

EPJ_Tauschboerse

[Core]

- Login
- User mit Liste
 - "required/wished"-Items
 - "possessed"-Items
- 1 - 1 Tausch
- Suche nach "required/wished"-Item

[Optional]

- [high]
 - Transaction History
 - Item entfernen (because of damage, nicht mehr vorhanden)
 - Check auf inaktiver User => wird weniger gewichtet/priorisiert
 - Statistik(Most Need, Most Have, etc...)
 - https with lets encrypt
- [mid]
 - Kommunikation
 - Nachrichtenbox
 - Rate users (trustworthy, ...)
- [low]
 - User Registration
 - Best Match bei Mehrfachsuche
 - n - n Tausch
 - Multiple language support
 - Wertschätzung
 - Random swap, when users subscribe
 - Page to simply add all the cards or whatever item which will be exchanged
 - Simple wrapper, which opens our responsive website in an android app
 - for iOS as well
 - Security
 - Admin Panel
 - Ban users
- [Non Functional Requirements]
 - Accessibility (red green weakness, ...)
- [Limitation]
 - No IE and Edge support

[mögliche Technologien]

- Front End
 - Typescript
 - HTML / Handlebars
 - SASS
 - Font Awesome
 - CSS grid or Bootstrap
- Back End
 - Typescript
 - Node JS??? (Angular??? React???)
 - Express
 - Passport.js
 - Database: Postgres würd ich vorschlagen / (Mongo DB, NoSQL)
 - Grunt or Webpack

[Testing] - noch offen

- Karma for JS??? Just for Angular? Without Angular Mocha? => speedy fragen

[Server] - Server wird von der HSR bereit gestellt

- Für JIRA für Zeitverwaltung
- Node.JS with tomcat (Application Server)

[Projektverwaltung & Dokumentation]

- LaTeX, just if no word document is required
- JIRA [<https://de.atlassian.com/software/jira/core>] - Zeitverwaltung (anstatt RedMine)

[Code Verwaltung] - wird privat gestellt sobald Projektantrag akzeptiert wird

- Private Github repository [https://github.com/Resoulance/EPJ_Tauschboerse]

[Dokumentation] - use cases, screens, UML

- use cases [https://github.com/Resoulance/EPJ_Tauschboerse/blob/master/doc/usecases.md]
- screens [https://github.com/Resoulance/EPJ_Tauschboerse/blob/master/doc/screens.md]
- UML

[Helpful links]

- <https://www.ctl.io/developers/blog/post/build-user-authentication-with-node-js-express-passport-and-mongodb>