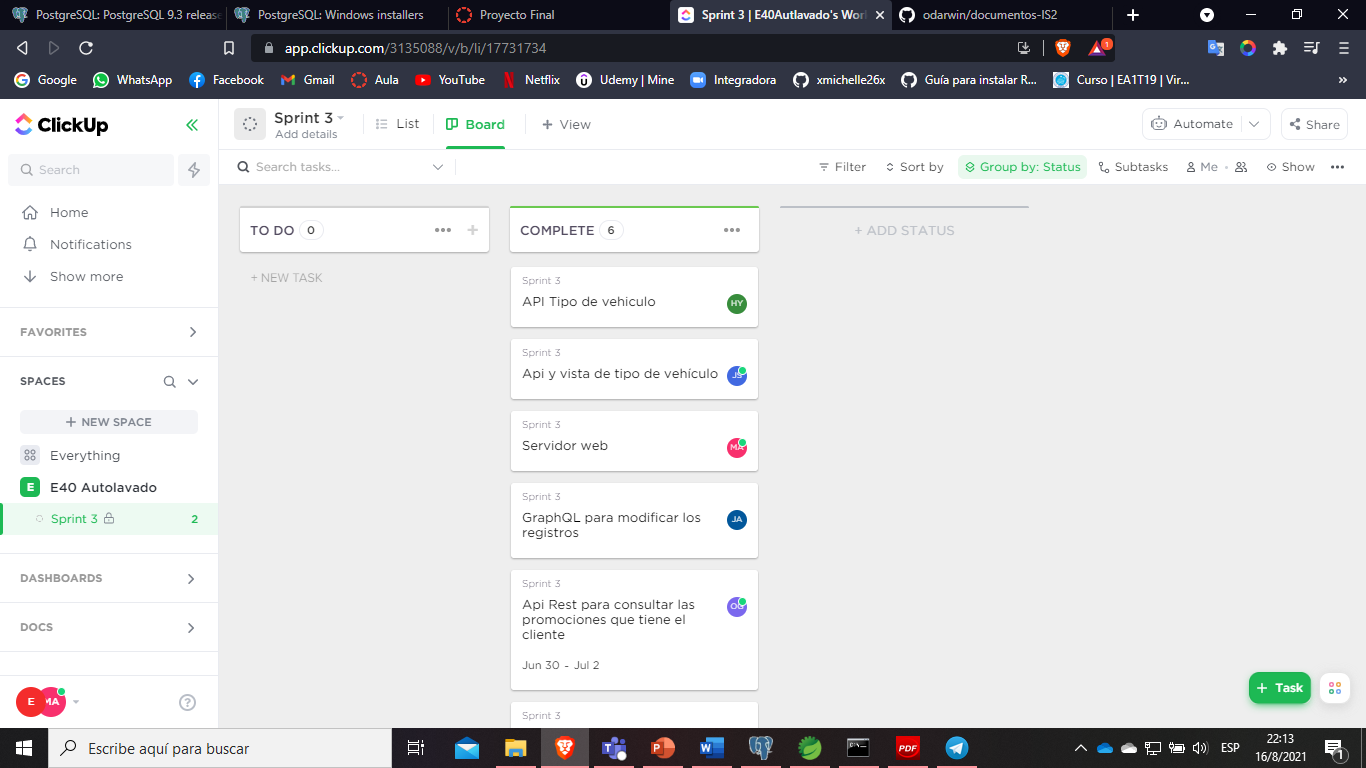
**Tool:** Code Builder



# Evidence of adherence to SCRUM [1]: backlogs, planning, roles definition, etc. Adequate Use of a Teamwork Management Tool (e.g. Clickup)

**Tool:** ClickUp





Coding Standard Documentation. Detecting and Enforcing Coding Standards **(e.g. Checkstyle)**

