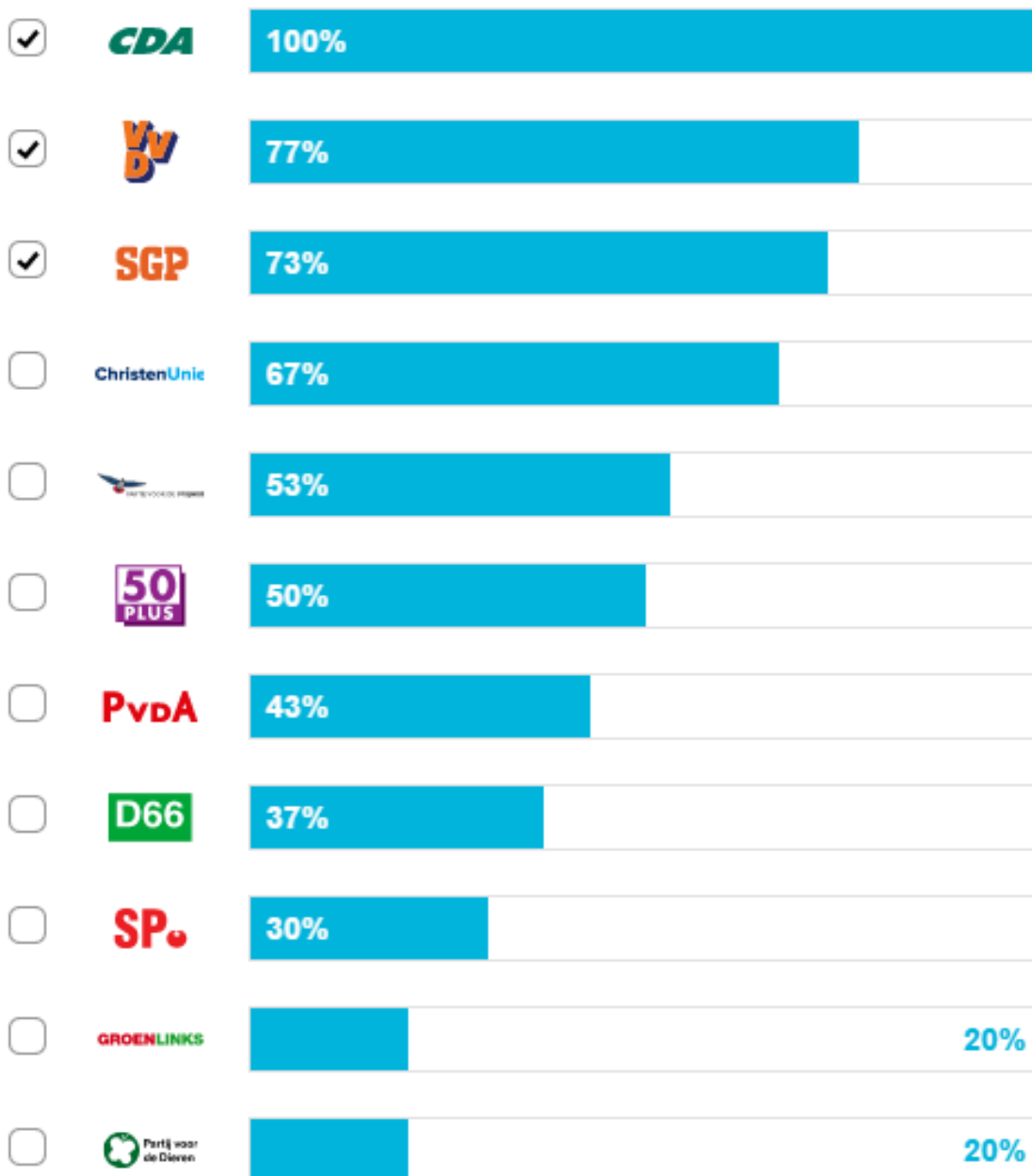


Use case realisatie

Stemwijzer



Kevin de Graaf
363112
Hanzehogeschool
Groningen
22-06-2025

Inleiding

In dit document worden de belangrijkste use cases(tabel 1) gerealiseerd voor de kieswijzer applicatie. Voor elke use case wordt een korte functionele beschrijving gegeven, gevolgd door een technische realisatie. In deze realisatie wordt uitgelegd hoe de flow van de use case verloopt binnen het zelfontwikkelde backend framework. Hierbij wordt rekening gehouden met de architectuur zoals voorgeschreven in de opdracht.

