VPAT® 2.5 — WCAG EDITION

Accessibility Conformance Report

Play With ASL: School Edition

Report Title	Play With ASL: School Edition Accessibility Conformance Report
VPAT Version	2.5 — WCAG Edition
Product / Version	Play With ASL: School Edition, v1.x (current production, July 2026)
Report Date	July 13, 2026
Product Description	Play With ASL: School Edition is a gamified American Sign Language (ASL) learning application for K-12 and higher education institutions, built on a single Flutter codebase (native iOS/Android apps, plus a web build). It includes three games — Memory Match, TILT!, and Multiple Choice — organized by vocabulary categories. Institutions launch it via LTI 1.3 (1EdTech LTI Advantage Complete certified) from their Learning Management System (Canvas, Brightspace, Schoology, Blackboard, Moodle) — a standards-based browser handshake that provisions the student and hands off into the same app experience used by all learners.
Contact Information	Jon Mowl, Founder & CEO — jon@playwithasl.com
Notes	**Screen reader support (VoiceOver/TalkBack): Partially Supports.** Standard UI (menus, forms, navigation) is automatically exposed to VoiceOver and TalkBack via Flutter's built-in accessibility framework. The three custom-built games (Memory Match, TILT!, Multiple Choice) have not yet had a hands-on screen-reader pass. Conformance levels below reflect an architecture-level review of the app's engineering documentation. A third-party accessibility audit — including a hands-on assistive-technology review of all three games — is planned; the audit date has not yet been determined.
Evaluation Methods Used	Architecture-level review, grounded in: (1) Flutter's built-in accessibility semantics framework, which automatically exposes name/role/value for

standard UI components to iOS UIAccessibility (VoiceOver) and Android AccessibilityNodeInfo (TalkBack); (2) the app's documented responsive design system (clamped text scaling, overflow-safe layout, adaptive breakpoints); (3) confirmed game-specific mechanics (TILT!'s tap alternative to tilting, its per-difficulty adjustable timer). A hands-on VoiceOver/TalkBack pass specific to the three games has not yet been performed.

APPLICABLE STANDARDS / 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

TERMS

Term	Definition
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. (Used only for AAA.)

Table 1: Success Criteria, Level A

Criteria	Conformance Level	Remarks and Explanations
1.1.1 Non-text Content	**Partially Supports**	Standard UI elements (buttons, navigation, icons) use Flutter's built-in Semantics framework, which exposes accessible labels by default. ASL video content is the primary educational medium and is inherently visual/gestural; game cards pair it with text labels. Alt-text coverage for any purely decorative imagery has not been fully audited.
1.2.1 Audio-only and Video-only (Prerecorded)	**Partially Supports**	ASL video clips are the core content and are inherently visual — ASL is a visual-gestural language, not an audible one, so there is no audio track to caption. Text labels/English equivalents accompany each sign video.
1.2.2 Captions (Prerecorded)	**Not Applicable**	The application does not contain prerecorded audio content requiring captions. ASL videos have no audio track by design — this supports ASL learning by requiring learners to watch the sign rather than rely on voice narration.
1.2.3 Audio Description or Media Alternative (Prerecorded)	**Supports**	ASL videos show signs with corresponding English text labels visible in the interface, providing an equivalent text alternative for users who cannot see the video content.
1.3.1 Info and Relationships	**Partially Supports**	Standard UI structure (headings, lists, buttons, form fields) is exposed via Flutter's Semantics tree by default. The three game screens are custom-built interactive canvases (card grids, tilt sensor, multiple-choice cards); full programmatic exposure of their visual relationships has not been independently audited.
1.3.2 Meaningful Sequence	**Supports**	Content is presented in a logical reading order consistent with English reading order.
1.3.3 Sensory Characteristics	**Supports**	Instructions do not rely solely on sensory characteristics. TILT!'s core mechanic is device tilt, but the game also ships with a tappable on-screen alternative, so play is not solely dependent on one sensory/motor interaction method.
1.4.1 Use of Color	**Partially Supports**	Standard UI (navigation, buttons, form state) does not rely on color alone — Flutter Material components pair color with text/iconography by default. Whether in-game correct/incorrect feedback specifically also uses non-color cues has not been independently audited.
1.4.2 Audio Control	**Not Applicable**	The application does not auto-play audio.
2.1.1 Keyboard	**Partially Supports**	Standard UI (menus, forms, buttons) inherits keyboard operability from the underlying framework. TILT! does not require device motion — its tappable on-screen alternative can be operated by any pointer input. Keyboard-only operability of custom in-game gesture areas (across all three games) has not been separately verified.
2.1.2 No Keyboard Trap	**Supports**	The application does not trap keyboard focus.