**Tool:** JavaScript Code Standars

Applied Coding Practices and standards

4.1 Meaningful names Pronounceable and searchable names

One word per concept

Intention-Revealing Names

4.2 Well written functions Small

One thing

Less than 3 arguments

4.3 Good comments Explanation

Informative

4.4 Formatting Indentation

4.5 Error handling Try catch

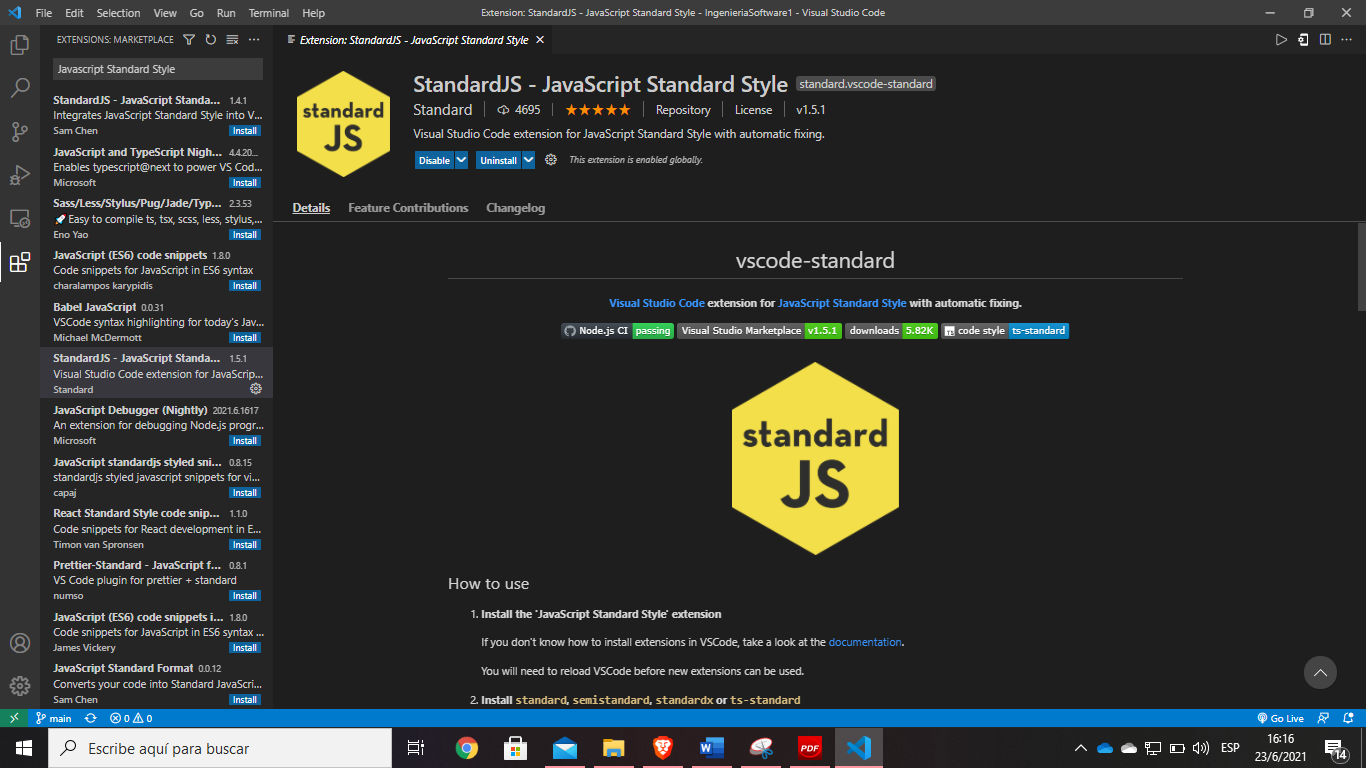
Don’t return or pass null

4.6 Don’t repeat yourself

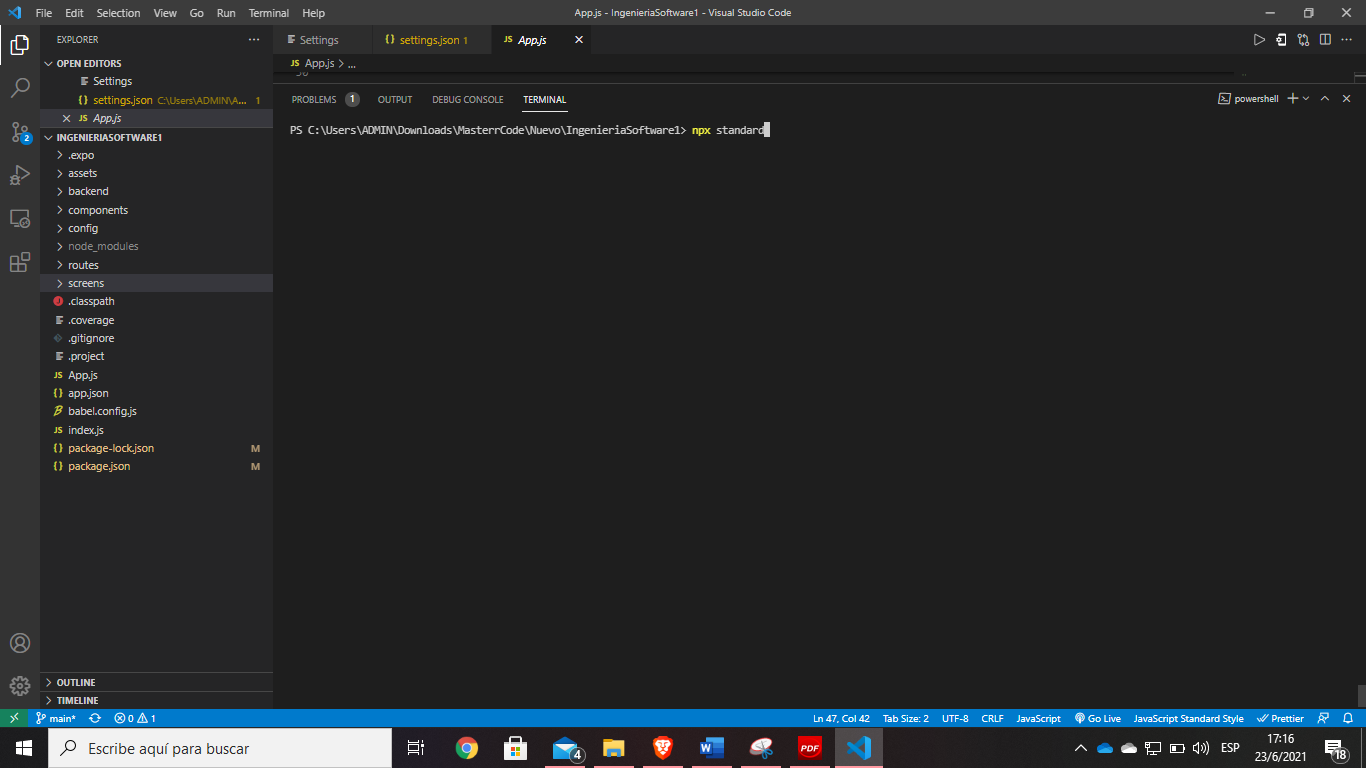
4.7 Avoid deep nesting

