

Overzicht verschijningsvormen van de App

inzake

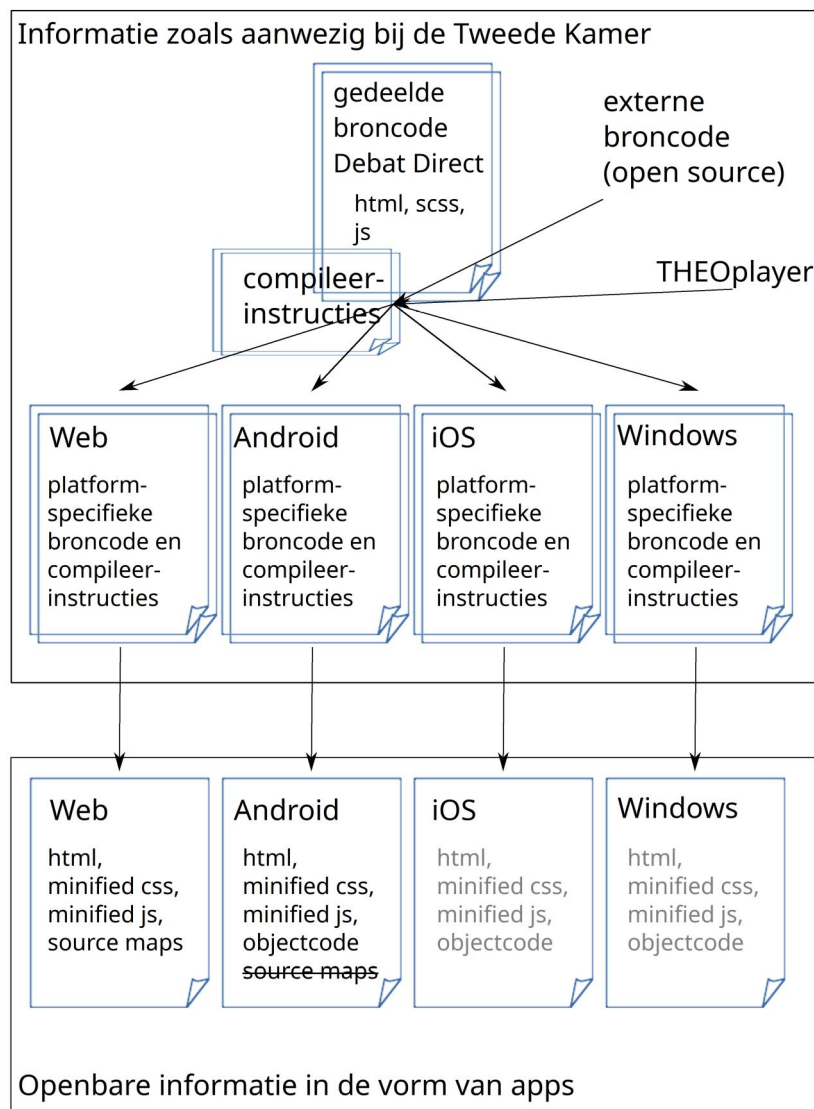
De heer J.M.P. van den Oever
wonende te Borne
eiser

tegen

Tweede Kamer der Staten-Generaal
zetelend te Den Haag
verweerder

De app Debat Direct van de Tweede Kamer wordt gepubliceerd in vier uitvoerbare vormen: webapp, Android app, iOS app en Windows app. Deze verschijningsvormen van de app delen veel broncode. De gedeelde broncode is JavaScript.

In figuur 1 wordt een overzicht gegeven van de verschijningsvormen van de App en de relatie tussen de verschijningsvormen. De informatie zoals aanwezig bij de Tweede Kamer is de herbruikbare verschijningsvorm van de App. De vier verschijningsvormen in het vak 'Openbare informatie in de vorm van apps' zijn via een automatisch proces gemaakt uit die verschijningsvorm. Die vorm bevat broncode en compileerinstructies. Een groot deel van de broncode en compileerinstructies geldt voor alle verschijningsvormen en er is een klein deel broncode en compileerinstructies specifiek voor de ver-



Figuur 1: Overzicht verschijningsvormen App

schillende platformen.

Het automatische proces waarmee uit de broncode uitvoerbare apps gemaakt worden, bevat verschillende stappen afhankelijk van het type broncode.

