

## PoC 101 Dynamische Generierung von Statistikaufgaben

### Beschreibung des Vorhabens:

Statistikaufgaben sollen automatisch und konsistent aus Basis-Datensätzen erzeugt werden.

Der Aufgabengenerator soll auf Basis eines zufälligen Datensatzes:

- Einen zufälligen (aber kompatiblen) Diagrammtypen wählen
- Einen zufälligen (aber kompatiblen) Manipulationstypen wählen
- Zufällige (aber sinnvoll begrenzte) Stärke der Manipulationsparameter generieren
- Zufällige Antwortmöglichkeiten (mit passender richtiger Antwort) generieren

### EXIT:

- Aufgaben werden ohne Fehler erzeugt
- Jede Aufgabe enthält genau eine korrekte Antwort
- Ungültige Kombinationen (von Diagrammtyp/Manipulations-Typ/Datensatz) werden ausgeschlossen
- Jede erzeugte Aufgabe ist vollständig (Diagrammdaten, Manipulationsdaten, Antworten)

### FAIL:

- leere oder fehlerhafte Aufgaben
- fehlerhafte Antwortmöglichkeiten

### Fallbacks:

1. Reduktion der Zufälligkeit
2. Einschränkung auf feste Zuordnung von Manipulationstyp für Diagrammtyp
3. Vorübergehend statische Aufgaben

## PoC 102 Darstellung von Diagrammen

### Beschreibung des Vorhabens:

Verschiedene Diagrammtypen sollen korrekt dargestellt und erzeugt werden können (mit externen Statistik Libs)

- Horizontales Balkendiagramm
- Vertikales Balkendiagramm
- Liniendiagramm
- 

### EXIT:

- Alle Diagrammtypen werden ohne Darstellungsfehler gerendert
- Achsen und Beschriftungen sind korrekt positioniert
- Diagramme reagieren korrekt auf unterschiedliche Datensatzgrößen

### FAIL:

- Diagramme werden nicht oder fehlerhaft dargestellt

### Fallbacks:

1. Nutzung von Canvas-Composables anstatt externer Statistiklibs
2. Reduktion der Diagrammauswahl

## **PoC 103 Generierung von manipulierten Diagrammen**

### **Beschreibung des Vorhabens**

Es soll geprüft werden, ob manipulierte Diagramme automatisch aus unveränderten Basisdatensätzen erzeugt werden können

Das System soll

- ein unverändertes Basisdiagramm als Ausgangspunkt verwenden
- abhängig vom Manipulationstyp und Manipulationsstärkeparameter Diagrammmerkmale dynamisch verändern
- die Manipulation rein über Darstellungslogik umsetzen, nicht durch Veränderung der Rohdaten

### **EXIT:**

- Manipulierte Diagramme werden automatisch aus denselben Rohdaten erzeugt
- Jede Manipulation ist visuell eindeutig erkennbar
- Manipulation kann jederzeit rückgängig gemacht werden

### **FAIL:**

- Manipulation erfordert manuelle Anpassung der Daten
- Manipulation wirkt zufällig oder nicht nachvollziehbar

### **Fallbacks:**

1. Vorab berechnete Manipulationswerte statt vollständig dynamischer Berechnung
2. Einschränkung der Manipulationstypen
3. Statische Beispielmanipulationen

## **PoC 104 Animierte Auflösung von Manipulationen**

### **Beschreibung des Vorhabens:**

Es soll geprüft werden, ob statistische Manipulationen animiert aufgelöst werden können. Dabei soll zwischen manipulierten und korrekten Zustand gewechselt werden, ohne das Diagramm neu zu laden

### **EXIT:**

- Diagrammveränderungen werden sichtbar animiert
- Übergang zwischen Manipuliert und korrekt ist flüssig

### **FAIL:**

- Sprunghafter Wechsel ohne Animation
- Diagramm wird falsch oder gar nicht transformiert

