



## Amina

29 Jahre alte  
Software-Entwicklerin

### SPRACHKENNTNISSE:

In Schriftsprache:  
Arabisch (Erstsprache),  
Deutsch (fließend),  
Englisch (fließend)

### KOGNITIVE FÄHIGKEITEN:

Sehr hohe kognitive  
Fähigkeiten

### IT-KOMPETENZ:

Expertenniveau (z. B. Visual  
Studio Code, Linux-  
Umgebungen, Git, Docker)

### ARBEITSWEISE:

Selbstständig, strukturiert,  
visuell orientiert

**„Ich brauche klare, visuelle Signale und Untertitel, damit ich jederzeit alles verstehen und effizient arbeiten kann – ohne auf Ton angewiesen zu sein.“**

### GERÄTE & TECHNIK

- **Geräte:**  
Desktop-PC mit Ubuntu 22.04 und Gnome-Desktop
- **Tastatur/Maus:**  
Mechanische Tastatur mit klarem akustischem Feedback (für besseres haptisches Gefühl), hochpräzise Maus
- **Bildschirm:**  
27" 4K-Monitor mit hohem DPI-Wert, um die Darstellung von Code und visuellen Komponenten zu optimieren
- **Hilfsmittel:**  
Untertitelung (Live Caption), visuelle Pop-up-Notifications, Slack-Channels sind visuell getriggert

### ZIELE

- Selbstständige Nutzung von Dev-Tools
- Barrierefreier Zugang zu komplexen technischen Informationen
- Sich als vollwertiges Teammitglied in inklusiven Entwicklungsumgebungen entfalten
- Qualität und Wartbarkeit durch klare, visuelle Dokumentation verbessern
- Gehörlose Entwickler\*innen sichtbarer machen und stärken

### EINFLUSSFAKTOREN

#### Motivierende Faktoren:

- ✓ Tools mit exzellenter visueller Rückmeldung und barrierefreien Funktionen
- ✓ Inklusive Teams, die auf klare schriftliche Kommunikation setzen
- ✓ Open-Source-Communitys mit aktiver Unterstützung für Accessibility
- ✓ Erfolgserlebnisse durch Selbstwirksamkeit und technologisches Empowerment

#### Frustrierende Faktoren:

- ✗ Fehlende oder unzuverlässige Untertitel und Transkriptionen
- ✗ Software, die ausschließlich akustisches Feedback nutzt
- ✗ Meetings ohne Rücksicht auf visuelle Zugänglichkeit oder klare Sprache
- ✗ Vorurteile und Mikroaggressionen wegen Hörbeeinträchtigung und Herkunft

### DETAILLIERTE USABILITY- & BARRIEREFREIHEITS-ANFORDERUNGEN

#### 1. Vollständige visuelle Informationsdarstellung

- Alle System- und Applikationswarnungen müssen zusätzlich zur (möglicherweise automatischen) Soundausgabe eine deutliche visuelle Komponente haben (z. B. Toast-Nachrichten, sich einblendende Farbbalken).

#### 2. Kein akustik-only Feedback

- Fehler- oder Erfolgsmeldungen dürfen nicht ausschließlich per Ton erfolgen. Sie brauchen eine klare, dauerhaft sichtbare Statuszeile („Upload abgeschlossen: Grün hinterlegt“) oder ein statisches Icon, das nur durch „Akzeptieren“ ausgeblendet werden kann.

#### 3. Untertitel & Transkription für alle Audio-/Videosegmente

- Automatisierte Untertitel (z. B. WebVTT) müssen bei Media-Elementen standardmäßig aktiviert sein.
- Für Live-Meetings: Ein Text-Chatfenster mit manueller oder automatischer Transkription.

#### 4. Häufig übersetzte und verständliche Terminologie

- Code-beispielbezogene Anleitungen sollten nicht auf akustische Erläuterungen angewiesen sein. Alle Beispiele sind in gut lesbare Textblöcke zu gliedern.
- Farbindikatoren (z. B. **grün** = erfolgreich, **rot** = Fehler) müssen von zusätzlichem Symbol (✓, ✗) begleitet sein – niemals Farbe allein.