JetBrains Accessibility Conformance Report

International Edition

(Based on VPAT® Version 2.5Rev)

Name of Product/Version:

TeamCity 2025.03

Report Date:

April 2025

Product Description:

TeamCity is a continuous integration and continuous delivery (CI/CD) server developed by JetBrains. It supports building, testing, and deploying applications across multiple platforms, offering intelligent build queues, parallel execution, detailed test history, and integrations with popular development tools. TeamCity is designed to optimize development workflows and enhance collaboration among software teams.

Contact Information:

teamcity-support@jetbrains.com

Notes:

This Accessibility Conformance Report evaluates TeamCity version 2025.03 as of April 2025. The evaluation reflects accessibility features and user experience available at the time of testing. Future updates or changes to TeamCity may affect conformance with accessibility standards. This report covers the main web-based user interface of TeamCity; self-hosted server setup, build agents, and third-party plugins are not included.

Evaluation Methods Used:

Accessibility evaluation was performed through a combination of manual inspection, keyboard-only navigation, screen reader testing, and automated accessibility testing using axe DevTools. Testing was conducted across Windows, Linux, and macOS operating

systems. Screen readers such as NVDA and VoiceOver were used to assess focus behavior, navigation order, and screen reader output.

Evaluation included manual review of HTML, CSS, and ARIA markup to validate semantic structure and accessibility implementation.

Testers had knowledge of general TeamCity product functionality and tested real user flows to validate conformance with WCAG 2.1 and 2.2 Level A and AA success criteria.

Applicable Standards/Guidelines

This report covers the degree of conformance for the following accessibility standard/guidelines:

Standard/Guideline	Included In Report
Web Content Accessibility Guidelines 2.0	Level A (Yes) Level AA (Yes) Level AAA (No)
Web Content Accessibility Guidelines 2.1	Level A (Yes) Level AA (Yes) Level AAA (No)
Web Content Accessibility Guidelines 2.2	Level A (Yes) Level AA (Yes) Level AAA (No)
Revised Section 508 standards published January 18, 2017 and corrected January 22, 2018	(Yes)
EN 301 549 Accessibility requirements for ICT products and services - V3.1.1 (2019-11) AND EN 301 549 Accessibility requirements for ICT products and services - V3.2.1 (2021-03)	(Yes)

Terms

The terms used in the Conformance Level information are defined as follows:

- **Supports**: The functionality of the product has at least one method that meets the criterion without known defects or meets with equivalent facilitation.
- **Partially Supports**: Some functionality of the product does not meet the criterion.
- **Does Not Support**: The majority of product functionality does not meet the criterion.
- **Not Applicable**: The criterion is not relevant to the product.
- **Not Evaluated**: The product has not been evaluated against the criterion. This can only be used in WCAG Level AAA criteria.

WCAG 2.x Report

Tables 1 and 2 also document conformance with:

- EN 301 549: Chapter 9 - Web, Sections 10.1-10.4 of Chapter 10 - Non-Web documents, and Sections 11.1-11.4 and 11.8.2 of Chapter 11 - Non-Web Software (open and closed functionality), and Sections 12.1.2 and 12.2.4 of Chapter 12 – Documentation
- Revised Section 508: Chapter 5 – 501.1 Scope, 504.2 Content Creation or Editing, and Chapter 6 – 602.3 Electronic Support Documentation.

Note: When reporting on conformance with the WCAG 2.x Success Criteria, they are scoped for full pages, complete processes, and accessibility-supported ways of using technology as documented in the WCAG 2.0 Conformance Requirements.

Table 1: Success Criteria, Level A

Criteria	Conformance Level	Remarks and Explanations
1.1.1 Non-text Content (Level A)	Partially Supports	Most essential non-text elements in the TeamCity UI include accessible labels via aria-label or title attributes. For example, buttons such as “Create new project/build configuration” and “Run custom build” use aria-label, while icons like “Configure favorite projects” and status indicators for build configurations and agents use title attributes. However, some UI elements—for example, project and build configuration icons, and the status icon for a change—do not currently provide any text alternatives, which may impact screen reader users. Manual inspection and screen reader testing confirmed correct announcements of controls and icons in the majority of cases.

1.2.1 Audio-only and Video-only (Prerecorded) (Level A)	Not Applicable	The TeamCity UI does not include embedded prerecorded audio-only or video-only content. Any media is provided externally (e.g., on the JetBrains website or YouTube) and is not part of the product interface itself.
1.2.2 Captions (Prerecorded) (Level A)	Not Applicable	TeamCity does not include prerecorded video content within the UI. Any external video content, such as tutorials or promotional materials, is typically hosted on platforms like YouTube and includes captions where applicable.
1.2.3 Audio Description or Media Alternative (Prerecorded) (Level A)	Not Applicable	The TeamCity interface does not contain embedded video content that would require audio descriptions. Any external videos are hosted outside the product and are not considered part of the application's user interface.
1.3.1 Info and Relationships (Level A)	Partially Supports	The TeamCity UI generally conveys information and relationships programmatically. Form fields are correctly labeled (e.g., when editing or creating build configurations or steps), and lists use appropriate semantic elements such as and (e.g., Permissions and Notification filters). Tables include header cells marked with <th>, but the scope attribute is not consistently used (e.g., in the Build Steps and Connections tables). Landmarks such as <main> are present, but <nav> is not consistently implemented.

1.3.2 Meaningful Sequence (Level A)	Partially Supports	The TeamCity UI generally follows a logical and meaningful sequence during keyboard and screen reader navigation. Most interactive elements are presented in an order consistent with the visual layout. However, some areas — such as navigation between the sidebar (e.g., agents or projects) and the main content, or transitions between the main area and popups — may disrupt the expected focus sequence, potentially affecting users relying on assistive technologies.
1.3.3 Sensory Characteristics (Level A)	Partially Supports	The TeamCity UI generally avoids using sensory characteristics alone to convey information, and most interactive elements include labels or tooltips. However, some components rely solely on color (e.g., build statuses in the Build History Chart or Builds Overview, and error highlights in logs) without providing textual indicators or accessible descriptions. In some cases, shape is also used without textual reinforcement (e.g., the magic wand–shaped branch selector).
1.4.1 Use of Color (Level A)	Partially Supports	In most areas, the TeamCity UI uses both color and additional cues (such as icons or labels) to convey information. However, there are cases where color is used as the sole indicator—for example, in the Build History Chart, Builds Overview, and log entries—without accompanying text or symbols. This may hinder users with color vision deficiencies or those using assistive technologies.

