



Analyse und menschenzentrierte Erweiterung von lost.university

Studienarbeit HS24

Betreuer
Autoren

Frieder Loch
Laura Thoma, Stefanie Jäger

20.12.2024

1. Abstract

Dies ist das Abstract

Inhaltsverzeichnis

1. Abstract	2
1.1.1.1. Contextual Inquiry 1	4
1.1.1.2. Generelle Fragen	4
1.1.1.3. Überprüfen des bisherigen Plans	4
1.1.1.4. Austausch mit Kollegen	5
1.1.1.5. Recherche zu Modulen	5
1.1.1.6. Semesterplaner anpassen	6
1.1.1.7. Plan speichern	7
1.1.1.8. Nachgespräch	7
2. Hintergrund	8
2.1. Ist-Zustand	8
2.1.1. Architektur	8
2.1.1.1. Daten	8
2.1.1.2. Speicherformat der Daten	9
2.1.1.3. Code	12
2.1.1.4. Hosting	12
2.1.2. Funktionalität	12
2.1.3. Probleme	16
3. Methoden	17
4. Resultate	17
5. Diskussion	17
6. Danksagungen	17
7. Bibliographie	17
8. Glossar	17

1.1.1.1. Contextual Inquiry 1

Teilnehmer	Testperson 1 [Patrick]
Datum	09.10.2024
Ort	In ihrem Zimmer
Dauer	1 Stunde

1.1.1.2. Generelle Fragen

Seit wann studierst du?

Die Testperson hat ihr Studium im Herbstsemester 2021 gestartet.

In welchem Studienmodell (TZ2/3 oder Vollzeit, unterschiedlich)?

Sie hat für 1 Semester vollzeit studiert. Alle anderen Semester im TZ2-Modell.

In welchem Semester bist du gerade?

Sie ist aktuell im 7. Semester.

Wann planst du deinen Abschluss?

Sie hat seinen Abschluss für das Frühjahrsemester 2025 geplant. Sie schliesst also nach 8 Semestern ab.

Woher kennst du den Semesterplaner schon?

Der Semesterplaner wurde von Studenten und Dozenten empfohlen.

Wann nutzt du den Semesterplaner typischerweise?

Sie nutzt den Semesterplaner bei der Modulanmeldung, um den Stand seiner Studien abzufragen, aber auch zum Spass. Sie passt den Semesterplaner meist an, wenn die Modulauswahl in der 1. Phase bereits relativ festgelegt ist.

Wie häufig nutzt du den Semesterplaner?

Sie nutzt den Semesterplaner etwa monatlich. Während der Modulanmeldung nutzt er ihn häufiger, etwa 5 Mal, über mehrere Tage verteilt.

Hast du in der Vergangenheit auf eine andere Weise dein Studium geplant als mit dem Semesterplaner? Wenn ja, wie?

Sie hat seine ersten zwei Semester nach dem Musterstudienplan geplant.

Planst du eine Vertiefung?

Zu Beginn des Studiums hatte sie die Vertiefung „Software Engineering“ verfolgt, mittlerweile aber nicht mehr. Der Grund sei, dass ihr doch nicht alle Module der Vertiefung interessieren.

Folgst du einem Musterstudienplan? Wenn ja/nein, weshalb?

Sie folgt grösstenteils dem Musterstudienplan für Teilzeit „Software Engineering“, weicht aber leicht ab. Der Grund sei, dass ihr doch nicht alle Module des Musterstudienplanes interessieren.

Arbeitest du momentan neben dem Studium oder hast du mal neben dem Studium gearbeitet? Wenn ja, in welchem Pensum?

Ja, sie arbeitet zu 60%.

1.1.1.3. Überprüfen des bisherigen Plans

Wo/wie hast du den Plan gespeichert?

Sie hat die URL zu seinem Plan in einem Bookmark gespeichert. [Wirklich?]

Prüfst du den Plan? Wenn ja, wie und auf was?

Sie hat sich die URL zuletzt ohne Startsemester abgespeichert, also setzt sie das Startsemester und kontrolliert, dass sich die URL entsprechend angepasst hat.

Welche sichtbaren Informationen sind relevant für dich?

Die URL, die Module pro Semester, die Kategorien und ihre Erfüllung und das Meme.

Wie weiss der Planer, welche Credits du bereits erreicht hast?

Sie ist sich bewusst, dass er über die Wahl des Startsemesters die erreichten Credits berechnen lassen kann. Sie bemerkt auch, dass durch die Wahl des Startsemesters die Semester einen passenderen Namen erhalten.

Wie weiss der Planer, dass du ein Modul nicht bestanden hast?