2.1.4 Character Key Shortcuts	Not Applicable	The application does not implement single character key shortcuts.
2.2.1 Timing Adjustable	Supports	TILT!'s round timer is user-adjustable before play via difficulty selection (Easy = 120s, Normal = 60s, Hard = 45s), giving the learner direct control over the time constraint. Memory Match and Multiple Choice do not impose a round timer.
2.2.2 Pause, Stop, Hide	Partially Supports	ASL video clips play on user action and can be replayed (unlimited replays on Easy difficulty; 3 replays on Normal; 2 on Hard — a difficulty-tuning choice, not an accessibility restriction, since Easy has no limit). Whether an explicit pause control exists mid-round has not been confirmed for this report.
2.3.1 Three Flashes or Below Threshold	Supports	The application does not contain content that flashes more than three times per second.
2.4.1 Bypass Blocks	Not Applicable	The application does not have repeated blocks of navigation content requiring a bypass mechanism; navigation is a single persistent menu structure.
2.4.2 Page Titled	Supports	Screens have descriptive titles/headings.
2.4.3 Focus Order	Supports	Interactive elements receive focus in a logical order consistent with the visual layout.
2.4.4 Link Purpose (In Context)	Supports	Links and buttons have descriptive labels indicating their purpose.
2.5.1 Pointer Gestures	Supports	Memory Match, Multiple Choice, and TILT! (via its tappable on-screen alternative) all support single-point tap/click activation. No multipoint or path-based gestures are required for core functionality in any of the three games.
2.5.2 Pointer Cancellation	Supports	UI actions activate on the up-event, allowing cancellation by moving away before release (standard platform behavior).
2.5.3 Label in Name	Supports	Visible text labels for interactive elements match their accessible names.
2.5.4 Motion Actuation	Supports	TILT!'s core mechanic uses device tilt, but it also ships with a tappable on-screen button alternative — all TILT! functionality can be completed via tap, without device motion. Memory Match and Multiple Choice do not use motion actuation at all.
3.1.1 Language of Page	Supports	The application declares English as the primary language.
3.2.1 On Focus	Supports	Focusing on an element does not trigger a change of context.
3.2.2 On Input	Supports	Game choices require explicit confirmation before context changes.
3.3.1 Error Identification	Supports	Errors are clearly identified with text descriptions, not solely color or icons.

3.3.2 Labels or Instructions	**Supports**	Input fields (login, account creation) have visible labels.
4.1.1 Parsing	**Supports**	The application uses platform UI frameworks that produce valid accessibility trees; no known parsing errors.
4.1.2 Name, Role, Value	**Partially Supports**	Standard UI components expose name, role, and value correctly via Flutter's Semantics framework, which bridges to iOS/Android accessibility APIs by default. Accessibility-role coverage for the three games' custom interactive elements (card flip, tilt/tap controls, multiple-choice cards) has not been independently audited.

Table 2: Success Criteria, Level AA

Criteria	Conformance Level	Remarks and Explanations
1.2.4 Captions (Live)	Not Applicable	The application does not contain live audio content.
1.2.5 Audio Description (Prerecorded)	Not Applicable	ASL video content is inherently visual with text equivalents provided; audio description would be functionally equivalent to the existing text labels.
1.3.4 Orientation	Partially Supports	Game screens (Memory Match, TILT!, Multiple Choice) are built against a landscape design reference (2400×1080) in the app's responsive layout system. Full portrait-orientation behavior for gameplay screens has not been independently confirmed for this report.
1.3.5 Identify Input Purpose	Supports	Form fields for account creation and login use appropriate autocomplete attributes and input types.
1.4.3 Contrast (Minimum)	Partially Supports	The brand system's primary body-text colors meet WCAG AA contrast against white: purple (#8742E3) at 5.36:1 and dark green (#2B5841) at 8.16:1 — both exceed the 4.5:1 threshold for normal text. Brand orange (#FF8C00) is reserved by design rule for decorative chrome only (dividers, bullets), never body/heading text — consistent with orange-on-white failing contrast (2.33:1). A full screen-by-screen audit covering game art and decorative elements has not yet been performed.
1.4.4 Resize Text	Supports	The app's responsive system (`flutter_screenutil` with `minTextAdapt: true`, clamped `.sp` font tokens) scales text across device sizes, and text is wrapped defensively to avoid clipping at extreme scale factors.
1.4.5 Images of Text	Supports	The application uses rendered text rather than images of text for interface elements.
1.4.10 Reflow	Supports	The app's dual responsive system (`ScreenScale` viewport-relative units + breakpoint-based layout decisions for phone/tablet/desktop) combined with systematic overflow protection (`FittedBox`, `AspectRatio`, `LayoutBuilder`) is built specifically to avoid content loss or forced horizontal scrolling across device sizes; game screens are the reference implementation for this pattern.
1.4.11 Non-text Contrast	Partially Supports	Contrast for UI components (buttons, form fields, focus indicators) against the 3:1 ratio has not been independently audited for this report.
1.4.12 Text Spacing	Supports	Content remains readable and functional when users override text spacing via browser/OS accessibility settings.
1.4.13 Content on Hover or Focus	Supports	Tooltips and hover content, where present, are dismissible, hoverable, and persistent.
2.4.5 Multiple Ways	Supports	Users can reach content via category browsing, home-screen game selection, and LTI-provisioned deep links from the LMS.

