Wissenslandkarte

- Sitzungsprotokoll: Breakout Meeting "Frontend-Coding", 18.05.2021
- Sitzungsprotokoll: Kickoff Meeting Mo, 08.02.2021
- Sitzungsprotokoll: Milestone & Feature Planning Meeting, 23.02.2021
- Sitzungsprotokoll: Technologie- & Framework-Meeting, 04.05.2021
- Sitzungsprotokoll: Technologie- & Framework-Meeting, 14.04.2021
- Sitzungsprotokoll: Technologie- & Framework-Meeting, 31.03.2021
- Sitzungsprotokoll: Technologie- & Workflow-Meeting, 03.03.2021
- Übersicht

Sitzungsprotokoll: Breakout Meeting "Frontend-Coding", 18.05.2021

Übersicht über die verwendeten Frameworks

- Django: Serverside-Web-Framework – bindet Datenbanken und Logik an, stellt Informationen zum Rendern eines HTMLs oder für eine REST-API bereit, inkl. Nutzerlogin, Rechteverwaltung
- Jinja2: Serverside-Template-Engine – wird zur Darstellung von Informationen als HTML genutzt
- Vue: Client-Template-Engine – wird zur Darstellung von per REST abgerufenen oder in Javascript berechneten informationen als HTML-DOM genutzt.
- Bulma: CSS-Stylesheet & UI-Framework – stellt CSS-Klassen bereit, um einem HTML-Dokument ein bestimmtes nutzerfreundliches Aussehen zu verleihen.
- Fontawesome: CSS-Stylesheet – stellt CSS-Klassen bereit, um Symbole anzuzeigen
- Gunicorn: Kümmert sich darum, Python-Prozesse mit Django zu starten, und Anfragen von einem Webclient zu parsen
- Nginx: Webserver – liefert statische Dateien aus, die ihm Django hingelegt hat.

Diagramm (Mischung statische Referenzierung und dynamisches Verhalten)

g24.png

Nächstes reguläres Treffen

31.05.2021 18:00

Sitzungsprotokoll: Kickoff Meeting Mo, 08.02.2021

Wissenslandkarte für Makerspace Erlangen

> “
> Webentwicklungsprojekt: Kontaktprofile mit skalaren Schlagworten, aka. Wissenslandkarte

Überblick siehe auch unter: https://wiki.betreiberverein.de/books/projekte-und-ideen/page/wissenslandkarte

URL für virtuelles Meeting: https://fau.zoom.us/j/97276218341?pwd=ZmhhTU5iaTJoK2NQQ1VCNUVRcHZ3Zz09

Agenda:

1. Team Building / Kennenlernen
2. Organisatorisches
   - nächstes Meeting
   - Kommunikationsmedium
3. Beteiligte (Stakeholder)
4. Projektziele
5. Anforderungen (Requirements)

6. Brainstorming: Geeignete Plattformen
7. Sonstiges

Nächstes Meeting

Di, 23.02. 18:00

Kommunikationsmedium

Julian richtet einen Mailverteiler ein: wissenslandkarte@betreiberverein.de

https://lists.erlangen.ccc.de/mailman/listinfo/wissenslandkarte
Christian verwaltet ihn.

Beteiligte

- Anwender
  - auch technisch unversierte!
  - potentielle Anwender auf verwandten Webdiensten (Forum, etc.)
  - auch komplett offline (Ideen gesucht.)
- Öffentliche Leser / "Internet"
- Betreiberverein/Makerspace
  - Ansprechpartner / Expertise finden
  - Vernetzung von Mitglieder:innen
  - Basis für Projekte schaffen
- Vorstand
  - Juristische Vertretung (Impressum, Datenschutzerklärung)
  - Auftraggeber
- Bits'n'Bugs
  - Administration / Bereitstellung von Infrastruktur
- ... auf der Suche nach Expertise/Mitstreiter
  - Gruppen/Vereine
  - Privatpersonen
  - Startups?
- Entwicklungsteam
  - Projektleitung / Release-Management
  - Entwickler
  - Design
  - Test & UX
- Moderation

- Open-Source-Nutzer
- städtische Projekte
  - "Post Corona"

