

ANFORDERUNGSSPEZIFIKATION

Version 1

06. Mai 2022



SPAsS-Tool
Arnold, Fieguth, Bohnert

Inhalt

Projektgrundlagen	2
Einführung und Ziele	2
Rahmenbedingungen	2
Abläufe und Funktionen	2
Anwendungsfälle	2
Szenario 001 „Zu Beginn des ersten Semesters“	2
Szenario 002 „Verzweiflung nach dem dritten Semester“	3
Szenario 003 „Fortschrittsregel beachten“	3
Szenario 004 „Admin bearbeitet Module“	3
Anwendungsfunktionen	4
Anwendungsfälle	5
Login	5
Stundenplan anzeigen	6
Berechnung der noch verbleibenden Studienzeit	6
Studienplan bearbeiten	7
Modul erzeugen	8
Module bearbeiten	9
Daten / Domänenmodell	9
„Gegenstandswelt“ des Systems / Datenmodell	9
Datentypenverzeichnis	9
Benutzungsschnittstellen	9
Dialogspezifikationen (z.B. GUI-Skizze)	9
Nichtfunktionale Anforderungen	12
Qualitätsanforderungen	12
Usability / Benutzbarkeit	12
Verlässlichkeit, Robustheit	12
Performance, Skalierbarkeit	12
Randbedingungen / Einschränkungen	12
Technischen Voraussetzungen	12
Ergänzende Bausteine	12
Glossar	12

Projektgrundlagen

Einführung und Ziele

Entwickelt wird das SPAsS-Tool (Semester-Planungs-Anwendung für studierende Studierende). Hierbei handelt es sich um einen individuellen interaktivem Studienplaner.

Das System ist dazu gedacht Studenten / Studierenden (wie auch immer) den Aufbau und die Planung ihres Studiums (Modulwahl etc.) zu erleichtern. Es wird möglich sein, dass jeder Studierende sich einen individuellen Stundenplan bauen kann. Es ist uns wichtig, die Anwendung Benutzerfreundlich zu gestalten, so dass der User mit gezielten Schritten durch unser System geleitet wird.

Rahmenbedingungen

Unser Team besteht aus 3 Leuten (Arthur Fieguth, Beate Arnold und Marie Bohnert). Der zu erfüllende Zeitraum streckt sich vom 06. Mai 2022 bis 07. Juli 2022.

Es sind zwei Zwischenbesprechungen vereinbart. Am 27. Mai wird der Standpunkt „Anforderungsspezifikation“ (Was soll gemacht werden) überprüft. Am 17. Juni wird das „Design-Dokument“ (Konzept der softwaretechnischen Umsetzung) vorgestellt. Es wird nach Abgabe des Projekts eine Vorstellung der fertigen Anwendung geben.

Die technischen Rahmenbedingungen setzen sich aus einer Linux funktionsfähigen Anwendung mit Gradle build-, test- und ausführbaren JavaFX-Anwendung (Java 17) zusammen.

Abläufe und Funktionen

Anwendungsfälle

Szenario 001 „Zu Beginn des ersten Semesters“

„Der Student Wolfgang beginnt im Wintersemester sein Studium an der Hochschule und möchte eine Gesamtplanung seines zukünftigen Studiums erstellen.

Er meldet sich im SpasS Tool mit seinem Namen und seiner Matrikelnummer an, um sich als Student auszuweisen. Nachdem Login wird Ihm eine grobe Studienplanung vorgeschlagen. Diese Planung richtet sich nach dem Standardprogramm der Hochschule, also alle Module je Semester.

Wolfgang hat im Zeitraum des ersten Semesters noch einige persönliche Termine, weswegen er sich dazu entscheidet ein Modul ins 3. Semester zu schieben, und sich da mehr Zeit zu nehmen.

Durch die Studienplanung wird errechnet, bis zu welchem Zeitpunkt sein Studium voraussichtlich laufen wird. Darüber hinaus wird der Zeitaufwand pro Semester berechnet und jeweils angezeigt.

Demnach ist es leichter für Wolfgang zu entscheiden, wie er seine Fächer schieben kann/soll um sein Studium optimal zu gestalten.

