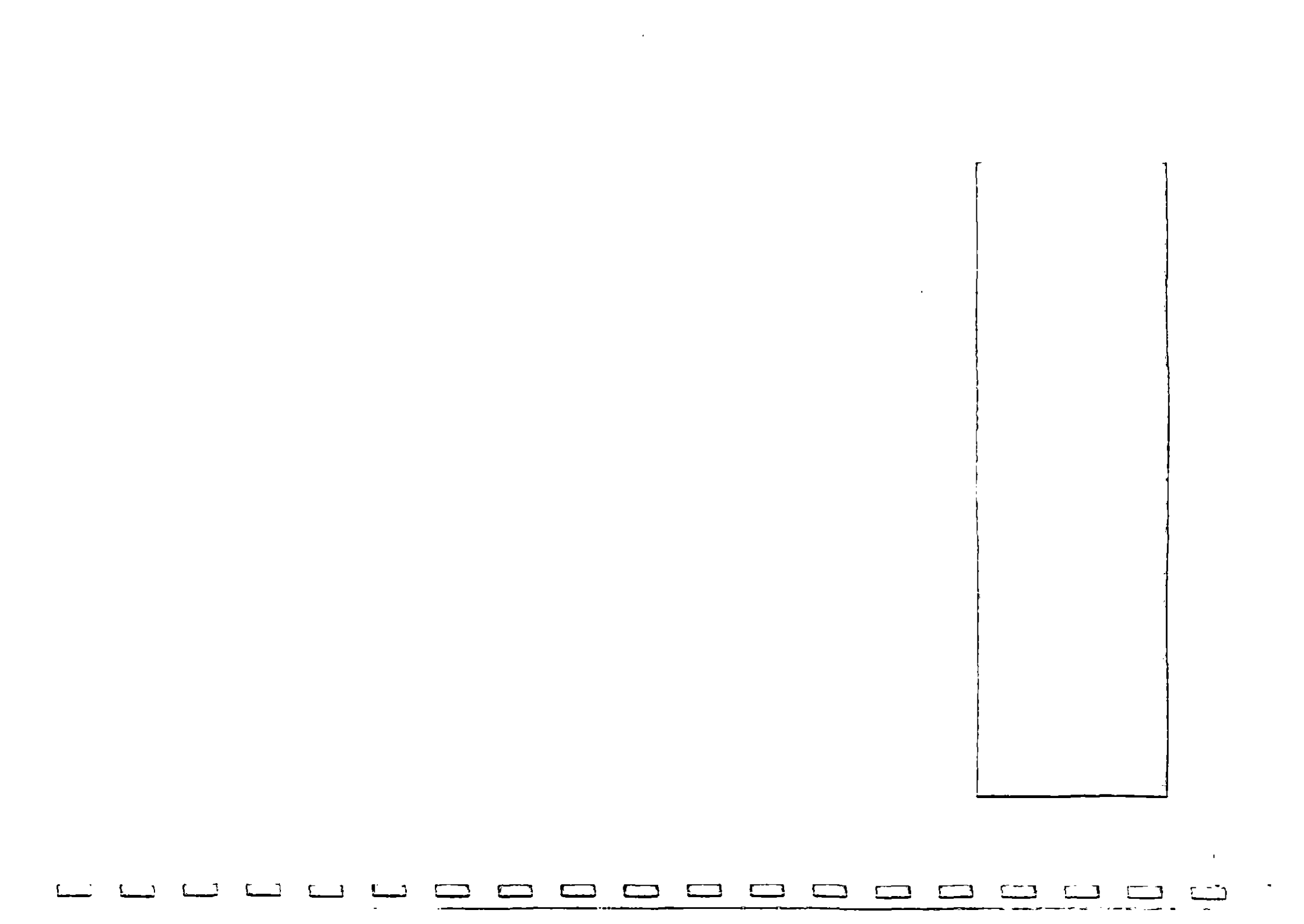


Software Improvement Group



Toetsing eisen OSV 4 en 5 voor alle soorten verkiezingen

Rapport t.b.v. de Kiesraad

10 februari 2011

Dr. Sieuwert van Otterloo

+31 20 314 0950

s.vanotterloo@sig.eu

Dr. Bas Cornelissen

+31 20 314 0950

b.cornelissen@sig.eu

Software Improvement Group

P.O. Box 94914
1090 GX Amsterdam
The Netherlands
t +31 20 314 0950
f +31 20 314 0955
info@sig.eu
www.sig.eu

Het onderzoek dat in dit rapport is beschreven is uitgevoerd in opdracht van Mw. Mr. R. Hoorweg, plaatsvervangend secretaris-directeur van de Kiesraad.

Het rapport is geschreven door Dr. S. van Otterloo en Dr. B. Cornelissen van de Software Improvement Group.

© 2011, Software Improvement Group
Postbus 94914
1090 GX Amsterdam
The Netherlands

Managementsamenvatting

De Kiesraad heeft programmatuur laten ontwikkelen ter ondersteuning van het verkiezingsproces en stelt deze programmatuur beschikbaar aan de stembureaus voor gebruik bij verkiezingen. De eerste versie van deze software is gebruikt voor de Europese verkiezingen op 4 juni 2009. In december 2010 is versie 2.6 ontwikkeld voor gebruik in de proviciale statenverkiezingen op 2 maart 2011 en de Eerste Kamerverkiezingen op 23 mei 2011.
De staatssecretaris van Binnenlandse Zaken en Koninkrijksrelaties heeft eisen gesteld waaraan deze programmatuur moet voldoen. Eén van deze eisen is dat er door een onafhankelijke instantie een toets gedaan wordt. Het gaat hierbij om toetsing van deel 4 en 5 van de programmatuur *Ondersteunende Software Verkiezingen* (OSV), omdat deze delen gebruikt zullen worden voor de verwerking van verkiezingsresultaten. Deze toetsing wordt periodiek uitgevoerd, en dit rapport is tot stand gekomen in het kader van deze periodieke toetsing, en betreft versie 2.6 van OSV.

Uit de toetsing aan gestelde eisen is naar voren gekomen dat de programmatuur OSV 4 en OSV 5 op de volgende kanttekeningen na voldoet aan gestelde eisen:

- Er wordt voldaan aan eis 2 (documentatie) zodra de specificatie van OSV 4 en 5 wordt bijgewerkt. De leverancier heeft toegezegd dit voor eind maart 2011 gereed te hebben. Het gaat hierbij onder andere om toegevoegde functionaliteit voor Eerste Kamerverkiezingen.
- Er is niet volledig voldaan aan eis 3d (foutief gebruik), omdat met het oog op gebruiksgemak enkele beveiligingen zijn verwijderd.
- Er is slechts gedeeltelijk aan eis 4c (open source) voldaan. Als positief punt valt op te merken dat de broncode beschikbaar is ter inzage. Om vast te stellen dat de programmatuur conform eis 4c 'open source' ontwikkeld is, moet de broncode onder een goedgekeurde open source-licentie worden gepubliceerd. Net als bij vorige versies is dit niet gebeurd voor deze versie.
- Gerelateerd aan eis 7 (authenticiteit programmatuur) geldt dat zolang de programmatuur in een afgeschermde omgeving gebruikt wordt, er aan eis 7 voldaan is.
- In eis 9 (formele methodes) wordt gesproken over wiskundig aangetoonde correctheid. Dit is een zeer zware eis die alleen voor zeer kritieke en lastig wijzigbare programmatuur gebruikelijk is. De geleverde wiskundige definitie is, net als in de eerdere toetsing, niet voldoende volledig om deze eis als voldaan te beschouwen. In een vorig rapport concludeerde de SIG dat de programmatuur wel uitvoerig is gedocumenteerd en getest. Een aanvullende analyse toont aan dat dit nog steeds het geval is.

Tevens is vastgesteld dat de programmatuur van gemiddelde technische kwaliteit is en daardoor voldoende onderhoudbaar en toekomstvast is. De technische kwaliteit is ten opzichte van de vorige toetsing niet gewijzigd.

© 2011 Software Improvement Group

Inhoudsopgave

1 INLEIDING 6
1.1 Context 6
1.2 Aanleiding 6
1.3 Scope 6
1.4 Onderzoeksvragen 6
1.5 Structuur van dit rapport 7

2 ONDERZOEKSPROCES 8
2.1 Uitgangspunten 8
2.2 Bronnen 8
2.3 Betrokken personen 9

3 ANTWOORDEN TOETSING AAN EISEN 11
3.1 Eis 1: functionaliteit 11
3.2 Eis 2: documentatie 11
3.3 Eis 3: ontwerp 12
3.4 Eis 4: open standaarden en open source 13
3.5 Eis 5: verschillende besturingssystemen 15
3.6 Eis 6: diakritische tekens 16
3.7 Eis 7: authenticiteit programmatuur 16
3.8 Eis 8: authenticiteit gegevens 17
3.9 Eis 9: formele methodes 17
3.10 Eis 10: onafhankelijke toetsing 19
3.11 Eis 11: elektronisch stemmen 19

4 TECHNISCHE KWALITEIT VAN OSV 4 EN 5 20
4.1 Conclusie voor de technische kwaliteit 20
4.2 Onderbouwing 20

5 CONCLUSIES EN AANBEVELINGEN 21
5.1 Conclusies 21
5.2 Aanbevelingen 21

A. AANVULLENDE INFORMATIE VOOR TOETSING EISEN 22
A.1 Eis 1: functionaliteit 22
A.2 Eis 2: documentatie 22
A.3 Eis 3: ontwerp 26
A.4 Eis 4: open standaarden en open source 28

© 2011 Software Improvement Group

A.5 Eis 5: verschillende besturingssystemen 29
A.6 Eis 6: diakritische tekens 29
A.7 Eis 7: authenticiteit programmatuur 29
A.8 Eis 8: authenticiteit gegevens 30
A.9 Eis 9: formele methodes 31

B. AANVULLENDE INFORMATIE VOOR TECHNISCHE KWALITEIT 35
B.1 Model voor technische kwaliteit 35
B.2 Toelichting bij beoordelingen 36
B.3 Scope van de metingen 36
B.4 Volume 36
B.5 Duplicatie 37
B.6 Unit lengte 38
B.7 Unit complexiteit 39
B.8 Unit interfacing 40
B.9 Module coupling 41
B.10 Foutafhandeling 42

C. DISCLAIMER 44

© 2011 Software Improvement Group

1 Inleiding

1.1 Context

De Kiesraad heeft programmatuur laten ontwikkelen ter ondersteuning van het verkiezingsproces en wil deze programmatuur beschikbaar stellen aan de stembureaus voor gebruik bij verkiezingen. Het gaat om een verzameling programma's genaamd 'Ondersteunende Software Verkiezingen (OSV)'. Deze programmatuur stelt stembureaus in staat om getelde stemmen in te voeren, op te slaan, samen te voegen en tenslotte de gekozen kandidaten te bepalen. In december 2010 is versie 2.6 ontwikkeld voor gebruik in de provinciale statenverkiezingen op 2 maart 2011 en de Eerste Kamerverkiezingen op 23 mei 2011.

Er zijn zes OSV-programma's, OSV 0 tot en met OSV 5. De programma's OSV 4 en OSV 5 zijn de programma's uit OSV die door de verschillende stembureaus gebruikt worden na de verkiezingen voor verwerking van uitslagen. De overige programma's zijn voor het aanmaken van de benodigde informatie (verkiezingsdefinities en kandidatenlijsten) ter voorbereiding van verkiezingen.

1.2 Aanleiding

Door middel van een brief aan de Tweede Kamer op 9 april 2008 heeft de staatssecretaris van Binnenlandse Zaken en Koninkrijksrelaties eisen gesteld waaraan OSV 4 en 5 moeten voldoen, voordat zij gebruikt kunnen worden in het verkiezingsproces. Eén van deze eisen is dat er door een onafhankelijke instantie een toets gedaan wordt in opdracht van de Kiesraad of de programmatuur aan deze eisen voldoet. De Software Improvement Group (SIG) voert deze toetsing periodiek uit. Daarnaast heeft de Kiesraad er belang bij om de technische kwaliteit en onderhoudbaarheid van de programmatuur vast te laten stellen. Dit onderzoek heeft als doel om deze toetsing uit te voeren en secundair de onderhoudbaarheid vast te stellen.

Een eerdere toetsing is uitgevoerd op de OSV programma's in 2009; Hiervan zijn drie rapportdelen verschenen, waarvan het laatste is gepubliceerd op 5 oktober 2009. Dit rapport onderscheidt zich van de eerdere rapporten doordat een nieuwe versie is getoetst. De toetsing is gedaan tegen dezelfde eisen, de aanleiding voor dit rapport is de wens tot periodieke toetsing.

