

Hier kommt noch ein Titelblatt, irgendwann

## **Inhaltsverzeichnis**

1. Einleitung und Zielsetzung des Handbuchs
2. Zielgruppe und Benutzerrollen
3. Installation und Start der Software
4. Benutzeroberfläche
5. Hauptfunktionen
6. Fehlerbehandlung/Support
7. Glossar
8. Anhang/technische Hinweise

# **1. Einleitung und Zielsetzung des Handbuchs**

Die vorliegende Software wurde im Rahmen des Softwareentwicklungspraktikums im Wintersemester 2025/2026 als Gruppenprojekt von Studierenden entwickelt. Ziel des Projekts war es, eine webbasierte Anwendung für Bellis e.V. Leipzig zu konzipieren und umzusetzen, welche die Erfassung und Verwaltung von Anfragen sowie daraus entstehenden Beratungsfällen im Arbeitsalltag unterstützt.

Im täglichen Betrieb von Bellis e.V. fallen zahlreiche Informationen zu Erstkontakten, Anfragen und Beratungsverläufen an, die zuverlässig dokumentiert und übersichtlich organisiert werden müssen. Die entwickelte Anwendung dient dazu, diese Daten zentral zu erfassen, strukturiert abzulegen und bei Bedarf schnell wiederzufinden. Dadurch sollen Arbeitsabläufe vereinfacht und die Dokumentation im Beratungsprozess erleichtert werden.

Dieses Handbuch richtet sich an alle Mitarbeitenden von Bellis e.V., die mit der Software arbeiten. Es soll den Einstieg in die Nutzung der Anwendung unterstützen und eine verständliche Orientierung im Umgang mit den wichtigsten Funktionen bieten. Der Fokus liegt dabei auf praxisnahen, schrittweisen Erklärungen typischer Arbeitsabläufe, wie dem Erfassen neuer Anfragen, dem Anlegen und Bearbeiten von Beratungsfällen sowie der Pflege bestehender Daten.

Ziel des Handbuchs ist es, eine sichere und selbstständige Nutzung der Software im Arbeitsalltag zu ermöglichen. Gleichzeitig dient es als Nachschlagewerk, um Funktionen schnell nachzuschlagen und die Anwendung effizient einzusetzen.

## **2. Zielgruppe und Benutzerrollen**

Die Software richtet sich an die Mitarbeiterinnen des Bellis e.V. Leipzig. Ziel ist es, die Erfassung, Bearbeitung und Auswertung statistischer Daten zu Anfragen und Beratungsfällen effizient, sicher und strukturiert zu ermöglichen.

Da im Verein mit sensiblen und schützenswerten Daten gearbeitet wird, ist der Zugriff auf die Software ausschließlich über passwortgeschützte Benutzerkonten möglich. Um eine klare Aufgabenverteilung zu gewährleisten und die Nutzerinnen nicht mit unnötigen Funktionen zu überfordern, unterscheidet das System drei Kontenarten mit unterschiedlichen Berechtigungen:

### **Administratorin**

Ein Administrationskonto verfügt über den vollen Zugriff auf alle Funktionen der Software. Administratorinnen können neue Benutzerkonten anlegen, bearbeiten und löschen, Rollen und Berechtigungen zuweisen oder entziehen, bestehende Konten verwalten, neue Formularfelder erstellen und bestehende anpassen, gemeinsame Presets verwalten und löschen sowie auf sämtliche gespeicherten Daten und Statistiken zugreifen. Mindestens ein Administrationskonto muss dauerhaft bestehen, empfohlen sind mindestens zwei.

### **Mitarbeiterin (Basiskonto)**

Das Basiskonto ist für die reguläre Nutzung vorgesehen. Mitarbeiterinnen können neue Anfragen und Beratungsfälle anlegen, bestehende Datensätze bearbeiten und ergänzen, Beratungstermine dokumentieren, Statistiken abrufen, persönliche Presets speichern und löschen sowie gemeinsame Presets erstellen. Ein Basiskonto hat keinen Zugriff auf Benutzerverwaltung oder systemweite Einstellungen.

