

# **SMART**

## Langfristiges Ziel

Jeder soll von zuhause aus wählen können.

## Spezifisch

Jeder soll auch von zuhause aus wählen können, um das Wählen zu erleichtern und die Wahlbeteiligung zu steigern.

## Messbar

Eine Wahlbeteiligung von 75 % oder höher bei den nächsten Wahlen  
Briefwahlanteil soll dadurch verringert werden

## Akzeptiert/attraktiv

gutes Design  
gute Useability  
sicher (Einhaltung der 5 Wahlgrundsätze)  
allgemein, unmittelbar, frei, gleich und geheim

## Realistisch

Umsetzung in Form eines Programmes → Ausbau zu einer Webanwendung möglich  
Eindeutige Identifizierung der Person.

## Terminiert

Bis zum Abgabedatum.  
Ende 4. Semester. (26.05.2023)

## Zusammengefasst

Bis zum 26.05.2023 soll jeder leicht und sicher von zuhause aus wählen können, um die Wahlbeteiligung zu steigern.

## **Persona Definitionen**

<https://www.justinmind.com/blog/user-persona-templates/>

- Motivation
- Goals
- Frustrations
- 

### Stakeholder

- Bürger
- Staat
- Entwickler

## Persona Staat:

- männlich, weiß, 50 Jahre
- NRW Dortmund
- unglücklich verheiratet
- Beruf: Anwalt für Familienrecht
- Ziele:
  - Frührente
  - gute Reputation gegenüber seiner Partei und der Wählerschaft
  - Höhere Wahlbeteiligung
- Frustration:
  - integer
  - nicht auswertbare Wahlzettel
  - Lobbyismus ist geil

## Persona WählerInnen mittel:

- weiblich, 30 Jahre, weiß
- München
- ledig
- Beruf: bachelor of Arts BWL Schwerpunkt HR, aktuell Gap-Year
- Ziele:
  - You only live once
  - Frühe Altersvorsorge
  -
- Frustration:
  - Probleme mit Technik
  - Wählen dauert zu lange

## Persona Wähler\*innen jung:

- diverse, 20 Jahre, dreadlocks
- Berlin
- #singlelikeapingle
- Beruf: arbeitslos, Aktivistin, sitzt immer wieder für festkleben auf Straßen ein
- Ziele:
  - CO2 Ausstoß minimieren und Klimawandel stoppen
  - immer auf dem neuesten Stand rein
- Frustration:
  - Politik
  - Industrie
  - Autofahrer\*innen

## Persona Wähler\*innen alt:

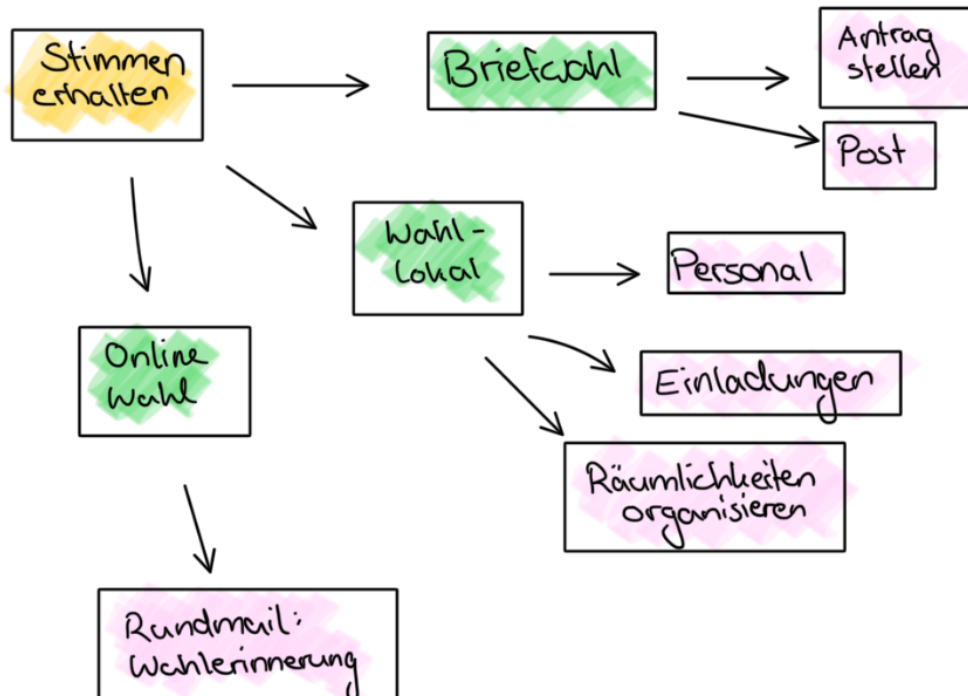
- männlich, 50 Jahre, weiß
- Nusplingen
- verheiratet
- Beruf: Ingenieur bei Zeiss
- Ziele:
  - Altersvorsorge
  - E-Bikes
  - Wohnmobile
- Frustration:
  - Inflation
  - Erhöhung des Renteneintritts
  - Baustellen

## Persona Entwickler:

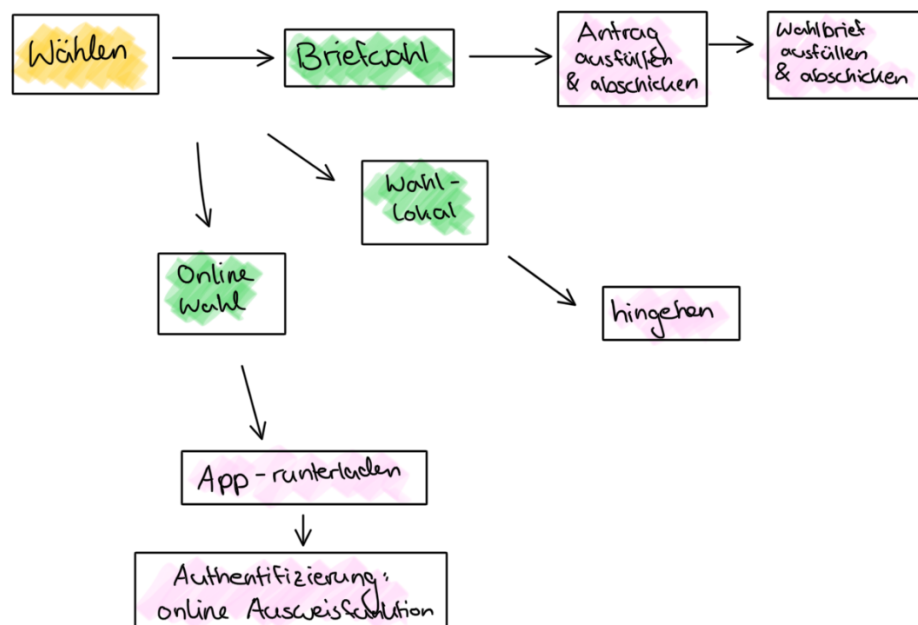
- männlich, 35 Jahre, Bildschirmbraun
- Stuttgart
- ledig
- Beruf: Entwickler
- Ziele
  - Einfache Umsetzung
  - Leicht wartbar
- Frustration:
  - fehlendes Semikolon im Code
  - Single
  - nervige Mutter (Weil noch Zuhause wohnt)