1.3 Scope

Het onderzoek heeft betrekking op programma 4 en programma 5 van de OSV programmatuur, in het kader van de Europese verkiezingen, Eerste Kamer, Tweede Kamer, gemeenteraden, deelraden, provinciale staten en referenda. De getoetste versie is 2.6.

1.4 Onderzoeksvragen

De onderzoeksvraag voor het uitgevoerde onderzoek is:

© 2011 Software Improvement Group

Toets programma's 4 en 5 van OSV aan de 11 gestelde eisen (omschreven in "Eisen voor programmatuur die gebruikt wordt ..." - brief aan Tweede Kamer van 9 april 2008).

In het kader van dit onderzoek is tevens de technische kwaliteit van OSV 4 en 5 getoetst.

1.5 Structuur van dit rapport

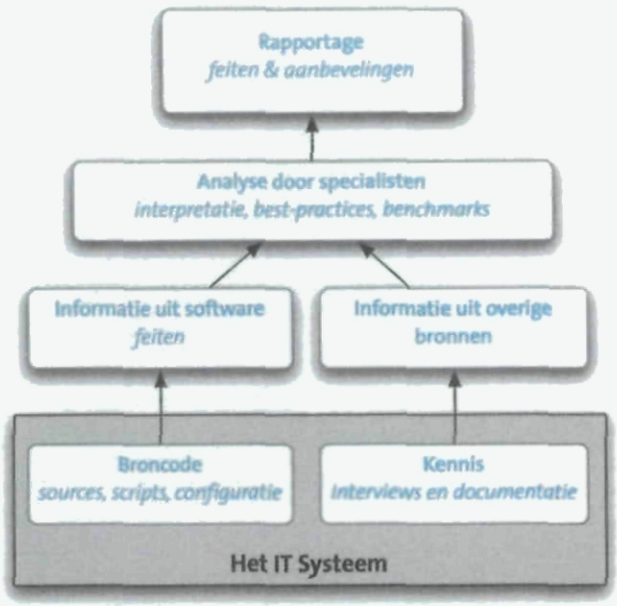
De structuur van dit rapport is als volgt. Hoofdstuk 2 beschrijft het onderzoeksproces. Hoofdstuk 3 bevat het resultaat van de toetsing van de 11 gestelde eisen. Hoofdstuk 4 beschrijft het oordeel over de technische kwaliteit van OSV 4 en 5. Hoofdstuk 5 besluit met conclusies en aanbevelingen. Appendices A en B geven aanvullende informatie over respectievelijk de toetsing van de eisen en de technische kwaliteit.

© 2011 Software Improvement Group

2 Onderzoeksproces

2.1 Uitgangspunten

SIG is gespecialiseerd in het uitvoeren van onderzoek naar kwaliteit en onderhoudbaarheid van programmatuur op basis van broncode-onderzoek. De basis van het onderzoek wordt daarom gevormd door de feiten verzameld door onderzoek van de aangeleverde broncode. Daarnaast is er aanvullende documentatie als bron gebruikt en zijn gesprekken gevoerd met medewerkers van de leverancier IVU ter verduidelijking.
Op basis van de hieruit vastgestelde feiten is er een interpretatie gedaan door SIG medewerkers om te komen tot antwoorden op de onderzoeksvragen. Deze antwoorden zijn gemotiveerd vanuit de vastgestelde feiten. Deze werkwijze is schematisch weergegeven in Figuur 1.



Figuur 1: Opzet van een onderzoek naar software op basis van broncode. De SIG hanteert een werkwijze waarin eindconclusies gebaseerd zijn op vastgestelde feiten.

2.2 Bronnen

De volgende broncode is door de SIG ontvangen en gebruikt als basis voor dit onderzoek:

- Broncode OSV 4 en 5, versie 2.6, ontvangen op 13 december 2010
- Broncode OSV 4 en 5, versie van 25 september 2009. Deze versie is gebruikt voor verschilanalyse in het kader van eis 9.

Naast deze broncode heeft de SIG de volgende ondersteunende documentatie ontvangen ten behoeve van dit onderzoek:

- Gedetailleerde specificatie OSV 1.3.4, ontvangen op 16 december 2010.
- Handleiding programma P4 v2.6, ontvangen op 13 december 2010.
- Handleiding programma P5 v2.6, ontvangen op 13 december 2010.
- 'Determination of the election result v5.14,' ontvangen op 13 december 2010.
- 'List of plausibility checks,' bevestigd op 4 januari 2011.
- Configuratie-bestanden v2.6, ontvangen op 13 december 2010.

© 2011 Software Improvement Group

- Installatiebestanden P4 en P5, ontvangen op 13 december 2010.
- 'On the use of hash codes within OSV,' ontvangen op 30 december 2010.
- Document 'Feedback-eis2', ontvangen op 9 februari 2011

Tevens heeft de SIG van de Kiesraad ontvangen:

- Overeenkomst inzake ondersteunende software verkiezingen – overeenkomst tussen de Kiesraad en IVU.
- Verklaring van Destatis over gebruik van de WAS-programmatuur – Brief van 26 maart 2009.
- 'Eisen voor programmatuur die gebruikt wordt bij de berekening van de uitslag van verkiezingen die vallen onder de werking van de Kieswet" - brief van de staatssecretaris van Binnenlandse Zaken en Koninkrijksrelaties aan de Tweede Kamer op 9 april 2008 met daarin de aan de programmatuur gestelde eisen.

De volgende algemeen beschikbare documenten zijn tevens geraadpleegd tijdens dit onderzoek:

- Actieplan *'Nederland Open in Verbinding'* – gepubliceerd door het Ministerie van Economische Zaken.
- GBA logisch ontwerp, versies 3.6, 3.7 en 3.8.
- *'Legal, operational and technical standards,'* aanbeveling door de Raad van Europa voor elektronisch stemmen.
- *'Open source initiative'* - http://www.opensource.org/ - geraadpleegd op 7 januari 2011.
- *'JBoss Enterprise Application platform certified and compatible configurations'* - http://www.jboss.com/products/platforms/application/testedconfigurations/ - geraadpleegd op 7 januari 2011.
- Kieswet, geraadpleegd via http://wetten.overheid.nl, versie met geldigheidsdatum van 7 januari 2011.

2.3 Betrokken personen

Bij dit onderzoek is er contact geweest, direct of per telefoon of e-mail, met de volgende personen.

- Dhr. S. Eulitz (IVU),
- Dipl.-Inform. T. Ducke (IVU),
- Dipl.-Math. J. Nottebaum (IVU),
- Mw. Mr. R. Hoorweg (Kiesraad),
- Mr. M. Bakker (Kiesraad),
- Mw. Drs. P. J. Young (Kiesraad),
- Mr. Ing. J.-J. Vos (Kiesraad).

Tijdens het onderzoek hebben de volgende meetings plaatsgevonden als vaste contactmomenten als onderdeel van de onderzoeksaanpak.

© 2011 Software Improvement Group

Datum	Meeting	Aanwezig
30 december 2010	Validatie van door SIG vastgestelde technische feiten door leverancier IVU (telefonisch)	S. Eulitz, T. Ducke, J. Nottebaum, S. van Otterloo, B. Cornelissen, J. Heijmans
5 januari 2011	Delen van definitieve bevindingen met IVU (per e-mail)	S. Eulitz , B. Cornelissen
17 januari 2011	Presentatie van het concept-rapport	M. Bakker, P. Young, J.-J. Vos, S. van Otterloo, B. Cornelissen
27 januari 2011	Nadere bespreking concept-rapport (telefonisch)	R. Hoorweg, S. van Otterloo, B. Cornelissen

© 2011 Software Improvement Group

3 Antwoorden toetsing aan eisen

In deze sectie is per eis weergegeven of er aan voldaan is en wat de motivatie is voor dit oordeel.

3.1 Eis 1: functionaliteit

3.1.1 Eis en conclusie

Gestelde eis:

De programmatuur bevat de functionaliteiten die (conform wet- en regelgeving) nodig zijn voor de berekening van de uitslag (inclusief tussenstappen en tussenresultaten) door het centrale stembureau en de uitvoer daarvan.

Conclusie:

- Ja, er is voldaan aan deze eis.

3.1.2 Motivatie

- De leverancier heeft per sectie uitgelegd welke functionaliteit bevat is in elke module. De SIG heeft de broncode van iedere module geïnspecteerd om te zien of dit overeenkomt met de uitleg, en heeft gecontroleerd of het geheel van de modules voldoende is voor berekening van de uitslag en uitvoer daarvan. Appendix A.1 vermeldt de gegeven uitleg van de functionaliteit.
- SIG heeft de werking van de programmatuur getest.
- Van 20 januari 2011 tot 5 februari 2011 is tevens een integrale gebruikerstest uitgevoerd in opdracht van de Kiesraad. Hierin is getoetst dat de aanwezige functionaliteit ook voor gebruikers bruikbaar is.

3.2 Eis 2: documentatie

3.2.1 Eis en conclusie

Gestelde eis:

De functionaliteit van de programmatuur is beschreven en vastgelegd in documenten (functioneel ontwerp, technisch ontwerp, etc.). Deze documenten zijn openbaar.

Conclusie:

- Ja, er is voldaan aan deze eis zodra de leverancier de gedetailleerde specificatie heeft bijwerkt. De leverancier heeft toegezegd dit voor eind maart 2011 te doen.

3.2.2 Motivatie

- De functionaliteit van de programmatuur wordt beschreven in de gedetailleerde specificatie *Specificatie_OSV_1.3.4.doc*, versie 1.3.4. Deze is gepubliceerd op de website van de Kiesraad (gecontroleerd op 7 januari 2011).
- De SIG heeft een review gedaan op de gedetailleerde specificatie en per sectie vastgesteld dat de beschreven functionaliteit overeenkomt met de door de SIG in de programmatuur aangetroffen functionaliteit.

© 2011 Software Improvement Group

- De programmatuur bevat meer functionaliteit dan wordt beschreven in de genoemde versie van de specificatie. Deze functionaliteit heeft betrekking op ondersteuning voor verkiezingen voor de Eerste Kamer. Leverancier IVU heeft toegezegd de specificatie vóór de aankomende Eerste Kamerverkiezingen te zullen bijwerken. Appendix A.2.4 bevat een lijst met functionele wijzigingen die minimaal aan de specificatie moeten worden toegevoegd.

3.3 Eis 3: ontwerp

3.3.1 Eis en conclusie

Gestelde eis:

Het ontwerp van de programmatuur voldoet aan geaccepteerde kwaliteitseisen c.q. best practices voor de ontwikkeling van programmatuur: Daartoe:

a. Is de programmatuur gestructureerd opgebouwd, zodanig dat modulaire aanpassingen mogelijk zijn.
b. Zijn kritische functies in de programmatuur gescheiden.
c. Zijn gegevens die aan verandering onderhevig zijn (configuratieparameters) zonder aanpassingen van programmatuur te wijzigen.
d. Wordt toevallig of opzettelijk foutief gebruik van de programmatuur, voor zover als redelijkerwijs technisch mogelijk is, door het ontwerp voorkomen.

Conclusie:

- Nee, er is niet volledig aan deze eis voldaan.
  - Ja voor 3a
  - Ja voor 3b
  - Ja voor 3c
  - Er is niet volledig aan eis 3d voldaan omdat een controle of bestanden zijn gewijzigd in deze versie in minder situaties wordt vereist.
  - In het algemeen is wel voldaan aan geaccepteerde kwaliteitseisen c.q. best practices voor ontwikkeling van programmatuur.

3.3.2 Motivatie 3a en 3b

- Er is een duidelijke module-structuur, die zorgt voor een scheiding van bijvoorbeeld berekening uitslag, dataopslag en in- en uitvoer. Deze modulestructuur is weergegeven in Appendix A.1. Hiermee wordt voldaan aan 3a en 3b.
- SIG beveelt wel aan om een aantal kleine verbeteringen uit te voeren die de onderhoudbaarheid op lange termijn verbeteren. Het gaat hierbij om een aantal punten in de broncode van het systeem, zogeheten 'wederzijdse afhankelijkheden' tussen modules onderling in de broncode (zie Appendix A.3.2 voor details). Dit verbetert de aanpasbaarheid van deze modules in de toekomst. Leverancier IVU heeft toegezegd om hier in de volgende versie aan te werken.

