ANALYSING SOUP

BAS BRUNINK

Een weg naar veiligere software

Software Engineering
FDMCI
Hogeschool van Amsterdam

Februari 2022 – version 0.1

SAMENVATTING

Short summary of the contents. . . a great guide by Kent Beck how to write good abstracts can be found here:

https://plg.uwaterloo.ca/~migod/research/beckOOPSLA.html

INHOUDSOPGAVE

LIJST VAN TABELLEN

LISTINGS

ACRONIEMEN

API Application Programming Interface

CEO Chief Executive Officer

CFO Chief Financial Officer

CTO Chief Technology Officer

OSS Open Source Software

SOUP Software of Unkown Pedigree / Provenance

UML Unified Modeling Language

1

INLEIDING INLEIDING.TEX

Het document dat voor u ligt is een resultaat van een onderzoek en product oplevering als afstudeeropdracht door Bas Brunink voor het bedrijf Eaglescience. Het zal het process beschrijven die ik gelopen heb om een module te schrijven die automatisch een SOUP analyse doet op zowel bestaande als nieuwe projecten.

1.1 LEESWIJZER

Deze scriptie neemt de lezer mee door het verloop van het project van idee tot implementatie. Met als einde een resultaat beschrijven en een prognose in de verbetering gezien de verwachting is dat er niet direct een significante verbetering te zien is.

Deel I

OPDRACHT

2

Het hier beschreven onderzoek en daarbij behorende applicatie is geschreven in opdracht van Eaglescience wat gevestigd is in Amsterdam Sloterdijk. Eaglescience ontwikkeld complexe software op projectbasis voor diverse klanten vaak met een wetenschappelijke inslag. Het bedrijf telt +/- 20 medewerkers waarvan 75% ontwikkelaar/designer zijn en de resterende 25% een support rol bekleden (project managers finance manager, Quality manager) Aan het hoofd van Eaglescience staat een CEO, Marc Grootjen, CTO Bas Breier, CFO Wender van Mansvelt. Naast het ontwikkelen van nieuwe software biedt Eaglescience ook de mogelijkheid om zorg te dragen voor de eventuele hosting van het opgeleverde product. Hiermee kan Eaglescience nog beter garanderen dat de geboden kwaliteit in de software gewaarborgd blijft tijdens de levensduur van de software.

2.1 MISSIE

De missie van Eaglescience is het bedienen van onze partners door een ontwerp, ontwikkeling en service te bieden op het gebied van op maat gemaakte IT oplossingen . Om dit te kunnen bewerkstelligen heeft Eaglescience goed opgeleide IT professionals in dienst die zichzelf continue ontwikkelen op de "cutting edge" van IT technologie. De hoofd competenties van de medewerkers zijn: innovatief, intelligent, klant georienteeërd, flexibel en ambitieus.

2.2 VISIE

Eaglescience streeft er als innovatief IT bedrijf naar om software te ontwikkelen als een Business-to-Business dienst. Met onze technische vaardigheden bouwen we veilige en hoogwaardige software die bijdraagt aan een betere wereld. Omdat we agile werken, leveren we precies wat nodig is, niets meer en niets minder. Wij helpen onze klanten zoeken naar een langdurige betrokkenheid en samenwerking op basis van zowel vertrouwen als wederzijds respect. Omdat elke vraag uniek is, ontwikkeld Eaglescience op maar gemaakte en innovatieve software. We zijn van plan deel uit te maken van het hele proces van het formuleren van een idee tot het lanceren van het product en het waarborgen van de productie levenscyclus. Onze belangrijkste succesfactor zijn de mensen, die zich continu ontwikkelen door met de nieuwste technieken te werken op diverse projecten. Wij streven naar een optimale balans tussen werk en privé. Dit geeft onze medewerkers veel vrijheid, maar vereist zelfdiscipline en verantwoordelijkheid.