Bei jeder Verschiebung werden mögliche Fortschrittsregelungen geprüft. Wolfgang speichert seinen Plan nach erfolgreicher Planung.“

Szenario 002 „Verzweiflung nach dem dritten Semester“

„Die Studentin Liselotte Del a Bocha hat alle Module des ersten, zweiten und dritten Semesters erfolgreich abgeschlossen. Da sie nebenher als Studentin in einem Café arbeitet, war das dritte Semester allerdings sehr stressig und hat sie an ihre Grenzen gebracht. Sie möchte nun Module des vierten Semester schieben, damit sie ihren Nebenjob behalten kann. Sie meldet sich im SPAsS-Tool mit ihrem Hochschulbenutzernamen und Passwort an. In der Modulübersicht werden ihr die bereits abgeschlossenen Module mit einem grünen Haken symbolisiert. Die Anordnung der Module ist in der Standard Ansicht der Regelstudienzeit gewählt. Über einen Button, kann sie die Anordnung der Module verändern. Mit einer individuellen Anpassung der CPs, die sie pro Semester belegen möchte, wird ihr sofort ein neuer Plan, mit weniger Modulen pro Semester angezeigt. Über Drag & Drop kann sie jedoch die Module zusätzlich verschieben, wie es ihr am besten passt.“

Szenario 003 „Fortschrittsregel beachten“

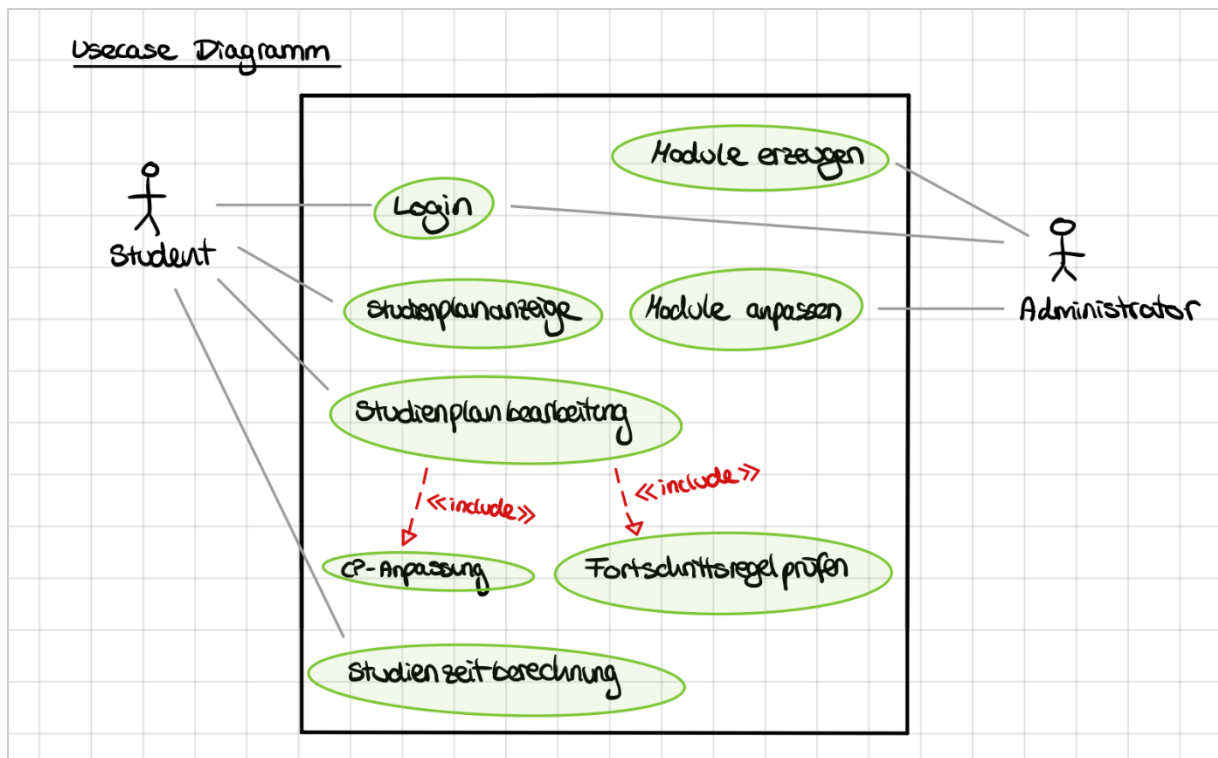
Die Studentin Theresa befindet sich im vierten Fachsemester des Studiengangs Medieninformatik. Sie möchte die Kursbelegung für das nächste Fachsemester planen. Auf einer Übersichtsseite sieht sie sich die Module des fünften Semesters an und überlegt, welche sie davon belegen möchte. Das System erinnert sie daran, dass ihr aus dem dritten Semester noch das Modul „Programmieren 3“ fehlt, damit sie nicht an der Fortschrittsregel hängen bleibt. Wählt sie dieses Modul noch dazu, wird ihr die gesamte CP Zahl des nächsten Semesters angezeigt. Sie kann in einem Feld auswählen, dass sie nur 30 CP machen möchte, woraufhin das System ihr vorschlägt ein Modul aus dem fünften Semester zu schieben, damit sie Prog3 machen kann und trotzdem innerhalb der vorgegeben 30CP bleibt. Nachdem sie sich für ihre Fächer entschieden hat, gelangt sie zurück zur Hauptseite und bekommt ihr geplantes nächstes Semester in ihrem aktuellen Studienverlaufsplan für sie individualisiert angezeigt.

