

APPENDIX D – SUMMARY OF THE NARRATIVE BASED ON THE D&O MODEL

Phases	Learning objective	Concept Orientation	Concept Practice	Learning checkpoint
Phase #1	Learning about Knowledge Sharing & Source code management (SCM)	The orientation of the concepts is characterized by the explanation of the concepts of Culture of collaboration, Sharing knowledge and GCF, where GCF is premised on the explanation of Version Control.	1) The practice will begin by dividing the students into groups where there will be a presentation. 2) After the presentation there will be a stand-up meeting to discuss the demands of the PO. 3) The groups should use the concept of pair programming to obtain explicit and tacit knowledge. 4) Using the concepts of collaborative culture and knowledge sharing, they should fork the project. 5) Install the Nodemon and Express packages. 6) Complete the demands requested by the PO. 7) At the end of each demand, the Semantic Versioning standard and the Commits Convention must be followed.	1) Knowledge of the points reported by the PO. 2) Presentation of team members. 3) Pair programming 4) Sharing knowledge among peers 5) Have the version of the package.json file updated with the SemVer convention. 6) have made the Commits to the appropriate convention standard.
Phase #2	Learn about Build Process (BP), Continuous Integration (CI) and Deployment Automation	The concept orientation of the current phase groups together explanations of three concepts: Release Engineering (LE) Continuous Delivery (CD) Automation Test Automation (TA) Static Analysis (SA) Frequent and reliable release process Release Engineering Continuous Integration Deployment Pipeline Artifact Management Configuration Management Continuous Deployment Infrastructure as Code Virtualization, Containerization Cloud Services, Automation	1) Installation of the following packages: Mocha, Prettier and ESLint for test execution and static code analysis. 2) Creation of the YAML file. 3) Explanation of the YAM file and final changes. 4) Install Fly.io. 5) Create the secret key. 6) Inclusion of the code snippet in the YAML file for the automatic deployment of the application.	1) Installation of all the packages provided to build the test cases. 2) Construction of the automated test files and success in all cases. 3) Prettier and ESLint installation and configuration 4) Have built the file that will make it possible to build the CI pipeline 5) Achieving 100% success 6) Successful completion of all steps to install Fly.io 7) Add the necessary scripts to our yaml file to configure the CD. 8) Make one more commit to trigger the CI/CD pipeline and see the API deployment in Fly.io
Phase #3	Learn about Monitoring and Logging	Your orientation will cover: You built it, you run it After-hours support for devs Continuous monitoring Performance, Availability, Scalability Resilience, Reliability, Automation Metrics and Logse your definition and your definition	1) The practice will begin by downloading Sonarqube and configuring it as well as SonarScanner 2) The body-parser and morgan-body packages must be installed to use Logs. 3) Create a folder in the root of the project named "logs". 4) A set of constants must be added to the "index.js" file	1) Understanding the concepts of You built it, you run it 2) Understanding the concepts of continuous monitoring 3) Understanding the concepts of Metrics and having completed their respective practice. 4) Correct deployment of the packages and obtaining the log file.

Source: own author (2023)