

RoostGPT Platform Feature Comparison

Supported Features Across Deployment Options

Feature Category	Web Interface (UI)	Command Line Interface (CLI)	IDE Plugin
Workspace Integration	Not Available	☐ Local workspace trigger supported	☐ Local workspace trigger supported
Version Control Integration	☐ Full Git activity support via web interface	☐ Full Git activity support via CLI commands	☐ Limited to local workspace operations
Notifications	☐ Real-time notifications when test generation completes	☐ No notification system	☐ No notification system
License Management	☐ Integrated license validation in UI	☐ API-based or license file validation	☐ API-based or license file validation
Event Monitoring	☐ Webhook support for Git events (PR creation, push, etc.)	☐ No Git event monitoring	☐ No Git event monitoring
User Management	☐ Role-based access control with separate admin and user views	△ Individual user basis on local system	△ Individual user basis on local system
Analytics & Reporting	☐ Comprehensive dashboard with detailed test generation reports and metrics	☐ No dashboard functionality	△ Basic UI showing added tests only
Test Execution Environment	☐ Isolated Docker containers for secure test execution and compilation verification	△ Local Mode: Manual dependency management Docker Mode: Requires Docker runtime	△ Manual dependency management required

Feature Category	Web Interface (UI)	Command Line Interface (CLI)	IDE Plugin
Ease of Use	☐ Intuitive web forms with one-click test generation	△ env file configuration or download env from web UI for CLI trigger	☐ Context menu integration with code lens support
Test Comparison Tools	☐ Built-in comparison view for multiple test scenarios	☐ No comparison functionality	☐ No comparison functionality
API Server	☐ Dedicated API server for programmatic access	☐ No API server	☐ No API server
History & Audit Trail	☐ Comprehensive event history and audit logging	△ CLI-based history for current user session	☐ No history management
Team Sharing	☐ Centralized credential sharing via connectors and team-wide test configuration distribution	☐ No sharing capabilities	☐ No sharing capabilities
Database Persistence	☐ Test configurations and connectors are stored in a central database	△ No persistence. Configuration is persisted locally in the system.	△ No persistence. Configuration is persisted locally in the system.
Stand-alone system	☐ UI can not be deployed as a stand-alone and will need to be hosted on a server	☐ CLI can be deployed as a stand-alone without any API Server or Hosted Server	☐ Plugin can be deployed as a stand-alone without any API Server or Hosted Server
Deployment	On Cloud and On Prem	On Linux, Windows, MacOS	From VSCode Marketplace and IntelliJ Disk install

Legend

- ☐ **Full Support:** Feature is fully implemented and available
- △ **Partial Support:** Feature available with limitations or manual setup required
- ☐ **Not Available:** Feature is not supported in this deployment option

Deployment Options

Web Interface (UI)

Best for: Teams requiring comprehensive project management, detailed analytics, and centralized administration.

Command Line Interface (CLI)

Best for: Developers preferring terminal-based workflows and automated CI/CD pipeline integration.

IDE Plugin

Best for: Individual developers seeking seamless integration within their development environment.

Application Programming Interface (API)

Best for: Automated CI/CD pipeline integration (limited API are exposed to access the hosted RoostGPT API server).

Revision #6

Created 8 September 2025 13:55:36 by Divyesh

Updated 8 September 2025 16:35:16 by Harish