Tabel 1. Belangrijkste use cases.

Use case	Actor	Doel
Gebruiker maakt een account aan.	Gewone gebruiker	Een gebruiker registreert zich zodat zijn resultaten opgeslagen kunnen worden.
Gebruiker vult de stemwijzer in.	Gewone gebruiker.	De gebruiker vult de stellingen in om te zien met welke partijen hij het eens is.
Partij vult antwoorden op een stelling in.	Partij	Een partij geeft per stelling aan wat hun mening is en kan dit toelichten.
Beheerder maakt een nieuwe stelling aan.	Beheerder	De beheerder voegt een nieuwe stelling toe die zichtbaar wordt in de stemwijzer.
Beheerder maakt een account aan.	Beheerder	De beheerder maakt een nieuw account aan(voor bijvoorbeeld een partij of nieuwe admin).
Beheerder verwijdert een stelling.	Beheerder	De beheerder verwijdert een stelling om de lijst up-to-date te houden.
Beheerder verwijdert een account.	Beheerder	De beheerder kan ongebruikte accounts verwijderen en toegang afnemen.

Use cases

Use case 1: Gebruiker maakt een account aan.

Een niet-geregistreerde gebruiker wil een account aanmaken om toegang te krijgen tot extra functionaliteiten van de kieswijzer, zoals het opslaan en terugzien van eerdere resultaten. De gebruiker navigeert naar het registratieformulier, vult zijn gegevens in (zoals gebruikersnaam en wachtwoord) en bevestigt de registratie. Na succesvolle registratie wordt hij automatisch ingelogd of doorgestuurd naar de loginpagina.

Technische realisatie:

De browser doet een GET-request naar de backend om de registratiepagina op te halen. De router bepaalt welke controller deze request afhandelt. De controller gebruikt de templating-engine om de HTML van het registratieformulier te genereren en terug te sturen naar de gebruiker.

Wanneer de gebruiker het formulier invult en verzendt, wordt er een POST-request naar de backend gestuurd met de ingevulde gegevens. Deze gegevens worden gevalideerd binnen de controller. Bij geldige input wordt een User-object aangemaakt en via de UserRepository of Mapper opgeslagen in de database. Na succesvolle registratie wordt de gebruiker doorgestuurd naar de inlogpagina.

Use case 2: Gebruiker vult de stemwijzer in.

Een gebruiker vult de stemwijzer in door bij elke stelling aan te geven in hoeverre hij het ermee eens of oneens is. Als de gebruiker is ingelogd, worden de resultaten opgeslagen zodat hij deze later kan bekijken. Na het invullen van alle stellingen krijgt de gebruiker te zien welke partijen het best passen bij zijn/haar antwoorden.

Technische realisatie:

De gebruiker doet een GET-request naar de backend om de stemwijzer pagina op te halen. De backend haalt via de repository alle actieve stellingen op uit de database, samen met de partij antwoorden, en rendert de pagina via de template engine.

Wanneer de gebruiker de stemwijzer invult en indient, wordt er een POST-request gedaan met zijn antwoorden. Deze antwoorden worden gevalideerd en verwerkt. De backend vergelijkt de gebruikers antwoorden met de partij antwoorden en berekent welke partij het best past bij de gebruikers antwoorden. Als de gebruiker is ingelogd, worden de antwoorden en het resultaat opgeslagen via de bijbehorende repositories. De gebruiker krijgt vervolgens een resultaatpagina te zien waarin de best matchende partijen worden weergegeven.

De berekening en opslag vinden plaats in de controller laag, mogelijk met behulp van aparte services of helpers.

De sessie- en access control-componenten zorgen ervoor dat alleen user en anonymous accounts deze functionaliteit kunnen gebruiken.

Use case 3: Partij vult antwoorden op een stelling in.

