

Pattern Pix Trainer Panel

Andrea Stöckelberger
Dominik Kerzl

Inhaltsverzeichnis

1	Zielbestimmung.....	3
1.1	Ziel des Projekts.....	3
1.2	Zielgruppe	3
2	Produkteinsatz	3
2.1	Anwendungsbereich	3
2.2	Systemumgebung.....	3
3	Funktionale Anforderungen.....	3
3.1	Benutzeroberfläche (Views).....	3
3.2	Datenverarbeitung	4
4	Nicht-funktionale Anforderungen	4
5	Architektur & Datenhaltung.....	4
6	Abgrenzung	5
7	User Stories.....	5
8	Tasks	6
8.1	Startseite: Kinderliste.....	6
8.2	Kind-Detailansicht	7
8.3	Analyseansicht für ein Kind	7
8.4	Kind anlegen (Eingabeansicht).....	8
8.5	Speicheroptionen: Datenbank vs. JSON	8

1 Zielbestimmung

1.1 Ziel des Projekts

Die Anwendung „Pattern Pix Konzentrationstraining – Trainer Panel“ soll es Trainern ermöglichen, Kinderprofile sowie deren Trainingsleistungen übersichtlich zu verwalten, zu analysieren und auszuwerten. Die Anwendung dient zur Unterstützung des Konzentrationstrainings „Pattern Pix“ und soll ein benutzerfreundliches, robustes Werkzeug zur Erfolgsmessung und Verlaufskontrolle sein.

1.2 Zielgruppe

Pädagogische Fachkräfte (z. B. Lehrer:innen, Trainer:innen)

Erziehungsberechtigte mit professioneller Begleitung

2 Produkteinsatz

2.1 Anwendungsbereich

Desktop-Anwendung zur Unterstützung der mobilen Pattern-Pix-App

Einsatz in Schulen und therapeutischen Einrichtungen

Fokus auf lokaler, offline-fähiger Datenverarbeitung

2.2 Systemumgebung

Plattform: Windows 10 oder höher

Technologien: .NET (WPF), MVVM, Entity Framework Core, SQLite / JSON

3 Funktionale Anforderungen

3.1 Benutzeroberfläche (Views)

Startansicht (View 1): Übersicht aller Kinder (Listenansicht, Suche, Filter)

Detailansicht (View 2): Einzelansicht mit persönlichen Daten und Trainingsverlauf

Analyseansicht (View 3): Visualisierung der Leistungen über Zeiträume (Diagramme)

Hinzufügeansicht (View 4): Ansicht um Kind hinzuzufügen

3.2 Datenverarbeitung

Eigenes Datenmodell: Child, Training

Datenhaltung:

Wählbar: SQLite (Datenbank) oder JSON-Dateien (lokale Serialisierung)

Umschaltbar zur Laufzeit

Eventbasierte Kommunikation zwischen ViewModels

Eingaben über UI mit Validierung (Pflichtfelder, Datentypen)

4 Nicht-funktionale Anforderungen

Benutzerfreundlichkeit:

Responsives Design, keine feste Fenstergröße

Tooltips, Hilfetexte, barrierearme Farben

Fehlertoleranz

try-catch-Fehlerbehandlung bei Datei-/Datenbankzugriffen

Gültigkeitsprüfungen aller Benutzereingaben

Diagramme

Balken- und Liniendiagramme zur Darstellung von Leistungsdaten

Vergleich mit Altersdurchschnitt

5 Architektur & Datenhaltung

Architektur: MVVM mit Dependency Injection

Datenquellen Umschaltbar:

SQLite via Entity Framework Core

JSON via Serialisierung

Erweiterbarkeit:

Modular, Unterstützung zukünftiger Features wie Cloud-Sync oder Mehrsprachigkeit

6 Abgrenzung

Nicht enthalten	Begründung
Online-Synchronisierung	Fokus auf lokale Nutzung, spätere Erweiterung möglich
Benutzerverwaltung / Login	Anwendung ist für Einzelnutzer konzipiert
Durchführung der Trainings	Trainings erfolgen ausschließlich über die mobile Pattern Pix App

