

---

# Entwicklungsprojekt

---

## AUDIT 3

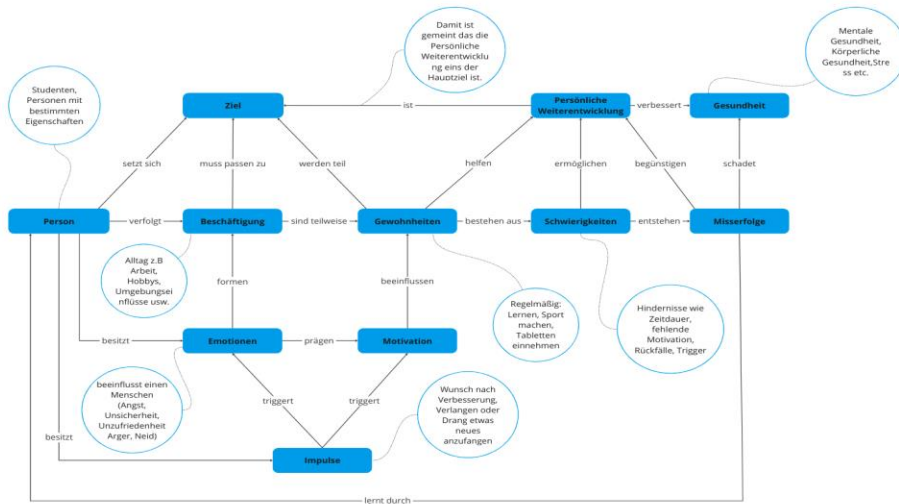
ADONISA GASHI  
MADINA IBRAGIMOVA  
SEKARJA BENAGGOUNE

# Inhaltsverzeichnis

---

1. Überarbeitetes Domänenmodell
2. Überarbeitete Stakeholder Tabelle
3. Überarbeitete Anforderungen und Erfordernisse
4. Überarbeitete PoCs
5. Überarbeitete Webseite gegen mobile App
6. Use Cases
7. Anwendungslogik
8. Anwendungsfalldiagramm
9. Aktivitätsdiagramme
10. Rapid Prototype User Journey
11. Umsetzung des Systems im Code - Implementierung
12. Durchgeführte PoCs

## Überarbeitetes Domänenmodell



Das Domänenmodell wurde noch einmal verbessert und angepasst.

Für eine bessere

Darstellung: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_lbragimova\\_Benaggoun\\_U%CC%88berarbeitetes\\_Doma%CC%88nenmodell.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_lbragimova_Benaggoun_U%CC%88berarbeitetes_Doma%CC%88nenmodell.pdf)

## Überarbeitete Stakeholder Tabelle

Bezeichnung	Einzelperson/Organisation	Bezug zum System (Anrecht, Anteil, Anspruch, Interesse)	System/Merkmale	Erfordernisse/Erwartungen
Jede Person, die gerne eine neue Gewohnheit etablieren möchte.	Einzelperson (Privatperson) und Organisationen	<p><b>Anrecht:</b> Die Person hat das Recht oder den Anspruch darauf, ihr Leben zu verbessern oder ihre Ziele zu erreichen, indem sie durch das System eine neue Gewohnheit in ihrem Leben einführt. Es deutet darauf hin, dass sie das Recht hat, positive Veränderungen in ihrem Leben vorzunehmen.</p> <p><b>Anteil:</b> Personen haben einen Anteil an einem neuen Gewohnheitssystem, indem sie dazu beitragen. Dieser Beitrag erfordert natürlich Zeit, Energie oder andere Ressourcen, die investiert werden, und ermöglicht es ihnen somit, einen persönlichen Anteil an den Ergebnissen des Systems zu haben.</p> <p><b>Anspruch:</b> Der Anspruch von Personen, die eine neue Gewohnheit entwickeln wollen, besteht darin, dies als etwas zu betrachten, das ihnen zusteht. Dies gilt in mehreren Fällen, zum Beispiel für Personen, die von einer Krankheit betroffen sind und den Anspruch haben, ein System zu haben, das ihnen hilft.</p> <p><b>Interesse:</b> Die Personen zeigen Interesse am System, da sie das Gefühl haben, etwas Neues zu lernen oder neues Wissen zu erwerben. Dieses Interesse kann als Initiative dienen, um die Motivation aufrechtzuerhalten und die Herausforderungen bei der Einführung einer Gewohnheit zu überwinden.</p>	<ul style="list-style-type: none"> <li>Entwicklung neuer Gewohnheiten und persönlichen Weiterentwicklung</li> <li>bessere Zeitmanagement zur Planung und Organisation von Aufgaben und Lernaktivitäten</li> <li>Fortschrittsverfolgung der Gewohnheit</li> <li>Motivationsanreize</li> <li>Selbstreflexion</li> <li>Flexibilität und Mobilität</li> <li>Unterstützung der Nutzer bei der mentalen Gesundheit</li> <li>Erinnerungen (Benachrichtigung) für die Medikamenteneinnahme etc.</li> <li>Anpassung der Schriftgröße und klare visuelle Darstellungen</li> <li>Unterstützung der Gesundheitsüberwachung</li> <li>Effektive Planung und Organisation von Aufgaben, Anpassungsfähigkeit an die Terminen und Projekten</li> <li>Einfluss Profit</li> <li>Kollaborationsplattform</li> <li>Erwerb von Nutzer(Kunden)</li> <li>Informationen weitergeben an Nutzer und dessen Interesse für Nutzerquoten nutzen / gewinnen</li> <li>Verbesserung von Lerngewohnheiten</li> </ul>	<ul style="list-style-type: none"> <li>Effektive Planungsfunktionen besitzen, um seine Gewohnheiten oder sein Zeitmanagement zu organisieren oder zu verbessern.</li> <li>Belohnungssysteme, Challenges, Inspirationsinhalte oder ähnliches anbieten um die Motivation aufrecht zuhalten und zu fördern.</li> <li>Zuverlässigkeit</li> <li>Persönliche Weiterentwicklung</li> <li>Support</li> <li>Anpassungsfähigkeit an die Einzelperson</li> <li>Sicherheit</li> <li>Benutzerfreundlichkeit</li> <li>Flexibilität und Mobilität</li> <li>Einfacher Zugang zu Support</li> <li>Work-Life-Balance</li> <li>Teambuilding und Zusammenarbeit</li> <li>Gesundheitsfördernd (Mentale und Physische Gesundheit)</li> <li>Effizienzsteigerung</li> <li>Benachrichtigung als Erinnerung</li> </ul>

