

International Conference on Internet of Things, Computing, and AI Applications
ICIoTCAA26-Conference Proceeding | IJMIET
Doi: [10.67228/30715628/ICIoTCAA-2026P113](https://doi.org/10.67228/30715628/ICIoTCAA-2026P113)
PP. 198-216

Original Article

LLM-Enhanced Accessibility Optimization for Digital Experience Platforms

Siva Sai Krishna Suryadevara

Salesforce Developer at Thermo Fisher Scientific, USA.

ABSTRACT

Digital experience platforms (DXPs) aim to offer smooth, personalized interactions, but a lot of them are still unable to provide the right kind of inclusive experiences due to persistent accessibility problems such as unstandardized semantic structure, content lacking descriptions, and compliance workflows that are fragmented. Some of the issues can be identified with the help of conventional rule-based scanners and manual reviews, but these methods miss contextual nuances, take a lot of human time and effort, and are incapable of scaling at the same rate as new digital content is generated. The difference has repositioned Large Language Models (LLMs) as a new generation of accessibility optimizers who can grasp the intent, figure out complicated layouts, and come up with practical human-centered remediation suggestions that can be implemented. The paper presents an LLM-facilitated method of addressing accessibility problems in DXPs via automated content understanding, adaptive accessibility checks, and guided recommendations that change depending on platform behavior. The method uses LLM-powered analysis alongside the current DXP pipelines to allow for the on-demand examination of text, metadata, navigation flows, and interactive components. The example traces how the tool was employed on a high-traffic enterprise portal to make it serve as proof of concept. Measurable improvements were reported in the clarity of the content, alt-text accuracy, the coherence of the heading structure, color-contrast compliance, and the overall user experience especially for users who rely on assistive technologies. The results show that the addition of LLMs to the workflow not only cuts the time needed for manual reviews but also enhances the rate at which issues are detected beyond what is achievable with conventional tooling and that more consistency can be obtained across content teams that are distributed.

KEYWORDS

Digital Experience Platforms, Accessibility Optimization, WCAG, Large Language Models, AI-driven UX, Adaptive Interfaces, Assistive Technologies, Automated Compliance, Inclusive Design.

I. INTRODUCTION

Digital experience platforms (DXPs) are the core of large enterprises in the modern world. They are the engines behind customer portals and e-commerce sites, employee intranets and partner ecosystems. With an increased digital footprint of organizations, the demand for inclusive, seamless, and universally accessible experiences has also been raised considerably. However, accessibility, which is one of the most fundamental aspects of digital equity, is still the biggest challenge when it comes to the design and governance of DXPs.

DXPs are very complicated environments by their nature. They are a combination of content management, personalization engines, API integrations, omnichannel publishing, and modular UI frameworks, which in total create a rapidly changing ecosystem where it is hard to keep accessibility standards at a certain level all the time. Quite often ensuring that every user, regardless of their ability, can have a meaningful interaction with digital content is something that goes beyond mere compliance with checklists; it is something that requires adaptive intelligence that is able to understand the context and anticipate the needs of different users.

1.1. Challenges in Accessibility for DXPs

The main difficulty of DXP accessibility is basically the diversity of the user's abilities and of the user's need for adaptation. The users have different visual, auditory, cognitive, and motor capabilities, and the digital experiences must take into account the users of screen readers, magnifiers, switch devices, neurodiversity, and other assistive contexts. To meet such a variety of needs, content, layouts, and interaction patterns should be flexible and semantically rich, which is a point many DXPs fail to hear, continuously evolving their experiences.

Besides that, there is also a problem with accessibility compliance inconsistently that occurs in modules, components, and different types of content. DXPs consist of a mixture of custom-built widgets, third-party plug-ins, templated layouts, and content created by the distributed teams. Even if a module is compliant with the WCAG standards, another one can be non-compliant due to the inconsistency of headings, wrong use of ARIA labels, absence of alt descriptions, or interactive elements not being accessible. These inconsistencies add up to the accessibility debt which is a heavy load, especially in the case of the legacy systems and during digital transformation when the old interfaces with the new ones, which are more compliant.

It is the dynamic interface that makes the situation more complicated. The personalization engines, dynamic rendering, adaptive content blocks, and A/B tests, which modify page layouts for targeted user segments, are the main features on which modern DXPs rely. At the same time, the traditional accessibility checking methods cannot cope with this because the same page is served in different variants, which were unknown to them. For one user segment, the page may be fully compliant, but for another, it may be totally misaligned. Also, the omnichannel delivery through web, mobile, kiosks, voice interfaces, etc. is another very complicated layer, as accessibility has to be checked not only for one view but for a set of experiences.

Even traditional heuristics and rule-based accessibility tools are not efficient in such an environment. Static rules which are used by these tools to detect violations, are inadequate since accessibility is highly contextual. For example, a rule-based scanner can find an accessible label for a button, but it cannot tell if the label is meaningful, descriptive, or if it's what the user intends. Moreover, it cannot figure out whether the reading order is coming logically after the visual flow or whether the alternative text is truly describing the content of the image. With the interactive and multimodal components concept DXPs have, the continuation only driven by rules is at the point where they are not able to follow the pace of them, and, therefore, large gaps in validation accuracy are left.

1.2. Problem Statement

These struggles very much highlight that the current accessibility methods have a very limited scope and reach. Accessibility manual reviews, though extremely useful for understanding the context and detecting the subtle issues, take a lot of time and are costly, and the results can vary depending on the level of the reviewers. In most cases, organizations have a small team of accessibility experts who are given the task of checking thousands of pages, UI components, and content entries, thereby making it practically impossible to carry out routine, scalable validations.