Die Testperson hat bis jetzt kein Modul nicht bestanden, weshalb dies nicht aufkam.

1.1.1.4. Austausch mit Kollegen

Tauschst du dich mit Kollegen über deinen oder deren Plan aus?

Ja, sie tauscht sich mit Kollegen aus.

Weshalb und über was?

Sie interessieren die Meinungen und Erfahrungen von Absolventen eines Modules. Bei Kollegen holt sie sich Informationen zum effektiven Aufwand. Im Studentenportal holt sie sich Informationen zum Dozenten. Sie ist bereit ein Modul zu besuchen, auch wenn kein Kollege dies gleichzeitig tut.

Wie tauschst du dich aus? Online, IRL?

Sie tauscht sich in Person oder über Chats aus.

Wie teilst du/zeigst du deinen Plan?

Während der 1. und 2. Anmeldephase tauscht sie sich gelegentlich mit Kollegen über ihren Plan aus, aber teilt den Link eigentlich nie.

1.1.1.5. Recherche zu Modulen

Zu welchen Modulen holst du dir Infos?

Zu Modulen aus dem Musterstudienplan oder zu Modulen, die einer bestimmten Kategorie angehören.

Welche Infos zu Modulen holst du dir?

Infos zu den Kursen im Modul, die „Empfohlene Module“ und wer der durchführende Dozent ist. Sie achtet sich auf darauf, welche Module in der Liste zur Modulanmeldung rot oder gelb hinterlegt sind. In der 2. Phase fragt sie allenfalls auch bei Dozenten direkt nach, ob diese beabsichtigen die Vorlesung zum Modul zu streamen oder andersweitig aufzuzeichnen.

Wo holst du dir diese Infos?

Infos zu Module über Adunis. Meinung und Erfahrungen zu Module über Kollegen, die es bereits besucht haben. Sie wusste, dass man über ein Modul im Semesterplaner zu dessen Beschreibung auf Adunis gelangen kann.

Holst du Infos zu den Modulabhängigkeiten? Wenn ja, wo?

Schaut die „Empfohlene Module“ auf Adunis an.

Holst du dir Infos zu den Semestern/Moduldurchführungen? Wo?

Schaut die Durchführung auf Adunis nach.

Wie planst du die Erreichung einer Vertiefung?

Gar nicht mehr.

Wie planst du die Erfüllung der notwendigen Credits pro Kategorie?

Sie schaut sich die ProgressBar der Kategorien an. Über Adunis findet sie pro Kategorie eine Liste aller zugehörigen Module, welche sie mit den Modulen der momentanen Anmeldephase abgleicht, um ein Modul auszuwählen.

Hast du ein Zielmodul am Horizont?

Sie möchte gerne Compilerbau belegen. Dazu schaut sie sich die „Empfohlene Module“ an.

Hältst du dich an zuvor empfohlene Module? Wie planst du diese ein?

Sie plant Semester grob nach dem Musterstudienplan.

Was ist mühsam bei der Recherche und weshalb?

Adunis ist zu mühsam, um Module zu finden. Die Liste der Module bei der Modulanmeldung enthält zu viele Module und kann nicht gefiltert werden. Dasselbe mit der Liste der Module bei der Studienordnungsbeschreibung. Sie würde gerne Module nach deren Kategorie, Thema oder „Empfohlene Module“ filtern.

Was ist einfach bei der Recherche und weshalb?

Sie konsultiert gerne mit einer Kollegin, die bei der Planung hilfreiche Tipps gibt.

1.1.1.6. Semesterplaner anpassen

Wie findest du Module im Planer?

Sucht Module nach Namen. Suche nach Kürzel funktioniert bei ihr auf Chrome. Nutzt DragNDrop selten.

Auf welchen Geräten nutzt du den Planer/passt du den Planer an?

Laptop und Desktop.

Wie validierst du die Studierbarkeit deiner Modulauswahl?

Ihr war nicht bewusst, dass Module in Semestern hinzugefügt werden können, in denen sie gar nicht angeboten werden. Da sie Module anhand vom Musterstudienplan und der Liste bei der Modulanmeldung wählt, war dies jedoch nie ein Problem.

Wie erweiterst du die Semesterauswahl?

Sie weiss, dass sie auf den Knopf mit dem Plus klicken kann.

Welche sichtbaren Informationen sind relevant und weshalb?

Die Erfüllung der Kategorien ist ihr wichtig. Sie interessieren nur das jetzige und die nächsten Semester, die vorherigen nicht. Wenn sie jedoch die besuchten Semester nachschauen will, bevorzugt sie den Semesterplaner, da sie dort nach Semester gruppiert und farblich markiert sind. Sie mag, dass die Reihenfolge der Module in der URL persistiert werden.

