

Testkonzept V2.1

Unit Tests : Ziel ist es, das Verhalten elementarer Codeteile zu überprüfen.

Unit Tests sind bereits im Code vorhanden und in fünf Programme gruppiert, welche die drei Funktionalitäten der App sowie die Schnittstelle zu den Audio- und Videoplayern testen.

- Die bestehenden Tests funktionieren immer noch nur teilweise, der erste Schritt besteht darin, sie zum Funktionieren zu bringen.

- Für sechs Funktionalitäten der Anwendung (Tools, Quiz usw.) müssen weitere gleichwertige Testprogramme (bestehend aus Unit Tests) implementiert werden.

Auch für die Datenbankverwaltung müssen Unit-Tests geschrieben werden (siehe nächster Abschnitt)

Datenbank Tests : Ziel ist es, die Gültigkeit der Daten nach verschiedenen Zugriffs- und Änderungsarten zu überprüfen.

Insbesondere soll geprüft werden, ob die verschiedenen Aktivitäten nach ihrer Beendigung korrekt gespeichert werden.

Da die Codebasis keine Datenbanktests enthält, müssen Unit-Tests für das Modul „database_helper.dart.“ geschrieben werden. Dieses verwaltet die Aufzeichnung der Benutzeraktivität.

Integrationstest : Ziel ist es, zu überprüfen, ob die verschiedenen für eine Story verwendeten Komponenten wie vorgesehen zusammenarbeiten.

Wir werden Mocking verwenden, um die verschiedenen Interaktionen zu simulieren.

Installationstest : Die App ist schon und wird weiter auf Google-Play und Apple-Store verfügbar. Der Test besteht lediglich darin, zu überprüfen, ob die in den Stores angebotenen Versionen tatsächlich die sind, die gepusht wurden.

GUI Test : Das Ziel : die Funktionalität und das Erscheinungsbild der App-GUI zu überprüfen.

Wir werden zunächst selbst von Hand testen. Wir werden auf den verschiedenen Geräten, die die Gruppe besitzt, testen und nach Unstimmigkeiten im Erscheinungsbild suchen. Wir werden auch versuchen, das für Android entwickelte Framework „UI Automator“ zu verwenden.

Stress-Test : Wir möchten unsere App zufällig, aber dennoch wiederholbar einem Stresstest unterziehen.

Für diesen Test verwenden wir Monkey auf Android und AutoMonkey auf dem iPhone. Diese Programme laufen auf einem Emulator oder Gerät. Sie generieren pseudozufällige Ströme von Benutzerereignissen wie Klicks, Berührungen oder Gesten sowie eine Reihe von Ereignissen auf Systemebene.

Usability-Test : Ziel ist es, das Benutzererlebnis zu verbessern und das Produkt benutzerfreundlicher zu gestalten.

Unsere Kundin hat uns eine Testgruppe zur Verfügung gestellt. Wir werden sie bitten, verschiedene Aufgaben zu lösen und dazu Feedback einzuholen.