

Projekt XXX

Anforderungen

Version 3 / 28 Nov 2022

Inhaltsverzeichnis

1. Hintergrund	1
2. Allgemeine Beschreibung	3
2.1. Anspruchsgruppen	3
2.2. Benutzende und deren Eigenschaften	4
2.3. Risiken	5
2.4. Ressourcen	5
3. Funktionale Anforderungen	6
4. Weitere Anforderungen	10
4.1. Nicht-funktionale Anforderungen	10
4.2. Externe Schnittstellen	10

1. Hintergrund

Dieser Teil beschreibt die wichtigsten Anforderungen an das System und dessen grundlegende Eigenschaften in einigen wenigen Sätzen.

Folgende Fragen sollen in diesem Abschnitt beantwortet werden

- Welches Problem wird durch dieses Projekt gelöst und für wen?
- Was sind die wichtigsten Bedürfnisse der Benutzenden und warum?
- Welche Ziele sollen erreicht werden?

und auch grob skizziert:

- Wie wird das Problem gelöst und welcher Ansatz wird gewählt?
- Was ist der Unterschied zu bestehenden Lösungen?

Dazu kann z.B. folgendes Schema verwendet werden:

Problem Statement

The problem of [describe the problem]
affects [the stakeholders affected by the problem]
the impact of which is [what is the impact of the problem?]
a successful solution would be [list some key benefits of a successful solution]

Product Position Statement

For [target customer]
Who [statement of the need or opportunity]
The (product name) is a [product category]
That [statement of key benefit; that is, the compelling reason to buy]
Unlike [primary competitive alternative]
Our product [statement of primary differentiation]

Ein Beispiel:

Die Informationen über Inhalte und Stundenpläne der Module an der HLS sind unübersichtlich, unattraktiv, schwer zugänglich und wechseln jedes Semester.

Die Studierenden der HLS kommen nicht effizient an diese Informationen und erhalten ein negatives Bild der HLS als moderne Bildungsinstitution.

Eine bessere Lösung würde stets aktuelle und dem Kontext (Zeit, Person) angepasste Informationen einfach und elegant zur Verfügung stellen.

Die HLS Stundenplan-App richtet sich an Studierende der HLS. Sie ermöglicht die Einsicht in alle aktuellen HLS Semesterstundenpläne und Modulbeschreibungen. Die HLS Stundenplan-App ist eine Smartphone Applikation und kann via "App Store" (iOS) respektive "Google Play" (Android) kostenlos installiert werden.

Im Gegensatz zur aktuellen Lösung mit Moodle-basierten PDF Dokumenten wird die HLS Stundenplan-App die Stundenpläne und Modulinhalte effizient und attraktiv nutzbar machen.

2. Allgemeine Beschreibung

Dieser Teil beschreibt die generellen Faktoren welche das Produkt und die Anforderungen daran beeinflussen. Es sollte sich um Hintergrundinformationen handeln und nicht um spezifische Anforderungen.

Das hier ist nur ein Kommentar und hier muss eigentlich nichts geschrieben werden. die Inhalte befinden sich in den folgenden Unterabschnitten ;-)

2.1. Anspruchsgruppen

In diesem Abschnitt sollen folgende Fragen beantwortet werden

- Wer hat welches Interesse am zu erstellenden System?
- Durch wen sind sie im Projekt repräsentiert?

Dies können Personen oder Organisationen sein, etc. Dies beinhaltet auch Leute, welche das Produkt nicht unbedingt benutzen werden, aber ein Interesse daran haben oder davon betroffen sind (z.B. Geldgeber, Gesetzgeber). Trotzdem können sich aus ihren Bedürfnissen Anforderungen ergeben.