Wer denkst du, steht hinter lost.university?

Es wurde bereits bei der Empfehlung erwähnt, dass es ein Studenten-Projekt ist. Sie weiss dank dem Footer auch, wer diese Studenten sind.

1.1.1.7. Plan speichern

Wie persistierst du deinen gemachten Plan?

Sie speichert sich die URL in ihrem OneDrive ab. [Oder Bookmark?]

Weisst du, dass die Daten im LocalStorage gespeichert werden?

Sie wusste es zuvor nicht. Als sie uns einen leeren Plan zeigen wollte und es stattdessen ihren Plan vorausgefüllt hat, war sie zuerst verwirrt. Sie kam dann aber selbst darauf, dass dieser wohl im LocalStorage hinterlegt sein muss.

Was ist daran einfach?

Sie mag, dass sie keinen Account anlegen muss. Sie mag auch, dass sie ihren Plan als Link teilen kann.

Was ist daran schwierig?

Sie hatte hier keine Anmerkungen.

1.1.1.8. Nachgespräch

Reflektiert diese Journey dein Vorgehen realistisch? Was machst du anders und gleich?

Ja.

Gehst du diese Schritte mehrfach durch? Auch in Bezug auf die verschiedenen Anmeldephasen?

Ja.

Inwiefern hilft dir der Planer?

Hilft bei der Übersicht über geplante und erreichte Credits pro Kategorie.

Welche Aspekte sind hilfreich am Planer?

Hier wurde nichts spezifisches erwähnt.

Welche Aspekte sind schwierig am Planer?

Sie empfindet die Farben der Kategorien als nicht zusammenpassend, da sie sehr verschiedene „Hues“ aufweisen. Der Planer übernimmt die Systemeinstellungen von Light & Dark Themes nicht. Das stört sie, da sich geblendet fühlt. Sie würde gerne die „Empfohlene Module“ von Modulen abgebildet sehen, um die Planung zu erleichtern. Sie würde gerne vergangene Semester einklappen können, um sich besser auf das aktuelle und die kommenden Semester konzentrieren zu können.

Gibt es weitere Aspekte, die bei der Modulwahl wichtig sind und wir noch nicht angesprochen haben?

Sie empfindet es als umständlich, wie viele Quellen sie konsultieren muss. Sie zählte auf: Adunis, Studentenportal, Kollegen und PDF des Musterstudienplanes.

Was möchtest du uns sonst noch sagen?

Das Anmelden eines Stundenplanes in der 2. Anmeldephase im TZ2-Modell empfindet sie als mühsam und wünscht sich auch hier Unterstützung durch ein Tool. Sie merkt an, dass der Semesterplaner wohl für den Campus Rapperswil optimiert sei und wundert sich, ob Studenten von

anderen Kampussen hier benachteiligt seien. Sie ist sehr vom Meme im unteren Bereich der Seite begeistert und wünscht sich sogar noch mehr Memes die abwechseln dargestellt werden.

2. Hintergrund

2.1. Ist-Zustand

Das ursprüngliche Projekt, auf dem unsere Arbeit aufbaut, kann als öffentliches GitHub Repository hier gefunden werden: <https://github.com/lost-university/web>.

Wir haben uns entschieden, einen Fork des Originalprojekts zu erstellen, um unsere Anpassungen unabhängig davon vornehmen zu können. Dadurch stellen wir sicher, dass spätere Änderungen am Originalprojekt, die nach Beginn unserer Semesterarbeit erfolgen, keinen Einfluss auf unseren Entwicklungsprozess haben.

Ein weiterer Vorteil eines Forks gegenüber einem Branch ist die einfachere Handhabung des Hostings (siehe Abschnitt 2.1.1.4).

2.1.1. Architektur

2.1.1.1. Daten

Alle Daten zu Modulen, Kategorien, Spezialisierung und Studienordnungen, welche die Applikation nutzt, sind öffentlich <https://studien.ost.ch> zugänglich.

Die Daten werden von einem Python-Crawler gesammelt, verarbeitet und anschliessend als JSON-Dateien im Data Repository abgelegt.

Die Detailseiten der Studienordnungen dienen dem Crawler als Einstiegspunkt. Informationen zu den geltenden Kategorien und benötigten Credits, den möglichen Spezialisierungen und den zugehörigen Modulen können so entnommen werden.

Die Aktualisierung der Daten erfolgt manuell. Vor Beginn eines jeden Semesters führt ein Maintainer den Crawler lokal aus, überprüft die Änderungen der Daten auf Spezialfälle und erstellt anschliessend ein neues Tag für die Daten. Zuletzt kann die verwendete Version der Daten im Client über den Tag in der URL angepasst werden.