# Persona Diagramme

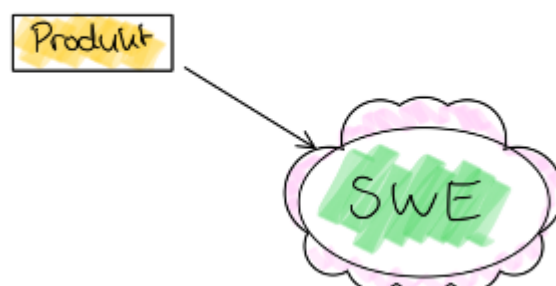
Persona Staat



Wähler\*innen



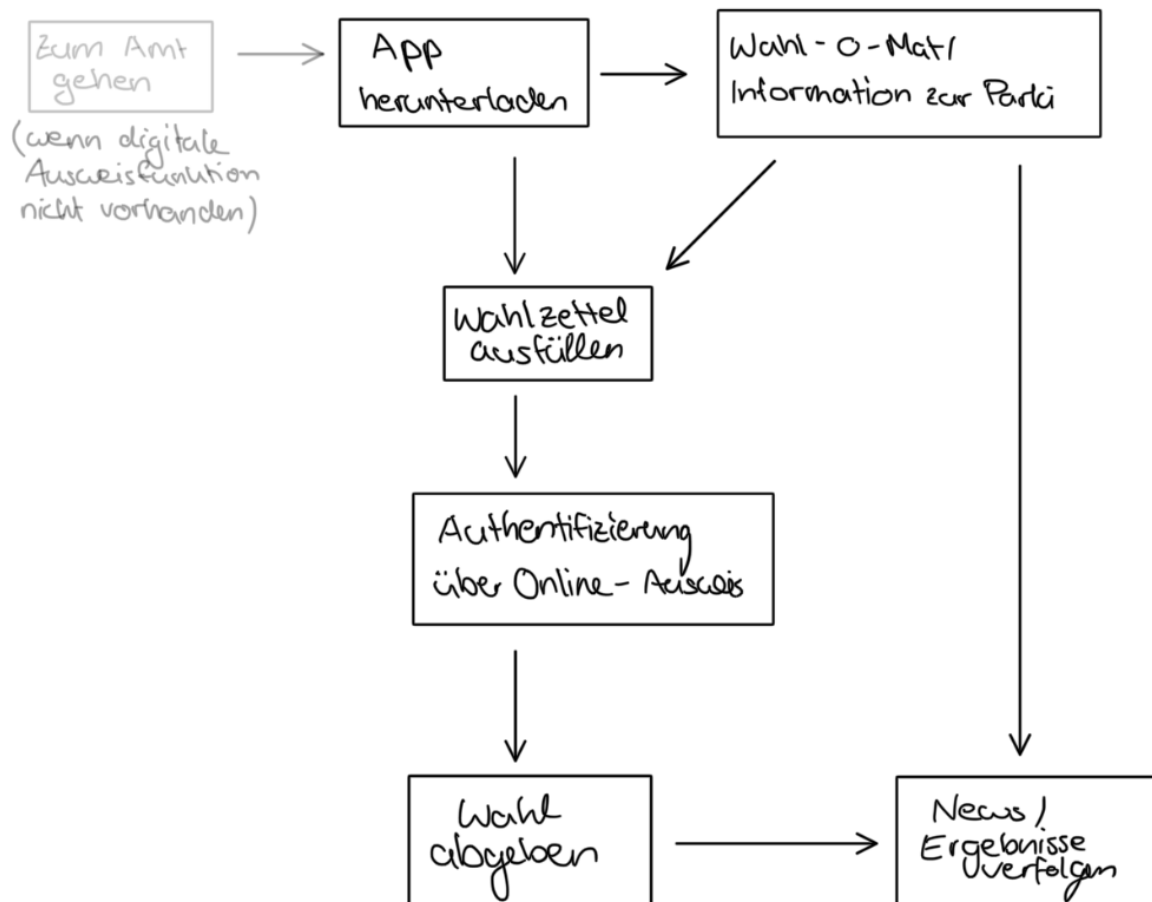
Entwickler



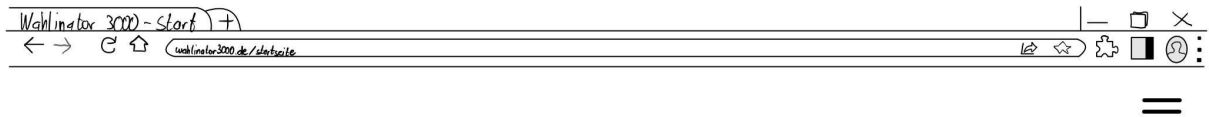
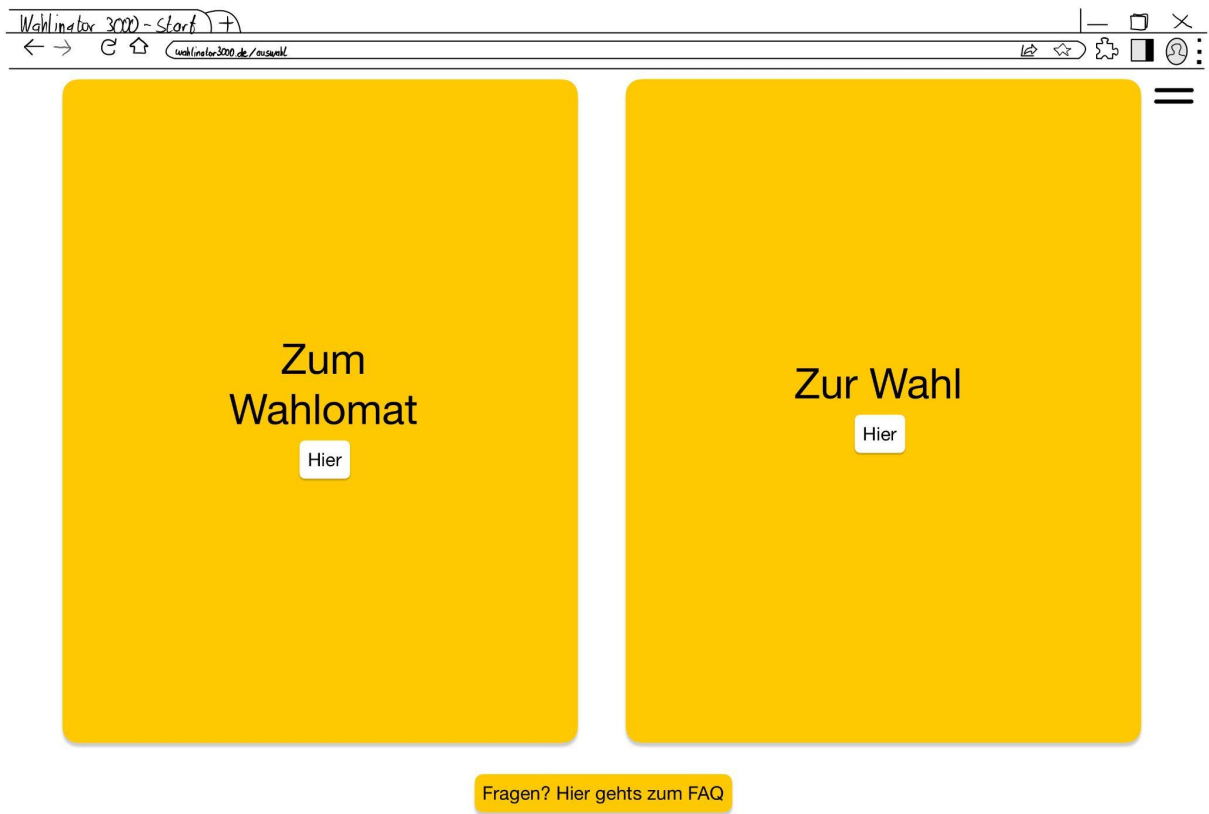
## Wie können wir...?

