



PROJECTVERSLAG_

OOSE-Project eindverslag

Tobias Feld (649385)

9 juni 2022

INHOUDSOPGAVE

INLEIDING	3
1 KWALITEIT DEELPRODUCTEN.....	4
1.1 Plan van aanpak	4
1.2 SDD	5
1.3 SRS.....	6
1.4 Codekwaliteit.....	7
1.5 Unit tests	8
1.6 Testrapportage.....	8
2 PROJECTMETHODE	9
2.1 Scrum.....	9
2.2 Ervaringen.....	9
3 ROLLEN	10
3.1 Rolbeschrijving.....	10
3.1.1 Product owner by proxy	10
3.2 Resultaat.....	10
4 COMPETENTIES	11
4.1 OOSE P-04	11
4.2 OOSE P-05	12
4.3 OOSE P-07	13
5 LEERDOELEN.....	14
5.1 Leerdoel 1	14
5.2 Leerdoel 2	15
6 CONCLUSIE	16

INLEIDING

Bij dit project hebben wij als groep een HR-Portaal moeten maken voor JDI smart web applications. Het moet een portaal worden waar JDI verlof, werkplekbeheer en declaraties in kon bijhouden.

Het was hierbij nodig om Spring boot & Vue gebruiken, beide technieken waar ik zelf niet bekend mee was. Dit was dus een mooie uitdaging om mijn al bestaande skills te recyclen. Ook was dit een extra uitdaging omdat ik verder geen OOSE vakken heb gevolgd vanwege persoonlijke omstandigheden.

In dit project heb ik geprobeerd te werken aan een goede sfeer in het projectgroep, en een goed eindresultaat in code en tevredenheid van de opdrachtgever. In dit document oordeel ik over zowel mijn eigen inbreng in de code & documenten als mijn werk tijdens het project.

1 KWALITEIT DEELPRODUCTEN

1.1 Plan van aanpak

Kwaliteitseis	Beoordeling	Toelichting
Voldoet aan AIM-Controlekaart	Goed	De vereiste gegevens zijn aanwezig. Het document is gecontroleerd op spellingsfouten en komt professioneel over. Het document heeft geen voorblad of studentgegevens, maar dit is logisch omdat een confluence document is.
Overdracht van informatie	Goed	Alle informatie die de opdrachtgever nodig heeft, staat in het document. Informatie staat goed gegroepeerd en is duidelijk te vinden op basis van de hoofdstukken.
Meetbaarheid eisen opleverproducten	Neutraal	De meeste eisen zijn smart meetbaar, al zijn er nog wat vagere eisen die misschien wat verhelderd hadden kunnen worden.
Planning is duidelijk om misverstanden in de planning te voorkomen	Goed	De projectgrenzen zijn streng aangegeven, hierbij is het voor zowel de opdrachtgever als voor ons fijn dat er duidelijk is wat gepland staat over 8 weken en hoe wij onze tijd verdelen daarbinnen.

Figuur 1

1.2 SDD

In het SDD zou het product & gemaakte afspraken duidelijker moeten worden, en al staat er wel veel informatie in, vind ik het persoonlijk wel lastig om snel te navigeren als ik iets nodig heb. Dit heeft niet per se te maken met hoe wij het hebben opgesteld, maar omdat de documenten in het nederlands is geschreven en de kopjes dan weer in het engels. Het was ons niet duidelijk of dit vertaald mocht worden.

Kwaliteitseis	Beoordeling	Toelichting
Voldoet aan de AIM-controle kaart	Goed	Het document is gecontroleerd op spellingsfouten. Het bevat een bronnenlijst & inhoudsopgave.
Het architectureel overzicht is logisch opgesteld	Voldoende	Het is duidelijk wat de relaties zijn tussen de verschillende servers en componenten. Al had de specifieke data die werd uitgewisselt misschien iets beter kunnen worden uitgelegd
De ontwerpkeuzes zijn logisch opgesteld	Goed	De onderdelen zijn per sub-systeem opgesteld, en er is uitgelegd hoe deze keuzes tot stand zijn gekomen.
De deploy diagrammen zijn logisch opgesteld	Goed	Het diagram is duidelijk te lezen, en afwijkingen van het diagram zijn verduidelijkt
Het document is compleet en volledig	Matig	Voor de back-end ligt er voldoende sequence diagrammen. Voor de front-end mist er echter nog een groot onderdeel
Database ontwerpen zijn compleet en uitgelicht	Goed	Tabellen zijn beschreven, er is een compleet PDM en CDM en kardinaliteit is goed aangegeven.