Nachfolgend sind die relevanten Felder für eine Studienordnung gelistet. Die Beispieldaten dazu stammen von https://studien.ost.ch/allStudies/10191_I.json.

```
{
  "kredits": [
    {
      "kategorien": [
        {
          "bezeichnung": "Aufbau",
          "kuerzel": "I_Auf"
        },
        ...
      ],
      "minKredits": 48
    }
  ],
  "zuordnungen": [
    {
      "bezeichnung": "Application Architecture",
      "kuerzel": "M_AppArch",
      "url": "allModules/28236_M_AppArch.json",
    }
  ]
}
```



```

    "istAbschlussArbeit": false,
    "istPflichtmodul": false,
    "semEmpfehlung": 7,
    "kategorien": [
      {
        "bezeichnung": "Aufbau",
        "kuerzel": "I_Auf",
        "kreditpunkte": 4
      },
      ...
    ]
  },
  "spezialisierungen": [
    {
      "bezeichnung": "Frontend Engineering",
      "kuerzel": "FrEng",
      "url": "allStudies/10193_FrEng.json"
    },
    ...
  ]
}

```

Für jedes Modul wird anhand der "zuordnungen[].url" eine Anfrage gestellt, um die Informationen des folgenden Feldes zu erhalten.

```

{
  "kreditpunkte": 4
}

```

Für jede Spezialisierung wird anhand der "spezialisierungen[].url" eine Anfrage gestellt, um die Informationen der folgenden Felder zu erhalten.

```

{
  "zuordnungen": [
    {
      "kuerzel": "M_AppArch"
    },
    ...
  ]
}

```

2.1.1.2. Speicherformat der Daten

Das Vorgehen des Crawlers bei der Verarbeitung der zuvor genannten Daten ist im Diagramm Abbildung 1 ersichtlich.

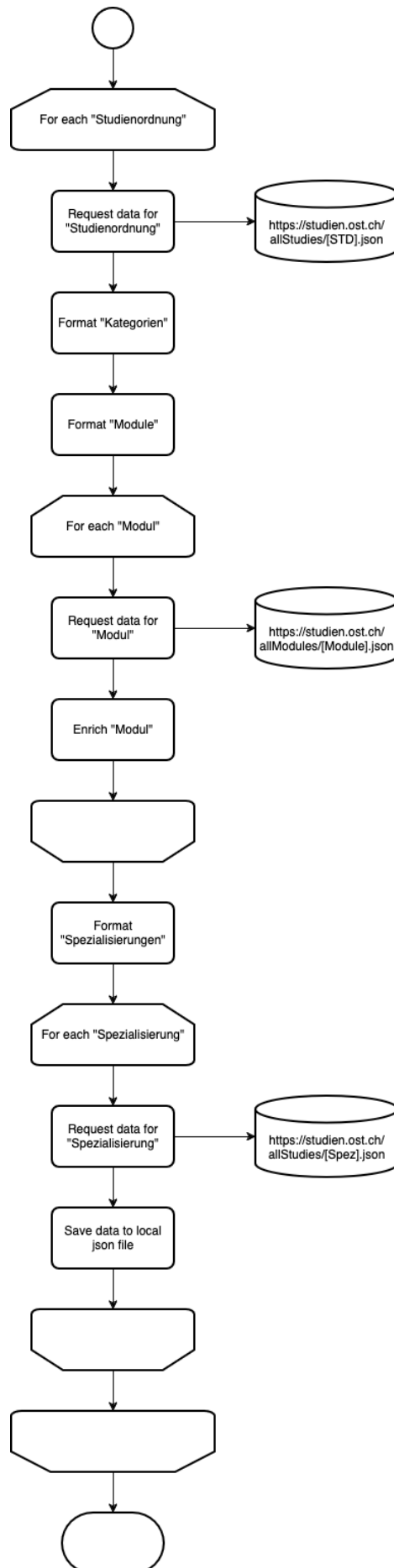


Abbildung 1: Ablauf des Crawlers

Um sicherzustellen, dass die Daten von der Applikation effizient abgefragt und verarbeitet werden können, wird ihr Format entsprechend angepasst. Auf diese Weise werden zudem redundante Daten für die Applikation nicht übernommen.

