

Autonomics

AMELIA's Autonomic IT Management Platform

Release Notes 3.11.0 (Document Version 1.0)

This AMELIA® documentation is copyright © 2025 Soundhound AI Amelia and their affiliated companies. All rights reserved.

This document is considered the confidential information of Soundhound AI Amelia and its affiliates. Disclosure to other parties is prohibited unless agreed to in a license or confidentiality agreement.

Trademarks, including AMELIA® and the AMELIA logo, are the intellectual property of Soundhound AI Amelia and its affiliated companies. Any other marks or intellectual property remain the property of their respective licensors or owners.

Table of Contents

1. RELEASE 3.11.0 3

1.1 RELEASE HIGHLIGHTS 3

1.1.1 ROLLBACK TO A PREVIOUS VERSION 3

1.1.2 DEPRECATED APIS 3

1.2 BUGS 3

1.3 EPIC 6

1.4 STORY 6

1.5 TASKS 8

1.6 SUB-TASKS 9

Document History

Author	Version	Date	Comments	Final Approval?
AMELIA Research & Development	1.0	April 2, 2025	Added tickets for 3.11.0 release	Yes

1. Release 3.11.0

This section briefly lists changes to AI Ops for this release.

1.1 RELEASE HIGHLIGHTS

1.1.1 Rollback to a previous version

Database schema is not compatible with previous releases. Once updated, it is not possible to rollback to the previous version without restoring from the backup.

1.1.2 Deprecated APIs

The automation executions API is deprecated:

```
GET /automaton-executions/state-execution/{id}
```

Please use instead:

```
GET /automaton-executions/state-execution/{executionId}/{executionPointerId}/{stateExecutionId}
```

The API call should be also updated in automation scripts from the current:

```
IPautomataExecutionerApiExecutionApi.getStateExecution(stateExecutionId);
```

to:

```
IPautomataExecutionerApiExecutionApi.getStateExecution(execution.id, executionPointer.id, stateExecutionId);
```

1.2 BUGS

- **Ensure "Fail To" Automations Cannot Be Deleted When In Use.** The system now prevents the deletion of automations if they are being used as a "Fail To" reference, similar to how it handles linked automations. This ensures that exporting automations through Content Packs doesn't fail due to missing dependencies. The change was implemented on the backend, which affects the way automations are managed and validated when exporting through Content Packs. There is no direct UI change visible to the user. The purpose of the change was to prevent export failures in Content Packs caused by the accidental deletion of automations that serve as "Fail To" references. This ensures data integrity and prevents users from encountering errors during export due to missing dependencies. This change primarily affects the backend functionality related to automation management and export processes in Content Packs. It ensures users have a smoother experience during export operations, but there are no direct customer-facing changes. (AP-30774)
- **Improved Content Pack Import Status Reporting in Multi-instance Setups.** A change was made to improve the status reporting for Content Pack imports in environments with multiple instances,

reducing the delay observed when the status changes to 'Completed'. The change affects the UI elements displaying import statuses of Content Packs, which can be found in the tasks or automations execution view, similar to the link provided for task executions. The purpose of this change was to address performance issues and ensure timely and accurate status updates for Content Pack imports in multi-instance environments to prevent miscommunication of task completion times. This change is backend-focused but affects customer-facing aspects as it directly impacts the timing and accuracy with which customers see updates in the UI regarding Content Pack import statuses. (AP-30875)