1.3 SRS

Het Srs is naar mijn mening logisch ingedeeld en is goed te gebruiken om door naar te verwijzen vanuit andere documenten. Usecase diagrammen zijn logisch ingedeeld en goed te volgen door externe partijen. Ik zou dit document prima kunnen volgen om de juiste domein informatie

Kwaliteitseis	Beoordeling	Toelichting
SRS is duidelijk uit te lezen voor een externe partij	Goed	Het document is goed gegroepeerd en beschrijft usecases goed. Ook zijn de user story's en schetsen goed te begrijpen
Voldoet aan de AIM-controle kaart	Goed	Het hele document is wel gecontroleerd op spellingsfouten, maar mist een voorblad i.v.m. confluence.
Het is duidelijk per usecase welke scenario's gebruikt kunnen worden	Goed	Naar mijn mening is dit prima uitgevoerd, omdat er per usecase 1 of meerdere extensies met alternatieve flows zijn toegevoegd.
Alle usecases zijn fully dressed	Goed	Alle usecases zijn compleet uitgewerkt. Hierbij is een uitzondering gemaakt voor CRUD usecases, wat het in mijn ogen juist duidelijker maakt. Omdat je hier geen overbodige info in hebt staan.
Het domein model is compleet	Goed	Het domeinmodel is compleet, en goed uitgelicht in de glossary. Hierdoor is het ook goed te lezen voor externe partijen
User interface sketches zijn compleet en duidelijk	Goed	De interface schetsen laten een goede mix van meerdere stukken zien. Ook is er een link naar de overige ontwerpen.

1.4 Codekwaliteit

Aangezien ik gezien mijn eerdere ervaringen voornamelijk ben ingezet in de front-end van de applicatie, heb ik voornamelijk daar naar gekeken.

Ik ben tevreden met wat we hebben neergezet tijdens ons project, maar omdat we allemaal niet heel erg thuis waren in de gevraagde programmeertool (Vue.js), zijn er wel meerdere beginnersfouten gemaakt. Onze projectstructuur is niet heel erg logisch, en onze componenten zijn veel te groot, waardoor ze extreem lastig te unit-testen of te herbruiken zijn. Alsnog vind ik dat de logica die in de componenten wordt geschreven prima uitgevoerd.

Kwaliteitseis	Beoordeling	Toelichting
Code is goed uit te lezen voor een externe partij	Onvoldoende	Code bevat vaak geen toelichtende documentatie en is vrij groot per component.
Traceable naar specifieke constraints & requirements	Voldoende	Code is wel opgedeeld in de verschillende usecases. Code is grotendeels gemerged via branches gemaakt in Jira.
Bevat geen bugs die geschreven usecases blokkeren	Goed	Volgende de uitgevoerde integratie-testen is dit niet het geval.
Code gebruikt geen magic strings / numbers	Goed	Alle speciale url's en variables staan in een helpers bestand om herbruikbaarheid toe te laten.

1.5 Unit tests

De unit-tests voor de deelproducten zijn goed geschreven, en handelen de situaties zo goed mogelijk af in mijn ogen. Wel is er bij zowel de front als backend besloten bepaalde items niet te testen. Omdat dit in de praktijk niet veel toevoegde, of omdat de code al gedekt zou zijn door integratie-tests. In de front-end had dit wel beter gekund als we onze producten logischer hadden ingedeeld, maar omdat we hier weinig ervaring in hadden konden we daar niet heel veel in veranderen.

1.6 Testrapportage

Het testrapport geeft alle methodes aan die worden aangeroepen, en welke tests hiervoor gebruikt zijn. Ook geeft het aan hoe de integratie-testen zijn uitgevoerd, welke stappen er gevolgd zijn en wat het resultaat is.