Projektziele

- Experten finden und in Kontakt treten
- Vernetzung verschiedener Fach- & Interessensrichtungen begünstigen
- Dokumentation von Projekten (Verein, Einzelpersonen, auch Externe)
  - Verlinkung von aus der Wissenskarte entstandenen Projekten / Veranstaltungen / etc.
- Abbildung
  - Informale Expertise und Interessen
  - Formale Expertise / Ausbildung / Kompetenzen / Einweisungen dokumentieren
  - Themenmoderation / Themencluster (Redundante / Inkonsistente / Ungepflegte Daten vermeiden)
  - Bereitschaft zu Kontakt und Mitarbeit
- Gamification / Badges
- Nutzeranforderungen an Werkstätten aus Vereinssicht messen (kurzfristig)
  - Kontext: Umfrage unter allen Beteiligten für Mitwirkung bei Planung / Betrieb / künftigen Projekten auf Website geplant, um Ausstattungsbedarf abzufragen -> kann als Basisdatensatz dienen
- Expertise sichtbar machen; Werbewirksamkeit gegenüber ...
  - Öffentlichkeit
  - Stadt
  - andere Makerspaces
  - Projektträge/Geldgeber PCS
- Veranstaltungskalender
- Dokumentation des Makerspaces
  - Verfügbare Tools (Hardware, Maschinen, Software) darstellen -> Dokumentation der Werkstatt & vorhandenen Ausstattung
  - Verfügbarkeit und Ausstattung von Räume (Besprechungsräume, Ateliers)
- Integration (von Nutzern?) anderer Werkstätten und Einrichtungen in der Wissenslandkarte
- Nutzer überraschen; andere Menschen im Vorübergehen besser kennen lernen
  - Hierzu Integration in andere Plattformen
- Moderation & Pflege
  - Menschliches Vermittlungsgremium für Expertise
  - Menschliche Eintragung in das Tool
  - Veraltete Daten vermeiden -> Archivierung?
- Vertrauensvoll und Sicher
  - Steuerung der Sichtrechte
    - Unterscheidung zwischen Benutzergruppen "intern"/"extern"

Weitere Punkte

- Nächstes Mal:
  - Priorisierung der Projektziele (Milestoneplanung)
  - Anforderungen
- Was war mit "Plattformen" gemeint?
  - Bspw. Friendica & ähnliche bestehende Software, die man nutzen könnte.
  - -> später; vorab erst einmal nur Ideensammlung & Einzelfeatures zum Kopieren

Andere Projekte mit ähnlicher Zielsetzung

- https://fablabs.io
- https://collaborates.io

Sitzungsprotokoll: Milestone & Feature Planning Meeting, 23.02.2021

Vorgesehene Dauer: 60min

Agenda:

1. Mindmap mit Gruppierung von Features
2. Stories erfassen, über deren Umsetzung wir gezielt entscheiden können.
3. Nächsten Milestone festlegen -- was ist unser MVP?
4. Wann ist unser nächster Termin?
5. Wir müssen Anforderungen an Technologien festlegen -- was kann unser Team leisten, was wollen wir nicht tun?
6. Wir müssen Technologien / Plattformen evaluieren -- wer tut was?
7. Dieses Pad wird voll. Entwicklungsplattform (Github? Gitlab? Shared Folder? Board?) festlegen?

Mindmap

Wissenslandkarte.png

Siehe: https://wiki.betreiberverein.de/attachments/2

Kurzzusammenfassung letztes Mal / Stories nach Themenbereich:

Abgleich mit Mindmap -- Features ohne Stories:

- Vernetzungsfeatures

- Interesse generieren, Öffentlichkeitsarbeit & PR
- Ressourcenverfügbarkeit
- Gruppenvernetzung, Gruppenkommunikation