2.3 STRATEGIE

strategische thema's

Eaglescience levert de visie via vier strategische thema's:

- Maatschappelijke verantwoordelijkheid
- Persoonlijke groei
- Tevredenheid

We streven ernaar om veilige en hoogwaardige software diensten te leveren die waarde toevoegen aan onze samenleving. We streven naar een bedrijfscultuur waarin alle collega's hun talenten kunnen laten groeien. We hebben een ongecompliceerd werkethos: we richten ons op resultaten van hoge kwaliteit, maar met een gezonde balans tussen werk en privé en voldoende tijd voor leuke en sociale evenementen. Eaglescience verwacht van alle medewerkers dat zij hun handelen baseren op vier kwaliteitsprincipes:

- Meld situaties die niet voldoen aan onze interne procedures

- Evalueer risico's wanneer grote veranderingen worden verwacht
- Help en daag elkaar uit
- Kennis behouden over compliancy en kwaliteitsmanagement

2.4 RELEVANTE EN ACTUELE ONTWIKKELINGEN BINNEN EAGLESCIENCE

Eaglescience is aan het groeien, zowel in het aantal projecten waar aan gewerkt wordt als het aantal medewerkers. Daarnaast worden de diensten die Eaglescience aanbied ook uitgebreid. Waarbij het hosten van de ontwikkelde applicaties steeds meer wordt aangeboden. Door deze inzet ligt de verantwoordelijk niet alleen bij het leveren van een veilige en hoogwaardige software maar het leveren van service waarbij de applicaties in een veilige omgeving worden aangeboden. Mede door de groei van het bedrijf maar zeker ook de diensten die aangeboden wordt is het zeer relevant om taken die geautomatiseerd kunnen worden te automatiseren.

3

OPDRACHT –CHAPTER02.TEX

Tegenwoordig zijn software-bibliotheken niet meer weg te denken in het software ontwikkelproces van nu. Bibliotheken geven ontwikkelaars de mogelijkheid code her te gebruiken in meerdere projecten om zo efficiënter te kunnen ontwikkelen. Dit helpt weer mee om een snelle Time-To-Market te behalen. Bibliotheken kunnen door bedrijven zelf geschreven worden, in het geval van EagseScience is dit Arches, of worden overgenomen van andere bedrijven/ instellingen. Zelfs Arches is afhankelijk van een aantal bibliotheken die niet ontwikkeld zijn door Eaglescience. Dus ontkom je er tegenwoordig niet aan om bibliotheken te gebruiken waarvan je de afkomst niet geheel kan herleiden.
Deze bibliotheken wordt Software of Unknown Provenance/Pedigree (SOUP) genoemd. Door het gebruik van SOUP bibliotheken kan er een aannemelijk risico worden genomen op het gebied van kwetsbaarheden. Om inzicht te krijgen in deze kwetsbaarheden en daarmee dus mogelijk veiligheidsissues dient er een SOUP analyse gedaan worden. Binnen Eaglescience wordt het belang gezien om deze analyse te doen en is daarom op zoek naar een efficiënte en mogelijk geautomatiseerde manier voor het uitvoeren van een dergelijke analyse om zo de veiligheid van de ontwikkelde applicaties te waarborgen zonder afbreuk te doen aan kwaliteit.

3.1 OPDRACHT VANUIT EAGLESCIENCE

Vanuit de CTO is de wens ontstaan om een gestructureerde methode te ontwikkelen waarbij er automatisch periodiek een SOUP analyse gedaan wordt op bestaande en nieuwe projecten. Het uiteindelijke resultaat moet zijn dat er een module wordt toegevoegd aan de reeds bestaande portal van Eaglescience waarbij project verantwoordelijken inzicht kunnen verkrijgen in de kwetsbaarheden die in een project aanwezig kunnen zijn door het gebruik van externe bibliotheken.

3.1.1 *Eisen aan de opdracht*

Vanuit Eaglescience zijn er een aantal eisen gesteld waaraan het eindproduct moet voldoen. Als er aan deze eisen is voldaan dan is er voor Eaglescience een waardevol product wat men dan ook in gebruik kan nemen. Daarnaast zijn er een aantal oplever eisen die gehaald dienen te worden om de kwaliteit te waarborgen.