JavaScript wordt gebundeld en eventueel *geminified*. Bundeling betekent dat meerdere broncodebestanden samengevoegd worden tot een groot JavaScriptbestand. Bij *minificatie* wordt de code kleiner gemaakt door commentaar en witruimte weg te halen en functie- en variabelenamen om te zetten naar kortere systematische namen. In deze vorm wordt de software veel sneller uitgevoerd, maar is deze niet meer herbruikbaar. JavaScript die geminificeerd is, gaat vaak vergezeld van een zogenaamde *source map* die de link legt tussen de oorspronkelijke broncode en de verkleinde vorm. Door publicatie van een source map is software beter herbruikbaar. Het maken van source maps is onderdeel van de compilatiestap.

De broncode bevat ook SCSS bestanden. Dit is broncode met stijlinformatie. In de compilatiestap wordt SCSS omgezet naar CSS, een vorm van stijlinformatie voor webbrowsers. Ook in deze stap wordt een source map gemaakt.

De compilatiestap van de JavaScript en SCSS broncode is uitgevoerd met het pakket WebPack. Voor deze stap zijn minstens twee bestanden nodig: *package.json* en *webpack.config.js*. De informatie in deze bestanden bepaalt hoe de informatie die in de uitvoerbare verschijningsvormen er uit ziet. Dit is dus ook informatie die onderdeel is van de herbruikbare verschijningsvorm van de app.

Voor het Android platform wordt Java broncode ingezet. Deze wordt tijdens de compilatiestap omgezet naar objectcode. Deze objectcode kan vergezeld gaan van de oorspronkelijke broncode, maar dat is bij de Debat Direct app niet het geval.

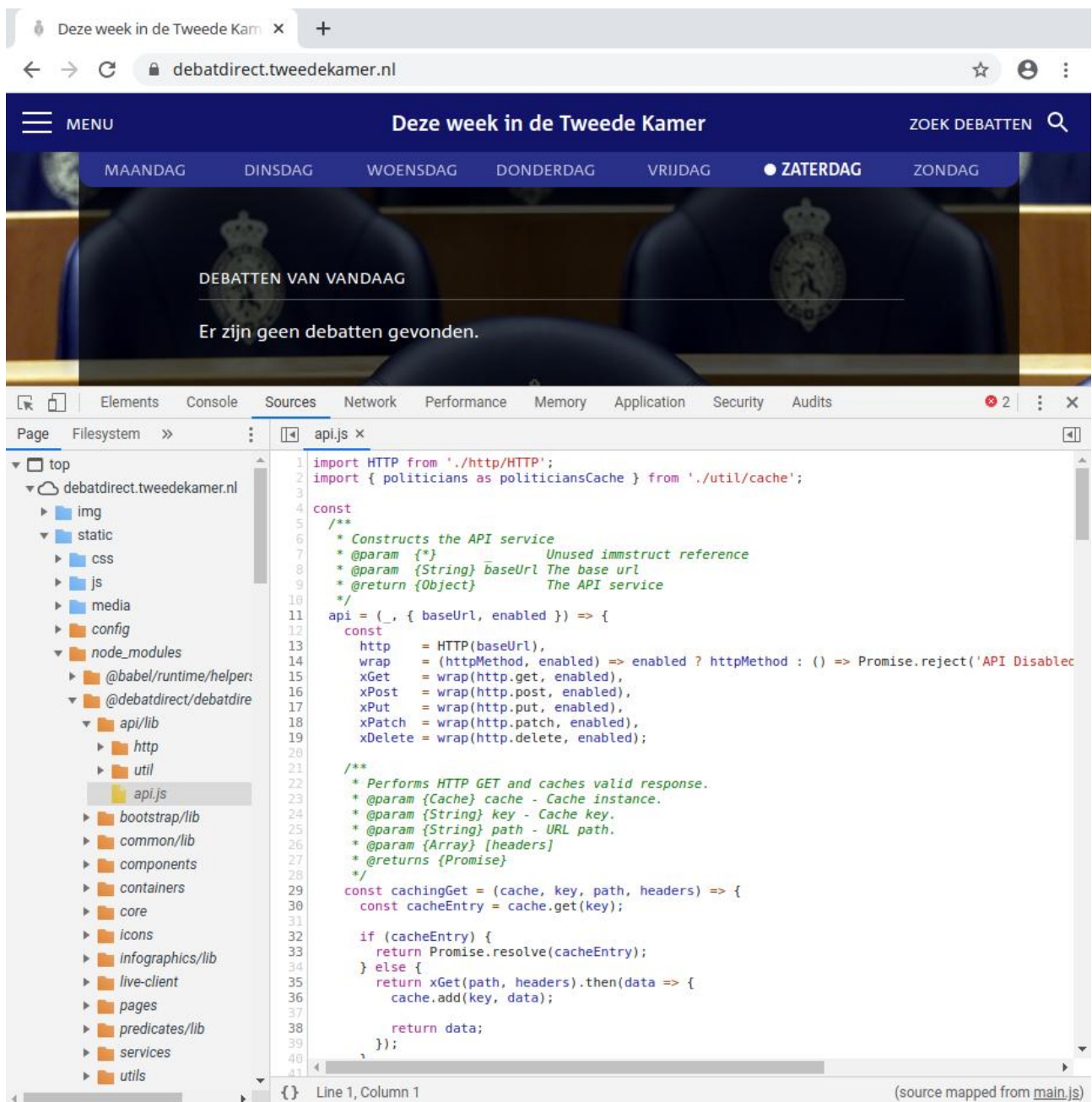
De tekst 'source maps' onder het kopje Android is doorgestreept omdat de actuele versie van de Android app geen source maps meer bevat.

