

# Teilbericht: Frontend-Implementierung

## Einleitung (Zuständigkeitsbereich)

Im Rahmen der Frontend-Entwicklung war ich für die vollständige Umsetzung der Barrierefreiheit (Accessibility), der Kontakteverwaltung sowie der Chat-Funktionalität verantwortlich. Mein Aufgabenbereich umfasste:

### Accessibility:

Entwicklung eines dynamischen Accessibility-Menüs mit High-Contrast- und Big-Font-Modus, inklusive Speicherung der Einstellungen im localStorage und umfassender Anpassung der UI per CSS für maximale Zugänglichkeit.

### Kontakteverwaltung:

Umsetzung der gesamten Kontaktlogik: Anzeigen, Suchen, Hinzufügen, Filtern und Entfernen von Kontakten. Dazu gehörte auch die Anzeige von Kontaktdetails, Status (online/offline) und die Möglichkeit, direkt einen Chat mit einem Kontakt zu starten.

### Chat-System:

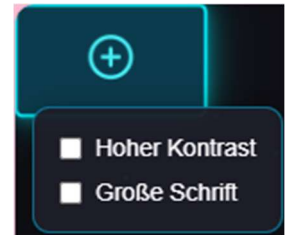
Entwicklung des kompletten Chat-Frontends:

- Dynamische Listen für Chatrooms und private Chats
- Erstellen und Löschen von Chatrooms
- Nachrichtenanzeige mit Gruppierung nach Datum
- Zeitstempel der Nachrichten
- Senden und Empfangen von Nachrichten
- Markierung des aktiven Chats
- Willkommensbildschirm bei leerem Chat
- Löschen von privaten Chats
- UI-Feedback und Statusanzeigen (z.B. Benutzeranzahl im Raum)
- Integration und Berücksichtigung der Accessibility-Features im gesamten Chatbereich

Ich war sowohl für die JavaScript-Logik als auch für die Gestaltung der zugehörigen CSS-Regeln und die Einbindung in die HTML-Struktur zuständig. Ziel war eine moderne, inklusive und benutzerfreundliche Oberfläche, die alle Kernfunktionen von NEXON abdeckt und für alle Nutzergruppen zugänglich ist.

## 1. Accessibility (Barrierefreiheit)

Für die Barrierefreiheit wurde ein eigenes Modul implementiert (jaccessibility.js), das einen stets sichtbaren Button (.a11y-btn) einfügt. Nutzer können damit per Checkbox zwischen einem High-Contrast-Modus und einer großen Schrift umschalten. Die Zustände werden im localStorage gespeichert und beim Laden der Seite automatisch wiederhergestellt. Die zugehörigen CSS-Regeln (accessibility.css) sorgen für starke Kontraste, größere Schrift und die Deaktivierung von Animationen, um die Zugänglichkeit für Menschen mit Sehbehinderung zu verbessern.