- Das wählen vereinfachen (z.B. Die App möglichst ansprechend für alle Anwender gestalten)
- Die Person feststellen/Authentifizierung
- Datenschutz/Sicherheit
- Leute anwerben/ auffordern & Marketing

## Goldener Pfad



## Design-Konzepte



Wahlinator 3000<sup>®</sup>  
Bundestagswahl 2025

START

## Häufig gestellte Fragen



Frage 1: Hier könnte ihre Frage 1 stehen? Es ist die am häufigsten gestellte Frage.

Antwort

Frage 2: Hier könnte ihre Frage 2 stehen? Es ist die am zweit häufigst gestellte Frage.

Antwort

Frage 3: Hier könnte ihre Frage 2 stehen? Es ist die am dritt häufigst gestellte Frage.

Antwort

Frage 1: Hier könnte ihre Frage 1 stehen? Es ist die am häufigsten gestellte Frage.

Antwort

Frage 2: Hier könnte ihre Frage 2 stehen? Es ist die am zweit häufigst gestellte Frage.

Antwort

Frage 3: Hier könnte ihre Frage 2 stehen? Es ist die am dritt häufigst gestellte Frage.

Antwort

Frage 1: Hier könnte ihre Frage 1 stehen? Es ist die am häufigsten gestellte Frage.

Antwort

Frage 2: Hier könnte ihre Frage 2 stehen? Es ist die am zweit häufigst gestellte Frage.

Antwort

Frage 3: Hier könnte ihre Frage 2 stehen? Es ist die am dritt häufigst gestellte Frage.

Antwort

## Anmeldung - Digitaler Ausweis



Wie geht das?

Ausweis-ID

Passwort

Anmelden



## Erststimme

Partei 1: Direktmandat 1 ☐

Partei 2: Direktmandat 2 ☐

Partei 3: Direktmandat 3 ☐

Partei 4: Direktmandat 4 ☐

Partei 5: Direktmandat 5 ☐

## Zweitstimme

Partei 1: Parteislogan ☐

Partei 2: Parteislogan ☐

Partei 3: Parteislogan ☐

Partei 4: Parteislogan ☐

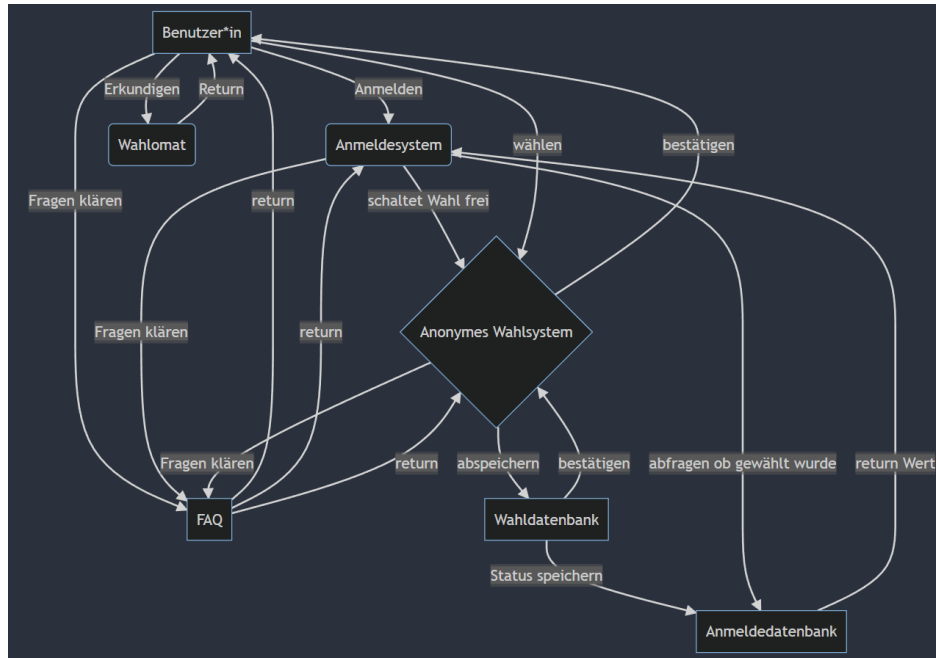
Partei 5: Parteislogan ☐

[Fragen? Hier gehts zum FAQ](#)

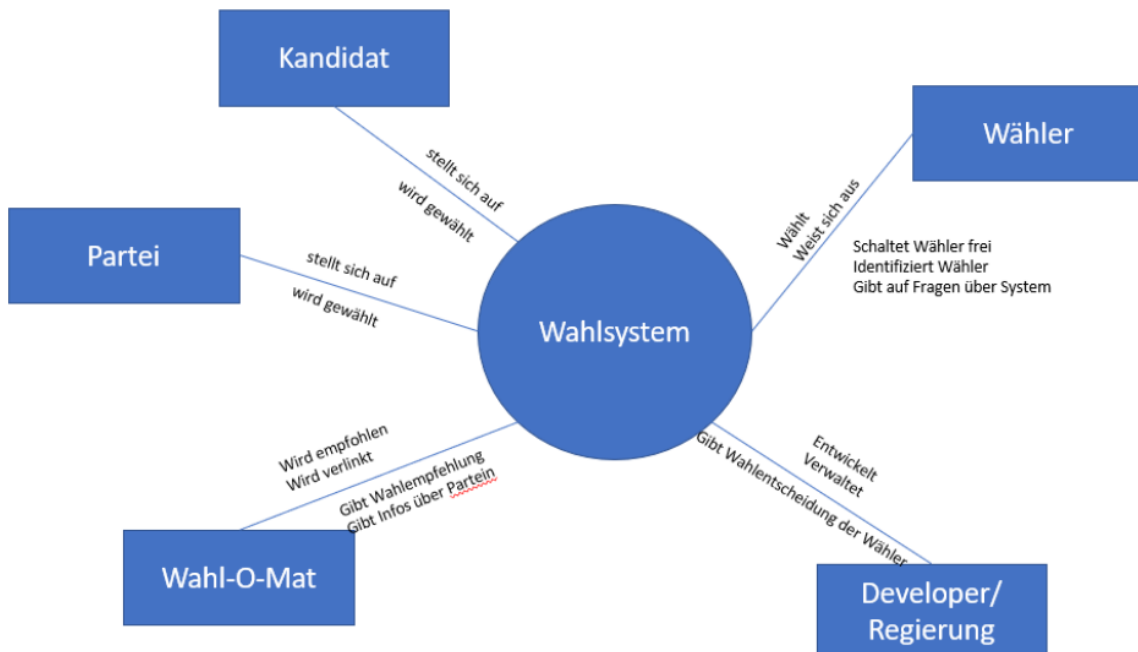
[Stimme abgeben](#)

