

# ODSOFT 2020/2021: CA – Part 2 self-assessment

TEAM ID: G\_001

**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.

**Implementation Grading:** 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely.

Description	Analysis					Implementation				
	0	1	2	3	4	0	1	2	3	4
<b>Overall Team Grading</b>									X	
• Each component has, at least, one issue in Bitbucket									X	
• Bitbucket issues are closed and refer to the commit									X	
• All source code is committed										X
• Commit messages are clear and well formed									X	
• The repository readme.md is well structured, clear and coherent with the work done									X	
• Issues have a clear explanation on the scope and differences with other issues									X	
• All pipelines must be able to run on any of the student's machines									X	
• An evaluation report on the performance of the CA component's, comparing the different execution times, was written (report.md)									X	
• Other issues:										
○ ( )										
○ ( )										
○ ( )										
○ ( )										
○ ( )										
○ ( )										
○ ( )										
○ ( )										

# ODSOFT 2020/2021: CA – Part 2 self-assessment

TEAM ID: G\_\_001\_\_

**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.

**Implementation Grading:** 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely.

Description	Analysis					Implementation				
	0	1	2	3	4	0	1	2	3	4
<b>Component #1</b>										
• A sketch or schematic of the pipeline with a small description is provided				x						
• A Jenkins Pipeline using the “ <b>Promoted Build</b> ” or “ <b>Build Pipeline</b> ” plugin, performing a <b>sequential build</b> , is configured and described				x						x
1. <b>Repository Checkout</b> - checkout the GIT repository										x
2. <b>War file</b> - the .war file is built and published on Jenkins										x
3. <b>Javadoc</b> - the Javadoc is built and published on Jenkins										x
4. <b>Unit Tests</b>										
a. The Unit Tests are executed and explained			x					x		
b. The Unit Tests Report is generated and published on Jenkins									x	
c. The Unit Tests Coverage Report is generated and published on Jenkins									x	
5. <b>Integration Tests</b>										
a. The Integration Tests are executed and explained			x					x		
b. The Integration Tests Report is generated and published on Jenkins									x	
c. The Integration Tests Coverage Report is generated and published on Jenkins									x	
6. <b>Mutation Tests</b>										
a. The Mutation Tests are executed and explained			x					x		
b. The Mutation Coverage Report is generated and published on Jenkins									x	
7. <b>System Test</b>										
a. The application (.war file) is deployed to a pre-configured Tomcat Server instance (Tomcat Server is running offsite)			x						x	
b. An automatic smoke test is performed			x						x	
8. <b>UI Acceptance Manual Tests</b>										
a. An email notification is sent, regarding the successful execution of all the previous tests and asking to perform a manual test				x					x	
b. An UI Acceptance Manual Test is executed in order to cancel or proceed with the pipeline									x	
c. The pipeline must wait for a user manual confirmation on Jenkins									x	
9. <b>Continuous Integration Feedback</b> - a tag to the repository with the Jenkins build number and status is published								x		
• Jenkins configuration files (Build Promotions and Build Pipelines jobs) are in the repository									x	
• The overall “build” fails if any of the stages fail				x					x	
• Each gradle task is executed only once in the pipeline			x						x	

# ODSOFT 2020/2021: CA – Part 2 self-assessment

TEAM ID: G\_001\_

**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.

**Implementation Grading:** 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely.

Description	Analysis					Implementation				
	0	1	2	3	4	0	1	2	3	4
<b>Component #2</b>										
• A sketch or schematic of the pipeline with a small description is provided				x						
• A Jenkins Pipeline using the “ <b>Promoted Build</b> ” or “ <b>Build Pipeline</b> ” plugin, performing a <b>parallel build</b> , is configured and described				x					x	
1. <b>Repository Checkout</b> - checkout the GIT repository										x
2. <b>War file</b> - the .war file is built and published on Jenkins										x
3. <b>Javadoc</b> - the Javadoc is built and published on Jenkins										x
4. <b>Unit Tests</b>										
a. The Unit Tests are executed and explained			x						x	
b. The Unit Tests Report is generated and published on Jenkins									x	
c. The Unit Tests Coverage Report is generated and published on Jenkins									x	
5. <b>Integration Tests</b>										
a. The Integration Tests are executed and explained			x						x	
b. The Integration Tests Report is generated and published on Jenkins									x	
c. The Integration Tests Coverage Report is generated and published on Jenkins									x	
6. <b>Mutation Tests</b>										
a. The Mutation Tests are executed and explained			x						x	
b. The Mutation Coverage Report is generated and published on Jenkins									x	
7. <b>System Test</b>										
a. The application (.war file) is deployed to a pre-configured Tomcat Server instance (Tomcat Server is running offsite)				x					x	
b. An automatic smoke test is performed			x					x		
8. <b>UI Acceptance Manual Tests</b>										
a. An email notification is sent, regarding the successful execution of all the previous tests and asking to perform a manual test				x					x	
b. An UI Acceptance Manual Test is executed in order to cancel or proceed with the pipeline									x	
c. The pipeline must wait for a user manual confirmation on Jenkins									x	
9. <b>Continuous Integration Feedback</b> - a tag to the repository with the Jenkins build number and status is published								x		
• Jenkins configuration files (Build Promotions and Build Pipelines jobs) are in the repository									x	
• The overall “build” fails if any of the stages fail				x					x	
• Each gradle task is executed only once in the pipeline			x						x	

