

Entwicklungsprojekt interaktive Systeme

Wintersemester 2016/2017

Dozenten

Prof. Dr. Gerhard Hartmann

Prof. Dr. Kristian Fischer

Betreuer

Franz-L. Jaspers, B.Sc.

Dokumentation von Gruppe 22

Jessica Lee Schulz

Enrico Gette

Inhaltsverzeichnis

| | |
|---|----|
| Exposé | 3 |
| Name des Systems „Thes-O-Naise“ | 5 |
| Domänenrecherche | 5 |
| Marktrecherche | 7 |
| Alleinstellungsmerkmale | 7 |
| Zielhierarchie | 8 |
| Methodischer Rahmen der Mensch Computer Interaktion | 9 |
| Vorgehen im „Usability Engineering Lifecycle“ | 10 |
| Stakeholder Analyse..... | 10 |
| Benutzermodellierung | 12 |
| Anforderungen | 13 |
| Funktionale Anforderungen | 13 |
| Organisationale Anforderungen | 14 |
| Qualitative Anforderungen | 14 |
| Anforderungen an die Benutzerschnittstelle | 14 |
| Technische Anforderungen..... | 14 |
| Task-Analysis..... | 15 |
| Usability Goals | 15 |
| Phase 2: Design Testing Development..... | 16 |
| Level 1 Work Reengineering | 17 |
| Level 1: Conceptual Model Design..... | 18 |
| Level 2: Screen Design Standards | 19 |
| Level 3: Detailed User Interface Design | 20 |
| Evaluation | 28 |
| WBA Teil des Projektes..... | 30 |
| Kommunikationsmodelle | 30 |
| Deskriptives Kommunikationsmodell..... | 30 |
| Präskriptives Kommunikationsmodell..... | 31 |
| Architektur und Architekturdiagramm | 32 |
| Datenstrukturen | 34 |
| Risiken | 38 |
| Proof of Concept | 40 |
| Fazit | 42 |
| Literaturverzeichnis | 42 |
| Anhang | 44 |

Tabellenverzeichnis

| | |
|--|----|
| Tabelle 1: Stakeholder „Wähler“..... | 10 |
| Tabelle 2: Stakeholder „Kandidat“ | 10 |
| Tabelle 3: Stakeholder Analyse..... | 58 |
| Tabelle 4: Definition der User Profiles..... | 12 |
| Tabelle 5: Regel des Conceptual Model Designs..... | 18 |
| Tabelle 6: Screen Design Standards..... | 19 |
| Tabelle 7: Punktesystem für die Berechnung..... | 33 |
| Tabelle 8: Interaktion mit Thesen beim Client..... | 40 |
| Tabelle 9: Berechnung der Übereinstimmung von Wähler und Kandidaten..... | 40 |
| Tabelle 10: Sichere HTTPS Verbindung zwischen Client und Server..... | 40 |
| Tabelle 11: Verschlüsselung der privaten Daten des Wählers..... | 41 |
| Tabelle 12: Vermeidung von Manipulation bei öffentlichen Zählern..... | 41 |
| Tabelle 13: Vermeidung von inhaltlich doppelten Thesen..... | 41 |
| Tabelle 14: Ressourcen der Server API..... | 51 |
| Tabelle 15: 1.User profile Wähler..... | 52 |
| Tabelle 16: 2.User profile Wähler..... | 53 |
| Tabelle 17: 3.User profile Wähler..... | 54 |
| Tabelle 18: 4.User profile Wähler..... | 55 |
| Tabelle 19: 2.User profile Kandidat..... | 56 |
| Tabelle 20: 2.User profile Kandidat..... | 57 |
| Tabelle 21: Generelle Designprinzipien..... | 60 |
| Tabelle 22: Use Case Definitionen..... | 61 |
| Tabelle 23: Use Case 1: Eine These veröffentlichen..... | 62 |
| Tabelle 24: Use Case 2: Eine Position zu einer These beziehen..... | 63 |
| Tabelle 25: Use Case 3: Eine Begründung zu einer These hinzufügen | 64 |
| Tabelle 26: Use Case 4: Kommentar zu einer These hinzufügen..... | 65 |
| Tabelle 27: Use Case 5: Die Übereinstimmung berechnen lassen | 66 |
| Tabelle 28: Use Case 6: Thesen bewerten..... | 67 |
| Tabelle 29: Use Case 7: Thesen sortieren..... | 68 |
| Tabelle 30: Use Case 8: Registrieren..... | 69 |
| Tabelle 31: Use Case 9: Einloggen..... | 70 |
| Tabelle 32: Use Case 10: Eine These veröffentlichen..... | 71 |
| Tabelle 33: Cognitive Walkthrough – Use Case 1..... | 72 |
| Tabelle 34: Cognitive Walkthrough – Use Case 2..... | 73 |
| Tabelle 35: Cognitive Walkthrough – Use Case 3..... | 74 |
| Tabelle 36: Cognitive Walkthrough – Use Case 4..... | 75 |
| Tabelle 37: Cognitive Walkthrough – Use Case 5..... | 76 |
| Tabelle 38: Cognitive Walkthrough – Use Case 6..... | 77 |
| Tabelle 39: Cognitive Walkthrough – Use Case 7..... | 78 |
| Tabelle 40: Cognitive Walkthrough – Use Case 8..... | 79 |
| Tabelle 41: Identifizierte Usability Probleme..... | 80 |

Exposé

Nutzungsproblem

Die nächste Bundestagswahl findet 2017 statt und die Massenmedien berichten hauptsächlich über die etablierten Parteien. Der Wahl-o-Mat der Bundeszentrale für politische Bildung beinhaltet nur redaktionell vorgefertigte Thesen, zu denen nur Parteien Stellung (pro/neutral/contra) beziehen können. Der mündige Wähler hat möglicherweise speziellere Thesen und hat kein zentrales System, um diese Thesen allen Direktkandidaten des eigenen Wahlkreises zu stellen, sodass die Kandidaten eine Stellung dazu beziehen könnten. Die veröffentlichten Thesen sollen dabei die politischen Wünsche und Forderungen der Wähler für die Bundesrepublik Deutschland oder für den jeweiligen Wahlkreis darstellen. Benutzer eines solchen Systems können dann ihre eigenen Ansichten zu den Thesen mit den Positionen der Kandidaten vergleichen und öffentlich oder privat über die Begründungen zu einer These von den Kandidaten diskutieren.

Zielsetzung

Im Rahmen des Projektes soll ein zentrales System entwickelt werden, welches für die Wähler eine Unterstützung der Meinungsbildung zu den Kandidaten sein soll und die Kommunikation zwischen Wähler und Kandidaten verbessern könnte. Hinzukommt, dass das System die Möglichkeit bieten soll, Thesen an alle Kandidaten eines Wahlkreises öffentlich zu stellen, seine persönlichen Ansichten mit den Positionen der Kandidaten zu vergleichen und die Kandidaten mit der größten Übereinstimmung zu den persönlichen Ansichten zu ermitteln. Dabei soll ein besonderer Fokus auf den Datenschutz gelegt werden. Das System soll den Wählern auch die Möglichkeit bieten, Diskussionen mit anderen Wählern und den Kandidaten über Thesen zu führen. Zudem soll es eine anschauliche Darstellung aller Kandidaten im Wahlkreis des Benutzers geben.

Verteilte Anwendungslogik

Nach der Beantwortung von Thesen durch Wähler und Kandidaten sollen die Kandidaten ermittelt werden, welche die größte Übereinstimmung mit dem Wähler haben. Ein Client überträgt die verschlüsselten privaten Daten an den Server, dabei soll der Server die Berechnung der Übereinstimmung durchführen ohne dauerhaft private Informationen zu speichern. Außerdem soll das System ein Bewertungssystem für die Benutzer beinhalten, das berechnet, welcher Kandidat die beste Begründung zu den jeweiligen Thesen gegeben hat. Zudem soll das System automatisch die Wahlbeteiligung der Benutzer im Wahlkreis sowie bundesweit berechnen und eine Prognose basierend auf der Übereinstimmung zwischen Wählern und Kandidaten für die jeweiligen Wahlkreise liefern. Die Anwendungslogik des Clients beinhaltet Benachrichtigungen zu eigenen oder abonnierten Thesen, sowie zu abonnierten Kandidaten.

Wirtschaftliche und gesellschaftliche Relevanz

Für den unentschlossen und interessierten Wähler ist durch ein zentrales System der Aufwand geringer, sich mit den verschiedenen Kandidaten für seinen Wahlkreis auseinandersetzen und seine Ansichten zu bestimmten Thesen mit den Positionen der Kandidaten zu vergleichen. Dadurch würde der Wähler in Hinblick auf die Meinungsbildung zu den Kandidaten unterstützt. Hinzukommt, dass die Thesen nicht durch eine Redaktion ausgewählt werden, sondern vom Wähler selbst verfasst werden könnten.

Name des Systems „Thes-O-Naise“

Der Name des Systems ist zusammengesetzt aus den Begriffen „These“, „Wahl-o-Mat“ und „Polonaise“. Damit soll verdeutlicht werden, dass sich Wähler mit anderen Wählern über das System anhand von Thesen zusammenschließen und damit den Kandidaten eine Übersicht der Wünsche und Forderungen der Wähler geben können.

Domänenrecherche

Im Folgenden soll zum einen die Domäne der Bundestagswahl und die des Crowdsourcing erfasst und analysiert werden und zu anderem ein Fazit gezogen werden.

Domäne Bundestagswahl

In der Bundesrepublik Deutschland beschließen die Parteien auf ihren Parteientagen, bei denen nur Mitglieder der Partei stimmberechtigt sind, ihr Wahlprogramm für die nächste Bundestagswahl. Mit diesen Wahlprogrammen wollen die Parteien die Wahlberechtigten von ihren politischen Positionen überzeugen, um in der nächsten Legislaturperiode an die Macht zu kommen und die Positionen entsprechend umsetzen zu können. Die etablierten Parteien CDU, SPD, Linke und Bündnis90/Die Grünen boten zur letzten Wahl auf unterschiedliche Weise den Bürgerinnen und Bürgern die Möglichkeit auf die Gestaltung der Wahlprogramme einzugehen (vgl. Landeszentrale für politische Bildung Baden-Württemberg, 2016). Bei der CDU gab es die Möglichkeit, ihre Vorhaben in acht ausgewählten Politikfeldern zu kommentieren, bei der SPD konnten sich die Bürger durch einen Bürgerdialog an der Gestaltung ihres Programms beteiligen, die Linke hat alle Bürger zur Teilhabe an einer Debatte über ihren Programmentwurf eingeladen und die Grünen hatte Parteimitglieder und die Öffentlichkeit dazu eingeladen, auf Programmforen in den Ländern und in Online-Fragestunden mit den SpitzenkandidatInnen und den Bundesvorsitzenden über ihren Programmentwurf zu diskutieren. Diese Möglichkeiten standen nur wenige Monate vor der Wahl den Bürgern zur Verfügung und sind nach der Wahl größtenteils nicht mehr einsehbar. (vgl. SPD, 2016)

Will der Wähler sich zur derzeitigen Prognose der Wahl informieren, so muss er auf die Umfragen von verschiedenen privaten und profitorientierten Meinungsforschungsinstituten wie Infratest dimap, INSA/YouGov, Forsa, Allensbach oder TNS Emnid zurückgreifen, welche sich nicht auf die einzelnen Wahlkreise beziehen und welche regelmäßig in der Kritik standen. (vgl. Plickert, Philip, 2013)

Zurzeit kann sich jeder Wähler über die verschiedenen Medien wie Internet, Zeitung, Radio und Fernsehen oder bei einer Wahlkampfveranstaltung der Parteien oder unabhängiger Direktkandidaten über diese und ihre politischen Positionen informieren. Der Einfluss der Wähler zu den Positionen der Kandidaten bei den Online-Nachrichtenportalen kann als gering beschrieben werden, da die Wähler lediglich Kommentare bei diesen abgeben können, auf welche nicht zwingend eingegangen werden muss seitens der Kandidaten. Einige Online-Nachrichtenportale lassen keine Kommentarfunktion bei kritischen Themen zu. Bei den Wahlkampfveranstaltungen der Parteien werden dem Wähler die Positionen der Partei bezüglich verschiedener Themen erläutert.

Um die Position eines bestimmten Kandidaten mit der eines anderen zu einer These zu ermitteln und zu vergleichen, muss viel Aufwand seitens des Wählers betrieben werden. Entweder muss jeder Kandidat direkt angeschrieben werden oder viel Zeitaufwand in Recherchen gesteckt werden. Die steigende Nachfrage der Wähler ihre persönlichen Ansichten mit den Positionen der Kandidaten oder Parteien schnell und einfach vergleichen zu können, kann anhand der Nutzungen des Wahl-O-Mat

belegt werden, da dieser bis kurz vor der Bundestagswahl 2009 rund 6,7 Millionen Mal benutzt wurde. (vgl. Marschall, Stefan (2011))

Letztendlich beschränkt sich der Einfluss der Wähler auf die Wahlprogramme und Positionen der Kandidaten auf einen kurzen Zeitraum vor der Wahl, und endet nach der Wahl. Die Wähler haben bislang keinerlei Möglichkeiten schnell und einfach an alle Kandidaten eines Wahlkreises Thesen zu stellen und deren Stellungnahme und Begründungen zu bewerten.

Domäne Crowdsourcing

Unter dem Begriff Crowdsourcing wird verstanden, dass ein Unternehmen auf die verschiedenen Kompetenzen und Ideen der Masse zurückgreift. Das Unternehmen, in der Domäne auch Crowdsourcer genannt, beginnt den Crowdsourcing-Prozess, indem es die zu bearbeitenden Aufgaben definiert und Anreizstrukturen erschafft. Die Individuen der Masse, auch Crowd Worker bzw. Digital Worker genannt, suchen sich ihre Aufgaben selbst aus und bearbeiten diese. Innerhalb dieser Domäne kann zwischen zwei Arbeitsansätzen unterschieden werden. Es gibt einen wettbewerbsbasierten Ansatz und einen zusammenarbeitsbasierten Ansatz, im Folgenden wird nur letzterer beschrieben. Bei diesem Ansatz kann ein Crowd Worker einen Beitrag einreichen, welcher anschließend von anderen Crowd Workern überarbeitet oder erweitert werden könnte. Dies führt zu einer gemeinsam erarbeiteten Lösung der Aufgaben. (vgl. Leimeister, J. M.; Zogaj, S.; Durward, D. & Blohm, I. (2015))

Fazit der Domänenrecherche

Für das System "Thes-O-Naise" eignet sich Crowdsourcing mit dem zusammenarbeitsbasierten Ansatz besonders gut, da jeder Benutzer die Möglichkeit haben soll eigene Thesen zu formulieren, sowie die Begründung der Position zu den Thesen von den Kandidaten zu diskutieren und bewerten. Die Aufgabe, welche das System der „Crowd“ stellt, ist, durch die Formulierung von Thesen zu verdeutlichen, was sich die Wähler für den Wahlkreis und für die Bundesrepublik Deutschland politisch wünschen. Je mehr Benutzer mitmachen, desto mehr kristallisiert sich der Wille der Wähler heraus. Dadurch bekommen auch die Kandidaten einen besseren Blick auf die Wünsche der Wähler, wie auch umgekehrt.

Marktrecherche

Im Folgenden wurden im Zuge der Marktrecherche Konkurrenzprodukte in der Domäne kurz beschrieben und in ihrer Funktion als Wahlhilfe analysiert. Die jeweiligen Analysen (Wahl-O-Mat der Bundeszentrale für politische Bildung, www.erststimme2013.de und www.parteivergleich.eu) befinden sich im Anhang auf den Seiten 43 bis 48.

Fazit der Marktrecherche

Zusammengefasst kann anhand der Marktrecherche kein Konkurrenzprodukt identifiziert werden, welches den Wählern ermöglicht Thesen an alle Direktkandidaten eines Wahlkreises zustellen und einen Diskurs über die Thesen und die Begründungen der Kandidaten zu diesen Thesen zu führen. Außerdem sind die beschriebenen Konkurrenzprodukte erst wenige Monate oder Wochen vor der Wahl den Wählern zugänglich, wodurch den Wählern weniger Zeit zur Verfügung steht sich mit den Positionen der Kandidaten oder Parteien auseinander zu setzen.

Alleinstellungsmerkmale

Anhand der Ergebnisse der Recherchen, konnte kein Konkurrenzprodukt ermittelt werden, welches folgende Alleinstellungsmerkmal besitzt:

Benutzer haben die Möglichkeit eigene Thesen zu formulieren und diese den Kandidaten eines Wahlkreises zu stellen, sodass die Kandidaten eine Stellung (pro/neutral/contra) dazu einnehmen können.

Eigene Thesen zu formulieren, ist eine Form der freien Meinungsäußerung, welche den politischen Diskurs und damit auch die Demokratie fördert. Nur durch redaktionelle vorgefertigte Thesen können viele Themen, welche den Wählern wichtig sind, nicht berücksichtigt werden. Durch die Formulierung von eigenen Thesen können die Wähler den Kandidaten ihre politischen Wünsche und Forderungen äußern. Anschließend können die Wähler die Thesen bewerten und das System kann diese dann nach Beliebtheit innerhalb einer Kategorie sortieren. Dadurch bekommen die Kandidaten des Wahlkreises einen besseren Überblick auf diese Forderungen und Wünsche der Wähler.

Nachdem ein Kandidat eine These beantwortet und seine Antwort begründet hat, können die Wähler diese Begründung zur These bewerten und kommentieren. Verschiedene Kandidaten können zwar beispielsweise einer These zustimmen, jedoch aus unterschiedlichen Gründen. Deshalb ist es für die Wähler wichtig bewerten zu können, welche Begründung zu einer These die persönlich bessere ist. Um Diskussionen zu den Begründungen zu ermöglichen, sollten die Wähler auch Kommentare schreiben können.

Haben die Kandidaten und der Wähler Thesen mit Pro, Neutral oder Contra beantwortet, vergleicht das System diese Antworten zu den jeweiligen Thesen und zeigt dem Wähler die Kandidaten mit der größten Übereinstimmung. Somit bietet das System dem Wähler eine Hilfe leichter abwägen zu können, welche Kandidaten seiner persönlichen Ansicht nahe kommen.

Möchte ein Wähler über neue Information eines Kandidaten oder über bestimmte Thesen in Kenntnis gesetzt werden, könnte er diese abonnieren. Dies hilft dem Wähler auf dem aktuellen Stand zu bleiben.

Zielhierarchie

Im Folgenden werden die strategischen (langfristig), taktischen (mittelfristig) und die operativen (kurzfristig) Ziele des Projektes erläutert.

Strategisches Ziel

1. Die Kommunikation zwischen Wähler und Kandidaten soll in Bezug auf den aktuellen Zustand verbessert werden.

Taktisches Ziel

- 1.1. Wähler und Kandidaten müssen die Möglichkeit haben, eigene Thesen über ein zentrales System zu veröffentlichen.

Operatives Ziel

- 1.1.1. Die Thesen sind den richtigen Wahlkreisen und Kategorien zu geordnet.
- 1.1.2. Der Wähler kann entscheiden, ob er eine These anonym veröffentlicht.
- 1.2. Wähler sollten die Möglichkeit haben, über Thesen und die Begründungen der Kandidaten zu diskutieren und diese zu bewerten.
 - 1.2.1. Die Positionen der Kandidaten zu einzelnen Thesen müssen öffentlich zugänglich sein.
 - 1.2.2. Wähler sollten die Möglichkeit haben öffentlich oder privat zu diskutieren.
 - 1.2.3. Wähler sollten die Möglichkeit haben jede Begründung eines Kandidaten zu bewerten.
2. Wähler sollen bei der Meinungsbildung über die Kandidaten des Wahlkreises in Bezug auf den aktuellen Zustand unterstützt werden.
 - 2.1. Der Wähler muss die Möglichkeit haben seine persönlichen Ansichten bei Thesen mit den Positionen aller Kandidaten des Wahlkreises zu vergleichen und die Kandidaten mit der größten Übereinstimmung zu sehen.
 - 2.1.1. Es muss bei gleichen Daten immer das gleiche Ergebnis berechnet werden.
 - 2.1.2. Die Darstellung des Ergebnisses sollte anschaulich und hilfreich für den Wähler sein.
 - 2.1.3. Die Übereinstimmung soll auch bei einzelnen Kategorien berechnet werden.
3. Die privaten Daten des Wählers müssen geschützt werden.
 - 3.1. Die privaten Daten des Wählers müssen sicher übertragen werden.
 - 3.2. Die privaten Daten des Wählers sollen verschlüsselt gespeichert werden.

