+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 09/21/2026

Sign Up

Ona Training: Development Environments and Tools

(formerly Gitpod)

3 days (21 hours)

Overview

Ona, formerly Gitpod, is a development platform that combines standardized, reproducible environments with AI agents capable of directly interacting with software projects.

This Ona training course will teach you how to create ready-to-use development environments to reduce manual configuration, speed up onboarding, and ensure a consistent environment for all your teams.

You'll learn how to configure your environments using devcontainer.json, automate dependency installation, testing, and service deployment, as well as integrate your Git repositories and your usual development tools.

The training will also enable you to leverage Ona Agents to understand, modify, test, and improve your projects. You'll learn how to provide the right context to agents, customize their instructions, and extend their capabilities with MCP-based tools and integrations.

By the end of this training, you'll be able to standardize your development environments, automate their setup, and integrate AI development agents into your workflows while managing access, secrets, and security policies.

Like all our training courses, this one will introduce you to **the latest stable version** of GitPod technology (0.5.2) and its new features.

Objectives

- Create and configure reproducible development environments with Ona
- Automate the installation, startup, and recurring tasks of a project
- Integrate Ona environments with Git repositories and existing development workflows
- Configure and use Ona Agents to support development tasks
- Customize the context and capabilities of agents using AGENTS.md and MCP
- Secure and standardize the use of environments and agents within a team

Target Audience

- Developers
- DevOps Engineers
- Platform Engineers
- Tech Leads
- Software and Cloud Architects

Requirements

- Basic knowledge of Git and version control systems
- Experience with a development environment and an editor such as VS Code
- Basic understanding of Docker and containers is recommended
- General knowledge of the software development lifecycle

Our Ona Training Program: Development Environments and Agents

[Day 1 - Morning]

Understanding cloud development environments

- Principles of cloud development environments and the evolution of development practices
- Understanding the challenges of reproducibility, isolation, and standardization of environments
- Comparing local environments, development containers, and remote environments
- The role of IDEs, code editors, and automated development environments
- Identifying the benefits for onboarding, security, and team productivity
- Hands-on workshop: Analyze an existing development environment and identify steps that can be standardized with Ona

[Day 1 - Afternoon]

Discover Ona and create your first environments

- Understand the evolution from Gitpod to Ona and the platform’s current positioning
- Explore Ona’s architecture and key components
- Create and launch an Ona Environment from an existing project
- Connect GitHub and GitLab repositories and manage the associated permissions
- Use an environment with VS Code and compatible development tools
- Hands-on workshop: Connect a Git repository and launch your first reproducible development environment

Configure reproducible environments

- Understand the role of devcontainer.json in defining the environment
- Select a development image and add the necessary tools to the project
- Customize the environment with Dev Container Features
- Configuring environment variables, dependencies, commands, and project settings
- Manage the ports and services required for an application to run
- Hands-on workshop: Build a complete, reproducible Ona environment with devcontainer.json

[Day 2 - Morning]

Automate project setup and execution

- Define project automations with automations.yml
- Automate dependency installation, compilation, testing, and service launch
- Organize reusable tasks and commands to simplify developers’ work
- Use Docker and containers to reproduce application dependencies
- Optimize environment startup and minimize manual onboarding tasks
- Hands-on workshop: Automate the installation, startup, and testing of an application in an Ona environment

[Day 2 - Afternoon]

Getting started with Ona development agents

- Understand how Ona Agents work and how they interact with development environments
- Assign tasks to an agent to understand, modify, test, and validate code
- Provide the agent with the necessary context to work effectively on an existing project
- Break down a complex request into tasks suitable for a development agent
- Review proposed changes and keep the developer in the validation loop
- Hands-on workshop: Assign a functional enhancement to an agent, review their changes, and validate the tests

Customize the agents’ context and capabilities

- Structure project instructions using AGENTS.md to communicate conventions and constraints
- Create reusable commands and instructions to standardize frequent tasks

- Provide agents with the necessary technical context without overloading their instructions
- Extend agent capabilities with MCP-based tools and integrations
- Organize collaboration among developers, environments, and agents on the same project
- Hands-on workshop: Configure project instructions and connect an external tool to a development agent

[Day 3 - Morning]

Orchestrate agent-based development workflows

- Identify tasks suitable for autonomous execution or execution supervised by an AI agent
- Run development tasks in parallel in isolated environments
- Combine code generation, testing, analysis, and correction within a single workflow
- Organize human reviews and verify changes before integration
- Integrate agents into Git practices and the existing development cycle
- Hands-on workshop: Build a complete agent-driven workflow from a change request to code validation

[Day 3 - Afternoon]

Secure and manage Ona environments

- Manage secrets and environment variables without exposing sensitive information in repositories
- Understand the principles of environment isolation and access control
- Organize users, teams, roles, and permissions according to the principle of least privilege
- Guide agents' actions with policies and guardrails tailored to projects
- Leverage authentication, logging, and auditing mechanisms for enterprise use cases
- Hands-on workshop: Defining access, secrets, and security rules for a shared environment used by developers and agents

Integrate Ona into the development cycle

- Standardize development environments at the team or organizational level
- Integrating Ona into existing Git workflows, tests, and CI/CD processes
- Define responsibilities between developers and support staff to maintain control and traceability
- Optimize onboarding and ensure a consistent environment across development, testing, and automation
- Define an adoption strategy combining standardized environments, AI agents, security, and governance
- Hands-on workshop: Build a complete Ona environment with automations and a development agent, then define its deployment process within a team

Target Audience

This training is intended for both individuals and companies—large or small—

wishing to train their teams in a new, advanced IT technology or to acquire specific business knowledge or modern methods.

Assessment at the Start of Training

The pre-training assessment complies with Qualiopi quality standards. Upon final registration, the learner receives a self-assessment questionnaire that allows us to evaluate their estimated proficiency level across various types of technologies, as well as their expectations and personal goals for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could pose challenges for monitoring and ensuring the smooth running of the training session.

Teaching Methods

Hands-On Training: 60% Practical, 40% Theoretical. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the instructor—supported by examples and reflection sessions—and group work.

Assessment

At the end of the session, a multiple-choice quiz is administered to verify that participants have successfully acquired the skills.

Certification

A certificate will be issued to each trainee who has completed the entire training program.