Hier ein Beispiel der Anspruchsgruppen für die Entwicklung einer mobilen Gesundheits-Coaching App:

Stakeholder	Is user of the system	Description	Represented by
Patients	x	Young people that want to be healthier. Are looking for support to change their lifestyle/behaviour. Need for long-term professional coaching.	Members of GetFit Health Club
Caregivers	x	Want to provide personalized care in an efficient and effective way. Want to be able to overview about 100-200 patients. Have some medical background, but are typically no doctors.	Sheldon Cooper
Administrators	x	Want to be able to efficiently manage participant data like addresses, phone numbers and the like.	Sheldon Cooper
Family doctor		The family doctor (or general practitioner) is the first contact person for any health issues. His opinion is very important and he should be informed through reports.	Dr. Amy Santiago
Sponsor		Wants to be able to demonstrate that they develop innovative products in the area of integrated ambulatory health care.	Michael Scott
Developer		Wants that the project produces a gain in know-how and attracts as much publicity as possible. This is a flagship project for the young Institute of Interdisciplinary Information Technology.	Howard Wollowitz
Government (BAG)		Wants to enforce laws (e.g. privacy, quality) and guidelines for medical services.	Dwight Schrute

2.2. Benutzende und deren Eigenschaften

In diesem Abschnitt sollen folgende Fragen beantwortet werden

- Wer wird das System benutzen und wie sind diese charakterisiert?

Fragen wie etwa: wie häufig werden sie das Produkt nutzen? (täglich, einmal pro Jahr, etc.) Unter was für Bedingungen? (unter Stress, gemütlich im Büro, im Gedränge unterwegs, etc.)

Hier wieder ein Beispiel der Benutzenden für die Entwicklung einer mobilen Gesundheits-Coaching App:

User	Description
Patients	Typically aged 14-20. Will mostly use the mobile version of the application on their mobile phones while under way or between activities.
Caregivers	Have some medical background, but are typically no doctors. Sometimes also staffed by part-time volunteers that infrequently use the application. Will mostly use the application in an office setting.
Administrators	This will typically be one of the caregivers.

Die Benutzenden sollen dabei als Rollen und nicht als konkrete Personen betrachtet werden. Ein Person kann verschiedene Rollen spielen, z.B. wenn einer der Caregivers gleichzeitig noch die Administration der Patienten übernimmt.

Zu jeder Rolle muss mindestens ein Use Case/eine User Story definiert sein.

Die Beschreibung der Benutzenden kann bei Bedarf auch ausführlicher, z.B. in der Form von Personas sein. Wie immer gilt, dass sie der Grösse/Wichtigkeit/Komplexität des Projektes angepasst sein soll.

2.3. Risiken

Eine Beschreibung der wichtigsten Risiken und welche Massnahmen getroffen werden, um diese zu vermindern.

Typische Beispiele sind die Verwendung von neuen Technologien, Abhängigkeiten von anderen Organisationen, personelle Flaschenhälse oder Klumpenrisiken, etc.

2.4. Ressourcen

Welche Ressourcen stehen zur Verfügung (Personen, Know-how, Werkzeuge, etc.) und wie sind diese organisiert?

3. Funktionale Anforderungen

Dieser Teil beschreibt die spezifischen funktionalen Anforderungen an das System.

Die funktionalen Anforderungen legen fest, WAS das System tun soll, im Gegensatz zu den nicht-funktionalen Anforderungen (folgen weiter unten), welche festlegen, WIE ein System etwas tun soll (Performanz, Sicherheit, verwendete Technologie, etc.).

Dazu müssen die Antworten zu folgenden Fragen bekannt sein

- Welche Aufgaben wollen die Benutzenden mit dem System durchführen können?
- Welche Bedürfnisse haben sie?
- Welche Ziele wollen sie erreichen?

Die daraus resultierenden Anforderungen an das System müssen detailliert genug sein, damit sie einerseits als Grundlage für den Entwurf des Systems dienen können und damit die Kunden andererseits die Funktionalität verifizieren können.

Funktionale Anforderungen können entweder als Anwendungsfälle (use cases, user stories) oder als Liste von Produkteigenschaften (Features) formuliert werden.

Anwendungsfälle eignen sich für interaktive Systeme mit einem grossen Anteil an Benutzerinteraktion (z.B. Analyse von Daten, Suche nach Informationen). Features sind eher für technische Systeme mit wenig Interaktion geeignet (z.B. Steuerung von Anlagen, Regelung von Prozessen).

User Story: "Die Benutzer wollen ... machen können, weil ..."

Use Case: "Die Benutzer machen ..., das System reagiert mit ..., etc."

Feature: "Das System soll ... können"

Use Cases und User Stories sind ein Versuch, die funktionalen Anforderungen an ein System zu beschreiben. Use Cases sind in der Regel formeller und beschreiben genaue Abläufe, während User Stories informelleren Charakter haben.