Methodischer Rahmen der Mensch Computer Interaktion

Auswahl des Design-Prinzips und des Vorgehensmodell

Im Folgenden wird zuerst ein Design-Prinzip festgelegt, auf dessen Basis ein geeignetes Vorgehensmodell aus der Mensch-Computer-Interaktion ausgewählt wird, sodass eine angemessene Umsetzung des Projektes sichergestellt werden kann.

Auswahl des Design-Prinzips

Bei den Design-Prinzipien muss zwischen dem Design-Prinzip „User centered design“ und dem „Usage centered design“ entschieden werden.

Für das Projekt wird das Design-Prinzip „User centered design“ ausgewählt, da die Stakeholder hinsichtlich ihrer Eigenschaften bestimmte Anforderungen an das System stellen. Bezogen auf das Projekt ist es wichtig die Aufgaben, Ziele und den Nutzungskontext des Benutzers zu erfassen um die Gebrauchstauglichkeit des Systems für die Benutzer zu gewährleisten. Der Benutzer sollte bei dem Gebrauch des Systems intuitiv wissen, wie er neue Thesen veröffentlichen kann, die Begründungen der Kandidaten bewerten kann und seine Ansichten mit den Positionen der Kandidaten vergleichen kann.

Auswahl des Vorgehensmodell

Nun wird ein Vorgehensmodell aus dem Bereich des „User centered design“ gewählt, um den Arbeitsvorgang des Projekts an dessen Ansatz und Verlauf anzupassen.

Mit dem Modell „Discount Usability-Engineering“ von Nielsen wird argumentiert, dass kostengünstig und mit wenigen einfachen Techniken eine sichtbare Verbesserung der Gebrauchstauglichkeit möglich ist. Die Evaluation in diesem Modell besteht aus Prototypen als Szenarien in papier-basierter Form, lautem Nachdenken und zehn Heuristiken nach Nielsen, auf die ein besonderer Fokus gelegt wird. Allerdings weist dieses Modell Schwächen auf, da die Anforderungsanalyse nicht berücksichtigt wird. Sie ist aber für die Basis des Projekts signifikant, um allen Anforderungen der Stakeholder gerecht zu werden.

Das „Scenario Based Usability Engineering“ nach Rosson und Carroll fokussiert sich auf dem Verstehen, Beschreiben und Modellieren menschlichen Handelns anhand der Nutzung von Szenarien. Da dieses Projekt sich aus Zeitgründen nur auf den mobilen Nutzungskontext beschränkt, ist dieses Modell nicht geeignet da es sich auf das Verstehen mehrerer Nutzungskontexte bezieht und nicht skalierbar ist.

Das Vorgehensmodell „Usability Engineering Lifecycle“ von Deborah Mayhew konzentriert sich auf die Benutzer und ihre Anforderungen an das System, welche in diesem Projekt eine hohe Priorität besitzen. Durch die iterativen Prozesse ist dieses Modell gut skalierbar und somit angemessen für dieses System, da die zu lösenden Entwicklungsaufgaben an den Benutzer angepasst werden.

Das Modell soll die relevanten Anforderungen analysieren und Stakeholder ermitteln und festlegen, welche in „user profiles“ dokumentiert werden. Unter anderem wird ein deskriptives Modell der Aufgaben erstellt, welches zur Recherche von Potentialen und Einschränkungen der Hard- und Softwaremöglichkeiten dient.

Vorgehen im „Usability Engineering Lifecycle“

In Mayhews Vorgehensmodell werden einige Techniken für die jeweiligen Prozessstufen dargestellt, welche in Hinsicht auf eine optimale Lösung durchgeführt werden sollten, aber nach Größe des Projekts angepasst werden können. Aufgrund der kurzen Projektdauer ist es notwendig das Modell im Umfang anzupassen und teilweise Artefakte auszulassen.

Phase 1: Requirement-Analysis

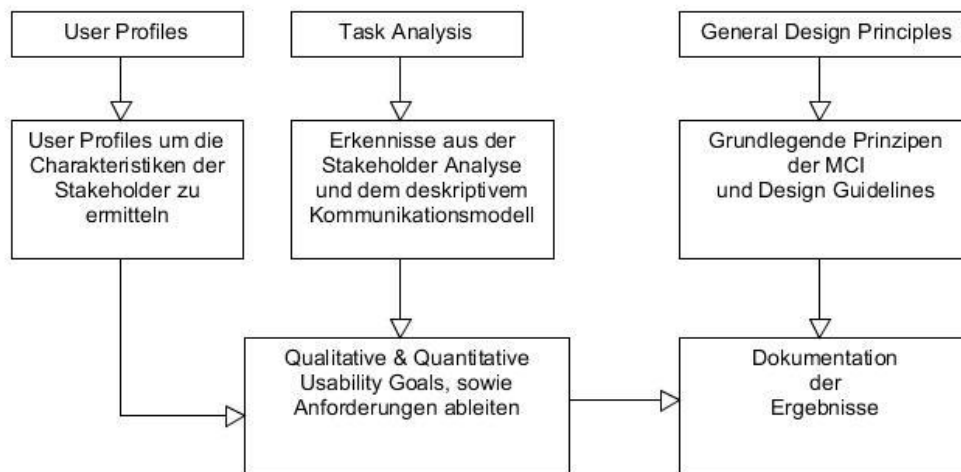


Abbildung 1: Vorgehen in Phase 1 - Requirement-Analysis

Stakeholder Analyse

Im Folgenden werden die einzelnen Stakeholder des Systems identifiziert und in ihrer Funktion kurz beschrieben.

Tabelle 1 : Stakeholder „Wähler“

| Rolle des Stakeholders | Benutzer: Wähler |
|------------------------|---|
| Beschreibung | Wähler möchten ihre politische Ansicht mit den Positionen der Kandidaten vergleichen. Allerdings müssen Wähler viel Zeit investieren, um sich mit diesen Positionen auseinanderzusetzen und den für sich optimalen Kandidaten zu finden. Ihr Recht auf Privatsphäre und Datenschutz muss in diesem System speziell berücksichtigt werden. Das System soll die Wähler bei ihrer Entscheidung unterstützen. |
| Wissensgebiet | Unterschiedlich |
| Begründung | Da die Wähler Anwender des Systems sind, muss das System auf ihre Bedürfnisse und Anforderungen angepasst werden. |

| | |
|---|------|
| Grad der Mitwirkung während der Qualitätssicherung und Analyse | Hoch |
| Entscheidungsbefugnis | Hoch |

Tabelle 2 : Stakeholder „Kandidat“

| | |
|---|---|
| Rolle des Stakeholders | Benutzer: Kandidat |
| Beschreibung | Kandidaten haben verschiedene Positionen und stellen sich den Wählern zur Wahl. Dabei beantworten sie eine Auswahl an Thesen und geben Begründungen zu diesen Thesen ab. Die weniger bekannten Kandidaten haben es bei einem Wahlkampf besonders schwer, da der Fokus der Medien die Spitzenkandidaten der Parteien in den Vordergrund stellt. Das System soll allen Kandidaten den Kontakt zu den Wählern erleichtern und ihnen einen besseren Überblick auf die Wünsche der Wähler geben. |
| Wissensgebiet | Unterschiedlich |
| Begründung | Da die Kandidaten Anwender des Systems sind, muss das System auf ihre Bedürfnisse und Anforderungen angepasst werden. |
| Grad der Mitwirkung während der Qualitätssicherung und Analyse | Hoch |
| Entscheidungsbefugnis | Hoch |

Nachfolgend sollen die identifizierten Stakeholder in ihrer relevanten Beziehung zum System beschrieben werden, um daraus entsprechende Anforderungen an das System und Risiken abzuleiten. Die Tabelle Nr 3. der Stakeholder Analyse befindet sich im Anhang ab der Seite **XY**.

Benutzermodellierung

Aus den Erkenntnissen der Domänenrecherche und der Stakeholder Analyse, sollen an dieser Stelle mithilfe von „User profiles“ angemessene Merkmale der Stakeholder auf Basis der Zielsetzung des Projekts ermittelt werden. Im Anhang auf den Seiten 52 bis 57 befinden sich die dazugehörigen User profiles der einzelnen Stakeholder.

Tabelle 3 : Definition der User Profiles

| Merkmal | Ausprägung |
|--|---|
| Demographische Charakteristiken | Alter, Geschlecht, Wohnort und Art des Wohnens, sozial-ökonomischer Status |
| Formale Qualifikationen | Informationen über Berufsausbildungen, Qualifikationen, Fortbildungen |
| Fachwissen | Kenntnisse im Nutzungskontext und den für die Entwicklung relevanten Bereichen |
| Fähigkeiten und Einschränkungen | Körperliche Einschränkungen und Fähigkeiten, Behinderungen; Sozialwesen |
| Verfügbare Technologien | Hard- und Software |
| Computerkenntnisse und -erfahrungen | Kenntnisse im Umgang mit dem System |
| Spezielle Produkterfahrung | Erfahrungen mit speziellen Produkten innerhalb der Domäne; Kenntnisse über entsprechende Technologien |
| Motivation und Einstellung | Vorlieben, Ängste, Motive und Motivationen im Nutzungskontext |
| Aufgaben | Zu erfüllende Aufgaben in Bezug auf den Nutzungskontext |
| Auswirkungen von Fehlern | Mögliche Auswirkungen von menschlichem Fehlverhalten in Bezug auf das System |

Anforderungen

Aus den Erkenntnissen der Marktrecherche, der Domänenrecherche sowie der Benutzermodellierung werden nun entsprechend funktionale Anforderungen (Systemaktivitäten und Funktionen), organisationale Anforderungen (Prozess und Kontrolle), qualitative Anforderungen (Leistung, Sicherheit und Standards), Anforderungen an die Benutzerschnittstelle und technische Anforderungen formuliert. Diese Anforderungen wurden im Verlauf des Projektes iterativ überarbeitet und bilden die Bearbeitungsgrundlage für die Gestaltung und Entwicklung des Systems. Die funktionalen Anforderungen wurden um die Anforderungen Nr. 10 bis Nr. 16 erweitert.

Funktionale Anforderungen

1. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten sich auf diesem System anhand gewisser Merkmale zu registrieren und diese Merkmale gegeben falls zu ändern oder zu löschen.
2. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten neue Thesen in einer Kategorie für ihren Wahlkreis auf diesem System zu veröffentlichen und vorhandene Thesen zu sehen, falls sie beim System authentifiziert sind.
3. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten zu einer These eine Position (pro/neutral/contra) auf diesem System zu beziehen, falls sie beim System authentifiziert sind.
4. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten eine Begründung ihrer Position zu einer These auf diesem System zu veröffentlichen und gegeben falls ihre Begründung und Position zu ändern, falls sie beim System authentifiziert sind.
5. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten Thesen und Begründungen einer Position zu einer These auf diesem System zu bewerten, falls sie beim System authentifiziert sind.
6. Version 2: Das System muss den Wählern die Möglichkeit bieten ihre persönlichen Ansichten zu einer oder mehreren Thesen mit den Positionen aller Kandidaten ihres Wahlkreises auf diesem System zu vergleichen und es muss die durchschnittlichen Punkte der Kandidaten pro These zu einer Kategorie berechnen.
7. Version 1: Das System muss den Wählern die Möglichkeit bieten private Diskussionen über Thesen und die Begründungen der Nutzer zu den Positionen auf diesem System zu führen, falls sie beim System authentifiziert sind.
8. Version 1: Das System muss die Informationen für die Benutzerschnittstelle geeignet darstellen.
9. Version 1: Das System muss den Stakeholdern die Möglichkeit bieten alle Thesen des Wahlkreises zu durchsuchen.
10. Version 1: Die Thesen sollten nach bestimmten Kategorien gefiltert aufgelistet werden.
11. Version 1: Die Thesen müssen nach Aktualität, Beliebtheit oder nach unpositionierten Thesen sortiert werden können.
12. Version 1: Das System muss die Statistiken des Wahlkreises berechnen und darstellen können.
13. Version 1: Die Wähler sollten Kandidaten oder Thesen abonnieren können, sodass Sie über neue Informationen bezüglich dieser abonnierten Thesen oder Kandidaten informiert werden können.
14. Version 1: Die Wähler und Kandidaten sollten über neue Informationen zu eigenen Thesen oder Begründungen benachrichtigt werden.

15. Version 1: Das System muss verhindern, dass eine These mehr als einmal in einem Wahlkreis veröffentlicht werden kann.
16. Version 1: Das System muss den Kandidaten die Möglichkeit bieten, Informationen bezüglich ihrer Biographie zu ihrem Profil hinzuzufügen.

Organisationale Anforderungen

1. Version 1: Der Kontext des Projektes richtet sich an Wahlen aus.
- 2.

Qualitative Anforderungen

1. Version 1: Das System soll Daten zuverlässig und fehlerfrei verarbeiten.
2. Version 1: Das System soll den Erfordernissen der Stakeholder gerecht werden.
3. Version 1: Das System muss die Datenbestände persistent und korrekt verwalten.
4. Version 1: Die Funktionen des Systems sollen von den Stakeholder nutzbar sein.
5. Version 1: Fehlerhafte Eingaben der Nutzer sollen nicht zu Systemabstürzen oder fehlerhaften Daten in der Datenbank führen.
6. Version 1: Das System muss die privaten Daten der Wähler schützen.

Anforderungen an die Benutzerschnittstelle

1. Version 1: Die Benutzerschnittstelle des Systems soll hinsichtlich der Farbgebung und der Textgröße lesefreundlich gestaltet werden.
2. Version 1: Die Benutzerschnittstelle des Systems soll einen geringen Komplexitätsgrad der Interaktion aufweisen.
3. Version 1: Die Benutzerschnittstelle des Systems soll gebrauchstauglich hinsichtlich der Ansprüche der Stakeholder sein.
4. Version 1: Die Benutzerschnittstelle des Systems soll Thesen und Begründungen der Positionen der Nutzer einfach und übersichtlich darstellen.
5. Version 1: Die Benutzerschnittstelle des Systems soll dem Benutzer Feedback über erfolgreiche und erfolglose Interaktionen geben und dem Benutzer eine Lösung anbieten.

Technische Anforderungen

1. Version 1: Der Server des Systems soll als Schnittstelle für die Kommunikation REST verwenden.
2. Version 1: Das System soll nach den Vorgaben des Architekturdiagramms entwickelt werden.
3. Version 1: Die Kommunikation zwischen Client und Server soll über eine sichere HTTPS Verbindung stattfinden.
4. Version 1: Das System soll für die Repräsentation der Daten, welche zwischen Client und Server übertragen werden, das Datenformat JSON verwenden.
5. Version 1: Das System soll für die asynchrone Kommunikation zwischen Server und Client *Googles Firebase Cloud Messaging* verwenden.

Task-Analysis

Laut Mayhew sollte sich eine „Contextual Task Analysis“ bei einer Webanwendung mehr auf die Bedürfnisse und Wünsche der Benutzer konzentrieren, anstatt wie sie derzeit Aufgaben erledigen, da die zu erledigenden Aufgaben so noch nicht von den geplanten Benutzern ausgeführt werden (vgl. Mayhew, 1999, S. 104). Derzeit ist es für die Wähler nicht möglich Monate vor der Wahl zu wissen, welche Kandidaten in seinem Wahlkreis zur Wahl antreten möchten, aber um die Kommunikation zwischen Wählern und Kandidaten zu verbessern, sollte dem Wähler diese Informationen zur Verfügung stehen. Eine detaillierte, deskriptive Aufgaben-Analyse kann bei diesem Projekt nicht durchgeführt werden, da in der Domäne eine Vielfalt an Möglichkeiten besteht seine Meinung über einen Kandidaten zu bilden oder mit ihnen zu kommunizieren. Die verschiedenen Möglichkeiten der Kommunikation sind im deskriptiven Kommunikationsmodell (siehe Seite 30) beschrieben. Die Bedürfnisse und Erwartungen der Benutzer sind in der Stakeholder Analyse (siehe Anhang Seite 58) verfasst.

Generelle Designprinzipien

Die Abkürzung der generellen Designprinzipien sieht vor die Anzahl der Literaturhinweise zu limitieren, um den Aufwand gering zu halten. Für das Projekt wurden die Designprinzipien Visuelle Hierarchie, Hick's Law, Fitt's Law und die „Material design Guidelines“ von Google festgelegt. (siehe Seite 60)

Möglichkeiten und Einschränkungen der Plattform

Die Möglichkeiten und Einschränkungen der Plattform werden weitgehend ausgelassen, da dies die Abkürzung nach Mayhew vorsieht (vgl. Mayhew, 1999, S. 155) und das Projekt mit einem kleinen Team durchgeführt wird.

Usability Goals

Trotz der kurzen Projektdauert sollten Usability Goals definiert werden, um den Designvorgang und das Testen zu vereinfachen. Die Abkürzung der Usability Goals laut Mayhew sieht vor, eine weniger umfangreiche Recherche zur Bestimmung von Zielen durchzuführen und die Ziele weniger formal zu dokumentieren. Außerdem empfiehlt Mayhew für Webanwendungen den Fokus auf „Ease-of-Learning“ statt auf „Ease-of-Use“ zu setzen, da viele Benutzer möglicherweise die Anwendung nicht täglich oder nur einmal benutzen (vgl. Mayhew, 1999, S. 139).

Qualitative Usability Goals

1. Das Design soll minimalistisch gestaltet werden, so dass der Benutzer sich mit geringen Aufwand zurechtfinden kann.
2. Das Design soll auf mögliche Fehler des Benutzers bei der Interaktion mit dem System hinweisen und eine Lösung anbieten.
3. Das Design soll die Zufriedenheit, die Motivation und den Spaß des Benutzers bei der Interaktion mit dem System fördern.

Quantitative Usability Goals

1. Novizen (werden definiert als Erstbenutzer) sollen innerhalb einer Minute wissen, wie sie eine These veröffentlichen können.
2. Novizen sollen innerhalb einer Minute wissen, wie man sich zu einer These positionieren kann.
3. Novizen sollen innerhalb einer Minute wissen, wie man eine Begründung zu einer These hinzufügen kann.
4. Novizen sollen innerhalb einer Minute wissen, wie man Begründungen und Thesen bewerten und kommentieren kann.
5. Novizen sollen innerhalb einer Minute wissen, wie sie die Berechnung der Übereinstimmung starten können.

