

Das ist eine detaillierte technische und funktionale Dokumentation für die Liegestütz-Trainingsapp **Version 7.0**.

Dokumentation: Liegestütz-Trainingsapp (v7.0)

1. Einleitung

Diese Webanwendung dient der Planung, Durchführung und Protokollierung von körperlichem Training. Der Fokus liegt auf einer progressiven Liegestütz-Routine, ergänzt durch Kardio-, Unterwegs- und benutzerdefinierte Übungen sowie einer Bild-Fortschrittsdokumentation. Die Anwendung basiert auf PHP und benötigt keine SQL-Datenbank, da alle Daten in JSON- oder Textdateien gespeichert werden.

2. Dateistruktur und Funktionsweise

A. Hauptdateien (Frontend & Logik)

index.php (Das Dashboard)

- **Funktion:** Die zentrale Startseite. Sie berechnet basierend auf einem festen Startdatum die aktuelle Trainingswoche und zeigt die Übungen für die aktuelle Kalenderwoche an.
- **Logik:**
 - Errechnet Trainingswoche und verbleibende Tage bis zum Ziel.
 - Unterscheidet zwischen Kraft-Tagen (Mo, Mi, Fr) und Kardio-Tagen (Di, Do, Sa).
 - **Modus-Schalter:** An Kraft-Tagen kann zwischen "Standard" und "Unterwegs" gewechselt werden. Dieser Status wird pro Tag in `kraft_modes.json` gespeichert.
 - **Anzeige:** Lädt Übungen aus `inhalte.txt` und filtert diese basierend auf Gültigkeitsdatum (`created_at`, `valid_until`) und Archivierungsstatus.
 - **Interaktion:** Checkboxes senden per AJAX (JavaScript fetch) den Status an den Server. Zusätzlich absolvierte Übungen (die nicht im Plan standen) werden am Ende des Tages dynamisch eingeblendet.
 - **Tagebuch:** Enthält ein Formular zum Speichern von Freitext-Notizen in `trainings_verlauf.txt`.

backend.php (Administration)

- **Funktion:** Verwaltungsoberfläche für alle Inhalte der App.
- **Features:**
 - Hinzufügen, Bearbeiten (Titel, Gültigkeitszeitraum, Kategorie-Zuteilung) und Löschen von Übungen.

- Erstellen und Löschen von eigenen Kategorien (z.B. "Calisthenics").
- Ändern des Webseitentitels und der App-Versionnummer.
- Übersichtliche, nach Kategorien gruppierte Liste aller Übungen.
- **Speicherung:** Alle Konfigurationen werden in `inhalte.txt` geschrieben.

picture.php (Fortschrittsdokumentation)

- **Funktion:** Upload von Körperbildern inklusive Gewicht und Notizen.
- **Features:**
 - Bilder werden in den Ordner `pix/` hochgeladen (Dateiname: `tmmjj_hhmmss.ext`).
 - Erstellt automatisch ein speicherschonendes Thumbnail (Breite 250px).
 - Speichert Metadaten (Gewicht, Datum, Notiz) in `pictures_data.json`.
 - Schreibt einen Log-Eintrag in `trainings_verlauf.txt`.
 - Bietet eine Galerieansicht mit Modal (Großansicht) und Rotations-Funktion.

unterwegs.php & zusatz.php

- **Funktion:** Statische Seiten für spezifische Kategorien ("Unterwegs" und "Zusätzlich").
- **Logik:** Zeigen alle aktiven Übungen der jeweiligen Kategorie an. Dienen als schnelle Abhak-Liste außerhalb des Wochenplans.

dynamic_page.php

- **Funktion:** Eine generische Schablone für selbst erstellte Kategorien.
- **Logik:** Wird über den Parameter `?cat=ID` gesteuert (z.B. `dynamic_page.php?cat=calisthenics_123`). Lädt dann dynamisch alle Übungen, die dieser Kategorie zugeordnet sind.

log_viewer.php

- **Funktion:** Eine simple Ansicht, die den Inhalt von `trainings_verlauf.txt` ausliest und anzeigt. Wird meist als Modal (iFrame) in der `index.php` eingebunden.

footer.php

- **Funktion:** Enthält die globale Navigation und die Versionsnummer.
- **Logik:** Wird mittels `include` in fast alle Seiten eingebunden. Das CSS sorgt für eine responsive Darstellung (Umbruch bei kleinen Bildschirmen).