1.4.2 Audio Control (Level A)	Not Applicable	TeamCity does not play any audio content automatically. There are no media elements or sounds triggered by the application that require user control.
2.1.1 Keyboard (Level A)	Partially Supports	TeamCity supports extensive keyboard interaction through hotkeys and standard navigation. Most dialogs can be opened and closed, menus navigated, and controls triggered using the keyboard alone. However, there are areas where keyboard navigation may be inconsistent — particularly when moving between the sidebar (e.g., agents or projects) and the main area, or between the main area and popup menus. These issues may impact seamless navigation for keyboard-only users.
2.1.2 No Keyboard Trap (Level A)	Partially Supports	TeamCity generally allows keyboard users to navigate in and out of most components. However, some popups lack focus management, which can result in keyboard users being unable to interact or exit. Additionally, areas like the build-switching chart and iframe-based content tabs (e.g., Build Log, Artifacts) may trap or misdirect focus, limiting navigation for users relying solely on keyboard input.
2.1.4 Character Key Shortcuts (Level A 2.1 and 2.2)	Does Not Support	TeamCity includes single-character keyboard shortcuts (e.g., “P” to open the project list, “S” to toggle the sidebar) that are always active, regardless of focus context. These shortcuts cannot be turned off, remapped, or limited to specific interface areas, which

		may cause conflicts for users relying on voice input or assistive technologies.
2.2.1 Timing Adjustable (Level A)	Supports	TeamCity does not enforce strict time limits on user interactions in its Web UI. User sessions remain active as long as activity is detected. Inactivity-based session timeouts are configurable by administrators. When timeouts occur, it is only after a prolonged period of user inactivity, without impact on active users. No other functionality enforces critical or non-adjustable time constraints.
2.2.2 Pause, Stop, Hide (Level A)	Supports	TeamCity's dynamic content, such as live-updating build logs or dashboards, does not include disruptive animations or movement. Auto-updates do not override user control or cause unexpected focus shifts. Where content refreshes automatically, it is limited to non-intrusive elements, and users maintain full ability to scroll, navigate, or review content without interference.
2.3.1 Three Flashes or Below Threshold (Level A)	Supports	TeamCity uses dynamic indicators, such as build status icons, to convey information. These indicators may change color or state but do not flash or blink rapidly. No content flashes more than three times within a one-second period. Manual testing confirmed that the product meets the threshold defined by WCAG 2.3.1.
2.4.1 Bypass Blocks (Level A)	Partially Supports	The TeamCity Web UI provides a “Skip to content” link that becomes visible when focused using keyboard navigation, allowing

		keyboard users to bypass repeated navigation elements and move directly to the main content. However, this mechanism is not directly accessible via screen reader navigation alone, limiting its availability for screen reader users who rely on virtual cursor navigation.
2.4.2 Page Titled (Level A)	Partially Supports	The TeamCity Web UI provides a clear and descriptive page title in most cases, helping users to understand the content and context of the current page. However, there are instances where page titles are generic or lack sufficient detail, which may impact users navigating via screen readers or managing multiple browser tabs. Further improvements are recommended to ensure all page titles are consistently descriptive.
2.4.3 Focus Order (Level A)	Partially Supports	The TeamCity Web UI generally provides a logical and consistent focus order when navigating via keyboard, following the visual layout of the interface. However, some focus management limitations have been identified in specific scenarios, such as within modal dialogs or dynamic content areas. These issues may impact the navigation experience for keyboard and screen reader users.
2.4.4 Link Purpose (In Context) (Level A)	Partially Supports	The TeamCity Web UI generally provides descriptive link text that conveys the purpose of each link. In most cases, link text alone or its surrounding context makes the destination or action clear. However, there are some instances of short or ambiguous

		link text, such as “More” or “Details,” which may require users to rely on surrounding content. Improvements are planned to provide consistently descriptive link text across all interface elements.
2.5.1 Pointer Gestures (Level A 2.1 and 2.2)	Does Not Support	In the TeamCity Web UI, certain functionality — such as reordering build steps — relies solely on path-based pointer gestures (drag-and-drop). There is no alternative method, such as buttons or menus, to perform the same operation using simple pointer input. As a result, users who cannot perform complex gestures may be unable to complete this task independently.
2.5.2 Pointer Cancellation (Level A 2.1 and 2.2)	Supports	In the TeamCity Web UI, actions initiated with a pointer input, such as clicks or drag-and-drop interactions, are generally completed on pointer release rather than on pointer down. This allows users to cancel actions by moving the pointer away or releasing it outside the target area. Testing confirmed that critical operations do not finalize immediately on pointer down without the possibility of cancellation.
2.5.3 Label in Name (Level A 2.1 and 2.2)	Supports	In the TeamCity Web UI, interactive elements that display visible text as part of their label also include that text in their accessible name. This ensures consistency between what is visually presented and what is announced by assistive technologies. Manual testing using screen readers confirmed that the visible labels of

		buttons and links are reflected in their accessible names.
2.5.4 Motion Actuation (Level A 2.1 and 2.2)	Not Applicable	The TeamCity Web UI does not use device motion or orientation to trigger any functionality. All actions are performed through standard user interface controls accessible via keyboard, mouse, or touch input.
3.1.1 Language of Page (Level A)	Supports	The TeamCity Web UI specifies the primary language of the page using the `lang` attribute in the HTML element. This allows assistive technologies to interpret and present content with the appropriate pronunciation and language rules. Testing confirmed that the `lang` attribute is present and correctly reflects the page language.
3.2.1 On Focus (Level A)	Supports	In the TeamCity Web UI, moving focus to an element does not automatically trigger changes of context or initiate actions. Elements such as buttons, links, and form fields only perform actions when explicitly activated by user input (e.g., click or Enter key). Testing confirmed that focus movement alone does not result in navigation, data submission, or unexpected content changes.
3.2.2 On Input (Level A)	Supports	In the TeamCity Web UI, user input into form fields, dropdowns, and controls does not trigger unexpected changes of context or navigation. Actions such as configuration changes or filtering require explicit user

		confirmation through buttons or links. Input changes are visually communicated and do not auto-submit or redirect without user intent.
3.2.6 Consistent Help (Level A 2.2 only)	Supports	In the TeamCity Web UI, help information such as tooltips and context-sensitive guidance is provided consistently across similar screens. Help icons or hints are positioned in predictable locations, typically near corresponding form labels or controls. Testing confirmed that users can reliably locate help options in repeated user interface patterns.
3.3.1 Error Identification (Level A)	Supports	The TeamCity Web UI provides clear identification of input errors in forms and configuration panels. When a user enters invalid or incomplete data, error messages appear adjacent to the relevant fields and include descriptive text. Visual indicators are supported by text cues to ensure accessibility for all users, including those using assistive technologies.
3.3.2 Labels or Instructions (Level A)	Supports	The TeamCity Web UI provides visible and descriptive labels for user input fields, along with contextual instructions or placeholder text where appropriate. Required fields are clearly indicated, and labels are programmatically associated with their corresponding inputs to support screen reader navigation. Testing confirmed that users can understand input expectations across key forms and configuration areas.

3.3.7 Redundant Entry (Level A 2.2 only)	Supports	The TeamCity Web UI minimizes redundant data entry by retaining previously entered information within a given workflow or session. User input is generally preserved across steps where applicable, and sensitive data such as passwords may require re-entry for security reasons. Testing confirmed that the product avoids requiring users to manually re-enter the same information multiple times within a single task.
4.1.1 Parsing (Level A)	Supports	Parsing requirements for HTML content were clarified in the WCAG 2.0 and WCAG 2.1 Errata (September 2023). Based on this clarification, TeamCity Web UI is considered to meet the parsing success criterion.
4.1.2 Name, Role, Value (Level A)	Supports	Interactive UI elements in the TeamCity Web UI expose appropriate names, roles, and values. Screen reader testing confirmed proper announcement and interaction for key components across the interface.

Table 2: Success Criteria, Level AA

Criteria	Conformance Level	Remarks and Explanations
1.2.4 Captions (Live) (Level AA)	Not Applicable	The product does not contain live multimedia content.
1.2.5 Audio Description (Prerecorded) (Level AA)	Not Applicable	The product does not include prerecorded synchronized media requiring audio description.