# UML

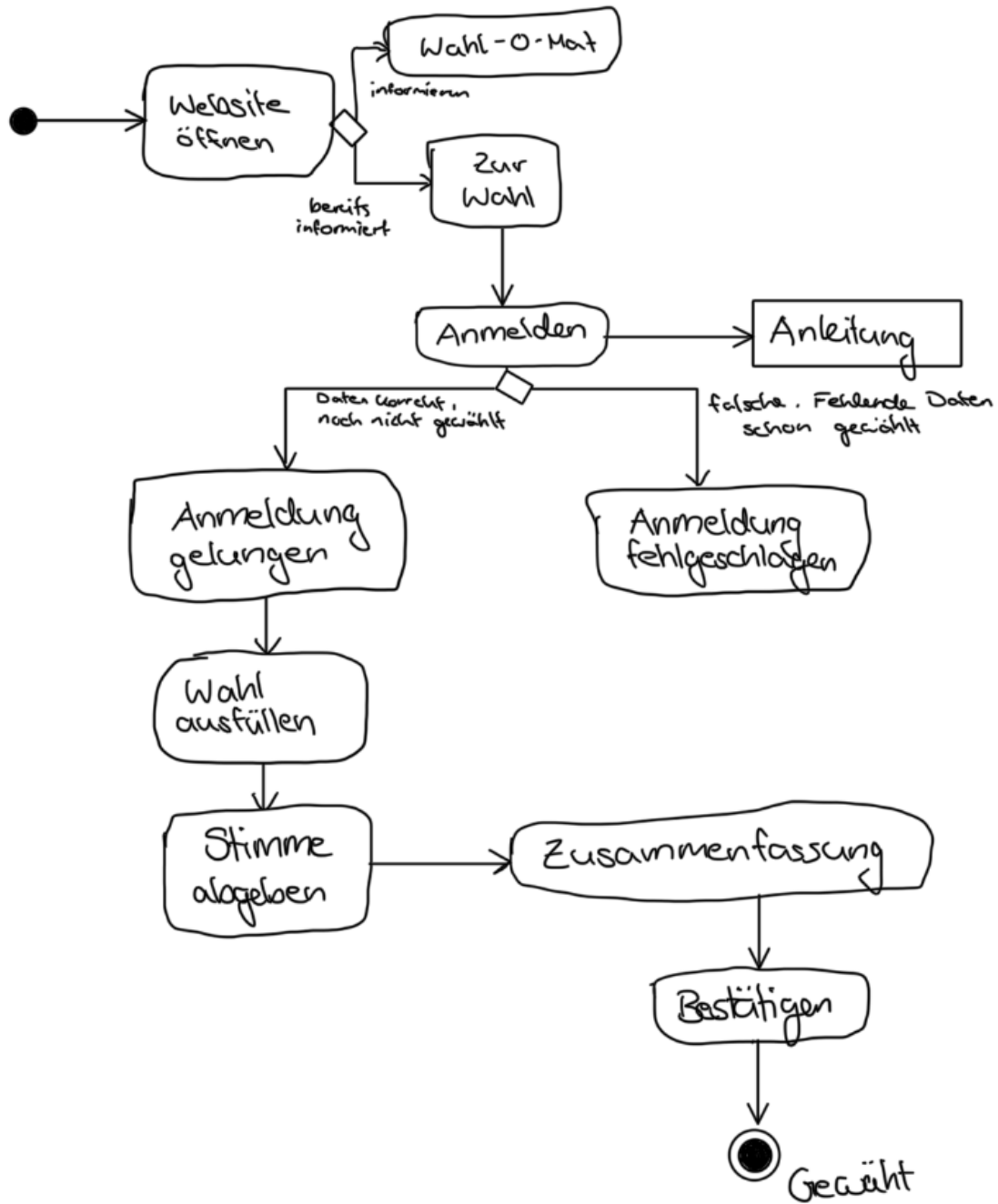
## Klassendiagramm (noch falsch)