# ODSOFT 2020/2021: CA – Part 2 self-assessment

TEAM ID: G\_\_ \_\_

**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.

**Implementation Grading:** 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely.

Description	Analysis					Implementation				
	0	1	2	3	4	0	1	2	3	4
<b>Component #3</b>										
• A sketch or schematic of the pipeline with a small description is provided				x						x
• A Jenkins Pipeline using a <b>Scripted Jenkinsfile</b> , performing a <b>sequential build</b> , is configured and described				x					x	
1. <b>Repository Checkout</b> - checkout the GIT repository										x
2. <b>War file</b> - the .war file is built and published on Jenkins										x
3. <b>Javadoc</b> - the Javadoc is built and published on Jenkins										x
4. <b>Unit Tests</b>										
a. The Unit Tests are executed and explained				x						x
b. The Unit Tests Report is generated and published on Jenkins									x	
c. The Unit Tests Coverage Report is generated and published on Jenkins									x	
5. <b>Integration Tests</b>										
a. The Integration Tests are executed and explained				x					x	
b. The Integration Tests Report is generated and published on Jenkins									x	
c. The Integration Tests Coverage Report is generated and published on Jenkins									x	
6. <b>Mutation Tests</b>										
a. The Mutation Tests are executed and explained				x					x	
b. The Mutation Coverage Report is generated and published on Jenkins									x	
7. <b>System Test</b>										
a. The application (.war file) is deployed to a pre-configured Tomcat Server instance (Tomcat Server is running offsite)				x					x	
b. An automatic smoke test is performed				x					x	
8. <b>UI Acceptance Manual Tests</b>										
a. An email notification is sent, regarding the successful execution of all the previous tests and asking to perform a manual test				x					x	
b. An UI Acceptance Manual Test is executed in order to cancel or proceed with the pipeline									x	
c. The pipeline must wait for a user manual confirmation on Jenkins									x	
9. <b>Continuous Integration Feedback</b> - a tag to the repository with the Jenkins build number and status is published								x		
• Jenkins configuration files (Build Promotions and Build Pipelines jobs) are in the repository										x
• The overall "build" fails if any of the stages fail				x					x	
• Each gradle task is executed only once in the pipeline				x					x	

# ODSOFT 2020/2021: CA – Part 2 self-assessment

TEAM ID: G\_\_001\_\_

**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.

**Implementation Grading:** 0 – No Submission; 1 – Attempt; 2 – Achieves Sufficiently; 3 – Achieves with Minor Faults; 4 – Achieves completely.

Description	Analysis					Implementation				
	0	1	2	3	4	0	1	2	3	4
<b>Component #4</b>										
• A sketch or schematic of the pipeline with a small description is provided				x					x	
• A Jenkins Pipeline using a <b>Scripted Jenkinsfile</b> , performing a <b>parallel build</b> , is configured and described				x					x	
1. <b>Repository Checkout</b> - checkout the GIT repository										x
2. <b>War file</b> - the .war file is built and published on Jenkins									x	
3. <b>Javadoc</b> - the Javadoc is built and published on Jenkins									x	
4. <b>Unit Tests</b>										
a. The Unit Tests are executed and explained				x						x
b. The Unit Tests Report is generated and published on Jenkins									x	
c. The Unit Tests Coverage Report is generated and published on Jenkins									x	
5. <b>Integration Tests</b>										
a. The Integration Tests are executed and explained				x					x	
b. The Integration Tests Report is generated and published on Jenkins									x	
c. The Integration Tests Coverage Report is generated and published on Jenkins									x	
6. <b>Mutation Tests</b>										
a. The Mutation Tests are executed and explained				x					x	
b. The Mutation Coverage Report is generated and published on Jenkins									x	
7. <b>System Test</b>										
a. The application (.war file) is deployed to a pre-configured Tomcat Server instance (Tomcat Server is running offsite)				x					x	
b. An automatic smoke test is performed				x					x	
8. <b>UI Acceptance Manual Tests</b>										
a. An email notification is sent, regarding the successful execution of all the previous tests and asking to perform a manual test				x					x	
b. An UI Acceptance Manual Test is executed in order to cancel or proceed with the pipeline									x	
c. The pipeline must wait for a user manual confirmation on Jenkins									x	
9. <b>Continuous Integration Feedback</b> - a tag to the repository with the Jenkins build number and status is published								x		
• Jenkins configuration files (Build Promotions and Build Pipelines jobs) are in the repository										x
• The overall "build" fails if any of the stages fail				x					x	
• Each gradle task is executed only once in the pipeline				x					x	