3.3.3 Motivatie 3c

- Door middel van een 'election definition file' kan het programma zonder aanpassingen voor een volgende verkiezing gebruikt worden. Door herinstallatie

© 2011 Software Improvement Group

kan het programma in een andere rol of voor een andere regio gebruikt worden zonder aanpassingen aan de code. Hiermee is voldaan aan eis 3c.
- In de meest recente versie is ondersteuning voor het aanpassen van statische teksten toegevoegd. Hierdoor zijn ook deze teksten aanpasbaar zonder dat codewijzigingen zijn benodigd.

3.3.4 Motivatie 3d

- In eerdere versies van OSV was vereist dat een gebruiker bij het inlezen van verkiezingsgegevens gedwongen was om de correctheid van het digitale bestand te controleren met de papieren versie. De methode is beschreven onder eis 8. De controle kon niet worden overgeslagen omdat de gebruiker een code vanaf de papieren versie moest invoeren. In de nieuwe versie is het wel mogelijk om in bepaalde gevallen controles over te slaan. Deze gevallen hebben betrekking op situaties waarin het in te lezen bestand is aangemaakt door één en dezelfde gebruiker, of waarin een zeer korte tijd is verstreken tussen het aanmaken en het lezen. Appendix A.3.4 geeft een overzicht van de verschillende stappen en bijbehorende controles. Hierdoor wordt niet meer voldaan aan eis 3d.
- De programmatuur zal gebruikt worden op een afgezonderde netwerkomgeving die geen verbinding met de buitenwereld heeft. Hiermee wordt misbruik van buitenaf uitgesloten.
- Programma P4 bevat toetsen die onopzettelijke foutieve invoer tegengaan. Deze toetsen zijn weergegeven in Appendix A.3.4. Opzettelijk foutieve invoer door een stembureaumedewerker is ook zonder programmatuur mogelijk en kan redelijkerwijs technisch niet voorkomen worden.
- Programma P5 bevat geen invoerschermen die controles behoeven die onopzettelijk foutief gebruik tegengaan. Opzettelijk foutieve invoer door een stembureaumedewerker is ook zonder programmatuur mogelijk en kan redelijkerwijs technisch niet voorkomen worden.

3.3.5 Motivatie eis 3 als geheel

- De hoofdtekst van eis 3 gaat verder dan de genoemde deel-eisen, omdat er ook in het algemeen gesproken wordt over geaccepteerde eisen. De SIG doet daarom een toetsing aan door de SIG gehanteerde kwaliteitseisen voor technische kwaliteit van programmatuur. De resultaten van deze toetsing staan in hoofdstuk 4.
- Het resultaat van deze toetsing is dat de programmatuur voldoende scoort op alle kwaliteitseisen. Hiermee wordt voldaan aan de best practices en aan eis 3 als geheel.
- SIG doet wel de aanbeveling om de focus op technische kwaliteit te versterken in de verdere ontwikkeling van OSV. Dit voorkomt dat onderhoud in de komende jaren onnodig duurder wordt. Deze aanbeveling wordt beschreven in hoofdstuk 5.

3.4 Eis 4: open standaarden en open source

3.4.1 Eis en conclusie

Gestelde eis:

© 2011 Software Improvement Group

Conform het actieplan Nederland open in verbinding van het kabinet geldt voor de programmatuur:

a. *Dat gebruik wordt gemaakt van open standaarden. Voor verkiezingsgegevens (waaronder kandidatenlijsten en zetelverdeling) wordt de open standaard EML gebruikt.*
b. *Dat deze is geschreven in een gangbare programmeertaal, waarvoor een door een actieve gemeenschap onderhouden open source compiler en/of interpreter beschikbaar is.*
c. *Dat deze als open source ontwikkeld is. De broncode van de programmatuur is openbaar. Indien de programmatuur voor de centrale stembureaus wordt ontwikkeld dan dient het intellectueel eigendom van de broncode van de programmatuur te berusten bij een van de centrale stembureaus.*

Conclusie:

- Nee, er is niet volledig aan deze eis voldaan.
  - Ja voor 4a
  - Ja voor 4b
  - Er is niet volledig aan eis 4c voldaan omdat niet vastgesteld is of de programmatuur open source ontwikkeld is. De broncode is openbaar gemaakt. De programmatuur is oorspronkelijk niet voor de Kiesraad ontwikkeld, waardoor het derde deel van eis 4c niet van toepassing is.

3.4.2 Motivatie 4a

- De programmatuur maakt gebruik van de open standaard EML en de open standaard PDF.
- Naast deze open standaarden maakt de programmatuur gebruik van de niet-open standaard RTF. Deze standaard wordt niet onafhankelijk beheerd en is daardoor niet open, maar is wel een door veel partijen gebruikte standaard voor bestandsuitwisseling. Omdat de in RTF aangeboden informatie ook in PDF beschikbaar is, gaat dit niet in tegen de eis.
- Nieuw in versie 2.6 is het gebruik van de CSV standaard (comma separated values). Dit is een informele 'standaard', geen echte standaard en daarom geen 'open standaard'. Omdat CSV in de praktijk dezelfde toegankelijkheid biedt als een open standaard, is beoordeeld dat nog steeds aan de eis is voldaan.

3.4.3 Motivatie 4b

- Het programma is hoofdzakelijk geschreven in de programmeertaal Java. Voor deze taal is een open source compiler beschikbaar, namelijk Eclipse.
- De overige talen zijn Javascript, JSP en XSLT. Ook hiervoor is aan de eis voldaan.
- Voor Javascript is door ECMA een standaard gedefinieerd, en is Firefox beschikbaar als interpreter.
- JSP is een open standaard. Het JBoss-platform bevat een open source interpreter voor deze taal.
- XSLT is een open standaard, waarvoor Xalan een open source interpreter is.
- Voor de database-opslag wordt Derby en MySQL gebruikt. Dit zijn beide open source oplossingen.

© 2011 Software Improvement Group

3.4.4 Motivatie 4c

- Het actieplan Nederland Open In Verbinding stelt op pagina 28: *"Open source software is software die een door het open source initiative goedgekeurde licentie heeft en daarmee voldoet aan twee kenmerken: de broncode is vrij beschikbaar; in het licentiemodel is het intellectueel eigendom van de software en de bijbehorende broncode dusdanig geregeld dat de licentienemer de broncode mag inzien, gebruiken, verbeteren, aanvullen en distribueren."*
- De broncode is niet gepubliceerd onder een licentie die voldoet aan de geciteerde definitie van open source.
- De kern van de programma's OSV 4 en OSV 5 is gebaseerd op een eerder voor een derde partij ontwikkeld programma, namelijk een programma ontwikkeld voor het Duitse overheidsorgaan Destatis. Destatis heeft het intellectueel eigendomsrecht op deze kern.
- Er zijn wel afspraken gemaakt met Destatis die de Kiesraad in staat stellen de programmatuur ter inzage te publiceren, te gebruiken en aan te passen.
- De Kiesraad heeft de broncode van de programmatuur nog niet gepubliceerd. Om aan het tweede deel van eis 4c te voldoen moet deze beschikbaar worden gesteld en ook beschikbaar blijven. Op het moment van schrijven is de broncode van de vorige versie niet meer opvraagbaar op de website van de Kiesraad (gecontroleerd op 10 januari 2011).
- De kern van OSV 4 en 5 is niet specifiek voor de Kiesraad ontwikkeld, en dus is het derde deel van eis 4c niet van toepassing.

3.5 Eis 5: verschillende besturingssystemen

3.5.1 Eis en conclusie

Gestelde eis:

De programmatuur is beschikbaar op verschillende systeemarchitecturen en verschillende besturingssystemen, waaronder in ieder geval gangbare open source besturingssystemen.

Conclusie:

- Ja, er is aan deze eis voldaan.

3.5.2 Motivatie

- Het programma is gebaseerd op het JBoss platform. JBoss is zelf gebaseerd op het Java-platform.
- De leverancier van JBoss geeft aan dat JBoss geschikt is voor alle besturingssystemen die een juiste versie van het Java-platform bieden en een standaard database-omgeving. Hieraan is voldaan voor onder andere Linux, Windows en Mac OS X.
    - Voor zowel Linux, Windows en Mac OS X is er een juiste versie van het Java-platform.
    - Voor zowel Linux, Windows en Mac OS X is er een geschikte database.
- Als additionele zekerheid biedt de leverancier van JBoss ondersteuning van een groot aantal gecertificeerde configuraties die gebaseerd zijn op het open source besturingssysteem Linux.

© 2011 Software Improvement Group

3.6 Eis 6: diakritische tekens

3.6.1 Eis en conclusie

Gestelde eis:

Voor naamgeving dient de programmatuur de diakritische tekens van de GBA tekenset te ondersteunen.

Conclusie:

- Ja, er is aan deze eis voldaan.

3.6.2 Motivatie

- De programmatuur is geschikt voor verwerking van alle Unicode tekens, omdat het gebruik maakt van de UTF-8 codering voor Unicode. Appendix A.6 bevat een tabel die toont dat OSV 4 en 5 alle vereiste tekens ondersteunt.
- In de eisen wordt verwezen naar het logisch ontwerp 3.6 van de GBA. Hierin wordt ook Unicode codering genoemd als alternatieve codering in een webomgeving.
- De meest recente versie van het logisch ontwerp van de GBA is 3.8. Hierin zijn geen veranderingen ten aanzien van diakritische tekens.

In 2010 is er een bevinding geweest met het weergeven van de apostrof in kandidaatnamen. Dit probleem zat in OSV 1,2 en 3 en niet in 4 en 5 en is dus strikt genomen niet van invloed op dit rapport, maar wordt hier genoemd om de lezer een zo volledig mogelijk beeld te geven. De bevinding bestond eruit dat een apostrof in een naam altijd gevolgd werd door een spatie (*H. d' Ancona* in plaats van de correcte versie *H. d'Ancona*). In versie 2.6 van OSV is deze bevinding opgelost.

3.7 Eis 7: authenticiteit programmatuur

3.7.1 Eis en conclusie

Gestelde eis:

Het is mogelijk de authenticiteit van de programmatuur vast te stellen.

Conclusie:

- Ja, er is aan deze eis voldaan.

3.7.2 Motivatie

- Er is een op 'hashcodes' (digitale vingerafdruk) gebaseerde methode om de authenticiteit van de installatiebestanden vast te stellen. Hiermee wordt aan de eis voldaan.
- Voor hetzelfde doeleinde heeft de leverancier het gebruik van 'jar signing' getoond. Deze methode is geschikt wanneer alle programmatuur hiermee wordt gecontroleerd. Het is een vooruitgang dat deze alternatieve methode is toegevoegd, doordat gebruikers nu op meer manieren kunnen controleren.
- Bij het gebruik van de 'jar-signing' methode moet men er wel op letten dat installatiemedia alleen 'jar'-bestanden bevatten. Men moet geen andere bestanden toevoegen die niet controleerbaar zijn, anders is niet aan de eis voldaan.

© 2011 Software Improvement Group

3.8 Eis 8: authenticiteit gegevens

3.8.1 Eis en conclusie

Gestelde eis:

> *Alle elektronische communicatie van of naar andere programmatuur, hetzij via een netwerk, via opslagmedia of anderszins, is voorzien van een mogelijkheid om de authenticiteit van de gegevens vast te stellen, bij voorkeur door middel van een gekwalificeerde elektronische handtekening.*

Conclusie:

- Ja, er is aan deze eis voldaan.

3.8.2 Motivatie

- Bij elke uitvoer van gegevens wordt een 'hashcode' berekend en weergegeven in een afdrukbaar document. Door dit afgedrukte document kan de authenticiteit bij inladen van gegevens worden gecontroleerd.
- Er wordt gebruik gemaakt van een cryptografisch sterk hash algorithme (SHA-1) dat voor 2011 voldoende veiligheid biedt. Als het in de toekomst nodig mocht blijken om dit algoritme te wijzigen kan dit door wijziging van één regel in de broncode.
- In de huidige versie zijn wijzigingen doorgevoerd die foutief gebruik mogelijk kunnen maken. Deze zijn reeds behandeld bij Eis 3d in hoofdstuk 3.3.4.

Appendix A.8 toont de opzet van de report generator module, waarin is weergegeven hoe een hashcode berekend wordt.

3.9 Eis 9: formele methodes

3.9.1 Eis en conclusie

Gestelde eis:

> *Met behulp van formele methoden is wiskundig aangetoond dat berekeningen in de programmatuur precies datgene doen wat door de wet- en regelgeving is voorgeschreven.*

Conclusie:

- Nee, er is niet voldaan aan deze eis omdat het wiskundig document waarin de correctheid aangetoond wordt niet voldoende volledig is.

