ODSOFT 2022/2023 – Project Part Two self-assessment — TEAM ID: G202

Instructions: Please mark put an X on the non-greyed column that better reflects your project's item achievement. Fill the gaps (_____) when appropriate.
Analysis Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently 50%; 3 – Achieves partially, includes justification for main options; 4 – Achieves completely with justifications and alternatives.
Requirements Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely including advanced topics.

Description	Analysis					Requirements				
	0	1	2	3	4	0	1	2	3	4
2.1 Base Pipeline – Overall Grading									X	
• Each concern has, at least, one issue in Bitbucket										X
• The overall pipeline should be designed. This includes a high-level design for all the concerns					X					
• The design should include a description of the process, including the Git organization (i.e., branching model) model to be adopted and how it relates to the pipeline;				X						
• The application should be completed with the specific features for the project:										
○ Supplier concept; (or entity 1);									X	
○ Customer concept; (or entity 2);									X	
○ Helpdesk Ticket concept; (or entity 3);									X	
○ Product Category concept; (or entity 4);									X	
○ Administrator concept; (or entity 5);									X	
○ Helpdesk Ticket cannot be associated to different customers at the same time; (or constraint 1);									X	
○ Suppliers must supply at least one category of products: (or constraint 2);									X	
○ Product Categories are only created by Administrators; (or constraint 3);									X	
• Unit tests are implemented, Unit Tests Report and Coverage Report are published on Jenkins;				X						X
• Pit Mutation tests (or similar) are executed, and Mutation Coverage Report is published on Jenkins;				X						X
• The application (e.g. .jar/war file) is deployed to a pre-configured staging Server running offsite (e.g., Tomcat Server instance)				X						X
• An automatic smoke test (curl) is performed after staging deployment				X						X
• UI Acceptance Manual Tests are performed (send email notification and pipeline waits for manual test)				X						X
• A tag on the repository with the Jenkins build number and status is pushed				X						X
• Implement the pipeline for the specifics of this concern. The base pipeline implementation should also include all the features addressed in Project Assignment – Part One, using parallel stages when suitable. **Which steps are run in parallel?**									X	
○ Check(Checkstyle and spotbugs), Initializing Staging Environment, Mutation Test and Unit Test;			X						X	
○ Integration Test and End to End Tests;			X						X	
○ JMeter and Cucumber;			X						X	
○ Deploy to Production, Convert Markdown to Pdf and Javadoc;			X						X	
• The pipeline works both on Windows and Linux operating systems				X						X
• Classify your final project according to the Maturity Model and support your decision.				X						

ODSOFT 2022/2023 – Project Part Two self-assessment   TEAM ID: G202

Instructions: Please mark put an X on the non-greyed column that better reflects your project's item achievement. Fill the gaps (_____) when appropriate.
Analysis Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently 50%; 3 – Achieves partially, includes justification for main options; 4 – Achieves completely with justifications and alternatives.
Requirements Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely including advanced topics.

Description	Analysis					Requirements				
	0	1	2	3	4	0	1	2	3	4
2.2 Documentation and Database – Overall Grading | Student #1220257 Name: <u>João Gonçalo Pinho</u>				X					X	
• The specific stages for this concern of the pipeline should be designed				X						
• Implement the pipeline for the specifics of this concern. Analysis results must be published and explicitly discussed in the project report;				X						X
• The application should have a persistence layer that must use a relational database. You may choose the technology to use for this non-functional requirement; **The application should use an SGBD running on a different system than the CRM web application** using containerisation (e.g., Docker images). o **Used SGBD:** <u>PostgreSQL</u>					X					X
• Generate the Project Report PDF file from your project's Readme.md. You may choose the technology to use o **Used technology to generate PDF:** <u>fntsoftware.gradle.markdown-to-pdf</u>					X					X
• Generate and archive the project's Moodle zip submission file, containing the Project Report PDF, the Jenkinsfile and the generated artefacts for the build (e.g., Javadoc, reports, etc – excluding all temporary files)					X					X
o Jenkinsfile										X
o Project Report PDF										X
o Artefacts (please detail)										X
o Source code;										X
o Deployed war file;										X
o All the reports generated;										X
o Dockerfile and docker-compose;										X
o Webdrivers;										X
o _____;										
o _____;										
o _____;										
o _____;										
o _____;										
• The pipeline works both on Windows and Linux operating systems					X					X
• **Advanced requirement:** The CRM application should be adapted to foreseen third-parties version releases. In this case, explicitly document and adapt the current CRM solution for supporting Gradle's latest release (v. 7.5.2) and latest JDK's release (v. 11 or superior).	X					X				
• **Advanced requirement:** Use a migration tool to maintain database upgrades and downgrades, such as Flyway, Liquibase or other migration tool				X					X	

ODSOFT 2022/2023 – Project Part Two self-assessment   TEAM ID: G202

Instructions: Please mark put an X on the non-greyed column that better reflects your project's item achievement. Fill the gaps (_____) when appropriate.
Analysis Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently 50%; 3 – Achieves partially, includes justification for main options; 4 – Achieves completely with justifications and alternatives.
Requirements Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely including advanced topics.