De inhoud van de iOS en Windows apps hebben we niet onderzocht. We weten niet of deze verschijningsvormen van de app broncode bevatten en daarom is de inhoud van deze apps grijs weergegeven.

Publiek gemaakte broncode in Debat Direct web app

Via de Debat Direct web app is de gedeelde JavaScript broncode te bestuderen. Dit kan met gangbare webbrowsers zoals Chrome, Firefox en Safari. De procedure hiervoor is:

1. Browse naar <https://debatdirect.tweedekamer.nl>
2. Druk op F12 (of $\text{⌘}+\text{⌘}+\text{I}$ op OS X). Dit opent *Developer Mode*.
3. Klik op de tab met label 'Debugger' of 'Source'
4. Nu is in een tab met label 'Page', 'Pagina', 'Bronnen' of 'Sources' een boom met folders en be-

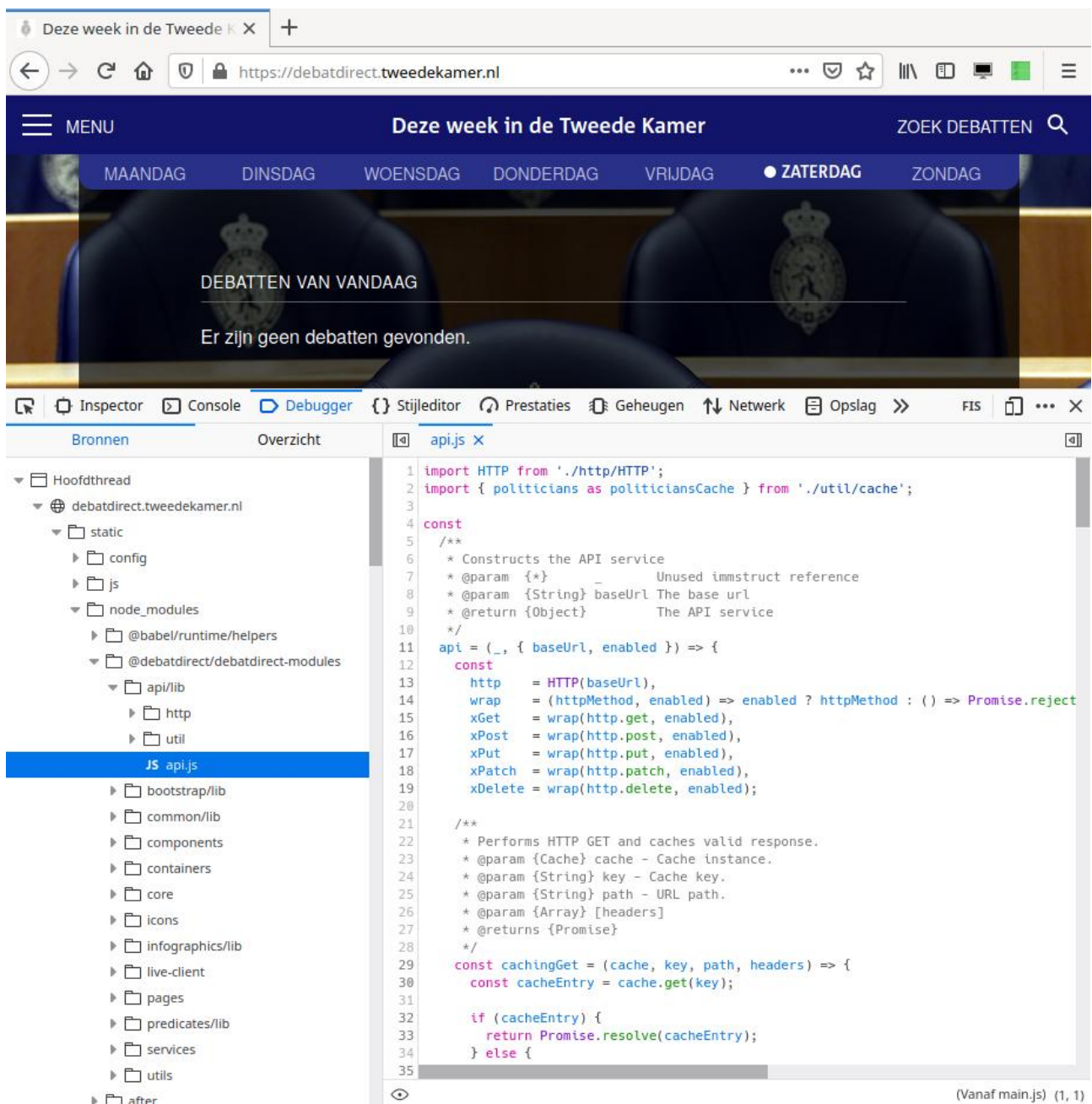


Figuur 2: Broncode van Debat Direct zichtbaar in de browser Chromium

standen zichtbaar. Deze boom bevat bijvoorbeeld een bestand met pad *debatdirect.tweedekamer.nl/static/node_modules/@debatdirect/debatdirect-modules/api/lib/api.js*.

5. Navigeer naar een bestand, bijvoorbeeld het hierboven genoemde *api.js*. Nu wordt de broncode in dit bestand getoond.
6. Sla het bronbestand op door rechts te klikken op de broncode en op de actie met het label 'Save as', 'Bestand downloaden' of woorden van gelijke strekking te klikken.

Het resultaat van het volgen van de stappen 1-5 in twee verschillende browsers is getoond in de figuren 1 en 2. Deze broncode bevat commentaren en variabelenamen.



Figuur 3: Broncode van Debat Direct zichtbaar in de browser Firefox