4

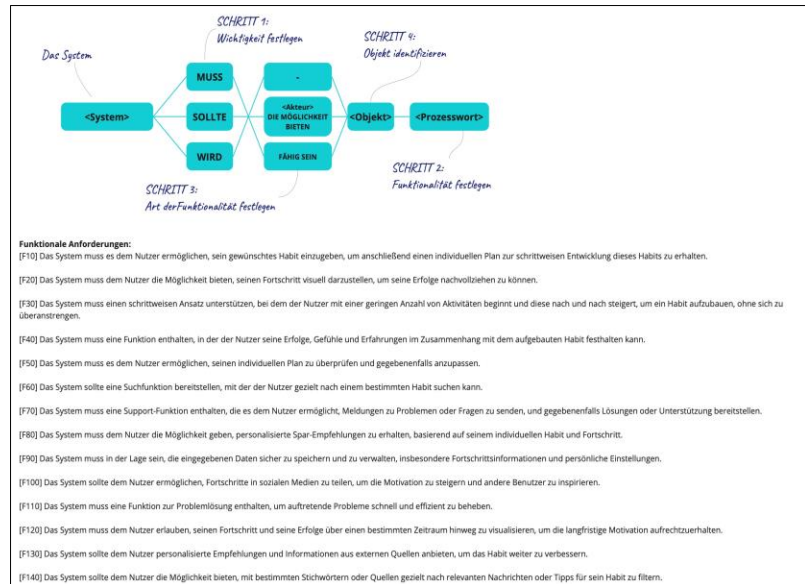
Die Stakeholder-Tabelle wurde ebenfalls überarbeitet. Die Unterteilung in Primäre, Sekundäre und Tertiäre Stakeholder wurde entfernt. Wir haben uns nach dem Feedback des zweiten Audits orientiert und eine Tabelle erstellt für Menschen die gerne Eine neue Gewohheit entwickeln möchten und die spalten und zeilen angepasst und erläutert.

Für eine bessere qualität der

Tabelle: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_Ibragimova\\_Benaggoun\\_U%CC%88berarbeitete\\_Stakeholdertabelle\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_Ibragimova_Benaggoun_U%CC%88berarbeitete_Stakeholdertabelle_.pdf)

# Anforderungen

## Funktionale Anforderungen



Die Anforderungen, Nicht – funktionalen Anforderungen und Erfordernisse wurde auch verbessert und angepasst.

Hier finden sie die

Anforderungen: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_Ibragimova\\_Benaggoune\\_U%CC%88bearbeitetes\\_Anforderungen.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_Ibragimova_Benaggoune_U%CC%88bearbeitetes_Anforderungen.pdf)

# Anforderungen

Funktionale - Anforderungen

1. Das System muss benutzerfreundlich und intuitiv gestaltet sein, um eine einfache Navigation und Nutzung zu gewährleisten.
2. Die Sicherheit der im System gespeicherten Daten, insbesondere persönlicher Informationen und Journal-Einträge, muss gewährleistet sein.
3. Das System muss auf verschiedenen Plattformen (iOS, Android) und Geräten nutzbar sein, um eine breite Zugänglichkeit sicherzustellen.
4. Das Antwortzeiten der System sollen angemessen sein, um eine reibungslose Benutzererfahrung zu gewährleisten.
5. Das System sollte ressourcenschonend sein und den Energieverbrauch des Geräts minimieren.
6. Das System muss barrierefrei sein, um Menschen mit verschiedenen Einschränkungen die Nutzung zu ermöglichen.
7. Regelmäßige Software-Updates sollten verfügbar sein, um Fehler zu beheben und die Funktionalität des System zu verbessern.
8. Das System sollte Datenschutzrichtlinien transparent kommunizieren und den Datenschutzgesetzen entsprechen.

6

Für eine bessere

Darstellung: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_lbragimova\\_Benaggoune\\_U%CC%88bearbeitete\\_Funktionale\\_Anforderungen.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_lbragimova_Benaggoune_U%CC%88bearbeitete_Funktionale_Anforderungen.pdf)

## Erfordernisse

**Als <spezifischer Benutzer> muss man <X> [wissen/verfügbar haben], um <Y> [entscheiden/tun] zu können.**

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um eine neue Gewohnheit etablieren zu können (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um für seine gewünschte erlernte Gewohnheit einen individuellen Zeitplan erstellen können.

Als Einzelperson muss man Zugang zu einer Plattform haben, die leicht zu navigieren ist, (X) um seine Habits nach zu verfolgen und durchführen zu können (Y).

