

Pflichtenheft

Praxis der Softwareentwicklung

Gruppe 1

Entwicklung einer Software zur Berechnung
der Mandatsverteilung im Deutschen
Bundestag

Philipp Löwer, Anton Mehlmann, Manuel Olk, Enes Ördek,
Simon Schürg, Nick Vlasoff



WS 2013 / 14

Inhaltsverzeichnis

1	Produktübersicht	3
1.1	Lizenz	3
2	Zielbestimmung	3
2.1	Musskriterien	3
2.2	Sollkriterien	3
2.3	Kannkriterien	3
2.4	Abgrenzungskriterien	3
3	Produkteinsatz	3
3.1	Anwendungsbereiche	3
3.2	Zielgruppen	3
3.3	Betriebsbedingungen	3
4	Produktumgebung	4
4.1	Software	4
4.2	Hardware	4
4.3	Orgware	4
4.4	Schnittstellen	4
5	Funktionale Anforderungen	4
5.1	GUI	4
5.2	Schnittstellen	6
5.3	Datenhaltung und Verarbeitung	6
6	Produktdaten	7
7	Produktleistungen	7
8	Nicht-funktionale Anforderungen	8
9	Qualitätsanforderungen	8
10	Globale Testfälle und Szenarien	9
11	Systemmodelle	9
11.1	Systemarchitektur	9
12	Benutzungsoberfläche	10
13	Spezielle Anforderungen an die Entwicklungsumgebung	10
14	Zeit- und Ressourcenplanung	10
15	Ergänzungen	10
16	Glossar	10

1 Produktübersicht

Bei dem Produkt handelt es sich um ein Programm, das die Sitzverteilung im Deutschen Bundestag gemäß der gesetzlichen Bestimmungen exakt berechnet und deren Zustandekommen verständlich darlegt. Erreicht wird dies durch eine minimalistische und intuitive Benutzeroberfläche.

1.1 Lizenz

Der Quellcode des Programms wird der Öffentlichkeit frei zur Verfügung gestellt. Es wird die GPL V3 Lizenz verwendet, damit das Projekt nach einer Modifizierung weiterhin öffentlich bleibt.

2 Zielbestimmung

2.1 Musskriterien

- Auswertung von Wahlergebnissen nach gesetzlicher Bestimmung
- Grafische Benutzeroberfläche
- Vergleich mehrerer Wahlausgänge
- Importmöglichkeit von Wahlergebnissen (.csv)
- Manipulation von Eingabedaten

2.2 Sollkriterien

- Auffinden paradoxer Wahlausgänge
- Kartographische Darstellung der Bundesländer

2.3 Kannkriterien

- Hilfe (Benutzerhandbuch)

2.4 Abgrenzungskriterien

- Keine Mobile Anwendung oder Web-App
- Keine namentliche Nennung von Abgeordneten

3 Produkteinsatz

3.1 Anwendungsbereiche

- Überprüfung der Wahlergebnisse

3.2 Zielgruppen

- Menschen, die kritisch gegenüber Wahlergebnissen sind
- (Unabhängige) Medien
- Politisch Interessierte

3.3 Betriebsbedingungen

- Keine Verbindung zum Internet nötig

4 Produktumgebung

4.1 Software

- Java Runtime Environment SE 1.7 oder neuer.
- Betriebssystem z.B. Windows, Linux, Mac OS

4.2 Hardware

Mindestanforderungen:

- 128 MB Arbeitsspeicher
- 100 MB freien Festplattenspeicher
- 500-MHz-Prozessor
- Farbdisplay/ Bildschirmauflösung: 1024x768

Empfohlen:

- 512 MB Arbeitsspeicher
- 100 MB freien Festplattenspeicher
- 1-GHz-Prozessor
- Farbdisplay/ Bildschirmauflösung: 1024x768

4.3 Orgware

- Keine weiteren Rahmenbedingungen notwendig

4.4 Schnittstellen

- Importieren/ Exportieren von Zuständen und Daten

5 Funktionale Anforderungen

Funktionale Anforderungen werden durch eine vierstellige Nummer gekennzeichnet. Die erste Nummer kennzeichnet den folgenden Bereich:

1. GUI
2. Schnittstellen
3. Datenhaltung

Die restlichen Nummern dienen zur Durchnummerierung.