Szenario 004 „Admin bearbeitet Module“

Der Administrator logt sich in das System mit speziellen Daten ein, und gelangt in eine Ansicht, in der er die Module bearbeiten kann. Er passt Lehrveranstaltungsbezeichnungen und Kompetenzen an. Da in diesem Jahr ein neues Modul hinzugekommen ist, aber ein Modul wegfällt, erstellt der Admin ein neues Modul und löscht das Modul, welches wegfällt.

Anwendungsfunktionen

TODO: BESCHRIFTUNG FEHLT





Anwendungsfälle

Login

Autor: Marie Bohnert

Akteure: Administrator/In, Studierende

Fachlicher Auslöser: User will beliebige Funktion der Anwendung nutzen

Vorbedingungen: Anwendung starten, Nutzer muss angelegt sein

Standardablauf:

- 1. Administratorin / Studierende: Benutzername und Passwort eingeben
- 2. System: Korrektheit der Eingabe sicherstellen
- 3. System: Rechte prüfen und Eingaben mit Account verknüpfen
- 4. System: Startseite öffnen

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 2a System findet keine Übereinstimmung der Eingabe

Nachbedingung/Ergebnis: Startseite ist offen

Nicht-funktionale Anforderungen: Reaktionszeit <10 sec.

Parametrisierbarkeit / Flexibilität: Je nach Fehlversuchsanzahl konfigurierbarer Hinweistext

Nutzungshäufigkeit / Mengengerüst: bei jeder Benutzung

Stundenplan anzeigen

Autor: Marie Bohnert

Akteure: Studierende

Fachlicher Auslöser: Kursbelegungen

Vorbedingungen: Korrekte Eingabe beim Login

Standardablauf:

- 1. System: lädt aufgrund des Nutzerkontos den individuellen Stundenplan
- 2. System: Stundenplan anzeigen

Nachbedingung/Ergebnis: Stundenplan und Funktionen sind sichtbar

Nicht-funktionale Anforderungen: Reaktionszeit <10 sec.

Parametrisierbarkeit / Flexibilität: individuelle Stundenplananzeige

Nutzungshäufigkeit / Mengengerüst: nach Login, immer

Berechnung der noch verbleibenden Studienzeit

Autor: Beate Arnold

Akteure: Studierende

Fachlicher Auslöser: der Studierende möchte wissen, wie lange er voraussichtlich noch studiert

Vorbedingungen: es existiert ein individueller Studienplan, es ist eingetragen, welche Module schon bestanden wurden, (erfolgreicher login?)