Produkt Styleguide

Bei einem größeren Projekt sollte im Verlauf des Vorgehensmodells nach jedem Bearbeitungsschritt der Produkt Styleguide ergänzt werden, um zentrale Erkenntnisse zu sammeln. Die Abkürzung von Mayhew sieht bei kleinen Projekten vor diesen Schritt auszulassen und stattdessen gute Notizen zu führen. (vgl. Mayhew, 1999, S.320)

Phase 2: Design Testing Development

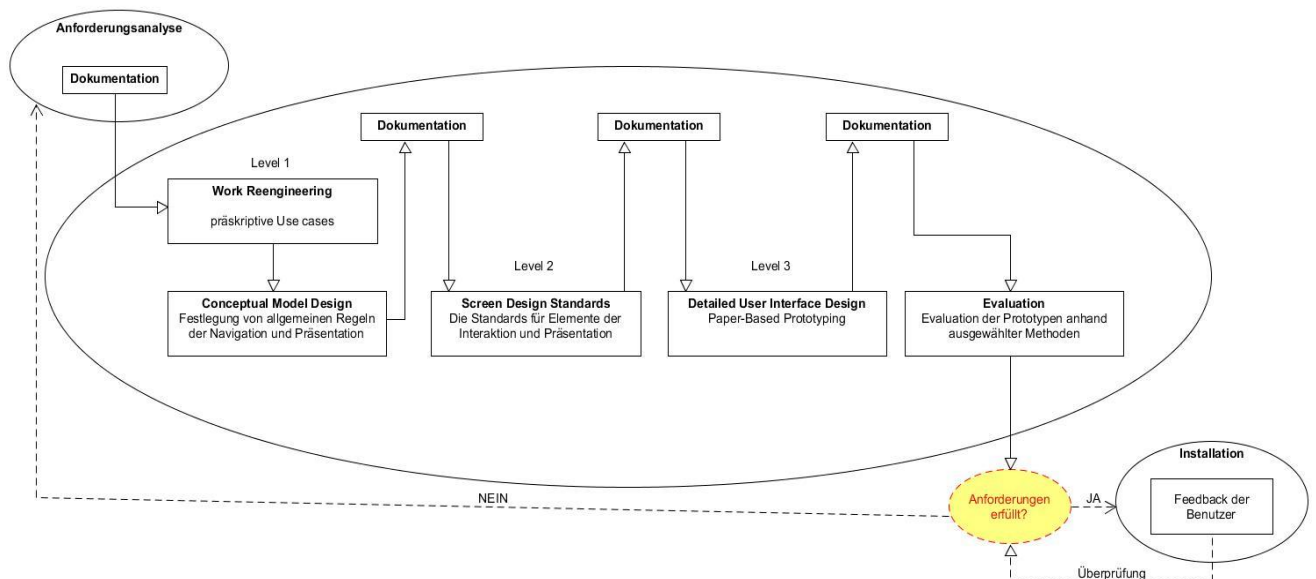


Abbildung 2: Vorgehen in Phase 2 – Design / Testing / Development

Normalerweise sollte jedes der drei Design Level jeweils iterativ evaluiert werden, aber aufgrund des kurzen Projektzeitraums und da das ganze Team während des gesamten Designprozesses involviert ist, werden erst nach Abschluss des gesamten Designprozesses die Prototypen evaluiert. Anschließend wird überprüft ob die Anforderungen und die Usability Goals erfüllt worden sind. Falls dies nicht der Fall ist, wird der Gesamtprozess nochmal durchlaufen.

Level 1 Work Reengineering

Bei diesem Schritt sieht der „Usability Engineering Lifecycle“ von Mayhew vor die Arbeit der Benutzer neu zu modellieren. Die Abkürzung dieses Schrittes sieht vor weniger formal die Use Cases zu dokumentieren. (vgl. Mayhew, 1999, S.182)

Da bei diesem Projekt die Kommunikation zwischen Wählern und Kandidaten, sowie die Meinungsbildung der Wähler über Kandidaten neu entwickelt statt überarbeitet wird, konzentriert sich der Fokus des Work Reengineering auf die Ergebnisse der Marktrecherche. Aus diesen Erkenntnissen wurden präskriptive Use Cases entwickelt, welche den Arbeitsablauf mit dem zu entwickelnden System beschreiben sollen und sich im Anhang auf den Seiten 61-71 befinden.

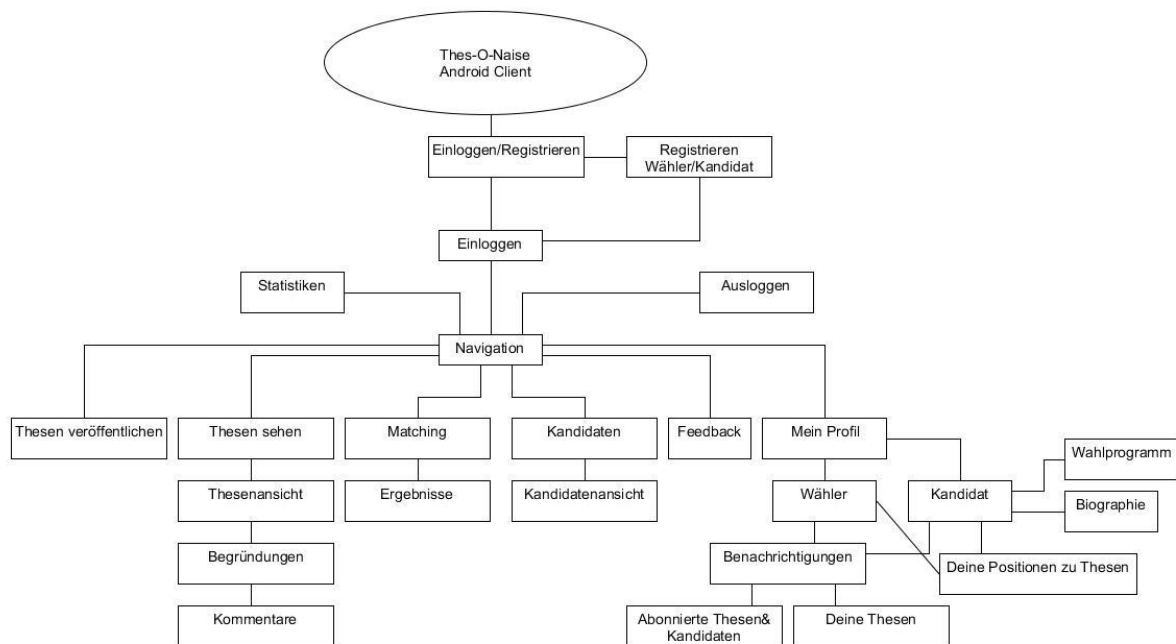


Abbildung 3: Re-engineering Task Organization Model

In der Abbildung oben wird die Reihenfolge der Aufgaben, die in den präskriptiven Use Cases beschrieben wurden, in dem Re-engineering Task Organization Model zusammengefasst.

Level 1: Conceptual Model Design

Aus den Erkenntnissen des Work Reengineering werden nun die Regeln für die grundlegende Struktur und Organisation der gesamten „user interface architecture“ festgelegt, um ein einheitliches Design zu garantieren. Aus dem Re-engineering Task Organization Model resultieren klar identifizierbare Prozesse, wodurch nun der prozessorientierte Ansatz verfolgt wird. Diese Regeln wurden in einer Tabelle konkretisiert, und werden als Grundlage für zukünftige Designentscheidungen dienen. (vgl. Mayhew, 1999, S.187)


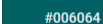






Tabelle 4 : Regeln des Conceptual Model Designs

| Bereich | Regel |
|-------------------------|---|
| Logo | Im ersten Screen nach dem Start wird das Logo etwas größer mittig platziert; Das Logo soll in der Navigationsansicht oben, zentral und etwas kleiner platziert werden; |
| Navigation | Soll in die Bereiche „Mein Profil“, „These veröffentlichen“, „Thesen sehen“, „Matching“, „Kandidaten“, „Statistiken“ und „Ausloggen“ unterteilt werden |
| Kategorien | Thesen werden in Themengebieten kategorisiert um einen besseren Überblick zu schaffen |
| Begründungen von Thesen | Begründungen von Thesen sollen in „Pro“, „Neutral“ und „Contra“ eingeteilt werden, um sich klar und sichtbar positionieren zu können; |
| Visuelle Hinweise | Die Navigationsleiste soll den Benutzer bei der Orientierung helfen; Die Unterteilung der Begründungen in „Pro“, „Neutral“ und „Contra“ zeigt den Benutzer deutlich bei welcher Position er sich befindet; |

Level 2: Screen Design Standards

Als nächster Schritt im Designprozess sollen nun Standards für Elemente eingeführt werden, mit denen interagiert werden kann oder die entsprechend auf dem Bildschirm dargestellt werden. Durch diese Screen Design Standards werden einzelne Elemente des Layouts genauer definiert, wodurch Einfachheit und Einheitlichkeit bei den Layouts erzielt werden soll. Dabei müssen die Usability Goals und die generellen Designprinzipien berücksichtigt werden. (vgl. Mayhew, 1999, S.173)

Tabelle 5 : Screen Design Standards

| | |
|----------------------|---|
| Schriften | <p>Schriftart: Serifenlose Schriften, z.B. Arial oder Calibri</p> <p>Schriftgröße: 12 pt / 14 pt</p> <p>Schriftschnitt: Medium / bold</p> <p>Thesen werden durch eine größere Schrift hervorgehoben.</p> |
| Farben | <p>Color Primary: #0097a7 </p> <p>Color Primary Dark: #006064 </p> <p>Color Accent: #18ffff </p> |
| Hintergründe | <p>Die Schrift braucht zum Hintergrund einen starken Kontrast für die Lesbarkeit. Es wurde die Farbe #80deea  mit einer schwarzen Schrift gewählt;</p> <p>Bei der Registrierung, wird der Hintergrund vom Typ Wähler in der Farbe #84ffff  gehalten und der Hintergrund vom Typ Kandidat in der Farbe #00b8d4 . Dies hilft dabei sichtbar beide Typen direkt voneinander zu unterscheiden;</p> <p>Es werden keine gemusterte Hintergründe verwendet;</p> |
| Buttons/Radiobuttons | <p>Das Design der Buttons soll sich an das „Material design“ von Google orientieren</p> <p>Button Design: </p> <p>Button Farbe: </p> |
| Terminologie | Stil: Informationen sollen klar, vertraut und positiv präsentiert werden |
| Spinner | Das Design der Spinner soll sich an das „Material design“ von Google orientieren |
| Navigation | Das Design der Navigation soll sich an das „Material design“ von Google orientieren |

Level 3: Detailed User Interface Design

Es wurde mehrere „Paperbased“ Prototypen entwickelt mit verschiedenen Varianten von der Auflistung der Thesen, der Ansicht einer These und der Ansicht eines Kandidatenprofils. Außerdem wurden folgende Ansichten als „Paperbased“ Prototyp entwickelt: die Ansicht der Veröffentlichung einer These, die Ansicht des Wählerprofils, die Ansicht der Kandidatenliste und die Ansicht der Ergebnisse des Matching.

Varianten der Ansicht der Thesen und der Thesenliste

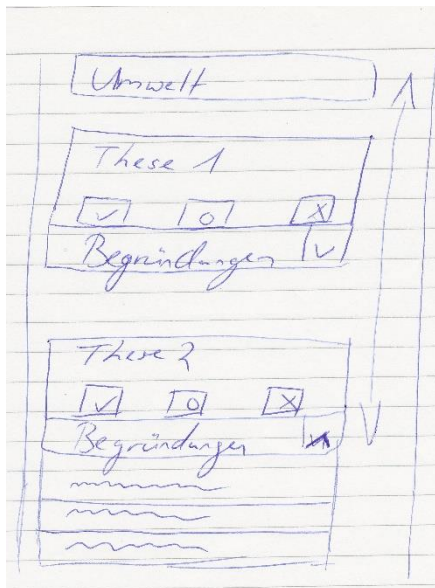


Abbildung 4: Variante 1 – Thesenliste
Variante 1 – Ansicht einer These

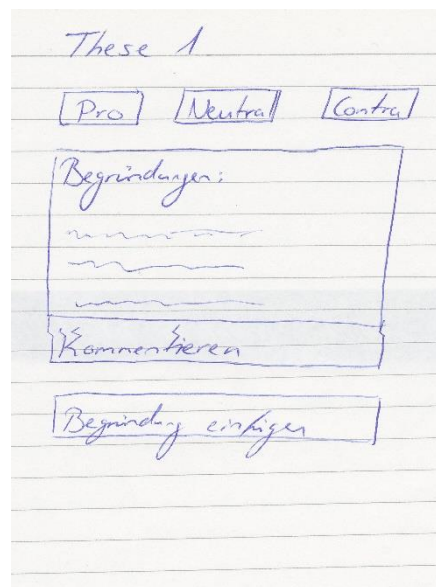


Abbildung 5:

Bei der ersten Variante der Ansicht der Thesenliste (siehe Abbildung 4) werden die Positionen Pro, Neutral und Contra als Imagebutton dargestellt. Dies ist ungeeignet da nicht genau zu erkennen ist ob es sich um Positionen handelt. Da die Begründungen ausklappbar sind, wirkt das Layout überladen und verletzt somit das Qualitative Usability Goal Nr. 1. Außerdem wurde das Projekt im Verlauf um mehrere funktionale Anforderungen erweitert, welche bei dieser Variante noch nicht berücksichtigt wurden. (Nr.11, Nr.13)

Bei ersten Variante der Ansicht einer These (siehe Abbildung 5) ist nicht genau zu erfassen, wo und wie Kommentare sich einordnen.

(Funktionale Anforderungen Nr.3 und Nr.4)

Abbildung 6: Variante 2 – Thesenliste

Abbildung 7: Variante 3 – Thesenliste

Die dritte Variante der Thesenliste (siehe Abbildung 7) zeigt im oberen Bereich den „Navigationsbutton“, eine Suchleiste, um anhand von Begriffen nach Thesen zu suchen, darunter mehrere „ToggleButtons“ mit den Thesen Sortieroptionen und darunter einen Spinner, um die Thesen nach einer Kategorie zu filtern. Darunter wird die Liste mit den einzelnen Thesen dargestellt, wobei jede These am rechten Rand einen „Mehr“ Button (genauere Ansicht einer These) und einen „Folgen“ Button hat. Letzterer dient dazu eine These zu abonnieren, damit der Benutzer über neue Informationen bezüglich der These benachrichtigt werden kann. Der Text der These beginnt im Thesenkasten oben rechts. Unterhalb des Textes der These befindet sich der „Like“ Button (wenn die These gefällt) und die Anzahl der Likes neben dem Button. Damit sich der Benutzer zur These positionieren kann, befinden sich 3 Radiobuttons (Pro, Neutral, Contra) am unteren Rand des Thesenkastens. (Funktionale Anforderungen Nr.3, Nr.9, Nr.10, Nr.11, Nr.13)

Abbildung 8: Variante 4 – Thesenansicht kombiniert mit Thesenliste

21

wischen zum Bildschirmrand hin ausgewechselt werden kann. Dieses Verfahren ist zu umständlich um mehrere Thesen nacheinander durchzulesen und sich schnell zu positionieren. Außerdem verletzt diese Variante das Qualitative Usability Goals Nr. 1. (Funktionale Anforderung Nr.3, Nr.9, Nr.10, Nr.11, Nr.13)

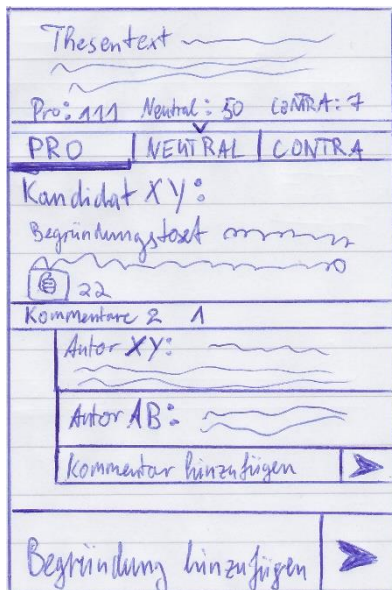


Abbildung 9: Variante 2 - Ansicht einer These

Wenn der „Mehr“ Button bei der Variante 3 in der Ansicht der Thesenlist gedrückt wurde, wird diese Ansicht einer These dargestellt (siehe Abbildung 9). Im oberen Bereich beginnt der Thesentext und darunter die Anzahl wie viele der Benutzer sich zu Pro, Neutral oder Contra positioniert haben. Dieser Teil ist mit einem Imagebutton (Pfeil nach unten) einklappbar. Im mittleren Bereich befindet sich die Tableiste der Positionen (Pro, Neutral, Contra), die anzeigt zu welcher Position Begründungen und Kommentare vorhanden sind. Um die Position der Begründungen zu wechseln, kann einerseits der Tab der Position angeklickt werden und andererseits kann dies auch mittels wischen zum Bildschirmrand hin geschehen. Unterhalb dieser Tableiste befindet sich die Liste der Begründungen, wobei die Begründungen der Kandidaten über den Begründungen der Wähler stehen. Jede Begründung enthält den Namen des Autors und den Text der Begründung, sowie einen „like“ Button und die Anzahl der „Likes“. Unter einer Begründung steht die Anzahl der Kommentare in einer Leiste. Mittels eines Klicks auf diese Leiste, fahren die Kommentare zu der jeweiligen Begründung heraus. Die Kommentare werden rechts unter der Begründung eingerückt, um besser unterscheiden zu können zu welcher Begründung die Kommentare jeweils gehören. Unter dem letzten Kommentar befindet sich das Eingabefeld, um einen weiteren Kommentar hinzuzufügen. Am unteren Rand befindet sich ein Eingabefeld, um eine weitere Begründung der Position zur These hinzuzufügen.

Fazit der Ansicht der Thesen

Damit die Benutzer schnell mehrere Thesen durchlesen und sich einfach zu den Thesen positionieren können, wurde sich für die Variante 3 der Thesenlisten Darstellung entschieden. Wenn der Benutzer die Begründungen der Positionen von Kandidaten oder Wähler und die Anzahl der Positionen der Wähler zur konkreten Thesen sehen möchte, wird er dann per Klick auf den „Mehr“ Button zur Thesen Ansicht weitergeleitet. Diese soll dann wie in der Variante 2 (siehe Abbildung 9) dargestellt werden.

Varianten der Ansicht eines Kandidatenprofils

Abbildung 10: Variante 1- Ansicht des Kandidaten Profils

Bei der ersten Variante der Ansicht eines Kandidaten Profils (siehe Abbildung 10) werden allgemeine Informationen des Kandidaten zentriert untereinander dargestellt, was keine platzsparende Alternative bietet. Unterhalb dieser Informationen befindet sich eine Tableiste mit den Tabs „Lebenslauf“, „Positionen“ und „Wahlprogramm“. Es fehlt die Möglichkeit die allgemeinen Informationen mit einem Button einzuklappen, wodurch weniger Platz für die Informationen der Tabs zur Verfügung steht.

Abbildung 11: Variante 2- Ansicht des Kandidaten Profils mit ausgewähltem Lebenslauf

Abbildung 12: Variante 2- Ansicht des Kandidaten Profils mit der ausgewählten Ansicht der Positionen des Kandidaten

Abbildung 13: Variante 2- Ansicht des Kandidaten Profils mit der ausgewählten Ansicht der Begründungen des Kandidaten

Die zweite Variante (siehe Abbildung 11 bis 14) hat eine erweiterte Tableiste mit dem Lebenslauf, Positionen, Begründungen und das Wahlprogramm des Kandidaten, diese Informationen können durch Wischen des Screens gewechselt werden. Zudem lässt sich der obere Informationsteil über den Kandidaten mittels einem Button ein- und ausklappen. Diese Anordnung bietet eine gute Übersicht über alle relevanten Informationen. (Funktionale Anforderung Nr.16)

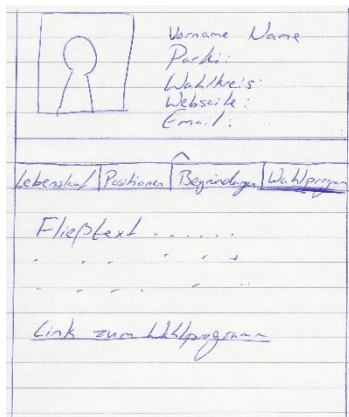


Abbildung 14: Variante 2- Ansicht des Kandidaten Profils mit der ausgewählten Ansicht des Wahlprogrammes des Kandidaten

Fazit der Ansicht eines Kandidatenprofils

Die Variante 2 ...