De broncode is gepubliceerd middels een *source map*. Een source map is een bestand dat een koppeling legt tussen objectcode of geminifide code enerzijds en broncode anderzijds. Source maps worden geladen als Developer Mode geactiveerd wordt. De Tweede Kamer publiceert drie source maps voor de web app vorm van Debat Direct.

De adressen van die source maps zijn <https://debatdirect.tweedekamer.nl/static/js/main.js.map>,
<https://debatdirect.tweedekamer.nl/static/js/worker.js.map> en
<https://debatdirect.tweedekamer.nl/static/css/main.css.map>.

Door publicatie van de source maps kan de gebruiker van de software de broncode bestuderen en onderzoeken of de software geen activiteiten onderneemt die de gebruiker ongewenst acht.

De broncode die door de Tweede Kamer via source maps is gepubliceerd bevat de procedures voor communicatie met de servers van de Tweede Kamer. Deze procedures zijn te besturen en aan te passen. Deze bestanden zijn vermoedelijk niet exact gelijk aan de oorspronkelijke informatie zoals die bij de Tweede Kamer aanwezig is. Vermoedelijk zijn tijdens het maken van de source map aan ieder bestand nog een paar regels informatie toegevoegd die beginnen met de tekst '// WEBPACK FOOTER //

De individuele bestanden met broncode uit de source maps kunnen als losse bestanden worden opgeslagen. Dit levert 872 bestanden met ongeveer 200.000 regels broncode inclusief commentaar op. Het grootste deel hiervan is broncode die door derden is ontwikkeld en als Open Source beschikbaar is gesteld. De module *@debatdirect* en de folder *static/js* bevatten broncode die in opdracht van of door de Tweede Kamer is ontwikkeld. Dit zijn 295 bestanden met ongeveer 35.000 regels broncode.

