

**University of M'sila**

**Faculty of Mathematics and Computer Science**

**Department of Computer Science**

# Web Ergonomics Audit

Academic Year 2025-2026

Prepared and presented by SAOUDI Lalia

# 1. What is a Web Ergonomics Audit?

- A web ergonomics audit is a systematic evaluation of a website to measure:
  - Usability (ease of use)
  - Efficiency (speed of task completion)
  - Satisfaction (user comfort)
- It is grounded in:
  - ISO 9241
  - Web Content Accessibility Guidelines (WCAG)
  - Principles from Human-Computer Interaction

# 2. Audit Process (Step-by-Step)

- **Step 1: Define Objectives**
- Before analyzing anything, clarify:
  - Who are the users? (age, device, context)
  - What are the key tasks? (buy, read, register)
  - What devices? (mobile-first or desktop?)

# 2. Audit Process (Step-by-Step)

## **Step 2: Choose Evaluation Methods**

- 1. Heuristic Evaluation
  - Based on: Jakob Nielsen's 10 heuristics
- 2. User Testing
  - Observe real users as they complete tasks on the website and identify pain points
- 3. Automated Analysis
  - Use tools like:
  - Google Lighthouse
  - Wave (for accessibility)
  - GTmetrix (for performance)

# 2. Audit Process (Step-by-Step)

- **Step 3: Define Evaluation Criteria**
  - Determine which criteria you will evaluate the website against:
    - Usability (e.g., ISO 9241-11)
    - Accessibility (e.g., WCAG 2.1)
    - Performance (e.g., load times, smoothness)
    - Mobile ergonomics (e.g., mobile-first design, tap targets)

# 2. Audit Process (Step-by-Step)

## Step 4: Identify and Classify Issues

- Each issue should be:
- Clearly described
- Linked to a principle
- Rated by severity
- Example severity scale:
-  Critical → blocks task
-  Major → slows user
-  Minor → cosmetic

## 2. Audit Process (Step-by-Step)

- **Step 5: Provide Recommendations**
- Each problem must have:
  - A concrete fix
  - A justification
- 👉 Example:
- Problem: Small clickable area
- Fix: Increase button size to 48px
- Why: Improves touch accuracy (Fitts's Law)

# 2. Audit Process (Step-by-Step)

- **Step 6: Deliver Audit Report**
- A professional report includes:
  - 1. Executive Summary
    - Key issues
    - Overall UX score
  - 2. Methodology
    - Heuristics, tools, testing
  - 3. Findings
    - Organized by category
  - 4. Recommendations
    - Prioritized actions

# 3. Jakob Nielsen's 10 heuristics=usability evaluation method

- **What are Nielsen Heuristics?**
  - Jakob Nielsen's heuristics are a set of 10 usability principles used to quickly evaluate whether an interface is user-friendly.
- Nielsen Heuristics is for Usability
- They help evaluate:
  - Ease of use
  - Navigation
  - Feedback
  - Error handling
- 🙌 They are considered a heuristic evaluation method, meaning experts inspect an interface based on recognized rules rather than testing with users.

# Jakob Nielsen's 10 heuristics

- **2. When to Use Them?**
- Use Nielsen heuristics when you want to:
  - Audit a website or mobile app
  - Identify usability issues early
  - Improve UX without expensive user testing

# Jakob Nielsen's 10 heuristics

- 1.  **Visibility of System Status**
- The system should always keep users informed.
-  Good:
  - Loading indicators
  - "Item added to cart" message
-  Bad:
  - Clicking a button and nothing happens
-  Users should never wonder: "Did it work?"

# Jakob Nielsen's 10 heuristics

- 2. 🌍 **Match Between System and Real World**
- Use language and concepts familiar to users.
- ✅ Good:
  - “Cart”, “Checkout” in e-commerce
- ❌ Bad:
  - Technical terms like “Execute transaction protocol”
- 👉 Speak the user’s language, not the developer’s.

# Jakob Nielsen's 10 heuristics

- 3.  **User Control and Freedom**
- Users need easy ways to undo or exit actions.
-  Good:
  - Undo button
  - Cancel option
-  Bad:
  - Forced actions with no way back
-  Users should feel in control, not trapped.

# Jakob Nielsen's 10 heuristics

- 4.  **Consistency and Standards**
- Similar elements should behave the same.
-  Good:
  - Same button style across pages
-  Bad:
  - Green button = confirm on one page, cancel on another
-  Follow common conventions (don't reinvent basics).

# Jakob Nielsen's 10 heuristics

- **5.  Error Prevention**
- Prevent problems before they happen.
-  Good:
  - Password rules shown before typing
-  Bad:
- Showing error only after form submission
-  Prevention is better than correction.