Standardablauf:

- 1. System: Anzeige des momentanen Studienplans
- 2. System: Berechnung der Zeit der weiteren Studiendauer in Monaten oder Jahren
- 3. System: Anzeige der Zeit

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 2a der Studierende bearbeitet/verändert seinen Studienplan
- 2a1 Neuberechnung durch das System

- 2a2 weiter bei 3

Nachbedingung/Ergebnis: Anmeldesystem ist wieder in der Startansicht (-> Studienplananzeige)

Nicht-funktionale Anforderungen: Reaktionszeit <10 sec.

Parametrisierbarkeit / Flexibilität: Anzeige in Monaten, Jahren oder Semestern auswählbar

Nutzungshäufigkeit / Mengengerüst: bei jeder Anmeldung im System

Studienplan bearbeiten

Autor: Beate Arnold

Akteure: Studierende

Fachlicher Auslöser: Veränderung des Studienplans durch Wunsch des Studierenden

Vorbedingungen: Module sind im System angelegt

Standardablauf:

- 1. System: Anzeige des momentanen Studienplans im Bearbeitungsmodus
- 2. Studierende: Verschieben der einzelnen Module zwischen den Fachsemestern
- 3. System: Berechnung der CP pro Semester
- 4. System: Prüfung, ob Veränderung möglich ist
- 5. System: Veränderung im neuen Studienplan anzeigen
- 6. Studierende: Bestätigung der Veränderung
- 7. System: Speicherung
- 8. System: Verlassen des Bearbeitungsmodus

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 3a Wunsch CP Anzahl ist überschritten
- 3a1 System zeigt Hinweis an
- 3a2 System schlägt Alternativlösung vor
- 3a3 Studierende kann Alternativlösung annehmen, oder erneut eigenständig Veränderungen vornehmen (weiter bei 2)
- 3a4 weiter bei 5
- 4a Veränderung der Studierenden ist so nicht möglich
- 4a1 System schlägt Alternativlösung vor
- 4a2 Studierende kann Alternativlösung annehmen, oder erneut eigenständig Veränderungen vornehmen (weiter bei 2)
- 4a3 weiter bei 5
- 4b Studierende hat beim Verändern die Fortschrittsregel nicht beachtet
- 4b1 System zeigt Hinweis/ Erinnerung an
- 4b2 System schlägt Alternative vor, welche die Fortschrittsregel beachtet
- 4b3 Studierende kann Alternativlösung annehmen, oder erneut eigenständig Veränderungen vornehmen (weiter bei 2)
- 4b4 weiter bei 5
- 6a Ablehnung der Veränderung -> Abbrechen

- 6a1 weiter bei 8

Nachbedingung/Ergebnis:

Anmeldesystem ist wieder in der Startansicht (-> Studienplananzeige)

Nicht-funktionale Anforderungen: Reaktionszeit <10 sec.

Parametrisierbarkeit / Flexibilität: konfigurierbare Hinweistexte je nach Fehlersituation

Nutzungshäufigkeit / Mengengerüst: mehrfach im Semester

Modul erzeugen

Autor: Arthur Fieguth

Akteure: Administrator

Fachlicher Auslöser: Ein neues Modul wird in einem Semester angeboten, und muss dem Plan hinzugefügt werden

Vorbedingungen: Der Administrator muss sich erfolgreich eingeloggt haben und das Modul muss mit allen nötigen Informationen vorliegen

Standardablauf:

- 1. Admin: Tätigt Button um ein neues Modul zu erzeugen.
- 2. System: Gibt Ansicht, auf der das Modul angelegt werden kann aus
- 3. Admin: Trägt alle vorgegebenen Informationen in die Eingabefelder ein.
- 4. Admin: Speichert das neue Modul
- 5. System: Prüft, alle Eingabefelder

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 3a1 Das Modul ist schon vorhanden und kann nicht doppelt angelegt werden
- 5a1 Eingabefelder sind nicht ausgefüllt oder die Validierung schlägt Fehler aus

Nachbedingung/Ergebnis:

Admin bekommt Rückmeldung, dass das neue Modul erfolgreich angelegt wurde. Dieses ist nun bei den Studenten im jeweiligen Fachsemesters des Moduls sichtbar.