4.8 Limit line length

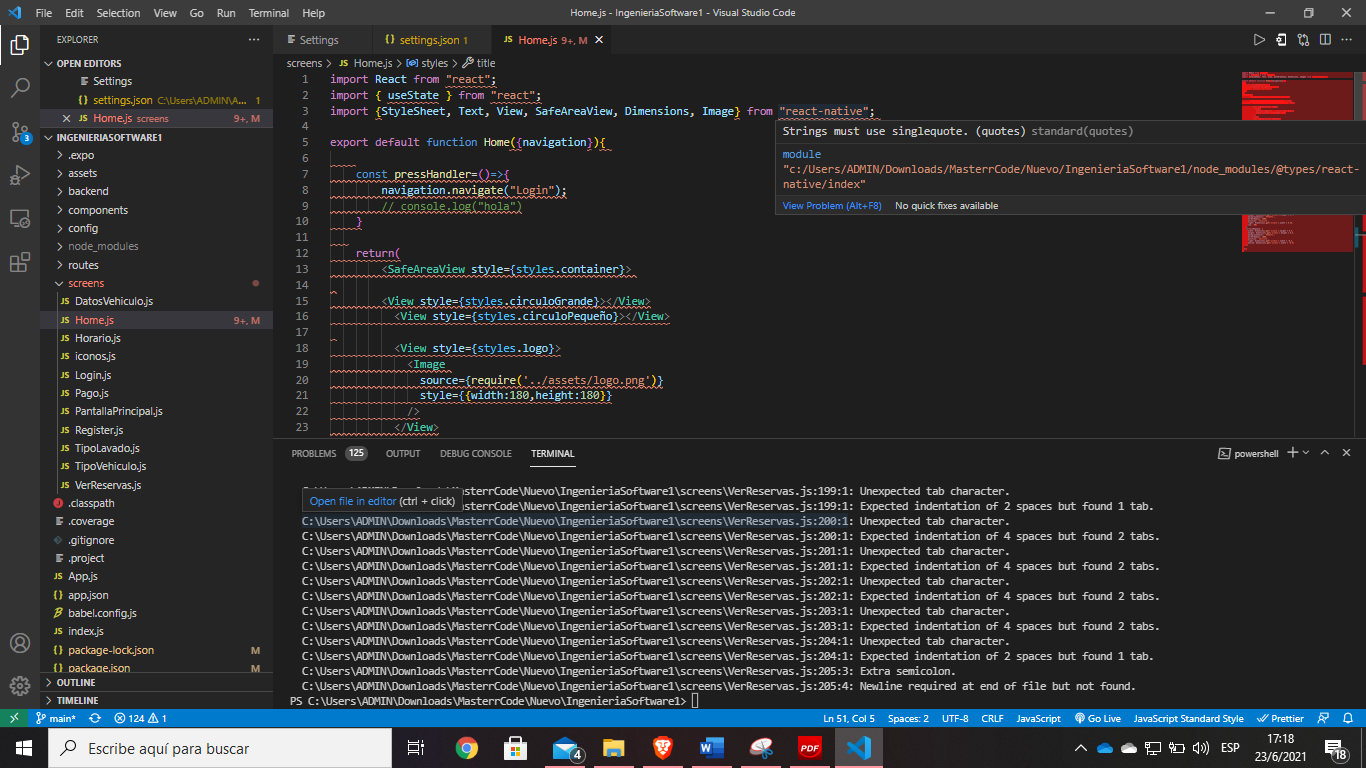
4.9 File and folder organization



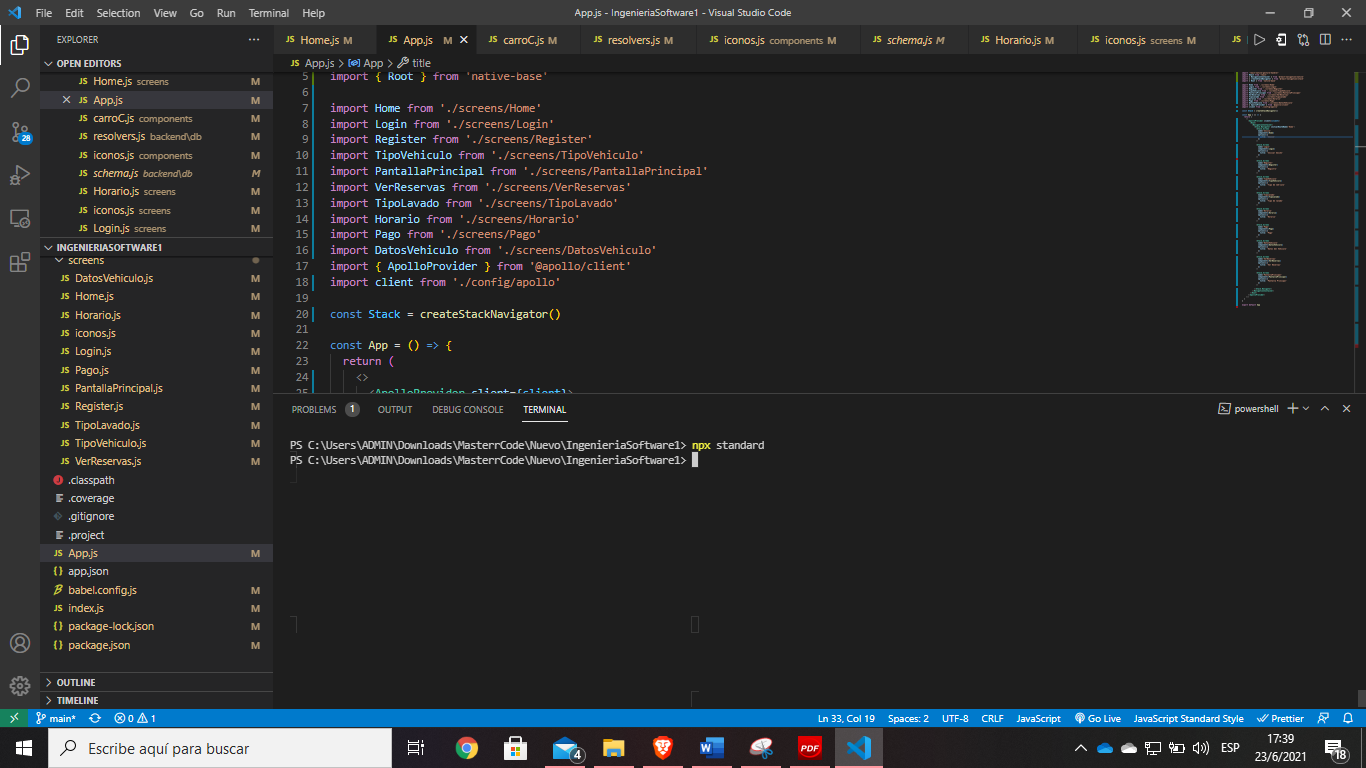
Run the tool



Code standards errors found



