05.10.2023

Testkonzept

 Zusammenfassung (**Introduction**)

Wir erstellen eine Webseite für einen Tennisclub mit Login mit gehashten Passwörtern, Newsseite, Courtbookingsystem etc.

 Big Picture - System Architektur mit den Test Items

Wie werden dies mit dem Python Web-framework «Django» realsisieren.  
Dabei wird eine sqlite Datenbanke benutzt um die User, News und Courtdaten speichern zu können

 Test Features - welche Elemente getestet werden

* Login Funktion
* DB-Funktionen
* Court Booking
* Join Memebers Funktion (neue User)

 Features not to be tested - welche Aspekte nicht getestet werden

* Visuelles

 Testvorgehen - Ihr Projekt wird nach dem TDD umgesetzt

* Wir schreiben als erste Tests, dann zu einem späteren Zeitpunkt kommt das kodieren der Seite.

 Kriterien für erfolgreiche / nicht-erfolgreiche Tests (optional)

 Testumgebung - welche Tools Sie für das Testen verwenden

Framework – Django

IDE – PyCharm

 Kurze Planung

A diagram of a diagram

Description automatically generated

* **GETESTET WIRD:**
  + **BENUTZERERSTELLUNG**
    - **Überprüfung der Gültigkeit des Benutzererstellungsformulars**
    - **Mit gültigen Anmeldeinformationen**
    - **Mit ungültigen Anmeldeinformationen**
  + **BENUTZERANMELDUNG**
    - **Erfolgreiche Anmeldung**
    - **Gescheiterte Anmeldung**
  + **DETAILANSICHT**
    - **Anzeige der Detailinformationen eines Benutzers**
  + **BUCHUNGSSYSTEM**
    - **Buchungserstellung**
      * **Überprüfung der Buchungsinformationen auf Gültigkeit**
      * **Mit gültigen Buchungsinformationen**
      * **Mit ungültigen Buchungsinformationen**
    - **Buchungsbestätigung**
      * **Erfolgreiche Bestätigung einer Buchung**
      * **Fehlschlag der Buchungsbestätigung**
    - **Stornierung einer Buchung**
      * **Überprüfung des Stornierungsprozesses**

# **Fazit zu TDD.**

# **Wir haben TDD ausprobiert und festgestellt, dass es für uns nicht so gut funktioniert. Die Idee, zuerst Tests zu schreiben, bevor man überhaupt etwas programmiert, klingt zwar gut, hat uns aber echt ausgebremst. Es hat viel Zeit gekostet und manchmal haben wir uns zu sehr auf die Tests konzentriert, statt darauf, unsere eigentliche Arbeit zu machen. Immer wenn sich was geändert hat, mussten wir auch die Tests ändern, und das war ziemlich lästig. Wir wollten schnell und flexibel sein, aber mit TDD haben wir uns irgendwie festgefahren gefühlt. Also haben wir entschieden, dass TDD vielleicht nicht für jedes Team oder Projekt passt. Man muss immer gucken, was am besten für die eigene Arbeit ist.**