### **Fallbacks:**

1. Animationen einschränken
2. Statischer Wechsel zwischen manipuliert und korrekt

## **PoC 105 Werteachsen Manipulationen**

### **Beschreibung des Vorhabens:**

Es soll geprüft werden, ob numerische Werteachsen gezielt manipuliert werden können(durch verkürzten Startpunkt der Achse(also Startpunkt über 0) oder veränderte Skalierung der Achse)

Die Manipulation soll unabhängig vom Diagrammtyp funktionieren(z.B y-achse bei vertikalBalkenDia/LinienDia, x-achse bei HorizontalBalkenDia)

**EXIT:**

- Werteachsen können gezielt verkürzt werden
- Manipulation ist visuell eindeutig erkennbar

**FAIL:**

- Werteachsen werden falsch oder gar nicht skaliert
- Manipulation ist visuell nicht eindeutig erkennbar

**Fallbacks:**

1. Wertachsen-manipulation nur bei vertikalBalkenDiagramm

## **PoC 106 Kategorieachsen Manipulationen**

**Beschreibung des Vorhabens:**

Es soll geprüft werden, ob Kategorieachsen(bei Balkendiagramm z.b die Anzahl der Balken) manipuliert werden können, indem nur ein Teilbereich der Kategorien angezeigt wird.(Auswahl eines günstigen Zeitfensters oder ausblenden früherer oder späterer Datenpunkte)

**EXIT:**

- Nur ein Teilbereich der Kategorien wird dargestellt
- Achsenbeschriftungen passen zum sichtbaren Bereich
- Übergang zwischen Teilbereich und vollständiger Ansicht ist möglich

**FAIL:**

- Kategoriepunkte werden falsch dargestellt/positioniert)
- Kategorien und Daten stimmen nicht überein
- Falsche oder Fehlende beschriftung

**Fallbacks:**

1. Statische Auswahl eines festen Kategorieausschnitts

## **PoC 107 Stilistische Manipulationen**

**Beschreibung des Vorhabens:**

Es soll geprüft werden, ob Diagramme über rein visuelle Stilmittel (Fehlende Beschriftungen,Style(Farben,Strichstärken, hervorhebungen etc.), Overlaps / Occlusion) manipuliert werden können

**EXIT:**

- Stilistische manipulationen funktionieren korrekt
- Stilistische Anpassungen beeinflussen die visuelle Interpretation der Daten eindeutig
- Daten bleiben unverändert

**FAIL:**

- Stilmanipulation hat keinen erkennbaren Einfluss auf die Wahrnehmung
- Stilmanipulationen funktionieren nicht richtig oder gar nicht

**Fallbacks:**

1. Einschränkung der Stilmittel

## **PoC 108 Antwortlogik und Eindeutigkeit der richtigen Lösung**

**Beschreibung des Vorhabens:**

Es soll geprüft werden, ob für jede automatische generierte Aufgabe:

- genau eine eindeutig korrekte Antwort existiert
- falsche Antworten plausibel aber klar unterscheidbar sind

**EXIT:**

- Jede aufgabe besitzt exakt eine korrekte Option
- Keine Mehrdeutigkeit zwischen Antwortoptionen
- Antworten passend zum Manipulationstyp

**FAIL:**

- Mehrere plausible richtige Antworten
- Richtige Antwort ist zufällig bzw falsch

**Fallbacks:**

1. Vordefinierte Antwortsets pro Manipulation
2. Einschränkung der Antwortvielfalt

## **PoC 109 Reale Anwendungsbeispiele zu Statistik-Manipulationen**

**Beschreibung des Vorhabens:**

Es soll geprüft werden, ob nach Lösungsabgabe passende hochwertige Realbeispiele bereitgestellt werden.