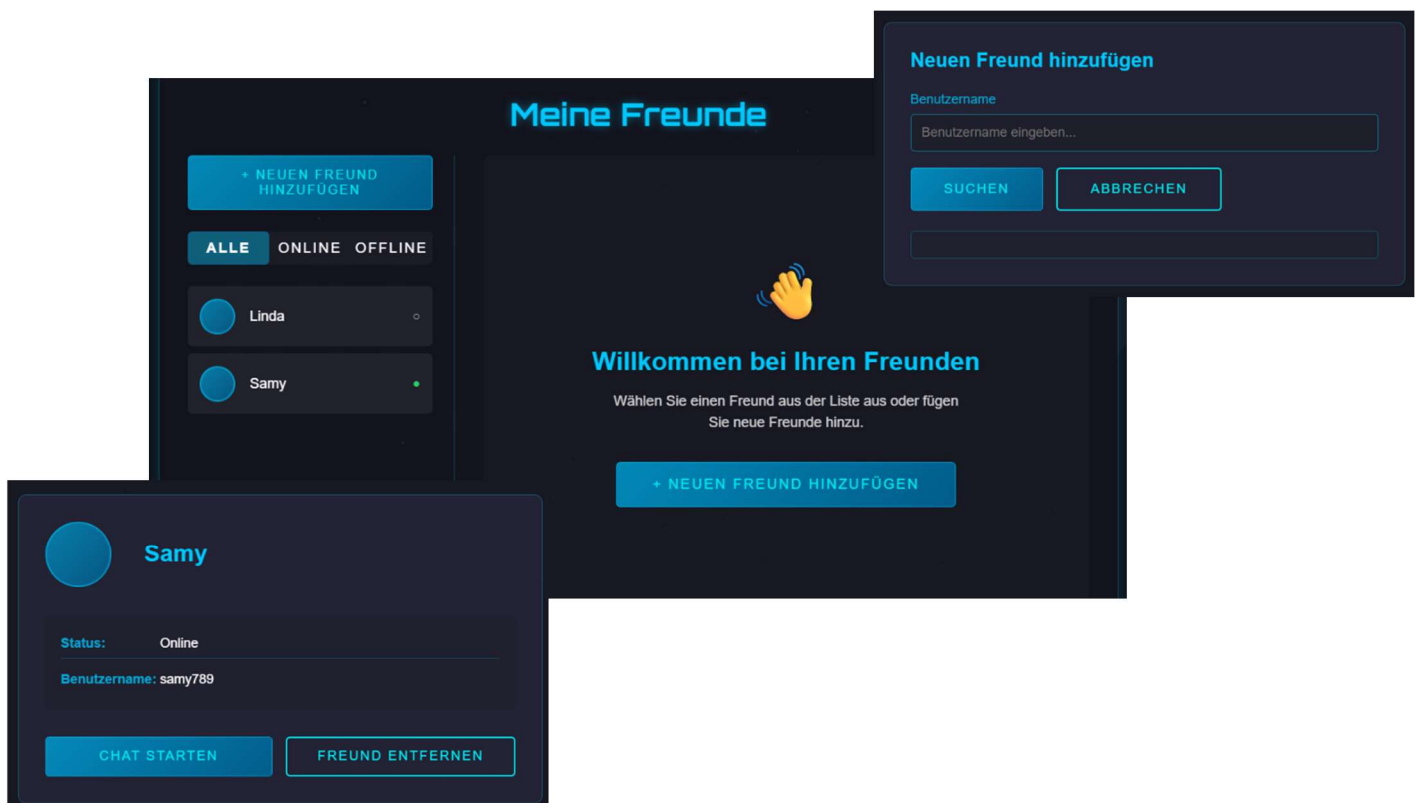


### Wichtige Features:

- Umschaltbarer High-Contrast-Mode und Big-Font-Mode
- Visuelle Anpassungen für alle UI-Komponenten (Buttons, Links, Formulare, Chat)
- Persistenz der Einstellungen über localStorage

## 2. Kontakteverwaltung

Die Kontakteverwaltung ist in contacts.js gekapselt. Sie ermöglicht das Anzeigen, Suchen, Hinzufügen und Entfernen von Kontakten. Die Kontaktliste wird dynamisch gerendert, Filter (alle/online/offline) sind möglich. Neue Kontakte können über ein Suchfeld gefunden und hinzugefügt werden. Die Daten werden in Demo-Arrays gehalten, in einer echten Anwendung kämen sie vom Server.