Ansichten der Kandidatenliste, des Wählerprofils, der Ergebnisliste und der Veröffentlichung von Thesen

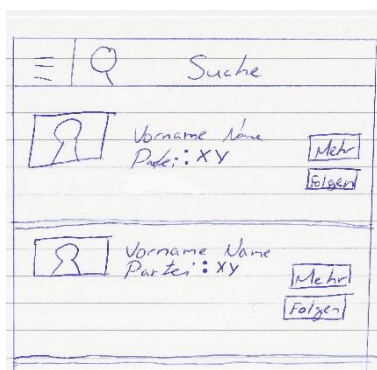


Abbildung 15: Ansicht der Kandidatenlisten

Die Abbildung 15 zeigt die Ansicht der Auflistung der einzelnen Kandidaten des Wahlkreises. Jeder Kandidat wird mit seinem Profilbild, seinem Vor – und Nachnamen und seiner Partei dargestellt. Auf der rechten Seite befindet sich der „Mehr“ Button (Ansicht des Kandidatenprofils) und der „Folgen“ Button. Außerdem kann mittels einer Suchleiste auch nach bestimmten Kandidaten gesucht werden. (Funktionale Anforderung Nr.13)

| | |
|---|---|
| Username Wahlkreis: XY Bearbeiten | Username Wahlkreis: XY Bearbeiten |
| Meine Thesen Meine Positionen Abonniert | Meine Thesen Meine Positionen Abonniert |
| Kandidat XY hat sich PRO zur These positioniert : Thesentext... Mehr | Kandidat XY hat sich zur NEUTRAL zur These positioniert : Thesentext... Mehr |
| Kandidat XY hat eine Begründung PRO zur These hinzugefügt : Thesentext... Mehr | Kandidat hat eine Begründung PRO zur These hinzugefügt : Thesentext... Mehr |

| |
|--|
| Username Wahlkreis: XY Bearbeiten |
| Meine Thesen Meine Positionen Abonniert |
| Deine Position: Pro Thesentext... Mehr <input checked="" type="radio"/> Pro <input type="radio"/> Neutral <input type="radio"/> Kontra |
| Deine Position: Neutral Thesentext... Mehr <input type="radio"/> Pro <input checked="" type="radio"/> Neutral <input type="radio"/> Kontra |

Abbildung 16: Ansicht des Wählerprofils

Die Abbildung 16 zeigt die Ansicht des Wählerprofils. Im oberen Bereich werden die allgemeinen Informationen des authentifizierten Wählers angezeigt, wobei dieser Bereich mit einem Button ein- und ausklappbar ist. Falls der Wähler seinen Wahlkreis, seinen Benutzernamen oder seine Email-Adresse ändern möchte, kann dies durch klicken des „Bearbeiten“ Button geschehen.

Unter dem oberen Bereich befindet sich eine Tableiste mit den Tabs „Meine Thesen“, „Meine Positionen“ und „Abonniert“, wobei man mittels Wischen vom Bildschirmrand aus und klicken auf einen Tab zwischen den einzelnen Tabs wechseln kann. Bei „Meine Thesen“ kann der Wähler seine veröffentlichten Thesen sehen und erhält Benachrichtigungen zu diesen, „Meine Position“ zeigt die Thesen zu denen der Wähler sich positioniert hat und „Abonniert“ zeigt die neusten Informationen über die abonnierten Thesen oder Kandidaten. (Funktionale Anforderung Nr.14)

| Kategorie: Insgesamt | |
|--|------|
| Vorname Nachname | |
| Partei: xy | |
| Abstand zu ihren Positionen: 28 | Mehr |
| Anzahl Positionen: 30 | |
| Durchschnittliche Punkte pro These: 0,66 | |
| | |
| Vorname Nachname | |
| Partei: xy | |
| Abstand zu ihren Positionen: 25 | Mehr |
| Anzahl Positionen: 30 | |
| Durchschnittliche Punkte pro These: 0,83 | |
| | |
| Vorname Nachname | |
| Partei: xy | |
| Abstand zu ihren Positionen: 30 | Mehr |
| Anzahl Positionen: 30 | |

Abbildung 17: Ansicht der Ergebnisse des „Matching“

Die Abbildung 17 zeigt, wie die Ergebnisse des „Matching“ dargestellt werden. Im oberen Bereich unter der Navigationsleiste befindet sich die Auswahl der Kategorie mittels eines Spinners, wobei standardmäßig das Gesamtergebnis aller Kandidaten dargestellt wird. Unter der Kategorie werden die Kandidaten aufgelistet, wobei zu jedem Kandidaten jeweils der Namen, die Partei, der Abstand zu den eigenen Positionen, die Anzahl der Positionen, die Durchschnittlichen Punkte pro These und rechts am Rand einen „Mehr“ Button für die Ansicht des Kandidaten dargestellt wird. Sortiert werden die Kandidaten aufsteigend nach absoluten Punktzahlen. Der berechnete Durchschnitt dient als weitere Entscheidungshilfe. Wenn ein Kandidat einen kleineren Abstand zu den Positionen des Wählers aufweist, aber weniger Thesen beantwortet hat und im Gegenzug ein anderer Kandidat bei dem gleichen Abstand der Punkte mehr beantwortete Thesen hat, würde dieser Kandidat näher an den Positionen des Wählers liegen. (Funktionale Anforderung Nr.6)

The sketch shows a mobile application screen with a white background and a blue border. In the top left corner, there is a hamburger menu icon (three horizontal lines). Below it, there is a large rectangular text input field with the handwritten label 'Theseneingabefeld'. Underneath the input field is a category spinner showing 'Kategorie: Lokal' with a downward arrow. At the bottom of the form is a rectangular button labeled 'Veröffentlichen'.

Abbildung 18: Ansicht der Veröffentlichung neuer Thesen

Die Abbildung 18 zeigt die Darstellung, wie ein Benutzer eine neue These in einer Kategorie veröffentlichen kann. Im oberen Bereich befindet sich ein Eingabefeld für die These. Unter diesem Eingabefeld befindet sich ein Spinner zur Auswahl der Kategorie der These. Durch das Klicken des „Veröffentlichen“ Buttons wird die These zum Server verschickt und im Falle einer erfolgreichen Veröffentlichung erscheint eine Meldung darüber. Falls der „Veröffentlichen“ Button betätigt wird, ohne dass ein Text im Eingabefeld geschrieben wurde, erscheint eine Meldung, die den Benutzer auffordert einen Text einzugeben. Falls der Benutzer versucht eine These zu veröffentlichen, welche schon in seinem Wahlkreis veröffentlicht wurde, erscheint ebenfalls eine Meldung, die den Benutzer darauf hinweist.

Evaluation

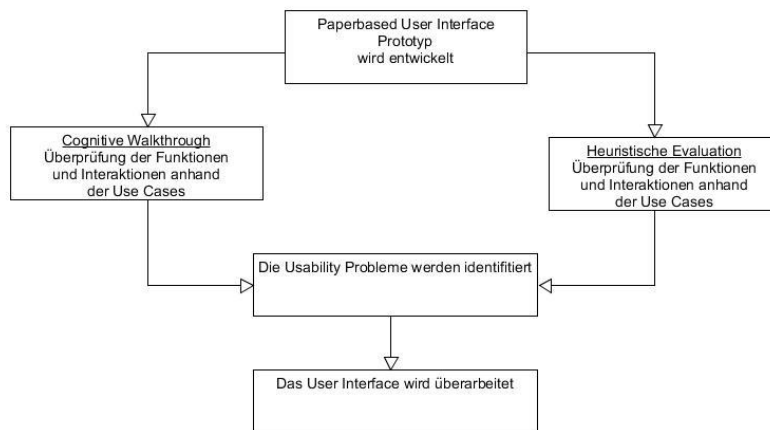


Abbildung 19: Vorgehen bei der Evaluation

Im nächsten Schritt wird eine Evaluation auf Basis des User Interface-Prototypen durchgeführt. Es werden zwei Evaluations Methoden angewendet: Zum einen der „Cognitive Walkthrough“ (vgl. Polson, Lewis, Rieman, & Whartson, 1992) der den Fokus auf den Benutzer im Umgang mit dem Interface legt und zum anderen die „Heuristische Evaluation“ nach Nielsen. Diese beiden Methoden werden zusammengelegt und jeweils pro Use Case angewendet (siehe Seiten 72-79). Die Tabelle zu den Ergebnissen der Evaluation befindet sich im Anhang auf Seite 80. Nachdem die Evaluations Methoden angewendet wurden, werden die Änderungen in der Reihenfolge der wichtigsten Vorgehensschritte umgesetzt. Das User Interface wird anschließend iterativ verbessert und optimiert.

Cognitive Walkthrough

Benutzercharakteristiken

Die Stakeholder dieses Systems sind Wähler und Kandidaten. Genauere Informationen liegen in den User Profiles vor. (siehe Seiten 52-57)

Beispielaufgaben

In den präskriptiven Use Cases wurden die Aufgaben definiert:

1. USE CASE: Eine These veröffentlichen
2. USE CASE: Eine Position zu einer These beziehen
3. USE CASE: Eine Begründung zu einer These hinzufügen
4. USE CASE: Kommentar zu einer Begründung hinzufügen
5. USE CASE: Die Übereinstimmung berechnen lassen
6. USE CASE: Thesen bewerten
7. USE CASE: Thesen sortieren
8. USE CASE: Thesen nach Begriffen durchsuchen

Untersuchung der Handlungssequenzen

Es wird davon ausgegangen, dass der optimale Lösungsweg gefunden wurde um die Handlungssequenzen zu untersuchen. Es sollen nach jedem Schritt folgende Fragen beantwortet werden:

1. Wird der Benutzer versuchen, den richtigen Effekt zu erzielen?
2. Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht?
3. Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt?
4. Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen?

Bei einer verneinten Frage soll das Usability Problem in der Tabelle (siehe Anhang Seite 80) protokolliert werden.

Heuristische Evaluation nach Nielsen

In der zweiten Evaluations Methode sollen die „Paperbased“ Prototypen mit den Heuristiken nach Nielsen evaluiert werden. Dabei wird bei den präskriptiven Use Cases überprüft, ob der Benutzer alle Aufgaben durchführen kann und ob alle Heuristiken erfüllt sind. Falls eine Heuristik verletzt wurde, steht dies nach dem jeweiligen Schritt des Use Cases. Die folgenden Heuristiken werden angewendet:

1. **Visibility of system status** (Sichtbarkeit des Systemzustandes)
2. **Match between system and the real world** (Übereinstimmung zwischen System und der Realwelt)
3. **User control and freedom** (Benutzerkontrolle und Zufriedenheit)
4. **Consistency and standards** (Konsistenz und Standards)
5. **Error prevention** (Fehlerprävention)
6. **Recognizing rather than recall** (Erkennen ist besser als Erinnern)
7. **Aesthetic and minimalist design** (Ästhetik und minimales Design)
8. **Help users recognize, diagnose, and recover from errors** (Hilfe für den Anwender beim Erkennen, Diagnostizieren und Rückgängig machen von Fehlern)

WBA Teil des Projektes

Kommunikationsmodelle

Nachfolgend wird der Informationsaustausch zwischen den Stakeholdern im Ist- und Sollzustand anhand von Diagrammen verdeutlicht und beschrieben.

Deskriptives Kommunikationsmodell

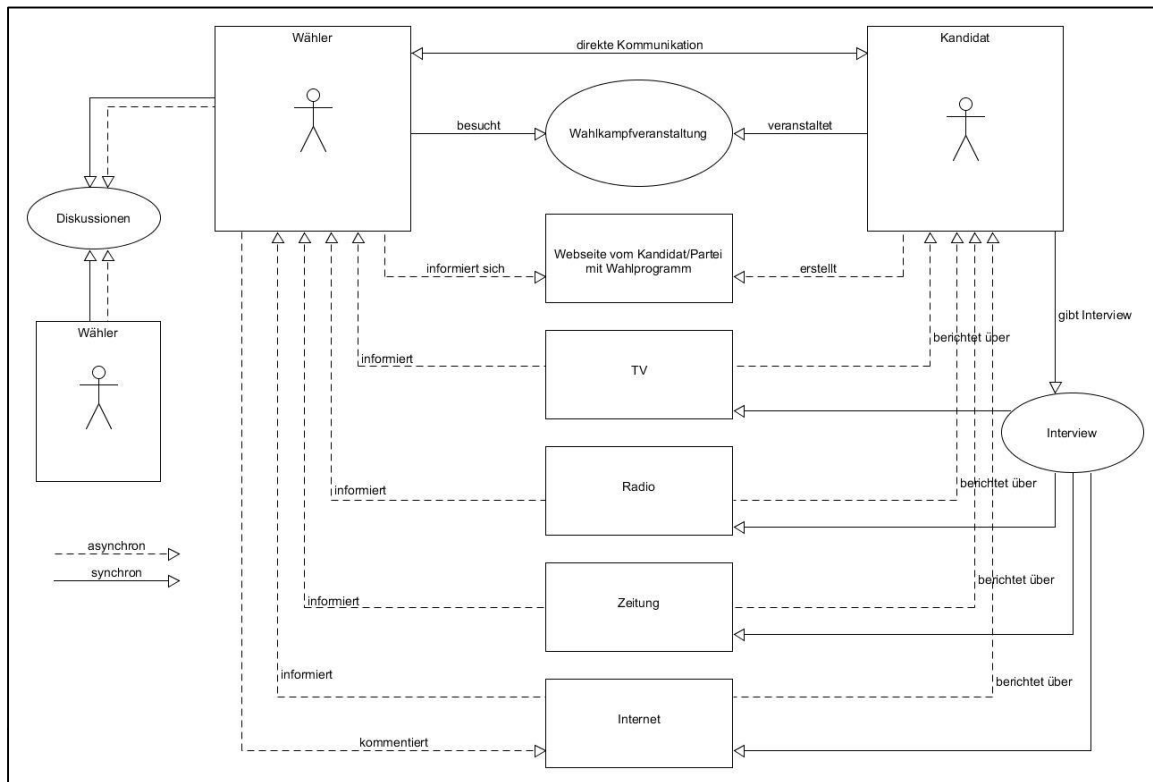


Abbildung 20: Deskriptives Kommunikationsmodell

Dieses Modell in Abbildung 20 veranschaulicht den Ist-Zustand der Kommunikation zwischen den Wählern und Kandidaten. Die Basis des Modells sind die Ergebnisse der Domänenrecherche und des Nutzungsproblems. Die Kommunikation läuft primär asynchron über verschiedene Medien wie TV, Radio, Zeitung und Internet ab, wobei im Internet eine synchrone Kommunikation durch das kommentieren der Beiträge von den Wählern aus vorliegt. Es ist selbstverständlich, dass sich Wähler mit anderen Wählern direkt austauschen können. Die Wähler können auch die direkte Kommunikation zum Kandidaten suchen, falls diese sich die Zeit dafür nehmen. Es besteht die Möglichkeit bei einer Wahlkampfveranstaltung der Kandidaten einer direkten Kommunikation mit den Kandidaten, falls die Wähler diese besuchen. Andernfalls gibt der Kandidat Interviews, die an die Medien weitergeleitet werden, wobei diese Information erst in einem späteren Zeitpunkt vorliegt und von den Wählern abgerufen werden kann.

Präskriptives Kommunikationsmodell

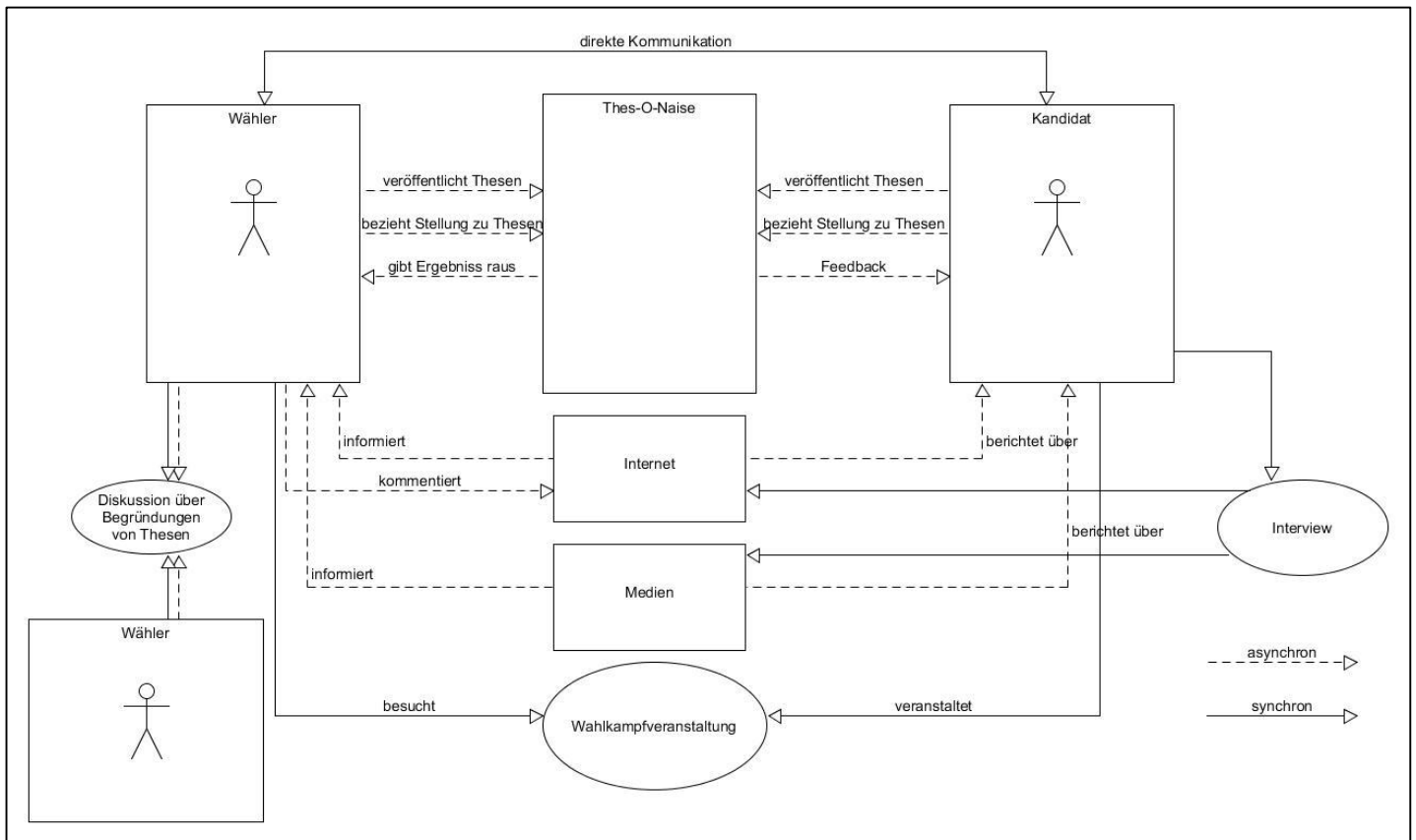


Abbildung 21: Präskriptives Kommunikationsmodell

In diesem Modell, zu sehen in Abbildung 21, soll verdeutlicht werden, wie die Kommunikation zwischen Wählern und Kandidaten optimiert werden kann. Die Wähler haben immer noch die Möglichkeit, Informationen über Medien abzurufen und sich mit anderen Wählern direkt auszutauschen. Zusätzlich können Wähler und Kandidaten über das System anhand von Thesen Informationen austauschen. Dies ermöglicht zum einen, dass die Wähler ihre Meinung, Wünsche und Forderungen verdeutlichen und zum anderen haben die Kandidaten eine bessere Übersicht auf die Meinung der Masse und verschaffen den Wählern durch ihre Stellungnahmen zu den Thesen ein klareres Bild über sich selbst.