functionele eisen

- De module dient eenvoudig te worden gebruikt in de huidige CI/CD pipeline voor bestaande en nieuwe projecten
- De module dient gebruik te maken van de bestaande ++huidige++ projectstructuur van het portal
- De module dient ondersteuning te bieden voor meerdere omgevingen(OTAP)
- De module dient met een instelbaar interval de analyse uit te voeren
- De module op project en omgeving niveau te rapporteren over bekende kwetsbaarheden
- De module dient kwetsbaarheden op minimaal drie niveau's in te schalen (kritisch, gemiddeld en laag)
- De module dient ondersteuning te bieden voor het instellen van quality gates ten aanzien van ieder niveau, per project, per omgeving
- De module wordt ontwikkeld in Angular en Play(scala), overeenkomstig bestaande portal modules

kwaliteitseisen

- De module voldoet aan de geldende kwaliteitsnormen binnen Eaglescience, minimaal meetbaar door:
  - test coverage > 70%
  - onderdeel van de bestaande CI/CD voor het Eaglescience Portal
- Geschreven code is gereviewd door een Eaglescience ontwikkelaar
- In de module zijn gescheiden componenten: Frontend, Backend, API onafhankelijk en goed gedocumenteerd.
- Voor de API documentatie wordt gebruik gemaakt van swagger.

3.1.2 *Deliverables*

Vanuit de CTO zijn er naast de functionele eisen ook eisen gesteld aan de oplevering:

- Geïntregreerde en aantoonbaar werkende module
- De code van de module in Eaglsescience GitLab
- API documentatie (middels swagger)
- Een handleiding hoe de module gebruikt dient te worden
- Eventuele aanvullende deliverables vanuit de HvA

3.2 OPDRACHT FASEN

Om de hierboven beschreven opdracht zo goed als mogelijk uit te voeren dient er eerst een onderzoek gedaan worden naar mogelijk beschikbare oplossingen van derden. mochten deze er niet zijn dan wordt er over gegaan naar een onderzoek naar een mogelijke oplossing om deze inhouse te gaan bouwen. Als hiervan de resultaten bekend zijn wordt overgegaan op het daadwerkelijk implementeren van een oplossing dan al niet van een derde partij.

3.2.1 *Fase 1: Onderzoek*

Het onderzoek dat gedaan moet worden is tweeledig: ten eerste dient er een marktonderzoek gedaan te worden om te kijken of er een bestaande oplossing is die direct al dan niet met enige aanpassing geïntegreerd kan worden in de huidige pipeline. Hier dient gelet te worden op de eisen die gesteld zijn vanuit de CTO maar ook de onderhoudbaarheid van de oplossing zelf. Naast het marktonderzoek dient er ook een literatuur studie gedaan te worden om de begrippen en onderwerpen die samenhangen met het begrip SOUP en de analyse van vulnerabilities. Met als uitgangspunt een beter begrip te vormen om een eigen module te kunnen schrijven.

3.2.2 *Fase 2: Oplevering SOUP analyse module*

De fase kan snel gaan als er een bestaande module bestaat die voldoet aan de eisen, anders moet er een module worden geschreven die aan de eisen voldoet.

3.3 PLAN VAN AANPAK

Het plan is als volgt:

