

Background	Explore the range of technologies, methodologies and supporting tools that encapsulate all fundamental modules covered during training.
Objectives	Create a fully functioning inventory management system.
Scope	<ul style="list-style-type: none"> • Code fully integrated into a Version Control System using the feature-branch model: master/dev/multiple features. • A project management board with full expansion on user stories, acceptance criteria and tasks needed to complete the project. • A risk assessment which outlines the issues and risks faced during the project timeframe. • A relational database used to persist data for the project, containing the customers, products, orders, and orders_items tables. Relationships should be modelled using an ERD. • A functional application 'back-end', following best practices and design principles. • A build of the application, including any dependencies it might need, produced using an integrated build tool. • Unit tests for validation of the application, aiming for 80% coverage.
Constraints	<ul style="list-style-type: none"> • Version Control System: Git • Source Code Management: GitHub • Kanban Board: Jira • Database Management System: MySQL Server 5.7 (local or GCP instance) • Back-End Programming Language: Java • Build Tool: Maven • Unit Testing: JUnit
Assumptions	The customer address is not required to be stored across the system.
Risks	<ul style="list-style-type: none"> • Shortage of time. • Self-management issues. • Maintenance of cloud services. • Required line coverage is not met.
Deliverables	<ul style="list-style-type: none"> • Codebase • Unit tests (80% coverage) • Project management board • Git repository • Working application • Documentation • Presentation (15 mins)