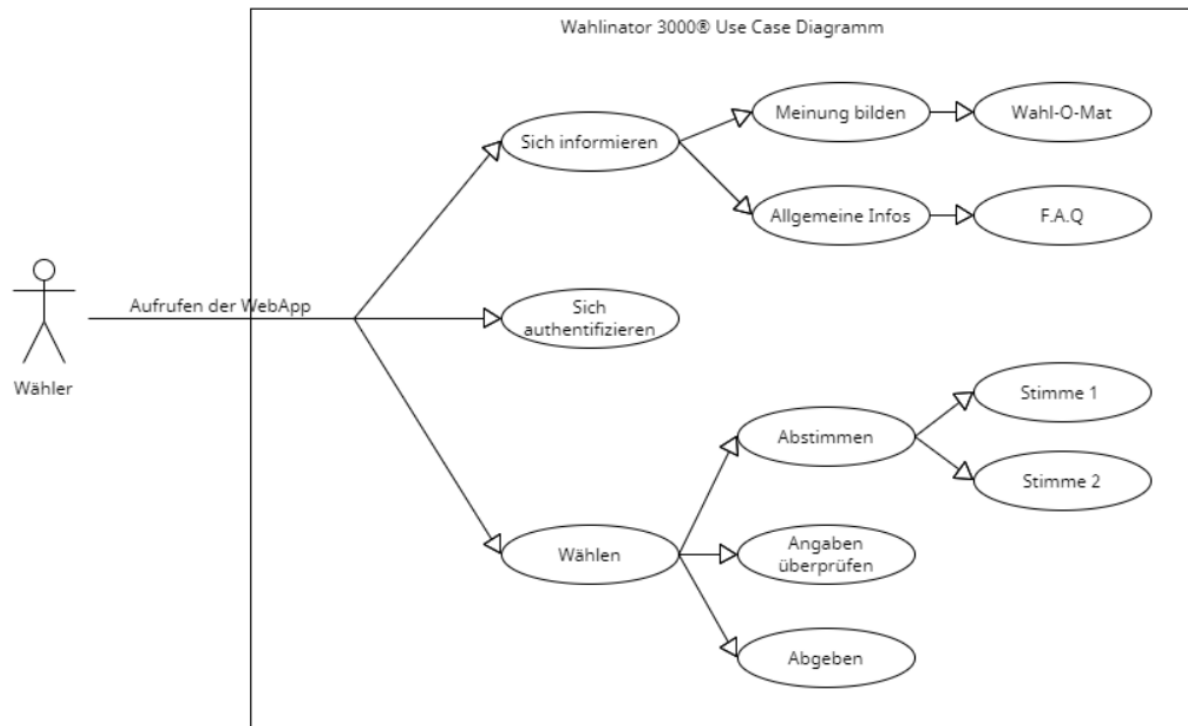
## Kontextdiagramm



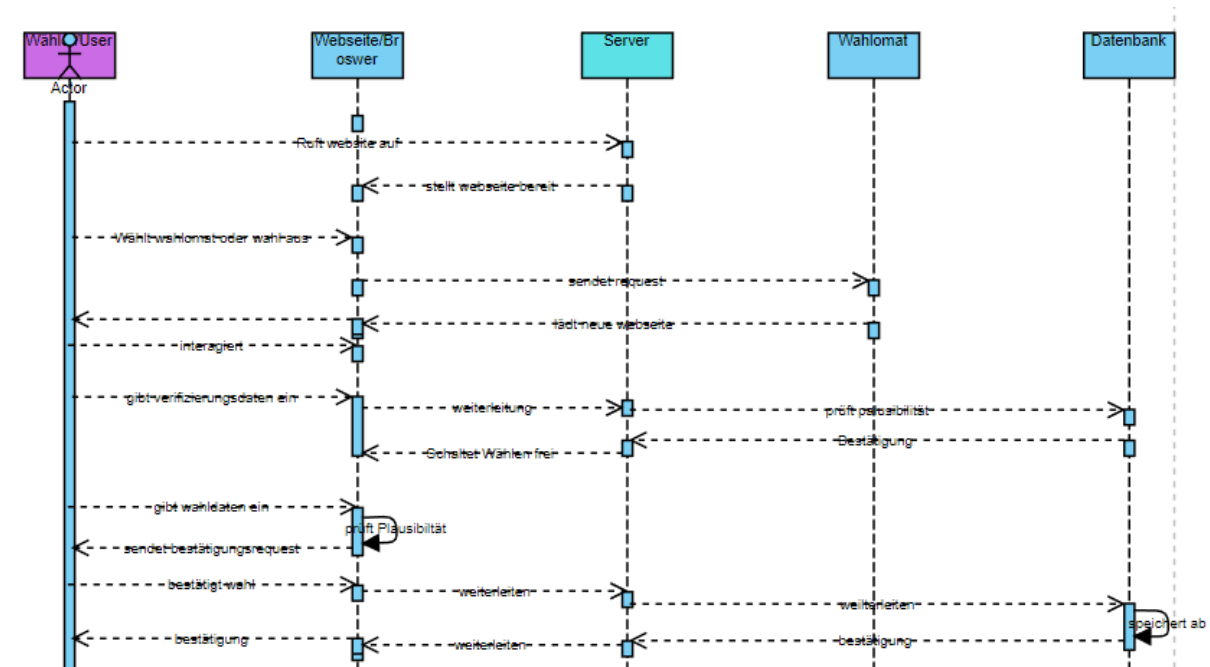
## Aktivitätsdiagramm



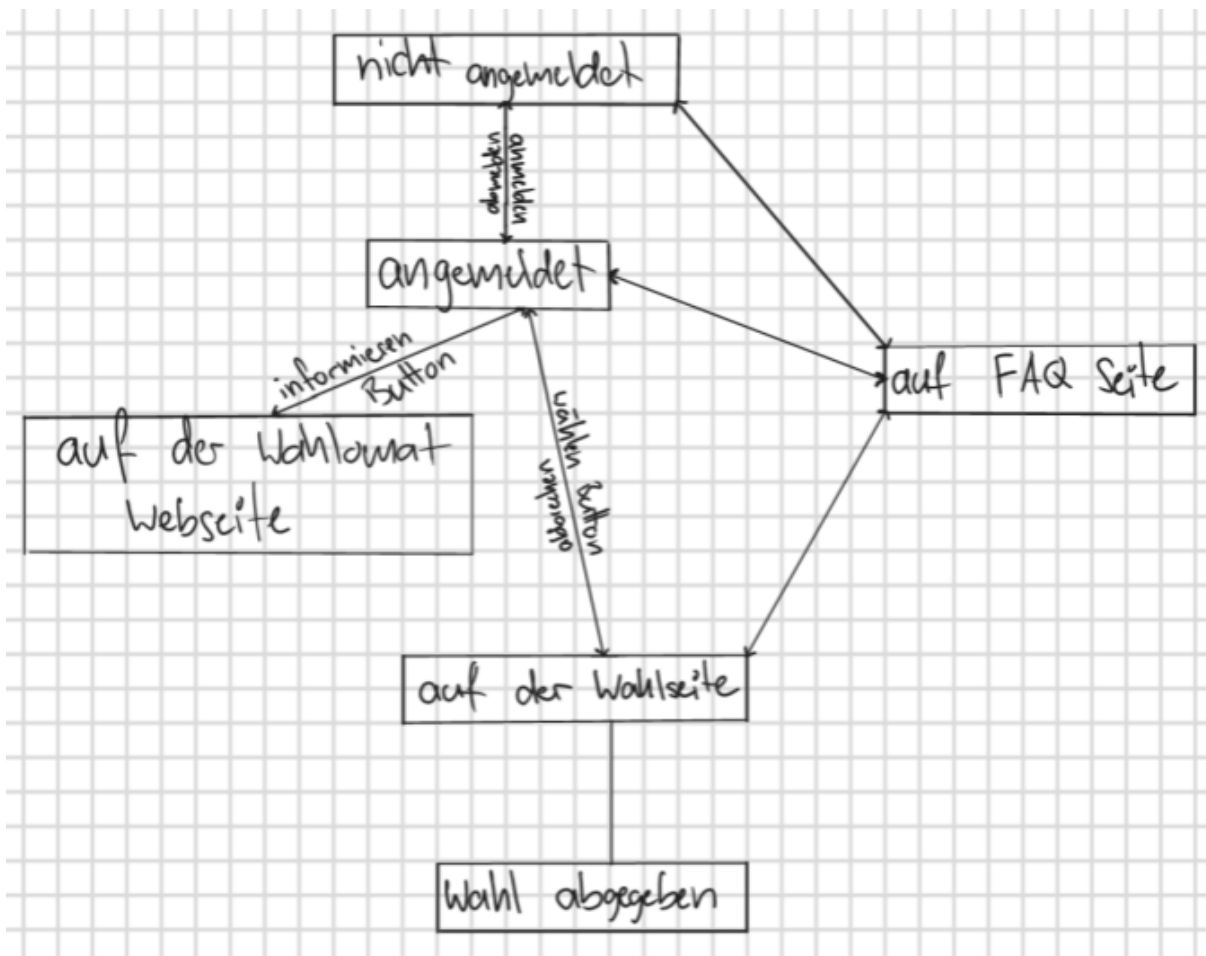
# Anwendungsfalldiagramm



# Sequenzdiagramm



## Zustandsdiagramm



# **Software Requirements Specification**

## for Wahlinator 3000

### **Version 1.0**

Prepared by

Leon Rensch, Alexander Witt, Adrian Bohnert, Daniel Monserrat, Corvin Pape, Jennifer Burek, Karlo Ivanovic

ITA21

02.11.2022

### Table of Contents

Table of Contents.....	ii
Revision History.....	ii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Product Scope.....	1
1.5 References.....	1
2. Overall Description.....	2
2.1 Product Perspective.....	2
2.2 Product Functions.....	2
2.3 User Classes and Characteristics.....	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints.....	2
2.6 User Documentation.....	2
2.7 Assumptions and Dependencies.....	3

3. External Interface Requirements.....	3
3.1 User Interfaces.....	3
3.2 Hardware Interfaces.....	3
3.3 Software Interfaces.....	3
3.4 Communications Interfaces.....	3
4. System Features.....	4
4.1 System Feature 1.....	4
4.2 System Feature 2 (and so on).....	4
5. Other Nonfunctional Requirements.....	4
5.1 Performance Requirements.....	4
5.2 Safety Requirements.....	5
5.3 Security Requirements.....	5
5.4 Software Quality Attributes.....	5
5.5 Business Rules.....	5
6. Other Requirements.....	5
Appendix A: Glossary.....	5
Appendix B: Analysis Models.....	5
Appendix C: To Be Determined List.....	6

#### Revision History

Name	Date	Reason For Changes	Version
Leon Rensch	02.11.2022	First Version	1.0.0

# 1. Introduction

## 1.1 Purpose

Das Ziel ist es, eine Webapplikation zu entwickeln, um das Wählen zu erleichtern und die Wahlbeteiligung zu steigern.