Architektur und Architektordiagramm

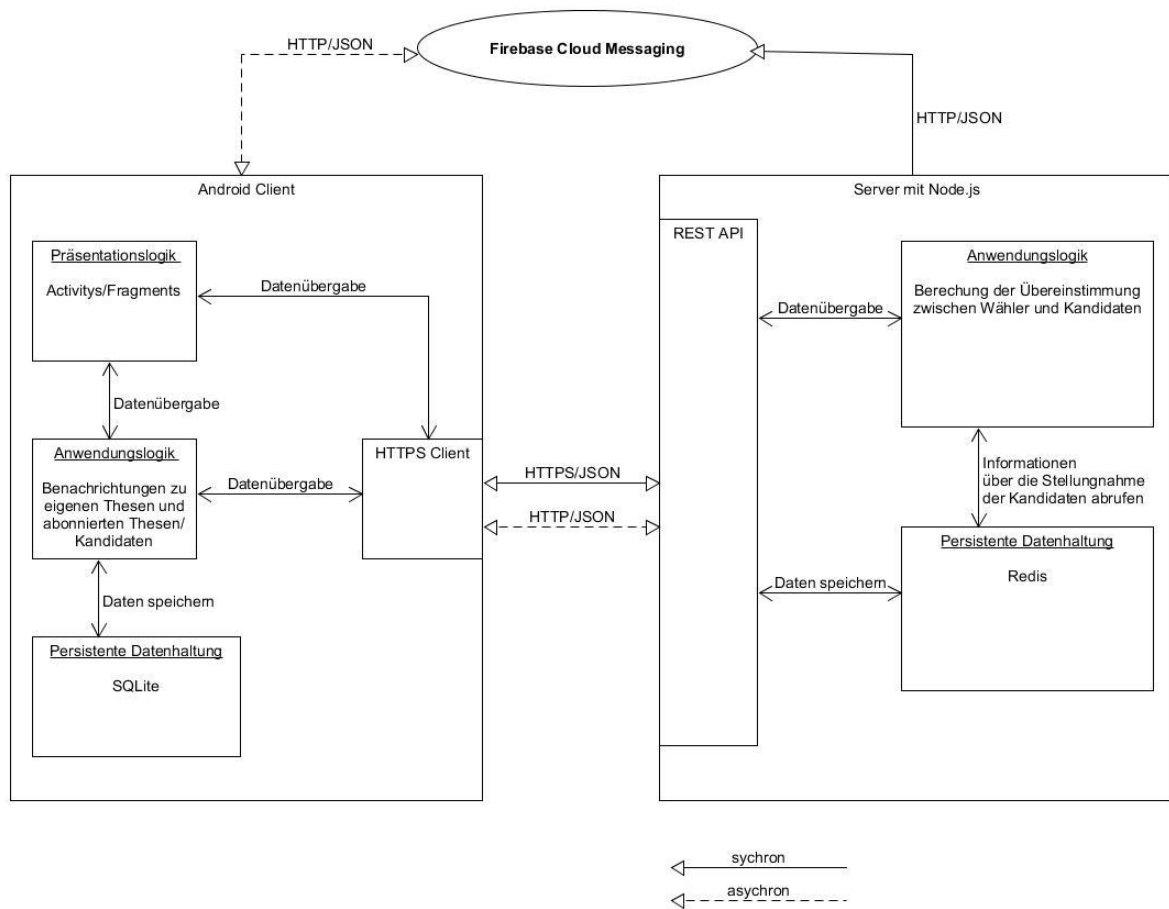


Abbildung 22: Architektordiagramm

Der Server soll als eine REST API mit Node.js und JavaScript entwickelt werden. Die synchrone und asynchrone Client-Server Kommunikation wird über das HTTPS Protokoll mit dem Datenformat JSON stattfinden. Der Client des Prototypen wird als Android App entwickelt.

Die Vorteile von einer Entwicklung mit Node.js sind die einfache Implementation von den benötigten Ressourcen (siehe Tabelle 14 im Anhang Seite 51) mit den generischen HTTP-Verben, sowie die Skalierbarkeit des Servers. Da eine Ressource mehrere Repräsentationen haben könnte, jedoch immer nach einer festgelegten Semantik arbeiten sollte, ergibt sich die Entwicklung des Servers als REST API.

Aufgrund einer zeitlichen Beschränkung des Projektes fokussiert sich die Entwicklung des Projektes nur auf den mobilen Nutzungskontext. Ein Vorteil einer mobilen Android Anwendung ist die Möglichkeit jederzeit unterwegs Thesen zu stellen und Stellung zu beziehen. Bei einer einfachen Webanwendung hingegen könnten die privaten Daten der Wähler nicht ausreichend geschützt werden.

Asynchrone Kommunikation

Die Asynchrone Kommunikation zwischen Server und Client soll über *Googles Firebase Cloud Messaging* (Google Inc., 2016, folgend auch *FCM*) erfolgen. Alternative Möglichkeiten wären *PARSE* oder *Amazon SNS*, jedoch lässt sich *FCM* mit geringsten Aufwand in der Android App implementieren. Um dem strategischen Ziel „Die privaten Daten des Wählers müssen geschützt werden“ zu folgen, sollen die Benutzer vor der Verwendung von *FCM* selbst entscheiden, ob Sie dieser Verwendung zustimmen. Wenn der Benutzer dem zugestimmt hat, kann der Server den Android Client des Benutzers über neue Informationen benachrichtigen, sodass automatisch die aktuellsten Informationen von Thesen oder Kandidaten im Client dargestellt werden. Andernfalls muss der Benutzer manuell neue Informationen über Thesen oder Kandidaten vom Server abfragen.

Der Payload der Benachrichtigung vom Server soll nur die Information enthalten, zu welcher These oder zu welchem Kandidat neue Informationen vorliegen, sodass keine sensiblen Daten übertragen werden.

Anwendungslogik

Die *Anwendungslogik des Servers* ist die Berechnung der Übereinstimmung zwischen Wähler und Kandidat. Für jede Abweichung zwischen dem Kandidaten und der Position des Wählers erhält der jeweilige Kandidat einen Punkt. Der Kandidat mit den wenigsten Punkten ist der Position des Wählers in Bezug auf die abgefragten Thesen am nächsten, der mit den meisten am fernsten. Für die Differenz zwischen "pro" und "contra" erhält der Kandidat zwei Punkte. Kommt die Bewertung "neutral" auf Seiten des Wählers oder des Kandidaten vor, wird nur ein Punkt vergeben - es sei denn beide Seiten stimmen "neutral", dann erhält der Kandidat keinen Punkt.

Tabelle 7: Punktesystem für die Berechnung

| | Wähler: „pro“ | Wähler: „neutral“ | Wähler: „contra“ |
|---------------------|---------------|-------------------|------------------|
| Kandidat: „pro“ | 0 | 1 | 2 |
| Kandidat: „neutral“ | 1 | 0 | 1 |
| Kandidat: „contra“ | 2 | 1 | 0 |

Die öffentliche Position der Kandidaten zu den einzelnen Thesen muss dabei persistent auf dem Server gespeichert werden, während die persönliche Ansicht des Wählers nur kurzfristig für diese Berechnung zum Server übermittelt wird. Um den Datenschutz der privaten Daten des Wählers zu gewährleisten, werden die Daten verschlüsselt auf dem Client persistent abgespeichert und über eine sichere HTTPS Verbindung zum Server übertragen. Dabei müssen die persönliche Position des Wählers zu einer einzelnen These, die Bewertung einer These, die Bewertung einer Begründung einer Position zu einer These und die Ergebnisse der Berechnung der Übereinstimmung beim Client gespeichert werden.

Die *Anwendungslogik des Clients* sind die Benachrichtigungen zu eigenen Thesen oder abonnierten Thesen/Kandidaten. Sobald sich Kandidaten zu den eigenen oder abonnierten Thesen positioniert haben oder Begründungen ihrer Position hinzugefügt haben, bekommt der Wähler eine Benachrichtigung in der Ansicht seines Profils.

Der Algorithmus der Berechnung der Übereinstimmung

Wenn der Wähler eine Berechnung durchführen lässt, wird ein JSON Array erzeugt, welches für jede These ein JSON Objekt enthält. Dieses Objekt enthält die Thesen ID, die Position des Wählers zur These, ein Feld zur Bestimmung ob diese These schon ausgewertet wurde und gegeben falls die

Position des Wählers bei der letzten Auswertung. Beim Server wird dieses JSON Array nun auf vier Arrays (Thesen_IDS[], USERPOS[], GESENDET[], LASTUSERPOS[]) aufgeteilt. Anschließend werden die Thesen aus der Datenbanken geladen und es wird ein Array für die Kandidaten ID's und mehrere Array für die Zähler der Punktzahlen erzeugt. Bei jeder These wird mittels einer for-Schleife über das „K_Positionen“ Array iteriert und jede neue Kandidaten ID wird dem KandidatenID Array hinzugefügt. Nun wird jede einzelne Kandidaten Position mit der Position des Wählers zu der jeweiligen These verglichen und die Punktzahl des Kandidaten an die Stelle im Punktzahlen Array geschrieben, an der sich die jeweilige Kandidaten ID im KandidatenID Array befindet. Nach dem alle Positionen des Wählers zu den Thesen verarbeitet wurden, wird ein JSON Array mit dem Ergebnissen erzeugt. Für jeden Kandidaten wird ein JSON Objekt mit seiner KID und den Punktzahlen auf dieses Ergebnis Array gepusht und anschließend wird dieses Ergebnis an den Client zurückgeschickt. Der Client berechnet anschließend dann die durchschnittlichen Punkte des Kandidaten pro These damit die Kandidaten, welche sich zu weniger Thesen positioniert haben, keinen Vorteil gegenüber anderen Kandidaten bekommen. Falls ein Kandidat sich in einer Kategorie zu keinen Thesen positioniert hat, wird er in der Ergebnisliste nicht aufgeführt.

Datenstrukturen

Im Folgenden werden die Datenstrukturen und die persistente Speicherung der Daten bei dem Server und bei den Android Clients beschrieben.

Datenstruktur bei dem Server

Für die persistente Datenhaltung wurde *Redis* beim Server gewählt, da schon auf herkömmlicher Hardware mehrere Zehntausend Schreibvorgänge pro Sekunde möglich sind und dies die Skalierbarkeit des Servers garantiert. Die verschiedenen Repräsentationen von Wähler, Kandidaten und Thesen werden mit einer einmaligen ID (z.B. „WID_2“, „KID_3“, „TID_4“) als „Key“ in *Redis* gespeichert.

```
var Waehler = {  
    username: "MaxMustermann",  
    password: "12345",  
    wahlkreis: "Oberbergischer Kreis",  
    email: "max@mustermann.de",  
    WID: "WID_23"  
};
```

Abbildung 23: Datenstruktur eines Wählers – serverseitig.

Zu jedem registrierten Wähler wird sein Username, sein Passwort, sein Wahlkreis und seine Email-Adresse, sowie seine eindeutige WID als String gespeichert. (siehe Abbildung 23)

```

var Kandidat = {
  username: "Kandidat_der_Herzen",
  vorname: "Peter",
  nachname: "Müller",
  password: "12345",
  wahlkreis: "Oberbergischer Kreis",
  email: "peter.müller@email.de",
  webseite: "peter.müller-kandidat.de"
  KID: "KID_34",
  Partei: "Unabhängig",
  Thesen_positioniert: [
    {
      TID: "TID_2",
      POS: "PRO",
      KATEGORIE: "Umwelt"
    },
    {
      TID: "TID_543",
      POS: "CONTRA",
      KATEGORIE: "Lokal"
    }
  ],
  Begründungen: [
    {
      TID: "TID_3",
      POS: "PRO",
      TEXT: "Diese These ist toll."
    }
  ],
  Biographie:
    {
      Geburtsdatum: "23.1.1975",
      Bildungsweg: "Abitur, Studium",
      Berufe: "Lehrer",
      Mitgliedschaften: "",
    }
  ,
  Wahlprogramm: {
    Text: "Deswegen sollte ich gewählt werden.",
    Link: "link.zum.wahlprogramm.de"
  }
};

```

Abbildung 24: Datenstruktur eines Kandidaten – serverseitig.

Zu jedem registrierten Kandidaten wird ebenfalls sein Username, sein Passwort, sein Wahlkreis und seine Email-Adresse und dazu noch seinen Vor- und Nachnamen und seine Parteizugehörigkeit als String gespeichert. Desweiteren wird ein JSON Array „Thesen_positioniert“ für jeden Kandidaten angelegt, in dem seine Position in einem JSON Objekt zu den jeweiligen Thesen festgehalten wird. Dieses JSON Objekt umfasst die TID der These, die konkrete Position des Kandidaten und die Kategorie der These. Außerdem werden die Begründungen des Kandidaten zu Thesen in einem Array namens Begründungen[] gespeichert. Hinzu kommen noch ein JSON Objekt mit der Biographie des Kandidaten und ein JSON Objekt mit dem Wahlprogramm des Kandidaten. (siehe Abbildung 24)

```

var These = {
  thesentext: "Es sollte für Bildung mehr Geld ausgegeben werden als für Krieg!",
  kategorie: "Wirtschaft",
  wahlkreis: "Oberbergischer Kreis",
  Anzahl_Zustimmung: 23454,
  Anzahl_Ablehnung: 34,
  Anzahl_Neutral: 347,
  Likes: 3785,
  TID: "TID_2",
  K_PRO: [
    {
      UID: "KID_2",
      Text: "Geld ist genug da!",
      likes: 5,
      Kommentare: [
        {
          UID: "WID_2",
          USERNAME: "Wähler123"
          Kommentar: "Endlich sagt es jemand"
        }
      ]
    },
    {
      UID: "KID_5",
      Text: "Bildung ist wichtig!",
      likes: 6,
      Kommentare: []
    }
  ],
  K_NEUTRAL: [],
  K_CONTRA: [],
  W_PRO: [],
  W_NEUTRAL: [],
  W_CONTRA: [],
  K_POSITION: [
    {
      UID: "KID_2",
      POS: "PRO"
    },
    {
      UID: "KID_5",
      POS: "PRO"
    }
  ],
];

```

Abbildung 25: Datenstruktur einer These – serverseitig.

Zu jeder veröffentlichten These wird der Text der These, die Kategorie der These, der Wahlkreis des Absenders und die Thesen ID (TID) gespeichert. Wenn die Wähler eine Berechnung der Übereinstimmung von Server durchführen lassen, wird ihre Position zu den Zählern Anzahl_Zustimmung, Anzahl_Ablehnung oder Anzahl_Neutral hinzugezählt. Die Begründungen der Positionen von Kandidaten und Wähler zur Thesen werden in verschiedenen JSON Arrays (von K_PRO bis W_CONTRA) gespeichert. Dabei repräsentiert ein JSON Objekt die Begründung und umfasst die UID des Absenders, den Text der Begründung, die Anzahl an „likes“ der Begründung, und ein JSON Array mit Kommentaren zu der Begründung. Da die Begründungen optional von Kandidaten zur These hinzugefügt werden können, werden die Positionen der Kandidaten zu der These in einem zusätzlichen JSON Array namens „K_POSITION“ gespeichert. (siehe Abbildung 25)

Datenstruktur bei dem Android Client

Um die Android App auch ohne ständige Internetverbindung nutzen zu können, müssen die Daten auch beim Client persistent gespeichert werden. Dafür wurde beim Client das Open-Source „SQLite“ Datenbanksystem verwendet, welches dem ACID-Prinzip folgt und sich mit relativ geringem Aufwand in die App implementieren lässt.

```
public static final String SQL_CREATE_THESENTABLE =
    "CREATE TABLE " + ThesenTable.TABLE_NAME + " (" +
        ThesenTable.COLUMN_NAME_TID + " STRING PRIMARY KEY," +
        ThesenTable.COLUMN_NAME_THESENTEXT + " TEXT," +
        ThesenTable.COLUMN_NAME_KATEGORIE + " TEXT," +
        ThesenTable.COLUMN_NAME_WAHLKREIS + " TEXT," +
        ThesenTable.COLUMN_NAME_LIKES + " TEXT," +
        ThesenTable.COLUMN_NAME_ANZAHL_PRO + " TEXT," +
        ThesenTable.COLUMN_NAME_ANZAHL_NEUTRAL + " TEXT," +
        ThesenTable.COLUMN_NAME_ANZAHL_CONTRA + " TEXT," +
        ThesenTable.COLUMN_NAME_K_PRO + " TEXT," +
        ThesenTable.COLUMN_NAME_K_NEUTRAL + " TEXT," +
        ThesenTable.COLUMN_NAME_K_CONTRA + " TEXT," +
        ThesenTable.COLUMN_NAME_W_PRO + " TEXT," +
        ThesenTable.COLUMN_NAME_W_NEUTRAL + " TEXT," +
        ThesenTable.COLUMN_NAME_W_CONTRA + " TEXT," +
        ThesenTable.COLUMN_NAME_K_POSITION + " TEXT" +
    ");";
```

Abbildung 26: Datenstruktur einer These – clientseitig.

Der Aufbau der Thesen Tabelle beim Client ist deckungsgleich zur Repräsentation beim Server. (siehe Abbildung 16)

```
public static final String SQL_CREATE_KANDIDATENTABLE =
    "CREATE TABLE " + KandidatenTable.TABLE_NAME + " (" +
        KandidatenTable.COLUMN_NAME_KID + " STRING PRIMARY KEY," +
        KandidatenTable.COLUMN_NAME_VORNAME + " TEXT," +
        KandidatenTable.COLUMN_NAME_NACHNAME + " TEXT," +
        KandidatenTable.COLUMN_NAME_PARTEI + " TEXT," +
        KandidatenTable.COLUMN_NAME_EMAIL + " TEXT," +
        KandidatenTable.COLUMN_NAME_WAHLKREIS + " TEXT," +
        KandidatenTable.COLUMN_NAME_BEANTWORTETETHESEN + " TEXT," +
        KandidatenTable.COLUMN_NAME_PUNKTE_INSGESAMT + " INTEGER," +
        KandidatenTable.COLUMN_NAME_PUNKTE_LOKAL + " INTEGER," +
        KandidatenTable.COLUMN_NAME_PUNKTE_UMWELT + " INTEGER," +
        KandidatenTable.COLUMN_NAME_PUNKTE_AP + " INTEGER," +
        KandidatenTable.COLUMN_NAME_PUNKTE_SATIRE + " INTEGER," +
        KandidatenTable.COLUMN_NAME_BIOGRAPHIE + " TEXT," +
        KandidatenTable.COLUMN_NAME_WAHLPROGRAMM + " TEXT" +
    ");";
```

Abbildung 27: Datenstruktur eines Kandidaten – clientseitig.

Die Kandidaten Tabelle beinhaltet wie bei dem Server verschiedene Informationen zum Kandidaten und wurde um die Spalten „Punkte_Insgesamt“, „Punkte_Lokal“, „Punkte_Umwelt“, „Punkte_AP“ und „Punkte_Satire“ erweitert, um nach dem Matching die Punktzahlen der Auswertung zu speichern und die Kandidaten nach diesen Punktzahlen sortiert darstellen zu können. (siehe Abbildung 27)

```

public static final String SQL_CREATE_USERPOSITIONDATATABLE =
    "CREATE TABLE " + UserpositiondataTable.TABLE_NAME + " (" +
        UserpositiondataTable.COLUMN_NAME_TID + " STRING PRIMARY KEY," +
        UserpositiondataTable.COLUMN_NAME_POSITION + " TEXT," +
        UserpositiondataTable.COLUMN_NAME_SERVER + " TEXT," +
        UserpositiondataTable.COLUMN_NAME_LAST_POSITION + " TEXT" +
    " )";

```

Abbildung 28: Datenstruktur der Benutzerpositionen – clientseitig.

In der Userposition Tabelle wird die Position des Benutzers zu jeweiligen These gespeichert. In der Spalte „Server“ wird erfasst, ob die Position des Benutzers schon von dem Server ausgewertet wurde und in der Spalte „Last_Position“ wird die letzte Position des User nach der Auswertung gespeichert, umgegeben falls die Zähler (Zustimmung, Ablehnung, Neutral) der jeweiligen Thesen bei der nächsten Auswertung zu korrigieren. (siehe Abbildung 28)

Des Weiteren wird bei dem Android Client gespeichert, welche Thesen vom Benutzer veröffentlicht wurden und welche Thesen und Kandidaten der Benutzer abonniert hat.

