

Workflow Assist

Mobile-Applikation zur Erfassung und Verwaltung von Arbeitsplatz-Problemen

ÜK Modul 335 – Mobile-Applikation mit dem Ionic-Framework (Angular) realisieren

Name: Halil Cankilic

Klasse: INA23C

Workflow Assist	1
Mobile-Applikation zur Erfassung und Verwaltung von Arbeitsplatz-Problemen	1
1.1 Ausgangslage / Problemstellung	1
1.2 Zielsetzung	2
1.3 Zielgruppen	2
2. User Stories	2
3. Storyboard und UI-Elemente	3
Seite: Home	3
Seite: Ticket-Liste	4
Seite: Ticket erstellen	5
Seite: Ticket-Details	6
Seite: Einstellungen	7
4. Datenbank-Modell	8
5. Technische Umsetzung	9
6. Fazit	10

!!!Readme file lesen für Installation!!!

<https://github.com/Eiermitsucuk/Modul335>

1. Projektbeschreibung

1.1 Ausgangslage / Problemstellung

In vielen Ausbildungsbetrieben werden technische, infrastrukturelle oder ergonomische Probleme über unstrukturierte Kanäle gemeldet – zum Beispiel per E-Mail oder Zuruf. Probleme entstehen durch Medienbrüche, fehlende Transparenz, lange Reaktionszeiten und Doppelspurigkeiten. Lernende haben zudem oft keinen PC zur Hand, was die Hemmschwelle erhöht.

1.2 Zielsetzung

Entwicklung einer mobilen Ionic/Angular-Applikation zur schnellen und strukturierten Meldung von Arbeitsplatz-Problemen.

Zentrale Ziele:

- Standardisierung des Meldeprozesses
- Mobile-first Nutzung
- Transparenz über Status und Bearbeitung
- Reduzierte Reaktionszeit
- Reduktion von Medienbrüchen

1.3 Zielgruppen

- Lernende
- IT-Support / Facility Management
- Ausbildungsverantwortliche

2. User Stories

US1: Ticket erstellen

Als Lernender möchte ich schnell ein Problem melden, damit es zeitnah behoben wird.

Akzeptanzkriterien: Titel, Beschreibung, Kategorie, GPS oder manueller Standort, Validierung, Offline-Speicherung, Speicherung in Datenbank.

US2: Tickets anzeigen

Als Lernender möchte ich meine gemeldeten Tickets einsehen, damit ich den Überblick behalte.

Akzeptanzkriterien: Liste, Filter, Status-Badges, Detailansicht.

US3: Ticket-Details einsehen

Als Lernender möchte ich Details sehen, um den Kontext zu verstehen.

Akzeptanzkriterien: Beschreibung, Standort, Karte, Zeitstempel, Löschen, Status.

US4: Ticket-Status ändern

Als Support-Mitarbeiter möchte ich Status ändern, damit der Fortschritt sichtbar bleibt.

US5: GPS erfassen

Als Lernender möchte ich meinen Standort automatisch erfassen können.

US6: Offline arbeiten

Als Lernender möchte ich Tickets auch offline erstellen können.

US7: Benachrichtigungen

Als Lernender möchte ich über wichtige Ereignisse informiert werden.

US8: Dark Mode

Als Benutzer möchte ich zwischen hellem und dunklem Design wechseln.

