웹 접근성 자동 진단 및 수정 가이드 생성 시스템의 설계와 구현: 발표 원고(텍스트판)

적용 사례의 자체 웹사이트는 발표자가 만들고 운영하는 사이트입니다. 장마다 그림과 발표에서 하는 말을 차례로 적었습니다. 괄호 속 멈춤은 발표자가 쉬는 시간입니다.

1. 웹 접근성 자동 진단 및 수정 가이드 생성 시스템

그림: 표지. 제목과 부제, 발표자 이름과 소속, 학술대회 이름과 논문 번호.

안녕하십니까. 대구사이버대학교 정보화지원팀 권기표입니다. 웹 접근성 진단 도구를 만들고 직접 운영하고 있는데요.

(1초 멈춤)

오늘 드릴 결론은 이것입니다. 진단에서 멈추지 않고 수정까지, 자동이 못 하는 판정은 사람이 이어받습니다.

왜 그런 설계가 필요한지, 어떻게 만들었는지, 자체 적용에서 무엇을 얻었는지 차례로 말씀드리겠습니다.

2. 자동 검사는 위반 목록에서 멈추고, 수정은 사람의 몫으로 남습니다

그림: 큰 수치 두 개. 70.4점은 8개 업종 1,000개 웹사이트의 접근성 평균 점수, 플러스 3.7점은 전년 대비 상승 폭.

2025년 웹 접근성 실태조사부터 볼까요. 이용 빈도가 높은 여덟 개 업종, 천 개 웹사이트의 평균 점수가 70.4점입니다. 전년보다 3.7점 올랐죠. (2초 멈춤)

올랐는데도 왜 현장은 더딜까요. 장애인차별금지법이 웹 접근성을 의무화한 지는 오래됐습니다. 올해 1월에는 디지털포용법도 시행됐고요. 제도는 쌓였는데 점수는 아직 70점대입니다.

도구 쪽에서 보면 이유가 하나 보입니다. 자동 검사가 위반 목록을 내주는 데서 멈춥니다. 그다음 수정은 온전히 사람의 몫으로 남거든요. 무엇을 어떻게 고칠지는 개발자가 처음부터 다시 알아냅니다.

3. 영문 메시지, 위치 찾기, KWCAG 미대응이 그 간극을 만듭니다

그림: 번호 붙은 항목 셋. 첫째 영문 중심 진단 메시지, 둘째 화면 위치를 따로 찾아야 함, 셋째 KWCAG 검사항목 대응 없음.

널리 쓰이는 자동 검사 엔진에는 한계가 셋 있습니다.

첫째, 진단 메시지가 영문 중심이라 국내 개발자가 곧바로 쓰기 어렵습니다. 둘째, 위반 요소가 화면 어디에 있는지 보려면 선택자를 들고 따로 찾아야 하는데요. 셋째, 국내 인증과 실태조사가 기준으로 쓰는 KWCAG 검사항목에 맞춰 주지 않습니다.

특히 셋째가 큽니다. 국내 품질인증은 33개 검사항목의 평균 준수율 95% 이상을 합격 기준으로 두거든요. 도구가 그 체계로 말해 주지 않으면 결과를 사람이 다시 옮겨 적어야 합니다.

4. 네 단계로 짜고, 점검자 판정과 재검사를 돌아오는 길로 두었습니다

그림: 네 칸이 화살표로 이어진 그림. 표본 수집, 페이지 진단, 집계·산정, 산출물이고, 위에 점검자 수동 판정이 집계·산정으로 들어온다는 줄, 아래에 고친 뒤 다시 검사한다는 줄이 있다.

시스템은 네 단계입니다. 구조도를 말로 옮겨 볼까요. 대상 URL에서 출발해 표본 수집, 페이지 진단, 집계·산정, 산출물 생성 네 칸이 화살표로 이어집니다.

돌아오는 길이 둘 있는데요. 아래쪽에서 점검자의 수동 판정이 집계 단계로 들어갑니다. 오른쪽 산출물에서 가이드대로 고친 뒤 다시 검사하는 고리가 맨 위로 이어지고요.

표본은 크롤링으로 모읍니다. WCAG-EM이 말하는 구조적 표본과 무작위 표본을 고르고, 점검자가 목록을 직접 정할 수도 있습니다. 진단은 headless Chromium으로 페이지를 그린 다음 axe-core를 돌립니다. 여기에 자체 검사 규칙을 더했는데요. 초점 표시, 키보드 접근, 건너뛰기 링크, 새 창 고지, 자막 트랙 같은 것들입니다. 원본 소스를 다시 파싱하는 마크업 유효성 검사도 있습니다.