Als Einzelperson benötigt man Zugang zu einer Plattform (X), um Erinnerungen und Benachrichtigungen zu erhalten, um seine Gewohnheiten durchzuführen können. (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um seine Fortschritte anhand von Belohnungssystemen oder Ähnlichem einsehen zu können (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um berufliche Uhrzeiten und Arbeitstage im seinem Zeitplan zu integrieren und seine Gewohnheitsentwicklung zeitgerecht verfolgen zu können (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um seine eigene persönliche Weiterentwicklung vorantreiben zu können (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um Support am System anzufragen (Y).

Als Einzelperson muss man Zugang zu einer Plattform haben (X), um gezielte Gewohnheiten für seine verbesserte Gesundheit (Mentale und Physische Gesundheit) zu entwickeln und umzusetzen (Y).

Als Einzelperson/Mehrpersonen muss man Zugang zu einer Plattform haben (X), um gemeinsam mit anderen Nutzern Gewohnheiten zu entwickeln und sich gegenseitig zu unterstützen (Y).

7

Die überarbeiteten Erfordernisse finden sie

hier: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_Ibragimova\\_Benaggoune\\_U%CC%88berarbeitete\\_Erfordernisse.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_Ibragimova_Benaggoune_U%CC%88berarbeitete_Erfordernisse.pdf)

# Überarbeitete PoCs

## Proof of Concept: Hauptseite und Features

---

### Beschreibung:

Die Startseite zeigt alle, im Rahmen dieses Projektes wichtigen Features. Die wichtigen Features sind Schedule / Calendar und Journaling. Alle weiteren der insgesamt 4 Hauptfeatures sind anklickbar, werden aber nicht weiter ausgeführt (es wird lediglich an die Stelle mit einem Platzhalter gesprungen). Die Hauptfeatures befinden sich in der Navigationsleiste im Header. Nach klicken soll zu dem geklickten Feature gesprungen werden.

### Exit-Kriterien:

Die für dieses Projekt wichtigen Features werden angezeigt, sind anklickbar und springen zu den gewünschten Features nach anklicken.

### Fail-Kriterien:

Die für dieses Projekt wichtigen Features werden weder angezeigt noch sind diese anklickbar. Nach klicken folgt kein Sprung an das gewünschte Feature.

### Fallbacks:

Integrieren von zusätzlichen internen Verlinkungen direkt im Seiteninhalt. Beispielsweise kann man am Anfang der Seite eine Übersicht mit Links zu den Hauptfeatures platzieren.

Die PoCs wurden Anhand des Feedbacks verbessert. Es folgen die realisierbaren und durchführbaren Funktionen unserer Systemlösung zum jetzigen Punkt.



# Überarbeitete PoCs

## Proof of Concept: Schedule / Calendar

---

### Beschreibung:

Die Funktion Schedule / Calendar zeigt die erstellten Zeitpläne in Form eines Kalenders. Hier können die Gewohnheiten (Habits) beliebig (per Drag-and-Drop) auf andere Tage verschoben werden. Außerdem können zusätzliche Einträge gesetzt werden.

### Exit-Kriterien:

Schedule / Calendar Zeitpläne werden erstellt. Alle Gewohnheiten (Habits) werden innerhalb eines Tages angezeigt. Gewohnheiten (Habits) können auf andere Tage verschoben werden. Zusätzliche Einträge können gesetzt werden.

### Fail-Kriterien:

In Schedule / Calendar werden die erstellten Zeitpläne nicht übernommen.  
Es werden nicht alle Gewohnheiten (Habits) an den gesetzten Tagen gezeigt.  
Der Nutzer kann die Gewohnheiten (Habits) nicht auf andere Tage verschieben und auch keine zusätzlichen Einträge erstellen.

### Fallbacks:

Eine manuelle Eingabemöglichkeit für den Nutzer bereitstellen, sodass die Zeitpläne auch innerhalb der Schedule / Calendar Funktion erstellt und die Gewohnheiten (Habits) verschoben werden können.

## Überarbeitete PoCs

### Proof of Concept: Journaling

---

#### Beschreibung:

Journaling bietet dem Nutzer die Möglichkeit die Gefühle, Erfahrungen und den Fortschritt in Einträgen festhalten (speichern). Alle Einträge werden in einer Übersicht dargestellt und können individuell bearbeitet oder gelöscht werden.

#### Exit-Kriterien:

Die neuen Einträge können gespeichert werden und werden auch in der Übersicht angezeigt. Einträge können bearbeitet oder gelöscht werden.

#### Fail-Kriterien:

Die Einträge werden nicht gespeichert und/oder nicht in der Übersicht angezeigt.  
Einträge können nach Erstellung nicht mehr bearbeitet oder gelöscht werden.

#### Fallbacks:

Eine Möglichkeit wäre, in Journaling ein einfaches Texteingabefeld bereit zu stellen.

10

Das speichern von Daten verlangt einen Speicherort, wie z.B. eine Datenbank. Das ist uns momentan noch nicht möglich abzubilden, da auch in Verbindung damit eine Login-Funktion geplant ist.