5.1 GUI

- /F10010/ Programmstart
Es erscheint ein Startfenster. Der Benutzer hat die Wahl zwischen der Bundestagswahl 2013, falls vorhanden dem letzten Programmzustand (/F20010/), das Laden eines vorher gespeicherten Programmzustandes und das Importieren von Wahlergebnissen.
- /F10020/ Menü
Im Menü sind folgende Punkte gelistet:

- Datei
 - * Neuer Tab /F20001/
 - * Tab schließen /F20002/
 - * Speichern des aktuellen Zustandes /F20010/
 - * Laden eines Zustandes /F20011/
 - * Importieren von Daten /F20020/
 - * Exportieren von Daten /F20030/
 - * Beenden /F20070/
 - Bearbeiten
 - * Rückgängig /F20040/
 - * Wiederherstellen /F20041/
 - Extras
 - * Ändern der aktuellen Ansicht
 - * Wahlgesetz auswählen
 - * Vergleichen mit...
 - Hilfe /F10030/
- /F10030/ Hilfe

Durch den Klick auf diesen Menüpunkt öffnet sich ein Fenster, in der Handbücher zum Programm zu finden sind. Des weiteren ein kleines About, mit den wichtigsten Informationen zum Programm, vorhanden.
 - /F10060/ Programmfenster

Das Programmfenster wird in drei Bereiche aufgeteilt.

 - Tabellenfenster /F10070/
 - Diagrammfenster /F10080/
 - Kartenfenster /F10090/

Es gibt hierbei verschiedene Ansichten.

 - Bundesansicht
 - Landesansicht
 - /F10070/ Tabellenfenster

Es werden Bundesländer in tabellarischer Form angezeigt. Klickt der Benutzer auf ein Bundesland, werden die Wahlkreise des betroffenen Bundeslandes angezeigt. Der klick auf die Wahlkreise öffnen die Ergebnisse pro Partei, die Anzahl der Wahlberechtigten und die Zweitstimmen.

Es gibt die Möglichkeit die Sortierung zu ändern.

Ein zurück Pfeil wechselt zur vorherigen Ansicht.
 - /F10080/ Diagrammfenster

Im Diagrammfenster sieht man Diagramme den Daten entsprechend und zur aktuellen Ansicht.
 - /F10090/ Kartenfenster

Die Länder werden nach einer Überprüfung (/F30010/), kartografisch im Fenster dargestellt. Der klick auf ein Bundesland öffnet die Bundesansicht.

Ein zurück Pfeil wechselt zur vorherigen Bundesansicht.
 - /F20091/ Neuer Tab

Beim Anklicken wird ein neuer Tab (/PD02/) generiert, in dem eine neue Bundestagswahl (/PD03/) geladen werden kann. Diese Dateiauswahl korrespondiert zu dem ”+Button in der Tab leiste.

- /F20092/ Tab schließen
Beim Anklicken wird der aktuelle Tab (/PD02/) geschlossen. Wurden Änderung an den Dateien (/PD02/) vorgenommen, wird vor dem Schließen des Tabs dem Benutzer die Möglichkeit gegeben seine Einstellungen zu speichern, zu verwerfen oder das Schließen abzubrechen.
- /F20093/ Rückgängig machen
Es können Operationen rückgängig gemacht werden. Hierfür wird das History-Objekt (/PD05/) verwendet.
- /F20094/ Wiederherstellen
Falls Änderungen rückgängig gemacht wurden (/F20093/), können diese wieder hergestellt werden.
- /F10100/ Programmende
Es erscheint ein Fenster. Der Benutzer hat die Wahl zwischen abbrechen, speichern des aktuellen Zustands (/F20010/) und dem Verwerfen des aktuellen Zustands.

5.2 Schnittstellen

- /F20010/ Speichern des aktuellen Programzustandes
Es wird das Zustands-Objekt (/PD01/) in einer Datei abgelegt. Dabei kann der Benutzer beim Speichern einen internen Namen und ein Kommentar abgeben, welche mit gespeichert werden.
- /F20011/ Laden eines Programzustandes
Der Benutzer wählt eine Datei aus. Es wird überprüft, ob es sich um ein gültiges Objekt handelt. Falls die Version unterschiedlich ist, wird die Datei konvertiert und geladen.
- /F20020/ Importieren von Daten
Die .csv-Dateien des Bundeswahlleiters können importiert werden.
- /F20030/ Exportieren von Daten
Daten können als .csv-Dateien oder als JSON/XML exportiert werden.