1. LiteratuurOnderzoek
   Wat gebeurt er als ik hier text plaats

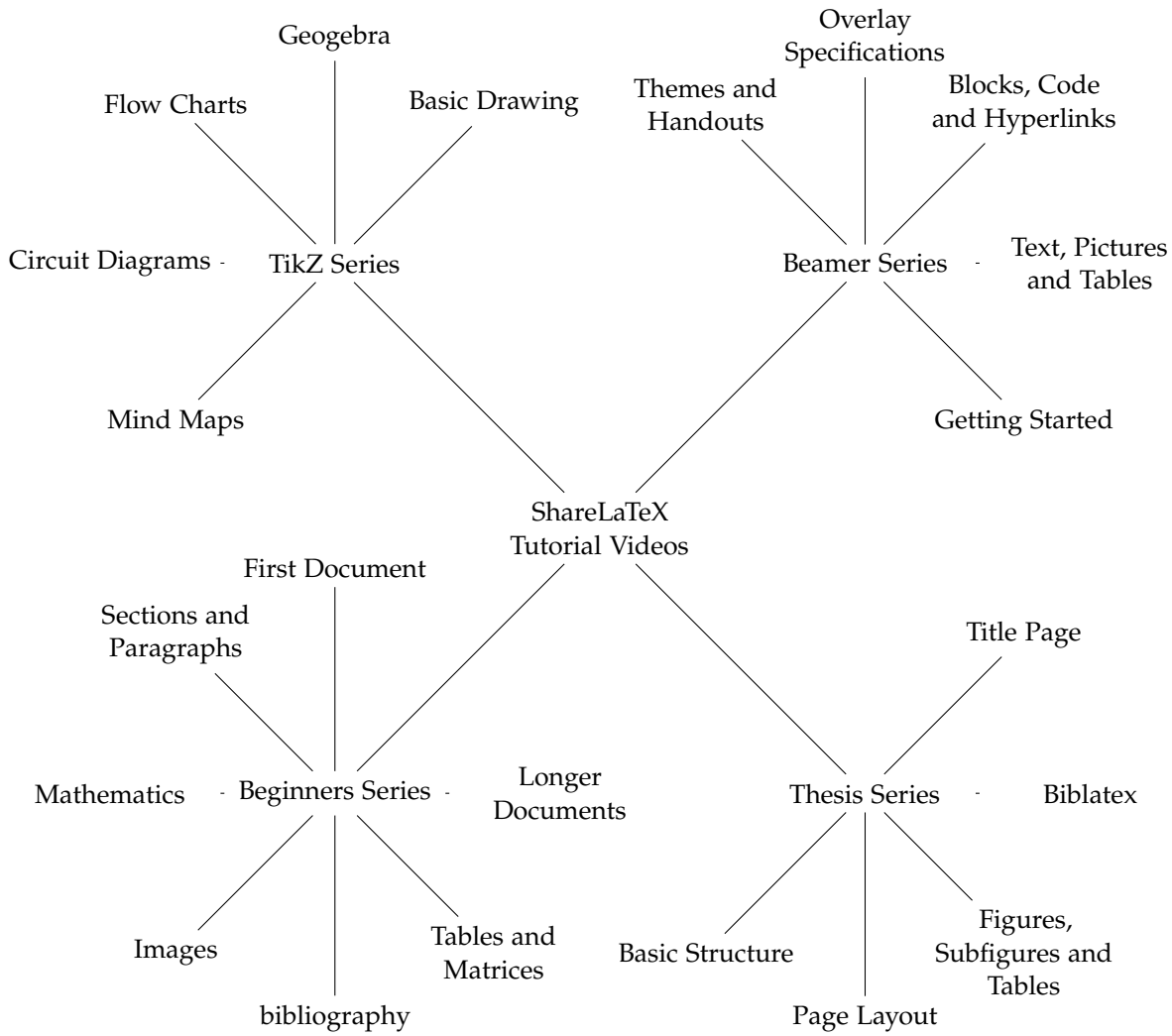
2. Markt onderzoek
3. Resultaat onderzoek
4. ontwerp implementatie
5. Ontwikkeling implementatie
6. Deploy implementatie

3.4 PLANNING

De planning zal als volgt zijn:

- (juli 2021) opzetten van verslag/ literatuur studie
- (Aug 2021) Marktonderzoek
- (Sept 2021) Architectuur onderzoek opzetten van architectuur voor module
- (Okt 2021) Opbouw Artchitectuur en frontend.

3.5 MINDMAP TEST



Deel II

ONDERZOEK

4

INLEIDING –CHAPTER03.TEX

Zoals in de opdracht duidelijk is gemaakt maakt een SOUP analyse duidelijk welke mogelijke kwetsbaatheden er zich kunnen bevinden in de ontwikkelde software. Deze kwetsbaarheden kunnen onder andere door het gebruik van extern ontwikkelde bibliotheken worden geintroduceerd.
Voordat er een methode en module ontwikkeld kan worden dient er onderzoek gedaan worden naar het doen van een SOUP analyse

Binnen dit onderzoek worden een aantal vragen beantwoorde die duidelijkheid moeten geven over een aantal zaken alvoorns met de de daadwerkelijke implementatie te beginnen.

- Wat is de Definitie van Open Source Software (OSS)
- Wat is de Definitie vaan Software of unkown Pedigree
-

5

Voordat we daadwerkelijk onderzoek kunnen doen naar de beste methode voor een soup analyse moeten er eerst een aantal definities worden verduidelijkt.

5.1 WAT IS OPEN SOURCE SOFTWARE

Er is veel debat over wat nou precies Open Source Software is. Met als resultaat dat er twee verschillende interpretaties zijn: **free software** en **open-source software**. Waarbij free software niet gaat over de kosten van de software maar eerder over wat er met de software gedaan mag worden. volgens GNU free software foundation is de defunitiue van free software :

Free software is a matter of liberty, not price. To understand the concept, you should think of f̈reeäs in "free speech ,, not as in f̈ree beer.. Free software is a matter of the users freedom to run, copy, distribute, study, change and improve the software.
...
In order for the freedoms to make changes, and to publish improved versions, to be meaningful, you must have access to the source code of the program. Therefore, accessibility of source code is a necessary condition for free software.

