

Besprechung:

- wir benutzen Java 11
 - ↳ Probleme beim aufsetzen von Spring; Tobias schaut sich das nochmals an
- oreilly: Lernplattform
 - ↳ Spring & Spring Boot Fundamentals
- Mike hat einen großen UML-Entwurf erstellt
- weitere Features, die noch fehlen
 - ↳ Playerliste
 - ↳ Chat
 - ↳ ...

- Datenbank?
- Projekt generieren mit "start.spring.io"
- Aufgabenteilung:

Testen: Jonas

Dokus: Nils

Implementation: Tobias, Daniel, Mike

Glossar: Corinna

- Aufteilung in Front- & Back-End
 - ↳ ~80% Backend
 - ↳ Frontend: Daniel
 - ↳ Backend: Rest
 - ↳ Mike teilt ein

Besprechungsduauer:

~1h

Besprechung mit Server Owner:

- Server owner: Bastian
 - ↳ hat auch bei ArH Projekt gemacht
 - ↳ stellt für „einfache“ Lösungen den Server zur Verfügung
 - ↳ gibt Docker-Container vor (Wir machen Standalone in Docker-Container, abgeben then wir die Standalone)
- Frontend: Webseite
- Backend: Rest (Industrie-Standard)
- Wir verwenden Spring Boot mit wahrscheinlich Tomcat
- Kein PHP

- Versionen lokal testen und erst fertige Versionen auf den Server pushen
- Mail-Versandt sollte kein Problem sein
 - ↳ von innen nach außen kein Problem
 - ↳ von außen nach innen muss eine Portfreigabe gemacht werden
- Front- & Backend strikt voneinander trennen wäre sehr sinnvoll
- UML kann man mit REST (auch in Spring vorhanden) super umsetzen

- Backend definiert, Frontend läuft auf
- Zugriff auf Server durch Web-Oberfläche oder Protokolle wie z.B. FTP
- nur fertige, lauffähige Dateien auf den Server laden
- das Produkt muss am Ende funktionieren, aber das Außenrum ist nichtiger
- SQLite statt MySQL
- Frontend: HTML + JavaScript
- Backend: C# o. Python o. ...

- Serverowner kontaktieren wenn wir wissen, ob Java, C++, ...
- JavaScript Framework: Vue
- Prototyp HTML + JS Frontend
+ Backend wird durch Dummy ersetzt
- Vorschlag: HTML mit Vue JS als Frontend
- Serverowner kontaktieren sobald wir genau wissen was wir brauchen
↳ nur fertige Sachen auf den Server packen

Besprechungsdauer: ~2h