- **Fixed export failure for automata with UpdateTask state and specified clientId.** Automations with an UpdateTask state that have a clientId explicitly set can now be exported. The system will generate a warning but the export will proceed, instead of failing. The change affects the Content Pack export functionality within the exporting system interface. The change was implemented to address an issue where specifying a clientId in an UpdateTask state caused the export to fail. It ensures that exports do not terminate unexpectedly but instead proceed with a warning. This is a backend change affecting users of the automation export functionality who rely on clientId specifications within their automation tasks. (AP-30886)
- **Fixed issue with downtime notes not displaying correctly.** The issue that caused downtime notes to not display as expected was resolved. Now, when a note is added to a downtime entry, it appears followed by "Created from downtime" on a new line, rather than being replaced with just "Created from downtime." The change affects the note column in the downtime items list within the Autonomics Downtime page. The purpose of this change was to correct a bug reported by users where notes added to downtimes were not displayed properly, impacting visibility and record-keeping for users relying on those notes. This change is customer-facing, directly impacting users who create and manage downtime entries in the Autonomics platform. (AP-30648)
- **Fix for Incorrect Folder Elements Displayed from Integration Locksmith Link.** A bug was fixed that addressed the issue of incorrect folder elements being loaded when clicking on an integration locksmith link, ensuring that the correct elements are displayed based on their folders. The change was implemented in the integration section of the user interface specifically within the locksmith feature that handles element retrieval based on folder selection. The purpose of the change was to resolve a bug where elements with the same name in different folders were incorrectly loaded, leading to user confusion and potential data mishandling. This change affected the customer-facing aspect of the application, directly impacting users who interact with the integration locksmith feature. (AP-30772)
- **Resolution of Dependency Issue in Office365 Email Integration on Alite.** A missing dependency issue for the Office365 Email Integration on the Alite platform was identified and resolved, specifically the `jackson-module-parameter-names` module, which was causing integration failures. The change pertains to the backend configuration related to API service calls and dependency management, and does not directly affect any UI elements. The adjustment involves the system's internals, notably within integration execution settings found at the given integration URL. The purpose of this change was to fix the integration malfunction due to a dependency (`jackson-module-parameter-names`) issue, ensuring that the Office365 Email Integration functions correctly without unnecessary dependencies that do not impact other modules. This is a backend change affecting system administrators and developers involved with managing and executing integration services on the Alite platform. It ensures reliability in the integration processes but is not directly customer-facing. (AP-30902)
- **Fix for Audit Log Page Loading Issue.** The problem with the Audit Log page not loading was resolved, ensuring that it now loads correctly when accessed. The change occurred on the Audit Log page within

the user interface. The change was necessary to fix an issue where the Audit Log page was failing to load, likely caused by the addition of a new `{{SCHEDULING_EVENT}}` Entity Type. This change was customer-facing, as it affected users attempting to access the Audit Log page. (AP-30855)

- **Prevent Execution of Deleted Automations in IPautomata.** Deleted automations are now properly excluded from being triggered or executed within the system, including via workflows, manual triggers, or APIs. No changes were needed in the user interface, as this is a backend adjustment ensuring deleted automations are not triggered or executed. The purpose of this change is to prevent unintentionally executing automations that users have deleted, thereby maintaining system integrity and ensuring that actions are not performed using outdated or unwanted automation scripts. This change primarily affects the backend system processes, ensuring that deleted automations are filtered out, enhancing the reliability of automation management for backend users. (AP-30554)
- **Fix for Prioritization Issue with Stop Match On Others in IPautomata.** Matchers with a lower priority were being ignored when a higher priority matcher had "Stop Match On Others" enabled. This issue is now resolved to ensure that matching will only stop if the higher priority matcher actually triggers. This change does not directly affect the UI as it pertains to the underlying matching logic within the IPautomata component. The purpose of this change was to fix a logic error where lower-priority matchers were being unnecessarily bypassed regardless of whether the higher-priority matcher had triggered, ensuring correct match processing and rule application. This change affects the backend processing in the IPautomata component, ensuring accurate matcher prioritization and functioning for technical users who configure these matchers. (AP-30795)
- **Fixed Automata Trigger Issue for 'Active' Tasks.** The automata that were supposed to trigger on tasks with the "Active" status were fixed to function correctly. Previously, they failed to activate under the defined conditions. Though the change primarily impacts backend processing, it is associated with the task management and automation functionalities accessible via the Autonomics platform, which can be viewed under Task Details, Workflow Designer, and Automation Matchers sections. The purpose of this change was to resolve a defect where automata did not trigger as expected for tasks marked with the "Active" status. Ensuring that automation processes work as intended enhances the efficiency and reliability of the system. This change primarily affects backend processes, although it also has customer-facing impact as it ensures that automations for tasks are executed as expected for the end-users managing active tasks. (AP-30823)
- **Fix custom prompt compatibility with tcsh shell in UBS devices.** A compatibility issue with the tcsh shell's custom prompt was fixed, allowing it to work properly without timing out when connecting. This change is not visible in the UI; it occurs within the backend system handling shell prompts for specific device connections. The purpose was to resolve a timeout issue occurring during connections with devices using the tcsh shell, which was caused by an incompatible custom prompt configuration. This change affects the backend systems dealing with shell prompt configuration, specifically for devices using the tcsh shell. (AP-30717)
- **Fix NullPointerException in Host Command Execution with Disabled Traps.** The issue causing a NullPointerException when executing a Host command with traps disabled has been resolved. No visible changes in the UI. The fix was applied to the backend functionality that handles Host command execution. The change addresses a bug that caused a NullPointerException due to a null `trap_script`, ensuring that Host commands can execute correctly even when traps are disabled. This change affected the backend processing of Host commands, providing a more stable and predictable behavior for developers and systems relying on these processes, though it is not directly customer-facing. (AP-30930)