## 1.2 Document Conventions

Überschriften haben verschiedene Prioritäten

## 1.3 Intended Audience and Reading Suggestions

Dieses Dokument ist für die Benotung gedacht. Hauptsächlich wird es von Herrn Grün gelesen. Eigentlich ist das Dokument aber als Dokumentation für alle Stakeholder und als gemeinsames Protokoll gedacht für alle Entwickler, um bestimmte Dinge nachlesen zu können.

Im Folgenden wird das entstehende Produkt näher erläutert. Welche Perspektiven werden eingenommen, welche Funktionen hat das Produkt und tiefer liegende Informationen, wie User Classes, Design Infos und Dokumentation. Die restliche Struktur und die empfohlene Reihenfolge kann man dem Inhaltsverzeichnis entnehmen.

## 1.4 Product Scope

Der Wahlinator 3000 soll Wähler\*innen von Zuhause dazu animieren auch zu wählen. Zusätzlich zu den Wähler\*innen, die in Wahllokale gehen und Briefwahl machen, schafft der Wahlinator 3000 eine weitere Möglichkeit zu wählen. Zusätzlich verweist die Website auf den Wahlomat und schafft somit die Möglichkeit, sich vor der Wahl direkt nochmal zu erkundigen. Es soll möglich sein, gleichgültig wie bei den anderen Wahlverfahren an einer Bundes- oder Landtagswahl teilzunehmen. Dabei sollen alle Wahlgrundsätze und Sicherheitsrichtlinien sowie Datenschutz eingehalten werden.

## 1.5 References

Wahlomat, Bundeszentrale für politische Bildung, 19.10.2022:

<https://www.bpb.de/themen/wahl-o-mat/>

Erarbeitete Inhalte aus der Vorlesung: Oberhalb in diesem Dokument



## 2. Overall Description

### 2.1 Product Perspective

Das Produkt wird entwickelt, um das Wählen zu vereinfachen; dadurch können mehr Menschen zum Wählen angeregt werden. Das bereits bestehende Wahlsystem wird dabei nicht verändert.

### 2.2 Product Functions

Das Ziel des Projekts ist es, eine Wahl online zu ermöglichen. Dadurch zeichnet sich die Wahlfunktion als Hauptfunktion unseres Projekts aus. Dies kann sich in die 5 Wahlgrundsätze untergliedern:

- Allgemein:  
Bevor der User wählen kann, muss authentifiziert werden, ob der Wähler wahlberechtigt ist. Als Bedingung gilt: Der Wähler darf noch nicht gewählt haben, er muss min. 18 Jahre alt sein und muss Wahlrecht haben (Deutscher).
- Unmittelbar:  
In der Wahl sollen die zur Wahl stehenden Personen direkt gewählt werden
- Frei:  
Wähler darf ohne Druck wählen können
- Geheim:  
Es muss umgesetzt sein, dass man nicht erkennen kann wer, welchen Kandidaten gewählt hat (möglicher Einsatz einer Zero-Knowledge-Proof-Function)
- Gleich:  
Es darf keine Gewichtung der Stimmen geben. Jede Stimme muss gleich viel wert sein.

## 2.3 User Classes and Characteristics

### Wähler\*innen technik Affin:

- Nutzung bei jeder Wahl
- Wahlfunktion & Wahlomat werden genutzt
- technische Begabt
- normaler Sicherheitsstandard wie bei allen Wähler\*innen
- Berechtigung zur einmaligen Stimmabgabe
- Von keinem Abschluss bis zu keiner Bildungs-Grenze nach oben
- Erfahrung abhängig vom Alter und bisheriger Wahlbeteiligung
- Benötigt werden technische, internetfähige Endgeräte mit Internetzugang und eine Wahlberechtigung
- Wichtigste und einzige Nutzer der Anwendung

### Wähler\*innen nicht technik Affin:

- Nutzung für jede staatliche Wahl soll ermöglicht werden
- Wahlfunktion & Weiterleitung zum Wahl-O-Mat
- keine oder geringe technische Erfahrung
- gleicher Sicherheitsstandard und gleiche Berechtigung für jeden Wähler
- geringe bis hohe Bildung
- Erfahrung abhängig vom Alter und bisheriger Wahlbeteiligung
- Benötigt werden technische, internetfähige Endgeräte mit Internetzugang und eine Wahlberechtigung
- Der Wählergruppe soll die Alternative angeboten werden, auch online zu wählen. wichtig dabei ist ein einfaches und übersichtliches Layout, sodass Wähler intuitiv mit der Webseite umgehen können

## 2.4 Operating Environment

Das Endergebnis des Projekts soll eine funktionierende Webseite sein. Mögliche Erweiterungen sind: Authentifikation über digitalen Ausweis, Implementation einer Datenbank zur Speicherung der Wahlergebnisse und eine zusätzliche Datenbank über Personen, die gewählt haben.

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

## 2.5 Design and Implementation Constraints

Wahlgrundsätze sollen durch unser Projekt erfüllt werden. Dabei steht besonders der Datenschutz im Vordergrund. Zusätzlich muss ein starker Fokus auf die Gewährleistung der Integrität der Wahl gelegt werden. Hierzu muss die Verfügbarkeit des Systems immer sichergestellt werden, auch bei vielen Nutzern gleichzeitig, denn eine Wahl kann nur schwer kurzfristig verschoben werden. Dies ist mit Redundanz und einer sehr hohen maximalen Auslegung der maximalen gleichzeitigen Anfragen umzusetzen. Eine Downtime während der Wahlphase würde Wähler benachteiligen und so die Wahlgrundsätze verletzen und zudem das Vertrauen in den Wahlprozess zerstören. Mit der gleichen Begründung muss ebenfalls jegliche Manipulation oder sonstige Einflussnahme von außen ausgeschlossen werden. Zur Verifizierung des Wählers kann hierbei ein Zero Knowledge Proof eingesetzt werden. Um Datenmanipulation im Nachhinein zu verhindern, kann beispielsweise eine Blockchain aufgebaut werden. Zudem muss verhindert werden, dass später aufgrund von Zeitvergleichen nachvollziehbar ist, wer wen gewählt hat.

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).

## 2.6 User Documentation

(siehe Anfang Dokument)

- SMART
- Personas/Stakeholder
- Wie können wir...?
- Goldener Pfad
- Design-Konzept
- UML Diagramme

## **2.7 Assumptions and Dependencies**

Die Regierung muss mitspielen (Zustimmung im Bundestag etc.)

Die Bevölkerung muss davon erfahren

Die Entwicklung der Anwendung ist abhängig von unseren Entwicklungsfähigkeiten. Je besser wir die Website coden können, desto besser wird die Seite.

Einschränkungen würden wir wahrscheinlich vom Bund bekommen und müssten im Allgemeinen sehr eng mit den Gesetzgeber\*innen zusammenarbeiten

## **3. External Interface Requirements**

### **3.1 User Interfaces**

Siehe Design-Konzept

Startseite: Hier kommt der User über Buttons zum Wahlomat, zur Anmeldung für die Wahl und direkt zum FAQ.

FAQ: Hier sind alle Fragen aufgelistet und mit einem Klick auf diese Frage wird darunter die Antwort sichtbar.