5. WCAG 2.2와 KWCAG 2.2를 함께 매핑해 국내 기준으로 집계합니다

그림: 아래에서 위로 쌓인 네 층. 맨 아래가 axe-core와 자체 검사 규칙, 그 위가 규칙 카탈로그, 그 위가 검사항목별 페이지 준수율, 맨 위가 인증 기준과 나란히 두기.

집계는 국내 기준으로 합니다. 규칙 카탈로그에 규칙마다 WCAG 2.2 성공기준과 KWCAG 2.2 검사항목을 함께 적어 두었거든요.

그래서 같은 검사 결과를 33개 검사항목 기준으로 다시 셀 수 있습니다. 검사항목별 페이지 준수율을 인증 기준 95%와 나란히 놓고 보여 줍니다. 도구가 내는 점수와 인증 심사가 보는 숫자가 같은 자리에 오는 셈이죠.

평가 근거가 있는 항목만 모아 평균 준수율도 냅니다. 전문가 심사 합격선 95%, 2차 심사 구간 85에서 95%에 견준 인증 준비 요약이죠. 다만 실제 심사와는 다른 근사치라고 보고서에 적어 둡니다.

6. 위반마다 한국어 가이드와 AI 수정 요청 문서를 함께 냅니다

그림: 보고서에서 위반 하나를 펼친 화면. 심각 표시와 제목 아래에 WCAG 2.1.1과 KWCAG 검사항목 10이 나란히 적혀 있고, 개선 방법이 한국어로 붙어 있다.

구현에서 가장 공을 들인 곳은 진단 다음입니다.

한국어 개선 가이드는 규칙 카탈로그에 원인과 개선 방법, 코드 예시를 정리해 두고 보고서에 함께 출력합니다. axe-core 진단 메시지에도 공식 한국어 로케일을 입혔고요.

그다음이 AI 수정 요청 문서인데요. 위반 목록을 페이지, 규칙, CSS 선택자, 해당 코드, 진단 내용까지 묶어서 Markdown과 JSON으로 내보냅니다. 개발자는 이 문서를 AI 코딩 도구에 붙여 넣고 바로 수정을 시작합니다. 보고서를 읽고 코드를 찾고 무엇이 문제인지 다시 설명하는 과정이 빠지는 거죠.

7. 웹, CI, MCP가 같은 검사 루프를 쓰므로 같은 점수가 나옵니다

그림: 왼쪽에 개발자와 점검자, 가운데에 같은 검사 루프와 판정 기준, 오른쪽에 웹 서비스·GitHub Action·MCP 서버·브라우저 확장 네 갈래가 선으로 이어진 그림.

같은 엔진이 네 경로로 나갑니다. 웹 서비스, GitHub Action, MCP 서버, 브라우저 확장입니다.

MCP는 Model Context Protocol인데요. 같은 엔진을 MCP 서버로 열어 두면 AI 코딩 도구가 진단과 가이드를 직접 부릅니다. 개발 중인 페이지, localhost까지 포함해서요. 사람이 복사해서 붙여 넣는 과정 없이 고치고 다시 검사합니다.

웹 서비스와 CI와 MCP, 이 세 진입점은 검사 루프와 판정 기준을 공유합니다. 그래서 같은 페이지에는 같은 점수가 나오죠. 브라우저 확장은 보고서가 준 선택자를 사람이 다시 찾지 않게 해 줍니다. 검사 시점의 정지 화면이 아니라 지금 그려진 화면에서 위치를 짚어 주고요.

8. a11ychk.com에서 진단부터 수정 요청 문서까지 보여 드립니다

그림: 보고서의 인증 준비 요약 화면. 97.5라는 수치와 함께 33개 검사항목 중 4개를 평가한 기준의 평균이라는 설명, 95% 미만 항목 1개가 적혀 있다.

화면으로 보시겠습니다. (2초 멈춤)

대상 주소를 넣고 검사를 시작합니다. 표본을 모으고 페이지마다 검사를 돌립니다. 끝나면 보고서가 열리는데요.

맨 위에 자동, 수동, 통합 준수율이 따로 나오고, 직전 검사와 비교한 변화도 함께 남습니다. 그 아래가 검사항목별 페이지 준수율입니다. 95%에 못 미치는 항목은 위반으로 표시돼서 어디가 모자라는지 바로 보이죠.

위반 하나를 펼쳐 보겠습니다. 규칙 이름 아래에 원인과 개선 방법, 코드 예시가 한국어로 붙어 있고, 그 아래에 위반 위치와 실제 코드가 나옵니다.

위쪽 내보내기 메뉴에서 AI 수정 요청 문서를 받습니다. Markdown과 JSON 두 가지인데요. 받은 문서에는 작업 지침과 위반 목록이 들어 있어서, AI 코딩 도구에 그대로 붙여 넣으면 수정이 시작됩니다.