## Überarbeitete Webseite gegen mobile App

### Vorteile einer Website:

- Die Zeitpläne und Deadlines können besser und übersichtlicher gestaltet werden.
- Durch die Nutzung der Funktion "Journaling" hätte man eine verbesserte Klarheit, da man mehr platz hat am Laptop oder Pc.
- Die User experience angenehmer.
- Spricht mehrere Zielgruppen an (ältere menschen etc.)
- Große Nutzerreichweite möglich.
- Ein Produkt für alle Geräte und Betriebssysteme.
- Benachrichtigen auch in website möglich (Push-Mitteilungen) im Laptop, Pc und Handys.
- Unsere Ziele und Anforderungen passen besser zur einer Website.
- Aktualisierbarkeit in Echtzeit.
- Kosteneffizienz
- Unkompliziertes Teilen über Links in E-Mails, Chats und Social Media.
- Für die angestrebte Gewohnheitsentwicklung ist eine Website besser geeignet.
- Personalisierungsmöglichkeiten
- Programmierung aufwand ist geringer

### Nachteile einer Website:

- Abhängigkeit von Internetverbindung
- Nicht so komfortabel wie eine App
- Nicht mobil genug
- Die UX ist geräteunabhängig, was bedeutet, dass sie für keines der Geräte perfekt optimiert ist und die Interaktion mit dem Inhalt möglicherweise nicht so nutzerfreundlich ist wie bei einer App.

11

Nach gründlichen Überlegungen und basierend auf dem erhaltenen Feedback haben wir eine kurze Liste mit den Vor- und Nachteilen einer Website erstellt. Uns wurde empfohlen, erneut zu überdenken, ob eine Website möglicherweise am besten zu uns passt. Es entstand erneut eine Diskussion, aber letztendlich haben wir uns doch dafür entschieden, eine Website zu erstellen. Die detaillierten Vorteile und Nachteile finden Sie im Audit 2 hier:

- App: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324\\_Gashi\\_Ibragimova\\_Benaggoun\\_Lo%CC%88sungsfindung\\_Warum\\_App.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324_Gashi_Ibragimova_Benaggoun_Lo%CC%88sungsfindung_Warum_App.pdf)
- Website: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324\\_Gashi\\_Ibragimova\\_Benaggoun\\_Lo%CC%88sungsfindung\\_Warum\\_Website.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324_Gashi_Ibragimova_Benaggoun_Lo%CC%88sungsfindung_Warum_Website.pdf)
- Bezogen auf unser System: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324\\_Gashi\\_Ibragimova\\_Benaggoun\\_Lo%CC%88sungsfindung\\_Warum\\_App%3AWebsite\\_Tabelle.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20/WS2324_Gashi_Ibragimova_Benaggoun_Lo%CC%88sungsfindung_Warum_App%3AWebsite_Tabelle.pdf)

## Use Case #1

**use case:** Verbesserung des Zeitmanagements und effektive Planung durchführen.

**actors:** Student(en)

**trigger:** Der Student möchte bis zur seiner Klausurphase anzugewöhnen mehr zu lernen.

**precondition:** Der Student hat Zugriff auf das Habiton-System und kann entweder ein Konto erstellen oder bereits über ein vorhandenes Konto verfügen.

**main flow:**

1. Der Student meldet sich im "Habiton" System an.
2. Der Student gibt das gewünschte Habit ein und startet den Habit-Zeitplan.
3. Das System erstellt einen Plan, der schrittweise kleine Ziele für die Gewohnheitsentwicklung festlegt. Zum Beispiel würde das System, am ersten Tag nur eine Stunde lernen planen, nach einer Woche zwei Stunden, und nach einem Monat 5 Stunden lernen.
4. Der Student führt die täglichen Gewohnheitsaktivitäten gemäß dem Plan durch.
5. Jeder erfolgreiche Tag schaltet einen Pixel des Badges frei, das den Fortschritt des Habits darstellt.
6. Der Student kann das erreichte Badge in der Achievements Ansicht des Systems sehen.

**postcondition:**

Der Student hat die gewünschte Gewohnheit entwickelt und kann das komplette Badge einsehen.

**exceptional flow:**

Das System konnte kein Zeitplan erstellen und keine Benachrichtigungen senden.

**postcondition:** Es wurde keine neue Gewohnheit erlernt.

**End:** Die Verbesserung des Zeitmanagements und effektive Planung wurden durchgeführt.

12

Da im zweiten Audit die Stakeholder eingeteilt waren und verschiedene Nutzer des Systems sowie deren Anforderungen und Erfordernisse berücksichtigt wurden, wurde uns empfohlen, Use Cases für die unterschiedlichen Nutzer des Systems und deren Anforderungen zu erstellen. Die Use Cases beziehen sich eher auf die Verwendung des Systems, die Abläufe der einzelnen Stakeholder und deren Anforderungen und Erfordernisse, und weniger auf die Funktionen selbst. Mit den Use Cases wollten wir einfach die User Experience etwas näher bringen. Die ganze pdf zu den Use Cases finden sie

hier: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_Ibragimova\\_Benaggoune\\_Use\\_Cases\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_Ibragimova_Benaggoune_Use_Cases_.pdf)

## Use Case #2

**use case:** Benachrichtigungen als Erinnerung erhalten.

**actors:** Person, die von einer Krankheit betroffen ist.

**trigger:** Die Personen möchte das System zu benutzen um sich erinnern zu lassen, wann er seine Tabletten einnehmen soll und es als Gewohnheit anzueignen.

**precondition:** Die betreffende Person kann auf das Habiton-System zugreifen und hat die Möglichkeit, entweder ein neues Konto zu erstellen oder sich über sein Konto anzumelden.

**main flow:**

1. Die Person meldet sich an.
2. Die Person erstellt seine gewünschte Gewohnheit und gibt an wann er daran erinnert werden möchte.
3. Das System erstellt einen angepassten Plan für die Gewohnheitsentwicklung.
4. Die Person startet dann seinen Zeitplan.
5. Die Person wird dann gemäß seines Zeitplans erinnert und kriegt seine Mitteilungen.
6. Jeder geschaffte Tag schaltet einen Pixel im Badge frei.
7. Der vollständige Badge wird in den Achievements angezeigt.