1.3.4 Orientation (Level AA 2.1 and 2.2)	Supports	TeamCity does not restrict display orientation. The interface adapts to both portrait and landscape layouts when accessed from devices that support rotation.
1.3.5 Identify Input Purpose (Level AA 2.1 and 2.2)	Supports	TeamCity includes user-facing forms that collect information such as email or username, and these fields support programmatic identification of input purpose.
1.4.3 Contrast (Minimum) (Level AA)	Partially Supports	The TeamCity Web UI provides sufficient color contrast for the majority of text and interface elements. Colorful components such as the purple “Run” and “Settings” buttons meet the required contrast ratio standards. However, some elements, such as disabled fields in build configurations stored in a VCS or placeholder text in certain input fields, may not meet the minimum contrast ratio of 4.5:1. Also, there is an older interface, which has slightly lower contrast in some areas. These exceptions may impact users with low vision or color perception differences.
1.4.4 Resize text (Level AA)	Partially Supports	The TeamCity Web UI allows text to be resized up to 200% using browser zoom without significant loss of content or functionality. Most forms, menus, tables, inputs, and charts scale correctly, and layout remains intact during testing at 150% and 200% zoom. However, some dropdowns—such as those in the popup for configuring the statistics chart—may not render or function as expected. Additionally, horizontal or vertical scrollbars

		may appear when navigating inside iframes, which can affect usability.
1.4.5 Images of Text (Level AA)	Supports	The TeamCity Web UI uses images of text only where essential, such as the TeamCity logo or third-party tool logos (e.g., in the "New Build Step" dialog). In all other interface areas, information is presented using real text rendered with HTML and CSS. This ensures that text content remains accessible, resizable, and compatible with assistive technologies.
1.4.10 Reflow (Level AA 2.1 and 2.2)	Partially Supports	The TeamCity Web UI supports reflow when the viewport is resized to 320 pixels. Most text and content reflow vertically without issues, and key interface components such as menus, sidebars, tabs, and dialogs remain usable. However, horizontal scrolling may be required in specific areas such as charts (e.g., Statistics or Diagnostic views) and log files. While overall functionality is retained, there may be usability challenges when using the interface at full size on smaller screens.
1.4.11 Non-text Contrast (Level AA 2.1 and 2.2)	Partially Supports	Most meaningful non-text elements in TeamCity meet the 3:1 contrast requirement, including buttons, icons, and focus indicators. However, testing with axe DevTools revealed insufficient contrast in specific areas, such as counters and certain tooltips. These may be difficult to perceive for users with low vision, particularly under default themes.

1.4.12 Text Spacing (Level AA 2.1 and 2.2)	Partially Supports	When WCAG-recommended text spacing styles are applied via a user stylesheet, some parts of the TeamCity interface experience layout issues such as overlapping text or clipped labels. While core content remains accessible in most views, certain elements do not fully adapt to increased spacing, limiting readability for users with low vision or dyslexia.
1.4.13 Content on Hover or Focus (Level AA 2.1 and 2.2)	Supports	TeamCity meets the requirements for additional content that appears on hover or focus. Tooltips and other interactive elements are accessible via both mouse and keyboard. The content remains visible long enough to be read, can be hovered or focused itself, and may be dismissed without requiring pointer movement.
2.4.5 Multiple Ways (Level AA)	Supports	The TeamCity Web UI provides multiple ways to locate and navigate content. Users can access areas using the sidebar, the quick search field, or breadcrumbs that support structured navigation between projects, agents, and build configurations. TeamCity also supports a hierarchical project structure to assist with discovery. Additionally, links to related items (e.g., builds, agents, changes) are provided across various pages, enabling users to move between interface elements easily. These navigation methods ensure that users with different preferences and abilities can efficiently explore the interface.

2.4.6 Headings and Labels (Level AA)	Supports	The TeamCity Web UI uses a clear and consistent heading and labeling structure. Each page includes a visible and meaningful `<h1>` heading, and many pages also include additional headings such as `<h2>` to support content segmentation. The interface is logically divided into areas such as Projects, Changes, Agents, Build Queue, and Administration. Form fields generally have visible labels, and most inputs include descriptive notes or hints. Consistent terminology is used across the UI to support predictable navigation and understanding. These practices enhance usability for all users, including those using screen readers.
2.4.7 Focus Visible (Level AA)	Partially Supports	The TeamCity Web UI generally provides visible focus indicators when navigating via keyboard. Most interactive elements, including buttons, links, and form fields, highlight or outline on focus. However, the focus indicator may be difficult to see in specific areas. For example, in the Administration area, elements are highlighted in purple, which may not contrast sufficiently against similarly colored buttons like “Run” or “Settings.” Additionally, disabled or read-only fields, such as the Build Configuration ID when settings are stored in a VCS root, do not always show a visible focus. These limitations may affect users relying solely on keyboard navigation.
2.4.11 Focus Not Obscured (Minimum) (Level AA 2.2 only)	Supports	When navigating TeamCity via keyboard, the element in focus remains at least partially visible. Focused buttons, inputs,

		and other controls are not hidden behind fixed UI elements such as headers or sidebars. This ensures users can always determine where the focus is located.
2.5.7 Dragging Movements (Level AA 2.2 only)	Partially Supports	Drag-and-drop is rarely used in TeamCity. However, in at least one action — reordering build steps — the interaction requires mouse-based dragging and has no accessible alternative such as keyboard controls or explicit move buttons. This may pose a barrier for users who rely on keyboard or assistive input methods.
2.5.8 Target Size (Minimum) (Level AA 2.2 only)	Partially Supports	Most interactive elements in TeamCity meet the required minimum target size or are adequately spaced to avoid accidental activation. However, some icons and controls — particularly in tables and dense views — fall below 24×24 CSS pixels and may be difficult to operate for users with motor impairments or on touch devices.
3.1.2 Language of Parts (Level AA)	Supports	The TeamCity Web UI is primarily presented in English and does not contain embedded phrases or passages in other languages that would require language identification. In cases where non-English content is introduced by users (e.g., custom labels or messages), it is assumed to be under their control. Testing confirmed that no system-generated content violates this requirement.
3.2.3 Consistent Navigation (Level AA)	Supports	Repeated navigation elements in TeamCity — such as the project sidebar and top

		menu — appear in a consistent location and order across views. This helps users with cognitive and visual disabilities rely on a stable navigation experience.
3.2.4 Consistent Identification (Level AA)	Supports	TeamCity ensures that user interface components with identical functions are labeled and presented consistently across the application. Buttons, icons, and controls such as “Run,” “Edit,” and “View Log” retain uniform naming, appearance, and behavior throughout the platform.
3.3.3 Error Suggestion (Level AA)	Supports	TeamCity provides descriptive error messages that help users understand what went wrong and how to correct it. When input errors occur in forms or interactive elements, the system offers suggestions or explanations to support successful correction.
3.3.4 Error Prevention (Legal, Financial, Data) (Level AA)	Supports	TeamCity does not typically handle user input involving legal commitments, financial transactions, or sensitive personal data. For the types of input relevant to the product, users are provided with opportunities to review or cancel changes where appropriate.
3.3.8 Accessible Authentication (Minimum) (Level AA 2.2 only)	Supports	TeamCity supports accessible authentication methods that do not rely on cognitive function tests. Users can copy and paste credentials from password managers when logging in, and authentication modules such as Google, GitHub, and other external identity providers are also

		supported. TeamCity does not implement CAPTCHAs. When two-factor authentication is enabled, users must enter a one-time code, which can also be pasted from an authenticator app. These authentication processes do not create barriers for users with cognitive or memory impairments.
4.1.3 Status Messages (Level AA 2.1 and 2.2)	Partially Supports	TeamCity provides status messages for actions such as form submissions and build progress. However, these messages are not consistently exposed to assistive technologies using ARIA live regions, which may limit screen reader users' awareness of dynamic changes.