Risiken

Im Folgenden werden die Risiken des Projektes identifiziert und der Umgang mit diesen beschrieben.

Fehlende Interaktionsmöglichkeit bei den Thesen.

Die Stakeholder haben keine Interaktionsmöglichkeit sich zu den Thesen zu positionieren. Ohne diese Möglichkeit kann später keine Berechnung der Übereinstimmung zwischen Wähler und Kandidaten erfolgen. Aus diesem Grund muss der Client solange weiterentwickelt werden bis diese Interaktion möglich ist.

Fehlerhafte Berechnung der Übereinstimmung zwischen Wähler und Kandidaten.

Der Algorithmus der Berechnung könnte fehlerhaft implementiert werden, sodass entweder nicht alle Positionen der Kandidaten erfasst werden oder die des Wählers.

Keine sichere HTTPS Verbindung zwischen Server und Client.

Damit die Daten zwischen Server und Client nicht manipuliert oder abgefangen werden können, muss die Verbindung zwischen beiden Komponenten mittels HTTPS erfolgen. Solange dies noch nicht implementiert wurde, wird eine einfache HTTP Verbindung verwendet.

Keine Verschlüsselung der privaten Daten des Wählers beim Client.

Die Verschlüsselung kann aufgrund des Zeitmangels nicht rechtzeitig implementiert werden. Falls dies geschieht werden die privaten Daten des Wählers noch nicht verschlüsselt gespeichert.

Unbekannte geben sich als Kandidaten aus und bewerten Thesen.

Wenn Unbekannte sich als Kandidaten ausgeben und dann auf dem System als diese Kandidaten zu Thesen Stellung beziehen, kann dies die Seriosität des Systems gefährden. Die Wähler müssen sich sicher sein, dass auch wirklich der echte Kandidat zu den Thesen sich positioniert, da ansonsten falsche Informationen verbreitet werden. Um diesem Risiko vorzubeugen, müssen sich Kandidaten mit einem Personalausweis verifizieren bevor sie zu Thesen Stellung nehmen können.

Durch mehrmaliges Abschicken von privaten Daten eines Wählers erhöht sich der öffentliche Zähler für die Positionen der Wähler zu der jeweiligen These mehr als einmal.

Damit ein Wähler nicht durch mehrmaliges Abschicken die öffentlichen Zähler der Positionen der Wähler zu einer Thesen manipulieren kann, muss der Client speichern, ob der Wähler sich schon zu der These positioniert hat und welche Stellung (pro/neutral/contra) er dabei eingenommen hatte. Diese Informationen werden dann mit zu dem Server übertragen, sodass der Server den öffentlichen Zähler nicht mehr als einmal pro Wähler erhöht.

Mehrere Thesen ähneln sich inhaltlich zu sehr.

Wenn viele Wähler viele Thesen veröffentlichen, könnten mehrere Thesen sich inhaltlich zu sehr ähneln. Dies birgt die Gefahr, dass Wähler und Kandidaten die Nutzungsmotivation verlieren könnten. Außerdem könnte die Berechnung der Übereinstimmung verfälscht werden, wenn beispielsweise ein Wähler und ein Kandidat sich zu mehreren inhaltlichen sehr ähnlichen Thesen positionieren und dadurch dieser Kandidat bei der Berechnung besser dasteht. Um dies zu verhindern, könnte zum einen das System den Hashwert des Textes mit den Hashwerten schon vorhandener Thesen abgleichen um zumindest exakt gleiche Thesen nicht doppelt zu speichern. Zum anderen könnte ein Moderator inhaltlich doppelte Thesen aus dem System entfernen. Außerdem könnten die Nutzer aufgefordert werden nach Thesen zu suchen, bevor sie ähnliche Thesen veröffentlichen wollen.

Nicht alle Kandidaten eines Wahlkreises beziehen zu den Thesen Stellung.

Die Berechnung der Übereinstimmung kann nur die Kandidaten berücksichtigen, die schon zu Thesen Stellung bezogen haben. Es könnte dadurch vorkommen, dass ein Kandidat besser zu einem Wähler passt, aber noch zu keinen Thesen Stellung bezogen hat. Deswegen müssen die Wähler bei den Ergebnissen auf diesen Umstand hingewiesen werden. Außerdem könnte das System Kandidaten benachrichtigen, welche noch keine oder kaum Thesen beantwortet haben, und diese dazu auffordern zu mehr Thesen Stellung zu beziehen. Des Weiteren könnten die Wähler Kandidaten zur Teilnahme beim System einladen.

Proof of Concept

In diesem Kapitel werden die Proof of Concepts, welche einige Risiken adressieren, mit Exit und Fail Kriterium tabellarisch spezifiziert und gegeben falls wird eine Alternative beschrieben.

Tabelle 8: Interaktion mit Thesen beim Client

| Kategorie | Definition |
|--------------|---|
| Beschreibung | Die Darstellung der Thesen muss den Stakeholder die Interaktionsmöglichkeit geben eine Stellung (pro/neutral/contra) zu wählen. |
| Exit | Die gewählte Stellung des Stakeholders zu einer einzelnen These wird korrekt erfasst und gespeichert. |
| Fail | Die gewählte Stellung des Stakeholders zu einer einzelnen These wird nicht korrekt erfasst oder nicht gespeichert. |
| Fallback | Diese Grundlage des Systems ist unverzichtbar und alternativlos. |

Tabelle 9: Berechnung der Übereinstimmung von Wähler und Kandidaten

| Kategorie | Definition |
|--------------|--|
| Beschreibung | Nachdem Kandidaten und der Wähler sich zu den Thesen positioniert haben, berechnet der Server die Übereinstimmung. |
| Exit | Die Berechnung erfolgt korrekt und erfasst alle Positionen aller Kandidaten des Wahlkreises. |
| Fail | Der Algorithmus der Berechnung ist fehlerhaft. |
| Fallback | Diese Grundlage des Systems ist unverzichtbar und alternativlos. |

Tabelle 10: Sichere HTTPS Verbindung zwischen Client und Server

| Kategorie | Definition |
|--------------|---|
| Beschreibung | Die Verbindung zwischen Client und Server erfolgt über HTTPS. |
| Exit | Die Verbindung zwischen Client und Server erfolgt über HTTPS. |
| Fail | Die Verbindung erfolgt nicht über HTTPS. |
| Fallback | Die Verbindung erfolgt über HTTP |

Tabelle 11: Verschlüsselung der privaten Daten des Wählers

| Kategorie | Definition |
|--------------|---|
| Beschreibung | Die privaten Daten werden beim Client verschlüsselt gespeichert |
| Exit | Die Daten sind sicher verschlüsselt. |
| Fail | Die Daten werden nicht sicher verschlüsselt. |
| Fallback | Andere Verschlüsselungsmethoden implementieren. |

Tabelle 12: Vermeidung von Manipulation bei öffentlichen Zählern

| Kategorie | Definition |
|--------------|--|
| Beschreibung | Ein Client darf nicht durch mehrmaliges Abschicken von Daten an den Server die öffentlichen Zähler der Positionen der Wähler zu einer These manipulieren. |
| Exit | Der Client speichert ob und wie der Wähler sich zu einer These positioniert hat. Diese Daten schickt der Client dann zum Server, sodass der Zähler nicht mehr als einmal pro Wähler und These erhöht wird. |
| Fail | Der Client speichert nicht oder fehlerhaft wie sich der Wähler zu einer These positioniert hat. |
| Fallback | Diese Grundlage des Systems ist unverzichtbar und alternativlos. |

Tabelle 13: Vermeidung von inhaltlich doppelten Thesen

| Kategorie | Definition |
|--------------|---|
| Beschreibung | Es sollten keine inhaltlich doppelten Thesen gespeichert werden, da dies ansonsten die Nutzungsmotivation gefährden könnte. |
| Exit | Der Server gleicht den Hashwert des Textes mit den Hashwerten schon vorhandener Thesen ab, sodass nicht exakt gleiche Thesen gespeichert werden. |
| Fail | Inhaltlich identische Thesen werden noch gespeichert. |
| Fallback | Moderatoren könnten sich inhaltlich ähnelnde Thesen löschen oder die Nutzer könnten aufgefordert werden erst in den vorhandenen Thesen nach ihrer These zu suchen bevor sie ihre These veröffentlichen. |

Fazit

An dieser Stelle sollen eine ergebnisorientierte Diskussion des Zielerreichungsgrades im Projektverlauf sowie ein Ausblick des Projektes folgen.

Die zentralen Anforderungen und Usability Goals für das System sind soweit durch die wichtigsten Funktionen und der Übersichtlichkeit des Designs erfüllt. Es wurde ein System entwickelt, welches den Wählern erleichtert Thesen an alle Kandidaten des eigenen Wahlkreises zu stellen.

Endsprechend der anfangs gesetzten Zielsetzung unterstützt das System die Meinungsbildung der Wähler zu den Kandidaten. Es wurde darauf geachtet dem Benutzer ein minimalistisches Design zu bieten, damit dieser sich problemlos durch die Funktionen des Systems navigieren kann.

Aufgrund des kurzen Projektzeitraums, konnte die Funktionale Anforderung das Benutzer privat untereinander diskutieren können nicht implementiert werden.

Das System bietet sich nicht nur als Wahlhilfe an, sondern berücksichtigt auch, die Bedürfnisse der Wähler untereinander zu diskutieren. Dennoch ermöglicht das System in Zukunft ausreichend Raum für neue Funktionen und Erweiterungen.

Literaturverzeichnis

Landeszentrale für politische Bildung Baden-Württemberg (2016). Wahlprogramme 2013. [Online]

Verfügbar bei : <http://www.bundestagswahl-bw.de/wahlprogramme1.html>

[Zugriff am 20.10.2016]

SPD (2016). Bürgerdialog. [Online]

Verfügbar bei : <https://www.spd.de/buergerdialog/>

[Zugriff am 20.10.2016]

Plickert, Philip (2013). Das schwierige Geschäft mit den Umfragen. [Online]

Verfügbar bei: <http://www.faz.net/aktuell/wirtschaft/meinungsforschung-das-schwierige-geschaeft-mit-den-umfragen-12541293.html>

[Zugriff am 30.10.2016]

Leimeister, J. M.; Zogaj, S.; Durward, D. & Blohm, I. (2015):

Crowdsourcing und Crowd Work - Neue Formen digitaler Arbeit. In: Mensch 2020 -

transdisziplinäre Perspektiven. Hrsg.: Bullinger, A. Verlag: Verlag Wissenschaft und Praxis, Chemnitz. Erscheinungsjahr: 2015.

Seiten/Pages: 119 - 124.

Marschall, Stefan (2005):

Idee und Wirkung des Wahl-O-Mat. [Online]

Verfügbar bei: <http://www.bpb.de/politik/wahlen/wahl-o-mat/45379/idee-und-wirkung?p=all>

[Zugriff am 20.10.2016]

Marschall, Stefan (2011):

Wahlen, Wähler, Wahl-O-Mat, in: Aus Politik und Zeitgeschichte 61(4), S. 40-46.

Verfügbar bei: <http://www.bpb.de/files/8R9IXV.pdf>

[Zugriff am 22.10.2016]

Mayhew, Deborah J. (1999):

The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design. s.l.:Academic Press.

bpb (2016):

Wahl-O-Mat Bundestagswahl 2013. [Online]

Verfügbar bei: <http://www.wahl-o-mat.de/bundestagswahl2013>

[Zugriff am 10.20.2016]

Anhang

Marktrecherche :

1. Wahl-O-Mat der Bundeszentrale für politische Bildung

Der Wahl-O-Mat wurde entwickelt um den Benutzern die Möglichkeit zu bieten, die eigene Einstellung mit den Positionen der Parteien zu vergleichen. Dabei sollen besonders Erst- und Zweitwähler/-inne/n angesprochen werden, da deren Wahlbeteiligung abnimmt (vgl. Marschall, Stefan (2005)). Jeder der 38 vorgefertigten Thesen kann der Benutzer mit "stimme zu", "neutral" oder "stimme nicht zu" beantworten, wie es bereits zuvor die Parteien getan haben. Außerdem kann der Benutzer eine These überspringen, wie es in Abbildung 29 auf der nächsten Seite zu sehen ist. Hinzukommt, dass der Nutzer die Möglichkeit hat, ihm persönlich wichtige Thesen zu markieren, die bei der Berechnung besonders gewichtet werden sollen (siehe Abbildung 30 & 31 nächste Seite). Anschließend können bis zu acht Parteien für die Auswertung vom Benutzer ausgewählt werden (siehe Abbildung 32 nächste Seite) und der Wahl-O-Mat ermittelt die Übereinstimmung zwischen den Stellungnahmen der Parteien und den Positionen des Nutzers und stellt die Parteien in abnehmender Reihenfolge im Grad der Übereinstimmung dar (siehe Abbildung 33 nächste Seite). Hinzukommt die Möglichkeit in einer Detailauswertung die eigenen Ansichten zu den Thesen mit den Standpunkten jeder einzelnen Partei zu vergleichen und dabei die Begründungen der jeweiligen Partei zu sehen (siehe Abbildung 34).

Vorteile :

- Schnelle und einfache Möglichkeit seine Ansichten mit den Positionen der Parteien zu vergleichen

Nachteile:

- Die 38 vorgefertigten Thesen decken nur einen kleinen Teil der möglichen politischen Positionen ab
- Nur Parteien werden vom Wahl-o-Mat zum Vergleich zugelassen, unabhängige Direktkandidaten können nicht teilnehmen
- Bestimmte Themenbereiche können nicht auf eine einfache Thesen reduziert werden
- Nutzer haben keinen Einfluss auf die Thesen, bzw. können den Parteien oder Kandidaten keine eigenen Thesen zur Beantwortung stellen
- Steht den Nutzer erst 2-4 Wochen vor der Wahl zur Verfügung
- Durch die Limitierung der Auswahl von nur 8 Parteien zur Auswertung, wird dem Nutzer der Aufwand erhöht seine Ansichten mit allen Parteien zu vergleichen



Abbildung 29: Thesendarstellung (bpb, 2016)



Abbildung 30: Gewichtung der Thesen (bpb, 2016)



Abbildung 31: Stern als Kennzeichen (bpb, 2016)

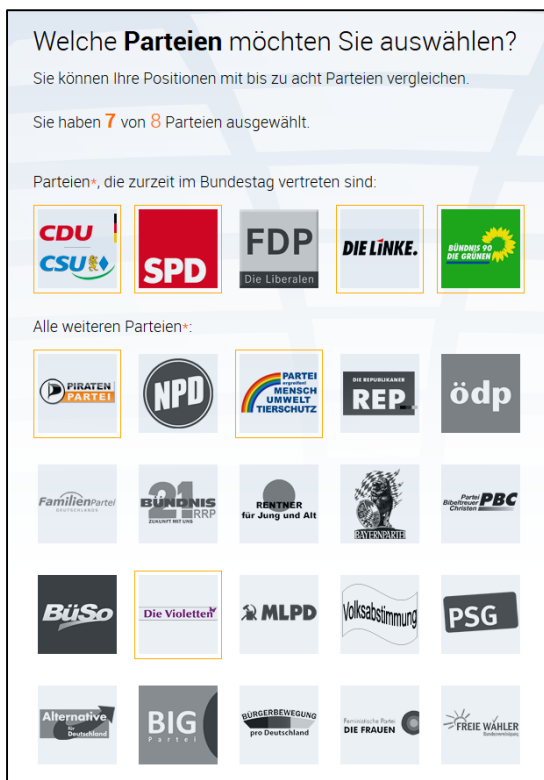


Abbildung 32: Parteiauswahl (bpb, 2016)

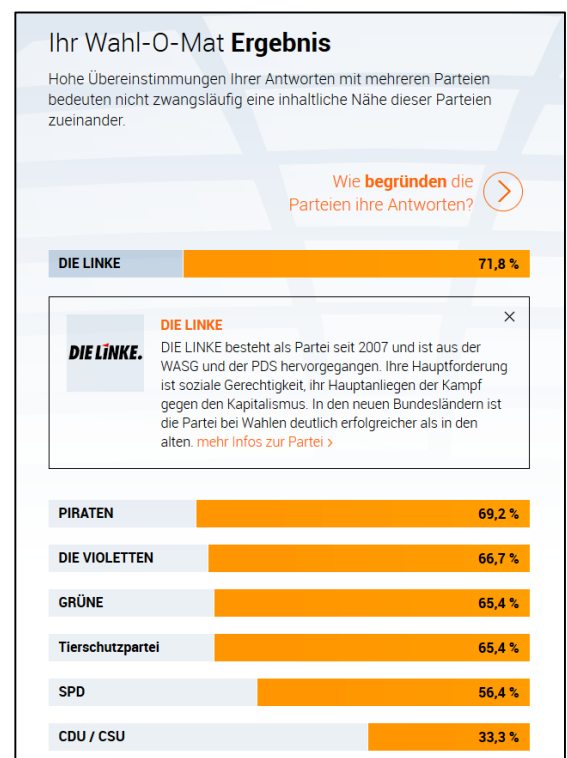


Abbildung 33: Ergebnisdarstellung (bpb, 2016)

Begründungen der Parteien

Vergleichen Sie Ihre Antworten mit denen der Parteien. Viele Parteien haben ihre Antworten zu den Thesen begründet. Sie können diese ein- oder ausblenden.

1/38 Mindestlohn ⓘ

Es soll ein gesetzlicher flächendeckender Mindestlohn eingeführt werden.

nächste These >

.....★.....

Ihre Position

✓

DIE LINKE

✓

PIRATEN

✓

Begründung der Partei:

„Bis zur Einführung eines Bedingungslosen Grundeinkommens setzen wir uns für einen bundesweiten gesetzlichen Mindestlohn ein. Bis zur Festlegung der Höhe durch eine Expertenkommission fordert die Piratenpartei als kurzfristige Maßnahme die Einführung eines Mindestlohns, der sich an 60% des durchschnittlichen Jahresarbeitslohn in Deutschland im Vorjahr orientiert (z.B. 9,02 Euro für das Jahr 2013).“

DIE VIOLETTEN

✓

Begründung der Partei:

„Als Vorstufe zum Grundeinkommen.“

Abbildung 34: Begründungen der Parteien (bpb, 2016)

2. www.erststimme2013.de

Die Webseite www.erststimme2013.de war eine Online-Wahlhilfe von Studierenden der Universität Bremen zur Bundestagswahl 2013 und sollte ein Informationsangebot über die Direktkandidatinnen und Direktkandidaten der Wahlkreise Bremen I und Bremen II und Osterholz-Verden zur Verfügung

stellen. Wie beim oben beschriebenen Wahl-O-Mat gab es 20 Thesen zu denen sich die Nutzer positionieren konnten. Außerdem gab es zu den einzelnen Kandidaten eine Profilseite mit Angaben zur Person, sowie deren Antworten und optionale Kommentare zu den einzelnen Thesen. Laut eigenen Angaben wurde die Wahlhilfe von 5773 Nutzern bis zur Bundestagswahl 2013 benutzt. Nach der Wahl ging die Wahlhilfe offline und es wurde die Auswertung der Umfrage und Nutzeranalyse veröffentlicht.