Niet alle objectcode in de web app van Debat Direct is als broncode in te zien. Voor de bestanden https://cast.cl-prod.livedebatapp.nl/cast/sdk/libs/sender/1.0/cast_framework.js en <https://cdn.debatdirect.tweedekamer.nl/player/THEOplayer.chromeless.js> is geen source map beschikbaar. Het eerste bestand bevat software van o.a. Google en Microsoft. Het twee bestand is closed source software van het bedrijf THEO Technologies NV.

Scenario's voor hergebruik

Hier beschrijven we een aantal algemene scenario's van hergebruik van software om inzichtelijke te maken waarom broncode en compileerinstructies essentieel zijn voor hergebruik. De broncode en compileerinstructies zijn de herbruikbare vorm van software.

Use case: onderdelen app hergebruiken voor andere organisaties

Het parlement is een van vele openbare besturen in Nederland. De app kan met aanpassingen ook voor andere organisaties nuttig zijn. De broncode van de Debat Direct app is opgedeeld in modules die hiervoor gebruikt kunnen worden. Zo is er een module 'Politicians' in het bestand Politicians.js die een scherm weergeeft met alle politici in de organisatie. De afhankelijkheden staan boven in dit bestand genoemd. Deze module zou in andere software hergebruikt kunnen worden. Zo wordt het maatschappelijke en economische voordeel waarmee de Who gemotiveerd is, bewaarheid.

De versie nummers van de externe afhankelijkheden zijn helaas momenteel niet openbaar beschikbaar. Voor goede werking van de software is belangrijk dat compatibele versies gebruikt worden. Deze informatie is opgeslagen in bestanden die de Tweede Kamer nog niet publiek maakt.

De variablenamen, functienamen en commentaren in de broncode zijn in het Engels. De broncode is dus ook in andere taalgebieden te gebruiken.

De geminificeerde versie van de software is niet zo gemakkelijk te hergebruiken. Daar is alle software gevat in een klein aantal grote onleesbare bestanden. De onderdelen specifiek voor de Tweede Kamer en de externe modules lopen daarin door elkaar. Functie- en variablenamen zijn vervangen door generieke namen met opeenvolgende nummers. In iedere nieuwe versie van de software kan dezelfde procedure een andere naam krijgen. Dus het is vrijwel onmogelijk inzicht te krijgen in de wijzigingen die hebben plaatsgevonden bij het maken van een nieuwe versie.

Use case: vertaling van de app maken

De app heeft momenteel uitsluitend een Nederlandstalige interface. De stukken tekst die in de broncode voorkomen kunnen vertaald worden. Daarvoor is context nodig. Die is in de broncode beschikbaar. In de geminificeerde versie is zijn teksten nog wel te herkennen, maar is de context weg. Het is dan niet mogelijk een accurate vertalen te maken.

Use case: app beschikbaar maken op een open platform

De app is beschikbaar voor de besturingssystemen iOS, Windows en Android via de commerciële buitenlandse bedrijven Apple, Microsoft en Google. Er zijn burgers die die platformen niet willen gebruiken en bijvoorbeeld Debian Linux, Ubuntu Linux of een Google-vrije versie van Android willen gebruiken. Dit zijn in absolute aantallen populaire platformen waarvoor de Tweede Kamer momenteel geen specifieke verschijningsvorm van de app maakt.

Deze platformen hebben een ethische code die vereist dat een app alleen opgenomen wordt als de herbruikbare vorm van de app, de broncode, bij de uitvoerbare versie gepubliceerd wordt. Het publiceren van een verschijningsvorm met enkel de geminificeerde versie van de app is daar niet acceptabel. Door enkel de gesloten platformen te ondersteunen bevoordeelt de Tweede Kamer platformen die de gebruiker benadelen.