Anmeldeseite: Hier muss der User seine ID und sein Passwort eingeben und wird über den Button "Anmelden" angemeldet. Außerdem gibt es einen Button zum FAQ.

Wahlseite: Hier kann der User seine Erst- und Zweitstimme über zwei Auswahlmenüs auswählen und über einen Senden-Button senden. Bevor gesendet wird, wird der User gefragt, ob er wirklich absenden will. Ein weiterer Button führt zum FAQ.

### **3.2 Hardware Interfaces**

Datenbank auf eigenem Server.

Abfragen von ID und Passwort zum Vergleich beim Anmelden.

Abfragen, ob mit der ID schon gewählt wurde.

### **3.3 Software Interfaces**

(CSS wird weitgehend über Bootstrap 5 geregelt.)

(Rest über HTML5 und JavaScript)

oder

(Angular + SQL Datenbank)

Daten werden abgefragt und gespeichert über eine SQL- oder NoSQL-Datenbank. Beim Anmelden wird das Passwort verschlüsselt und mit dem verschlüsselten Passwort in der Datenbank verglichen.

Nach dem Absenden wird die Auswahl in der Datenbank gespeichert.

### **3.4 Communications Interfaces**

(Benötigt wird ein Web Browser, der HTML5 kann.)

Kommunikation von Browser und Datenbank läuft über SQL/NoSQL. Dateneingabe über HTML forms und Verbindung mit dem Server über JavaScript und SQL/NoSQL

# 4. System Features

## 4.1 Login (Identifikation)

### 4.1.1 Description and Priority

Der Login hat eine hohe Priorität und soll den User ermöglichen, sich eindeutig zu identifizieren. Dieser Schritt ist notwendig, um die Berechtigung zum Wählen zu bekommen. Hierbei ist die Sicherheit des Logins und die User bezogenen Daten ein Risikofaktor, der entsprechend beachtet werden muss.

### 4.1.2 Stimulus/Response Sequences

- 2 Eingabefelder für "Username" und "Passwort"
  - Hier gibt der User seine anmeldedaten ein
- Anmelde Button
  - Wenn der User den Button drückt, wird überprüft, ob die beiden Eingabefelder auch befüllt wurden und gibt eine dementsprechende Fehlermeldung, falls eines der Felder nicht ausgefüllt wurde
  - Sind alle Felder ausgefüllt, werden die Eingabedaten mit der Datenbank überprüft und leitet den User weiter, oder gibt ihm eine Fehlermeldung

### 4.1.3 Functional Requirements

#### REQ-1: Eingabe Fehler

Wenn nicht beide Eingabefelder gefüllt sind, wird es dem User mit einer Fehlermeldung angezeigt.

#### REQ-2: Datenabgleich Fehler

Wenn die eingegebenen Daten nicht mit der Datenbank übereinstimmen, gibt es eine dementsprechende Fehlermeldung für den User

#### REQ-3: Schon gewählt

Wenn über die im Login eingegebenen Daten Abgestimmt wurde, bekommt der User eine Meldung darüber und wird nicht zum wählen weitergeleitet

## 4.2 Wählen (Speichern der Wahl)

### 4.2.1 Description and Priority

Diese Funktion soll es ermöglichen, dass der User seine Wahl abgeben kann und diese gespeichert wird. Dabei ist es wichtig, dass das Wahlergebnis des Users nicht zu ihm zurückverfolgt werden kann. Diese Funktion hat die höchste Priorität im Projekt.

### 4.2.2 Stimulus/Response Sequences

- Auswahl der beiden Wählergruppen (Direktmandat und Partei) über Radio Groups
- Abgabe der Wahl über Button "Wählen"
  - Die Wahlentscheidung wird angezeigt und der User wird gefragt, ob dies seine endgültige Wahl ist → der User hat die Möglichkeit zurück zur Wahl zu gehen oder seine Wahl zu bestätigen
  - Wird die Wahl bestätigt, wird ein neuer Datensatz in die Datenbank "Wahl" aufgenommen, der die Wahlergebnisse speichert
  - Ist der Wähler mit seiner Wahl nicht zufrieden wird, er zurück auf die Wahlseite geleitet und er kann erneut wählen
  - Wenn der Wähler seine Stimme abgegeben hat, wird der Zugang zur Wahl für ihn gesperrt, sodass er nicht nochmal wählen kann.

REQ-4: unvollständige Wahl

Wenn der Wähler nicht beide Stimmen abgegeben hat, wird er aufgefordert beide Stimmen abzugeben

## 4.3 Verlinken auf Wahl-O-Mat

### 4.3.1 Description and Priority

Die Funktion soll den Nutzer ermöglichen, sich über die Wahl und die einzelnen Wahlparteien zu informieren. Dafür soll ein Link zur Seite "wahl-o-mat.de" erstellt werden, der den User auf diese weiterleitet. Dieses Feature hat eine geringere Priorität

### 4.3.2 Stimulus/Response Sequences

Es soll als Button einen Link geben, der den User auf Wahl-O-Mat weitergeleitet.

REQ-5: Weiterleitung

Nach dem Drücken des Buttons, soll der User auf die Seite vom Wahl-O-Maten weitergeleitet werden.



## **4.4 FAQ**

### **4.4.1 Description and Priority**

Das FAQ hat eine mittlere Priorität und soll den User über alle aufkommenden Fragen eine Antwort geben können. Die Seite besteht aus einer Auflistung von Fragen.

### **4.4.2 Stimulus/Response Sequences**

Klickt der User auf eine Frage, so wird die Antwort darunter sichtbar und verschwindet beim erneuten Klicken.

#### **REQ-6: FAQ Erreichbarkeit**

Das FAQ soll von jeder Seite durch einen Button erreichbar sein.

#### **REQ-7: Antwort anzeigen**

Wenn auf die Frage geklickt wird, wird die Antwort darunter sichtbar und verschwindet beim erneuten Klicken.

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

Die Webseite muss unabhängig vom Web Browser und Betriebssystem aufgerufen werden können. Auch von einem Handy muss das Anmelden und Wählen funktionieren.

### 5.2 Safety Requirements

Wenn der User wählt und nicht die richtige Anzahl an Stimmen gesetzt wurden, sollen die Daten nicht gespeichert werden und der User informiert werden.

### 5.3 Security Requirements

Datenbank soll vor Angriffen geschützt sein. Daten auf der Datenbank sollen verschlüsselt sein. Daten vom User und das, was er gewählt hat, dürfen in keinem Zusammenhang stehen.

### 5.4 Software Quality Attributes

Das Design der Webseite soll auf allen Bildschirmgrößen und auch auf dem Handy ansprechend sein.

Das Wählen soll über jeden Browser und Betriebssystem ohne Probleme möglich sein.

### 5.5 Business Rules

Wähler können nur 1 mal wählen. User bezogene Daten sollen abgekapselt von der Wahl sein, damit die Wahl auch wirklich anonym ist. Alle Passwörter sollen verschlüsselt auf der Datenbank liegen und auch nie entschlüsselt werden.