Vorteile :

- Schnelle und einfache Möglichkeit seine Ansichten mit den Positionen der Direktkandidaten der Bremer Wahlkreise zu vergleichen
- Profile der einzelnen Direktkandidaten der Wahlkreise

Nachteile:

- Die 20 vorgefertigten Thesen decken nur einen kleinen Teil der möglichen politischen Positionen ab
- Bezog sich nur auf die drei Bremer Wahlkreise
- Bestimmte Themenbereiche können nicht auf eine einfache Thesen reduziert werden
- Nutzer haben keinen Einfluss auf die Thesen, bzw. konnten Kandidaten keine eigenen Thesen zur Beantwortung stellen
- Stand den Nutzern nur bis zur Wahl zu Verfügung

3. www.parteivergleich.eu

Die Webseite www.parteivergleich.eu ist eine Online-Wahlhilfe, welche nach eigenen Angaben „kostenlos, unabhängig und neutral“ sein soll. Bei der Wahlhilfe können die Nutzer Fragen zu verschiedenen Wahlen beantworten und je nach Wahl gibt es zwischen 70 und 100 Fragen um die Ansichten des Nutzers mit den Positionen der Parteien zu vergleichen. Im Gegensatz zum Wahl-O-Mat wurden die Fragen nicht von einer Redaktion bestimmt, sondern „um größtmögliche Neutralität beim Parteienvergleich zu erreichen, wurde allen Parteien die Möglichkeit gegeben, die für sie wichtigsten Forderungen einzuschicken“. Zu jeder Frage bietet die Wahlhilfe dem Nutzer die Möglichkeit mit einem Klick eigene Differenzierungen vorzunehmen. So kann der Nutzer die persönliche Wichtigkeit der Frage von „extrem wichtig“, über „wichtig“ bis zu „nicht so wichtig“ bestimmen und gleichzeitig die Frage mit „Ja“, „Neutral“ oder „Nein“ beantworten oder ein Desinteresse mit der Antwortmöglichkeit „egal“ ausdrücken (siehe Abbildung 35 nächste Seite).

Nach der Beantwortung durch den Nutzer berechnet die Wahlhilfe die Nähe zu den jeweiligen Parteien, welche dann mittels Balkendiagrammen dargestellt werden (siehe Abbildung 36 übernächste Seite). Außerdem kann der Nutzer sich die Positionen der Parteien noch im Detail anzeigen lassen, wobei eine Tabelle mit den eigenen Antworten und den Antworten der Parteien dem Nutzer dargestellt wird (siehe Abbildung 37 nächste Seite).

Vorteile:

- Durch 70-100 Thesen lässt sich eine genauere Übereinstimmung errechnen
- Einfache Differenzierungen zu den Thesen durch den Benutzer
- Direkter Vergleich mit allen Parteien
- Nach der Wahl nicht offline

Nachteile:

- Unabhängige Direktkandidaten können nicht teilnehmen
- Thesen können nicht von den Wählern formuliert werden
- Es müssen alle Thesen beantwortet werden um eine Auswertung durchführen zu lassen
- Die Darstellung der Ergebnisse ist verbesserungswürdig
- Steht den Wähler nur wenige Monate vor der Wahl zur Verfügung

parteivergleich.eu

[Start](#)[Anleitung](#)[Europawahl 2014](#)[PDG 2014](#)[Nationalratswahl 2013](#)[FAQ](#)[Weblinks](#)[Impressum](#)

[Bundestagswahl 2013](#)[Bundestagswahl 2009](#)[NRW 2010](#)[Bremen 2011](#)[Tabellen \(alle Parteiantworten\)](#)

[Permalink](#)[Sitemap - Übersicht](#)

Bundestagswahl 2013

30 Parteien zur Wahl des Deutschen Bundestages im Vergleich

Thema: Demokratie - Staatsform

Soll über alle Gesetze und Beschlüsse auf Bundesebene per Volksentscheid abgestimmt werden dürfen?

| Ja | Neutral | Nein | Hilfe |
|-----------------------|-----------------------|-----------------------|------------------|
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | extrem wichtig |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | wichtig |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | nicht so wichtig |
| <input type="radio"/> | | | egal |

Bereits beantwortete Fragen (in Prozent):

0%

zurück

Abbildung 35: Darstellung einer These bei parteivergleich.eu

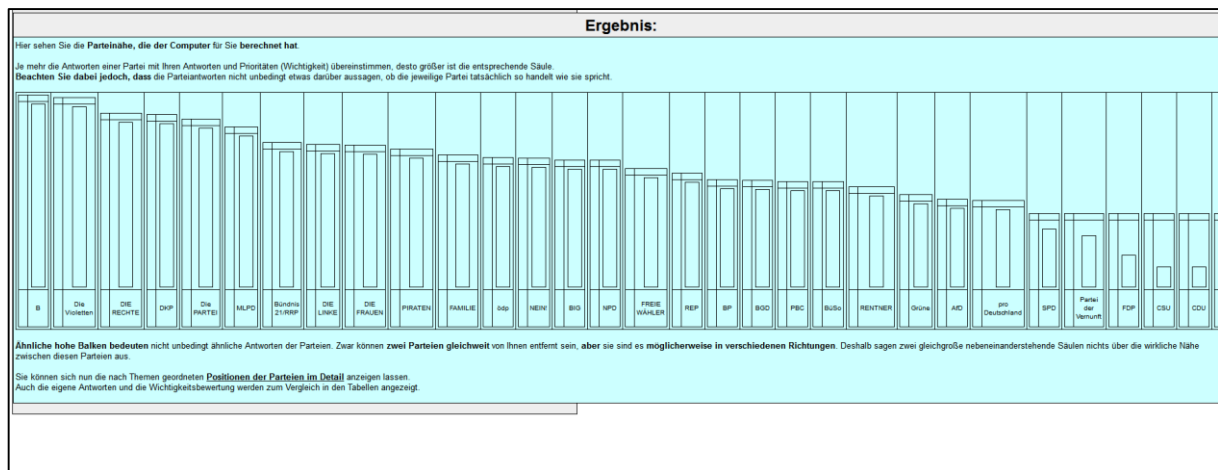


Abbildung 36: Darstellung der Ergebnisse bei parteivergleich.eu

| 30 Parteien zur Wahl des Deutschen Bundestages im Vergleich Bundestagswahl 2013 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|--------------|------|----|------|------|------|----------------|------|------|------|------------|-----------|------------|------------|-------------|------|---------|------|--------------|--------|------|-------|------|------|---------------------|------|---------|-----------------|---------|------|------|
| Demokratie und Staatsform (Teil 1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Demokratie und Staatsform (Teil 1) | Ihre Antwort | AfD | B | BIG | BGD | BP | Bündnis 21/RRP | BüSo | CDU* | CSU* | DIE FRAUEN | DIE LINKE | Die PARTEI | DIE RECHTE | Die Violett | DKP | FAMILIE | FDP | FREIE WÄHLER | Grüne* | MLPD | NEIN! | NPD | ödp | Partei der Vernunft | PBC | PIRATEN | pro Deutschland | RENTNER | REP | SPD* |
| Soll über alle Gesetze und Beschlüsse auf Bundesebene per Volksentscheid abgestimmt werden dürfen? | Ja | Ja | Ja | Nein | Nein | Ja | Nein | Nein | Nein | Nein | Nein | Ja | Nein | Ja | Ja | Ja | 0 | | 0 | Ja | Ja | Nein | Ja | Ja | Ja | Ja | Ja | Nein | Nein | Ja | 0 |
| Sind Volksbefragungen nötig, wenn staatliche Souveränität ganz oder teilweise übertragen wird? | Ja | Ja | Ja | Ja | Nein | Ja | Ja | Ja | Nein | Nein | 0 | Ja | Ja | Ja | Ja | Ja | Ja | 0 | Ja | Nein | Ja | Ja | Ja | Ja | Ja | 0 | Ja | Ja | Nein | Ja | Nein |
| Sind Wahlen deshalb nicht verboten, weil sie nichts ändern? | Ja | 0 | Ja | Nein | Nein | Nein | Nein | 0 | Nein | Nein | Nein | Nein | Nein | Ja | Ja | 0 | Nein | Nein | 0 | Nein | | 0 | Ja | Nein | Nein | 0 | Nein | Nein | Nein | Nein | Nein |
| Soll ein Wahlrecht ab Geburt eingeführt werden, bei dem Eltern pro Kind eine zusätzliche Wahlstimme haben (Stellvertreterwahlrecht)? | egal | Nein | 0 | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Ja | Nein | Nein | Nein | Nein | Nein | Ja | Nein | Ja | 0 | Nein | Nein | Nein | Nein | Nein |
| Ist Nichtwählen eine geeignete Möglichkeit, um alle zur Wahl stehenden Parteien abzulehnen? | Nein | Nein | 0 | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | 0 | Ja | Ja | Nein | Nein | Ja | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Nein | Ja | Nein | Nein |
| Sollen Fraktionszwang und Koalitionsverträge abgeschafft werden? | Ja | 0 | 0 | Ja | Ja | Ja | | 0 | Nein | Nein | Nein | Nein | Ja | Ja | Ja | 0 | 0 | Nein | | Nein | 0 | | Nein | Nein | Ja | 0 | Ja | Nein | | Ja | Nein |
| Demokratie und Staatsform (Teil 1) | Ihre Antwort | AfD | B | BIG | BGD | BP | Bündnis 21/RRP | BüSo | CDU* | CSU* | DIE FRAUEN | DIE LINKE | Die PARTEI | DIE RECHTE | Die Violett | DKP | FAMILIE | FDP | FREIE WÄHLER | Grüne* | MLPD | NEIN! | NPD | ödp | Partei der Vernunft | PBC | PIRATEN | pro Deutschland | RENTNER | REP | SPD* |
| Abkürzungen bei Ihren Antworten: 0 = Neutral, kleine Schriftgröße = kaum wichtig, mittlere Schriftgröße = wichtig, große Schriftgröße = extrem wichtig Abkürzungen bei den Parteiantworten: 0 = Neutral oder keine Antwort, unterstrichene Antworten enthalten einen Kommentar der jeweiligen Partei (um den Kommentar zu sehen, schieben Sie den Mauszeiger darüber, nicht anklicken!) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Die Kommentare der Parteien zum Thema Demokratie und Staatsform (Teil 1): Soll über alle Gesetze und Beschlüsse auf Bundesebene per Volksentscheid abgestimmt werden dürfen? Familien-Partei Deutschlands: Es soll über wichtige Gesetze und Beschlüsse auf Bundesebene per Volksentscheid abgestimmt werden dürfen FREIE WÄHLER: Ja, aber nicht alle Sind Wahlen deshalb nicht verboten, weil sie nichts ändern? Marxistisch-Leninistische Partei Deutschlands: Wahlen sollen dem Bürger die Illusion vermitteln, dass "alle Staatsgewalt vom Volk ausgeht". Tatsächlich aber wird unsere Politik bestimmt von dem allein herrschenden internationalen Finanzkapital, das aus den 500 größten Konzernen der Welt besteht und zu dem in Deutschland die 29 größten gehören. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Abbildung 37: Detailauswertung mit zufallsgenerierten Daten (parteivergleich.eu, 2016)

Tabelle 14: Ressourcen der Server API

| Ressource | Methode | Semantik | Content-Type (req) | Content-Type (res) |
|---|---------|--|--------------------|--------------------|
| /register | POST | Sich beim Server registrieren | Application/json | Application/json |
| /login | POST | Sich beim Server anmelden | Application/json | Application/json |
| /user | PUT | Informationen über den User aktualisieren | Application/json | Application/json |
| /user | DELETE | User aus dem System löschen | Application/json | - |
| /thesen | POST | Eine neue These veröffentlichen | Application/json | - |
| /thesen | DELETE | Thesen löschen | Application/json | |
| /thesen?tid=[] | PUT | Zu einer These eine Position (PRO/NEUTRAL/CONTRA) und optional eine Begründung öffentlich hinzufügen | Application/json | - |
| /thesen?tid=[] | DELETE | Eine Position zu einer These löschen | Application/json | - |
| /thesen?wahlkreis=[]&kategorie=[]&anzahl=[] | GET | Thesen aus einem Wahlkreis und einer Kategorie mit einer gewünschten Anzahl bekommen | - | Application/json |
| /kandidaten?wahlkreis=[] | GET | Alle Kandidaten eines Wahlkreises bekommen | - | Application/json |
| /kandidaten?kid=[] | GET | Infos über einen Kandidaten mit einer KID bekommen | - | Application/json |
| /matching | POST | Den Server die Übereinstimmung von persönlicher Ansicht und der Positionen der Kandidaten berechnen lassen | Application/json | Application/json |
| /fcm | POST | Sein Gerät für Googles Firebase Cloud Messaging registrieren | Application/json | Application/json |

User profiles für den Stakeholder „Wähler“

Tabelle 15: 1. User profile - Wähler

| Merkmal | Ausprägung |
|-------------------------------------|--|
| Demographische Charakteristiken | 18-55 Jahre |
| Fachwissen | Allgemeines Politisches Wissen und Wissen über die eigenen Interessen |
| Verfügbare Technologien | Computer und Smartphone |
| Computerkenntnisse und -erfahrungen | Wissen im Umgang mit Computern und Mobilien Geräten |
| Spezielle Produkterfahrung | Erfahrung mit Wahlhilfen |
| Motivation und Einstellung | Hat eine eigene Meinung und viele Ideen und möchte mehr über die Kandidaten des eigenen Wahlkreises wissen |
| Aufgaben | Recherchiert über Positionen der Kandidaten um eine geeignete Wahl zu treffen |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

Tabelle 16: 2. User profile - Wähler

| Merkmal | Ausprägung |
|-------------------------------------|---|
| Demographische Charakteristiken | 18-55 Jahre |
| Fachwissen | Weniger politisches Interesse und Wissen |
| Verfügbare Technologien | Computer und Smartphone |
| Computerkenntnisse und -erfahrungen | Wissen im Umgang mit Computern und Mobilen Geräten |
| Spezielle Produkterfahrung | Keine Erfahrung mit Wahlhilfen |
| Motivation und Einstellung | Durch weniger politisches Interesse besteht auch weniger Motivation den passenden Kandidaten zu finden |
| Aufgaben | Recherchiert weniger/gar nicht über Positionen der Kandidaten, da weniger politisches Interesse vorhanden ist |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

Tabelle 17: 3. User profile - Wähler

| Merkmal | Ausprägung |
|-------------------------------------|---|
| Demographische Charakteristiken | 55-90 Jahre |
| Fachwissen | Fundiertes Politisches Wissen und Wissen über die eigenen Interessen |
| Verfügbare Technologien | Computer oder Smartphone; Beide Geräte; |
| Computerkenntnisse und -erfahrungen | Eher weniger Wissen im Umgang mit Computern und Mobilien Geräten |
| Spezielle Produkterfahrung | Weniger Erfahrung mit Wahlhilfen |
| Motivation und Einstellung | Hat eine eigene Meinung und viele Ideen |
| Aufgaben | Recherchiert über Positionen der Kandidaten um eine geeignete Wahl zu treffen |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

Tabelle 18: 4. User profile - Wähler

| Merkmal | Ausprägung |
|-------------------------------------|---|
| Demographische Charakteristiken | 55-99 Jahre |
| Fachwissen | Weniger politisches Interesse und Wissen |
| Verfügbare Technologien | Computer oder Smartphone; Keine Geräte; |
| Computerkenntnisse und -erfahrungen | Eher weniger Wissen im Umgang mit Computern und Mobilien Geräten |
| Spezielle Produkterfahrung | Keine Erfahrung mit Wahlhilfen |
| Motivation und Einstellung | Durch weniger politisches Interesse besteht auch weniger Motivation den passenden Kandidaten zu finden |
| Aufgaben | Recherchiert weniger/gar nicht über Positionen der Kandidaten, da weniger politisches Interesse vorhanden ist |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

User profiles für den Stakeholder „Kandidat“

Tabelle 19: 1. User profile - Kandidat

| Merkmal | Ausprägung |
|-------------------------------------|--|
| Demographische Charakteristiken | 18-90 Jahre |
| Formale Qualifikationen | Jahrelanges Parteimitglied |
| Fachwissen | Ausgeprägtes Politisches Wissen und Wissen über die eigenen Interessen |
| Verfügbare Technologien | Computer und Smartphone |
| Computerkenntnisse und -erfahrungen | Wissen im Umgang mit Computern und Mobilien Geräten |
| Spezielle Produkterfahrung | Keine spezielle Erfahrung |
| Motivation und Einstellung | Möchte Wähler von seiner eigenen Position überzeugen |
| Aufgaben | Positioniert sich zu Thesen und begründet seine Position und stellt sich dem Wähler zur Wahl |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

Tabelle 20: 2. User profile - Kandidat

| Merkmal | Ausprägung |
|-------------------------------------|--|
| Demographische Charakteristiken | 18-90 Jahre |
| Formale Qualifikationen | Unabhängiger Kandidat |
| Fachwissen | Ausgeprägtes Politisches Wissen und Wissen über die eigenen Interessen |
| Verfügbare Technologien | Computer und Smartphone |
| Computerkenntnisse und -erfahrungen | Wissen im Umgang mit Computern und Mobilen Geräten |
| Spezielle Produkterfahrung | Keine spezielle Erfahrung |
| Motivation und Einstellung | Möchte Wähler von seiner eigenen Position überzeugen |
| Aufgaben | Positioniert sich zu Thesen und begründet seine Position und stellt sich dem Wähler zur Wahl |
| Auswirkungen von Fehlern | Unangemessenes Verhalten hat Einfluss auf die Seriosität des Systems |

Stakeholder Analyse

Tabelle 6 : Stakeholder Analyse

| Bezeichnung des Stakeholders | Beziehung zum System | Objektbereich | Erfordernis | Erwartung |
|------------------------------|----------------------|---------------------|---|--|
| Wähler | Anrecht | Gesamtsystem | Nutzung des Gesamtsystems | Die privaten Daten des Wählers müssen geschützt werden. |
| Wähler/Kandidaten | Anspruch | Merkmal des Systems | Vergleich von Ansicht des Wählers mit den Positionen der Kandidaten | Der Vergleich zwischen Ansicht des Wählers und der Positionen der Kandidaten muss konsistent und stetig sein |
| Wähler/Kandidaten | Anspruch | Merkmal des Systems | Veröffentlichung eigener Thesen | Eigene Thesen müssen öffentlich sichtbar sein |
| Wähler/Kandidaten | Anspruch | Gesamtsystem | Nutzung des Gesamtsystems | Lauffähigkeit des Systems entsprechend der Architektur |
| Wähler/Kandidaten | Anspruch | Gesamtsystem | Nutzung des Gesamtsystems | Korrektheit der Informationen |
| Wähler/Kandidaten | Anspruch | Merkmal des Systems | Berechnung der Prognose | Die Prognose soll korrekt errechnet werden |
| Wähler/Kandidaten | Anspruch | Merkmal des Systems | Kommunikation zwischen Stakeholdern | Über Thesen oder Begründungen der Kandidaten zu diskutieren oder zu bewerten |
| Wähler/Kandidaten | Interesse | Gesamtsystem | Nutzung des Gesamtsystems | Kein Missbrauch des Systems durch Nutzer |

| | | | | |
|-------------------|-----------|---------------------|--------------------|---|
| Wähler/Kandidaten | Interesse | Merkmal des Systems | Thesen Darstellung | Thesen sollten nach Aktualität, Beliebtheit oder unpositionierten Thesen sortiert werden können |
| Wähler/Kandidaten | Interesse | Merkmal des Systems | Eigenes Profil | Man sollte über neue Informationen zu eigenen oder abonnierten Thesen/Begründungen informiert werden. |
| Wähler/Kandidaten | Interesse | Merkmal des Systems | Eigenes Profil | Man sollte eine Auflistung aller schon positionierten Thesen sehen können. |
| Wähler/Kandidaten | Interesse | Merkmal des Systems | Statistiken | Es sollten Statistiken über die Beteiligung am System berechnet werden |
| Kandidaten | Interesse | Merkmal des Systems | Eigenes Profil | Man sollte seine Biographie und sein Wahlprogramm in seinem Profil veröffentlichen können |
| Wähler | Interesse | Merkmal des Systems | Wahlprogramme | Man sollte Informationen zu gebrochenen Wahlversprechen vorheriger Wahlen dargestellt bekommen |
| Wähler | Interesse | Merkmal des Systems | Kandidatenprofile | Man sollte die Vertrauenswürdigkeit der Kandidaten bewerten können |
| Wähler/Kandidaten | Interesse | Merkmal des Systems | Thesen | Es sollte nicht möglich sein, mehr als einmal die gleiche These in einem Wahlkreis zu veröffentlichen |