Revised Section 508 Report

Chapter 3: Functional Performance Criteria (FPC)

Criteria	Conformance Level	Remarks and Explanations
302.1 Without Vision	Partially Supports	TeamCity supports screen reader interaction. Users can navigate the UI, configure builds, and review results without relying on vision. However, there are isolated cases where elements—such as certain icons or controls—lack descriptive labels, which may impact the experience for screen reader users. Additionally, despite support for individual elements, users who rely solely on screen readers may experience challenges with overall navigation flow, context awareness, and maintaining orientation within complex screens or dynamic sections. These cases are documented under WCAG 1.1.1 and 4.1.2.
302.2 With Limited Vision	Partially Supports	TeamCity supports users with limited vision by maintaining readable content at 150–200% zoom without layout breakage in most areas. However, in some specific views—such as certain popups or iframe-contained elements—layout or control behavior may be affected. Most interactive elements meet minimum color contrast and maintain visibility with increased text spacing or zoom.
302.3 Without Perception of Color	Partially Supports	TeamCity uses both color and text/icon-based cues to convey status and actions in most areas. However, there are exceptions where color is

		the sole indicator without an accompanying tooltip or label. This may limit accessibility for users who cannot perceive color differences. Also some focus indicators and low-contrast icons (e.g., in disabled fields or in the Administration area) may be difficult to perceive for some users.
302.4 Without Hearing	Supports	TeamCity does not use audio or sound-based cues to convey information. All system feedback, alerts, and status indicators are provided visually through text, icons, or dialogs. The product is fully operable without hearing.
302.5 With Limited Hearing	Supports	TeamCity functionality is fully accessible without the use of sound. All interactions and notifications are provided visually, ensuring users with partial hearing can fully operate the product. No special audio-related accommodations are required.
302.6 Without Speech	Supports	TeamCity does not include or require speech input for any functionality. All tasks and interactions can be completed using keyboard, mouse, or assistive technology, ensuring full accessibility for users who cannot speak or use voice-based controls.
302.7 With Limited Manipulation	Partially Supports	TeamCity supports keyboard-only navigation and provides access to most interactive elements without requiring mouse input. Users can open dialogs, activate controls, and complete workflows using only the keyboard. However, there are areas where keyboard navigation may be inconsistent, and some interface elements or modal dialogs may trap

		or misdirect focus. Additionally, certain actions—such as reordering build steps—require drag-and-drop and do not offer alternative input methods. Focus styling and small interactive targets may also impact usability for users with motor impairments.
302.8 With Limited Reach and Strength	Partially Supports	TeamCity is largely operable without requiring extensive reach or physical strength. Standard functions are accessible via keyboard navigation and mouse or pointer input without time constraints. However, some UI elements—such as drag-and-drop—may present challenges for users with limited mobility or physical dexterity. These concerns correspond to WCAG 2.1.1, 2.5.7, and 2.5.8.
302.9 With Limited Language, Cognitive, and Learning Abilities	Partially Supports	TeamCity generally uses consistent UI structure, terminology, and predictable navigation to support users with cognitive or learning disabilities. Forms include contextual instructions and help text, and most user input is preserved across steps. However, some interface elements use vague link text or may lack sufficient input guidance, which can hinder comprehension for users with cognitive or language limitations.

Chapter 4: Hardware

Notes: Not Applicable.

Chapter 5: Software

Notes: This section simply affirms that **Chapter 5 applies to TeamCity** as a software product.

Criteria	Conformance Level	Remarks and Explanations
501.1 Scope – Incorporation of WCAG 2.0 AA	See WCAG 2.x section	See information WCAG 2.x section
502 Interoperability with Assistive Technology		
502.2.1 User Control of Accessibility Features	Supports	TeamCity provides user-selectable visual themes (light and dark), which can be toggled in the user interface. These options allow users to adjust the visual presentation according to their needs. Additional accessibility preferences, such as screen reader support or zoom, are handled by the operating system or browser, and are compatible with TeamCity.
502.2.2 No Disruption of Accessibility Features	Supports	TeamCity does not interfere with or disable accessibility features provided by the operating system or browser. The product works as expected with screen readers, magnification, zoom, high contrast modes, and other platform-level accessibility tools.
502.3 Accessibility Services		
502.3.1 Object Information	Partially Supports	TeamCity exposes names, roles, states, and values for most interactive elements through semantic HTML and ARIA attributes. This enables assistive technologies to accurately convey component information to users. However, some icons or custom controls do not consistently include accessible names or state indicators. These issues are addressed under WCAG 1.1.1 and 4.1.2.

502.3.2 Modification of Object Information	Partially Supports	TeamCity updates object states and values programmatically for many interactive components using ARIA attributes (e.g., aria-expanded for build or test info sections). These updates are correctly exposed to assistive technologies in most scenarios. However, some expandable elements—such as project folders in the sidebar—do not expose their expanded/collapsed state, limiting screen reader awareness. These issues are addressed under WCAG 4.1.2 and 4.1.3.
502.3.3 Row, Column, and Headers	Partially Supports	TeamCity tables generally use visible header styling and semantic <th> elements to present tabular data. However, scope attributes (e.g., scope="col") are not applied, and some interface tables are constructed using non-semantic elements such as instead of proper table markup. These limitations may affect how assistive technologies interpret table relationships. This is noted under WCAG 1.3.1.
502.3.4 Values	Supports	TeamCity exposes current values of most input fields and controls using semantic HTML and ARIA where appropriate. When users change values—such as in the Parameters tab with custom build parameters—the interface visually highlights the modified fields.
502.3.5 Modification of Values	Supports	TeamCity allows users to modify values using standard input mechanisms, including keyboard navigation and screen reader interaction. Editable fields,

		dropdowns, toggles, and selection controls are operable without mouse input. Value changes can be completed using assistive technologies, in accordance with WCAG 2.1.1 and 4.1.2.
502.3.6 Label Relationships	Supports	TeamCity provides programmatically associated labels for most form controls using <label> elements, aria-label, or similar mechanisms. This enables assistive technologies to announce the correct label when navigating or editing a field. Testing confirmed that users can understand input expectations across key forms and configuration areas.
502.3.7 Hierarchical Relationships	Partially Supports	TeamCity presents grouped and hierarchical relationships visually across its interface — for example, in project navigation, build configurations, and configuration sections. While some relationships are conveyed programmatically using semantic HTML or ARIA roles, structural elements like <section> or <fieldset> may be missing in certain groupings. This can limit how assistive technologies interpret the hierarchy. These issues are noted under WCAG 1.3.1.
502.3.8 Text	Supports	TeamCity displays text content using semantic HTML and ensures it is programmatically determinable for assistive technologies. Most UI text — including labels, descriptions, and buttons — is accessible via screen readers. Logos and decorative graphics are excluded, and meaningful icons are supported by labels or tooltips in most cases.