## 6. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

#### Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>

# Architecture Design

## Fragen

1. Gibt es eine allgemeine Anwendungsarchitektur, die als Schablone für das zu entwerfende System dienen kann?

Repository-Architektur

2. Wie wird das System auf Threads oder Prozessoren verteilt?

Das macht der Browser/Server automatisch

3. Welche Architekturmuster sind für das System geeignet?

Repository-Architektur und Client-Server-Architektur

4. Mit welcher Strategie wird der Betrieb der Systemkomponenten gesteuert?

5. Wie soll die Systemarchitektur dokumentiert werden?

Mit UML-Diagrammen vor allem Klassendiagrammen

6. Welche architektonische Strukturierung ist die beste für die nicht-funktionalen Anforderungen?

Client-Server-Architektur

7. Wie werden die Komponenten im System in Teilkomponenten zerlegt?

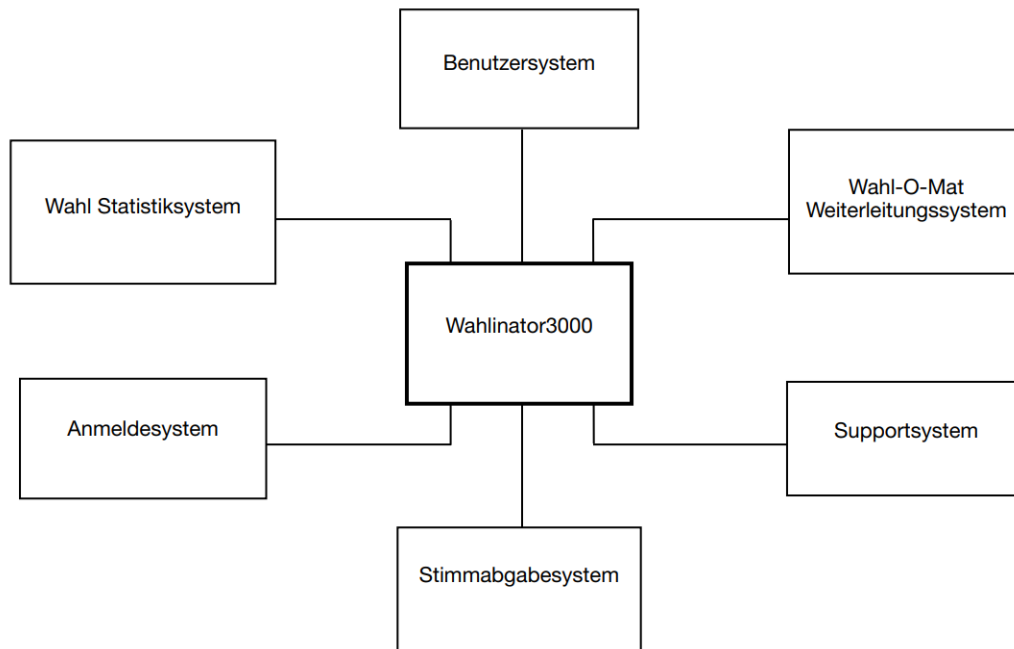
Die verschiedenen Seiten der Webseite werden aufgeteilt und haben unterschiedliche HTML- und Javascript Dateien. Außerdem ist das Design(css) auch abgekapselt. Auch der Server ist extern und wird einzeln behandelt

8. Wie lautet der grundlegende Ansatz zur Strukturierung des Systems?

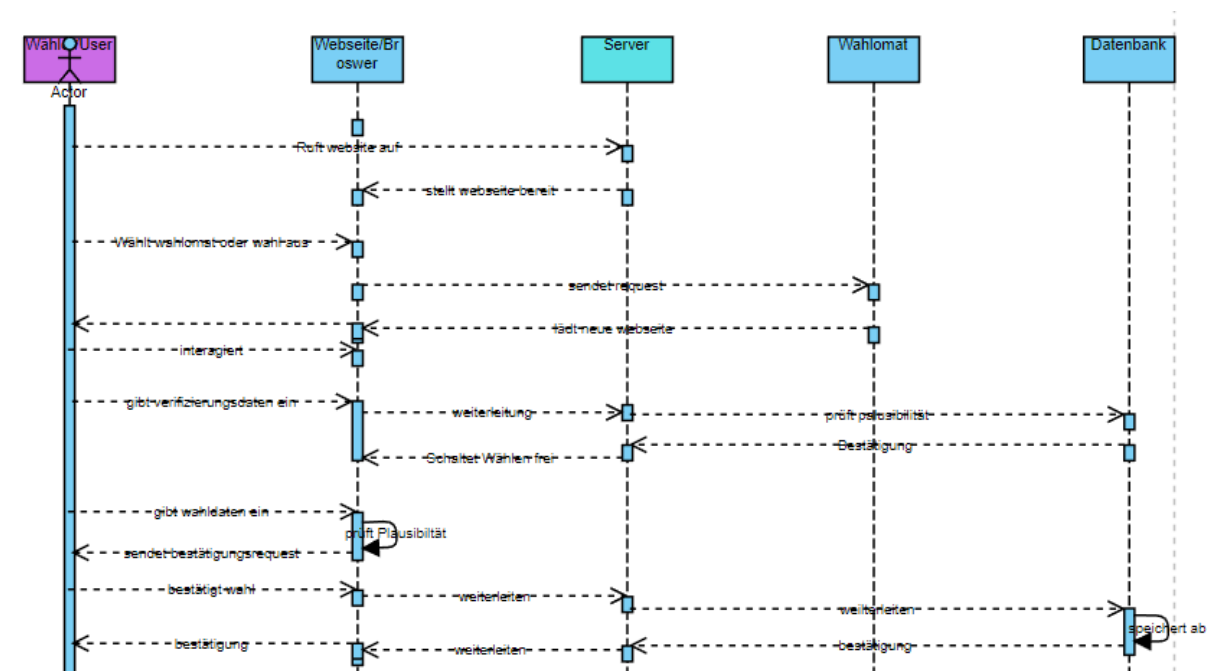
Die Website in möglichst kleine Komponenten aufteilen

# Systemmodellierung und Softwarearchitektur

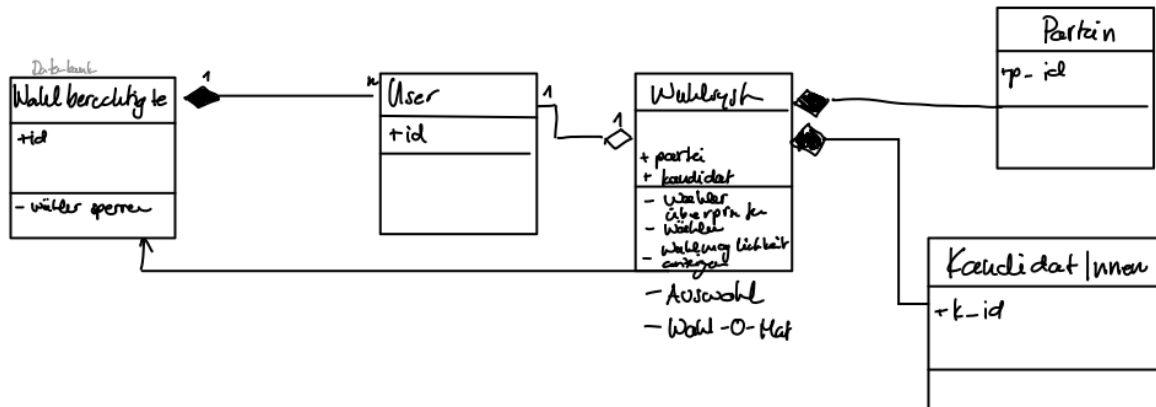
## Kontextmodell



## Verhaltensmodell



## Klassendiagramm



## Interaktionsmodell