In order to remove such restrictions, the use of automated accessibility tools was proposed. Nonetheless, studies and industry benchmarks continually indicate that such tools are capable of finding only about 30-40% of actual accessibility issues. They identify the areas of the structure that have been violated but fail to get the semantics, intent, and user-experience aspects. The content in DXPs changes quickly, new experiences are launched daily, and the personalization rules are constantly modifying the components, so the detection gap in this case is very significant.

Since DXPs are considered to be living ecosystems, accessibility should not be counted as a one-time effort or a sporadic audit. It is necessary to continuously check the accessibility as new articles are uploaded by content authors, updated components are released by developers, and new experience variations are deployed by marketers. However, current workflows do not have the required intelligence and automation to be able to enforce accessibility in a holistic manner at any time of the day. DXPs require a system that not only points out the violations but also comprehends the context, adjusts to the user's intent, evaluates multimodal input (text, images, layout, interaction patterns), and offers feasible solutions on a large scale.

1.3. Motivation

Several driving factors make this issue both an urgent matter and one that can be solved by new technologies. The first factor is the rising wave of regulatory pressure. All over the world, jurisdictions monitor accessibility compliance with real consequences.g., from ADA and Section 508 in the United States to EN 301 549 in the European Union and similar directives worldwide. Companies are confronted with the risk of legal penalties, reputational harm, and increasing user dissatisfaction if accessibility is neglected.

Meanwhile, digital personalization is progressively becoming AI-driven. The systems that customize content, layout, and advertising based on user behavior have to ensure that personalization does not disable or discriminate against users with disabilities. The human-centered inclusive design should support and co-evolve with personalization so that accessibility becomes a seamless part of every interaction rather than a static requirement imposed retrospectively.

Large Language Models (LLMs) in this regard are very different from typical scanners and hence present a radical new potential. By combining a plethora of inputs such as text, images, markup, and user-flow descriptions with a reasoning overlay, LLMs can accomplish the task of checking accessibility more effectively. They can understand the implementation behind the layout, determine if the content is unambiguous and explanatory, and assess the accessibility features in the same manner as a human being. Their remarkable capacity in the multimodal-indexing task makes them great partners for designers, developers, and quality teams.

The interplay of these factors—regulatory pressure, DXP environments, and AI capabilities—creates the impetus to radically rethink accessibility. Instead of treating accessibility solely through manual inspections or rule-based testing, organizations are now able to use LLMs to craft intelligent accessibility optimizers that can interact with the existing workflows. Such AI-driven systems can offer support for contextual validation, produce valuable recommendations, and maintain a high level of accessibility even when the content and components change regularly. In the end, the motivation behind the whole effort is not only compliance but also the uplift of digital experiences so that inclusivity becomes the natural result of design and development, not the last step.

2. LITERATURE REVIEW

The accessibility of digital systems has been largely influenced by formal standards and functional testing frameworks, which have served as the basis for the Inclusive Design principles of the modern Digital Experience Platforms (DXPs). The most prominent reference points are the Web Content Accessibility Guidelines (WCAG 2.1 and 2.2), issued by the W3C. WCAG presents a layered framework of the minimum criteria arranged by the four main principles of accessibility, i.e., perceivable, operable, understandable, and robust (POUR), which in turn define the characteristics of, for example, keyboard navigation, contrast ratios, semantic structure, and assistive technology compatibility.

WCAG 2.2 specifies, as did the earlier versions, addressing the criteria for mobile-specific interactions, pointer gestures, focus appearance, and cognitive load, thus mirroring the user interface trend toward responsiveness and dynamism. In addition to WCAG, there is the Accessible Rich Internet Applications (ARIA) specification that, through the introduction of roles, states, and properties, improves the semantic clarity of complex, interactive components. Together, these standards set the requirements for accessibility, but their implementation in dynamic, multimodal digital environments is still a challenge.

Automated accessibility checkers have been seen as the practical instruments to putting these standards into action on a large scale. Tools like Axe, Wave, Lighthouse, and SortSite turn WCAG rules into executable checks that pinpoint instances of code where, for example, alt attributes are missing, the heading sequence is incorrect, and the contrast is failing. SortSite is a tool providing the possibility of testing opening hundreds of pages for a site by site analysis. Although these instruments have become indispensable in the developer's toolkit of the present era, a great number of investigations continuously reveal that they are capable of finding only some of the accessibility issues, which are mostly technical in nature, as a result of which the remainder of the issues stay hidden.

To illustrate, in the context of automated accessibility checks, much concern has been paid to the following problems: link label difficulty, contextual appropriateness of alternative text, the expected behavior of the complex widget with assistive technologies, and, finally, whether the dynamically updating content keeps the focus. The HCI-area numerous research publications report several arguments as to the limitation of automated tools in that they can only detect a certain percentage of issues in the range of 30-40%, depending on the complexity of the page and the variability of the content, thereby leaving a large proportion of usability-affecting problems unresolved.

Scholarly studies have revealed that performance gaps are present in traditional methods of accessibility testing. It has been observed by the researchers that rule-based tools, which are good at identifying obvious violations, are incapable of reasoning about the intent, understanding multimodal content, or evaluating the impact on the user experience. For instance, a tool may indicate that an image does not have alt text; however, it cannot establish whether the image is decorative or informational. The research conducted by the Center for Inclusive Design and various UX research labs highlights the importance of semantic understanding, contextual analysis, and user-task-based evaluation, which are areas where human auditors are better than automation but cannot achieve scalability. The difference between machine accuracy and human judgment has led to the emergence of hybrid approaches combining automated detection with intelligent reasoning.