Use case: lesstof

De broncode van de app Debat Direct is goed opgezet. Deze zou gebruikt kunnen worden als lesmateriaal voor informaticaopleidingen. Een studieopdracht zou kunnen zijn om de app aan te passen of te bestuderen hoe deze is opgezet. De geminificeerde van de app is ook wel te bestuderen, maar die vorm is alleen nuttig voor offensieve scenarios waar gekeken wordt waar de zwakheden zitten. Die zijn zonder broncode namelijk prima te vinden.

Communicatie tussen app en server

De app communiceert met de server van de Tweede Kamer, om precies te zijn de server met de naam *api.debatdirect.tweedekamer.nl*. Verder heeft beweerd beducht te zijn voor manipulatie van de informatiekanalen. Deze manipulatie kan ook zonder de beschikking over de broncode plaatsvinden.

De namen van de servers waarmee de app communiceert zijn te zien in de 'Network' of 'Netwerk' tab van de 'Developer Mode' van gangbare webbrowsers. Daar is precies te volgen hoe de app communiceert met de servers. Zo levert een HTTP GET request naar <https://api.debatdirect.tweedekamer.nl/actors> een lijst op van politici. Deze lijst heeft in het formaat JSON en ziet er zo uit:

```
{
  "date": "2020-02-09",
  "currentDate": "2020-02-09",
  "politicians": [
    {
      "id": "18202122-0fdc-46e3-9198-27b1898724f7",
      "date": "2020-02-09",
      "slug": "pien-horst",
      "name": "Pien ter Horst",
      "partyId": "2D26B86F-BF3A-488D-99FF-AFD47957B8C0",
      "firstName": "Pien",
      "lastName": "ter Horst",
      "title": "Tweede Kamerlid",
      "seat": 120,
      "profileUrl": "https://www.tweedekamer.nl/kamerleden_en_commissies/
alle_kamerleden/horst-p-ter-party-partij",
      "slogan": "Als een werk het waard is om te doen, dan is dat het waard om
goed te doen. Alaaf!"
    },
    ...
  ]
}
```

Vergelijkbare communicatie tussen app en server verloopt via adressen als:

- <https://api.debatdirect.tweedekamer.nl/values/preferences>
- <https://api.debatdirect.tweedekamer.nl/app>
- <https://api.debatdirect.tweedekamer.nl/actors>
- <https://api.debatdirect.tweedekamer.nl/agenda>
- <https://api.debatdirect.tweedekamer.nl/agenda/overview/stemmingen>
- <https://api.debatdirect.tweedekamer.nl/actors/2020-02-10>

Deze interface is niet publiek gedocumenteerd, maar wel gemakkelijk in te zien. Het is een simpele en op het ook veilige interface. Als de interface onveilig zou zijn, dan zou het achterhouden van de broncode geen beperking zijn bij het manipuleren van de interface.

Verweerder heeft blijkbaar geen probleem met het openbaar zijn van de broncode die de communicatie beschrijft. De source maps zijn namelijk gepubliceerd via historische versies van de Android app en de web app.

Verweerder heeft geen scenario's geschetst waarin het achterhouden van broncode het manipuleren van informatiekkanalen voorkomt.

Beschikbaarstelling van de herbruikbare vorm van software

Het beschikbaar stellen van broncode via source maps is gebruikelijk. De overheidssite <https://developer.overheid.nl>, een project van het Ministerie van Binnenlandse Zaken en Koninkrijksrelaties en VNG Realisatie, publiceert op dezelfde wijze broncode. Publicatie van broncode middels source maps is voor hergebruik niet de gemakkelijkste manier van publiceren. Meestal wordt broncode als een repository gepubliceerd of gebundeld in een archiefbestand.

developer.overheid.nl publiceert de broncode van client én server via een publiek git repository: <https://gitlab.com/commonground/developer.overheid.nl/>. Dit is de meest gebruikelijke en gemakkelijke manier om de herbruikbare vorm van software te publiceren. Een steekproef wijst uit dat de bestanden die developer.overheid.nl middels source map publiceert, gelijk zijn aan de bestanden in de corresponderende git repository.

Om broncode te kunnen hergebruiken zijn ook de computerleesbare instructies voor het compileren van de broncode tot door de computer (of smartphone) uitvoerbare software nodig. Deze zijn geen onderdeel van de source maps en niet door de Tweede Kamer gepubliceerd. Normaal gesproken wordt broncode wel vergezeld van instructies voor het compileren en kunnen worden beschouwd als onderdeel van de herbruikbare vorm van software.