1. Kernfunktion Website
    1. Als neue Nutzer möchten wir eine Website haben, um unsere Daten eintragen und sehen zu können.
    2. Beim Start neuer Projekte möchten wir auf einer Website nach Mitmachbereitschaft & Interesse suchen können, um Teilnehmer zu finden.
    3. Beim Start neuer Projekte möchten wir auf einer Website nach Mitmachbereitschaft & Expertise suchen können, um Hilfe zu finden.
    4. Wir möchten, dass Nutzer formale Qualifikationen selbst eintragen können, damit andere Nutzer danach suchen zu können.
    5. Beim Betrieb des Vereins möchten wir nach Interesse suchen können, um den Bedarf für ständige Bereiche erkennen zu können.
    6. Als Betreiber möchten wir Impressum und Datenschutzerklärung haben, um gesetzeskonform zu sein.
    7. Nach dem Finden interessanter Personen möchte man diese kontaktieren können, um gemeinsame Projekte zu starten.
    8. Als Nutzer möchte ich ein Freitextfeld ausfüllen können, um mich in meinem Profil selbst vorstellen zu können.
    9. Nutzer sollen ein Profilbild hochladen können, um die Profile ansprechender zu gestalten.
    10. Die Website soll ein interessantes und ausgewogenes Design haben, um Interesse zu wecken und die Bedienung einfach zu gestalten.
    11. Bei Aufruf mit einem Handy soll die Website auch mit dem Finger bedienbar sein.
    12. Bei Suche nach einem Thema werden ähnliche Themen ebenfalls angezeigt, um eine Zersplitterung der Sucherergebnisse bei ähnlichen Begriffen (Holz/Schreinerei, Programmieren/Software, etc.) zu vermeiden. (bspw. via Wikipedia-Verlinkungen, Synonymwörterbücher, Google-Suchvorschläge, Semantic Similarity Anhand des Vereinswikis, Manuelles Clustering/Linking)
    13. Bei falschen Einträgen können Administratoren die Schreibweisen vereinheitlichen, um die Qualität der Plattform sicherzustellen.
    14. Sehr alte Einträge erscheinen weniger auffällig, um möglichst Personen zu finden, die auch noch im Umfeld des Vereins aktiv sind.
2. Benutzeridentifikation
    1. Als Betreiber möchten wir Benutzern den Schutz der eigenen Angaben mit einem Login ermöglichen, um fremde Verfälschung und Löschung bestehender (ggf. verlinkter) Profile zu verhindern.
    2. Bei beim Verein eingehenden Löschanfragen möchten wir Benutzerdaten vollständig löschen können, um gesetzeskonform zu sein.
    3. Beim Anmelden möchte ich die gleichen Loginmethode nutzen, den ich auch für andere Plattformen (Forum, Wiki) des Betreibervereins verwende, um weniger Einträge im Passwortmanager verwalten zu müssen.
    4. Beim Anmelden möchte man eine Loginmethode nutzen, die einfach zu bedienen ist, um möglichst inklusiv zu sein.

    5. Als Betreiber möchten wir keine Passwörter erfassen, um unsere Nutzer keinem Risiko auszusetzen.
3. Privacy- und Moderations-Features (Sichtschutz, Nutzergruppen, Autorisierung)
    1. Website ist nicht öffentlich einsehbar, um Bedenken auszuräumen
    2. Nutzer können beim Eintragen entscheiden, welche Einzelpersonen oder Nutzerkreise bestimmte Attribute (Fähigkeiten, Interessen, Kontaktdaten) sehen können
    3. Vereinsfunktionäre können eintragen, welche Nutzer Mitglieder sind, um eine Trennung nach Nutzerkreisen zu ermöglichen.
    4. Betreuer können eintragen, welche Nutzer Mitglied eines ständigen Teams sind, um im Team über die übrigen Mitglieder hinaus weitere zusätzliche Kontaktdaten (Telefonnummer) teilen zu können
    5. Vereinsmitglieder können andere interessierte Nutzer einer Nutzergruppe hinzufügen (Freunde von Freunden/Mitgliedern), damit Nutzer ihre Daten mit den Mitgliedern und anderen aktiven Interessierten teilen können.
    6. Nutzer mit versteckten Kontaktdaten können Kontaktanfragen (Mail per Kontaktformular, DM, o.Ä.) erhalten, um trotz Privatsphäre erreichbar zu sein.
    7. Nach einer Schulung kann durch eine je Qualifikation festgelegte Nutzergruppe eine formale Qualifikation bei Nutzern eingetragen werden
    8. Vereinsfunktionäre oder Nutzer können Nutzergruppen anlegen, um die Vergabe von formalen Qualifikationen vergeben, kommunizieren, oder sich als Gruppe darstellen zu können
    9. Verwalter von Nutzergruppen können die Mitglieder einer Nutzergruppe anzeigen oder auch entfernen, um veraltete Daten zu vermeide
    10. Wer eine Nutzergruppe auf bestimmte Daten freischalten kann, kann auch sehen, wie viele Personen Mitglied sind, welche Personen Mitglieder sind, und wer die Nutzergruppe verwaltet.
    11. Nutzer können verhindern, ggü. Nicht-Gruppen-Admins als Mitglied einer Nutzergruppe aufgelistet zu werden, um ihre Privatsphäre zu wahren.
    12. Nutzer können verhindern, dass ihr Profil unprivilegierten Nutzern angezeigt wird, um lediglich formale Qualifikationen ohne ein weitergehendes Profil nachweisen zu können.
    13. Nutzer können ihre Daten selbst einfach löschen, od. (bei berechtigtem Verarbeitungsinteresse ggü. Mitgliedern) ggf. unbegründeten Zugriff schützen.
    14. Gruppen können als öffentliche Darstellung genutzt werden, um Personenkreise kontaktieren zu können
    15. Gruppen können untereinander kommunizieren, um die Vernetzung zu fördern
4. Website-Features (Projektvorstellung, Veranstaltungskalender) & Anbindungs-Features
    1. Als Entwickler möchten wir eine API haben, mit der wir weitere Plattformen anbinden können.
    2. Die API wird um Sicherheitsmechanismen erweitert (CORS, OAuth), um auch in anderen Diensten den Sichtschutz entsprechend den Benutzerberechtigungen durchzusetzen, und dennoch die benutzerspezifisch freigegebenen Informationen anzuzeigen.
    3. Projektvorstellung evtl. besser aufgehoben in Wiki oder Forum, ggf. aber anzubinden (Verlinkung oder Anzeige von Backlinks)