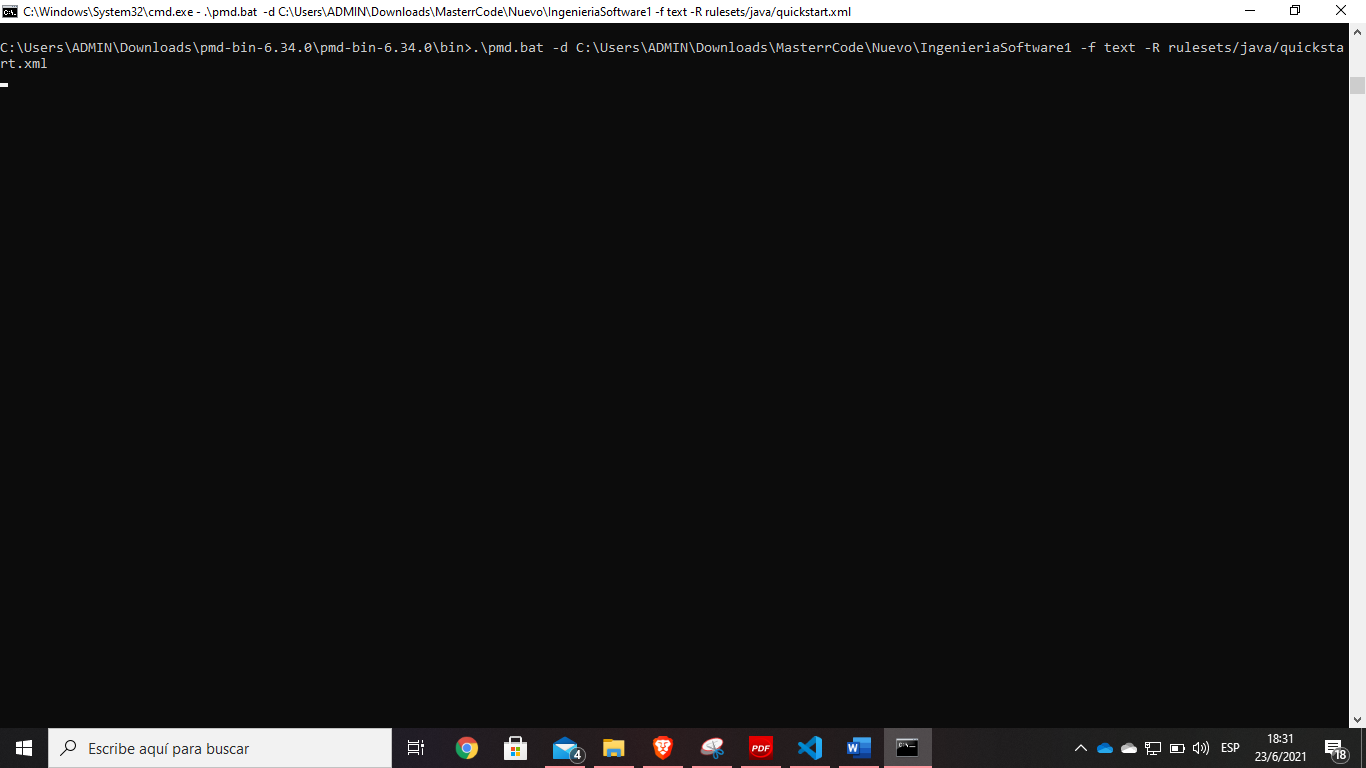
Finally, no errors in the code format



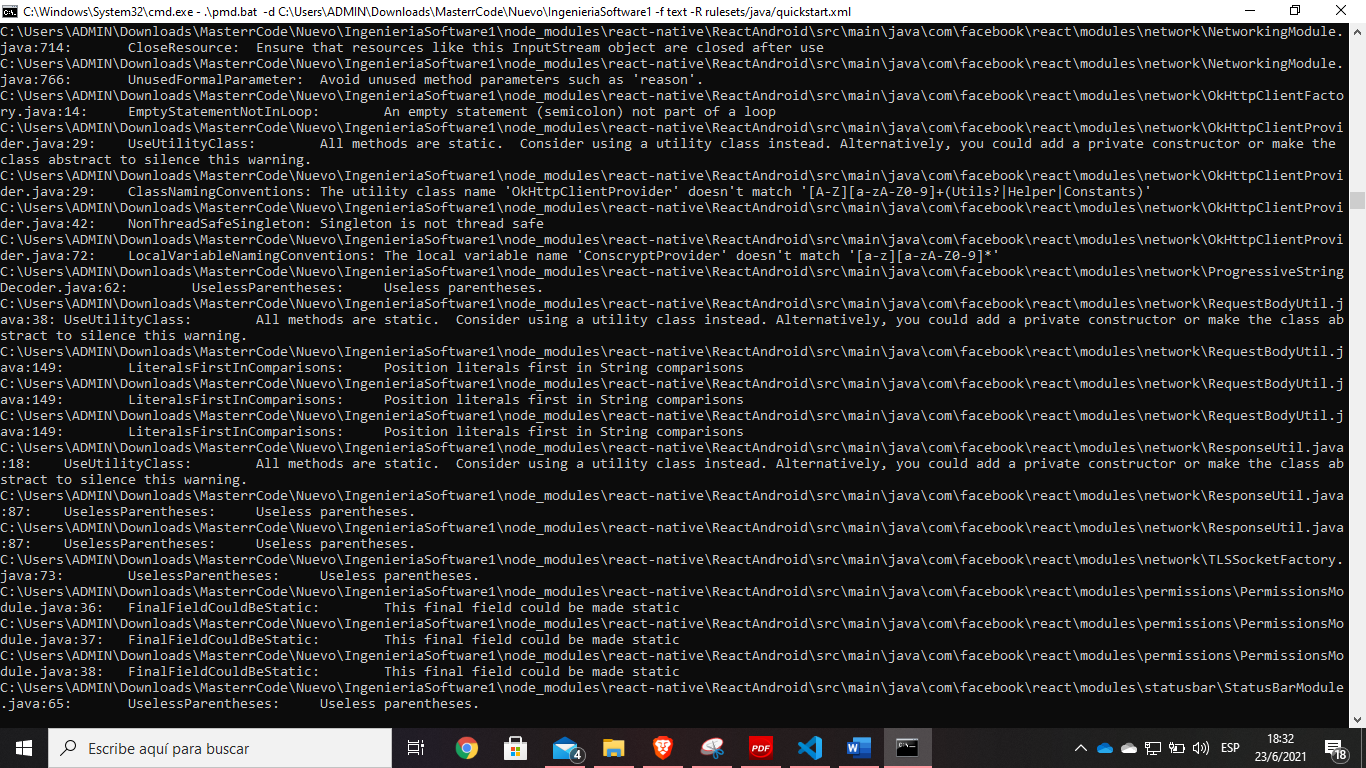
# Preemptive Error Detection (e.g. PMD)