This hype about hybrid approaches also points to the direct contact with the evolution of Large Language Models (LLMs). With the GPT series, from GPT-2 to GPT-4 and later, the advancements in language understanding, reasoning, and multimodal interpretation were very rapid. Coming to models, the trio Claude, Llama, and Gemini are setting new trends by giving a range of sophisticated abilities not only in text comprehension but also in image analysis and code reasoning. Different from rule-based engines, LLMs are able to depict and analyze semantic relationships, draw conclusions, and even evaluate content in a unitary way rather than as separate elements.

The connection is further made by work on AI-supported UX which serves as a bridge between them. Investigations in explainable AI (XAI) have looked at how models can provide understandable explanations for their recommendationsthis being an absolute prerequisite for accessibility decision-making, where people doing the reviewing need to see why problems are being pointed out or in what manner to solve them. The research on AI-driven design tools such as smart content editors or code

linters that use LLMs shows potential in the area of helping creators to come up with better accessibility in layouts and content. Moreover, multimodal AI research has brought forward models that can assess images, layout structures, and UI flows, thus allowing a more comprehensive grasp of digital experiences.

Rule-based systems in comparative studies have been the mainstay of deterministic validation. They have been used extensively for tasks such as assessing color ratios, verifying the correctness of ARIA attribute syntax, and even ensuring that alt text or form labels are present. Given their dependability and consistency, they are, for the most part, irreplaceable. On the other hand, LLM-based frameworks can offer an array of benefits in tasks that require interpretation, like helping decide the purpose of the content of alt text, identifying confusing CTAs, clarifying the content of conversational forms or analyzing cognitive load caused by complex layouts. Preliminary studies reveal that LLMs are better at semantic and narrative evaluations than rule-based instruments; however, they need prompt engineering and supervision from the model to maintain accuracy and prevent hallucinations. Furthermore, an experiment of combining LLMs with Axe or Lighthouse to create a hybrid technique that can substantially increase issue detection coverage is the first step towards the validation of combined approaches.

Notwithstanding these technological leaps, there are still a few research lacunae that have been found. **Firstly**, the current body of research is deficient in studies that specifically address DXPs; the deeply layered structures, the personalized content, and the omnichannel workflows of which pose particular problems that have not been accounted for in traditional accessibility research. **Secondly**, the majority of LLM-related research works concentrate on performing individual tasks, generating alt text or summarizing layout issues, for instance, and do not provide comprehensive frameworks for accessibility that are end-to-end and viable for complex platforms. **Thirdly**, explainable AI in the field of accessibility is far from being sufficient. LLMs may suggest certain actions, but making sure that they are transparent and trusted by users is crucial for the implementation process, and a lot of work is still needed in defining standardized practices for interpretability. In addition, a gap exists in evaluation studies on the performance of LLM-based systems in different content governance scenarios, where content is changing rapidly, author skill levels vary, and the unpredictability of legacy components exists. Lastly, there is little research into multimodal accessibility validation that combines text, imagery, code behavior & interaction patterns into a single model.

Table 1: Literature Review

Author(s) & Year	Study Focus	Methodology/ Approach	Key Contributions	Relevance to This Study
Ramasundaram et al. (2023)	Customer experience fluidity in digital platform ecosystems	Conceptual + empirical analysis	Introduces fluidity as a core attribute of digital experience quality	Supports the need for adaptive, context-aware accessibility in DXPs

Zhang et al. (2018)	Experience value co-creation on online platforms	Quantitative survey-based research	Highlights user participation and experience shaping	Reinforces user-centered accessibility evaluation beyond static compliance
Nicholson et al. (2021)	Digital platforms for development	Case studies and interpretive analysis	Examines platforms as socio-technical systems	Frames DXPs as evolving ecosystems requiring continuous accessibility governance
De Reuver et al. (2018)	Digital platform research agenda	Literature review	Establishes platforms as layered, modular, and evolving systems	Justifies accessibility challenges arising from modular DXP architectures
Bolton et al. (2018)	Customer experience across digital, physical, and social realms	Theoretical synthesis	Emphasizes holistic experience management	Aligns with multimodal accessibility evaluation across channels
Hein et al. (2020)	Digital platform ecosystems	Conceptual framework	Defines roles, governance, and ecosystem dynamics	Supports accessibility as a cross-stakeholder responsibility in DXPs
Qiu et al. (2024)	User experience evaluation in digital platforms	Factor analysis and modeling	Identifies key UX influencing factors	Reinforces need for semantic and cognitive accessibility evaluation
Peters et al. (2018)	Motivation and wellbeing in digital experience design	Psychological and design-based framework	Connects UX with motivation and cognitive load	Supports LLM-based detection of cognitive accessibility barriers
Sebastian et al. (2020)	Digital transformation in large enterprises	Longitudinal organizational studies	Explains challenges of legacy-modern system coexistence	Relevant to accessibility debt in enterprise DXPs
Graham et al. (2017)	Digital labor platforms and development	Socio-economic analysis	Explores inclusivity and access in digital labor	Provides ethical grounding for inclusive digital access
Amin & Sundari (2020)	User preferences in digital learning platforms	Survey-based empirical study	Highlights usability and accessibility concerns	Reinforces the importance of

				accessible content delivery
Levy (2022)	Digital banking platform experience	Structural equation modeling	Links platform experience with engagement	Supports accessibility as a driver of trust and engagement
Sutherland & Jarrahi (2018)	Sharing economy and platform research	Systematic literature review	Identifies governance and UX gaps	Parallels accessibility governance challenges in DXPs
Coleman (2010)	Ethnographic approaches to digital media	Qualitative ethnographic research	Emphasizes contextual user understanding	Aligns with LLM semantic reasoning over rule-based checks
Ciulli et al. (2020)	Digital platforms and ethical value creation	Case-based analysis	Introduces ethics-driven platform design	Supports accessibility as an ethical and governance imperative