**postcondition:**

Die Person hat erfolgreich eine neue Gewohnheit entwickelt und kann den Fortschritt anhand der erreichten Badges nachvollziehen.

**exceptional flow:**

Das System war nicht in der Lage, einen Zeitplan zu erstellen und Benachrichtigungen zu senden.

**postcondition:** Es wurde keine neue Gewohnheit erlernt.

**End:** Person hat Benachrichtigungen als Erinnerung erhalten.

## Use Case #3

**use case:** Unterstützung der Persönlichen Weiterentwicklung

**actors:** Ältere Person

**trigger:** Die ältere Person möchte, eine neue Gewohnheit entwickeln.

**precondition:** Die Person hat Berechtigungen, das Habiton-System zu nutzen, und kann dabei wählen, ob sie ein Konto erstellen oder bereits über ein vorhandenes Konto verfügen möchte.

**main flow:**

1. Die Person meldet sich im System an.
2. Die Person gibt die neuen Gewohnheiten ein, nämlich täglich spazieren zu gehen.
3. Das System erstellt einen individuell Plan, für die Person. Hier würde das System z.B. eintragen, dass der Nutzer einmal am Tag nur 10 min spazieren gehen sollte, danach die Woche 25 min und dann am Ende des Monats 30 Minuten oder mehr.
4. Die ältere Person führt die täglichen Gewohnheitsaktivitäten gemäß dem Plan durch, indem sie jeden Tag spazieren geht.
5. Nach und nach wird ein Pixel freigeschaltet, bis das Badge komplett ist.
6. In der Achievements Ansicht kann der Nutzer sein Badge ansehen.

**postcondition:**

Die ältere Person hat sein Ziel erreicht und eine neue Gewohnheit etabliert.

**exceptional flow:**

Weder die Erstellung eines Zeitplans noch das Senden von Benachrichtigungen konnte durchgeführt werden.

**postcondition:** Es wurde keine neue Gewohnheit erlernt.

**End:** Unterstützung der Persönlichen Weiterentwicklung.

## Use Case #4

**use case:** Gewohnheiten aufbauen, die zu einer Work-Life-Balance passen.

**actors:** Berufstätige Person

**trigger:** Die Person will lernen zu häkeln nach der Arbeit vergisst es aber jedes Mal und schafft es nicht sich selbst zu organisieren, wegen Ihrer Arbeit.

**precondition:** Der Nutzer hat Zugriff auf das Habiton-System und kann dabei wählen, entweder ein neues Konto zu erstellen oder auf sein bestehendes Konto zuzugreifen.

**main flow:**

1. Die berufstätige Person meldet sich im System an.
2. Die Person erstellt eine Gewohnheit mit der Funktion "Create a Habit", nämlich das Erlernen des Häkelns. Die Person kann bei Erstellen des neuen Habits eingeben, wann seine Arbeitstage und Zeiten sind, so dass sich das System daran orientieren kann.
3. Der Plan wird erstellt und gestartet, die zeitlichen Einschränkungen der beruflichen Tätigkeit berücksichtigt.
4. Die Gewohnheitsaktivitäten werden durchgeführt.
5. Die Pixel werden immer nach den täglichen Aktivitäten freigeschaltet.
6. Das erreichte Badge, der sich dem Habit "nach der Arbeit häkeln" anpasst, wird in den Achievements angezeigt.

**alternative flow:**

- 4a. Die Person schafft es nicht Zeitgemäß den generierten Zeitplan nachzuverfolgen.
- 4b. Der Nutzer kann dann seinen Zeitplan bearbeiten und so anpassen, dass er immer noch innerhalb seines gewünschten Zeitraums liegt.
- 4c. Danach wird der Zeitplan wie gewohnt weitergeführt.

**postcondition:** Die Gewohnheit wurde erfolgreich erlernt.

**exceptional flow:** Die Fähigkeit des Systems zur Erstellung von Zeitplänen und zum Versenden von Benachrichtigungen ging nicht.

**postcondition:** Es wurde keine neue Gewohnheit erlernt.

**End:** Gewohnheit aufgebaut, die zu einer Work-Life-Balance passt.

## Use Case #5

**use case:** Gemeinsame Entwicklung einer Gewohnheit

**actors:** Lehrer/-in

**trigger:** Die Unterstufenklasse soll für das nächste Englischthema eine Lektüre lesen und ein Lesetagebuch führen. Eine Lehrerin nutzt das System Habiton, um einen perfekten Zeitplan für die Schüler zu erstellen und ihre Aufgaben zu organisieren.

**Precondition:**

Der Nutzer hat Zugriff auf das Habiton-System, kann bei Bedarf ein Konto erstellen oder verfügt bereits über eins, während die Schüler bereit sind, am gemeinsamen Habit teilzunehmen.

**main flow:**

1. Die Lehrerin meldet sich an.
2. Die Lehrerin gibt das gewünschte Habit ein, nämlich das gemeinsame Lesen eines Buches und das Ausfüllen eines Workbooks. Sie stellt es so ein, dass es zudem Alltag der Schülerinnen und Schüler passt und ihren Stundenplan berücksichtigt.
3. Das System generiert wie immer einen individuellen Zeitplan.
4. Die Lehrerin informiert die Schüler über das gemeinsame Habit und den entwickelten Plan.
5. Die Schüler führen die täglichen Gewohnheitsaktivitäten gemäß dem Plan durch, indem sie das Buch lesen und das Tagebuch ausfüllen.
6. Nachdem die Lehrerin kontrolliert ob jeder Schüler die Aufgabe erledigt hat, trägt es die Lehrerin im System ein.
7. Um den Fortschritt der Klasse zu visualisieren, wird ein Pixel freigestaltet.
8. Der erreichte Badge ist einsehbar.