502.3.9 Modification of Text	Supports	TeamCity supports standard text input and editing across its UI. Users can enter, edit, and review content using the keyboard and assistive technologies. Copy-paste works in the majority of fields, and values typically reflect what is typed in real time. Editing behavior is consistent with platform expectations.
502.3.10 List of Actions	Partially Supports	TeamCity exposes action lists for many interactive elements, such as builds and configurations. However, not all contextual actions are consistently presented to assistive technologies or screen readers, which may limit usability for non-visual users.
502.3.11 Actions on Objects	Partially Supports	TeamCity provides object information for many standard interface components. However, some custom elements may not expose their name, role, or state consistently to assistive technologies, which can hinder accurate interpretation by screen readers.
5.3.12 Focus Cursor	Partially Supports	TeamCity generally maintains a visible and functional focus cursor for keyboard navigation. However, some dynamic UI elements may not correctly reflect focus changes or fail to return focus after interaction, which may disrupt navigation for users relying on assistive technologies.
502.3.13 Modification of Focus Cursor	Partially Supports	TeamCity allows assistive technologies to control the focus cursor in most areas of the interface. However, in certain components or dynamic contexts, custom behavior

		may interfere with expected programmatic focus control, limiting consistent navigation for screen reader users.
502.3.14 Event Notification	Partially Supports	TeamCity provides some status updates via visual cues, but these changes are not consistently exposed to assistive technologies through accessibility APIs. As a result, screen reader users may miss events unless focus is manually moved to the updated region.
502.4 Platform Accessibility Features	Supports	TeamCity is a web-based application that inherits accessibility features from the operating system through the browser. It supports platform-level assistive technologies such as screen readers, zoom, keyboard interaction, and contrast settings via standard browser behavior.
503 Applications		
503.2 User Preferences	Partially Supports	TeamCity allows some user preferences such as text resizing and theme switching. However, not all platform-level accessibility preferences (e.g., high contrast mode or reduced motion) are respected or automatically detected in the web interface.
503.3 Alternative User Interfaces	Partially Supports	TeamCity provides both a legacy and a modern user interface. Both have been evaluated for accessibility. The older interface has slightly lower contrast in some areas but continues to support screen readers and full keyboard navigation. Users are not limited to a single UI for accessible access.

503.4 User Controls for Captions and Audio Description		
503.4.1 Caption Controls	Not Applicable	TeamCity does not include built-in audio or video playback features that would require caption or audio description controls. Multimedia content, if available, is hosted externally.
503.4.2 Audio Description Controls	Not Applicable	TeamCity does not include built-in audio or video playback features that would require caption or audio description controls. Multimedia content, if available, is hosted externally.
504 Authoring Tools		
504.2 Content Creation or Editing (if not authoring tool, enter “not applicable”)	Not Applicable	TeamCity does not provide authoring functionality that allows users to create structured digital content intended for distribution or public consumption. Content creation in TeamCity is limited to configuration settings and internal data management workflows.
504.2.1 Preservation of Information Provided for Accessibility in Format Conversion	Not Applicable	TeamCity does not support the creation of structured content with explicit accessibility metadata. This criterion does not apply to its form-based or configuration-based editing functionality.
504.2.2 PDF Export	Not Applicable	TeamCity does not include functionality to generate or export PDF documents. This criterion is not applicable to the product.
504.3 Prompts	Not Applicable	TeamCity does not allow users to author structured content that would require prompts for accessibility features such as alt text,

		heading levels, or form labeling.
504.4 Templates	Not Applicable	TeamCity does not include templates intended for creating structured digital content that would require accessibility evaluation. This criterion does not apply.

Chapter 6: Support Documentation and Services

Criteria	Conformance Level	Remarks and Explanations
601.1 Scope		
602 Support Documentation		
602.2 Accessibility and Compatibility Features	Partially Supports	TeamCity documentation does not currently include dedicated information about accessibility features, such as keyboard navigation or screen reader compatibility. However, general usage instructions are available, and accessibility-related behavior may be indirectly documented in product usage examples. Accessibility-related notes are primarily available through the VPAT and external issue tracking.
602.3 Electronic Support Documentation	Partially Supports	TeamCity documentation is provided in structured HTML format and is generally screen reader compatible. However, it is not fully navigable using keyboard alone, which limits accessibility for users who rely on keyboard interaction.

602.4 Alternate Formats for Non-Electronic Support Documentation	Not Applicable	TeamCity documentation is distributed exclusively in electronic formats. No non-electronic documentation is provided by JetBrains.
603 Support Services		
603.2 Information on Accessibility and Compatibility Features	Partially Supports	JetBrains support services respond to accessibility-related inquiries submitted via general contact forms or the sales@jetbrains.com address. However, there is no dedicated accessibility support page, FAQ, or documentation to guide users or support agents in answering accessibility questions. Accessibility compatibility details are not available through the standard support portal.
603.3 Accommodation of Communication Needs	Partially Supports	JetBrains provides support only through email and YouTrack tickets. No specific accommodations for individuals with disabilities are implemented, but communication channels do not rely on audio or video.

EN 301 549 Report

Chapter 4: Functional Performance Statements (FPS)

Criteria	Conformance Level	Remarks and Explanations
4.2.1 Usage without vision	Partially Supports	TeamCity supports screen reader interaction. Users can navigate the UI, configure builds, and review results without relying on vision. However, there are isolated cases where elements—such as certain icons or controls—lack descriptive labels, which may impact the experience for screen reader users. Additionally, despite support for individual elements, users who rely solely on screen readers may experience challenges with overall navigation flow, context awareness, and maintaining orientation within complex screens or dynamic sections. These cases are documented under WCAG 1.1.1 and 4.1.2.
4.2.2 Usage with limited vision	Partially Supports	TeamCity supports users with limited vision by maintaining readable content at 150–200% zoom without layout breakage in most areas. However, in some specific views—such as certain popups or iframe-contained elements—layout or control behavior may be affected. Most interactive elements meet minimum color contrast and maintain visibility with increased text spacing or zoom.
4.2.3 Usage without perception of colour	Partially Supports	TeamCity uses both color and text/icon-based cues to convey status and actions in most areas. However, there are exceptions where color is the sole indicator without an accompanying tooltip or label. This may

		limit accessibility for users who cannot perceive color differences. Also some focus indicators and low-contrast icons (e.g., in disabled fields or in the Administration area) may be difficult to perceive for some users.
4.2.4 Usage without hearing	Supports	TeamCity does not use audio or sound-based cues to convey information. All system feedback, alerts, and status indicators are provided visually through text, icons, or dialogs. The product is fully operable without hearing.
4.2.5 Usage with limited hearing	Supports	TeamCity functionality is fully accessible without the use of sound. All interactions and notifications are provided visually, ensuring users with partial hearing can fully operate the product. No special audio-related accommodations are required.
4.2.6 Usage with no or limited vocal capability	Supports	TeamCity does not include or require speech input for any functionality. All tasks and interactions can be completed using keyboard, mouse, or assistive technology, ensuring full accessibility for users who cannot speak or use voice-based controls.
4.2.7 Usage with limited manipulation or strength	Partially Supports	TeamCity supports keyboard-only navigation and provides access to most interactive elements without requiring mouse input. Users can open dialogs, activate controls, and complete workflows using only the keyboard. However, there are areas where keyboard navigation may be inconsistent, and some interface elements or modal dialogs may trap or misdirect focus. Additionally, certain actions—such as reordering build steps—require drag-and-drop and do

		not offer alternative input methods. Focus styling and small interactive targets may also impact usability for users with motor impairments.
4.2.8 Usage with limited reach	Partially Supports	TeamCity is largely operable without requiring extensive reach or physical strength. Standard functions are accessible via keyboard navigation and mouse or pointer input without time constraints. However, some UI elements—such as drag-and-drop—may present challenges for users with limited mobility or physical dexterity. These concerns correspond to WCAG 2.1.1, 2.5.7, and 2.5.8.
4.2.9 Minimize photosensitive seizure triggers	Partially Supports	TeamCity uses dynamic indicators, such as build status icons, to convey information. These indicators may change color or state but do not flash or blink rapidly. No content flashes more than three times within a one-second period. Manual testing confirmed that the product meets the threshold defined by WCAG 2.3.1. Auto-updates do not override user control or cause unexpected focus shifts. Where content refreshes automatically, it is limited to non-intrusive elements, and users maintain full ability to scroll, navigate, or review content without interference.
4.2.10 Usage with limited cognition, language or learning	Partially Supports	TeamCity generally uses a consistent UI structure, terminology, and predictable navigation to support users with cognitive or learning disabilities. Forms include contextual instructions and