Generelle Designprinzipien

Tabelle 21: Designprinzipien

| Design Prinzip | Beschreibung |
|---|---|
| Visuelle Hierarchie | Relevante Inhalte sollten zuerst wahrgenommen werden, deshalb sollten die Inhalte hierarchisch angeordnet werden. |
| Hick's Law | „[...] with every additional choice increases the time required to take a decision.“ (cleverism, 2015) Die Auswahl der Alternativen darf nicht zu groß sein und Inhalte sollten kategorisiert werden, damit der Zeitaufwand für den Benutzer gering gehalten werden kann. |
| Fitt's Law | „[...] the time needed to move to a target is dependent upon the size of the target as well as the distance to the target.“ (cleverism, 2015) Die wichtigsten Elemente sollen erkennbar sein um den Benutzer leichter zu seinem Ziel zu führen. |
| „Material design Guidelines“ von Google | Für die Entwicklung der Android App, orientiert sich das Projekt an den „Material design Guidelines“ von Google. |

Präskriptive Use Cases

Tabelle 22: Use Case Definitionen

| | | |
|---|--|-------------------------------------|
| USE CASE #ID | | |
| Primary Actor | <Rollenbezeichnung und Beschreibung> | |
| Goal in Context | <ausführliche Beschreibung des Ziels> | |
| Scope | <Umfang: adressiertes System> | |
| Preconditions | <Ausgangssituation, Vorbedingungen> | |
| Success End Condition | <Ziel als intendierter Zustand> | |
| Failed End Condition | <Zustand bei dem das Ziel aufgegeben wird> | |
| Secondary Actors | <Rollenbezeichnung und Beschreibung> | |
| Trigger | <Auslöser (zeitliche Abhängigkeit)> | |
| DESCRIPTION <Erfolgsszenario als Aktivitäts- und Handlungsbeschreibung vom Trigger bis zum Ziel> | STEP | ACTION |
| | 1... | <Beschreibung der Aktion> |
| | | |
| EXTENSIONS <Erweiterung einzelner Schritte des Erfolgsszenarios> | STEP | Branching Action |
| | 1a... | <Bedingung>: <Aktion oder Use Case> |
| | | |
| SUBVARIATION <Liste von Variationen> | STEP | Branching Action |
| | 1... | |

Tabelle 23: USE CASE 1 Eine These veröffentlichen

| USE CASE 1: Eine These veröffentlichen | | |
|--|---|--|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Eine eigene These wird auf dem Server veröffentlicht und zur passenden Kategorie und dem Wahlkreis des Benutzers gespeichert | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | <p>Wähler/Kandidat ist beim System authentifiziert;</p> <p>Wähler/Kandidat hat seinen Wahlkreis festgelegt;</p> <p>Wähler/Kandidat hat das System nach seiner These durchsucht und nicht gefunden</p> | |
| Success End Condition | Wähler/Kandidat hat eine These veröffentlicht | |
| Failed End Condition | Wähler/Kandidat konnte die These nicht veröffentlichen oder hat die falsche Kategorie für die These ausgewählt | |
| Secondary Actors | - | |
| Trigger | Wähler/Kandidat möchte eine eigene These formulieren, damit die Kandidaten des Wahlkreises Position dazu beziehen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Hat eine eigene Position zu einem bestimmten Themengebiet |
| | 2 | Formulierung einer These |
| | 3 | Auswahl der passenden Kategorie |
| | 4 | Bestätigt die Veröffentlichung |
| EXTENSIONS | STEP | ACTION |
| | 2a | Keinen Text eingegeben: Erinnerung an den Benutzer noch einen Text einzugeben |
| | 3a | Keine Kategorie ausgewählt: Erinnerung an den Benutzer noch die passende Kategorie auszuwählen |

Tabelle 24: USE CASE 2 Eine Position zu einer These beziehen

| USE CASE 2: Eine Position zu einer These beziehen | | |
|---|--|--|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Position zu einer These wurde gespeichert | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | <p>Wähler/Kandidat ist beim System authentifiziert;</p> <p>Wähler/Kandidat hat seinen Wahlkreis festgelegt;</p> <p>Wähler/Kandidat ist zum Bereich „Thesen sehen“ navigiert;</p> | |
| Success End Condition | Wähler/Kandidat hat sich zu einer These positioniert und genau eine Position zu der These wurde gespeichert | |
| Failed End Condition | Wähler/Kandidat konnte sich nicht zu einer These positionieren oder es wurden mehr als eine Position zu einer These gespeichert | |
| Secondary Actors | - | |
| Trigger | Hat einen eigenen Standpunkt zu einer These und möchte diesen im System festhalten | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Entscheidet aus welcher Kategorie er Thesen sehen möchte |
| | 2 | Liest eine These |
| | 3 | Bildet sich seine eigene Meinung zu dieser These |
| | 4 | Bezieht eine Position und teilt sie dem System mit |
| | 5 | Bekommt eine Rückmeldung vom System |
| EXTENSIONS | STEP | Branching Action |
| | 1a | Möchte alle Thesen des Wahlkreises sehen |

Tabelle 25: USE CASE 3: Eine Begründung zu einer These hinzufügen

| USE CASE 3: Eine Begründung zu einer These hinzufügen | | |
|---|--|---|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Wähler/Kandidat hat eine Begründung seiner Position zu einer These hinzugefügt, welche dann veröffentlicht wird | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | <p>Wähler/Kandidat ist beim System authentifiziert;</p> <p>Wähler/Kandidat hat seinen Wahlkreis festgelegt;</p> <p>Wähler/Kandidat hat zum Bereich „Thesen sehen“ navigiert;</p> | |
| Success End Condition | Wähler/Kandidat hat eine Begründung seiner Position zu einer These hinzugefügt, welche dann veröffentlicht wird | |
| Failed End Condition | Wähler kann keine Begründung seiner Position zu einer These hinzufügen oder die Begründung wird zur falschen Position erfasst | |
| Secondary Actors | - | |
| Trigger | Wähler/Kandidat möchte eine Begründung zu einer These veröffentlichen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Liest sich eine These durch |
| | 2 | Bezieht eine Position zu der These |
| | 3 | Formuliert seine Begründung |
| | 4 | Bestätigt und veröffentlicht seine Begründung |
| EXTENSIONS | STEP | Branching Action |
| | 3a | Möchte seine Begründung überarbeiten |

Tabelle 26: USE CASE 4 Kommentar zu einer Begründung hinzufügen

| USE CASE 4: Kommentar zu einer Begründung hinzufügen | | |
|--|--|--|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Wähler/Kandidat hat ein Kommentar zu einer Begründung abgegeben, welcher dann veröffentlicht wird | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | <p>Wähler/Kandidat ist beim System authentifiziert;</p> <p>Wähler/Kandidat hat seinen Wahlkreis festgelegt;</p> <p>Wähler/Kandidat hat zur genauen Ansicht einer These navigiert</p> | |
| Success End Condition | Wähler hat ein Kommentar zu einer Begründung abgegeben, welcher dann veröffentlicht wurde | |
| Failed End Condition | Wähler/Kandidat könnte keinen Kommentar zu einer Begründung abgeben | |
| Secondary Actors | | |
| Trigger | Wähler/Kandidat möchte eine Begründung kommentieren | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Liest sich eine Begründung zu einer Position durch |
| | 2 | Entscheidet sich einen Kommentar zu verfassen |
| | 3 | Verfasst den Kommentar |
| | 4 | Bestätigt und veröffentlicht den Kommentar |

Tabelle 27: USE CASE 5 Die Übereinstimmung berechnen lassen

| USE CASE 5: Die Übereinstimmung berechnen lassen | | |
|--|--|---|
| Primary Actor | Wähler | |
| Goal in Context | Die Kandidaten mit der größtmöglichen Übereinstimmung zu der eigenen Meinung wurden berechnet und dargestellt | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Wähler ist beim System authentifiziert; Wähler hat seinen Wahlkreis festgelegt; Der Wähler und die Kandidaten des Wahlkreises haben Positionen zu mehreren Thesen bezogen; | |
| Success End Condition | Wähler kennt den passenden Kandidaten mit der größtmöglichen Übereinstimmung zu den Positionen | |
| Failed End Condition | Wähler findet keinen Kandidaten der mit seinen Positionen übereinstimmt oder die Berechnung verlief fehlerhaft | |
| Secondary Actors | Kandidaten des eigenen Wahlkreises | |
| Trigger | Der Wähler möchte die Übereinstimmung zu den Kandidaten des Wahlkreises berechnen lassen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Navigiert zum „Matching“ |
| | 2 | Der Wähler lässt die Übereinstimmung mit den Kandidaten berechnen |
| | 3 | Die Ergebnisse werden dargestellt |

Tabelle 28: USE CASE 6 Thesen bewerten

| USE CASE 6: Thesen bewerten | | |
|-----------------------------|--|----------------------------|
| Primary Actor | Wähler | |
| Goal in Context | Der Wähler hat eine These positiv bewertet und der öffentliche Zähler der These wurde hochgezählt | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Wähler ist beim System authentifiziert; Wähler hat seinen Wahlkreis festgelegt; Wähler hat zum Bereich „Thesen sehen“ navigiert; | |
| Success End Condition | Wähler hat die These positiv bewertet | |
| Failed End Condition | Wähler konnte nicht die These bewerten oder der Zähler der These wurde nicht erhöht | |
| Secondary Actors | - | |
| Trigger | Der Wähler findet eine These gut und möchte, dass die Kandidaten zu dieser These Position beziehen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Liest sich die These durch |
| | 2 | Findet die These gut |
| | 3 | Bewertet die These positiv |

Tabelle 29: USE CASE 7 Thesen sortieren

| USE CASE 7: Thesen sortieren | | |
|------------------------------|--|-----------------------------------|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Der Wähler/Kandidat hat einen Sortiermodus ausgewählt und die Thesen werden nach diesem sortiert | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Wähler ist beim System authentifiziert; Wähler hat seinen Wahlkreis festgelegt; Wähler hat zum Bereich „Thesen sehen“ navigiert; | |
| Success End Condition | Die Thesen werden nach dem gewünschten Sortiermodus sortiert und dargestellt | |
| Failed End Condition | Die Thesen konnten nicht sortiert werden | |
| Secondary Actors | - | |
| Trigger | Der Wähler/Kandidat möchte die Thesen nach Beliebtheit oder Aktualität sortieren | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Sich für Sortiermodus entscheiden |
| | 2 | Den Sortiermodus auswählen |

Tabelle 30: USE CASE 8 Thesen nach Begriffen durchsuchen

| USE CASE 8: Thesen nach Begriffen durchsuchen | | |
|---|--|---|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Der Wähler/Kandidat hat die Thesen zu dem gesuchten Begriff gefunden | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Wähler ist beim System authentifiziert; Wähler hat seinen Wahlkreis festgelegt; Wähler hat zum Bereich „Thesen sehen“ navigiert; | |
| Success End Condition | Die Thesen, welche den gewünschten Begriff beinhalten, werden angezeigt | |
| Failed End Condition | Es wurden keine Thesen zu dem gewünschten Begriff gefunden | |
| Secondary Actors | - | |
| Trigger | Der Wähler/Kandidat möchte Thesen nach einem gewünschten Begriff suchen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Überlegt sich einen Begriff |
| | 2 | Gibt diesen Begriff in eine Suchleiste ein |
| | 3 | Bestätigt die Eingabe |
| | 4 | Liest die Thesen zu seinem ausgewählten Begriff |

Tabelle 31: USE CASE 9 Registrieren

| USE CASE 9: Registrieren | | |
|--------------------------|--|---|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Der Wähler/Kandidat wurde im System mit seinen Daten gespeichert | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Die App wurde installiert | |
| Success End Condition | Der Wähler/Kandidat hat sich erfolgreich beim System registriert | |
| Failed End Condition | Der Wähler/Kandidat konnte sich nicht im System registrieren | |
| Secondary Actors | - | |
| Trigger | Der Wähler/Kandidat möchte eigene Thesen veröffentlichen, den passenden Kandidaten finden usw. | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Öffnet die Anwendung |
| | 2 | Geht zu „Registrieren“ |
| | 3 | Gibt den Benutzernamen, Email, Wahlkreis und Passwort ein |
| | 4 | Die Eingabe wird bestätigt und abgeschickt |

Tabelle 32: USE CASE 10 Einloggen

| USE CASE 10: Einloggen | | |
|------------------------|---|---|
| Primary Actor | Wähler/Kandidat | |
| Goal in Context | Der Wähler/Kandidat hat sich im System eingeloggt | |
| Scope | Thes-O-Naise Android Client | |
| Preconditions | Die App wurde installiert; Im System Registriert; | |
| Success End Condition | Der Wähler/Kandidat hat sich erfolgreich im System eingeloggt | |
| Failed End Condition | Der Wähler/Kandidat konnte sich nicht im System einloggen | |
| Secondary Actors | - | |
| Trigger | Der Wähler/Kandidat möchte die Möglichkeiten der Anwendung benutzen | |
| DESCRIPTION | STEP | ACTION |
| | 1 | Öffnet die Anwendung |
| | 2 | Geht zu „Einloggen“ |
| | 3 | Gibt seinen Benutzernamen oder seine Email-Adresse und Passwort ein |
| | 4 | Die Eingabe wird bestätigt und abgeschickt |

1. USE CASE: Eine These veröffentlichen

1. Nach der Registrierung/Anmeldung den Navigationsbutton finden; Aus den verschiedenen Auswahlmöglichkeiten zu „These veröffentlichen“ gehen

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Erkennen wo das Texteingabefeld für eine These ist; Die Kategorie Auswahl finden; Sich eine Kategorie für die eigene These aussuchen; Den „Veröffentlichen“ Button drücken

Verletzte Heuristiken: **Error prevention** und **Help users recognize, diagnose, and recover from errors**

Tabelle 33: Cognitive Walkthrough – Use Case 1

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Evtl. Nein(#3) | Ja |

2. USE CASE: Eine Position zu einer These beziehen

1. Zu „Thesen sehen“ Navigieren

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Nach lesen einer These die Radiobuttons finden und die entsprechende Position „Pro“, „Neutral“ oder „Contra“ drücken

Bei diesem Schritt werden keine Heuristiken verletzt.

Tabelle 34: Cognitive Walkthrough – Use Case 2

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Ja | Ja |

3. USE CASE: Eine Begründung zu einer These hinzufügen

1. Zu „Thesen sehen“ navigieren; Eine These aus der Liste auswählen; Den „Mehr“ Button zu einer These finden und drücken;

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Sich für eine Position der Begründung entscheiden; Das Texteingabefeld für eine Begründung finden; Nachdem eine Begründung verfasst wurde, den „Abschickpfeil“ finden und drücken

Bei diesem Schritt werden keine Heuristiken verletzt.

Extension:

3. Findet und drückt den „Bearbeiten“ Button wenn er seine Begründung ändern möchte; Bearbeitet seine Begründung

Verletzte Heuristiken: **User control and freedom**

4. Bestätigt seine Eingabe

Tabelle 35: Cognitive Walkthrough – Use Case 3

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Ja | Ja |
| 3 | Ja | Nein(#4) da nicht vorhanden | - | - |
| 4 | - | - | - | - |

4. USE CASE: Kommentar zu einer Begründung hinzufügen

1. Zu „Thesen sehen“ navigieren; Eine These aus der Liste auswählen; Den „Mehr“ Button zu einer These finden und drücken;

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Erkennen in welcher Position zu einer Begründung man sich befindet; Eine Begründung auswählen; Erkennen wo sich das Texteingabefeld für Kommentare befindet; Nach der Eingabe eines Kommentars, den „Abschickpfeil“ finden und drücken

Bei diesem Schritt werden keine Heuristiken verletzt.

Tabelle 36: Cognitive Walkthrough – Use Case 4

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Ja | Ja |

5. USE CASE: Die Übereinstimmung berechnen lassen

1. Zu „Matching“ navigieren; Den „Matching“ Button finden und drücken

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Ansicht und Reihenfolge der Auflistung der Kandidaten verstehen und erkennen welcher Kandidat am nächsten zur eigenen Meinung ist; Finden und drücken des „Mehr“ Buttons für die Ansicht der Positionen des jeweiligen Kandidaten zu Thesen

Verletzte Heuristiken: **User control and freedom**

3. Verstehen welche Position der Kandidat zu den jeweiligen Thesen hat

Bei diesem Schritt werden keine Heuristiken verletzt.

Tabelle 37: Cognitive Walkthrough – Use Case 5

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Evtl. Nein(#2) | Ja | Ja | Ja |
| 3 | Ja | Ja | Ja | Ja |

6. USE CASE: Thesen bewerten

1. Zu „Thesen sehen“ navigieren; Den „Like“ Button erkennen und drücken wenn die These gefällt

Verletzte Heuristiken: **Visibility of system status**

Tabelle 38: Cognitive Walkthrough – Use Case 6

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Evtl. Nein (#1) | Ja |

7. USE CASE: Thesen sortieren

1. Zu „Thesen sehen“ navigieren; Erkennen wo sich die Sortieroptionen befinden; Die einzelnen Sortieroptionen verstehen

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Eine Sortieroptionen auswählen und drücken

Bei diesem Schritt werden keine Heuristiken verletzt.

Tabelle 39: Cognitive Walkthrough – Use Case 7

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Ja | Ja |

8. USE CASE: Thesen nach Begriffen durchsuchen

1. Zu „Thesen sehen“ navigieren; Erkennen wo sich die Suchleiste befindet

Bei diesem Schritt werden keine Heuristiken verletzt.

2. Einen Begriff in die Suchleiste eintragen und abschicken; Verstehen wie sich die Thesen nach dem Suchbegriff sortiert haben

Bei diesem Schritt werden keine Heuristiken verletzt.

Tabelle 40: Cognitive Walkthrough – Use Case 8

| Schritt | Wird der Benutzer versuchen, den richtigen Effekt zu erzielen? | Wird der Benutzer erkennen, dass die korrekte Aktion zur Verfügung steht? | Wird der Benutzer eine Verbindung herstellen zwischen der korrekten Aktion und dem gewünschten Effekt? | Wenn die korrekte Aktion ausgeführt worden ist: wird der Benutzer den Fortschritt erkennen? |
|---------|--|---|--|---|
| 1 | Ja | Ja | Ja | Ja |
| 2 | Ja | Ja | Ja | Ja |

Usability Probleme

Tabelle 41: Identifizierte Usability Probleme

| Nr. | Kurzbeschreibung des Problems | Schweregrad | Priorisierung | Häufigkeit | Aufwand | Maßnahmen |
|-----|--|-------------|---------------|------------|---------|---|
| 1. | Beim drücken des „Like“ Buttons bei einer These, erkennt der Benutzer evtl. nicht ob er diese These „geliked“ hat oder nicht | gering | niedrig | 1x | gering | Der „Like“ Button muss sich nach Aktivierung umfärben |
| 2 | Der Benutzer wird evtl. nicht erkennen, dass die durchschnittliche Punktzahl pro These mehr über das Ergebnis aussagt, als die Punktzahl des Abstands der Positionen | mittel | mittel | 1x | gering | Der Wähler sollte die Ergebnis Ansicht aller Kandidaten auch nach dem Durchschnitt der Punktzahl sortieren können |
| 3 | Falls der Benutzer für seine These die falsche Kategorie ausgewählt hat, kann er die Kategorie im Nachhinein nicht mehr ändern | mittel | mittel | 1x | gering | Nachdem der Benutzer den „Veröffentlichen“ Button betätigt hat, soll eine Meldung erscheinen mit dem Thesentext und der ausgewählten Kategorie, welche nochmals bestätigt werden muss |
| 4 | Falls der Benutzer seine Begründung bearbeiten möchte steht ihm keine Interaktionsmöglichkeit zur Verfügung | schwer | hoch | 1x | gering | Es soll eine Interaktionsmöglichkeit geben, um die Begründung bearbeiten zu können |
| | | | | | | |