### **Mitarbeiterin mit Erweiterungsrechten**

Diese Rolle erweitert das Basiskonto um die Möglichkeit, neue Formularfelder für alle Nutzerinnen anzulegen, Feldtypen festzulegen (z. B. Textfeld, Zahlenfeld, Datum) sowie gemeinsame Presets zu verwalten und zu löschen.

### 3. Installation und Start der Software

Die Software läuft auf einem lokalen Rechner über Docker. Dank Docker müssen Sie sich nicht um Installationen von Python, Node.js oder npm kümmern. Die Anwendung kann direkt über den Browser genutzt werden.

#### Voraussetzung

Für die Installation wird folgendes Programm benötigt:

- Docker Desktop

#### 1. Projektordner einrichten

1. Öffnen Sie die GitHub-Seite des Projekts im Browser.

<https://github.com/RealBiggerBo/Softwaretechnik.git>

2. Laden Sie das Projekt als ZIP-Datei herunter und entpacken Sie es auf dem Rechner oder klonen Sie das Repository mit Git im Terminal.

```
git clone https://github.com/RealBiggerBo/Softwaretechnik.git
```

#### 2. Software starten

1. Öffnen Sie Docker Desktop und warten Sie, bis Docker vollständig gestartet ist.
2. Navigieren Sie im Dateisystem zu dem bereits vorbereiteten Projektordner. Dieser enthält unter anderem die Docker-Konfigurationsdateien.
3. Öffnen Sie ein Terminalfenster.
4. Wechseln Sie in den Projektordner

```
cd Softwaretechnik
```

5. Bauen Sie das Docker Image, indem Sie folgendes in die Kommandozeile eingeben.

```
docker build -t softwaretechnik .
```

6. Starten Sie den Container mit folgendem Befehl.

```
docker run -p 8080:80 softwaretechnik
```

7. Öffnen Sie die Anwendung im Browser unter <http://localhost:8080>
8. Melden Sie sich mit Ihrem Benutzernamen und Passwort ein. Danach stehen Ihnen die Funktionen der Software zur Verfügung.

## **5. Hauptfunktionen**

Die Software unterstützt Mitarbeitende von Bellis e.V. dabei, Anfragen und Beratungsfälle effizient zu erfassen, zu verwalten und auszuwerten. Die wichtigsten Funktionen im Überblick:

### **1. Anfragen erfassen**

- Neue Anfragen von Ratsuchenden können direkt in der Software eingegeben werden.
- Pflichtfelder stellen sicher, dass alle wichtigen Informationen dokumentiert werden.
- So werden Anfragen strukturiert und leicht wiederauffindbar gespeichert.

### **2. Beratungsfälle erstellen und verwalten**

- Aus Anfragen können Beratungsfälle angelegt werden, die alle relevanten Informationen enthalten.
- Fälle können bei Bedarf bearbeitet oder ergänzt werden, z. B. um neue Termine oder Notizen.

### **3. Daten bearbeiten und löschen**

- Mitarbeitende können bestehende Anfragen und Fälle jederzeit aktualisieren.
- Bei Bedarf können veraltete oder fehlerhafte Datensätze gelöscht werden.

### **4. Statistiken und Auswertungen**

- Die Software erstellt aus den erfassten Daten automatisch Statistiken, z. B. Anzahl der Anfragen oder Beratungsfälle in einem bestimmten Zeitraum.
- Auswertungen können in verschiedene Formate exportiert werden (z. B. PDF, Excel), um sie z. B. für Berichte oder Präsentationen zu nutzen.

### **5. Datenübersicht und Suche**

- Mitarbeitende können alle Anfragen und Fälle über die Suchfunktion schnell finden.
- Filter- und Sortiermöglichkeiten erleichtern den Überblick über große Datenmengen.