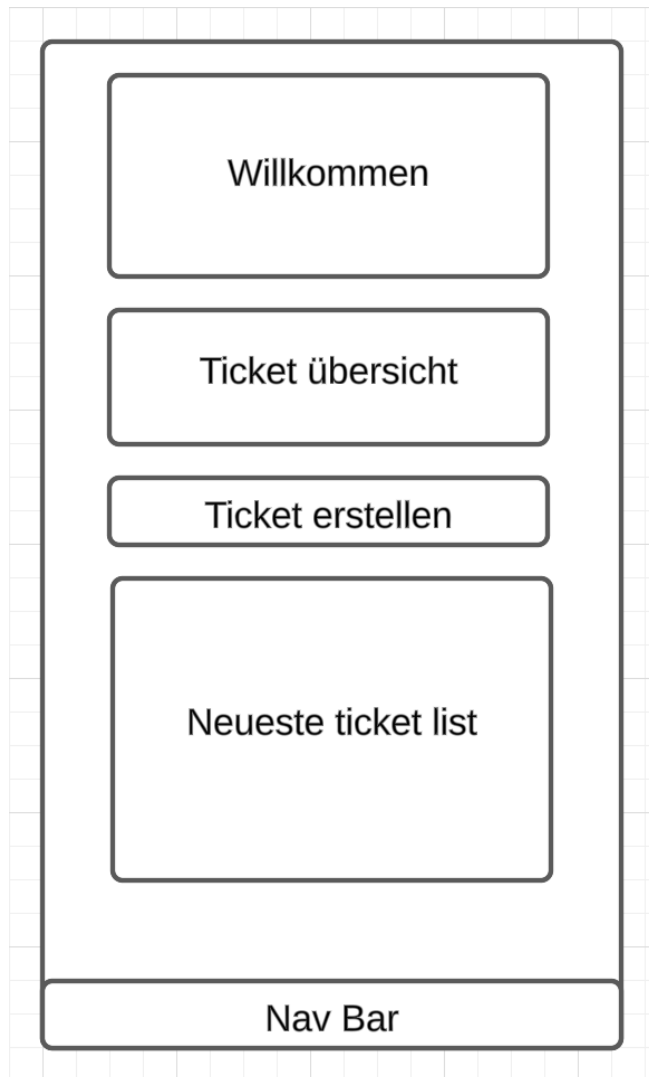
US9: Statistiken

Als Ausbildungsverantwortlicher möchte ich Ticket-Trends sehen.

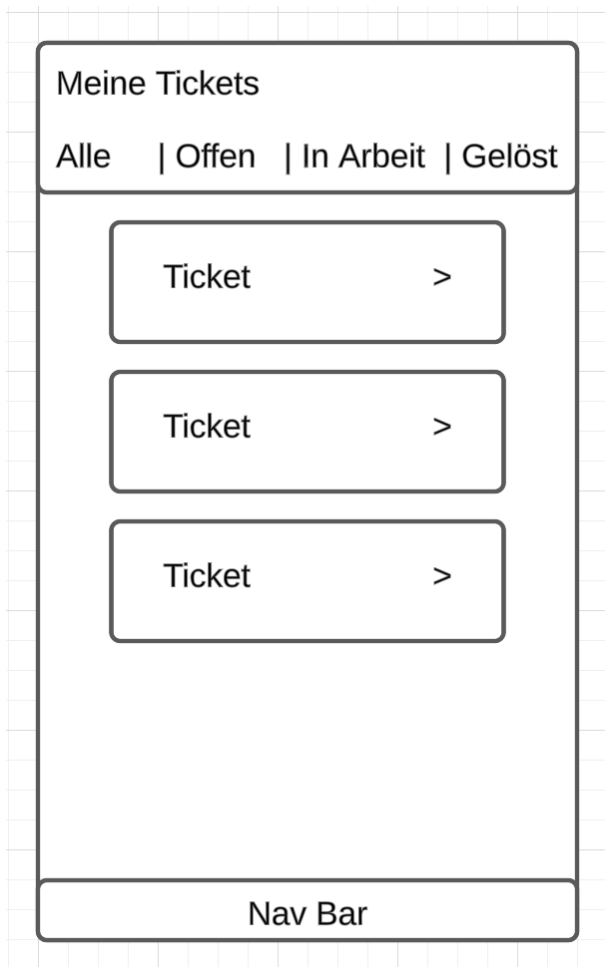
3. Storyboard und UI-Elemente

Die App basiert auf einer Tab-Navigation: Home, Tickets, Erstellen, Settings.

Seite: Home



Seite: Ticket-Liste



Seite: Ticket erstellen

← Neues Ticket

Titel

Kategorie

Beschreibung

Standort

Ticket erstellen

Seite: Ticket-Details

← Ticket Details

Titel/Kategorie

Beschreibung

Standort

Zeitstempel

Status Ändern

Seite: Einstellungen

Einstellungen

WorkFlow Assist
Beschreibung

Darstellung

Benachrichtigungen

Netzwerk & Speicher
Lokale Speicher leeren

Statistiken

4. Datenbank-Modell

tickets		
id	🔗 📄	UUID
title	📄	TEXT NN
description	📄	TEXT NN
category	📄	TEXT NN
status	📄	TEXT NN
location	📄	TEXT
user_id	📄	UUID
created_at	📄	TIMESTAMPZ
updated_at	📄	TIMESTAMPZ

5. Technische Umsetzung

Frontend: Ionic 8, Angular 20, SCSS.

Backend: Supabase (PostgreSQL + Storage + Realtime).

Mobile: Capacitor 7.4.

Plugins:

- Geolocation
- Local Notifications
- Network

Weitere Features: Offline-Mode, Filter, Dark Mode, GPS, Statistik-Dashboard.

Projektstruktur:

- Models
- Services (Supabase, GPS, Storage, Network, Notifications, Theme)
- Pages (Ticket-Create, Ticket-List, Ticket-Detail, Settings, Home, Tabs)
- Theme (variables.scss)

6. Fazit

WorkFlow Assist erfüllt alle Anforderungen des ÜK Modul 335 vollständig. Die App ermöglicht intuitive Problemerkfassung mit CRUD, GPS-Integration, Offline-Funktionalität und Dark Mode. Während der Entwicklung sammelte ich wertvolle Erfahrungen mit Capacitor-Plugins, Offline-First Strategien und Supabase-Integration. Besonders herausfordernd waren GPS-Implementierung und Online/Offline-Synchronisation. Das Projekt zeigt, dass mit Ionic und Angular professionelle, plattformübergreifende Apps effizient entwickelt werden können. Die App ist produktionsreif für den Einsatz in Ausbildungsbetrieben.