# Whitebox-Tests

**Ziel:** Interne Logik und Datenfluss in Funktionen testen  
**Vorgehensweise:**

* Quellcode wird schrittweise analysiert und auf korrektes Verhalten geprüft
* Beispielsweise: getQuestionObj wird für verschiedene Fragetypen auf vollständige DTO-Erzeugung getestet
* Entscheidungen (if/else, Switches) und alle Codepfade werden durchlaufen

# Blackbox-Tests

**Ziel:** Funktionalität aus Sicht des Benutzers testen, ohne Einblick in den Quellcode  
**Vorgehensweise:**

* Datei wird über die Benutzeroberfläche importiert
* Nur das Ergebnis (z. B. XML-Vorschau, Exportinhalt) wird geprüft
* Verhalten bei gültigen und ungültigen Eingaben testen

# Integrationstests

**Ziel:** Zusammenspiel verschiedener Komponenten sicherstellen  
**Vorgehensweise:**

* Kompletter Datenfluss: XML → IndexedDB → Anzeige → Export
* Es wird geprüft, ob Änderungen (z. B. Checkbox-Auswahl) korrekt in der Datenbank und im Export reflektiert werden
* Tests decken Funktionen über Modulgrenzen hinweg ab

# UI-/Interaktionstests

**Ziel:** Richtiges Verhalten der Benutzeroberfläche und Bedienelemente testen  
**Vorgehensweise:**

* Checkboxen selektieren/deselektieren → Auswirkungen im XML-Editor prüfen
* Label-Klick löst Scrollen zum XML-Abschnitt aus
* Auch Designverhalten bei Hover und Auswahl beobachten

# Persistenztests

**Ziel:** Sicherstellen, dass Daten in IndexedDB auch nach einem Seiten-Reload erhalten bleiben  
**Vorgehensweise:**

* Fragen importieren und auswählen
* Seite neu laden → Prüfen, ob Fragen und Auswahlen korrekt wieder angezeigt werden
* Optional: Kontrolle über Developer Tools > Application > IndexedDB

# Export-/Dateigenerierungstests

**Ziel:** Vollständigkeit und Richtigkeit des exportierten XML  
**Vorgehensweise:**

* XML-Datei nach Export öffnen
* Struktur prüfen: Wurzelelement <quiz>, enthaltene <question>-Elemente
* Nur ausgewählte Fragen dürfen im Export erscheinen
* Exportierte Datei in externem XML-Validator prüfen

# Usabilitytests

**Ziel:** Beurteilung der Benutzerfreundlichkeit und Verständlichkeit  
**Vorgehensweise:**

* Probanden verwenden die Anwendung ohne Erklärung
* Beobachtung: Finden sie alle Funktionen intuitiv? Gibt es Stolperstellen?
* Rückmeldung nutzen, um z. B. Labels, Buttons oder die Vorschau zu optimieren

# Robustheits-/Negativtests

**Ziel:** Verhalten bei unerwartetem oder fehlerhaftem Input prüfen  
**Vorgehensweise:**

* Ungültige XML-Dateien importieren (z. B. ohne <question>-Elemente)
* Verhalten bei leerer Datei oder beim Entfernen aller Checkboxen
* Erwartung: Kein Absturz, sondern sichere Fehlerbehandlung oder Benutzerhinweis