- **Icons Missing from Version 3.10.0 Announcement.** Icons were added to the announcement for version 3.10.0 to improve visual communication and user experience. The change occurred in the announcement section/interface for version 3.10.0. The purpose of this change was to rectify the absence of icons, ensuring the announcement is clear and visually engaging for users. This change is customer-facing and affects all users interacting with the version 3.10.0 announcement. (AP-30844)

1.3 EPIC

- **Enhanced ServiceNow CMDB Integration with Amelia Autonomics for Improved IT Operations.** The integration between ServiceNow's Configuration Management Database (CMDB) and Amelia Autonomics was enhanced. Users can now create, update, or remove integrations with their ServiceNow instance, execute one-time and automatic imports of Configuration Items (CIs), and manage ServiceNow CMDB Class Names and CI field mappings more efficiently. Additional functionalities include monitoring integration health, validating import queries, and synchronisation setups. The changes are reflected in several parts of the UI, including:
 - A new dashboard for monitoring the ServiceNow CMDB Integration health status.
 - Interfaces for creating, updating, deleting, and testing ServiceNow CMDB Integrations.
 - Screens for setting up and managing one-time and automatic CI import queries.
 - Interfaces for managing ServiceNow CMDB Class Names and CI field mappings.
 - Notification and message prompts for user actions and system statuses.

The purpose of this change is to streamline IT operations and enhance data accuracy by facilitating seamless integration between ServiceNow's CMDB and Amelia Autonomics. It aims to improve data synchronization, allow flexible configuration adapts to changing requirements, and empower users with tools for better integration management and operational efficiency. This change impacts both backend operations and is customer-facing. While backend changes ensure the integration functionalities operate smoothly, the primary effect is on customers who manage and interact with the ServiceNow CMDB Integration through the Amelia Autonomics platform. Users benefit from enhanced features and improved interface interactions for managing their system integrations. (AP-29974, AP-30654)

1.4 STORY

- **Added Native Java Integration for Enhanced Execution Flow.** A new integration type has been added for Native Java, which mandates the inclusion of two required fields: `*jar*` and `*execution_group*`. ALite will deploy this integration and use a custom ClassLoader to initiate flows defined within the integration jar. Additionally, the integration's /webhook endpoint will now delegate its invocation to execute the corresponding flow or route. This change primarily affects the backend and therefore does not involve direct UI alterations. It impacts how integrations are configured and executed within the ALite platform. The purpose of this change is to support a new integration type that facilitates Java-based execution flows. This new capability allows for more complex and custom operations by leveraging Java within the ALite system, thereby enhancing flexibility and performance. This change predominantly affects backend operations and developers who integrate or deploy custom Java flows within the system. There is no direct customer-facing UI impact. (AP-30358)
- **Implementation of UI for Creating Native Java Integration.** A new UI has been implemented to enable the creation of native Java integrations. This UI allows users to input integration details, upload a single

jar file, and manage metadata including descriptions and parameters. Additionally, users can view logs in a paginated list with search and date/time filtering capabilities after the integration is saved. The change occurred in the Integration section of the UI, where users can now access a new interface specifically designed for creating and managing native Java integrations. The purpose of this change was to provide users with the ability to create and manage Java integrations directly through a user-friendly interface. This feature extends the platform's integration capabilities by supporting native Java applications, improving overall usability and functionality. This change is customer-facing, affecting users who interact with the Integration UI to manage Java integrations. It enhances the user's ability to configure and deploy Java-based solutions through a straightforward and accessible interface. (AP-30427)