Welche der beiden Formen gewählt wird, hängt von der Art des Systems ab, das entwickelt werden soll.

Use Cases eignen sich für Systeme mit sequenziellen und wohl definierten Abläufen (z.B: Bedienung eines Bancomates zum Geld abheben). User Stories sind besser geeignet für ereignisgesteuerte und explorative Systeme.

Bei beiden gilt es aber folgendes zu beachten:

- Sie sind aus der Perspektive des Benutzers geschrieben
- Sie sind in der Sprache der Benutzer geschrieben und vermeiden technischen Jargon
- Sie beschreiben das Verhalten des Systems und führen zu einem Ziel das für den Benutzer einen Nutzen hat
- Sie sollten nummeriert sein, damit man sie referenzieren kann.

Hier ein Beispiel mit User Stories:

Als StudentIn möchte ich...

1. den passenden Stundenplan auswählen und anschauen können, weil ich mich vor allem für den von mir interessiere.
2. den Stundenplan für eine ganze Woche aufs Mal sehen, damit ich einen Überblick habe.
3. den Stundenplan für einen bestimmten Tag sehen, um detailliert planen zu können.
4. eine Modulbeschreibung suchen und anschauen können, damit ich sehe, was die Inhalte und Prüfungsmodalitäten sind.
5. Module nach Namen suchen können, weil ich nicht weiss, wann sie stattfinden oder sie nicht in meinem Stundenplan sind.

Ein Beispiel für Use cases in der einfachsten Form:

1. Stundenplan auswählen: Die Studierenden wählen Studiengang, Semester und Vertiefung. Das System zeigt den entsprechenden Stundenplan in der momentan gewählten Ansicht.
2. Stundenplan im Überblick anzeigen: Die Studierenden wählen den Modus "Wochenübersicht". Das System zeigt den ausgewählten Stundenplan als Wochenübersicht.
3. Stundenplan in Tagesansicht anzeigen: Die Studierenden wählen den Modus "Tagesansicht". Das System zeigt den ausgewählten Stundenplan als Tagesübersicht.
4. Modulverzeichnis anzeigen: Die Studierenden wählen den Modus "Modulverzeichnis". Das System zeigt eine alphabetische Liste der Module. Die Studierenden aktivieren einen Filter (Alle, MLS, LST) um die Liste der Module einzuschränken.
5. Modulbeschreibung anzeigen: Die Studierenden wählen ein Modul aus (in der Wochen, Tages oder Verzeichnisanzeige). Das System zeigt die dazugehörige Modulbeschreibung. Nach dem Lesen schliessen die Studierenden die Modulbeschreibung. Das System kehrt zurück zur vorherigen Ansicht.
6. Nach Modulen suchen: Die Studierenden geben den Namen (oder einen Teil davon) eines Moduls ein. Das System zeigt in der jeweiligen Ansicht die dazu passenden Module.

Und die Version mit Features:

Das System soll...

1. Auswahl des Stundenplans nach Studiengang, Semester und Vertiefung ermöglichen
2. Wochenübersicht anzeigen
3. Tagesansicht zeigen
4. Modulverzeichnis anzeigen
5. Filtern nach Modultyp ermöglichen
6. Modulbeschreibung anzeigen
7. Suche nach Modulnamen ermöglichen

Oft hängen funktionale Anforderungen auch stark vom Kontext ab, in dem das System tatsächlich eingesetzt wird. Um diesen Kontext zu beschreiben, eignet sich die Erstellung von Szenarien. Szenarien sind kurze Geschichten mit fiktiven Personen, welche eine typische Anwendung des Systems beschreiben.

Am Besten bittet man die zukünftigen Anwendenden, solche Szenarien zu schreiben. Falls das nicht geht, kann man sie auch selbst schreiben und dann mit den Anwendenden diskutieren, man wird dann schnell erfahren, ob man die gleiche Vorstellung vom System hat.