3.9.2 Motivatie

- Deze eis is van toepassing op het gedeelte van de broncode waarin de zetelverdeling bepaald wordt aan de hand van de getelde stemmen. De relevante regelgeving hiervoor is beschreven in hoofdstukken P en U van de Kieswet.
- Het wiskundig correct bewijzen van programmatuur is een zeer zware eis die alleen voor zeer kritieke en lastig wijzigbare programmatuur gebruikelijk is, zoals bijvoorbeeld besturingsprogrammatuur van voertuigen. Voor overige toepassingen wordt door de meeste organisaties het documenteren en testen van programmatuur als afdoende beschouwd.

© 2011 Software Improvement Group

- In een eerder rapport (5 oktober 2009) heeft de SIG geconcludeerd dat de wiskundige definities niet volledig zijn. Een verschilanalyse op het toenmalige DotER document (5.8) en de nieuwe versie (5.14) wijst uit dat de aanpak niet is gewijzigd. Daarmee wordt ook nu niet aan de eis voldaan.
- In een vorig rapport concludeerde de SIG tevens dat de programmatuur uitvoerig is gedocumenteerd en getest. Een aanvullende analyse toont dat dit nog steeds het geval is.

3.9.3 Toelichting

In de begrippenlijst behorende bij de brief van de staatssecretaris van 9 april 2008 worden formele methoden gedefinieerd als *op wiskunde gebaseerde technieken voor het specificeren, ontwikkelen en valideren van programmatuur.*

De SIG heeft in een eerder rapport geconcludeerd dat er weliswaar veel informatie over de berekeningswijze beschikbaar is in het document 'Determination of the election result' (DotER), maar dat dit geen wiskundig bewijs vormt. Een analyse van de verschillen tussen de DotER versies wijst uit dat er in dit opzicht geen andere aanpak wordt gebruikt: de gevonden wijzigingen hebben veelal betrekking op de Eerste Kamerverkiezingen (zie Appendices A.9.1 en A.9.2). Hiermee wordt ook nu niet voldaan aan de eis.
Er is voor dit rapport gecontroleerd of men nog steeds kan stellen dat de programmatuur uitvoerig is gedocumenteerd en getest. Dit is nog steeds het geval.

3.9.4 Beoordeling van documentatie en testset

Om te beoordelen of men nog steeds kan stellen dat de programmatuur uitvoerig is gedocumenteerd en getest, heeft de SIG een aanvullende analyse uitgevoerd die zich richt op de documentatie en de testset van OSV 4 en 5. In het bijzonder zijn drie onafhankelijke methoden toegepast:

a. De verschillen tussen het huidige DotER document (5.14) en de vorige versie (5.8) zijn bestudeerd. Hierbij zijn ook de bijbehorende verschillen in de broncode bepaald.
b. De verschillen tussen de huidige en de vorige broncodeversie zijn bestudeerd. Hierbij zijn ook de bijbehorende verschillen in het DotER-document beoordeeld.
c. De omvang van de testset is beoordeeld.

In het broncode-onderzoek heeft de SIG zich gericht op het package *'de.ivu.wahl.result,'* omdat dit package de voor deze eis relevante onderdelen bevat.

3.9.5 Resultaat van beoordeling documentatie en testset

De SIG concludeert dat de documentatie en testset van de programmatuur nog steeds uitvoerig zijn en er geen aanleiding is om te veronderstellen dat er nu meer kans op onjuiste berekeningen is.

- De verschilanalyses hebben uitgewezen dat alle code- en documentatiewijzigingen zijn terug te voeren op de toegevoegde ondersteuning voor de Eerste Kamerverkiezingen. Appendices A.9.1 en A.9.2 tonen de waargenomen wijzigingen.
- De testset van OSV 4 en 5 laat het programma zetelverdelingen bepalen op basis van fictieve stemuitslagen. Deze zetelverdelingen worden dan getoetst aan

© 2011 Software Improvement Group

voorgedefinieerde verwachte verdelingen. Van een goede testset mag worden verwacht dat deze alle zes ondersteunde verkiezingentypen test, en dat bijzondere situaties voldoende worden belicht (bijv. overleden kandidaten). SIG heeft hierbij de volgende bevindingen:

- Voor alle soorten verkiezingen zijn er meerdere testcases aanwezig. Appendix A.9.3 toont een overzicht van de aantallen testcases per verkiezingssoort.
- Voor alle soorten verkiezingen zijn bijzondere situaties getest. SIG beveelt wel aan om aanvullende tests te schrijven voor situaties die betrekking hebben op meerderheden bij 'provinciale staten 1,' 'provinciale staten 2,' 'gemeenteraden 2' en 'deelraden 2'. Leverancier IVU is hier reeds mee bezig en heeft reeds een aantal testen toegevoegd.

3.10 Eis 10: onafhankelijke toetsing

3.10.1 Eis en conclusie

Gestelde eis:

De programmatuur wordt in opdracht van de centrale stembureaus door een of meer onafhankelijke instanties getoetst voordat de centrale stembureaus de programmatuur accepteren en gebruiken. De uitkomst(en) van de toets(en) zijn openbaar.

Conclusie:

- Ja er wordt aan deze eis voldaan door publicatie van dit SIG-rapport.

3.10.2 Motivatie

- De SIG heeft deze toetsing onafhankelijk uitgevoerd.
- Dit eindrapport wordt door de Kiesraad openbaar gemaakt als resultaat van de toetsing van OSV 4 en 5 voor alle soorten verkiezingen.

3.11 Eis 11: elektronisch stemmen

3.11.1 Eis en conclusie

Gestelde eis:

Voor zover nog verder van toepassing dient de programmatuur te voldoen aan de aanbevelingen van de Raad van Europa voor elektronisch stemmen.

Conclusie:

- Deze eis is niet van toepassing.

3.11.2 Motivatie

- Er wordt niet elektronisch gestemd door middel van de onderzochte programmatuur. De aanbevelingen zijn daardoor niet van toepassing.

© 2011 Software Improvement Group

4 Technische kwaliteit van OSV 4 en 5

In dit hoofdstuk wordt het eindoordeel ten aanzien van de technische kwaliteit van OSV 4 en 5 uiteengezet. Deze technische kwaliteit is van belang voor de onderhoudbaarheid van het systeem in de komende jaren.

4.1 Conclusie voor de technische kwaliteit

Op basis van verschillende relevante technische aspecten concludeert SIG dat de technische kwaliteit van OSV 4 en 5 ★★★☆☆ is, op de SIG/TÜViT-schaal van ★☆☆☆☆ (minder onderhoudbaar) tot ★★★★★ (meer onderhoudbaar).

De technische kwaliteit (onderhoudbaarheid) van OSV 4 en 5 is ★★★☆☆.

Een score van ★★★☆☆ of hoger geeft aan dat het systeem de komende jaren onderhouden kan worden en daardoor toekomstvast is. Een score van ★★★☆☆ geeft aan dat dit kan gebeuren tegen marktgemiddelde inspanning.

4.2 Onderbouwing

Deze conclusie is gebaseerd op het model in Appendix B.1, dat de relatie toont tussen de verschillende ISO 9126 eigenschappen voor onderhoudbaarheid (*maintainability*) en de technische aspecten die SIG tijdens een Software Risk Assessment in beschouwing neemt. De volgende figuur toont de invulling van dit model voor OSV 4 en 5.

	Volume	Duplication	Unit size	Unit complexity	Unit interfacing	Module coupling	
Score	★★★★★	★★★	★★★	★★★	★★	★★★	**Score**
Analysability	X	X	X				★★★
Changeability		X		X		X	★★★
Stability					X	X	★★★
Testability			X	X			★★★

De beoordeling van elke eigenschap voor onderhoudbaarheid is het gemiddelde van de beoordelingen van de technische aspecten die deze eigenschap beïnvloeden (zoals blijkt uit de kruisjes in de tabel). Hierbij zijn de waarderingen vertaald in scores van ★☆☆☆☆ tot ★★★★★.

Het eindoordeel voor de technische kwaliteit (onderhoudbaarheid) is het gemiddelde van de beoordelingen van de vier onderhoudbaarheidseigenschappen.

© 2011 Software Improvement Group

5 Conclusies en aanbevelingen

5.1 Conclusies

Uit de toetsing aan gestelde eisen is naar voren gekomen dat de programmatuur OSV 4 en OSV 5 op de volgende kanttekeningen na voldoet aan gestelde eisen:

- Er wordt voldaan aan eis 2 (documentatie) zodra de specificatie van OSV 4 en 5 wordt bijgewerkt. De leverancier heeft toegezegd dit voor eind maart 2011 gereed te hebben. Het gaat hierbij onder andere om toegevoegde functionaliteit voor Eerste Kamerverkiezingen.
- Er is niet volledig voldaan aan eis 3d (foutief gebruik), omdat met het oog op gebruiksgemak enkele beveiligingen zijn verwijderd.
- Er is slechts gedeeltelijk aan eis 4c (open source) voldaan. Als positief punt valt op te merken dat de broncode beschikbaar is ter inzage. Om vast te stellen dat de programmatuur conform eis 4c 'open source' ontwikkeld is, moet de broncode onder een goedgekeurde open source-licentie worden gepubliceerd. Net als bij vorige versies is dit niet gebeurd voor deze versie.
- Gerelateerd aan eis 7 (authenticiteit programmatuur) geldt dat zolang de programmatuur in een afgeschermde omgeving gebruikt wordt, er aan eis 7 voldaan is.
- In eis 9 (formele methodes) wordt gesproken over wiskundig aangetoonde correctheid. Dit is een zeer zware eis die alleen voor zeer kritieke en lastig wijzigbare programmatuur gebruikelijk is. De geleverde wiskundige definitie is, net als in de eerdere toetsing, niet voldoende volledig om deze eis als voldaan te beschouwen. In een vorig rapport concludeerde de SIG dat de programmatuur wel uitvoerig is gedocumenteerd en getest. Een aanvullende analyse toont aan dat dit nog steeds het geval is.

Tevens is vastgesteld dat de programmatuur van gemiddelde technische kwaliteit is en daardoor voldoende onderhoudbaar en toekomstvast is. De technische kwaliteit is ten opzichte van de vorige toetsing niet gewijzigd.

5.2 Aanbevelingen

Voor de korte termijn zijn er geen andere aanbevelingen dan de reeds voorziene activiteiten (bijvoorbeeld bijwerken van de documentatie voor eis 2).

Op de lange termijn verdient het aanbeveling om de leverancier meer automatische regressietests te laten schrijven. Dit verkort de doorlooptijd van toekomstige wijzigingen. Ook wordt aanbevolen om de algehele onderhoudbaarheid van de software te verhogen, door een paar 'quick wins' op de bestaande code uit te voeren en hogere standaarden aan te houden voor nieuwe code. Hierbij moet men denken aan:

- Het oplossen van de gevonden wederzijdse afhankelijkheden om toekomstige wijzigingen gemakkelijker te maken;
- Meer specifieke foutafhandeling toevoegen als code wordt toegevoegd of gewijzigd om softwarefouten en uitzonderlijke situaties beter oplosbaar te maken.
- Het schrijven van nieuwe code met hogere onderhoudbaarheid.

© 2011 Software Improvement Group

A. Aanvullende informatie voor toetsing eisen

In deze appendix is per eis aanvullende informatie gegeven.

A.1 Eis 1: functionaliteit

De broncode van dit systeem is verdeeld in verschillende modules. Deze modules zijn weergegeven in onderstaande tabel, met daarbij de in de technische sessie gegeven omschrijving van de beoogde functionaliteit van de module. Er is sprake van een aantal kleine wijzigingen ten opzichte van de vorige versie, ontvangen in september 2009.

Module/package	Functionality	Location
Utilities	Utility functions not only used by part 4 or 5	Business layer
ResultModell	Functionality for displaying the data calculated in the Wahl.result module.	P4, P5 specific code
Web	Presentation of functionality as web pages.	P4, P5 presentation
Wahl.admin	Functionality of administrator role	Business layer
Wahl.anwender	User administration	Business layer
Wahl.auswertung	Reports and screen generation via the generator	Business layer
Wahl.base	Base and utility classes.	Business layer
Wahl.client	Presentation base for interaction with web browser (via JSP)	Business layer
Wahl.dataimport	Importing of election data in EML 230 format and election definition format	Business layer
Wahl.export	Exporting of election data in EML format	Business layer
Wahl.Eingang	Manual input of voting data in EML 150 format. Manual input of voting data.This part contains validity and plausibility checks.	P4, P5 specific code
Wah.l18n	Functionality to show program in different languages.	Business layer
Wahl.mbean	Extension to JBoss for creating data base structure, and for making exported files available via the web browser.	mbean
Wahl.Modell	Persistence Functionality to store election data elements in the data base.	Persistence
Wahl.result	Functionality of P5 to assign seats based on the voting data.	P4, P5 specific code
Wahl.runtime	Caching mechanism to store computed results to improve performance. It is not clear whether this is currently used, but may be needed in future.	Business layer
Reportgenerator	Functionality to create EML files and PDF/RTF output.	Report generator
xmlsecurity	Code to determine how EML files are validated, including the choice for the SHA-1 hash function.	Business layer

A.2 Eis 2: documentatie

In de onderstaande tabellen is het resultaat weergegeven van de review van de gedetailleerde specificatie. De eerste tabel bevat eisen specifiek voor de functionaliteit van P4, de tweede tabel bevat eisen specifiek voor de functionaliteit van P5, en de derde tabel bevat eisen over algemene functionaliteit die op beide programma's van toepassing is.