De definitie van OpenSource software is:
Open-source software (OSS) is computer software that is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software and its source code to anyone and for any purpose. Open-source software may be developed in a collaborative public manner. Open-source software is a prominent example of open collaboration.
Zo op het oog lijken de vormen van software gelijk het verschil zit hem in de manier waarom de software geschreven is. Bij OSS wordt de software veelal door een groep mensen ontwikkeld en bij Freesoftware niet. Hierdoor is het niet altijd duidelijk hoe de software is opgebouwd en wie verantwoordelijk is voor welke delen. Hierdoor kan er niet worden gezien of er daadwerkelijk kwetsbaarheden zijn.

5.2 WAT IS SOUP?

De definitie van SOUP luidt: "SOUP stands for software of unknown (or uncertain) pedigree (or provenance), and is a term often used in the context of safety-critical and safety-involved systems such as medical software. SOUP is software that has not been developed with a known software development process or methodology, or which has unknown or no safety-related properties."

Vanuit deze definitie kunnen we concluderen dat er software gebruikt kan worden waarvan niet zeker van is met welk proces het ontwikkeld is. En welke veiligheids gerelateerde eigenschappen deze software heeft. Hier door is het dus niet te garanderen dat de ontwikkelde software die gebruik maakt van OSS geen kwetsbaarheden bevat.

Dit vraagstuk is al een tijd gaan en er zijn een aantal instanties die zich wereldwijd bezighouden met het opslaan van bekende kwetsbaarheden in veel gebruikte software. Een aantal van deze instanties zijn:

- https://nvd.nist.gov/ **National Vulnerability Database** Een amerikaanse database waar ...

- https://vuldb.com/? **VULDB** Een communit driven vulnerability database.

5.3 HOOFDVRAAG EN DEELVRAGEN

De hoofdvraag die het onderzoek moet beantwoorden is: "Wat is de beste manier om een SOUP analyse te integreren in de huidige ontwikkel straat, en hoe kunnen we ervoor zorgen dat onze software veiliger wordt?"

De hoofdvraag luid als volgt: "Hoe kunnen we zien of blibliotheken van buiten af op het moment van checken geen bekende kwetsbaarheden bevat?"

Daaruit volgen een aantal deelvragen:

1. Hoe wordt er op het dit moment een SOUP analyse uitgevoerd door Eaglescience en wat zijn de resource die gebruikt worden?
2. Welke methoden zijn er buiten Eaglescience om te zien of een bibliotheek kwetsbaarheden bevat?
3. Wat is de methode die we nu gebruiken?
4. Wie zijn de uiteindelijke (eind)gebruikers?
5. Hoe ziet de portal er op dit moment uit?
6.

5.4 ONDERZOEKS MODEL

Content

5.5 SUBSECTION TITLE

Content

5.6 SECTION TITLE

Content

Deel III

ONTWIKKELING VAN SOUP MODULE

Deel IV

APPENDIX

You can put some informational part preamble text here. Illo principalmente su nos. Non message *occidental* angloromanic da. Debitas effortio simplificate sia se, auxiliar summarios da que, se avantiate publicationes via. Pan in terra summarios, capital interlingua se que. Al via multo esser specimen, campo responder que da. Le usate medical addresses pro, europa origine sanctificate nos se.

A

APPENDIX TEST

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

A.1 APPENDIX SECTION TEST

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

More dummy text

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

A.2 ANOTHER APPENDIX SECTION TEST

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

There is also a useless Pascal listing below: Listing 1.

LABITUR BONORUM PRI NO	QUE VISTA	HUMAN
fastidii ea ius	germano	demonstratea
suscipit instructior	titulo	personas
quaestio philosophia	facto	demonstrated

Tabel 1: Autem usu id.

Listing 1: A floating example (`listings` manual)

```
1 for i:=maxint downto 0 do
  begin
  { do nothing }
  end;
```

DECLARATION

Put your declaration here.

Amsterdam , Februari 2022

Bas Brunink

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography *"The Elements of Typographic Style"*. `classicthesis` is available for both LaTeX and LyX:

https://bitbucket.org/amiede/classicthesis/

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured here:

http://postcards.miede.de/