**alternative flow:**

- 4a. Die Klasse schafft es nicht Zeitgemäß den generierten Zeitplan nachzuverfolgen.
- 4b. Der Nutzer kann dann seinen Zeitplan bearbeiten und so anpassen, dass er immer noch innerhalb seines gewünschten Zeitraums liegt.
- 4c. Danach wird der Zeitplan wie gewohnt weitergeführt.

**postcondition:**

Die Lehrerin und Schüler haben erfolgreich die Gewohnheit entwickelt.

**exceptional Flow:** Das Erstellen eines Zeitplans und das Senden von Benachrichtigungen waren nicht möglich.

**postcondition:** Es wurde keine neue Gewohnheit erlernt.

**End:** Gemeinsame Entwicklung einer Gewohnheit.



## Use Case #6

**use case:** Das entwickeln einer neuen Gewohnheit im Kontext der Team Building

**actors:**

Freiwilligenorganisation (Benutzer des Systems)  
Gruppen von Freiwilligen (Teilnehmer der Organisation)

**trigger:** Eine Freiwilligenorganisation entscheidet sich, im Altersheim ehrenamtlich zu arbeiten.

**precondition:** Der Leiter verfügt über Zugriff auf das Habiton-System und kann sich ein Konto erstellen oder hat bereits ein Konto. Die Gruppe ist bereit, an der gemeinsamen Gewohnheitsentwicklung teilzunehmen.

**main flow:**

1. Der Leiter meldet sich an.
2. Die Gewohnheit mit den gewünschten Zeiten und Tagen wird erstellt.
3. Das System generiert den Zeitplan und berücksichtigt dabei die angegebenen Zeiten der Gruppe.
4. Die Freiwilligen führen die täglichen Gewohnheitsaktivitäten durch.
6. Der Badge wird nach jedem erfolgreichen Tag vollständiger.
7. Die Achievements zeigen dann das vollständige Badge.

**Postcondition:**

Die Freiwilligenorganisation hat ihr Ziel erreicht.

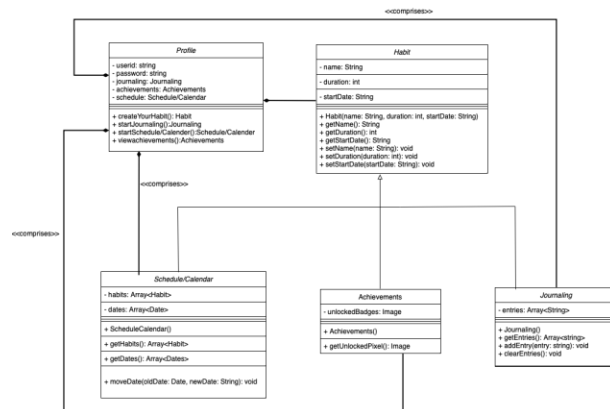
**exceptional flow:**

Es war dem System nicht möglich, einen Zeitplan zu generieren oder Benachrichtigungen zu versenden.

**postcondition:** Das Ziel wurde nicht erreicht.

**End:** Das entwickeln einer neuen Gewohnheit im Kontext der Team Building

# Anwendungslogik



18

Für die Darstellung der Anwendungslogik wurde ein Klassendiagramm entworfen. Dieses Diagramm zeigt die verschiedenen Funktionen des Systems und ihre gegenseitigen Beziehungen. Insbesondere zeigen die Funktionen "Habit", "Schedule/Calendar", "Achievements" und "Journaling" eine Beziehung zum Profil, da sie von diesem abhängen. Diese Funktionen können nur existieren, wenn der Nutzer ein aktives Profil hat.

Es ist wichtig zu betonen, dass "Schedule/Calendar", "Achievements" und "Journaling" eine Generalisierungsbeziehung zu "Habit" aufweisen. Dies bedeutet, dass sie auf einem bestehenden Habit aufbauen und erst genutzt werden können, wenn ein Habit erstellt wurde und aktiv ist. Für eine bessere

Darstellung: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20WS2324\\_Gashi\\_Ibragimova\\_Benaggoune\\_Anwendungslogik\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20WS2324_Gashi_Ibragimova_Benaggoune_Anwendungslogik_.pdf)

# Anwendungsfalldiagramm

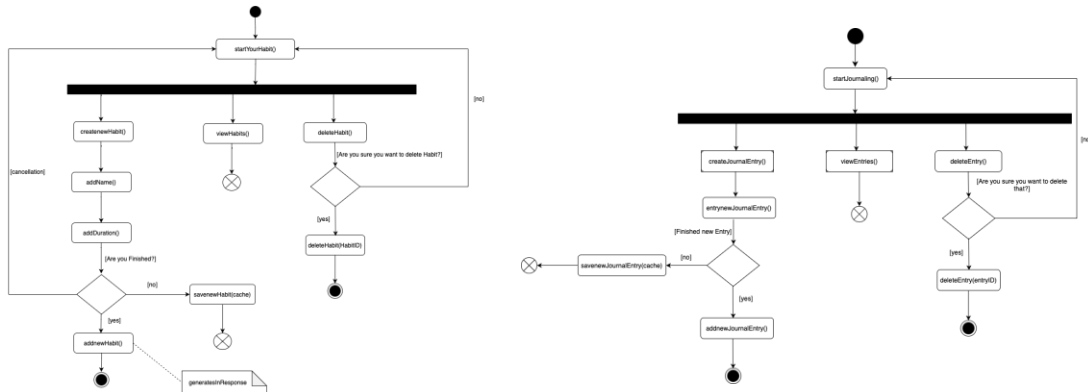


19

Um einen Groben Ablauf des Users zu zeigen, wurde ein Anwendungsfalldiagramm des ganzen Systems erstellt. Es wird auch gezeigt wie das System auf die Befehle des Users reagiert. Für eine bessere