© 2011 Software Improvement Group

A.2.1 Eisen specifiek voor OSV 4

Section (page)	Summary	OK?	Remarks SIG
2.5 (43)	P4 automates the adding of votes level by level. The number and kind of levels differs per type of election. Votes counts are transported using EML between levels, except for the first level.	Yes	Tested.
2.5.1 (44)	An administrator must prepare the application for use.	Yes	Tested.
2.5.2.1 (44)	Access restriction by ID/password. There are three kinds of use(r)s: administration, data entry and finalization of data.	Yes	All three uses were Tested, but all with an administration user.
2.5.2.2 (45)	Description of what UI looks like, the available menu items.	Yes	Tested, not verified if each button is present in the right situation.
2.5.2.3 (48)	Relevant changes to the data are logged.	Yes	Tested.
2.5.2.4 (48)	Process description of manual input of election results. • No concurrent input during manual entry. • Entered data is checked for plausibility. • After data is entered, edits are only stored if data passes plausibility check. • Manual data needs to be entered twice.	Yes	Tested.
2.5.2.5 (53)	Process description of input of election results by EML file. • Validity of input file is verified by hash code. The user has to enter (part of) a hash code provided together with the EML file.	Yes	Tested.
2.5.2.6 (54)	A status bar shows the current state of data entry.	Yes	Some status information was observed in testing, but not all specified here.
2.5.2.7 (55)	Administrators can add, edit and delete polling stations for a municipality.	Yes	Tested.
2.5.2.8 (55)	The file with voting lists and candidates can be uploaded by an administrator.	Yes	Tested.
2.5.2.9 (55)	Results can be exported (in EML, PDF and/or RTF). Detailed specifications of the various subtypes of EML exported and form names for the PDF/RTF documents.	Yes	Tested. Not checked if the documents are of the specified types.
2.5.3 (57)	Inline help functionality is present.	Yes	Tested.
2.6 (58)	An adapted version of the P4 for use in referenda. Use is similar to normal P4, P1 through P3 are not used because there is no list of candidates. Most relevant difference is that it is possible to create a referendum definition from the application.	Yes	Tested.

© 2011 Software Improvement Group

A.2.2 Eisen specifiek voor OSV 5

Section (page)	Summary	OK?	Remarks SIG
2.7 (63)	P5 determines the distribution of seats based on the votes.	Yes	Tested.
2.7.1 (63)	An administrator must prepare the application for use.	Yes	Tested.
2.7.2.1 (63)	Access restriction by ID/password. There are three kinds of use(r)s: administration, data entry and finalization of data.	Yes	All three uses were tested, but all with an administration user.
2.7.2.2 (64)	Description of what user interface looks like, the available menu items.	Yes	Tested, not verified if each button is present in the right situation.
2.7.2.3 (66)	The file with voting lists and candidates (including address data) can be uploaded by an administrator.	Yes	Tested.
2.7.2.4 (66)	Process description of input of election results by EML file. Validity of input file is verified by hash code. The user has to enter (part of) a hash code provided together with the EML file.	Yes	Tested.
2.7.2.5 (68)	The seat distribution is calculated in the manner described by the formal specification.	Yes	Tested.
2.7.2.6 (69)	In some situations, manual drawing of lots is required for the seat distribution.	Yes	Tested
2.7.3 (69)	Relevant steps during seat distribution are logged.	Yes	Tested.
2.7.4. (69)	A *proces verbaal* can be generated after calculation of the seat distribution (in EML, PDF and/or RTF).	Yes	Tested, but not verified if the documents are of the specified subtypes and contain all described fields.
2.7.5 (70)	Voting results can be stored on a predefined location on disk.	Yes	Tested.
2.7.6 (70)	Inline help functionality is present.	Yes	Tested

A.2.3 Overige eisen van toepassing op OSV 4 en OSV 5

Section (page)	Summary	OK?	Remarks SIG
2.8 (71)	Detailed description of the form generator, which reused between programs.	Yes	Tested.
2.8.1 (71)	Directories for import and export of files can be configured. Default settings available. Detailed specification of the naming conventions used for created directories and files.	Yes	Tested. Not all folders, naming conventions, etc. were checked.
2.8.2 (77)	Input of non-standard characters allowed by the GBA character set is possible using a hotkey.	Yes	Input is not relevant for P4 and P5, but it was tested that display of these characters works.
2.9 (78)	*The installation wizard will:* • Show the license agreement. • Allow for selection of an installation directory. • Allow the user to select the desired program components. • Ask for the installation of shortcuts. • Install the application.	Yes	

© 2011 Software Improvement Group

A.2.4 Functionele wijzigingen die nog niet in de specificatie zijn verwerkt

De volgende tabel toont een overzicht van functionele wijzigingen in de huidige versie van OSV 4 en 5, die nog niet zijn beschreven in de gedetailleerde specificatie. De ticketnummers en beschrijvingen zijn afkomstig uit het issue tracking systeem van leverancier IVU. De leverancier IVU heeft op al deze punten een reactie gegeven, die of een afdoende verklaring bieden of waaruit blijkt dat eind Maart 2011 de punten zijn opgelost.

Ticket no.	Description	Response IVU
-	Integrate Eerste Kamer elections support	2)
-	Integrate step-by-step instructions (Stappenplan) into commands menu	2)
OSV-394	P4: It's not possible to differ between briefstembureaus and normal stembureaus	2)
OSV-608	Modification of the hashcode handling	2)
OSV-697	User rights are too complex for P4	2)
OSV-727	P5: Model P 22 is probably going to be changed before gemeenteraadselections	3)
OSV-755	P4: CSV Export of counted votes	2)
OSV-845	P5: Export all candidate data in CSV file for further processing in other programs.	2)
OSV-931	P5: CSV export of names and addresses of all candidates	2)
OSV-947	P5: EK election: No fictitious seat distribution	1)
OSV-948	P0: Preferential barrier = 100% KT	1)
OSV-986	P5-EK: Details of model U 16	3)

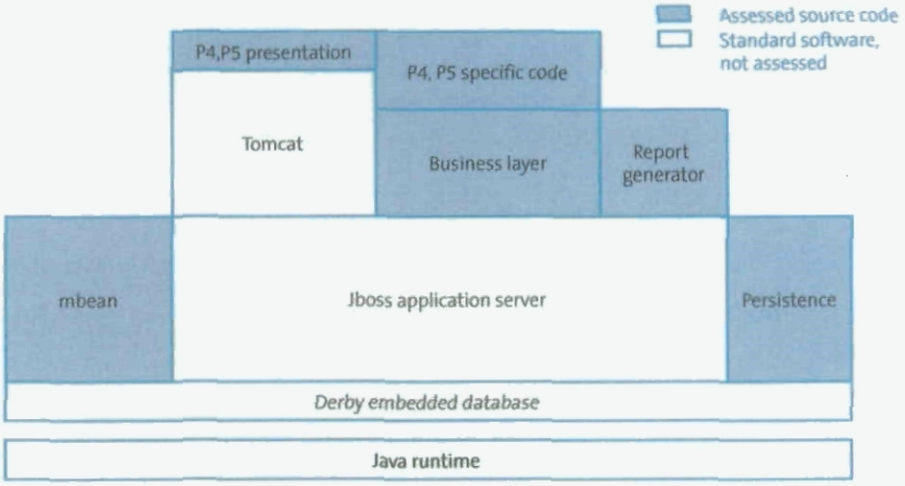
1) This functional modification is described in detail in the "DeterminationElectionResultEP_EN_NL.doc" document. Version 3.5 of this document dating from 03-06-2010 covers the functional modification OSV-947, OSV-948
2) The specification will be updated before end March 2011.
3) The layout and details of the official models is prescribed by decision of the ministry of the interior ("Kiesbesluit"). Another specification of the models does not exist and is not necessary. The requirements implemented in the context of OSV-727 were described in a decision of the ministry of the interior of Nov. 2009. The requirements implemented in the context of OSV-986 were described in a decision of the ministry of the interior of 07-12-2010.

© 2011 Software Improvement Group

A.3 Eis 3: ontwerp

A.3.1 Architectuur

De architectuur van het programma is weergegeven in de onderstaande figuur. Voor de blauw gearceerde modules in dit ontwerp is er broncode beschikbaar. De witte blokken zijn standaardpakketten die geen onderdeel uitmaken van de broncode van OSV 4 en 5.



A.3.2 Afhankelijkheden tussen modules

Door middel van broncode-analyse is vastgesteld wat de afhankelijkheden tussen deze modules zijn. Hiervoor is bepaald hoe vaak de code in een bepaalde module een aanroep doet van een code-eenheid in een andere module. Dit is weergeven in de onderstaande figuur.

De figuur toont een aantal wederzijdse afhankelijkheden tussen de modules: voorbeelden waarbij er aanroepen zijn van module A naar B en aanroepen van B naar A. Alleen afhankelijkheden die meer dan 5 keer voorkomen zijn weergegeven. Het cijfer bij elke pijl geeft het aantal aangetroffen afhankelijkheden weer. SIG beveelt aan om deze op te lossen om toekomstige veranderingen gemakkelijker te maken.

© 2011 Software Improvement Group



Figuur 2: In de broncode aangetroffen afhankelijkheden tussen modules. Weergegeven is welke modules elkaar meer dan 5 maal aanroepen. Op sommige plaatsen is er een pijl van A naar B en van B naar A. Dit zijn zogeheten wederzijdse afhankelijkheden die onderhoud bemoeilijken.

A.3.3 Aanpasbaarheid zonder programmatuurwijzigingen

Er zijn geen grote wijzigingen in de aanpasbaarheid van OSV 4 en 5: configuratie vereist nog altijd geen programmeerinspanning. Wel wordt nu ook de mogelijkheid geboden om statische teksten gemakkelijk aan te passen.

A.3.4 Voorkomen foutief gebruik

Voor het controleren van de authenticiteit van bestanden worden drie veiligheidsniveaus gehanteerd:

a) Geen hashcode-controles.
b) De gebruiker wordt gevraagd of de getoonde hashcode correct is.
c) De gebruiker wordt gevraagd om (een deel van) de hashcode.

Voor de programma's OSV 4 en 5 worden voor de verschillende stappen de volgende veiligheidsniveaus gebruikt. In de tabel betekent een schuine streep dat het lagere niveau wordt gebruikt als een gebruiker of organisatie het bestand zojuist zelf heeft geproduceerd, en dat het hogere niveau wordt gebruikt wanneer het bestand wordt ontvangen van een andere gemeente. Op gemeenteraadsverkiezingen is altijd het lagere level van toepassing.

© 2011 Software Improvement Group

Bestand	Gebruikt veiligheidsniveau in OSV 4	Gebruikt veiligheidsniveau in OSV 5
Election definition	a	a
(Candidate) list	b/c	a
Polling stations	a	n.v.t.
Count	a/c	a
Result	n.v.t.	n.v.t.

Ten slotte volgt hieronder een lijst van testen die het programma doet om verkeerd gebruik van de programmatuur te voorkomen. "Error" betekent dat de gebruiker niet verder kan zonder de fout te verbeteren; "Warning" betekent dat de gebruiker alvorens verder te gaan moet bevestigen dat de ingevoerde data correct is.