Item	0	1	2	3	4	0	1	2	3	4
2.3 Code Quality and Integration Tests – Overall Grading | Student #1220256 Name: João Bruno Macedo da Rocha			X						X	
• The specific stages for this concern of the pipeline should be designed				X						
• Implement the pipeline for the specifics of this concern. Analysis results must be published and explicitly discussed in the project report;				X					X	
• The pipeline should include a "check" on the code quality of the project by using Checkstyle (or similar tool). Settings for code quality should be defined, including the thresholds for the build health. Analysis results must be published; **Used Thresholds:** healthy: 10, unhealthy: 400				X						X
• The pipeline should include a "check" on the code quality of the project by using SpotBugs (or similar tool). Settings for code quality should be defined, including the thresholds for the build health. Analysis results must be published; **Used Thresholds:** threshold: 20 (unstable) threshold: 400 (failure)				X						X
• Integration tests should cover the specific features of the project. Integrations Tests Report and Coverage Reports should be published on Jenkins. Settings for integration test coverage should be defined, including the thresholds for the build health. The build should fail if coverage degrades more than the delta thresholds that should be configured; **Used Thresholds:** If it has less than 20% of coverage fails.				X						X
• The pipeline works both on Windows and Linux operating systems				X						X
• **Advanced requirement:** Test the CRM application for at least two different HTTP Servers versions. There should be a configuration file in the repository stating which versions of the webserver must be used and running the tests against more than two different versions should be supported:	X					X				
o Configuration file						X				
o Support more than two different HTTP Servers versions						X				
2.4 Functional and Smoke Testing – Overall Grading | Student #1220271 Name: Nuno Leite			X						X	
• The specific stages for this concern of the pipeline should be designed				X						
• Implement the pipeline for the specifics of this concern. Analysis results must be published and explicitly discussed in the project report;			X						X	
• Acceptance tests with Cucumber and Selenium should cover the specific features of the project. Acceptance Tests Report and Coverage Reports should be published on Jenkins;			X						X	
• These tests should be executed against an isolated staging environment (e.g., docker container) with the application (do not forget that the application now has a database!);			X						X	
• The image for this docker container should be published in the docker hub			X							X
• The smoke test should be executed against a docker container with the application (do not forget that the application now has a database!);			X						X	
• The pipeline works both on Windows and Linux operating systems			X							X
• **Advanced requirement:** Run the CRM's application acceptance tests using at least two different browsers. A configuration file, in the repository should state which browsers and versions must be used to run the acceptance tests. Use more than two different browser/versions should be supported:	X					X				
o Configuration file						X				

Instructions: Please mark put an X on the non-greyed column that better reflects your project's item achievement. Fill the gaps (_____) when appropriate.
Analysis Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently 50%; 3 – Achieves partially, includes justification for main options; 4 – Achieves completely with justifications and alternatives.
Requirements Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely including advanced topics.

Item	Analysis 0	1	2	3	4	Req 0	1	2	3	4
o Support more than two different browsers/versions in the same build				X						
**2.5 Non-Functional Testing – Overall Grading	Student #1220285 Name: Tiago Lacerda**				X					X	
• The specific stages for this concern of the pipeline should be designed			X							
• Implement the pipeline for the specifics of this concern. Analysis results must be published and explicitly discussed in the project report;			X						X	
• End 2 End (E2E) tests should be tested to evaluate non-functional requirements (i.e., system performance, throughput, and capacity) for two different usage scenarios (e.g., Suppliers and Helpdesk Tickets)			X						X	
• Non-functional tests should be performed to check the system capacity using JMeter (or any similar) and a report should be published on Jenkins			X						X	
• The non-functional tests should be executed against a docker container with the application (do not forget that the application now has a database!)				X						X
• The pipeline works both on Windows and Linux operating systems;				X						X
• **Advanced requirement**: Run the CRM's non-functional requirements tests using at least two different HTTP servers (or major release versions of the same server). A configuration file in the repository should state which server and versions must be used to run the tests. More than two different server/versions should be supported in the same build	X							X		
o Configuration file								X		
o Support more than two different browsers/versions in the same build								X		
**2.6 Continuous Deployment – Overall Grading	Students #1220257 and #1220256 Name: João Gonçalo Pinho and João Bruno Macedo da Rocha**				X					X	
• The specific stages for this concern of the pipeline should be designed			X							
• Implement the pipeline for the specifics of this concern. Analysis results must be published and explicitly discussed in the project report;				X						X
• This should be the last stage of the pipeline. The idea is to simulate a live deployment to production;				X						X
• This should simulate the production stage using a docker container;				X						X
• It should be able to deploy the binaries/resources and upgrade the database if necessary;				X						X
• It should be able to recover to the last "good" state if the upgrade fails;				X						X
• There should be a tag in the git repository related to the release and it also should be possible to identify this version from the deployed artefacts;			X						X	
• The pipeline works both on Windows and Linux operating systems				X						X
• **Advanced requirement:** Perform a Blue/Green deployment of CRM application. For this task, the production environment should be running on at least three different machines i.e., the front-end tomcat proxy (or similar), the green production machine, and the blue production machine (do not forget that the application now has a database that should also be running on a different machine!).	X							X		

Instructions: Please mark put an X on the non-greyed column that better reflects your project's item achievement. Fill the gaps (_____) when appropriate.
Analysis Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently 50%; 3 – Achieves partially, includes justification for main options; 4 – Achieves completely with justifications and alternatives.
Requirements Grading: 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely including advanced topics.

Project Report – Overall Grading			X				X	
• Evidence of continuous building of the pipeline (build history with increasing stages)							X	
• Specifies the details of the design							X	
• Specifies the technical details of the implementation							X	
• Provides analysis of possible alternatives and the justification of the options.				X				
Teamwork Effort								
o Student #1220257; Accomplishment %30; (The sum of all percentages should equal 100%)								
o Student #1220285; Accomplishment %25; (The sum of all percentages should equal 100%)								
o Student #1220256; Accomplishment %25; (The sum of all percentages should equal 100%)								
o Student #1220271; Accomplishment %20; (The sum of all percentages should equal 100%)								