# Jakob Nielsen's 10 heuristics

- 6.  **Recognition Rather Than Recall**
- Minimize memory load.
-  Good:
  - Visible menu options
  - Autofill suggestions
-  Bad:
  - User must remember previous inputs
-  Don't make users think too much.

# Jakob Nielsen's 10 heuristics

- 7. ⚡ **Flexibility and Efficiency of Use**
- Support both beginners and advanced users.
- ✓ Good:
  - Keyboard shortcuts
  - Autofill
- ✗ Bad:
  - Same slow process for everyone
- 👉 Experts should be able to move faster.

# Jakob Nielsen's 10 heuristics

- 8.  **Aesthetic and Minimalist Design**
- Avoid unnecessary information.
-  Good:
  - Clean layout
  - Focus on key actions
-  Bad:
  - Cluttered pages with too much content
-  Less is more.

# Jakob Nielsen's 10 heuristics

- 9.  **Help Users Recognize, Diagnose, and Recover from Errors**
- Error messages should be helpful.
-  Good:
  - “Email is invalid (missing @)”
-  Bad:
  - “Error code 504”
-  Explain the problem and how to fix it.

# Jakob Nielsen's 10 heuristics

- **10.  Help and Documentation**
- Provide help when needed.
-  Good:
  - FAQ
  - Tooltips
-  Bad:
  - No support or guidance
-  Even simple systems may need help.

# How to Apply Heuristics (Step-by-Step)

- **Step 1:** Choose a Task: Ex-Create account, Buy a product
- **Step 2:** Go Through Each Heuristic For every screen, ask: Does it respect this principle?
- **Step 3:** Identify Problems: Write issues like: “No feedback after clicking button”, “Error message unclear”
- **Step 4:** Assign Severity ; Typical scale: critical ,Major, Minor
- **Step 5:** Suggest Fixes Example: : Problem: No loading feedback , Fix: Add spinner, Heuristic: Visibility of system status

# 4. WCAG = accessibility evaluation standard

- **What is WCAG?**

- Web Content Accessibility Guidelines is the international standard used to evaluate whether a website is accessible to all users, including people with disabilities (visual, motor, cognitive, etc.).
-  It is essential in any web ergonomics audit.

# 4. WCAG = accessibility evaluation standard

- WCAG Structure (The 4 Principles):
  - WCAG is built on 4 core principles (called POUR):
    - Perceivable
    - Operable
    - Understandable
    - Robust



# Perceivable

- Users must be able to see or hear the content
- **Key checks:**
  - Images have alt text
  - Text has sufficient contrast
  - Content is readable (not too small)
-  **Example problem:**
  - Image without description ❌
-  **Fix:**
  - Add alt text ✅



# Operable

- Users must be able to interact with the interface

- **Key checks:**

- Website works with keyboard only
- Buttons are clickable and large enough
- No time limits that block users

-  Example problem:

- Menu only works with mouse ✗

-  Fix:

- Enable keyboard navigation ✓

# 3

# Understandable

- Content must be easy to understand
- **Key checks:**
  - Clear language
  - Predictable navigation
  - Helpful error messages
- 📌 Example problem:
  - "Error 123" ❌
- 📌 Fix:
- "Password must contain 8 characters" ✅

# 5. Robust

- Content must work across devices and assistive technologies
- Key checks:
  - Compatible with screen readers
  - Clean HTML structure
  - Works on different browsers
-  Example problem:
  - Broken layout in screen reader 
-  Fix:
  - Use proper semantic HTML 

# WCAG Conformance Levels

- WCAG has 3 levels:
  -  Level A → Basic accessibility
  -  Level AA → Standard (most required)
  -  Level AAA → Advanced (very strict)
-  In most projects, Level AA is the target.

# How to Use WCAG in Web Evaluation

- **Step 1:** Create a Checklist based on WCAG rules:
  - Alt text
  - Contrast
  - Keyboard access
  - Labels
- **Step 2:** Inspect the Website
  - Check each page manually:
    - Images
    - Forms
    - Navigation
    - Buttons

# How to Use WCAG in Web Evaluation

## Step 3: Use Tools

- Helpful tools:
  - Google Lighthouse
  - WAVE (accessibility checker)

## • Step 4: Identify Issues

### • Example:

- | • Issue             | WCAG Principle |
|---------------------|----------------|
| • Low contrast text | Perceivable    |
| • No keyboard nav   | Operable       |
| • Confusing error   | Understandable |

# How to Use WCAG in Web Evaluation

## **Step 5: Recommend Fixes**

- Example:
  - Increase contrast ratio
  - Add alt text
  - Improve labels