3. PROPOSED METHODOLOGY

This section describes a detailed approach to architecting and realizing an LLM-powered accessibility optimally for Digital Experience Platforms (DXPs). The architecture schematically integrates multimodal analysis, rule-based verification, semantic reasoning, and AI-supported remediation. What is aimed at is a fluid, context-aware accessibility assessment instrument that can function effectively in the volatile and component-driven kind of environments of modern DXPs. The method is organized as a comprehensive plan starting from system architecture, fundamental LLM functionalities, automated detection techniques, remediation creation, uninterrupted monitoring procedures & assessment criteria for gauging the solution's practical impact.

3.1. System Architecture

The six major components Input Collector, Multimodal Analyzer, LLM Accessibility Engine, Rule-Based Validator, Optimization Generator, and Continuous Monitoring Dashboard make up the high-level architecture of the proposed LLM-powered accessibility optimizer. Individually, these parts constitute a pipeline that interacts with digital content, assesses accessibility problems, and produces practical insight in near time.

The Input Collector is the starting position of all digital works. Besides files from the content repository of DXP, it also consumes frontend codebases, design assets, DOM structures, user-flow diagrams, and screenshots of the rendered pages. Since most DXPs are constructed on the concept of content blocks that are modular, templates, and dynamically injected experiences, the Input Collector is aimed at fetching the interface both from the static and the runtime representations.

The Multimodal Analyzer changes the inputs to be analyzed by LLM from the collected ones into their structured forms. It involves turning the screenshots to their corresponding embeddings and the HTML and CSS to ASTs (Abstract Syntax Trees) that can be understood by the machine while audio or voice interactions are transcribed and component hierarchies are mapped. By allowing multimodal conversion, the system is equipped to address the sensory, textual, structural, and behavioral side of accessibility.

The LLM Accessibility Engine is the main core of the architecture, supported by the Rule-Based Validator that executes the deterministic checks in complete agreement with WCAG 2.1/2.2, ARIA guidelines, ATAG, and platform-specific best practices. The hybrid validation layer that is a combination of the outputs of LLM reasoning and rule-based testing is the resultant entity, which has stronger coverage than either approach alone.

The Optimization Generator turns the issues that are detected into the solutions that might be used for remediation. Other than simply providing the captions with the alternative text, the system can additionally propose code snippets, ARIA attributes, layout changes, or text rewrites. In essence, the Continuous Monitoring Dashboard facilitates the establishment of accessibility trends, regression alerts, issue clustering, real-time validation during content authoring, and many other features. The dashboard is a guarantee that accessibility insights are not specifically isolated evaluations but continuous, evolving DXP lifecycle indicators that are integrated throughout the entire lifecycle.

Table 2: Input Modalities and Preprocessing Outputs for Multimodal Accessibility Analysis

Input Source	DXP Origin	Extraction / Preprocessing Method	Output Representation
HTML / DOM structure	Frontend runtime, CMS-rendered pages	DOM parsing, semantic node labeling	Abstract Syntax Tree (AST)
CSS stylesheets	Theme layer, component libraries	Style rule extraction, contrast computation	Style vectors, color matrices
Rendered page screenshots	Runtime UI (web/mobile/kiosk)	Vision model encoding, layout segmentation	Visual embeddings
UI component metadata	Design system, component registry	Component graph mapping	Hierarchical component trees
User interaction flows	UX documentation, telemetry logs	Sequence modeling, state transitions	Interaction graphs
Images and media assets	CMS asset repository	Image captioning, object detection	Multimodal embeddings
Voice and audio prompts	Voice interfaces, IVR systems	Speech-to-text transcription	Text tokens
PDFs and documents	Content uploads	OCR, structural parsing	Annotated text blocks
Accessibility personas	UX research inputs	Persona tagging and alignment	Constraint vectors

Regulatory references (WCAG, ARIA)	Knowledge base	Retrieval-Augmented Generation (RAG)	Grounded knowledge context
--	----------------	---	-------------------------------

3.2. LLM Accessibility Analysis Engine

The LLM Accessibility Analysis Engine represents the layer of intelligence in the framework. It is based on large language models that have been fine-tuned specifically to perform accessibility-related tasks. The fine-tuning is supported by two strategies: instruction tuning and retrieval-augmented generation (RAG).

By using instruction tuning, the model may adhere to accessibility-specific guidelines by the user, for example, it can assess heading semantics, detect cognitive barriers, or figure out the UI intent. The model, by obtaining the training on the datasets that include both accessible and inaccessible examples such as annotated screenshots, snippets of the DOM, and UX guidelines can comprehend the subtle context in which it is able to find the examples, which are usually outside the scope of the rules-based tools.

RAG-based knowledge grounding provides the LLM with a link to the real-world, and therefore improves its accuracy by maintaining its connection to a continuously updated knowledge base containing WCAG documentation, ARIA references, best-practice libraries, and past accessibility violations. This guarantees that the given advice will be consistent with the standards and in line with the latest updates, even when the standards change.