Module

```
[
  {
    "id": "AppArch",
    "name": "Application Architecture",
    "url": "allModules/28236_M_AppArch.json",
    "isThesis": false,
    "isRequired": false,
    "recommendedSemester": 7,
    "ects": 4,
    "categories_for_coloring": [
      "Auf",
      "Inf"
    ]
  },
  ...
]
```

Kategorien

```
[
  {
    "id": "Auf",
    "required_ects": 48,
    "name": "Aufbau",
    "modules": [
      {
        "id": "AppArch",
        "name": "Application Architecture",
        "url": "allModules/28236_M_AppArch.json"
      },
      ...
    ]
  },
  ...
]
```

Spezialisierungen

```
[
  {
    "id": "FrEng",
    "url": "allStudies/10193_FrEng.json",
    "name": "Frontend Engineering",
    "modules": [
      {
        "id": "AppArch",
        "name": "Application Architecture",
        "url": "allModules/28236_M_AppArch.json"
      },
      ...
    ]
  },
  ...
]
```

...
]

2.1.1.3. Code

Die Applikation selbst ist eine Single Page Application (SPA), entwickelt mit Vue. Anstelle von JavaScript wird dabei TypeScript verwendet. Die Icons stammen von Fontawesome, während das Styling überwiegend mit Tailwind CSS umgesetzt wird. Als Build-Tool kommt Vite zum Einsatz.

Vor Beginn unserer Arbeit wurde mit dem Stakeholder, welcher gleichzeitig Haupt-Maintainer ist, vereinbart, dass dieser Tech-Stack im Verlauf unserer Arbeit unverändert bleibt.

2.1.1.4. Hosting

Die Applikation wird über GitHub Pages gehostet. Da sie lediglich eine SPA ohne Backend ist, entfallen somit jegliche Kosten für das Hosting.

Die gewünschte URL, lost.university, wird auf GitHub unter Settings -> Pages -> Custom Domain hinterlegt. Die Domain wird so konfiguriert, dass sie über einen CNAME-Eintrag auf das Repository der Applikation verweist.

Wird ein Branch in den main-Branch gemergt, wird über einen GitHub Workflow eine GitHub Action ausgelöst, welche die SPA baut und deployt.

Anstelle einer Datenbank wird das Data Repository verwendet: <https://github.com/lost-university/data>. Die darin enthaltenen JSON-Dateien werden mithilfe von Tags versioniert.

2.1.2. Funktionalität

Folgende Funktionalitäten bestanden bereits vor Beginn dieser Semesterarbeit.

Als User kann ich über ein Dropdown einen Musterstudienplan auswählen, der vorausgefüllt angezeigt wird, sodass ich diesen als Grundlage für meinen eigenen Plan verwenden und bei Bedarf anpassen kann. Es stehen die Musterstudienpläne für alle Spezialisierungen, jeweils für das Teilzeit- und Vollzeit-Modell, zur Verfügung.

- Abbildung 2
- Als User sehe ich eine visuelle Repräsentation meines Planes.
 - Abbildung 3
- Als User kann ich Semester im Plan hinzufügen und entfernen, um meine Studiendauer abzubilden.
- Als User kann ich einem Semester ein Modul hinzufügen, das Modul beliebig verschieben und auch entfernen, um meinen Modulplan zu gestalten.
- Als User erhalte ich eine Fehlermeldung, wenn ich versuche, ein Modul einem zweiten Semester hinzuzufügen, um ungültige Pläne zu verhindern.
 - Abbildung 4
- Als User sehe ich, wie viele Credits ein geplantes Modul wert ist, und erkenne anhand der Farbgebung auch, zu welcher Kategorie es gehört.
- Als User kann ich durch einen Klick auf den Namen eines geplanten Modules zur entsprechenden Modulbeschreibung auf Adunis gelangen, um dort weitere Informationen abzurufen.
- Als User sehe ich die Summe der Credits aller Module eines Semesters in meinem Plan.

- Als User kann ich optional mein Startsemester eingeben, damit die Semester mit einem passenden Namen beschriftet, die bereits erreichten, geplanten und gesammthhaft benötigten Credits pro Kategorie dargestellt und nur mögliche Spezialisierungen angezeigt werden.
 - Abbildung 5
- Als User sehe ich, welche Module noch benötigt werden, um eine Spezialisierung zu erreichen.
 - Abbildung 6
- Als User erhalte ich eine Fehlermeldung, wenn mein Plan Module enthält, die nicht korrekt aufgelöst werden können. Diese Meldung gibt mir die Möglichkeit, das betroffene Modul aus meinem Plan zu entfernen.
 - Abbildung 7
- Als User sehe ich Memes im Semesterplaner, um die Stimmung beim Planen aufzulockern.
- Als Maintainer oder potentieller Maintainer sehe ich die Namen anderer Maintainer, die mit ihren GitHub-Profilen verlinkt sind, sowie einen Link zur GitHub-Seite des Semesterplaners, um mich zur Mitarbeit zu motivieren.
- Die geplanten Module, gruppiert nach Semester, und das ausgewählte Startsemester werden in der URL als Query-Parameter gespeichert.
- Die URL zum Plan wird im Local Storage abgelegt.
- Beim Öffnen eines leeren Planes, wird der Plan aus dem Local Storage verwendet, wenn ein solcher existiert.
- Beim Öffnen eines Plans, der den alten Namen eines Modules enthält, wird der Modulname automatisch auf den neuen Namen migriert.