5.3 Datenhaltung und Verarbeitung

- /F30010/ Überprüfen der Ländernamen
Überprüft ob die eingegebenen Ländernamen in dem Daten-Objekt (/PD03/) korrekt sind. Falls alle Ländernamen gefunden werden, wird die Kartografische Darstellung (/F10090/) aktiviert.
- /F30020/ Überprüfen der Stimmen
Überprüft ob die eingegebenen Stimmen in dem Daten-Objekt (/PD03/) korrekt sind. Falls die Anzahl der Stimmen ≥ 0 sind, kann die Sitzverteilung mit dem Wahlgesetz-Objekt (/PD04/) berechnet werden.
- /F30030/ Filtern der relevanten Daten
Die benötigten Dateien werden aus der ausgewählten .csv-Datei geladen und zu der Model-Klasse geschickt.
- /F30040/ Generierung von Wahlzeiten nach vorher definierten Mustern.
Welche Muster? ToDo!!!
- /F30050/ Manuelles ändern einzelner Stimmzahlen.
Der Benutzer kann in einer Tabelle die Zahlen der aktuellen Wahlsimulation manuell anpassen.
- /F30060/ Paradoxe Wahlausgänge
TODO
- /F30070/ Auswerten der Wahlergebnisse
- /F30080/ Verändern der Wahlergebnisse
- /F30090/ Generierung zufälliger Wahlergebnisse

6 Produktdaten

- /PD01/ Zustands-Object
Repräsentiert den aktuellen Zustand des Programms. Es beinhaltet alle Informationen, um den genauen Zustand des Programms wiederherzustellen. Es handelt sich um eine serialisierbare Klasse.
 - Version des Programms (um Abwärtskompatibilität zu gewährleisten)
 - Datum und Uhrzeit der Erstellung
 - Name/ID
 - Kommentar
 - Fenster-Objekt als Liste
 - History-Objekt
- /PD02/ Fenster-Objekt
 - Wahlergebnisse (Daten-Objekt)
 - Wahlgesetz-Objekt
 - Name/ID
- /PD03/ Daten-Objekt
Beinhaltet die Anzahl der (Erst- und Zweit-)Stimmen je Wahlkreis und Partei.
 - Name/ID
 - Kommentar (Quelle)
 - Wahlkreise mit Stimmen als Array.
- /PD04/ Wahlgesetz-Objekt
Beinhaltet das Algorithmus zur Berechnung der Sitze mit dem Daten-Objekt als Eingabedatum. Des weiteren überprüft das Objekt, ob mit dem Datenobjekt eine Wahl überhaupt simuliert werden kann (wenn ein Datenobjekt beispielsweise keine Erststimmen enthält wie in sehr alten Wahlen).
- /PD05/ History-Objekt
Dieses Objekt zeichnet alle Veränderungen am Programm auf. Mithilfe dieses Objektes können Operationen über das Menü Bearbeiten oder STRG+Z rückgängig gemacht werden (/F10020/ und /F20040/).

7 Produktleistungen

- Zeit
 - Starten + Laden des letzten Zustandes: unter 5 Sekunden.
 - Beenden + Speichern des aktuellen Zustandes: unter 5 Sekunden.
 - Exportieren/Importieren von Daten: unter 10 Sekunden.
- Genauigkeit
Die Genauigkeit des Algorithmus zur Sitzberechnung muss dem Wahlgesetz entsprechen und exakte Ergebnisse liefern.