		help text, and most user input is preserved across steps. However, some interface elements use vague link text or may lack sufficient input guidance, which can hinder comprehension for users with cognitive or language limitations. Additionally, configuring certain workflows or build configurations may require technical expertise, which could present challenges for users with limited cognitive skills or non-technical backgrounds.
4.2.11 Privacy	Supports	TeamCity supports user privacy during authentication and interaction. Sensitive inputs, such as passwords and API tokens, are masked and can be entered using assistive technologies without compromising privacy. TeamCity is designed to be self-hosted, meaning users set up and manage their own servers, further enhancing control over data and infrastructure. TeamCity does not collect user data without explicit consent. Users can enable or disable the “Periodically send usage statistics to JetBrains” option at any time. Details on data handling and protection are provided in the JetBrains Privacy Policy. Accessibility features do not expose personal data or compromise confidentiality.

Chapter 5: Generic Requirements

Criteria	Conformance Level	Remarks and Explanations

5.1 Closed functionality		
5.1.2 General		
5.1.2.1 Closed functionality	Not Applicable	TeamCity is a web-based product and does not operate in closed functionality environments.
5.1.2.2 Assistive technology	See 5.1.3 through 5.1.6	See information in 5.1.3 through 5.1.6
5.1.3 Non-visual access		
5.1.3.1 Audio output of visual information	Partially Supports	TeamCity supports screen reader output for most visual content. Buttons, labels, and forms are programmatically labeled using semantic HTML and ARIA attributes. Users can access essential information via screen readers such as NVDA and VoiceOver. However, some icons and chart elements lack descriptive labels or alternative text, which may limit access to specific visual content for users relying on audio output.
5.1.3.2 Auditory output delivery including speech	Not Applicable	TeamCity does not use audio or sound-based cues to convey information. All system feedback, alerts, and status indicators are provided visually through text, icons, or dialogs.
5.1.3.3 Auditory output correlation	Not Applicable	TeamCity provides no auditory outputs that would require synchronization.
5.1.3.4 Speech output user control	Not Applicable	TeamCity does not provide speech output or audio playback. All tasks and interactions can be completed using keyboard, mouse, or assistive technology, ensuring full accessibility

		for users who cannot speak or use voice-based controls.
5.1.3.5 Speech output automatic interruption	Not Applicable	TeamCity does not generate speech output or audio alerts that could interfere with assistive technologies. All system feedback is presented visually. This criterion is not applicable.
5.1.3.6 Speech output for non-text content	Partially Supports	TeamCity provides text alternatives for most non-text content using semantic HTML, aria-label, or tooltip text. Assistive technologies such as NVDA and VoiceOver can announce the meaning of many key icons and controls. However, some visual elements — such as status icons, charts, or decorative indicators — do not consistently include descriptive labels, limiting screen reader access to all non-text content.
5.1.3.7 Speech output for video information	Not Applicable	TeamCity does not include embedded video content. Any instructional or promotional media is hosted externally and not part of the TeamCity interface. This criterion does not apply.
5.1.3.8 Masked entry	Supports	TeamCity masks all sensitive data fields, including passwords, tokens, and user password parameters, using standard input types. These fields are accessible to assistive technologies, and screen readers do not expose the actual input values. However, the presence of sensitive content may not always be explicitly announced, which can make it unclear to some screen reader users that a field is for password entry. TeamCity supports secure clipboard-based interactions, including copying and pasting of

		passwords.
5.1.3.9 Private access to personal data	Supports	TeamCity allows users to securely access and review personal data, such as credentials, tokens, and profile information, using keyboard and screen reader navigation. Sensitive fields are masked visually and not exposed through assistive technologies. Interactions can be completed privately, even in shared environments. Additionally, TeamCity uses role-based access permissions to control visibility and modification rights for user data and configuration settings, further ensuring privacy through access restriction. While masked fields may not always announce their purpose explicitly, user data remains secure and accessible without public disclosure.
5.1.3.10 Non-interfering audio output	Not Applicable	TeamCity does not generate any audio output or sound effects that could interfere with assistive technologies. All feedback is conveyed visually. This criterion is not applicable.
5.1.3.11 Private listening volume	Not Applicable	TeamCity does not generate any audio output or sound effects that could interfere with assistive technologies. All feedback is conveyed visually. This criterion is not applicable.
5.1.3.12 Speaker volume	Not Applicable	TeamCity does not include any audio playback or speaker output. All interaction occurs visually within the web interface. This criterion does not apply.
5.1.3.13 Volume reset	Not Applicable	TeamCity does not produce audio or require volume settings. All interactions are visual. This criterion does not apply.

5.1.3.14 Spoken languages	Not Applicable	TeamCity does not use speech synthesis or spoken output. All communication occurs visually, and this criterion does not apply.
5.1.3.15 Non-visual error identification	Partially Supports	TeamCity generally provides screen reader-accessible error messages near relevant form fields or controls. When a user enters invalid or incomplete data, error messages appear adjacent to the relevant fields and include descriptive text. Input errors are communicated using textual cues rather than visual styling alone. However, some error highlights—such as log entry indicators or chart elements—may rely primarily on color or icons without accessible alternatives.
5.1.3.16 Receipts, tickets, and transactional outputs	Partially Supports	TeamCity does not produce traditional transactional artifacts like digital receipts or tickets. However, it generates operational outputs such as build logs, build artifacts, server and agent logs, and audit records. Some of this content is accessible to screen readers, but it may not always be clear that the content represents logs or artifacts. In particular, interpreting structured data such as the Audit table or visual diffs may pose challenges for users relying on assistive technologies.
5.1.4 Functionality closed to text enlargement	Partially Supports	TeamCity does not restrict user-side text enlargement. The web interface generally supports zooming up to 200% using browser controls. Most forms, controls, and content remain usable and readable at increased sizes. However, in some areas — such as dropdowns inside popups or content within iframes — layout or

		scroll behavior may be affected, which can impact usability for users relying on larger text or zoomed interfaces.
5.1.5 Visual output for auditory information	Not Applicable	TeamCity does not use auditory alerts, sounds, or speech to convey information. All feedback, status changes, and confirmations are presented visually through text, icons, and dialogs. This criterion does not apply.
5.1.6 Operation without keyboard interface		
5.1.6.1 Closed functionality	See 5.1.3.1 through 5.1.3.16	See information in 5.1.3.1 through 5.1.3.16
5.1.6.2 Input focus	Partially Supports	The TeamCity Web UI generally provides visible focus indicators when navigating via keyboard. Most interactive elements, including buttons, links, and form fields, highlight or outline on focus. However, the focus indicator may be difficult to see in specific areas. For example, in the Administration area, elements are highlighted in purple, which may not contrast sufficiently against similarly colored buttons like “Run” or “Settings.” Additionally, disabled or read-only fields, such as the Build Configuration ID when settings are stored in a VCS root, do not always show a visible focus. These limitations may affect users relying solely on keyboard navigation.
5.1.7 Access without speech	Supports	TeamCity does not require speech input for any functionality. All actions, workflows, and navigation can be completed using a keyboard, mouse, or alternative input methods without relying on speech.