Een vertegenwoordiger van een politieke partij logt in met het partij account en vult voor zijn partij de antwoorden in op de beschikbare stellingen. Daarnaast kan hij bij elk antwoord een toelichting of uitleg toevoegen. Deze gegevens worden opgeslagen zodat de kieswijzer de best matchende partij kan berekenen op basis van de antwoorden.

Technische realisatie:

De partij gebruiker doet een GET-request om het overzicht van stellingen op te halen. De backend haalt deze gegevens via de repositories op en stuurt ze naar de template engine om een formulier te genereren waarin de partij haar antwoorden kan invullen.

Na het invullen van het formulier verstuurt de partij een POST-request met de antwoorden en eventuele toelichtingen. De backend valideert de invoer en slaat de nieuwe antwoorden op via de repository en/of de onderliggende data mappen.

De sessie- en access control-componenten zorgen ervoor dat alleen partij accounts deze pagina kunnen bereiken.

Use case 4: Beheerder maakt nieuwe stelling aan.

Een beheerder logt in en kan nieuwe stellingen toevoegen aan de kieswijzer. Deze stellingen worden gebruikt om gebruikers en partijen de mogelijkheid te geven hierop te reageren. De beheerder voert de tekst van de stelling in en slaat deze op.

Technische realisatie:

De beheerder doet een GET-request om de admin pagina op te halen. De backend haalt de gebruikers- en stellingsgegevens via de repositories of mappers op en stuurt ze naar de template engine om de admin pagina te laden waar gebruikers en stellingen kunnen worden gemanaged.

Wanneer de beheerder het inputveld "Nieuwe stelling" invult en vervolgens op de knop "Voeg toe" klikt, wordt er een nieuwe stelling aangemaakt en toegevoegd aan de database via de repository of mapper.

De sessie- en access control-componenten zorgen ervoor dat alleen adminaccounts deze pagina kunnen bereiken.

Use case 5: Beheerder voegt account toe.

De beheerder logt in en kan gebruikers, partij en admin accounts toevoegen op de admin pagina.

Technische realisatie:

De beheerder doet een GET-request om de admin pagina op te halen. De backend haalt de gebruikers- en stellingengegevens via de repositories of mappers op en stuurt ze naar de template engine om de admin pagina te laden waar gebruikers en stellingen kunnen worden gemanaged.

Wanneer de beheerder de gebruikersnaam, het wachtwoord en de rol invoert en vervolgens op de knop "Voeg toe" klikt, wordt er een nieuwe gebruiker met deze gegevens aangemaakt en toegevoegd aan de database via de repository of mapper.

De sessie- en access control-componenten zorgen ervoor dat alleen admin accounts deze pagina kunnen bereiken.

Use case 6: Beheerder verwijdert account.

De beheerder logt in en kan gebruikers, partij en admin accounts verwijderen op de admin pagina.

Technische realisatie:

De beheerder doet een GET-request om de admin pagina op te halen. De backend haalt de gebruikers- en stellingengegevens via de repositories of mappers op en stuurt ze naar de template engine om de admin pagina te laden waar gebruikers en stellingen kunnen worden gemanaged.

Op de beheerderspagina wordt een lijst met gebruikers weergegeven. Achter elke gebruiker staat een verwijder knop. Wanneer de beheerder op deze knop klikt, wordt de betreffende gebruiker via de mapper of repository uit de database verwijderd.

De sessie- en access control-componenten zorgen ervoor dat alleen admin accounts deze pagina kunnen bereiken.

Use case 6: Beheerder verwijdt stelling.

De beheerder logt in en kan stellingen verwijderen op de admin pagina.

Technische realisatie:

De beheerder doet een GET-request om de admin pagina op te halen. De backend haalt de gebruikers- en stellingsgegevens via de repositories of mappers op en stuurt ze naar de template engine om de admin pagina te laden waar gebruikers en stellingen kunnen worden gemanaged.

Op de beheerderspagina wordt een lijst met stellingen weergegeven. Achter elke stelling staat een verwijder knop. Wanneer de beheerder op deze knop klikt, wordt de betreffende stelling via de mapper of repository uit de database verwijderd.

De sessie- en access control-componenten zorgen ervoor dat alleen admin accounts deze pagina kunnen bereiken.