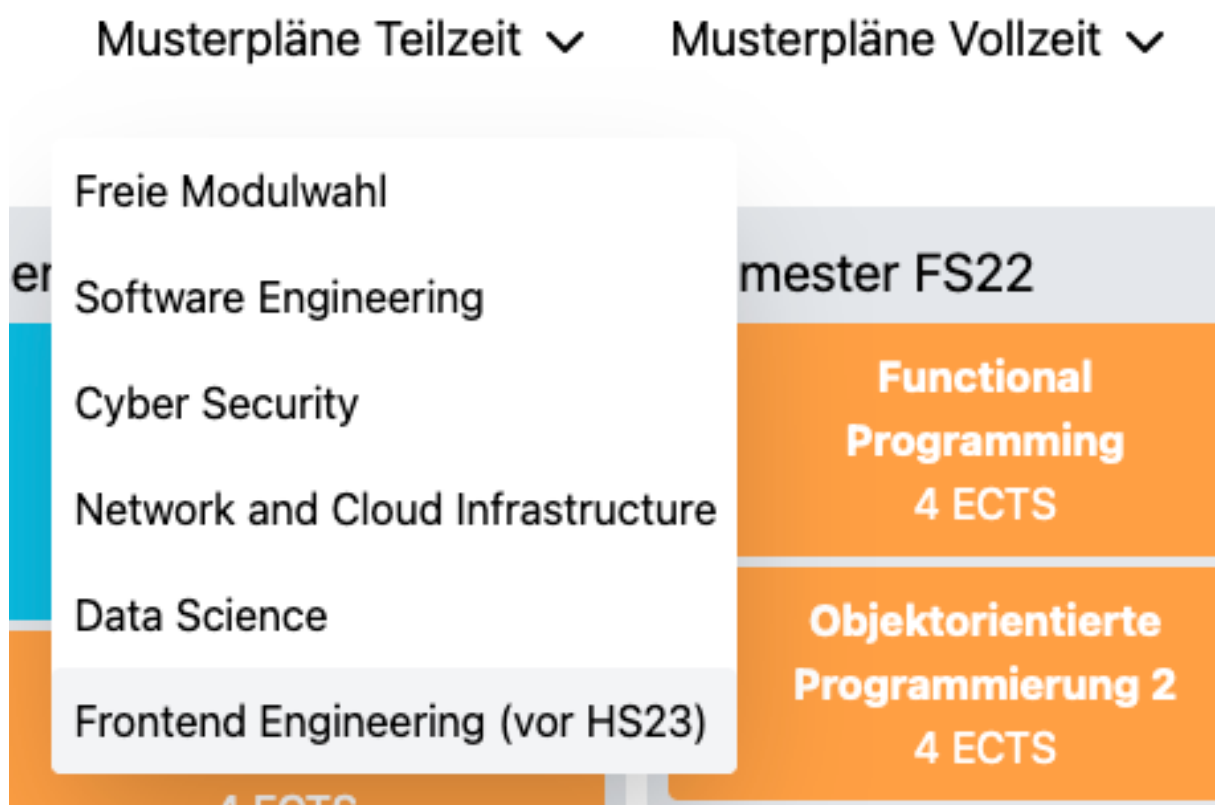


Abbildung 2: Dropdown zur Auswahl eines Musterstudienplans.