**EXIT:**

- Nach Lösungsabgabe ist der "Where was this used?"-Button sicht- und anklickbar
- Zu jeder Manipulationsart wird mindestens ein passendes reales Beispiel angezeigt
- Das Beispiel ist eindeutig der zuvor behandelten Manipulation zuordbar
- Darstellung ist verständlich
- Parallele zur gelösten Aufgabe wird deutlich

**FAIL:**

- "Where was this used button" liefert unpassende Beispiele
- Es wurden nicht ausreichend Beispiele aus seriösen Quellen gefunden
- Die Beispiele sind irrelevant/uninteressant

**Fallbacks:**

1. Anstatt Realbeispiele werden fiktive Beispiele erstellt
2. Anstatt dem "Where was these Used" Button wird eine Auflistung mit Beispielen zu allen Manipulationsmethoden bereitgestellt

## **POC 110 Gamification**

### **Beschreibung des Vorhabens:**

Es soll geprüft werden, ob die App genügend Spielspaß und Erfolgserlebnisse liefert, zur Gewährleistung der Nutzermotivation.

geplante Gamification Methoden:

- Experience System
- XP Rewards (Titel, Customization)
- Achievements
- Daily Challenges/Streaks
- User Statistik
- Schwierigkeitsgrade

### **EXIT:**

- Mindestens 4/6 der Gamification Methoden wurden implementiert
- Nutzer zeigen wiederholte Sitzungen
- Nutzer bewerten die App als spaßig
- Nutzer berichten von positiven Erfolgserlebnissen

### **FAIL:**

- Ungenügender Einsatz von Gamification methoden (<4/6)
- Nutzer brechen Aufgaben ab oder kehren nicht wieder
- Belohnungssystem wird als langweilig oder unverständlich erlebt

### **Fallbacks:**

1. Anpassung der Belohnungsfrequenz
2. leichter erreichbare Badges statt Achievements
3. Erweiterung mit anderen Gamification Methoden
4. Anpassung der Schwierigkeitsgrade

Alte POCs:

## **PoC 111 Prüfung der Kernfunktionen**

### **Beschreibung des Vorhabens:**

Es soll geprüft werden, ob sich der geplante Gesamtfunktionsumfang der App in modulare Kernfunktionen herunterbrechen lässt, die trotz begrenzter Team- und Zeitkapazität vollständig, stabil und in nutzbarer Qualität implementiert werden können.

Der Poc überprüft, ob:

- Aufgaben anzeigen, Antwortauswahl, Antwortauswertung und Navigation im Kernprinzip ohne Erweiterungen(Animationen/Transformationseffekte, Gamification) stabil laufen.
- Technische Modularität, die eine schrittweise Erweiterung ermöglicht, vorhanden ist

**EXIT:**

- Die Kernfunktionen (Aufgaben anzeigen, Antworten auswählen, Lösungsabgabe, Lösungsausgabe und Navigation) funktionieren zuverlässig
- Der geschätzte Zeitaufwand wird eingehalten(+/- 20%)
- Kernfunktionen können parallel entwickelt werden
- Features sind sauber voneinander getrennt

**FAIL:**

- Deaktivierte Features führen zu App-Crashes oder inkonsistenter UI
- Die Kernfunktionskette funktioniert nicht
- Features sind nicht sauber voneinander getrennt
- Ein Großteil der Kernfunktionen benötigt deutlich mehr Zeit als geplant

**Fallbacks:**

1. Funktionsumfang reduzieren
2. Aufgaben-Generierung vereinfachen

## **PoC 112 Persistenz Spiel- und Lernfortschritt**

### **Beschreibung des Vorhabens:**

Es soll geprüft werden, ob Nutzerfortschritt zuverlässig gespeichert und geladen werden kann (XP, Level, personelle Statistik, Achievements)

**EXIT:**

- Fortschritt bleibt bei App-Neustart erhalten
- Kein Datenverlust

**FAIL:**

- Fortschritt geht verloren

**Fallbacks:**

1. Temporäre Speicherung