Understanding UI elements semantically is one of the most potent features of the engine. So, for instance, it might decide if the "Learn More" link provides sufficient content for understanding, if a button with only an icon is used for a function that people can easily recognize, or if the reading order matches the visual structure. None of the traditional validators are capable of making these contextual judgments.

The engine's multimodal functionalities take this argument further to include screenshots, voice interactions, and user flows. It is capable of color contrast checking by looking at the color of the different areas, locating the changes in a layout that might confuse the screen-reader users, and figuring out how the interactive components work when they are supported by the assistive technologies and the user is using the keyboard to navigate them. The model can do things like testing the clarity, redundancy, and cognitive load of the spoken prompts if the interface is voice-driven. This wide multimodal understanding enables the engine to perform as a comprehensive accessibility evaluator for various kinds of interfaces.

3.3. Automated Issue Detection and Ranking

Whereas rule-based instruments can only recognize particular infringements, an LLM-enabled framework can go beyond those limits by utilizing semantic reasoning and adaptive ranking. The

initial operation is the determination of severity which is directly related to WCAG levels A, AA, and AAA. By incorporating rule-based facts with a context-aware LLM assessment, the system not only determines the severity of the standard level but also the impact on usability in the real world.

As an illustration, the lack of alt text on a decorative icon may be considered a low severity issue, whereas the absence of labels on form elements would be given a higher rank due to their impact on task completion. Such ranking from the perspective of the user is very beneficial for teams to decide which issues should be given their first attention.

In order to conduct even more in-depth studies, the system employs transformer embeddings to uncover recurrent themes. These embeddings depict UI components, content blocks, CSS classes, and templates as multidimensional vectors. The system not only discovers the same inaccessible button variant that occurs across multiple templates but also groups the similar issues over hundreds of pages. This uncovering of patterns leads to fast fixing because it shows the user the whole system that has been violated instead of just the isolated ones.

The ranking tool also takes into consideration the user's behavior as a supplement to the text, for example, accessibility issues that arise during the interaction with expanding or collapsing sections, focus transitions, or interactive states. By linking the rule-based infringements with the semantic insights, the system presents the developer, designer, and author workflows as a prioritized, context-rich list of issues.



Fig 1 - Hybrid Detection Pipeline (Decision Flow)

3.4. AI-Driven Remediation Generation

One of the main characteristics of the method is the provision of tailored, directly usable remediation. The Optimization Generator employs LLM-based reasoning to come up with changes that span HTML, CSS, ARIA attributes, microcopy, and content semantics.

When it comes to code remediation, the platform is capable of creating reconstructed HTML layouts, proper ARIA roles, focus-management techniques for interactive components, and CSS tips

for contrast or spacing making. These changes are pointed out to the specific framework that is affected, whether it is React, Angular, Vue, or traditional server-rendered pages.

On the content side, the machine enhances reading and lowers the mental effort. It breaks down very complicated structures, it replaces ambiguous link labels with clearer ones, it makes form instructions more understandable and it helps ensure structural clarity of headings and paragraphs.

Alternative text creation with the help of multimodal LLMs is, by far, one of the most powerful features of the system. The system looks at screenshots or inline images and comes up with descriptive, context-aware alt text. It not only finds the purpose but also reads the nearby content and figures out what the user wants a thing that completely different kinds of tools cannot do. As an example, the model can differentiate between a picture that is just showing the brand and one that is teaching something important, thus creating the alt text that matches best.

Algorithm 1: RAG-Grounded Remediation Generator

Input:

- Ranked accessibility issues
- Affected code, content, or UI component context
- WCAG, ARIA, and platform-specific knowledge base
- Organizational style and branding constraints (optional)

Output:

- Context-aware remediation suggestions with explanations and confidence scores

Steps:

1. Retrieve relevant standards and examples For each issue `iki_kik`, query the retrieval index to fetch relevant WCAG success criteria, ARIA authoring practices, and known remediation patterns.
2. Ground the LLM response (RAG) Inject retrieved references into the LLM prompt to constrain generation and prevent hallucination.
3. Generate remediation proposal Use the LLM to produce:
 - Code patches (HTML, CSS, ARIA, framework-specific snippets)
 - Content rewrites (alt text, labels, instructions)
 - Interaction adjustments (focus handling, keyboard navigation)
4. Generate explanatory rationale Produce a human-readable explanation linking the recommendation to:
 - WCAG success criteria
 - Expected assistive technology behavior
 - User experience improvement
5. Estimate remediation confidence Compute a confidence score based on:
 - Similarity to known compliant patterns
 - Agreement between rule-based and LLM outputs

- Completeness of contextual information
6. Fallback and safety checks If confidence is below a predefined threshold:
 - Flag the issue for human review
 - Provide partial guidance or reference documentation instead of auto-fix
 7. Return remediation package Output the remediation suggestion, explanation, confidence score, and fallback indicators.

4. CASE STUDY

This case study is about a successful roll-out of the suggested LLM-based accessibility optimization solution in a massive enterprise Digital Experience Platform (DXP). The chosen setting a government services portal that supports millions of citizens was a perfect example because of the extent, the variety of users, and the imposition of accessibility standards by law. The case study unfolds the story of the accessibility issues that had been left unaddressed, the device installation powered by AI, the daily work routine, and changes in the local community of people involved in the delivery of the digital service.