Input	Results in
In all fields: no negative value allowed	Error
In all fields: only numbers allowed (no characters)	Error
Number of people entitled to vote may not be 0	Error
Number of people entitled to vote may not be 9999999999 or larger	Error
Sum of blank, invalid and valid votes must be equal to the total number of votes, error when larger or smaller	Error
Sum of blank votes may not be larger than the total number of votes	Error
Number of blank votes may not be too high (according to defined threshold value)	Warning
Number of invalid votes may not be too high (according to defined threshold value)	Warning
Sum of all votes (distributed over political parties) must be equal to the total number of valid votes (error when larger or smaller number)	Error
Sum of votes for one political party must be equal to the votes distributed over the candidates for this party (error when smaller or larger)	Error
Second input: whenever a value is entered, that differs from the first input, an error appears	Error
Input of voting district is exactly the same as a previous district	Warning

A.4 Eis 4: open standaarden en open source

A.4.1 Gebruik open standaarden en EML

Naast de bestaande standaarden in OSV 4/5 (EML, PDF en RTF) is de informele standaard CSV toegevoegd. Het gaat echter niet om een standaard waarop ondersteuning geleverd hoeft te worden door een derde partij, wat betekent dat iedereen deze standaard te allen tijde kan gebruiken.

A.4.2 Gangbare programmeertaal met open source compiler

- Java: Er is een actieve Eclipse gemeenschap, wat blijkt uit de activiteiten vermeld op http://www.eclipse.org/community/.
- JavaScript: Er is een actieve FireFox gemeenschap, wat blijkt uit de website http://www.spreadfirefox.com/.
- JSP: Er is een actieve JBoss gemeenschap, wat blijkt uit de JBoss website op http://www.jboss.org/.
- XSLT: Er is een actieve Xalan gemeenschap die te bereiken is via http://xml.apache.org/xalan-j/.

© 2011 Software Improvement Group

- MySQL: Er is een actieve MySQL gemeenschap, wat blijkt uit de MySQL website op http://mysql.com/news-and-events/.
- Derby: Er is een actieve Derby gemeenschap, wat blijkt uit de Derby website op http://db.apache.org/derby/#Events.

A.4.3 Ontwikkeld als open source

De situatie omtrent de open source ontwikkeling en beschikbaarheid van OSV 4 en 5 is onveranderd ten opzichte van de vorige rapportage. Deze situatie staat omschreven in de hoofdtekst van dit rapport.

A.5 Eis 5: verschillende besturingssystemen

De in Appendix A.3.1 weergegeven architectuur toont aan dat de programmatuur gebaseerd is op het JBoss-platform dat weer gebaseerd is op het Java-platform. Er wordt gebruik gemaakt van JBoss versie 4.2.3 en Java versie 1.6. Deze platformen worden ondersteund in de besturingssystemen Windows, Linux en MacOS.

A.6 Eis 6: diakritische tekens

In het logisch ontwerp versie 3.6 van het GBA wordt ook gesproken over het mogelijke gebruik van Unicode voor het weergeven van de voor het GBA gebruikte diakritische tekens.

> *Bij de uitwisseling van gegevens gebruikt de LRD Unicode en niet de manier van coderen die is beschreven in Bijlage II Teletex. Unicode wordt gebruikt omdat de LRD met behulp van webtechnologie wordt bevraagd en het gebruik van Unicode in de webomgeving gebruikelijk (p.633).*

Inmiddels is versie 3.8 van het GBA beschikbaar; deze bevat voor wat betreft diakritische tekens geen wijzigingen ten opzichte van versie 3.6. De volgende tabel toont de ondersteuning van diakritische tekens in OSV 4 en 5.

Naam	Weergave (voorbeeld)	Ondersteund?
Accent acute	á	Ja
Accent grave	à	Ja
Accent circumflex	â	Ja
Trema	ä	Ja
Tilde	ã	Ja
Ring	å	Ja
Macron	ā	Ja
Ogonek	ą	Ja
Caron	č	Ja
Punt	ċ	Ja
Cedille	ç	Ja
Breve	ğ	Ja
Dubbele accent acute	ő	Ja

A.7 Eis 7: authenticiteit programmatuur

De authenticiteit van de programmatuur kan worden gewaarborgd via hashcodes:

© 2011 Software Improvement Group

- De aanleverende partij berekent een hashcode (digitale vingerafdruk) en geeft deze door per brief aan de gebruiker van de programmatuur.
- De gebruiker ontvangt de installatiebestanden elektronisch.
- De gebruiker berekent de hashcode over deze bestanden met behulp van reeds *aanwezige programmatuur*.
- De gebruiker vergelijkt deze hashcode met de in brief gegeven hashcode. Als deze gelijk zijn, zijn de installatiebestanden authentiek.

Daarnaast kan jar signing worden gebruikt:

- De aanleverende partij voorziet het installatiebestand (jar-file) van een digitale handtekening. Dit gebeurt in essentie op dezelfde wijze als bovenstaand proces, maar bij het berekenen van de hashcode wordt bovendien gebruik gemaakt van een zogeheten geheime sleutel. Voor partijen zonder deze sleutel is het niet mogelijk om de handtekening na te maken.
- De gebruiker ontvangt het installatiebestand elektronisch.
- De gebruiker voert middels reeds aanwezige programmatuur een controle uit op het ontvangen installatiebestand.
- De gebruiker ontvangt een mededeling of het installatiebestand authentiek is.

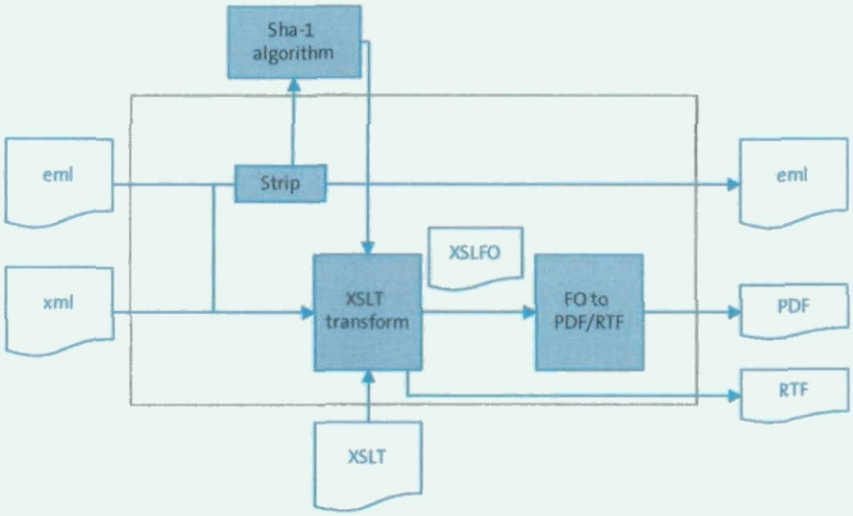
A.8 Eis 8: authenticiteit gegevens

Om de authenticiteit van gegevens die de programmatuur produceert te kunnen controleren, is gekozen voor controle van authenticiteit door middel van een hashcode (digitale vingerafdruk) in een begeleidend afgedrukt document. De module die hiervoor verantwoordelijk is is de report generator. De werking hiervan is weergeven in de onderstaande figuur. De stappen van de controle zijn de volgende:

- Vanuit de programmatuur wordt een EML bestand met stemgegevens aan de report generator gestuurd.
- *De report generator verwijdert niet-essentiële informatie* ('strippen') en slaat de gestripte EML op en stuurt deze bovendien naar het SHA-1 algoritme.
- De XSLT transformator ontvangt ook het oorspronkelijke EML bestand, en maakt hier een afdrukbaar document van, dat bovendien de door het SHA-1 algoritme berekende hashcode bevat.
- Het afdrukbare document wordt opgeslagen in PDF of RTF.
- De gebruiker zet het EML bestand op een informatiedrager (bijvoorbeeld CD of USB-stick) en drukt het afdrukbare document af.
- Een andere gebruiker ontvangt zowel de informatiedrager als het afgedrukte document.
- Deze gebruiker vraagt het programma om het EML bestand in te laden.
- Afhankelijk van het beveiligingsniveau wordt de gebruiker gevraagd om de hashcode te controleren en aan te vullen:
    - Op het hoogste beveiligingsniveau is de gebruiker verplicht om de eerste vier tekens van de hashcode in te voeren, en kan niet verder worden gegaan zonder het afgedrukte document.
    - Op het middelste beveiligingsniveau wordt de hashcode getoond en wordt de gebruiker gevraagd om deze te bevestigen middels een 'ja/nee' vraag.

© 2011 Software Improvement Group

- Op het laagste beveiligingsniveau wordt de hashcode alleen in een logbestand weggeschreven en hoeft de gebruiker deze niet te controleren.



A.9 Eis 9: formele methodes

A.9.1 Aanpassingen in het DotER document

De volgende tabel toont de aanpassingen die SIG heeft waargenomen tussen versies 5.8 en 5.14 van het DotER document. Hierbij zijn de bijbehorende broncodewijzigingen bestudeerd en becommentarieerd.

© 2011 Software Improvement Group

Section	Change	Corresponding code changes found in	Remarks SIG
1.4.6	No combined lists EK elections.	CombinedListValidator.java	
1.4.8	Votes cast and vote values distinction EK.	VotesBuilder.java ElectoralDistrict.java	
1.5	*All differences for EK elections.*	VotesBuilder.java ElectoralDistrict.java ElectionResultForCandidatesDeterminator.java GsdaParameters.java CombinedListValidator.java	
2.1	Absolute majority rule for EK elections.	GsdaParameters.java ElectionResultForCandidatesDeterminator.java	
2.3.6 – 6a	Absolute majority rule for EK elections.	ElectionResultForCandidatesDeterminator.java	
2.4.1 – 4	Only take lists with at least one vote into account.	GeneralSeatDistributor.java	Due to ticket OSV-813.
3.2.1 – 0	Votes cast and vote values distinction EK.	VotesBuilder.java ElectoralDistrict.java	
3.6.2 – 3b	Preferential barrier difference for EK elections.	GsdaParameters.java ElectionResultForCandidatesDeterminator.java	"TODO" found in this section of the document; Response IVU: this will be fixed in the *next version.*
4.2.3.1	No combined lists for EK elections.	CombinedListValidator.java	Throws exception when this condition is not met.
4.3 – 3	Absolute majority rule for EK elections.	GsdaParameters.java ElectionResultForCandidatesDeterminator.java	

A.9.2 Aanpassingen in de broncode

De volgende tabel toont de aanpassingen die SIG heeft waargenomen tussen de huidige OSV 4/5 programmatuur (13 december 2010) en die van de vorige toetsing (25 september 2009). Hierbij zijn de bijbehorende wijzigingen in het DotER document bestudeerd en becommentarieerd.

© 2011 Software Improvement Group

File	Change	Corresponding document changes	Remarks SIG
VotesBuilder.java	Votes cast and vote values distinction EK.	1.4.8 1.5 3.2.1	
CandidateForSorting.java	P3 list added	Not applicable	Introduced for functional requirement / convenience
CombinedListValidator.java	Combined lists, preferential barrier differences for EK elections.	1.4.6 1.5 4.2.3.1	
ElectionResultForCandidatesDeterminator.java	Preferential barrier difference for EK elections.	1.5 2.1 2.3.6 – 6a 3.6.2 – 3b 4.3 – 3	
P2ListForPrioritySeatsDeterminator.java	minOfRemainingSeatsAndNoOfCandidates changed from long to integer	Not applicable	Integer is sufficient to hold value
DrawingLotsIdentifier.java	P3 list handling changed	Not applicable	Introduced for functional requirement / convenience
GsdaParameters.java	Absolute majority rule for EK elections.	1.5 2.1 3.6.2 – 3b 4.3 – 3	
CandidateResult.java	Check if candidate is elected on one of the P2 lists of the current P3 list.	Not applicable	Redundant information stored for convenience
P3ListCandidate.java	Lot drawing functionality for P3 lists?	Not applicable	Introduced for functional requirement / convenience

A.9.3 Testdekking

De volgende tabel toont het aantal beschikbare testcases voor elke combinatie tussen een functionaliteitscategorie en een type verkiezingen. IVU heeft toegezegd om uiterlijk 14 januari 2011 meer testcases toe te voegen voor de Eerste Kamerverkiezingen.

	EP	EK	TK	PS2	PS1,GR2,DR2	GR1,DR1
Combined lists	22	N/A	28	23	17	18
Deceased candidate	3	2	4	4	3	3
Drawing lots	40	8	63	56	33	37
List exhaustion	9	8	18	18	9	10
Majority	4	1	4	0	0	6
Preferential votes	10	2	8	14	10	10
Residual seats	6	1	7	3	6	11
Other	1	4	15	18	4	10
Total	95	26	147	136	82	105

Ter illustratie volgt hieronder een voorbeeld uit het Excel-bestand *GRTestCasesAndResults.xls*, testcase *GR01*. Het definieert een verkiezing met een aantal parameters (bijv. aantal zetels, aantal kiesgerechtigden, etc.), en een verkiezingsuitslag (per lijst is per kandidaat het aantal stemmen gegeven). Daarnaast is ook de verwachte zetelverdeling aangegeven (d.w.z., wie de zetels ontvangt).