B. Datenspeicher (Backend & Persistenz)

Die App speichert Daten an folgenden Orten. Für ein Backup müssen diese Dateien gesichert werden.

1. `inhalte.txt` (Konfiguration):

- Enthält alle Übungsdefinitionen (ID, Beschreibung, Startdatum, Enddatum, Kategorien), den Seitentitel, die Version und benutzerdefinierte Kategorien.

2. **checked_exercises.json (Status):**
 - Speichert den Zustand der Checkboxes.
 - Format der Keys: exercise_YYYY-MM-DD_UNIQUE-ID. Dies garantiert, dass Änderungen am Trainingsplan (z.B. neue Übung hinzugefügt) den Status vergangener Tage *nicht* verfälschen.
 3. **trainings_verlauf.txt (Logbuch):**
 - Eine fortlaufende Textdatei (neuste Einträge oben).
 - Speichert: Abgehakte Übungen (Datum + Name), manuelle Tagebucheinträge und Bild-Uploads (Gewicht + Notiz).
 4. **kraft_modes.json (Tages-Modus):**
 - Speichert für spezifische Datumswerte, ob an einem Kraft-Tag der Modus "Standard" oder "Unterwegs" gewählt wurde (z.B. "2026-01-18": "unterwegs").
 5. **pictures_data.json (Bild-Metadaten):**
 - Speichert Referenzen zu den Bildern, Thumbnails, das Gewicht und Notizen.
 6. **Ordner pix/:**
 - Enthält die hochgeladenen Originalbilder und die generierten Thumbnails (thumb_...).
-

3. Verzahnung der Dateien

- **Zentrale Steuerung:** backend.php schreibt in inhalte.txt.
 - **Konsum:** index.php, unterwegs.php, zusatz.php und dynamic_page.php lesen inhalte.txt, um zu wissen, welche Übungen existieren.
 - **Status-Abgleich:** Alle Frontend-Seiten lesen und schreiben in checked_exercises.json. Wenn in zusatz.php ein Haken gesetzt wird, erkennt index.php dies anhand der ID und zeigt die Übung am entsprechenden Tag unter "Zusätzlich absolviert" an.
 - **Logging:** Fast alle Interaktionen (Abhaken, Upload, Notiz) triggern einen Schreibvorgang in trainings_verlauf.txt.
-

4. Beispielhafter Ablauf eines Trainingstages

Ein typischer User-Flow in Version 7.0 sieht wie folgt aus:

1. **Start:** Der Nutzer öffnet index.php.
2. **Information:** Er sieht im Header das heutige Datum, die aktuelle Trainingswoche (z.B. Woche 25) und das Ziel.
3. **Planung:**
 - Es ist Freitag (Kraft-Tag). Der Nutzer ist auf Reisen.
 - Er klickt in der Tagesbox auf den orangenen Button "Modus: STANDARD", um auf "UNTERWEGS" umzuschalten. Die Seite lädt neu, der Modus wird in kraft_modes.json gespeichert.
 - Die Übungen ändern sich von "Bankdrücken" zu "Liegestütze".

4. **Training:**

- Der Nutzer absolviert die erste Übung und klickt die Checkbox an.
- *Hintergrund:* Der Status wird in checked_exercises.json gespeichert. Ein Eintrag "18.01.2026 - Liegestütze" wird oben in trainings_verlauf.txt geschrieben.

5. **Zusatz:**

- Der Nutzer möchte noch Dehnübungen machen. Er navigiert über den Footer zu "Zusätzlich" (zusatz.php).
- Er hakt "Mobility Session" ab.
- Zurück auf der index.php erscheint nun im heutigen Tag unten der Bereich "Zusätzlich absolviert: Mobility Session".

6. **Dokumentation:**

- Der Nutzer klickt im Footer auf "Bild".
- Er lädt ein Foto hoch, gibt "85,4 kg" und die Notiz "Gutes Gefühl heute" ein.
- Das Bild landet in pix/, die Daten in pictures_data.json und ein Log-Eintrag ("Bild hochgeladen...") in trainings_verlauf.txt.

7. **Abschluss:**

- Auf der Startseite schreibt der Nutzer ins Textfeld: "Training war anstrengend, aber gut".
- Er klickt auf "Vergangene Einträge" (öffnet log_viewer.php im Modal), um seine Historie zu prüfen.

Dies gewährleistet eine lückenlose Dokumentation und hohe Flexibilität bei der Trainingsgestaltung.