    4. Makerspacevorstellung evtl. besser aufgehoben in Website oder Wiki, ggf. aber anzubinden (Räume für die Umsetzung von Expertisen)
    5. Veranstaltungskalender evtl. besser aufgehoben auf Website; ggf. aber anzubinden (Formale Expertise)
    6. Anzeige von Informationen aus der Wissenslandkarte in Wiki, Forum, o.Ä. -- je nachdem, was kommt.
5. Menschen-Papier-Display-Features (Visualisierung vor Ort, Beteiligung ohne Webbrowser, Menschliche Vermittlung)
    1. Anzeige der Anwesenden zuständigen formal qualifizierten oder erfahrenen Personen
6. Gamification

Milestone -- was ist unser konkretes Ziel für die erste Version?

1. Website ist nicht öffentlich einsehbar, um Bedenken auszuräumen
2. Als neue Nutzer möchten wir eine Website haben, um unsere Daten eintragen und sehen zu können.
3. Beim Start neuer Projekte möchten wir auf einer Website nach Mitmachbereitschaft & Interesse suchen können, um Teilnehmer zu finden.
4. Beim Start neuer Projekte möchten wir auf einer Website nach Mitmachbereitschaft & Expertise suchen können, um Hilfe zu finden.
5. Wir möchten, dass Nutzer formale Qualifikationen selbst eintragen können, damit andere Nutzer danach suchen zu können.
6. Als Betreiber möchten wir Impressum und Datenschutzerklärung haben, um gesetzeskonform zu sein.
7. Nach dem Finden interessanter Personen möchte man diese kontaktieren können, um gemeinsame Projekte zu starten.
8. Als Nutzer möchte ich ein Freitextfeld ausfüllen können, um mich in meinem Profil selbst vorstellen zu können.
9. Nutzer sollen ein Profilbild hochladen können, um die Profile ansprechender zu gestalten.

Nächstes Treffen

2021-03-03 18:00 -- 19:00

Bis dahin:

Alle überlegen sich, welche Technologien man nutzen möchte:

https://terminklick.stuve.fau.de/poll/nPPKg3Uahz/

Sitzungsprotokoll: Technologie- & Framework-Meeting, 04.05.2021

JS Frameworks

JS-Framework: Vue.js
GUI-Framework: Bulma

Arbeitspakete

- Statische HTML-Ansicht Seite mit Bulma je für
  - Suchergebnis
  - Profil
  - Login Bildschirm
- Vue Texteingabefeld mit Autocomplete-Dropbox (als Suchbox)
- optional Vue Label-Eingabe-Feld mit Autocomplete-Dropbox (als Profilbearbeitungsbox)
  - https://github.com/crabvk/bulma-tagsfield
  - https://wikiki.github.io/form/tagsinput/
- -> Ergebnisse in neues Gihub-Repo, ggf. mehrere Alternativen
- PoCs an Phillip, baut das dann in Django ein
- XSS in Vue
  - Do:
    - https://vuejs.org/v2/guide/security.html#Rule-No-1-Never-Use-Non-trusted-Templates
  - Don’t use:
    - v-html
    - innerHtml
    - domPropsInnerHTML
    - v-bind:style=
    - v-bind:href=

