

Treffen-05

Datum / Uhrzeit: 07.01.2014 von 8 Uhr bis 10 Uhr

Teilnehmer:

- Christoph Schultz (anwesend)
 - Elvira Akhtyamova (abwesend, Krankheit)
 - Felix Helfer (anwesend)
 - Lukas Eipert (anwesend)
 - Nick Brückner (anwesend)
 - Nico Graebing (anwesend)
 - Siegfried Zoetzsche (anwesend)
-

[0. TODO](#)

[1. Organisatorisches](#)

[2. Inhaltliches](#)

0. TODO

1. Umsetzbare SPARQL-Anfragen herausarbeiten [Christoph, Nico]
 2. Qualitätssicherung
 - 2.1. Dok-Konzept: [Siggi]
 - 2.2. Test-Konzept: [Felix]
 - 2.3. Organisatorische: [Felix]
 3. Vorprojekt Puzzlekonzept erstellen: [Lukas]
 4. Projektangebot
 - 4.1. Zielbestimmung: Ausformulierte Aufgabenstellung [Felix]
 - 4.2. Voraussetzungen: [Nick]
 - 4.3. Designübersicht und Funktionalität: [Christoph, Nico, Elvira]
 - 4.3.1 Nutzer-Use Cases
 - 4.3.1.1. Administrator:
 - 4.3.1.1.1. Beispiele Speichern
 - 4.3.1.1.2. Übersetzen
 - 4.3.1.1.3. Konfigurieren
 - 4.3.2 SPARQL-Use Cases
 - 4.4. Meilensteine [Lukas]
 - 4.4.1. Anfragen Speichern
 - 4.5. Qualitätssicherung [Felix]
 - 4.6. Glossar (siehe Blatt 2)
-

1. Organisatorisches

- Verlegung Termin auf Donnerstag? -> erstmal sehen...
 - Absprache Einführung git
 - Technische Frameworks
 - Wann ist der erste Meilenstein fällig?
 - "23.-30.1. in Absprache mit dem Betreuer (O)" laut Ablauf im olat
 - wer ist verantwortlich für das Vorprojekt und das Projektangebot
 - Beim Qualitätssicherungskonzept würde ich auf unseren Verantwortlichen für Qualitätssicherung tippen -- je nach Abschnitt den jeweils verantwortlichen reinpacken
 - Bis nächstes Mal: Festlegen von Javascript Kursen & Termin desselbigen [Lukas]
-

2. Inhaltliches

2.1 Aspekte (Blatt 2)

- Endpunktgebundenheit: schon eher dynamisch, Einstellungen über Konfig-Datei würde zunächst reichen
- [Virtuoso](#) SPARQL Endpunkt
- Benutzer/Intuitivität:
 - DAU mit wenigen Kenntnissen → große Intuitivität
 -
 - RDF/SPARQL möglichst versteckt; keine graf. Darstellung von Triplen
 - "geführtes Lernen" Dokumentation:
 - Stufe 1: technisch
 - Stufe 2: Anwendersicht
 - optional: Screencast (Youtube)
- Sprachen: HTML/JavaScript
- Anforderungen: fix soll es sein
- Vorprojekt:
- Browserkompatibilität: IE auf Minimum anpassen
- UTF8 !
- Frameworkwahl begründen (Angular JS wird für gut befunden)
- Modularität wäre super, aber Sekundärziel
- primär Englisch, sprachlich aber besser anpassbar

2.2. Vorprojekt

- MockUp-Projekt mit Designvorschlägen
- Wie können welche SPARQL-Elemente umgesetzt werden. Graphisches Design
- Scratch Programmierumgebung

2.2 Projektangebot

1. Zielbestimmung:
 - Ausformulierte Aufgabenstellung
2. Voraussetzungen:

- SPARQL-Use Cases
- 3. Designübersicht und Funktionalität:
 - Puzzlekonzept erstellen
- 4. Arbeitspakete und Meilensteine
- 5. Qualitätssicherung
- 6. Glossar (siehe Blatt 2)

2.3 Qualitätsicherung

Dokumentationskonzept -> Siggi

1. Bezeichner
 - a. Englisch
 - b. Camel Case: z.B. calculateMaxAndMin
 - c. Klassennamen: Beginnen mit Großbuchstaben
 - d. Öffnende Klammern selbe Zeile zB. function hans (a) { ...
 - e. if {
 - ...
 - } else {
 - ...
 - }
 - f. Konstanten: ALL CAPS mit Unterstrichen KON_STANZE
 - g. Ordentlich einrücken mit zwei Leerzeichen "Tabs"
 - h. Funktionen und Variablen beginnen klein
 - i. Variablen zu Beginn einer Funktion deklarieren var a,b,c,d,e...
 - j. Sprechende Variablennamen
 - k. Dokumentation [AngularJS]
 - <http://www.lsauer.com/2013/05/javascript-documentation-generator.html>
 - l. Codeüberprüfung durch js-lint [AngularJS]
- Entwurfsdokumentation
- Coding Standards
- Funktionslänge beschränken max 15 Zeilen
- Klassenlänge maximal 120 Zeilen
- Administrativ / Anwender
- Erwähnen das wir LaTeX-awesome sind

Testkonzept

- Testumgebung für AngularJS
- Komponententest
- Integritätstest
- GUI-Test -> DAUTest
- Systemtest
- Abnahmetest
- Liste von möglichen JS-Test-Frameworks: http://en.wikipedia.org/wiki/List_of_unit_testing_frameworks#JavaScript

BUCH: <https://www.dropbox.com/s/a4ffgm044hobjlu/semantic-web-buch.pdf>