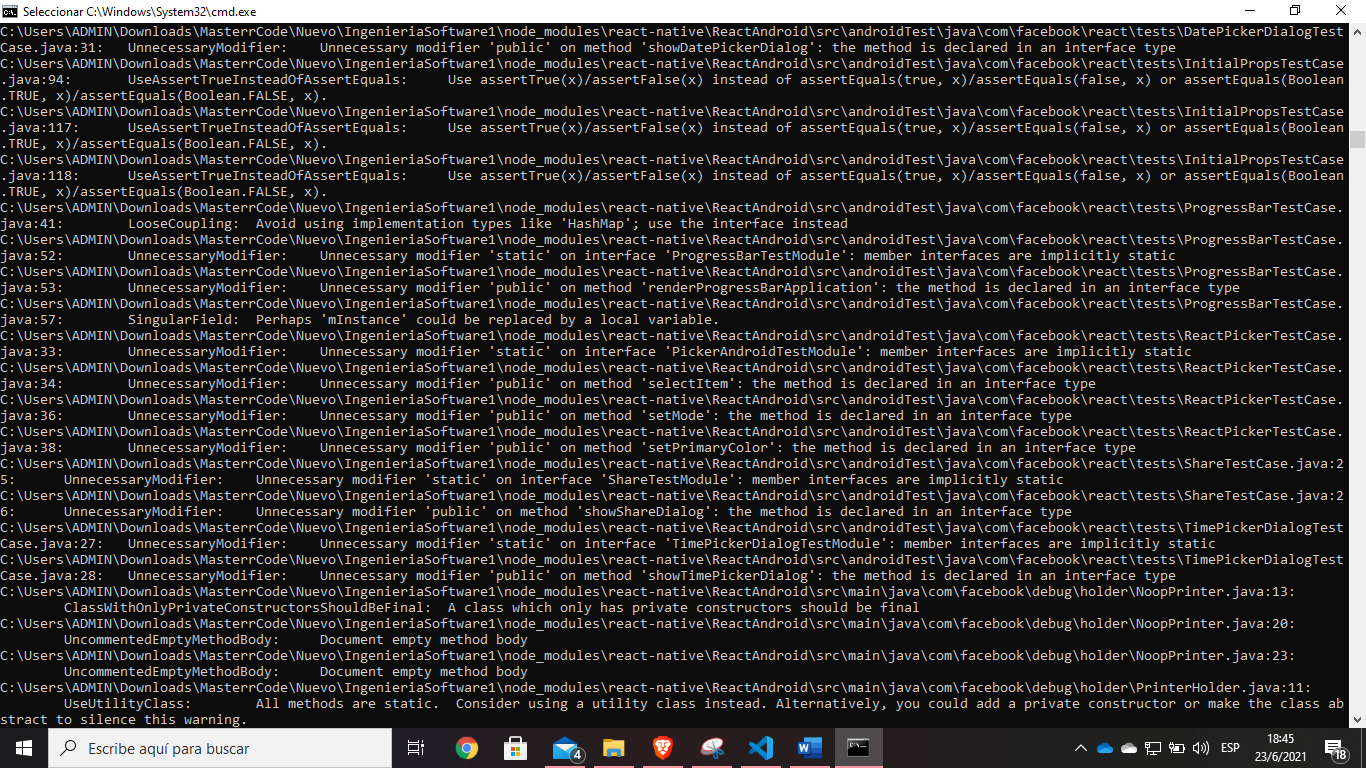
**Code static analizer**

**Tool**: PMD



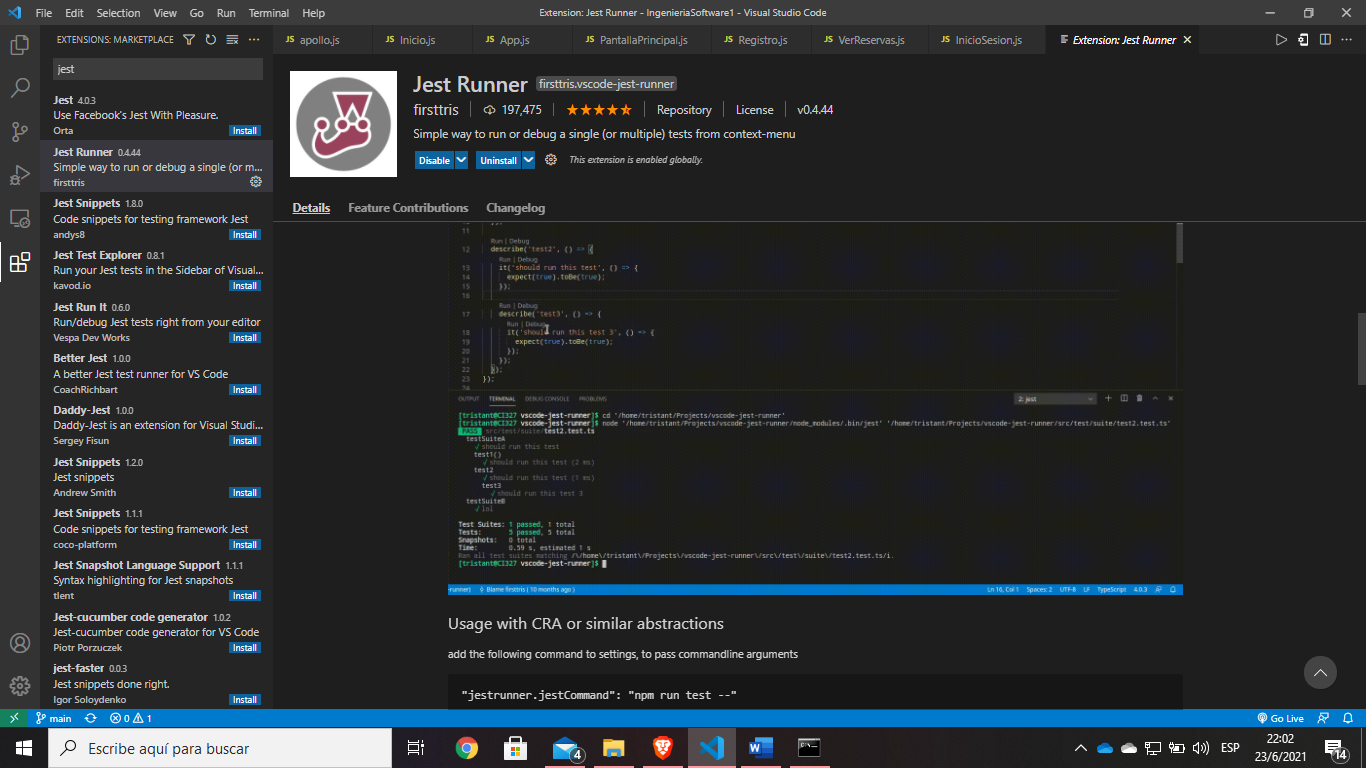
Errors found

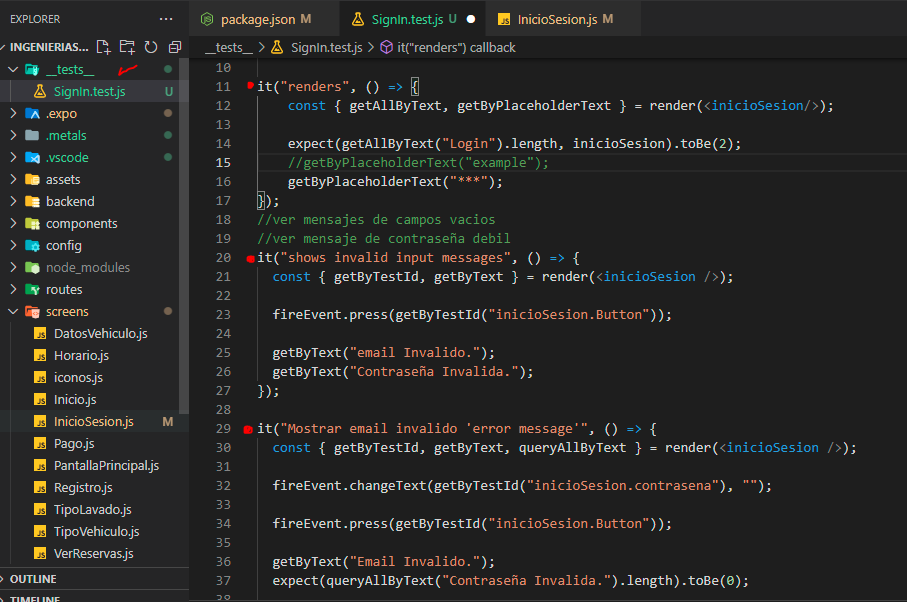




# Definition and Execution of Test Cases (e.g. JUnit)

**Tool**: Jest (Unit testing)





# System Development Guide

**Requirements:**

-React Native - Expo

npm install -g expo-cli

-Install MongoDB

-Install GraphQL

npm install --save graphql

-Install Apollo Client and Apollo Server

npm install -g apollo

**To execute:**

To run Frontend:





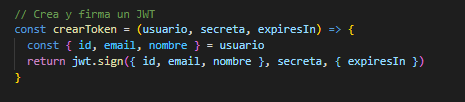
To run Backend:

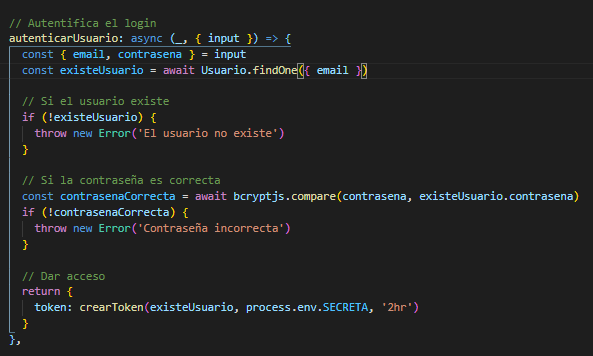




# Extra: Use of an Authentication and Access-Control Framework (e.g. Spring Security)

Tool: JSON Web Token Authentication and bcryptjs for hashing passwords





# Video showing the sprint review process