9. 자동이 못 하는 판정은 WCAG-EM 절차로 사람이 이어받습니다

그림: WCAG-EM 네 단계가 화살표로 이어진 줄: 범위 정의, 표본 선정, 판정 기록, 보고. 그 아래 이름표 셋: 자동 검사는 규칙별 위반 매핑, 점검자 판정은 다섯 판정어와 메모, 통합 집계와 보고는 나눠 산정하고 표준 형식으로 내보내기.

자동 검사만으로는 성공기준의 절반도 확정하지 못합니다. 그래서 그 나머지를 버리지 않고 사람이 이어받게 했죠.

WCAG-EM 절차를 그대로 따릅니다. 평가 범위를 정하고, 표본을 고르고, 판정을 기록하고, 보고하는 네 단계인데요. 점검자는 다섯 가지 판정어로 기록합니다. 통과, 실패, 판단불가, 해당없음, 미검사입니다. 여기에 관찰 메모를 붙이고, 페이지 집합에 귀속시킬 수도 있고요. 자동 결과와 같은 어휘라 한 표에서 같이 집계됩니다.

평가 범위도 함께 저장합니다. 목표 적합성 수준, 기준으로 삼은 브라우저와 보조기기 조합, 포함하고 제외한 URL 패턴까지요. 무엇을 어떤 기준으로 평가했는지가 보고서에 남습니다.

그리고 이 평가 기록을 W3C의 WCAG-EM Report Tool이 읽는 스키마와 EARL로 내보냅니다. 표준 도구로 옮겨 이어서 쓰거나 제3자와 주고받을 수 있게 하려는 것이죠.

10. 자체 적용에서 통합 준수율이 86.4%에서 100%로 올랐습니다

그림: 두 칸 비교. 왼쪽 초기 검사는 통합 준수율 86.4%, 요소 수준 위반 9건, 위반 규칙 3개. 오른쪽 최종 검사는 통합 준수율 100%, 요소 수준 위반 0건, 위반 규칙 0개.

자체 웹사이트에 적용해 봤습니다. 표본 여섯 페이지로 검사 세 번, 수정 두 번을 돌렸습니다.

초기 검사에서 요소 수준 위반이 9건, 사이트 수준 위반이 1건 나왔습니다. 그런데 명도 대비 위반 여덟 건을 들여다보니 다수가 도구 오탐이었거든요. 페이드인 애니메이션의 중간 상태를 재서 생긴 것이었습니다. 그래서 애니메이션을 감속한 상태로 그리도록 진단 엔진을 고쳤습니다. 도구를 믿고 고치기 전에 도구부터 고쳐야 했던 셈이죠.

실제 결함인 저대비 장식 텍스트, ARIA 속성 오용, 사이트맵 부재는 가이드대로 고쳤습니다. 최종 검사에서 위반 0건, 통합 준수율 100%가 됐고요. 다만 이 100%는 판정이 확정된 항목만 모수로 잡은 값입니다. 미평가 항목을 통과로 치지 않습니다.

11. 성공기준 55개 중 자동으로 확정된 것은 13개와 16개였습니다

그림: 큰 수치 두 개. 13개는 자체 웹사이트에서 자동 검사가 확정한 성공기준 수(미판정 42개, 76.4%), 16개는 대학 웹사이트에서 확정한 수(미판정 39개, 70.9%). 둘 다 성공기준 55개가 모수.

그래서 자동 검사가 어디까지 했느냐. 성공기준 55개 가운데 자동이 확정한 것은 자체 사이트에서 13개였습니다. 대학 사이트에서는 16개였고요. (2초 멈춤)

나머지는 사람이 봐야 하는 영역이죠. 시스템은 이 한계를 가리지 않습니다. 자동과 수동과 통합 점수를 나눠 보여 주고, 수동 판정 진행률을 함께 표시합니다. 점수 하나로 뭉뚱그리면 무엇을 아직 안 봤는지가 사라지거든요.

같은 방법으로 진단한 대학 웹사이트는 자동 검사 준수율 81.3%였습니다. 네 개 규칙에서 20건이 나왔고요.

12. 진단에서 멈추지 않고 수정까지, 못 하는 판정은 사람이 이어받습니다

그림: 한 문장. "진단에서 멈추지 않고 수정까지, 못 하는 판정은 사람이 이어받습니다." 아래에 공개 주소 a11ychk.com.

정리하겠습니다. 진단에서 멈추지 않고 수정까지, 자동이 못 하는 판정은 사람이 이어받습니다.

한국어 가이드와 AI 수정 요청 문서, MCP 서버로 수정 작업에 바로 쓰이는 산출물을 냅니다. 자동이 닿지 못한 항목은 WCAG-EM 절차의 점검자 판정으로 받아 하나의 평가 기록으로 보고하고요.

a11ychk.com에서 무상으로 쓰실 수 있습니다. 검사 엔진과 KWCAG 규칙 카탈로그를 포함한 전체 소스도 공개해 뒀고요. 궁금한 점은 논문 댓글로 남겨 주시면 답을 드리겠습니다. 고맙습니다.