- **Enhanced Error Handling for Command Execution Timeouts in Amelia Autonomics Automation Engine.** This ticket implemented error handling for command execution timeouts within the Amelia Autonomics Automation Engine, introducing specific error codes (-729 for soft timeouts and -730 for hard timeouts) and capturing command outputs for better troubleshooting. The changes were implemented in the automation settings section of the UI, adding a setting to toggle the return of error codes and command outputs on timeouts. The purpose of this change is to allow developers to quickly identify and troubleshoot issues related to command execution timeouts in automation processes by providing specific error codes and command outputs. This enhances the diagnostic capabilities of developers working with the Amelia Autonomics Automation Engine. This change affects the backend, specifically developers working with the automation settings and the execution handling of commands in the Amelia Autonomics Automation Engine. (AP-29559, AP-30806)
- **Enhanced Execution Group Management for Amelia Autonomics.** The management of Execution Groups in Amelia Autonomics has been improved by clarifying the functionality, resolving terminology conflicts, and introducing a more user-friendly interface for configuring and managing execution locations for automations. The changes were implemented in the automation management section of the UI, specifically within the configuration and linking interface for Execution Groups. The purpose of this change is to simplify the configuration process, eliminate misunderstandings due to overlapping definitions, and provide users with the flexibility to manage automation execution locations more efficiently, whether in the Cloud or on customer-managed infrastructure. The change primarily affects the backend, but it also has implications for users interacting with the Amelia Autonomics platform, as it improves their experience when configuring and managing automations. (AP-30593)
- **Experimental Python Support Added to Automation Engine's Script Editor.** Experimental support for Python was introduced to the Automation Engine's Script Editor, allowing users to write and run scripts in Python. The update includes syntax recognition, formatting, and error detection for Python scripts. The change occurred within the Automation Engine's Script Editor/Widget interface, where users can now select Python as a scripting language. The purpose of this change is to provide users with more powerful and flexible scripting capabilities by leveraging Python's standard library, thus enhancing workflow efficiency and ease of use. This change is customer-facing, specifically affecting users who utilize the Script Editor in the Automation Engine for scripting purposes. (AP-30799)
- **Release Announcement for SoundHound Autonomics Version 3.11.0.** A new announcement file was created for SoundHound Autonomics version 3.11.0, outlining updates, features, and improvements included in this release. This change is reflected in the documentation or release notes section of the SoundHound Autonomics user interface, where users can view or download the announcement file. The purpose of this change was to inform users about the latest version of SoundHound Autonomics, providing them with valuable information about new features, bug fixes, and other enhancements

included in version 3.11.0. This change is customer-facing as it impacts users by providing them with the latest release information through the UI documentation section. (AP-30882)