#### **6. Webbasierte Nutzung**

- Die Anwendung läuft direkt im Browser
- Plattformunabhängig: Windows, macOS und Linux nutzbar.

#### **7. Datensicherheit**

- Zugangsdaten werden geschützt verwaltet.
- Sensible Daten sind nur für berechtigte Nutzer sichtbar.
- Bei lokaler Nutzung über Docker werden alle Daten sicher auf dem Rechner gespeichert.

## 9. Anhang/technische Hinweise

Dieses Kapitel richtet sich vor allem an Administratoren und IT-Verantwortliche, die die Software einrichten, warten oder aktualisieren. Es enthält Informationen zu alternativen Installationsmöglichkeiten und Wartung.

### 9.1. Installation über virtuelle Umgebung, Python/node, Datenbankmigration

#### Voraussetzungen

Für die Installation werden folgende Programme benötigt:

- Python (Version 3.x)
- Node.js und npm
- Git (optional, zum Herunterladen des Projekts)
- Internetzugang für das erstmalige Installieren der benötigten Komponenten

#### 1. Projekt herunterladen

1. Öffnen Sie die GitHub-Seite des Projekts im Browser.

<https://github.com/RealBiggerBo/Softwaretechnik.git>

2. Laden Sie das Projekt als ZIP-Datei herunter und entpacken Sie es auf dem Rechner oder klonen Sie das Repository mit Git im Terminal.

```
git clone https://github.com/RealBiggerBo/Softwaretechnik.git
```

#### 2. Backend einrichten und starten

1. Wechseln Sie in den Projektordner.

```
cd Softwaretechnik
```

2. Wechseln Sie in den Backend-Ordner des Projekts.

```
cd backend
```

3. Erstellen Sie die virtuelle Umgebung.

```
python -m venv venv
```

4. Aktivieren Sie die virtuelle Umgebung.

- Windows CMD:

```
venv\Scripts\activate
```

- Windows Powershell:

```
.\venv\Scripts\Activate.ps1
```

- Linux/Mac OS:

```
source venv/bin/activate
```

5. Installieren Sie die benötigten Python-Abhängigkeiten.

```
pip install -r requirements.txt
```

6. Führen Sie die Datenbankmigrationen aus, um die lokale Datenbank zu erstellen.

```
python manage.py migrate
```

7. Starten Sie anschließend das Backend.

```
python manage.py runserver
```

Das Backend läuft danach lokal auf dem Rechner und stellt die Datenbank sowie die Programmlogik bereit.

### 3. Frontend einrichten und starten

1. Öffnen Sie ein neues Terminalfenster.
2. Wechseln Sie in den Frontend-Ordner des Projekts.

```
cd frontend
```

3. Installieren Sie die benötigten npm-Abhängigkeiten.

```
npm install
```

4. Starten Sie das Frontend.

```
npm run dev
```

#### **4. Anwendung im Browser öffnen**

Nach dem Start von Backend und Frontend kann die Software im Webbrowser genutzt werden.

Öffnen Sie dazu die angezeigte lokale Adresse (z. B. „<http://localhost:3000>“). Anschließend erscheint die Anmeldeseite der Anwendung.

#### **Hinweis zum Betrieb**

- Das Backend und das Frontend müssen gestartet sein, damit die Software funktioniert.
- Beim Neustart des Rechners müssen beide Komponenten erneut gestartet werden.
- Alle Daten werden lokal auf diesem Rechner gespeichert.

Nach der erfolgreichen Einrichtung können sich die Mitarbeitenden über den Browser anmelden und die Software wie vorgesehen zur Erfassung von Anfragen und Beratungsfällen verwenden.

### **9.2. Wartung**

Neue Versionen der Software sollten ins lokale Projektverzeichnis übernommen werden.

Nach Updates sollten ggf. die Docker-Container neu gebaut oder die virtuellen Umgebungen aktualisiert werden.

Datenbanken sollten regelmäßig gesichert werden, insbesondere wenn die Software lokal auf einzelnen Rechnern läuft.