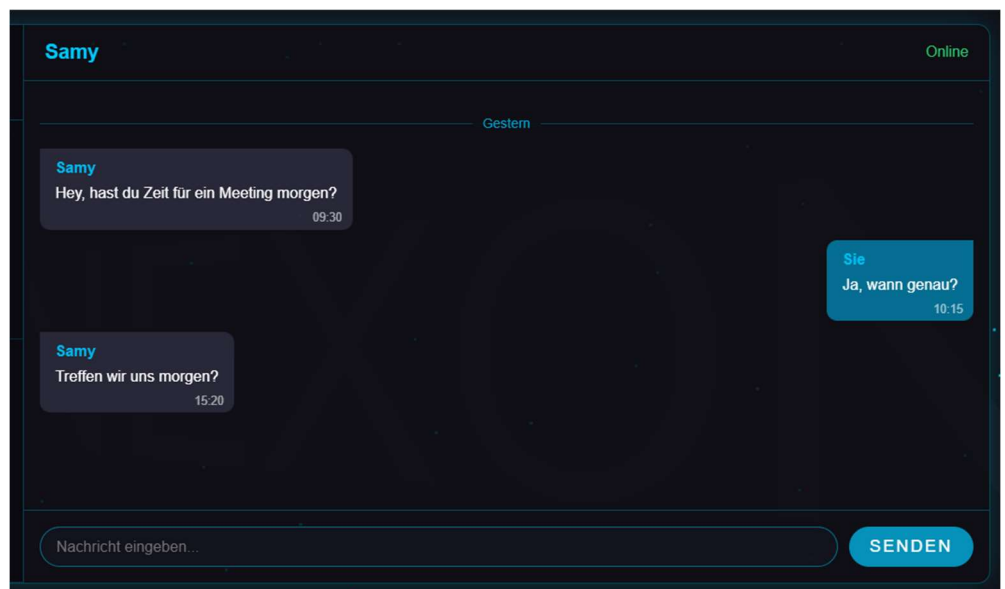
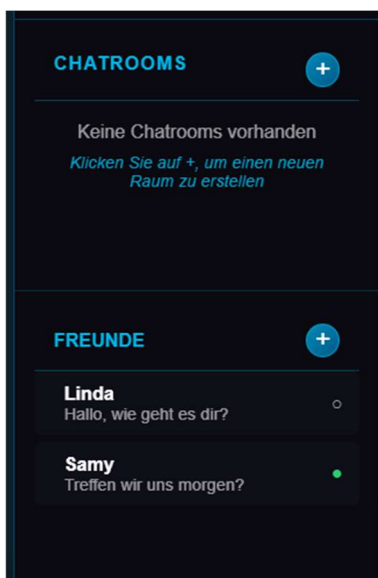


### Wichtige Features:

- Dynamische Kontaktliste mit Filterfunktion
- Suchfunktion für neue Kontakte
- Hinzufügen und Entfernen von Kontakten mit UI-Feedback
- Anzeige von Kontaktdetails und Status (online/offline)
- Start eines Chats mit einem Kontakt (Weiterleitung auf Chat-Seite)

## 3. Chat-System

Das Chat-System (chat-simple.js) unterstützt sowohl Chatrooms als auch private Chats. Die Chatrooms und privaten Chats werden dynamisch gerendert. Nutzer können neue Chatrooms erstellen und bestehende löschen. Nachrichten werden nach Datum gruppiert angezeigt, inklusive Datumsseparatoren. Für private Chats und Chatrooms gibt es jeweils eigene Löschfunktionen. Die UI ist so gestaltet, dass sie auch im Accessibility-Modus vollständig nutzbar bleibt.



### Wichtige Features:

- Dynamische Listen für Chatrooms und private Chats mit Löschfunktion
- Nachrichtenversand, Anzeige und Gruppierung nach Datum und Uhrzeit
- Willkommensbildschirm bei leerem Chat
- Visuelles Feedback und Statusanzeigen (z.B. Anzahl Benutzer im Raum)
- Integration mit Accessibility-Styles

## **Fazit**

Die Implementierung legt besonderen Wert auf Barrierefreiheit und Benutzerfreundlichkeit. Alle zentralen Interaktionen (Kontakte, Chat, Accessibility) sind modular, dynamisch und mit Fokus auf eine inklusive Nutzererfahrung umgesetzt. Die Accessibility-Features sind tief in die UI integriert und lassen sich jederzeit aktivieren, ohne die Funktionalität der Chat- und Kontaktmodule einzuschränken.