2.4.6 Headings and Labels	**Supports**	Screens and sections have descriptive headings; labels accurately describe associated content.
2.4.7 Focus Visible	**Partially Supports**	Standard interactive elements provide visible focus indication via platform defaults. Focus visibility for the three games' custom interactive elements has not been independently verified for this report.
3.1.2 Language of Parts	**Supports**	The application clearly distinguishes between English text and ASL video content throughout.
3.2.3 Consistent Navigation	**Supports**	Navigation mechanisms are consistent across all sections of the app.
3.2.4 Consistent Identification	**Supports**	Components with the same functionality are identified consistently throughout.
3.3.3 Error Suggestion	**Supports**	Input errors include correction suggestions; incorrect game answers clearly show the correct answer.
3.3.4 Error Prevention (Legal, Financial, Data)	**Not Applicable**	The application does not involve legal commitments or financial transactions. Account deletion includes a confirmation step.
4.1.3 Status Messages	**Partially Supports**	Game completion and score results are delivered via a persistent, full-screen completion overlay rather than a transient toast, keeping the result on screen for the user to discover. Brief, non-critical confirmations (e.g., "Saved!") use toast notifications, which are inherently transient; whether these are additionally exposed via screen-reader live-region APIs has not been confirmed for this report.

Summary

Play With ASL: School Edition **substantially supports** WCAG 2.1 Level A and Level AA criteria for its standard UI, with a narrow, clearly-scoped set of gaps specific to its three custom-built games.

Strengths

- ASL video content is the core medium, making the product inherently accessible to Deaf/HoH users
- English text labels accompany all ASL content, providing alternatives for users who cannot view videos
- Standard UI components inherit accessibility support by default from Flutter's Semantics framework (bridging to iOS UIAccessibility / Android AccessibilityNodeInfo) — no separate accessibility layer had to be hand-built for menus, forms, and navigation
- The app's responsive design system (clamped text scaling, overflow protection, adaptive breakpoints) directly supports Resize Text and Reflow
- Primary brand text colors (purple, dark green) exceed WCAG AA contrast on white (5.36:1 and 8.16:1); the brand system's own rule against using orange for body/heading text reflects contrast-awareness by design
- No audio-dependent functionality — all critical information is visual
- TILT!'s round timer is directly user-adjustable via difficulty selection; TILT! also ships with a tappable on-screen alternative to tilting — no game in the suite requires device motion to complete
- Logical navigation structure and consistent interface patterns

Known Gaps

- **VoiceOver/TalkBack: Partially Supports.** Standard UI is covered automatically by Flutter's accessibility framework; the three custom-built games have not yet had a hands-on screen-reader pass
- Accessibility-role coverage, keyboard-only operability, and focus visibility for the three games' bespoke interactive elements have not been independently audited
- A full screen-by-screen contrast audit covering game art and decorative elements has not yet been performed
- A third-party accessibility audit is planned; the audit date has not yet been determined

COMMITMENT

> Play With ASL will prioritize a hands-on accessibility audit of the three games and welcomes specific feedback from institutional accessibility teams in the meantime.

Jon Mowl — Founder & CEO, Play With ASL, LLC
jon@playwithasl.com | playwithasl.com

Date prepared: July 13, 2026