7 User Stories

1. Ein Trainer muss eine Liste mit allen Kindern auf der Hauptseite angezeigt bekommen, wobei jeder Listeneintrag aus ID, Name, Alter und Datum des letzten Trainings besteht.
2. Ein Trainer muss die Einträge in der Liste aufsteigend oder absteigend nach jedem Attribut sortieren können.
3. Ein Trainer muss beim Klicken auf einen Eintrag folgende Details zu einem Kind angezeigt bekommen: Name, Geburtsdatum, letztes Training, Anzahl der Trainingseinheiten.
4. Ein Trainer muss die Möglichkeit haben durch Klicken auf einen Button zur Detailansicht des ausgewählten Kindes zu gelangen.
5. Ein Trainer soll in der Suchleiste durch Eingabe eines Wertes nach Einträgen in der Liste suchen können.
6. Ein Trainer muss ein neues Kinderprofil anlegen können und durch Klicken auf einen Button zur Eingabeansicht gelangen.
7. Ein Trainer muss Vorname, Nachname und Geburtsdatum eines Kindes in die entsprechenden Felder eintragen können und diese durch Klicken auf einen Button speichern können.
8. Ein Trainer muss auf der Detailseite unter dem Namen, Geburtsdatum und Alter eines Kindes alle Trainingseinheiten dieses Kindes mit Ergebnissen anzeigen können. Der Eintrag eines Trainings umfasst: Datum, Uhrzeit, Symmetrietyp, benötigte Zeit und Fehler.
9. Ein Trainer soll auf der Detailseite eine Visualisierung des Trainingsverlaufs als Verlaufskurve von Fehlern und benötigter Zeit anzeigen können.
10. Ein Trainer muss auf der Detailseite die Möglichkeit haben durch Klicken auf einen Button zur Analyseansicht des entsprechenden Kindes zu gelangen.
11. Ein Trainer muss den Fortschritt eines Kindes visuell als Verlaufskurve darstellen können, wobei Fehler und Zeit berücksichtigt werden. Zusätzlich muss ein Trainer

auswählen können 1) für welchen Symmetrietyp 2) über welchen Zeitraum die Ergebnisse dargestellt werden.

12. Ein Trainer soll frühere Leistungen eines Kindes mit neueren vergleichen können, indem die beiden Kurven übereinandergelegt werden.
13. Ein Trainer muss die durchschnittliche Zeit und durchschnittlichen Fehler eines Kindes mit dem Durchschnitt von Gleichaltrigen in einem Balkendiagramm vergleichen können. Eine Aufschlüsselung nach Symmetrietyp soll möglich sein.
14. Ein Trainer muss zwischen Speicherung in der Datenbank und Speicherung als JSON-File umschalten können mithilfe eines Switches.

8 Tasks

8.1 Startseite: Kinderliste

User Stories: 1, 2, 5

Tasks:

- **Liste aller Kinder anzeigen**
 - Kind-Einträge mit ID, Name, Alter, letztem Trainingsdatum anzeigen.
- **Sortierfunktion pro Attribut umsetzen**
 - Sortierung nach ID, Name, Alter, letztem Training (auf-/absteigend).
- **Suchleiste für Kinderliste umsetzen**
 - Filtert Kinder basierend auf Name oder ID.
- **Altersberechnung auf Basis des Geburtsdatums implementieren**
 - Wird dynamisch angezeigt und aktualisiert sich täglich.

8.2 Kind-Detailansicht

User Stories: 3, 4, 8, 9, 10

Tasks:

- **Navigation zur Kind-Detailansicht implementieren**
 - Bei Klick auf einen Button wird die Detailseite geöffnet.
- **Anzeige der Basisdaten auf Detailseite**
 - Bei Klick auf einen Listeneintrag werden Basisdaten angezeigt:
Name, Geburtsdatum, Alter, letztes Training, Anzahl der Trainings.
- **Liste aller Trainings dieses Kindes anzeigen**
 - Darstellung als Tabelle: Datum, Uhrzeit, Symmetrie, Zeit, Fehler.
- **Visualisierung des Trainingsverlaufs (Linien-Chart)**
 - Verlaufskurven für Fehleranzahl und benötigte Zeit über Zeit.
- **Button zur Analyseansicht hinzufügen**
 - Navigiert zur tieferen Analyseansicht des Kindes.

8.3 Analyseansicht für ein Kind

User Stories: 11, 12, 13

Tasks:

- **Auswahl von Symmetriotyp und Zeitraum implementieren**
 - Dropdowns oder Filter-Controls zur Auswahl.
- **Verlaufskurven mit Vergleich „früher vs. aktuell“ darstellen**
 - Zwei übereinandergelegte Linien.
- **Balkendiagramm: Kind vs. Altersdurchschnitt**
 - Vergleich nach durchschnittlicher Zeit und Fehleranzahl.
- **Symmetriotyp-spezifische Balken einfügen**
 - Aufschlüsselung nach H, V, B, R, ? in Balkendiagrammen.

8.4 Kind anlegen (Eingabeansicht)

User Stories: 6, 7

Tasks:

- **Navigation zur Eingabeansicht für neue Kinder**
 - „+ Kind hinzufügen“-Button auf Startseite.
- **Eingabemaske erstellen: Vorname, Nachname, Geburtsdatum**
 - Pflichtfelder mit Validierung.
- **Speicherfunktion für neues Kind implementieren**
 - Speichert Daten in der aktuellen Speicherquelle (DB oder JSON).

8.5 Speicheroptionen: Datenbank vs. JSON

User Story: 14

Tasks:

- **UI-Switch zur Auswahl des Speichermechanismus**
 - Z. B. Toggle oder Dropdown in den App-Einstellungen.
- **Umsetzung der Umschaltlogik**
 - App verwendet dynamisch je nach Auswahl den richtigen Speicher.