8 Nicht-funktionale Anforderungen

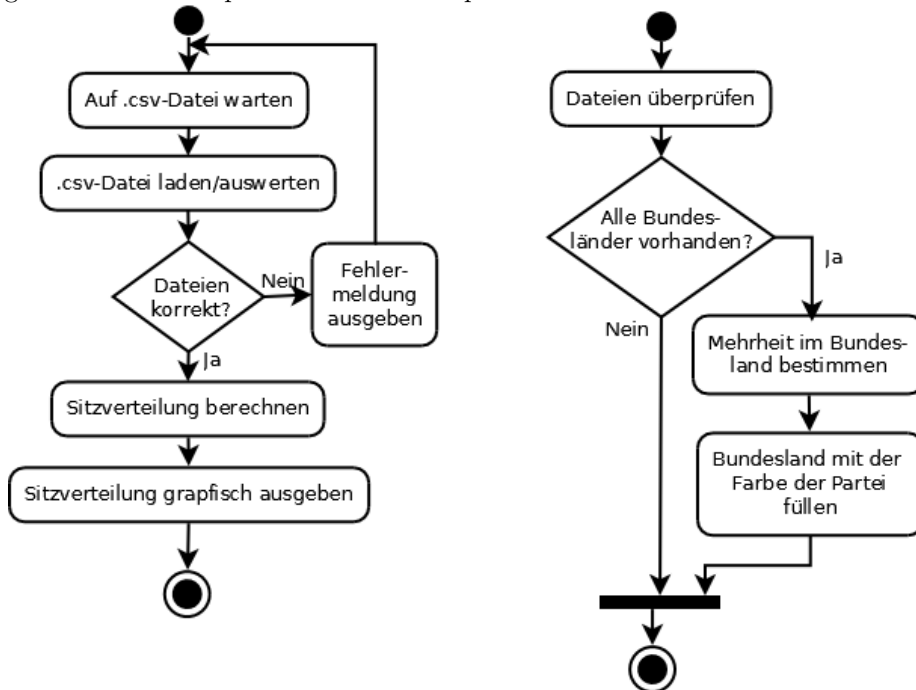
- Allgemeine Anforderungen:
 - Die Sitzverteilung muss für den Benutzer transparent und nachvollziehbar dargestellt werden.
- Sicherheitsanforderungen:
 - Die Eingabedaten dürfen während der Berechnung nicht verändert werden.
- Plattformunabhängigkeit:
Das Programm muss mit der offiziellen Oracle JRE laufen.

9 Qualitätsanforderungen

- Hilfreiche Fehlermeldungen
- Kein Datenverlust (auch nach Programmabstürzen)
- Gespeicherte Daten müssen immer konsistent gehalten werden

10 Globale Testfälle und Szenarien

Folgende Funktionssequenzen sind zu überprüfen:



Folgende Datenkonsistenzen müssen eingehalten werden:

- noch nichts...

Folgende unzulässige Aktionen müssen korrekt behandelt werden:

- Negative Stimmenanzahl
- Buchstaben als Stimmen

Testszenarien:

- Falsche Daten importieren:
Starten des Programms → Im Hauptmenü auf Datei klicken → Datei importieren auswählen → Im Dateibrowser die falsche .csv-Datei auswählen → Mit dem Button Laden bestätigen → Eine Fehlermeldung taucht auf → Programm befindet sich wieder im Startzustand
- Manuell Daten modifizieren:
Starten des Programms → Korrekte Daten laden → Die Sitzverteilung wird angezeigt → Den Wert zweier Parteien miteinander tauschen → Die Sitzverteilung erneut berechnen → Eine mögliche Veränderung der Sitzverteilung wird angezeigt

11 Systemmodelle

11.1 Systemarchitektur

Das Programm basiert auf der MVC- Architektur, wobei auf eine saubere Trennung der Einheiten Model, View und Controller geachtet wird. Dies sorgt nicht nur für einen flexiblen Programmentwurf, so dass spätere Änderungen bzw. Erweiterungen erleichtert werden, sondern garantiert auch die Trennung kritischer Komponenten, wie der Algorithmusimplementierung, von weniger sensiblen Komponenten, wie der GUI, und dient allgemein der Übersichtlichkeit.

12 Benutzungsoberfläche

13 Spezielle Anforderungen an die Entwicklungsumgebung

- Allgemein
 - Latex
 - Versionskontrolle mit SVN
- Entwicklung
 - IDE: Eclipse
 - GUI: Swing
- Entwurf
 - DIA für Diagramme
- Validierung
 - JUnit
- Teamkommunikation
 - Google-Groups Mailingliste

14 Zeit- und Ressourcenplanung

15 Ergänzungen

16 Glossar