https://codesandbox.io/s/upbeat-maxwell-7dmp7?file=/index.html:593-643

Autocomplete

- Phillip hätte Interesse an den Schlagwortlisten, um zu prüfen, ob man daraus ein funktionierendes und ausreichend schnelles Autocomplete-Modell aufbauen kann. -> mail [at] phi1010.com oder hier in's Pad.
- Konzeptideen https://erlangen.ccc.de/markdown/Ei_DnQnLS3ywjywvE3frjw

Lizenz

- AGPL / GPL / o.Ä. -> nächstes Mal

Breakout Meeting: Frontend-Coding

18.05.2021 18:00

Nächstes reguläres Treffen

31.05.2021 18:00

Sitzungsprotokoll: Technologie- & Framework-Meeting, 14.04.2021

Chris, Patrick und Julian haben ihre Schlagworte und Beschreibungen vorgestellt. Daraus entstanden folgende Einsichten:

- Pinwand mit Suchen/Finden und allgemeine Aushänge
  - Darstellung vor-ort auf großem Monitor
  - Langfristige und kurzfriste Gesuche
    Gesuche können sich auf beliebige Resourcen beziehen, z.B. Fähigkeiten, Geld, Raum, Wissen,...
- Hilfestellung für (numerische) Bewertung von Fähigkeiten geben, z.B. wie bei Sprachlevel
  - Nicht zu feingranular -> 0-6
    - 0 - weiß zwar was es ist, aber nichts gemacht und auch kein tieferes Wissen
    - 1 - Anwender:in, hat schon mal was unter/nach Anleitung gemacht
    - 2 - schon eigenständig angewendet
    - 3 - Erfahrung gesammelt, gibt aber noch einiges zu lernen
    - 4 - Fortgeschritten, kann anleiten und theoretisches hintergrund Wissen vermitteln
    - 5 - Absolute:r Experte:in, es wird schwer sein wen zu finden der sich besser auskennt
  - Darstellung über Emojis wie Schraubendreher, Pinsel usw. evt. in Abhängigkeit zum Schlagwort
- Möglichkeit eine Freitexterkärung zu Fähigkeiten
- Denkbares eingabe verfahren Freitext mit Begriffen -> Liste vorgeschlagener Schlagworte, die dann wiederum manuelle Nachbearbeitet werden können. Neue Schlagworte können angelegt werden
- Schlagworte sollten vermutlich auch in einem Graph strukturiert werden

Nächstes Treffen Dienstag 4. Mai, 18-19 Uhr.

https://fau.zoom.us/j/92803652912?pwd=VEp6NFAzbXhrRkpCMmNxbXFqK1J3dz09

Sitzungsprotokoll: Technologie- & Framework-Meeting, 31.03.2021

TOP

1. Nächstes Treffen
   - Mittwoch, 14.04.
   - https://fau.zoom.us/j/92803652912?pwd=VEp6NFAzbXhrRkpCMmNxbXFqK1J3dz09
2. Umgang mit Technologien
3. Eigenen Demodatensatz
   - Jeder überlegt sich Text (bspw. Sätze, Schlagworte, Emojis, oder andere Formate) mit denen man sich selbst beschreiben würde
   - Jeder überlegt sich Fragestellungen, mit denen man nach solchen Daten fragen könnte.
   - Jeder überlegt sich die Schlagwörter, die man sich selbst geben würde
   - Jeder überlegt sich die Schlagwörter, nach denen man suchen würde
   - -> Jeder für sich in einer Textdatei o.Ä., dann nächstes Mal besprechen.
   - Wenn wir die Fragen und das Format haben, dann auch weitere Personen fragen.
4. Webdesign
   - Kris, Matthias, Patrick: In 2 Wochen ein CSS-Framework und JS-Framework auswählen