Hier zwei kurze Beispiele für Anwendungsszenarien:

Szenario 1

Es ist Sonntag Abend und Susanne Meier packt ihren Laptop in den Rucksack. Morgen beginnt das neue Semester. Mit dem 5. Semester beginnt auch endlich die Vertiefung und Susanne freut sich auf die spezifischen Vorlesungen. Sie will auf ihrem Smartphone die Zugverbindungen für Montag nachschauen, da fällt ihr ein, dass ja der Stundenplan geändert hat. Sie startet die HLS App. Susanne wählt das aktuelle Semester sowie die Vertiefung "Biomedizinische Informatik". Sie sieht einen Wochenuübersicht über die gelesenen Module. Das erste Modul am Montag Morgen beginnt zum Glück erst in der zweiten Stunde, dafür ist die Mittagspause recht kurz. Sie schaut noch schnell nach, ob sie für das erste Modul ein Buch hätte kaufen müssen, stellt aber erleichtert fest, dass das nicht der Fall ist. Susanne schaltet ihr Smartphone in den Flugmodus und geht schlafen.

Szenario 2

Gerd Müller möchte nächsten Herbst an der HLS studieren. Er liest in den Stellenanzeigen, dass zur Zeit viele VerfahrenstechnikerInnen gesucht werden und möchte sich vergewissern, dass dieses Thema auch in der Ausbildung vorkommt. Er erinnert sich daran, dass Frank Pude am Infotag gesagt hat, dass die detaillierten Modulbeschreibungen in der HLS App verfügbar sind. Er findet die App im AppStore und installiert sie schnell. Er startet die App und wählt das Modulverzeichnis. Beim Durchstöbern der Modulbeschreibungen stellt er beruhigt fest, dass das der Fall ist.

Solche Szenarien sind auch gut dazu geeignet, das erstellte System (oder frühe Prototypen) zu testen. Man kann versuchen, das Szenario durchzuspielen und feststellen, ob das reibungslos klappt.

4. Weitere Anforderungen

Dieser Teil beschreibt die spezifischen weiteren Anforderungen an das System. Die Anforderungen müssen detailliert genug sein, damit sie einerseits als Grundlage für den Entwurf des Systems dienen können und damit die Kunden andererseits die Funktionalität verifizieren können.

Das hier ist nur ein Kommentar und hier muss eigentlich nichts geschrieben werden. die Inhalte befinden sich in den folgenden Unterabschnitten ;-)

4.1. Nicht-funktionale Anforderungen

In diesem Abschnitt sollen folgende Fragen beantwortet werden

- Welche Qualitäten soll das System aufweisen
- Wie werden diese überprüft/gemessen?

Auch Qualitätsanforderungen oder ergänzende (supplementary) Anforderungen genannt. Während die funktionalen Anforderungen beschreiben, was das System machen soll, beschreiben die nicht-funktionalen Anforderungen, wie gut das System etwas machen soll. Dies hat in der Regel Einfluss auf die Architektur des Systems.

Wichtig ist, dass die Anforderungen überprüfbar sind. "Das System muss einfach zu benutzen sein" ist z.B. schwierig zu überprüfen. Besser wäre z.B. "Mit dem standardisierten Fragebogen xy beantworten 80% der Befragten die Benutzbarkeit positiv".

- Benutzbarkeit
- Zuverlässigkeit
- Leistung
- Sicherheit
- Wartbarkeit
- Implementierung, Prozess
- Gesetzliche Einschränkungen
- Organisatorische Anforderungen

Dies ist keine abschliessende Liste. Es muss aber auch nicht überall etwas ausgefüllt werden, sondern nur bei den Punkten, die für das konkrete Projekt relevant sind.

4.2. Externe Schnittstellen

In diesem Abschnitt sollen folgende Fragen beantwortet werden

- Mit welchen externen technischen Systemen (Hardware, Software, Kommunikation) muss das zu bauende System Informationen austauschen?
- Wie sind diese Schnittstellen spezifiziert?

In diesem Abschnitt werden Schnittstellen definiert, die ausserhalb vom eigenen Einflussbereich

liegen. Das sind Abhängigkeiten, die - wenn nicht spezifiziert - zu Missverständnissen und Problemen führen, für die man nicht verantwortlich sei will.

Typische Schnittstellen sind etwa:

- Hardwareschnittstellen (z.B. Anbindung von Geräten)
- Softwareschnittstellen (z.B. Datenformate die gelesen werden müssen, APIs die in einer bestimmten Version genutzt werden müssen)
- Kommunikationsschnittstellen (z.B. Kommunikation mit verteilten Systemen)

Auch hier gilt natürlich, dass man nur etwas schreiben soll, wenn es auch tatsächlich für dieses Projekt relevante externe Schnittstellen gibt.