5.2 Activation of accessibility features	Supports	TeamCity does not include proprietary accessibility features but supports interoperability with platform-level features provided by the browser and operating system. Users can activate tools such as screen readers, high contrast modes, and zoom using standard input methods like keyboard or mouse. The product does not interfere with the activation of accessibility features.
5.3 Biometrics	Not Applicable	TeamCity does not implement biometric authentication. Users log in using credentials or external identity providers. Alternative authentication methods such as password managers and token input are fully supported. This criterion is not applicable
5.4 Preservation of accessibility information during conversion	Not Applicable	TeamCity does not support conversion of content to structured formats such as PDF or Word. The product does not generate documents that would embed or preserve accessibility metadata. Additionally, TeamCity does not store accessibility information internally; it leverages browser-level accessibility support such as zoom, text resizing, and screen reader interoperability.
5.5 Operable parts		
5.5.1 Means of operation	Partially Supports	TeamCity supports multiple means of operation, including keyboard navigation, pointer input, and screen reader interaction. Most tasks can be completed without reliance on a specific modality. However, certain functionality — such as reordering build steps — currently requires mouse-based drag-and-drop and does not offer keyboard-accessible

		alternatives. Additionally, there are areas where keyboard navigation may be inconsistent, and some interface elements or modal dialogs may trap or misdirect focus. Focus styling and small interactive targets may also impact usability for users with motor impairments or alternative input needs.
5.5.2 Operable parts discernibility	Partially Supports	TeamCity provides discernible operable elements using visual styling, semantic HTML, and ARIA roles. Most buttons, links, and input fields are clearly marked and accessible using assistive technologies. However, some interactive elements — such as icons, toggle buttons, or compact table controls — may lack clear labels or sufficient spacing. In addition, low-contrast focus indicators and small interactive targets in certain areas may affect discoverability and operability for users with visual or motor impairments. Some components also rely solely on color (e.g., build statuses in the Build History Chart or Builds Overview, and error highlights in logs) or shape (e.g., the magic wand–shaped branch selector) without providing textual indicators or accessible descriptions.
5.6 Locking or toggle controls		
5.6.1 Tactile or auditory status	Not Applicable	TeamCity does not use tactile or auditory indicators to convey system status. All feedback, including build outcomes and notifications, is presented visually through text, icons, or color indicators. This criterion does not apply.

5.6.2 Visual status	Partially Supports	TeamCity uses visual indicators such as icons, badges, and color changes to convey status. In many cases, these are supported by tooltips or labels. However, some areas — such as the Build History Chart, logs, and error highlights — rely solely on color without text or symbol alternatives. The interface meets contrast requirements for most elements, including the purple “Run” and “Settings” buttons. However, some items, like disabled fields, placeholder text, and legacy UI components, may fall below the 4.5:1 contrast threshold, impacting users with visual impairments.
5.7 Key repeat	Supports	TeamCity does not rely on key repeat for interaction. Holding keys such as Enter or arrow keys does not trigger unintended or repeated actions.
5.8 Double-strike key acceptance	Partially Supports	TeamCity does not consistently handle rapid repeated keypresses. For example, pressing the Enter key twice quickly on the “Run” button may unintentionally start multiple builds. This can impact users with limited fine motor control or those using assistive input devices. However, for some critical operations, confirmation mechanisms are implemented to prevent accidental execution.
5.9 Simultaneous user actions	Partially Supports	Most TeamCity operations can be performed through sequential actions using a keyboard or mouse. However, certain functionality — such as drag-and-drop reordering of build steps — requires simultaneous pointer input without an accessible alternative. These patterns may present barriers for users who rely on

		single-switch devices or cannot perform complex gestures.

Chapter 6: ICT with Two-Way Voice Communication

Notes: Not applicable to TeamCity. TeamCity does not include or support two-way voice communication features such as audio calls, telephony, or voice messaging. All interaction occurs through its web-based interface.

Chapter 7: ICT with Video Capabilities

Notes: Not applicable to TeamCity. TeamCity does not include native video playback or rely on video content within the product interface. Any instructional media is hosted externally (e.g., JetBrains YouTube channel) and not embedded in the software.

Chapter 8: Hardware

Notes: Not Applicable. TeamCity is a web-based software product and is not distributed with or dependent on dedicated hardware.

Chapter 9: Web *(see WCAG 2.x section)*

Notes: TeamCity is a web-based application, and conformance with EN 301 549 Chapter 9 is addressed through evaluation against WCAG 2.1 and 2.2 Level A and AA success criteria (see Tables 1 and 2).

Chapter 10: Non-Web Documents

Notes: Not applicable. TeamCity does not create or export standalone non-web documents such as PDF, Word, or Excel files. Logs and artifacts are presented as structured or plain text within the web interface.

Chapter 11: Software

Criteria	Conformance Level	Remarks and Explanations
11.0 General (informative)		
11.1.1.1 through 11.4.1.3	See WCAG 2.x section	See information in WCAG 2.x section

11.5 Interoperability with assistive technology		
11.5.1 Closed functionality		
11.5.2 Accessibility services		
11.5.2.1 Platform accessibility service support for software that provides a user interface	See 11.5.2.5 through 11.5.2.17	See information in 11.5.2.5 through 11.5.2.17
11.5.2.2 Platform accessibility service support for assistive technologies	See 11.5.2.5 through 11.5.2.17	See information in 11.5.2.5 through 11.5.2.17
11.5.2.3 Use of accessibility services	See information in 11.5.2.5 through 11.5.2.17	See information in 11.5.2.5 through 11.5.2.17
11.5.2.4 Assistive technology	Partially Supports	TeamCity supports assistive technologies through semantic HTML, ARIA attributes, and structured navigation. Most interactive elements expose appropriate name, role, and state, and can be accessed via screen readers. However, some icons, dynamic elements, and expandable components may not fully expose state changes or labels to AT users. These limitations are reflected under WCAG 4.1.2 and 4.1.3.
11.5.2.5 Object information	Partially Supports	TeamCity exposes names, roles, states, and values for most interactive elements through semantic HTML and ARIA attributes. This enables assistive technologies to accurately convey component information to users. However, some icons and custom controls do not consistently include accessible names or state indicators. These

		issues are documented under WCAG 1.1.1 and 4.1.2.
11.5.2.6 Row, column, and headers	Partially Supports	TeamCity tables generally use visible header styling and semantic <th> elements to present tabular data. However, the scope attribute is not applied consistently, and some interface tables are constructed using non-semantic elements such as instead of proper table markup. These limitations may affect how assistive technologies interpret table relationships. This is noted under WCAG 1.3.1.
11.5.2.7 Values	Partially Supports	TeamCity exposes current values for most standard text fields using semantic HTML, which allows screen readers to announce the content correctly. However, custom components such as dropdowns and toggles are not consistently ARIA-enhanced. As a result, it is not always clear to screen reader users which item is selected in a dropdown or whether a toggle is enabled or disabled. These limitations may affect users relying on assistive technologies.
11.5.2.8 Label relationships	Partially Supports	TeamCity provides programmatically associated labels for most form controls using <label> elements, aria-label, or aria-labelledby. This allows assistive technologies to announce the correct label when navigating or editing a field. Testing confirmed that users can understand input expectations across key forms and configuration areas. However, some UI elements — for example, project and build configuration icons, and the status

		icon for a change — do not currently provide text alternatives, which may impact screen reader users.
11.5.2.9 Parent-child relationships	Partially Supports	TeamCity presents grouped and hierarchical relationships visually across its interface — for example, in project navigation, build configurations, and configuration sections. While some relationships are conveyed programmatically using semantic HTML or ARIA roles, structural elements like <section>, <fieldset>, or aria-labelledby may be missing in certain groupings. This can limit how assistive technologies interpret the hierarchy. These issues are noted under WCAG 1.3.1.
11.5.2.10 Text	Supports	TeamCity displays text content using semantic HTML and ensures it is programmatically determinable for assistive technologies. Most UI text — including labels, descriptions, and buttons — is accessible via screen readers. Logos and decorative graphics are excluded, and meaningful icons are supported by labels or tooltips in most cases.
11.5.2.11 List of available actions	Partially Supports	TeamCity exposes action lists for many interactive elements. Dropdown menus are accessible to assistive technologies in most cases. However, some buttons lack accessible labels, making it unclear what list or action is being presented. Additionally, navigation through popup dialogs containing selectable values may result in focus loss, especially when switching between dialogs. These issues may limit usability for screen readers and keyboard-only users.