© 2011 Software Improvement Group

Op basis van deze gegevens worden EML-invoerbestanden gegenereerd. Deze bestanden worden ingelezen, en de zetelverdeling wordt bepaald. Deze uitkomst wordt vergeleken met de verwachte uitkomst uit het Excel-bestand. Indien deze niet gelijk is, faalt de test.

De code voor deze tests bevindt zich in de directory *osv_test*. Belangrijke classes zijn:

- *TestExcelGeneratedCases* – voert de tests uit de Excel-documenten uit.
- *ExcelToEMLTransformer* – converteert de Excel-invoer naar EML.
- *SeatDistributionTestCase* – controleert of de zetelverdeling e.d. gelijk is aan de verwachte uitkomst.

Basic test case, distribution by greatest remainder

Election Definition			Meta information	
numberOfSeats	13		Districts	1
preferenceBarrierNumerator	1		Lists	5
preferenceBarrierDenominator	2		Subdistrict	GR1,DR1
numberOfElectoralDistricts	1		KT	459.2308
total number of votes	5970		Vs	229.5154

Candidate lists	List nr.	# of Candidates	combined	Result
Greens	1	12		5
Reds	2	10		4
Blues	3	8		2
Blacks	4	4	a	1
Whites	5	4	a	1

Candidates	List position	List	seats (0 or 1)	Votes
Greens1	1	Greens	1	360
Greens2	2	Greens	1	330
Greens3	3	Greens	1	300
Greens4	4	Greens	1	270
Greens5	5	Greens	1	240
Greens6	6	Greens	0	210
Greens7	7	Greens	0	180
Greens8	8	Greens	0	150
Greens9	9	Greens	0	120
Greens10	10	Greens	0	90
Greens11	11	Greens	0	60
Greens12	12	Greens	0	30
		Greens	5	2340
Reds1	1	Reds	1	300
Reds2	2	Reds	1	270
Reds3	3	Reds	1	240
Reds4	4	Reds	1	210
Reds5	5	Reds	0	180
Reds6	6	Reds	0	150
Reds7	7	Reds	0	120
Reds8	8	Reds	0	90
Reds9	9	Reds	0	60
Reds10	10	Reds	0	30
		Reds	4	1650
Blues1	1	Blues	1	240
Blues2	2	Blues	1	210
Blues3	3	Blues	0	180
Blues4	4	Blues	0	150
Blues5	5	Blues	0	120
Blues6	6	Blues	0	90
Blues7	7	Blues	0	60
Blues8	8	Blues	0	30
		Blues	2	1080
Blacks1	1	Blacks	1	200
Blacks2	2	Blacks	0	150
Blacks3	3	Blacks	0	100
Blacks4	4	Blacks	0	50
		Blacks	1	500
Whites1	1	Whites	1	150
Whites2	2	Whites	0	120
Whites3	3	Whites	0	80
Whites4	4	Whites	0	50
		Whites	1	400

Assignment steps				Lists					
Index	Parent	Type	lot	a	1	2	3	4	5
1		FA		1	5	3	2		
2		LR		1	0	0	0		
3		LR		0	0	1	0		
1	a	FA						1	0
2	a	LR						0	1
				2	5	4	2	1	1

© 2011 Software Improvement Group

B. Aanvullende Informatie voor Technische Kwaliteit

Deze sectie bevat aanvullende informatie over de aspecten van technische kwaliteit die door de SIG zijn onderzocht.

B.1 Model voor technische kwaliteit

SIG gaat uit van het ISO 9126 model voor de kwaliteit van software. Dit model definieert met betrekking tot onderhoudbaarheid van software vier aspecten die invloed hebben op onderhoudbaarheid. Dit zijn *analyseerbaarheid, veranderbaarheid, stabiliteit,* en *testbaarheid*. Het model geeft ook definities over hoe deze aspecten te meten. Analyseerbaarheid kan bijvoorbeeld gemeten worden door statistieken bij te houden over hoe lang een ontwikkelaar nodig heeft om fouten op te sporen.

De ISO 9126 eigenschappen van onderhoudbaarheid kennen de volgende definities:

- **Analyseerbaarheid** – Eigenschappen van de software die van invloed zijn op de inspanning benodigd voor de diagnose van gebreken of foutoorzaken, of voor de identificatie van onderdelen die moeten worden gewijzigd.
- **Veranderbaarheid** – Eigenschappen van de software die van invloed zijn op de inspanning benodigd voor het doorvoeren van de benodigde wijzigingen in de software.
- **Stabiliteit** – Eigenschappen van de software die van invloed zijn op het risico van onverwachte effecten van wijzigingen.
- **Testbaarheid** – Eigenschappen van de software die van invloed zijn op de inspanning benodigd voor het valideren van de gewijzigde software.

Het is ook mogelijk om deze informatie af te leiden uit de directe analyse van broncode. Hieronder volgt een generiek model, ontwikkeld door de Software Improvement Group, waarin de ISO 9126 karakteristieken worden gerelateerd aan statische analyse van broncode en bijbehorende documentatie. In de matrix staan in de eerste kolom de ISO 9126 attributen. Op de bovenste rij staan een tiental metingen. Een kruis in de matrix geeft aan dat een meting meeweegt tot een specifiek karakteristiek. Zo staat er bijvoorbeeld een kruis bij volume en analyseerbaarheid. Dat betekent dat de omvang van een systeem invloed heeft op de analyseerbaarheid van software.

	Volume	Duplication	Unit size	Unit complexity	Unit interfacing	Module coupling
Analysability	X	X	X			
Changeability		X		X		X
Stability					X	X
Testability			X	X		

© 2011 Software Improvement Group

B.2 Toelichting bij beoordelingen

De volgende tabel geeft per beoordeeld aspect een korte toelichting van de waarnemingen waarop de beoordeling is gebaseerd. De volgende paragrafen tonen per technisch aspect de onderliggende metingen.

Voor de metingen is gebruik gemaakt van versie 3.11.2 van de SIG analyse tools.

Technical aspect	Assessment	Motivation
Volume	★★★★★	The source code represents 97 man-months of development effort.
Duplication	★★★☆☆	7% of the Java code and 65% of the JSP code concerns duplicated code.
Unit length	★★★☆☆	13% of the Java code resides in methods that are longer than 50 lines of code.
Unit complexity	★★★☆☆	3% of the Java code resides in methods that have a McCabe higher than 20.
Unit interfacing	★★☆☆☆	6% of the Java code resides in methods that use more than four parameters.
Module coupling	★★★☆☆	15% of the Java code resides in files that are called from more than 32 different locations.

B.3 Scope van de metingen

De scope van de metingen in dit hoofdstuk omvat versie 2.6.0 van de OSV 4 en 5 broncode (13 december 2010). Gegenereerde code is hierin niet meegenomen.

B.4 Volume

In de onderstaande tabel is het volume van de broncode weergegeven per gebruikte taal en technologie. Hieruit blijkt dat dit een hoofdzakelijk in Java ontwikkeld systeem is.

Technology	Lines of code
Java	43,000
JavaScript	300
Java Test	4,700
Java Generated	25,000
JSP	15,000
XSLT	65

De Java testcode betreft code die in Java geschreven is en die niet nodig is om het programma te gebruiken, maar controles doet op de werking van delen van het programma. Het beschikbaar hebben van testcode zorgt ervoor dat de werking van het programma automatisch gecontroleerd kan worden, wat de onderhoudbaarheid ten goede komt. Naast deze testcode zijn er ook in spreadsheets vastgelegde testen gemaakt specifiek voor het testen van de juistheid van de zeteltoekenning.

© 2011 Software Improvement Group

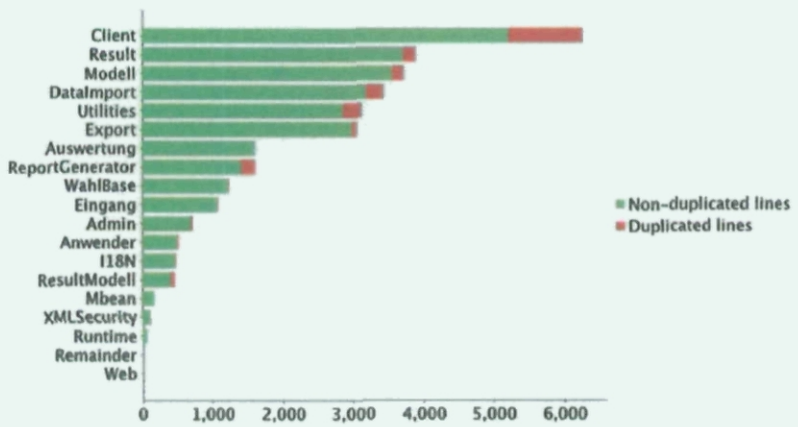
B.5 Duplicatie

Indien de broncode duplicatie bevat maken deze het systeem moeilijker te onderhouden. Een blok broncode wordt als duplicaat beschouwd als dezelfde regels op twee of méér plaatsen in het systeem voorkomen. Een blok broncode wordt als gedupliceerd geteld als er ten minste zes achtereenvolgende regels dubbel voorkomen. Als er sprake is van drie identieke blokken code van zes regels dan zijn er 18 regels gedupliceerd en 12 regels redundant. Redundante regels zouden bij opschoning uit het systeem verwijderd kunnen worden.

De volgende tabel toont de duplicatie per technologie.

Technology	Duplicated lines %	Redundant lines %
Java	7%	4%
Java Test	8%	4%
Java Generated	29%	21%
JavaScript	10%	5%
JSP	65%	40%
XSLT	28%	14%

De volgende figuur toont de duplicatie in de Java-code per module.



De volgende tabel toont een overzicht van de langste duplicaten in de Java-code.

Length	File	Start line	Perc.
66	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P4.java	182	20%
66	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommandReferendum_P4.java	126	27%
48	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P5.java	213	18%
48	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P4.java	180	15%
47	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P5.java	215	18%
47	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommandReferendum_P4.java	126	19%

© 2011 Software Improvement Group

39	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P4.java	411	12%
39	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommandReferendum_P4.java	314	16%
38	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommand_P5.java	225	15%
38	osv_programma4en5_source_v2.6/osv45/src/de/ivu/wahl/client/beans/InitGuiCommandReferendum_P4.java	136	16%

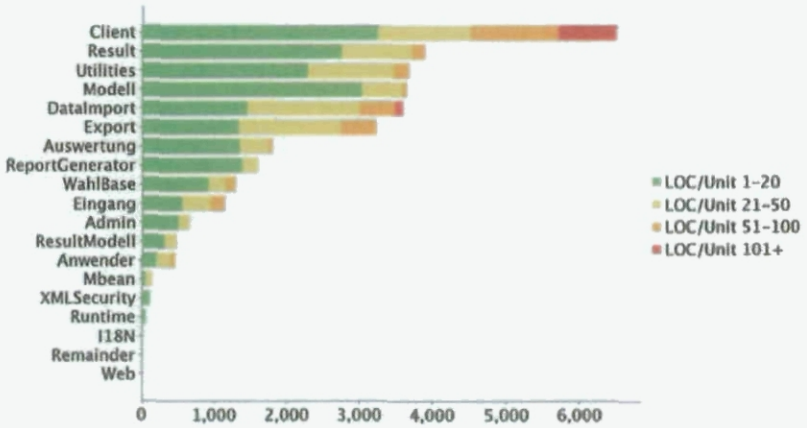
B.6 Unit lengte

De lengte van eenheden van code is een maat voor de begrijpbaarheid. Eenheden kunnen bijvoorbeeld methoden zijn in Java of functies in C. Hoe langer een eenheid, des te moeilijker deze is te begrijpen. Om de lengte van eenheden de classificeren wordt gebruik gemaakt van de volgende tabel.

Unit lengte	Risico
1-20	Makkelijke te begrijpen, laag risico
21-50	Enigszins lange unit, medium risico
51-100	Lange unit, hoog risico
>>100	Erg lange unit, zeer hoog risico

Om een systeem te kunnen beoordelen worden de regels van een eenheid die in een bepaalde categorie voorkomen opgeteld. Van alle regels per categorie samen wordt het percentage van het complete volume bepaald.

De volgende figuur toont de verdeling van de verschillende unit lengte-categorieën in de Java-code.



De volgende tabel toont een overzicht van de langste units in de Java-code.