Nicht-funktionale Anforderungen: Reaktionszeit

Parametrisierbarkeit / Flexibilität: TODO nochmal fragen

Nutzungshäufigkeit / Mengengerüst: Dauerhaft

Module bearbeiten

Autor: Arthur Fieguth

Akteure: Administration

Fachlicher Auslöser: Lehrveranstaltung oder Kompetenzen ändern sich und müssen auch im Spass-Tool angepasst werden

Vorbedingungen: Alle Informationen zu den Lehrveranstaltungen/Kompetenzen müssen vorliegen

Standardablauf:

- 1. Admin: Wählt das Modul, welches er anpassen möchte aus.
- 2. System: Zeigt das Modul an, mit der Möglichkeit Lehrveranstaltungen Kompetenzen etc. zu ändern
- 3. Admin: Passt Texte/Abhängigkeiten etc. bei den Lehrveranstaltungen an
- 4. Admin: Speichert die Änderungen ab
- 5. System: Prüft, alle Eingabefelder

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 3a1 Es dürfen nur Texte bis zu einer bestimmten Länge eingegeben werden (ansonsten Fehlermeldung)
- 5a1 Eingabefelder sind nicht ausgefüllt oder die Validierung schlägt Fehler aus

Nachbedingung/Ergebnis:

System zeigt eine Zusammenfassung der Module als Übersicht an

Nicht-funktionale Anforderungen

Reaktionszeit

Parametrisierbarkeit / Flexibilität:

Nutzungshäufigkeit / Mengengerüst: Soll dauerhaft möglich sein

Daten / Domänenmodell

„Gegenstandswelt“ des Systems / Datenmodell

Klassendiagramm TODO

Datentypenverzeichnis

TODO, Entitäten aufgelistet?

Benutzungsschnittstellen

Dialogspezifikationen (z.B. GUI-Skizze)

TODO: Beschriften

Spass-Tool

Login

Name

Passwort

Studienplanung

7. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
6. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
5. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
4. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
3. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
2. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>
1. 30Cp Fachsemester	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>	<div>Modulname</div> <div>Vorlesung 5CP Praktikum 5CP</div> <div>10 CP</div>

Übersicht

10/190 CP

5%

Abschluss voraussichtlich

📅 Sommer 2024

- 7. Semester
- 6. Semester
- 5. Semester
- 4. Semester
- 3. Semester
- 2. Semester
- 1. Semester**

Plan bearbeiten

abbrechen

Bearbeitungsmodus/Erstellungsmodus



Allgemein

Modulname

Anzahl CP

Winter & Sommersemester



Vorlesung ☒

Kompetenz 1

Kompetenz 2

Kompetenz 3

Kompetenz 4

Praktikum ☒

Kompetenz 1

Kompetenz 2

Kompetenz 3

Kompetenz 4

Praktikum ☒

Kompetenz 1

Kompetenz 2

Kompetenz 3

Kompetenz 4

Modulbearbeitung



Modulname



Modulname



Modulname



Modulname



Modulname



Modulname



Modulname



Modulname



Nichtfunktionale Anforderungen

Qualitätsanforderungen

Usability / Benutzbarkeit

- JavaFX GUI muss so eindeutig gestaltet sein, dass Benutzer es ohne Erklärung individuell nutzen kann
- Leistungen / Module (Kästchen) können per Drag & Drop zwischen Semestern verschoben werden (Orientierung an bekannten Angebots-Restriktionen)

Verlässlichkeit, Robustheit

- Mindestens alle Hauptfunktionen müssen anhand von JUnit5-Tests testbar sein
- Der Benutzer muss darauf hingewiesen werden, wenn er falsch mit dem System interagiert – zum Beispiel die Fortschrittsregelung beim Verschieben der Module nicht einhält

Performance, Skalierbarkeit

- Erweiterbarkeit / Anpassbarkeit, ein Administrator soll ohne weitere Schulung neue Module hinzufügen können
- Jeder Student kann seinen individuellen Studienplan im System einpflegen

Randbedingungen / Einschränkungen

- Das System muss individuelle Studien-Ausgangssituationen umsetzen können
- Das System muss lauffähig zum 07.07.2022 abgegeben werden
- Datenschutz muss eingehalten werden

Technischen Voraussetzungen

- Muss JavaFX verwenden
- Muss unter Java 17 laufen
- Ausführbare JUnit5-Test

Ergänzende Bausteine

Glossar