11.5.2.12 Execution of available actions	Partially Supports	TeamCity provides accessible execution of most actions using keyboard and screen readers. Users can activate buttons, links, and form controls through standard input methods. However, some custom or unlabeled controls may not expose the necessary name/role/action metadata, limiting activation via assistive technologies. Additionally, focus behavior in popup dialogs may disrupt the ability to execute certain actions consistently for non-visual users.
11.5.2.13 Tracking of focus and selection attributes	Partially Supports	TeamCity generally tracks focus and selection attributes using standard HTML semantics and ARIA where applicable. Selected tabs, rows, and list items are typically clear and visually distinct. However, in some cases, the visible focus indicator uses the same color as the button background (e.g., purple on purple), making it difficult to perceive. Focus may also be lost or misdirected in dynamic contexts such as popups, dropdowns, or after dialog transitions. Some selection states — particularly in hierarchical or dynamic components — may not be consistently announced to assistive technologies.
11.5.2.14 Modification of focus and selection attributes	Partially Supports	TeamCity allows assistive technologies to control focus and selection in most areas of the interface. Screen readers and keyboard users can navigate, select items, and move focus using standard methods. However, in some dynamic components or modal dialogs, custom behavior may

		interfere with expected programmatic control. Focus is not always restored after dialog transitions, and selection changes may not be fully exposed to assistive technologies.
11.5.2.15 Change notification	Partially Supports	TeamCity updates the DOM appropriately when dynamic changes occur — such as when a new build starts, a message appears, or status information updates. These changes are visually perceivable and reflected in the interface. However, ARIA live regions are not consistently used to ensure assistive technologies detect these updates in real time. As a result, some screen reader users may not be notified of changes unless they manually navigate to the updated region.
11.5.2.16 Modifications of states and properties	Partially Supports	TeamCity updates states and properties programmatically for many interactive components using standard HTML attributes and ARIA (e.g., aria-expanded for build or test sections). These changes are correctly exposed in most scenarios. However, some custom components — such as dropdowns, toggles, and expandable project folders — do not consistently communicate their updated state to assistive technologies, which may impact screen reader awareness.
11.5.2.17 Modifications of values and text	Supports	TeamCity allows users to modify values using standard input mechanisms, including keyboard navigation and screen reader interaction. Editable fields, toggles, and selection controls are operable

		without mouse input. Value and text changes are reflected visually and programmatically. Copy/paste is supported across input fields, and real-time feedback is provided in most cases.
11.6 Documented accessibility usage		
11.6.1 User control of accessibility features	Supports	TeamCity does not include native accessibility settings such as screen reader modes or contrast controls. However, it provides user-selectable visual themes (light and dark), which can be toggled in the user interface to enhance readability. The product also supports platform-level accessibility features such as keyboard navigation, screen readers, and browser zoom. These features can be independently activated by users using standard input methods, and TeamCity does not interfere with them.
11.6.2 No disruption of accessibility features	Supports	TeamCity does not interfere with or disable accessibility features provided by the operating system or browser. The product works as expected with screen readers, magnification, zoom, high contrast modes, and other platform-level accessibility tools.
11.7 User preferences	Partially Supports	TeamCity allows some user preferences such as text resizing through browser zoom and theme switching (light/dark mode). Certain UI preferences are retained across sessions. However, the product does not consistently respect platform-level accessibility settings such as high contrast mode or

		reduced motion. It also does not provide dedicated in-product accessibility controls for adjusting font size or spacing.
11.8 Authoring tools		
11.8.1 Content technology		
11.8.2 Accessible content creation	Not Applicable	See information in WCAG 2.x section. TeamCity does not function as a content authoring tool. Users may input configurations, comments, or labels within the CI/CD interface, but the product does not support creation of structured content intended for distribution or public consumption.
11.8.3 Preservation of accessibility information in transformations	Not Applicable	TeamCity does not support transformation of user-authored content between formats. The product does not export structured documents (e.g., PDF or Word) and does not perform format conversions that would involve preservation of accessibility information.
11.8.4 Repair assistance	Not Applicable	TeamCity does not provide authoring or publishing functionality and does not include accessibility checking or repair tools. The product does not function as a content creation or validation platform under this criterion.
11.8.5 Templates	Not Applicable	TeamCity does not include templates intended for creating structured digital content that would require accessibility evaluation. This criterion does not apply.

Chapter 12: Documentation and Support Services

Criteria	Conformance Level	Remarks and Explanations
12.1 Product documentation		
12.1.1 Accessibility and compatibility features	Partially Supports	TeamCity documentation does not currently include dedicated information about accessibility features, such as keyboard navigation or screen reader compatibility. However, general usage instructions are available, and accessibility-related behavior may be indirectly documented in product usage examples. Accessibility-related notes are primarily available through the VPAT and external issue tracking.
12.1.2 Accessible documentation	Supports	TeamCity documentation is provided in accessible HTML format on the JetBrains website. It includes structured headings, keyboard-navigable content, and formatting compatible with screen readers and supports browser zoom. Most images have titles, and diagrams are usually accompanied by explanatory text. However, in some cases it may not be clear whether an element is a link or plain text, and visual indicators alone are used. Similarly, while most pictures have titles, not all are explicitly labeled as informative or decorative for assistive technologies.

12.2 Support Services		
12.2.2 Information on accessibility and compatibility features	Partially Supports	JetBrains support services respond to accessibility-related inquiries submitted via general contact forms or the sales@jetbrains.com address. However, there is no dedicated accessibility support page, FAQ, or documentation to guide users or support agents in answering accessibility questions. Accessibility compatibility details are not available through the standard support portal.
12.2.3 Effective communication	Supports	JetBrains provides support via Zendesk, YouTrack, and web-based contact forms, all of which are compatible with screen readers and assistive technologies. Support is provided in written form, with clear and readable responses. Users with disabilities may request alternative contact methods if needed.
12.2.4 Accessible documentation	Supports	Documentation for accessing JetBrains support services is available on the website in structured HTML format. Users can navigate support instructions, ticket submission forms, and contact guidance using keyboard and screen readers. The content is readable and compatible with assistive technologies.

Chapter 13: ICT Providing Relay or Emergency Service Access

Notes: Not Applicable. TeamCity does not provide access to emergency or relay services. It is not intended for emergency communication or response and does not support related functionality.

Legal Disclaimer

This Accessibility Conformance Report is provided for informational purposes only and reflects the state of TeamCity 2025.03 as of April 2025. JetBrains reserves the right to revise this document at any time, without prior notice. JetBrains makes no warranties, express or implied, regarding the accuracy or completeness of this document or the product’s conformance at any future date. JetBrains shall not be held liable for any consequences arising from the use of this document, and no contractual obligations are created, assumed, or intended through its content.