Darstellung: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20WS2324\\_Gashi\\_Ibragimova\\_Benaggoune\\_Anwendungsfalldiagramm\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%20WS2324_Gashi_Ibragimova_Benaggoune_Anwendungsfalldiagramm_.pdf)

# Aktivitätsdiagramme



20

Um die Systemlogik klarer zu veranschaulichen und zu beschreiben, wurden zwei Aktivitätsdiagramme erstellt. Der erste zeigt die Prozesse, die in "Your Habit" möglich sind. Dieser Ablauf umfasst die Erstellung einer neuen Gewohnheit und zeigt, wie das System auf die Eingaben des Nutzers reagiert. Der zweite Ablauf in "Your Habit" konzentriert sich auf das Ansehen der Gewohnheiten. Da hier die Betrachtung der eigenen Gewohnheiten im Vordergrund steht, endet der Ablauf nach diesem Schritt. Der letzte Ablauf beschreibt das Löschen einer Gewohnheit. Hier wird einfach dargestellt, wie das System auf bestimmte Befehle reagiert, wenn eine Gewohnheit gelöscht wird.

Im zweiten Aktivitätsdiagramm wird dasselbe Prinzip angewendet, jedoch für die Funktion "Journaling". Aufgrund der einfachen Struktur unseres Systems ähneln sich die Abläufe der Funktionen.

Aktivitätsdiagramm

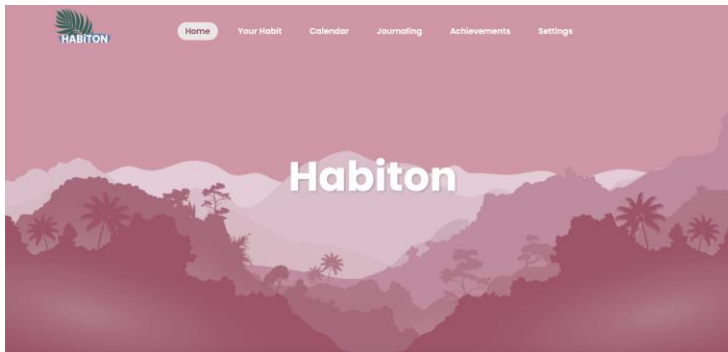
1: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_lbragimova\\_Benaggoune\\_Aktivita%C3%9Ctsdiagramm\\_2\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_lbragimova_Benaggoune_Aktivita%C3%9Ctsdiagramm_2_.pdf)

Aktivitätsdiagramm

2: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_lbragimova\\_Benaggoune\\_Aktivita%C3%9Ctsdiagramm\\_2\\_.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_lbragimova_Benaggoune_Aktivita%C3%9Ctsdiagramm_2_.pdf)

[ne Aktivita%CC%88tsdiagramm 1 .pdf](#)

# Rapid Prototype User Journey



1. Als Erstes gelangt man auf die Startseite der Website, auf der man zwischen verschiedenen Funktionen auswählen kann.

21

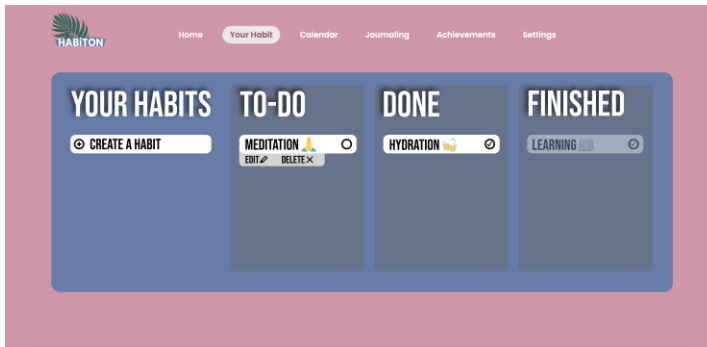
Für das Design haben wir uns entschieden, die Funktion "Your Habit" darzustellen, da wir es leider nicht geschafft haben, diese Funktion in den vertikalen Prototyp zu integrieren. Bei der Gestaltung haben wir uns bewusst für ein einfaches und modernes Design entschieden. Unser Ziel war es, sicherzustellen, dass Benutzer das System leicht verstehen und nutzen können.

Wir haben darauf geachtet, nicht zu viele Symbole zu verwenden, da dies vom wesentlichen Prinzip der Gewohnheitsentwicklung und des Schreibens des Journals ablenken würde. Die klare und minimalistische Gestaltung soll die Benutzerfreundlichkeit fördern und sicherstellen, dass die Nutzer ohne Ablenkungen durch das System navigieren können.

Die Designs in besseren

Qualität: [https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324\\_Gashi\\_Ibragimova\\_Benaggoun\\_Website\\_Design\\_Create\\_a\\_Habit.pdf](https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/blob/main/Artefakte/Artefakte%20f%C3%BCr%20Audit%203/WS2324_Gashi_Ibragimova_Benaggoun_Website_Design_Create_a_Habit.pdf)

## Rapid Prototype User Journey



2. Wenn der Nutzer auf die Funktion "Your Habit" klickt, werden ihm zunächst seine Gewohnheiten angezeigt. Der Nutzer kann dann entscheiden, ob er seine To-Do-Liste aktualisieren möchte oder ein neues Habit erstellen möchte.