JS-Framework

- Angular
  - Für Bewerbungen evtl. interessant zu können, weil viele Stellenausschreibungen
  - Stability https://angular.io/guide/releases
    - Mehrmals im Jahr Breaking Changes durch Major Updates
  - Nur für Web-Applikationen da, keine Anleitungen dazu, wie man einzelne Angular-Komponenten in Nicht-Angular-Websites einbauen kann?
  - Sehr Typescript-lastig. Ist das ein Vorteil? Ist das ein Hindernis?
- React

    - Ressourcen
    - Tutorial: https://reactjs.org/tutorial/tutorial.html
    - Architekturdiagramm: Keins?
    - Requirements:
        - Build mit npm auf Entwicklerrechner
        - Building for Relative Paths / By default, Create React App produces a build assuming your app is hosted at the server root.
        - React-Komponenten in Statischer MPA einbauen: https://reactjs.org/docs/add-react-to-a-website.html
    - Beispiel mit Python: https://www.digitalocean.com/community/tutorials/build-a-to-do-application-using-django-and-react
    - Stability https://reactjs.org/docs/design-principles.html#stability
        - Größere Breaking Changes zuletzt 2017
        - Letztes Major Release 2020; lt Facebook aber fast nicht relevant ("In practice, we've only had to change fewer than twenty components out of 100,000+")
- Vue
    - https://vuejs.org/v2/guide/
    - Stability:
        - 2.0 veröffentlicht in 2016
        - 3.0 veröffentlicht in 2020
    - Requirements
        - Funktioniert auch ohne CLI-Tool, und ohne npm, wenn man sich den node-Stack nicht antun möchte – so auch die offizielle Empfehlung für Anfänger:
            - https://vuejs.org/v2/guide/#Getting-Started
            - https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world
    - Ein Vue-Objekt wird an einen Container gebunden, dadurch vermutlich auch hier wie bei React mehrere unabhg. Komponenten auf der gleichen Website unproblematisch
        - https://codesandbox.io/s/stupefied-tdd-sivrd?file=/index.html

Vergleiche

- https://towardsdatascience.com/vue-vs-react-vs-angular-114718eac3c5
    - Vue einfacher
        - Würde Phillip nach kurzem Ausprobieren bestätigen – minimaler Overhead, um es zum laufen zu bekommen:
            - https://codesandbox.io/s/upbeat-maxwell-7dmp7?file=/index.html:593-643
        - Geht mit anderen Sprachen auch einigermaßen, wenn man das nicht auf der offiziellen Homepage im Tutorial sucht:
            - https://dev.to/luispa/lets-try-react-without-nodejs-3a7

        - https://github.com/antsmartian/hello-world-angular-2/blob/master/index.html
    - Angular umfangreich, aber auch viel Schreibarbeit und anspruchsvoll. Fokus auf ganzen Anwendungen
- https://vuejs.org/v2/guide/comparison.html
    - React erschwert Anpassungen an HTML, da alle Templates in JS; aber bessere Toolunterstützung dafür vorhanden
    - Vue nutzt auch HTML-Templates, dadurch Arbeit für Designer einfacher.
    - Angular hat Restriktionen zur Strukturierung von Anwendungen

Abseits:

- Multi-Page-Applikation mit statischem Rendering der eigentlichen Inhalte (bspw. Benutzerprofile), um evtl. Probleme mit Suchindizierbarkeit und Zugänglichkeit zu vermeiden?
- JS für einzelne Komponenten der Seiten nutzen?
    - Suchbox
    - Suchergebnisanzeige mit Filterfunktionen oder Suchbegriffs-Eingabefeld
    - Profileditor

Design

- https://getbootstrap.com/
- https://materializecss.com/
- https://bulma.io/
- https://semantic-ui.com/
- https://purecss.io/grids/

Sitzungsprotokoll: Technologie- & Workflow-Meeting, 03.03.2021

Tagesordnung

1. Tagesordnung
2. Termine
3. Nutzung des HedgeDoc Pads
4. Web-Technologien
5. Datenrepräsentation (Hierarchie / Suchfunktion)
6. verfügbare Plattformen

Termine

Nächstes Meeting: KW11, Mittwoch, 2021-03-17, 18:00

https://fau.zoom.us/j/92803652912?pwd=VEp6NFAzbXhrRkpCMmNxbXFqK1J3dz09

Dieses Pad so weiterführen?

Github-Issuetracker (Anmerkung: Vortrag Julian am RRZE zu Git-Plattformen als "HPC Cafe") -> TODO phi

https://hpc.fau.de/systems-services/support/hpc-cafe/

Protokolle in die Wissenhalde (wiki.betreiberverein.de) ablegen.