1.5 TASKS

- **Redesign of Main Menu with Icon Updates and Grouping.** The main menu was redesigned to include distinct icons for each module and organizational grouping of menu items. The change happened in the Autonomics Core UI, specifically in the main menu section. The purpose of this change was to improve navigation by providing users with visually distinct icons and a more intuitive grouping of menu items, thereby enhancing user experience. This change is customer-facing, as it impacts users interacting with the Autonomics Core UI menu. (AP-30731)
- **Build Tool for Creating Fat JARs with Camel4 Flow Integration.** This ticket introduced a tool that generates a fat JAR from flow XML files and their dependencies. The resulting JAR includes all necessary components and is capable of running Camel4 flows, which can be deployed and executed by the Autonomics system. There is no change in the UI as this is a backend implementation focused on building and integrating fat JARs for Camel4 flows. The purpose of this change was to simplify the deployment of Camel4 flows by consolidating them into a single executable fat JAR. This approach aids in deploying and managing flow integrations more effectively within the Autonomics platform, ensuring smoother operation and triggering capabilities. This change affected the backend development process, particularly impacting developers working on Camel4 flow integration and deployment within the Autonomics environment. It is not customer-facing and does not impact end-users directly. (AP-30357)
- **Implementation of 'Shared Beans' in Camel4 Flow Integration.** Shared beans have been reviewed and implemented across existing Camel4 flows to ensure consistency and proper functionality. There is no direct change in the user interface as this update involves backend flow integration improvements. The change was initiated to streamline and enhance the efficiency of existing Camel4 flows by implementing shared beans. This ensures a more modular, maintainable, and reusable integration setup. This change primarily affects the backend, specifically benefiting developers and system integrators working with Camel4 flow integrations. (AP-30359)
- **Enhancements to gRPC Communication for Optimized Message Exchange and Logging.** The ticket introduced optimizations to the gRPC communication by reducing the number and size of messages exchanged between the platform and ALite. Additionally, logging at the WARN level was added for messages that exceed a certain size limit that aren't expected to have large payloads. This change does not directly impact the UI as it is related to backend communication optimizations. The purpose of this change was to enhance performance by optimizing the efficiency of data exchange between the platform and ALite. It reduces unnecessary data processing and provides better monitoring through logging when message size constraints are exceeded. This change primarily affects backend operations by improving communication protocols. It does not directly affect the customer-facing aspects of the software but enhances backend performance and monitoring capabilities. (AP-30841)
- **Security Vulnerability Patches Applied to ALite Component.** The ticket addressed security vulnerabilities by upgrading the spring-security library to version 6.3.8 and musl to version 1.2.5-r1. These upgrades remediate the identified vulnerabilities CVE-2025-22228 and CVE-2025-26519 respectively. There are no direct UI changes as this change is related to backend security patching. The purpose of this change was to strengthen the security of the ALite component within the IPautomataLite framework by patching known vulnerabilities with appropriate upgrades to the affected libraries. This is

a backend change primarily affecting the security posture of the application. It is not directly customer-facing but ensures increased security for end-users. (AP-30779)

- **Removal of Unused 'Allowed Cc Pattern' Field from Client Configuration.** The 'Allowed Cc Pattern' field, which was not in use, has been removed from the client configuration. This change does not impact the UI directly as it pertains to backend configuration. The purpose of this change was to clean up the backend by removing unused fields, which helps in maintaining a streamlined and efficient codebase. This change primarily affects the backend and should not have any direct impact on end-users or customer-facing interfaces. (AP-30851)
- **Implement Manual Task Block for Decision Requests in Default Workflows.** The "Send a message to the Ops manager" block in all default workflows has been replaced with a "Manual task" block. This ensures that the workflow stops and waits for human intervention, requiring a decision to be made manually. The change applies to the default workflow configuration, specifically where the "Send a message to the Ops manager" block was initially placed. Users configuring or monitoring workflows will now see a "Manual task" block labeled as "Request a decision from the Ops manager" instead. The purpose of this change is to enhance workflow flexibility and decision-making by allowing manual intervention at a critical decision point instead of sending an automatic message. This reduces potential dependencies on automated messaging and allows for more controlled workflow processes. The change is technical-debt related, affecting backend workflow logic, but has implications for customer-facing scenarios as it alters new deployments' workflow execution. Existing customers with current workflows remain unaffected until they create new workflows. (AP-29184)
- **Patch Xwiki Vulnerability CVE-2024-45591.** This ticket initiates the process to address the security vulnerability CVE-2024-45591 in Xwiki. The action taken will include updating Xwiki to the versions where this vulnerability is patched. The change will not directly affect the UI as it is a backend security patch. However, users may experience scheduled downtime or brief periods of disabled service during the update process. The purpose of this change is to eliminate a medium-level security vulnerability in Xwiki that could potentially expose user information. By updating to the patched XWiki versions (15.10.9 or 16.3.0RC1), the risk associated with this vulnerability will be mitigated. The change is primarily backend-focused, but it is customer-facing in terms of its impact on functionality and security. (AP-30820)

1.6 SUB-TASKS

None in this release.