

Lastenheft

Anforderungen

- Abgabetermin: spätestens 18.06.2025, 23:59 Uhr

Organisatorisch

- 2er Teams
- Arbeit teils im Unterricht, teils zuhause

Technischer Inhalt

Must Haves

POS

- aktive Verwendung von GIT
- Projektplanung per Github
 - Github zum Tracken der Arbeitspakete
 - Kanban Board mit Tickets -> Zuweisen der Arbeit
- Klassendiagramme vor dem Start der Programmierung
- Grafische Anwendung
 - mind. 3 Fenster/Pages oÄ
- Vererbung; abstrakte Klassen
- Interfaces
- API Dokumentation
- Unit Tests
- Logging

DBI

- Mehrere Tabellen (min. 3)
 - mind. 3. Normalform
 - Queries mit Select, Joins, Aggregation
 - Schreibender und lesender Zugriff
- mind. 2 Rollen (admin, user) mit entsprechenden Rechten
- SQL Datenbank (kein SQLite) - MariaDB
 - min. lokale Datenbank
- REST Interface (Backend: Python/C#, Swagger)
- API Dokumentation (Swagger) + JSON Schema für Objekte
- Unit Tests
- Logging

Nice To Haves

POS

- Parallele Programmierung (Threads, Tasks, ...)
- Neues Framework
- Neue Programmiersprache
- usw.

DBI

- Datenbank in der Cloud
- usw.

Unsere Planung

Grobe Idee

Eine HandyApp, in welcher man täglich und wöchentlich neue Quests/Challenges bekommt. Bei diesen Challenges bekommt man ein Bild von einem Ort und muss dann diesen Ort in Real Life finden und ihn besuchen.

Dadurch ist sowohl der Challenge- bzw. Spaßfaktor, als auch der Aktivitätsfaktor gegeben.

technischer Inhalt

Must-Have

- Profile
- 15 Bilder aus Bregenz
- Benachrichtigung wenn neue Challenge da ist
 - VERWORFEN, weil ergibt keinen Sinn, wenn es DailyQuests/WeeklyQuests gibt
- Profil fortschritt
- Belohnung für erfolgreiche Challenge
- GPS Tracking um herauszufinden ob Spieler am richtigen Ort ist

Nice-To-Have

- Shop in welchem man seine Belohnungen ausgeben kann
- Schritte die man läuft werden gezählt
- Daily Streaks
- Bilder aus den größten Städten Vorarlbergs

Wont-Have

- Online Multiplayer

Aufteilung

Arbeitsteilung

Marlon

Ensar

Marlon	Ensar
REST-API	REST-API
Unit Tests	Login-System
Logging	GUI
GPS-Tracking	Shop
Belohnungen	User-Profile
Shop	API Dokumentation
Supabase Setup	Datenbank-Konzept