부록 1. 부록 1. 자체 검사 규칙이 메우는 영역

자체 검사 규칙이 메우는 영역: 초점 표시, 키보드 접근, 건너뛰기 링크, 새 창 고지, 자막 트랙, 마크업 유효성 검사(태그 열고 닫음·중첩·속성과 id 중복). 사이트 수준 검사는 페이지 제목의 유일성, 내비게이션의 일관성, 탐색 수단의 다양성을 따로 다룹니다.

부록 2. 부록 2. 두 사례의 진단 수치

적용 사례 요약

칸: 구분, 자체 웹사이트, 대학 웹사이트

- 표본 / 6페이지 / 7페이지(구조 6·무작위 1)
- 시점 / 검사 3회·수정 2회 / 2026년 8월
- 준수율 / 통합 86.4% → 100% / 자동 검사 81.3%
- 요소 수준 위반 / 9건 → 0건 / 20건(4개 규칙)
- 자동 확정 성공기준 / 13개 / 55개 / 16개 / 55개

부록 3. 부록 3. 보고서 내보내기 형식

1. 읽기 전용 공유 링크

2. PDF

3. 엑셀 호환 CSV

4. EARL

5. W3C WCAG-EM Report Tool이 읽는 평가 기록 스키마(기본값 1.0 호환, 2.0 절 앵커 매핑 형식 병행)

6. AI 수정 요청 문서(Markdown·JSON)

부록 4. 부록 4. (그림 1) 시스템 아키텍처

제출 원고 (그림 1) 시스템 아키텍처. 대상 URL(평가 범위 정의)에서 1. 표본 수집(크롤링, 구조적 표본, 무작위 표본, 직접 지정), 2. 페이지 진단(Chromium 렌더링, axe-core 검사, 자체 규칙 15종, 모바일 2차 검사), 3. 집계·산정(WCAG/KWCAG 이중 매핑, 사이트 수준 검사, 자동·수동·통합 준수율, 항목별 페이지 준수율 95% 기준), 4. 산출물(보고서, 한국어 가이드, AI 수정 요청, 전후 비교, 확장 표시, EARL) 순으로 이어집니다. 점검자 수동 판정이 3단계로 들어가고, 산출물에서 가이드에 따라 수정한 뒤 재검사가 대상 URL로 돌아갑니다.

읽기 도움말(통역과 낭독용)

- WCAG: 더블유캐그. 웹 콘텐츠 접근성 지침(Web Content Accessibility Guidelines). W3C가 낸 국제 표준
- KWCAG: 케이더블유캐그. 한국형 웹 콘텐츠 접근성 지침. 33개 검사항목으로 되어 있다
- WCAG-EM: 더블유캐그 이엠. 평가 범위와 표본을 정해 접근성 적합성을 평가하는 W3C 절차
- EARL: 얼. 평가와 보고 언어(Evaluation and Report Language). 검사 결과를 주고받는 표준 형식
- MCP: 엠시피. 모델 컨텍스트 프로토콜. AI 코딩 도구가 바깥 도구를 직접 부르게 하는 규약
- axe-core: 액스 코어. 널리 쓰이는 오픈소스 접근성 검사 엔진
- 성공기준: WCAG가 정한 판정 단위. 이 발표의 모수는 55개
- 검사항목: KWCAG가 정한 판정 단위. 모두 33개
- 오탐: 실제로는 문제가 없는데 도구가 위반으로 잡아내는 것
- 리플로우: 화면 폭이 좁아져도 가로로 밀리지 않고 내용이 다시 흐르는 것

참고 자료

- W3C, "Web Content Accessibility Guidelines (WCAG) 2.2," W3C Recommendation, Oct. 2023.
- 국립전파연구원, "한국형 웹 콘텐츠 접근성 지침 2.2," KS X OT0003:2022, 2022. 12. 28. 개정.
- W3C, "Website Accessibility Conformance Evaluation Methodology (WCAG-EM) 1.0," W3C Working Group Note, Jul. 2014.
- Deque Systems, "axe-core," https://github.com/dequelabs/axe-core
- 과학기술정보통신부·한국지능정보사회진흥원, "2025년 웹 접근성 실태조사," 2026.
- 한국웹접근성인증평가원, "웹 접근성 품질인증 심사 기준," https://www.wa.or.kr
- 디지털포용법, 법률 제20672호, 2025. 1. 21. 제정, 2026. 1. 22. 시행.
- 발표 시스템: https://www.a11ychk.com (소스 공개)

발표: 2026. 11. 5.(목)~7.(토) ACK 2026 세종대학교(온·오프라인 병행), 권기표(대구사이버대학교 정보화지원팀). 문의: isaaceryn@gmail.com