Semester HS21	Semester FS22	Semester HS22	Semester FS23	Semester HS23	Semester FS24	Semester HS24	Semester FS25	Semester HS25
Rhetorische Kommunikation für IngenieurInnen 4 ECTS Objektorientierte Programmierung 1 4 ECTS Automatisierung mit Python 2 ECTS Computernetze 1 6 ECTS Diskrete Mathematik für Informatiker 4 ECTS Analysis 1 für Informatiker 4 ECTS 24 ECTS	Functional Programming 4 ECTS Objektorientierte Programmierung 2 4 ECTS Digitale Codierungen 4 ECTS Cyber Security Foundations 4 ECTS Automaten und Sprachen 4 ECTS Analysis 2 für Informatiker 4 ECTS 24 ECTS	Web Engineering 1 4 ECTS Algorithmen und Datenstrukturen 4 ECTS Betriebssysteme 1 4 ECTS AI Foundations 4 ECTS Datenbanksysteme 1 6 ECTS English: The World of Science 4 ECTS 24 ECTS	Web Engineering 2 4 ECTS Secure Software 4 ECTS Betriebssysteme 2 4 ECTS AI Applications 4 ECTS Data Engineering 4 ECTS Kommunikation 2 für IngenieurInnen (Teamkommunikation) 4 ECTS 24 ECTS	Projekt- und Qualitätsmanagement 4 ECTS Web Engineering 3 4 ECTS .Net Technologien 4 ECTS UI Patterns and Frameworks 4 ECTS SE Practices 1 4 ECTS Experimentieren und Evaluieren für Informatiker 4 ECTS 24 ECTS	User Experience 4 ECTS Distributed Systems 4 ECTS SE Project 4 ECTS SE Practices 2 4 ECTS Digital Business für Informatiker 4 ECTS 20 ECTS	Computer Grafik 4 ECTS Application Architecture 4 ECTS Studienarbeit Informatik 6 ECTS Business Processes für Informatiker 4 ECTS 20 ECTS	Cloud Solutions 4 ECTS Bachelor-Arbeit Informatik 12 ECTS Physik Anwendungen für Informatiker 4 ECTS Modulsuche Physik 1 - Mechanik Physik 2 - Hydro-, Elektro und Thermo... Physik 3 - Schwingungen und Wellen, ... Physik Anwendungen für Informatiker 20 ECTS	+ 0 ECTS

Abbildung 3: Ein Beispiel für einen Plan.

Modul Physik Anwendungen für Informatik ist bereits im Semester 8

Abbildung 4: Die Fehlermeldung beim erneuten Hinzufügen eines Modules.