Bitte registrieren: https://github.com/login
GH Accounts: cod3monk, phi1010,

Wer kann/will mit welchen Technologien arbeiten?

https://terminklick.stuve.fau.de/poll/nPPKg3Uahz/

-> Python Tendentiell: Django (wg. inkrementellen Upgrades bei Django-Datenmodell) mit Unit-Test-Framework via Github Actions

IDE: diverse (Pycharm, VS Code)

Frontend: Entscheidung offen lassen; Person suchen, dann entscheiden.

Datenrepräsentation (Hierarchie / Suchfunktion)

-> Bibliothekare einladen

bspw. Eingabeprotokollierung Suchanfragen / Eintragung zwecks Verbesserung der Datenqualität

Wollen wir bestehende Plattformen erweitern?

Bspw. Friendica

-> Klären, welche Kommunikationsplattform der Verein haben will.

Welche Personengruppen schauen sich welche Themenbereiche genauer an, und stellen nächstes

Mal Ergebnisse vor?

phi: Github / Django / Docker Julian: Frontendler und Bibliothekar:innen; evtl. Fragen zu Django Christian: Protkolle ins Wiki

Was brauche ich für Entwicklung

- Linux
  - Installation: ggf. VM, od. WSL(2) auf Windows
  - Knowhow: Bash, Paketmanager der Wahl, Shebang
- Git
  - Installation: Betriebssystem-Paketmanager
  - Knowhow: Clone, Add, Commit, Push, Pull, Merge
    - https://ohmygit.org/
    - https://www.katacoda.com/courses/git
- Python3.8
  - Installation: Betriebssystem-Paketmanager
    - evtl. alternativ https://pipenv.pypa.io/en/latest/advanced/#automatic-python-installation
  - Knowhow: Syntax.
- Pip
  - Installation: Betriebssystempaketmanager
  - Knowhow: Wie finde ich Paketnamen und verlinkte Dokumentation
    - https://pypi.org/project/Django/
- Pipenv
  - Installation: `pip install pipenv`, oder aus Betriebssystempaketmanager
  - Knowhow: Befehle install/run/shell
    - https://pipenv.pypa.io/en/latest/#other-commands
    - https://pipenv.pypa.io/en/latest/basics/#example-pipenv-workflow
- Virtualenv / Venv
  - Installation: Automatisch mit Pipenv
  - Knowhow: Unterschied Virtualenv -- System-Python
    - https://docs.python.org/3/tutorial/venv.html#introduction
- Docker

    - Installation: Betriebssystempaketmanager
    - Knowhow:
        - Ausführen: docker build, docker run
        - Debuggen / Stoppen: docker exec, docker ps, docker kill
        - Aufräumen: docker container prune, docker image prune
        - Konfiguration: Dockerfile Syntax
        - https://docs.docker.com/get-started/overview/
- Docker-Compose:
    - Installation: Automatisch mit Docker
    - Knowhow:
        - Ausführen: docker-compose build, docker-compose up
        - Konfigurieren: docker-compose.yml-Syntax
        - https://docs.docker.com/compose/gettingstarted/
- Django:
    - Installation: `pipenv install` in dem Verzeichnis ausführen, in dem die Pipfile liegt
    - Knowhow:
        - Datenbank initialisieren: manage.py migrate
            - https://docs.djangoproject.com/en/3.1/ref/django-admin/#migrate
        - Admininterface: manage.py createsuperuser
            - https://docs.djangoproject.com/en/3.1/ref/django-admin/#createsuperuser
        - Ausführen: manage.py runserver
            - https://docs.djangoproject.com/en/3.1/ref/django-admin/#runserver

Übersicht

Note Pad:

https://erlangen.ccc.de/markdown/F8JieVS8ShmJY3c1U8sAlQ

Kommunikation:

Unsere Kommunikation findet über die Mailingliste wissenslandkarte@betreiberverein.de statt, administriert von Christian Borss. Siehe

https://lists.erlangen.ccc.de/mailman/listinfo/wissenslandkarte.

Skizzen und Dokumente:

Wissenslandkarte.mm (Freemind Datei; kann unter Ubuntu mit "snap install freemind" installiert werden)

Wissenslandkarte.svg