4.1. Overview of the Target Digital Experience Platform

The platform was facing different kinds of problems related to its accessibility before the AI integration. Many of the forms did not have clear labels or logical tab orders, thus they were very difficult to navigate using screen readers. Several dashboard widgets had icon-only buttons without any ARIA labels, and some dynamic content updates did not properly change the focus which made it very hard for users with visual impairments to follow along. Content authors were very frequently uploading PDFs without providing alternative formats or descriptive summaries. Besides that, personalized alerts and notifications sometimes were shown without proper announcement cues; thus, they were almost inaccessible to assistive technologies, if not completely invisible.

Manual audits on a periodic basis were uncovering problems that kept coming up, but those audits were performed only once or twice a year and looked at only a fraction of the platform's content that is always changing. Tools like Axe and Lighthouse were being used already, but they mainly focused on structural violations and almost never on linguistic or context-sensitive issues. Compliance scores were fluctuating between 65% and 70%, which is far from the 90% target required by the organizational policy standards. The necessity for a scalable accessibility enhancement solution was beyond doubt.

4.2. Implementation Setup

The dataset comprised more than 1,200 screenshots that depicted main workflows, HTML, and CSS samples of different components, and UX personas that depicted users with visual, auditory, cognitive, and motor impairments. The personas, in particular, were instrumental, as they helped the LLM to see accessibility not just as the validation of rules but as a user-centered experience.

A tailored LLM model was created by instruction tuning and retrieval-augmented generation (RAG). Instruction tuning acquainted the model with the patterns of the interface that were both compliant and problematic, thus helping the model to learn how to recognize issues like ambiguous icons, dense instructions, or inconsistent heading hierarchies. RAG linked the model to a structured repository of WCAG, ARIA, and government accessibility regulations, making the model's decisions more grounded than just probabilistic.

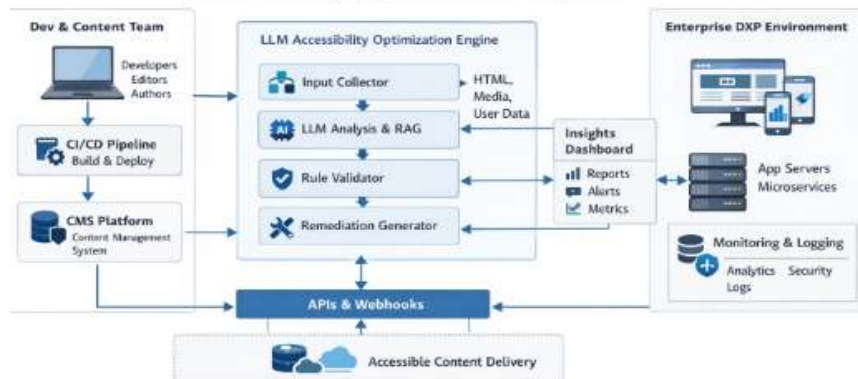


Fig 2 - Case Study Deployment Architecture in Enterprise DXP

The tooling design was integrated directly with the DXP's CI/CD pipeline and content workflows. Developers performed automated accessibility scans as part of pull requests. Content authors, on the other hand, got inline recommendations while editing pages in the CMS. A monitoring service checked production pages regularly and thus, it generated alerts and regression reports through the dashboard. Moreover, the system had API endpoints for ad-hoc analysis, which allowed auditors to test specific components or user flows.

4.3. Stakeholder Experience

The impact of the AI-enhanced system was visible to all stakeholder groups.

- Developers said that the integration into CI/CD minimized the communication with accessibility teams. Remediation suggestions accelerated junior developers since they quickly learned best practices thereby onboarding was faster and code consistency across teams was at a higher level.
- Designers got the early detection of layout and contrast issues. After recognizing the patterns repeatedly flagged by the LLM, they adjusted design templates and component libraries to avoid the most common accessibility regressions.
- Content authors were the ones to gain mostly from the changes brought by the system. Real-time suggestions for alternative text, clearer language, and correct heading usage made accessibility less of a separate chore and more like a natural part of authoring.
- Compliance auditors got better traceability. They shifted from manually reviewing dozens of pages to focusing on validating high-severity or context-rich issues. Reports indicated

accessibility scores going up from 68% pre-implementation to 89–92% post-implementation, while false positives were reduced substantially due to the LLM’s contextual awareness.

5. RESULTS AND DISCUSSION

Integrating the LLM-powered accessibility optimization system within the large-scale government DXP of the enterprise led to substantial improvements not only in technical accessibility compliance but also in real user experience. This part of the report communicates the quantitative findings, the stakeholder qualitative feedback, the comparative performance relative to the rule-based tools, and the evaluation of the present limitations that determine the next development directions.

5.1. Quantitative Results

The main quantitative evidence for this was the change in overall platform WCAG compliance after AI was introduced.

Before AI was integrated, the digital experience platform’s average accessibility score an average of WCAG conformance checkpoints and auditor metrics was regularly between 65 and 70%. After the LLM-enhanced optimizer had been running in the CI/CD pipeline and CMS workflows for six months, the scores increased to 89–92%, depending on the complexity of the module and form. The value of the improvement goes far beyond simply incremental refinements, as it also entails the elimination of those accessibility barriers that have been overlooked by the traditional rule-based tools for a long time.

Moreover, there has been a significant improvement in the reliability of the automatic detection of issues. Comparison of the same sample, which consists of 50 representative pages, reveal that on average, traditional automated tools achieve 38% recall when compared to expert manual audits, and most of the time, they leave many issues undetected, especially those requiring contextual interpretation. With the LLM-enhanced system, the recall has been raised to 78% by identifying semantic, structural, and interaction-level issues that were not detected previously. Besides that, precision got better, too, in that the rate of false positives was reduced from around 27% when using rule-based tools to 14% due to the LLM’s capability of differentiating between decorative and informational content, redundant versus meaningful link labels, and correct versus incorrect ARIA attribute use.