To-Do zeigt die zu erledigenden Aufgaben und allgemein alle Habits. Done zeigt die bereits erledigten Aufgaben. Finished zeigt die Abgeschlossenen Gewohnheiten, also in dem Beispiel ist Learning zur Gewohnheit geworden.

## Rapid Prototype User Journey

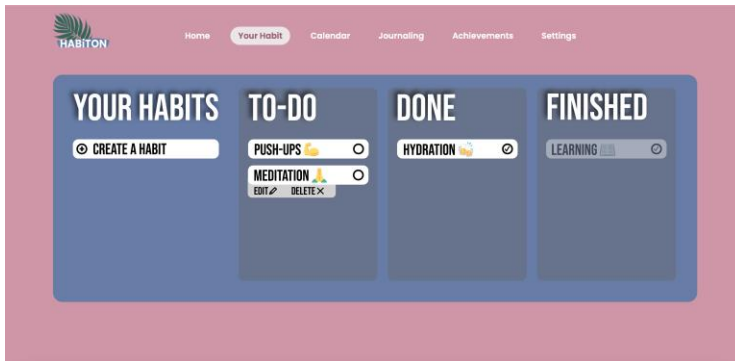
The screenshot shows the 'CREATE A HABIT' interface of the HABITON app. The app has a pink header with the logo and navigation links: Home, Your Habit (active), Calendar, Journaling, Achievements, and Settings. The main form is a blue box with the following fields:

- NAME OF HABIT:** PUSH-UPS 🍌 (with an information icon)
- PERIOD:** FROM: 22.01.2024 TO: 22.01.2025 (with an information icon)
- DAYS:** MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY (with an information icon)
- TIME:** FROM: 17:30 TO: 18:00 (with an information icon)

3. Wenn der Nutzer dann eine neue Gewohnheit erstellt, werden ihm verschiedene Kriterien angezeigt, die er ausfüllen muss, um die neue Gewohnheit zu erstellen. Für die Journey wurden die Kriterien im Design bereits einmal ausgefüllt.



## Rapid Prototype User Journey



4. Schließlich kann man sehen, dass die neue Gewohnheit erfolgreich integriert wurde.

# Umsetzung des Systems im Code - Implementierung

---

## Grundsätzliches:

- Entwicklung erfolgt durch **Visual Studio Code**
- Code HTML
- Code JavaScript
- Code CSS

25

Die Implementierung des vertikaler Rapid Prototype finden sie hier: <https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/tree/Webcode>

Der Branch ist nicht Main, sondern Webcode.

# Durchgeführte PoCs

---

## Durchgeführte PoC: Hauptseite und Features

### Beschreibung:

Die Hauptseite zeigt beim Scrollen den Willkommenstext. Die für diesen Teil des Projektes wichtigen Funktionen sind korrekt verlinkt und funktionieren.

Die Verlinkung zum

Code: <https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/tree/Webcode>

Der Code wurde mithilfe dieses Tutorials

erstellt: <https://www.youtube.com/watch?v=kmM6mqvnxcs&t=494s>

26

Wichtige Funktionen (vorläufig): Schedule / Calendar und Journaling  
Your Habits und Achiements folgen in Zukunft.

# Durchgeführte PoCs

---

## **Durchgeführte PoC: Schedule / Calendar**

Beschreibung:

Im Schedule / Calendar wird der zuvor erstellte Zeitplan nicht angezeigt, da im Rahmen dieses Projekt bislang die Funktion noch nicht implementiert wurde. Hier greift also nun die Fallbackoption, Zeitpläne innerhalb der Schedule / Calendar Funktion zu erstellen und die Gewohnheiten zu bearbeiten.

Die Verlinkung

zum Code: <https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/tree/Webcode>

Der Code wurde mithilfe dieses Tutorials erstellt: <https://www.youtube.com/watch?v=r1devGCrm2Y>

# Durchgeführte PoCs

---

## Durchgeführte PoC: Journaling

### Beschreibung:

Im Bereich Journaling können die Gedanken, Gefühle, Erfahrung und der Fortschritt in Einträgen festgehalten werden. Nach dem man in den Bereich Journaling gelangt, erhält man die Möglichkeit einen neuen Eintrag zu erstellen. Ein Eintrag besteht aus zwei Eingabefeldern, Name und Inhalt. Das Datum wird beim Erstellen bislang noch nicht automatisch festgehalten. Die Anzahl der Wörter im Namen und im Inhalt sind unbegrenzt. Zudem ist die Verwendung von Umlauten und Emojis durch UTF-8 Zeichencodierung gewährleistet. Die Einträge werden nach Erstellungsdatum sortiert und im Bereich Journaling aufgelistet. Diese können im nachhinein bearbeitet und gelöscht werden.

Die Verlinkung zum

Code: <https://github.com/sekbn/EPWS2324BenaggounelbragimovaGashi/tree/Webcode>

Der Code wurde mithilfe dieser Webseite und deren Tutorials erstellt: <https://www.w3schools.com/>

28

Die Einträge lassen sich momentan noch nicht langfristig speichern, da es noch keine Möglichkeit gibt Nutzer anzulegen.

## Projektplan für Audit 4

---

- Poster
- Funktionaler Prototyp
- Kritische Reflektion
- Fazit

Danke für Eure Aufmerksamkeit

---