Unit	Lines of code
RepositoryPropertyHandler.handlePropEingabeAllg(HttpServletRequest,Map)	173
WahlImportBean.$constructor()	146
InitGuiCommand_P4.initLevelUnabhaengig(Map,GUICommandList[],String,String,String,String)	130
ApplicationBean.doLogin(HttpServletRequest)	118
ErgebnisImportBean.$constructor()	116

© 2011 Software Improvement Group

EingabeBean.setGuiMsg(HttpServletRequest)	109
ImportHandlingBean.createStimmbezirke(URL)	103
AbstractImportClient.createGebiete(Element,String,URL)	98
InitGuiCommandReferendum_P4.initLevelUnabhaengig(Map, GUICommandList[],String,String,String,String)	98
InitGuiCommand_P5.initTopLevel(Map,GUICommandList[])	92

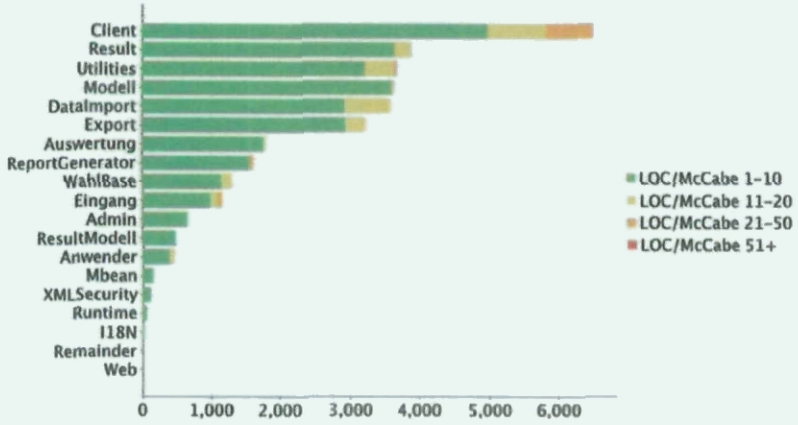
B.7 Unit complexiteit

Om de complexiteit van broncode van een systeem te bepalen gebruikt SIG de Cyclomatische Complexiteit van McCabe, een technologieonafhankelijke metriek die in 1976 gedefinieerd is door T. McCabe. De McCabe-waarde telt het kleinste aantal individuele executiepaden door de kleinste eenheid van broncode. Deze kleinste eenheid is bijvoorbeeld voor Java een methode. Hoe hoger de McCabe-waarde van een eenheid, des te lastiger de code is te begrijpen en te testen.

Om complexiteitsniveaus te benoemen en geassocieerde risico's te bepalen wordt gebruik gemaakt van de volgende classificatie van het Software Engineering Institute (SEI) van Carnegie Mellon University.

McCabe-waarde	Risico
1-10	Makkelijk te begrijpen, laag risico
11-20	Complex, medium risico
21-50	Erg complex, hoog risico
>> 50	Ontestbare code, zeer hoog risico

De volgende figuur toont de verdeling van de verschillende complexiteitscategorieën in de Java-code.



De volgende tabel toont een overzicht van de meest complexe units in de Java-code.

Unit	McCabe
RepositoryPropertyHandler.handlePropEingabeAllg(HttpServletRequest,Map)	50
ClientHelper.forHTML(String)	36
WahlImportBean.$constructor()	31
ApplicationBean.filterForState(List)	28
ApplicationBean.doLogin(HttpServletRequest)	27
ReportNameComponentsP4.getNameComponents()	24

© 2011 Software Improvement Group

AdministrationBean.executeCommand(HttpServletRequest,int)	23
ApplicationBean.initStandardBefehle()	22
EingangHandlingBean.fillEingangMsg(GUIEingangMsg,String,String)	22
SimpleReadWriteEJBLock.getWriteLock(Transaction)	22

B.8 Unit interfacing

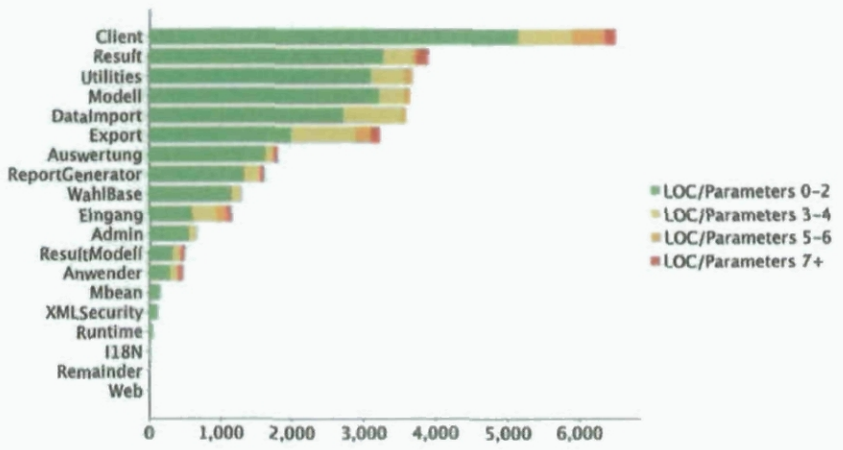
Unit interfacing is een maat voor de grootte van interfaces in units. Units die veel parameters gebruiken hebben een negatieve invloed op de stabiliteit van het systeem, omdat het moeilijker wordt om onverwachte invloeden van veranderingen op andere systeemonderdelen te voorkomen. Ook is bijvoorbeeld het wijzigen van een lijst soortgelijke objecten (zoals strings) lastig omdat de parametervolgorde behouden moet worden.

Unit interfacing wordt gemeten als het aantal parameters van een unit. De rating voor dit onderdeel kan worden verbeterd door het gebruik van datastructuren waarin meerdere parameters tegelijkertijd kunnen worden doorgegeven.

De verschillende niveaus en bijbehorende risico's zijn weergegeven in de volgende tabel.

Aantal parameters	Risico-inschatting
0-2	Geen risico (eenvoudig stabiliseerbaar)
3-4	laag risico
5-7	medium risico
> 7	hoog risico

De volgende tabel toont de verdeling van deze niveaus in de Java-code.



De volgende tabel geeft een overzicht van de units in de Java-code met de meeste parameters.

© 2011 Software Improvement Group

Unit	No. of parameters
ReportConfigurationImpl.$constructor(ReportTemplateEnum, ReportOutputFormatEnum,File,Date,ReportNameComponents,boolean,String, boolean,IPDFOpener,ExportEml,boolean,boolean,MultiReportConfiguration)	13
ReportConfigurationImpl.$constructor(ReportTemplateEnum, ReportOutputFormatEnum,File,Date,ReportNameComponents,boolean,boolean, IPDFOpener,ExportEml,boolean,boolean,MultiReportConfiguration)	12
ReportConfigurationImpl.$constructor(ReportTemplateEnum, ReportOutputFormatEnum,File,Date,ReportNameComponents,boolean,boolean, IPDFOpener,ExportEml,boolean,MultiReportConfiguration)	11
RG520Helper.appendListAndResults(Element,String,Set,Liste,Integer,Integer, Map,Set,List,Map)	10
AnwenderHandling.createOrModifyAnwender(AnwContext,String,String, String,Collection,String,int,boolean,boolean)	9
AnwenderHandlingBean.createOrModifyAnwender(AnwContext,String,String, String,Collection,String,int,boolean,boolean)	9
CandidateResult.$constructor(Candidate,P2List,int,long,Elected,DrawnByLot, int,boolean,P2List)	9
ElectionResultImpl.candidateResult(P2List,Candidate,int,int,boolean,boolean, boolean,boolean,boolean)	9
GeneralSeatDistributor.$constructor(DrawingLotsCallback,P,long,Map,Map, GsdaParameters,ElectionResult,AnomalyFactory,Distribution)	9

B.9 Module coupling

Module coupling is een maat voor de hoeveelheid code die afhankelijk is van een bepaald stuk code. Code die vanaf veel verschillende plekken wordt aangeroepen is lastig wijzigbaar, omdat eventuele aanpassingen aan deze units ongewenste effecten kunnen hebben op andere delen van het systeem. Er zal veel code getest moeten worden als deze code aangepast is.

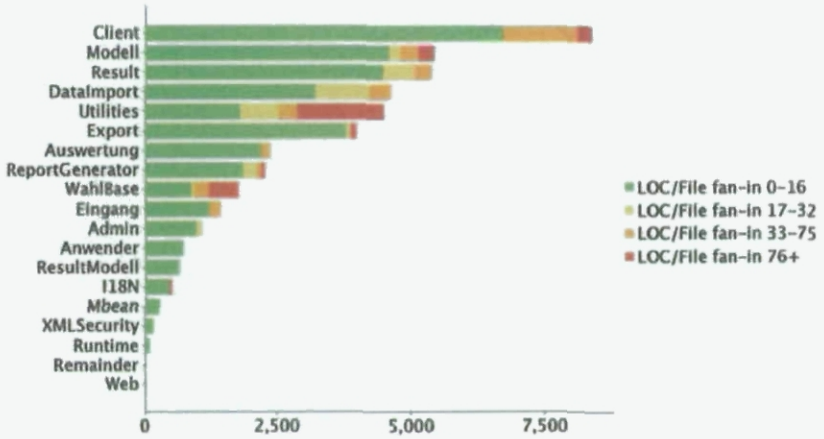
Module coupling wordt gemeten als het aantal plaatsen van waaruit een bestand wordt aangeroepen. Dit aantal aanroepen naar een bestand wordt de fan-in genoemd. De fan-in wordt verlaagd door het aantal aanroepen te reduceren, bijv. door de bestanden in kwestie te splitsen, of door hun implementatie te verbergen achter een interface.

De verschillende niveaus en bijbehorende risico's zijn weergegeven in de volgende tabel.

Fan-in op bestandsniveau	Risico-inschatting
1-16	geen risico (eenvoudig aanpasbaar)
17-32	laag risico
33-75	medium risico
76+	hoog risico

De volgende figuur toont de verdeling van deze niveaus in de Java-code.

© 2011 Software Improvement Group



De volgende tabel geeft een overzicht van de bestanden in de Java-code met de hoogste fan-in.

File	Fan-in	Lines of code
ModelImpl.java	403	39
InitGuiCommand.java	297	70
WahlInfo.java	270	367
Messages.java	249	34
Log4J.java	160	477
BMPBeanBase.java	157	481
DatabaseAccess.java	149	54
DBABase.java	145	411
XMLHelper.java	106	67
GUICommand.java	105	196

B.10 Foutafhandeling

Ieder softwaresysteem kan tijdens de operatie onverwachte situaties tegenkomen, bijv. tijdens het niet kunnen toewijzen van geheugen of bij een falende poging om een tekst file van het filesysteem in te lezen. Om dergelijke situaties op correcte wijze af te handelen kent software 'foutafhandelingsconstructies' zodat er een fout aan de eindgebruiker getoond kan worden en er bijv. een omschrijving van de foutsituatie in een error-log geschreven kan worden.

Het juiste gebruik van foutafhandelingsconstructies is een indicatie van de professionaliteit van het ontwikkelteam. Het mechanisme stelt het beheerteam in staat om te analyseren welke uitzonderlijke situatie opgetreden is, zodat de fout in de toekomst mogelijk voorkomen kan worden. Een juist gebruik van foutafhandelingsconstructies vergroot de betrouwbaarheid van het systeem.

De volgende tabel toont metingen aan de foutafhandeling in OSV 4 en 5. `Catch blocks' zijn voorkomens van foutafhandeling. Wanneer een catch block leeg is of alleen een generieke exception opvangt geldt dit als respectievelijk een `empty catch' of `illegal catch'.

© 2011 Software Improvement Group

Module	Catch blocks	Illegal catches	Empty catches
Admin	46	17	0
Anwender	25	11	0
Auswertung	27	11	0
Client	119	77	0
DataImport	95	29	1
Eingang	11	3	1
Export	37	1	0
I18N	2	1	0
Mbean	7	5	0
Modell	123	1	0
Remainder	0	0	0
ReportGenerator	17	7	0
Result	3	1	0
ResultModell	0	0	0
Runtime	0	0	0
Utilities	103	49	1
WahlBase	12	1	0
Web	0	0	0
Total	**631**	**215**	**3**

© 2011 Software Improvement Group

C. Disclaimer

De Software Improvement Group kan niet garanderen dat de interpretatie van de bevindingen in dit rapport foutloos is. Het is mogelijk dat verdere gesprekken met de onderhoudsmedewerkers van de systemen alsmede een verdere analyse van de broncode, tot een andere interpretatie van de bevindingen kunnen leiden dan die in dit rapport is beschreven.

© 2011 Software Improvement Group