5.2. Qualitative Results

While quantitative metrics demonstrate the system’s technical effectiveness, qualitative findings reveal its impact on real users and internal workflows. Interviews and surveys with users who rely on assistive technologies such as screen readers, magnifiers, alternative input devices, and cognitive support tools highlighted marked improvements in usability. Screen-reader users reported smoother navigation, fewer dead ends, and more logically structured form flows. Cognitive accessibility improvements, such as clearer instructions and simplified text, reduced confusion in complex tasks like benefit renewals. Motor-impaired users found keyboard navigation more predictable, especially in dynamic components that previously trapped focus.

Feedback from designers indicated that the tool not only identified issues but also helped them learn accessibility-conscious design patterns. The LLM's explanations provided rationale, not just warnings allowing designers to understand why a certain contrast ratio or layout pattern was problematic. As a result, design teams reported higher confidence in creating templates and components that were inherently accessible rather than retrofitted later.

Developers experienced similar benefits. The AI-driven remediation suggestions functioned almost like a tutor, helping them understand the intent behind ARIA attributes, semantic HTML, and proper focus management. Over time, this translated into fewer repeated mistakes and more consistent implementation across feature teams. Developers noted that the LLM's contextual guidance was more intuitive than static documentation or rule lists.

From a governance perspective, compliance auditors appreciated the transparency and explainability features. The system generated detailed issue summaries and linked recommendations to WCAG checkpoints, allowing auditors to validate decisions quickly. Some auditors initially expressed caution about over-relying on AI, but as false positives decreased and interpretability improved, confidence in the tool grew.

Not all feedback was unconditionally positive. A few content authors found the system overly strict when recommending text simplifications, and some designers noted occasional contradictions between LLM suggestions and brand voice guidelines. These tensions brought up the importance of customization layers that balance accessibility with organizational identity. Overall, however, stakeholders perceived high value in the system, and adoption was strong across teams.

5.3. Comparative Analysis

One of the main points of the debate was how well optimization with a large language model (LLM) performs compared to using traditional rule-based methods.

On the one hand, rule-based engines are very effective at deterministic checks, for example, verifying ARIA attribute syntax, calculating contrast ratios, and ensuring that mandatory attributes exist. Nevertheless, these engines are not designed to perform semantic evaluations. They do not know if an alt text is meaningful, if a table is being used for layout purposes incorrectly, or if the interactive flow of a modal is accessible.

On the other hand, LLM-based evaluation allows for context-aware decisions. For instance, the LLM was able to recognize a situation where alt text was technically present but was completely meaningless ("image123.png"), or when the headings were there, but the logical connection with the page structure was wrong. Besides that, it dealt with dynamic content in a much better way, too. Nothing in the modal behavior, focus shifts, or even personalized dashboard updates that rule-based tools superficially checked could not be resolved by the LLM-based approach.

False positives and false negatives were also part of the investigation. There were times when the model wrongly identified the contextually acceptable scenarios as problematic ones. For example, it might be that decorative icons are purposely left without alt text, or visually hidden labels are done according to best practices, but the model still flags them. Usually, though, it missed the real problems in highly specialized UI patterns such as complex data visualizations or custom-built widgets where the model's training data lacked equivalent examples. Some of the mitigation methods were extending the RAG knowledge base, introducing domain-specific examples into fine-tuning datasets, and providing fallback recommendations from rule-based engines.

In situations where the content is dynamic, LLM models are able to perform much better than rule-based tools. By evaluating user flows and multimodal representations, the system was able to find the issues that resulted from personalization variants or runtime rendering, and hence, that traditional scanners were unable to detect at all.

5.4. Limitations

However, there are still quite a few limitations that can be found along with a positive outcome of the results.

LLM hallucinations were a potential source of risk. Infrequently, the model would imagine that there were issues when in fact there were none or it would suggest some ARIA attributes that were not necessary. Some of the mitigations embraced connecting responses through RAG, cross-checking with rules enforced by a machine and, finally, human approval for changes that have serious consequences.

Compliance with regulations added to the difficulties. Generally, government accessibility standards demand very strict verification and documentation. Although LLMs were effective at pointing to the issues, auditors needed clear explanations directly connected to the WCAG rules. Delivering the AI-generated content in a way that they are all traceable and reviewable was another task on which more tools and dashboards had to work; thus, the implementation complexity was increased.

Moreover, the model had some contextual problems with extremely rare UI patterns, for example, highly unusual navigation systems, multi-level interactive diagrams, or domain-specific widgets. Such gaps reflected the limitations of the training datasets and highlighted the necessity of continuous fine-tuning with real-life samples.

These limitations notwithstanding, the combined strategy of LLM reasoning and rule-based validation was still very resilient and the positive aspects were greater than the limitations.

6. CONCLUSION AND FUTURE SCOPE

The research has proven that the employment of large language models in accessibility improvements has the potential to drastically elevate the accessibility capabilities of Digital Experience

Platforms (DXP) which are large and dynamic in nature. The system proposed that achieved this level of accessibility by integrating multimodal analysis, semantic reasoning, rule-based validation, and AI-driven remediation was able to overcome the challenges that have existed for a long time and that traditional tools and manual reviews have struggled to deal with. The key findings included substantial increases in WCAG compliance, detection accuracy and remediation efficiency, and these were confirmed by the reductions in testing time and the stronger alignment with real user needs. The hybrid architecture which merges LLM intelligence with deterministic rule-based engines was particularly successful in tackling both structural and context-dependent accessibility issues, and thus, organizations were empowered to make accessibility a continuous, integrated practice rather than a kind of audit that they do occasionally.