Übersicht der ECTS Punkte

Erstes Semester: **HS21** ▾

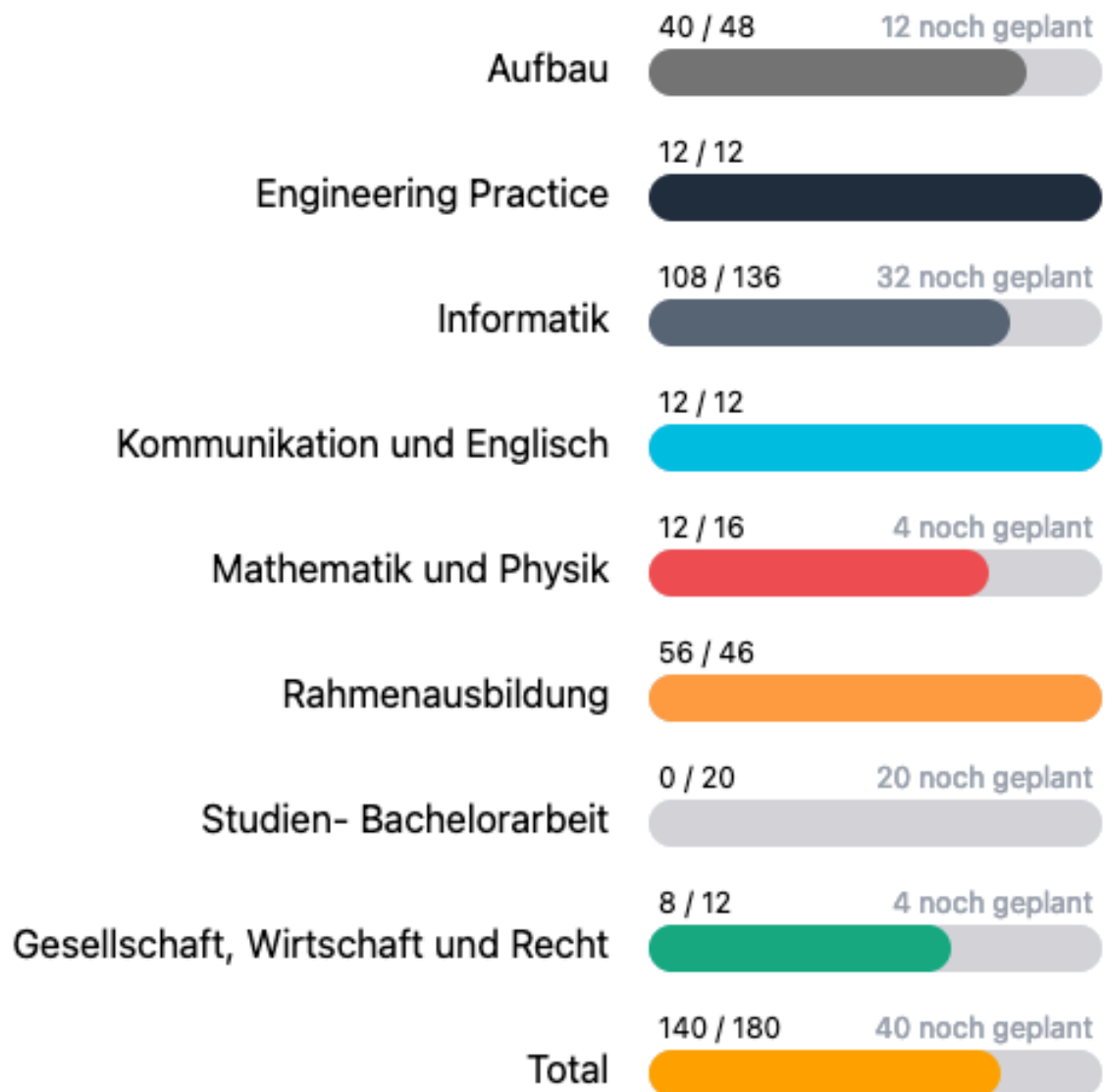


Abbildung 5: Die Auswahl des Startsemesters und die Übersicht der erreichten, geplanten und nötigen Credits pro Kategorie.

Vertiefungen

Cybersecurity	6 Module werden noch benötigt	▼
Data Engineering and Machine Intelligence	5 Module werden noch benötigt	▼
Frontend Engineering	Vertiefung geplant	▼
Network and Cloud Infrastructure	6 Module werden noch benötigt	▼
Software Engineering	2 Module werden noch benötigt	▲

Für die Vertiefung können noch folgende Module geplant werden:

- C++
- Patterns und Frameworks
- Parallele Programmierung

Abbildung 6: Die Spezialisierungen, mit Hinweis zu deren Erreichung.

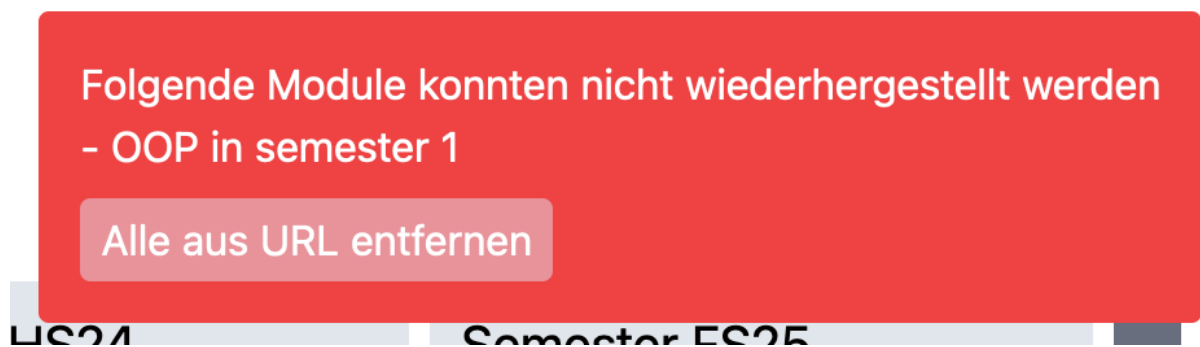


Abbildung 7: Die Fehlermeldung bei einem unbekannten Modul in der URL.

2.1.3. Probleme

Im folgenden werden alle Probleme und Bugs dokumentiert, die bereits zu Beginn unserer Arbeit bestanden, aber nicht im Rahmen unserer geplanten Anpassungen behoben werden sollen. Sollte genügend Zeit nach Erreichen unseres eigentlichen Zieles übrig sein, könnten wir diese ebenfalls noch angehen.

- Suche
 - Wird nach Hinzufügen eines Modules nicht zurückgesetzt.
 - Funktionalität eingeschränkt auf gewissen Browsern.
 - Modul nicht über Kürzel auffindbar.
- Daten
 - Keine Dokumentation der Daten von Adunis.

- Vermutete Inkonsistenzen.
- Jedes Semester manuelle Anpassungen für korrekte Migration notwendig.

3. Methoden

4. Resultate

5. Diskussion

6. Danksagungen

7. Bibliographie

8. Glossar

Kategorie: foo 8

Credit: foo 8

Spezialisierung: foo, auch Vertiefung genannt 8

Fontawesome. 12

Modul: foo 8

SPA – Single Page Application. 12

Studienordnungen: foo 8

Tailwind CSS. 12

Vite. 12