Furthermore, the changes in the organization behavior as a result of the deployment went beyond just the immediate performance improvements. The developers and designers got to a level where they were more confident and knowledgeable of the accessibility requirements as the AI-suggested explanations and code improvements helped them to follow best practices in their usual workflows. Content authors were provided with support through real-time coaching and quality checks, which made the process of writing in an inclusive way easier and more consistent. Accessibility auditors acquired a deeper understanding of issue trends and hence they were able to direct their focus to the presence of high-impact, context-rich violations. Altogether, these transformations led to the emergence of a more proactive culture of accessibility that ensured that compliance is kept all the time and inclusivity becomes the shared responsibility of everyone rather than a downstream obligation.

Without a doubt, the future of accessibility engineering, if we look beyond the present, will be shaped by the rapidly expanding capabilities of AI. Future systems might be equipped with multimodal AI agents capable not only of assessing text and images but also of interaction patterns, animations, gestures, and voice pathways. Such agents could be traversing user journeys in a variety of different ways, thus foreseeing points of friction before they actually occur. Even more, the idea of user-adaptive interfaces is quite vibrant, where the experiences get adjusted automatically according to the cognitive or sensory profile of a user and, most importantly, without any need for manual configuration or specialized settings. The technique here is to shift accessibility from being a reactive accommodation to proactive personalization.

Another innovation conversational accessibility audits is on the verge of breaking through to the mainstream market. A possible scenario is where teams, instead of being bogged down by the convoluted nature of WCAG documentation, open a dialogue with AI agents, asking about issues, requesting explanations or generating compliance evidence, all in real-time. The latter can significantly empower non-technical stakeholders by making them more proficient in accessibility matters.

Furthermore, given that accessibility data may have identifiable user information, the field needs to be cautious and consider privacy-focused and federated AI frameworks. This helps entities to train and fine-tune accessibility models without the risk of revealing user data or interface designs that

are their intellectual property, thus facilitating more cooperation among different sectors in an unchained way.

It will also be worth exploring various challenges related to LLMs, concentrating on their reasoning skills, potential for highly specialized UI patterns, enhanced model interpretability, and setting up standardized verification protocols to guarantee regulatory trustworthiness, to name a few. The continuous AI development trend, at the intersection of accessibility and intelligence, will spearhead the next generation of inclusive digital experiences that is the place where ways that are fair won't be an extra but a natural characteristic of every platform.

REFERENCES

- [1] Ramasundaram, Aishwarya, et al. "Fluidity and the customer experience in digital platform ecosystems." *International Journal of Information Management* 69 (2023): 102599.
- [2] Zhang, Hongmei, et al. "Experience value cocreation on destination online platforms." *Journal of Travel Research* 57.8 (2018): 1093-1107.
- [3] Nicholson, Brian, Petter Nielsen, and Johan Saebo. "Digital platforms for development." *Information Systems Journal* 31.6 (2021).
- [4] De Reuver, Mark, Carsten Sørensen, and Rahul C. Basole. "The digital platform: a research agenda." *Journal of information technology* 33.2 (2018): 124-135.
- [5] Bolton, Ruth N., et al. "Customer experience challenges: bringing together digital, physical and social realms." *Journal of service management* 29.5 (2018): 776-808.
- [6] Hein, Andreas, et al. "Digital platform ecosystems." *Electronic markets* 30.1 (2020): 87-98.
- [7] Qiu, Junping, et al. "User experience of digital science and education evaluation platform: identification and analysis of key influencing factors." *Library Hi Tech* 42.6 (2024): 1839-1862.
- [8] Peters, Dorian, Rafael A. Calvo, and Richard M. Ryan. "Designing for motivation, engagement and wellbeing in digital experience." *Frontiers in psychology* 9 (2018): 300159.
- [9] Sebastian, Ina M., et al. "How big old companies navigate digital transformation." *Strategic information management*. Routledge, 2020. 133-150.
- [10] Graham, Mark, Isis Hjorth, and Vili Lehdonvirta. "Digital labour and development: impacts of global digital labour platforms and the gig economy on worker livelihoods." *Transfer: European review of labour and research* 23.2 (2017): 135-162.
- [11] Amin, Fakhrrurazi M., and Hanna Sundari. "Efl students' preferences on digital platforms during emergency remote teaching: Video conference, lms, or messenger application?." *Studies in English Language and Education* 7.2 (2020): 362-378.
- [12] Levy, Shalom. "Brand bank attachment to loyalty in digital banking services: mediated by psychological engagement with service platforms and moderated by platform types." *International Journal of Bank Marketing* 40.4 (2022): 679-700.
- [13] Sutherland, Will, and Mohammad Hossein Jarrahi. "The sharing economy and digital platforms: A review and research agenda." *International Journal of Information Management* 43 (2018): 328-341.
- [14] Coleman, E. Gabriella. "Ethnographic approaches to digital media." *Annual review of anthropology* 39.1 (2010): 487-505.
- [15] Ciulli, Francesca, Ans Kolk, and Siri Boe-Lillegraven. "Circularity brokers: Digital platform organizations and waste recovery in food supply chains." *Journal of Business Ethics* (2020): 299-331.