Kwaliteitseis	Beoordeling	Toelichting
Testrapportage bevat alle unit-tests	Goed	De unit-tests zijn compleet beschreven, inclusief methodeheaders.
Testrapportage bevat alle resultaten	Goed	Het testrapport bevat resultaten, en opmerkingen als iets afwijkt.
Alle testen slagen	Goed	Alle geschreven testen van zowel de front-end als de back-end slagen.
Alle testen testen een uniek deel.	Goed	De testen lopen niet vaker over lijnen dan nodig is.
Er is een 80% of hoger line coverage over de belanghebbende bestanden	Goed	De testen in de backend hebben een coverage van 85% (services), de front-end heeft een coverage van 90% (tools)

2 PROJECTMETHODE

2.1 Scrum

Wij gebruikten voor dit project de methode Scrum. Een model waarin je in sprints werkt, waarbij je meerder ceremonies hebt, bijvoorbeeld een daily standup, sprint reviews en sprint retrospectives

2.2 Ervaringen

Dit is niet het eerste project wat ik heb gedaan met Scrum, dus ik was er al wel mee bekend. Ik vind scrum meestal goed werken, omdat het goed de grenzen aan werk aangeeft, en het voor iedereen duidelijk maakt wat er gedaan moest worden.

Het was wel nodig om nog een extra mid-daily standup erbij te nemen, om te voorkomen dat er in de middag iedereen een beetje half werk deed wat tegen elkaar aan liep. Doormiddel van deze MDSU's kwam de structuur weer ten goede.

Deze keer heb ik een andere rol op mezelf genomen. Ik heb in het verleden altijd standups gehouden en scrum master geweest. Deze keer heb ik gewoon meegedaan als lid van het Development team.

Wel heb ik dit project wat retro's mogen houden. Ik vond het vooral in het begin lastig, maar ik denk dat het wel veel beter ging tegen de laatste keer. Ik vind het nog steeds lastig om op het juiste moment mijn mening te geven, zonder dat dit mijn rol als retroleider verzwakt. Hier wil ik in latere projecten nog wel aan werken.

3 ROLLEN

3.1 Rolbeschrijving

3.1.1 Product owner by proxy

Als product owner by proxy was ik verantwoordelijk voor de communicatie met de product owner. Ook moest ik ervoor zorgen dat de eisen van de product owner verduidelijkt worden, zodat de rest van het team op de gewenste koers blijft werken.

3.2 Resultaat

In dit project had ik de rol van product owner by proxy. Ik heb hier naar mijn mening te weinig voor gedaan in het eerste deel van dit project. Ik vroeg te laat of iets was zoals de opdrachtgever dit verwachtte, waardoor we tussen sprint 1 & sprint 2 werk compleet moesten aanpassen.

Vanaf de tweede sprint heb ik een stuk actiever de opdrachtgever gemaild met vragen over bepaalde wensen of eisen. Hierdoor was het project duidelijker voor ons, en de opdrachtgever wist beter waar hij aan toe was.

Ik heb in mijn rol als PO by proxy meerdere argumenten gevoerd binnen het groepje om aan te geven waarom bepaalde onderdelen niet zouden voldoen aan de eisen van de opdrachtgever. Als we daar samen niet uit kwamen dan mailde ik de opdrachtgever voor verduidelijking.

4 COMPETENTIES

4.1 OOSE P-04

De student ontwerpt de software van een systeem en documenteert deze onder andere met behulp van UML diagrammen en decision templates in een Software Design Specification (SDD)

Ik heb in overleg met Thijmen voornamelijk bijgedragen aan dit onderdeel. Vaak ter verheldering van onderdelen en deze het beste konden worden omgezet in de juiste diagrammen. Ik heb geholpen met logische data types voor bepaalde beslissingen en ik heb geholpen met logische keuzes in het database ontwerp.

Hierbij heb ik wel vaak ideeën vermeld, maar niet zelf geïmplementeerd. Ik denk dat ik te snel door wilde, en daardoor soms te weinig heb gedocumenteerd waar dat nodig was, hierdoor moesten anderen mijn punten later doorvoeren in het document. Hierdoor liep het document soms scheef met hoe het in de praktijk was geïmplementeerd.

4.2 OOSE P-05

De student implementeert een gedistribueerd systeem, evalueert het ontwerp en de realisatie daarvan en zorgt voor traceerbaarheid daartussen en naar de functionele en niet-functionele eisen.

Allereerst hebben wij de usecases vastgesteld voor zover dat mogelijk was binnen de sprint, en is deze toegevoegd aan Jira.

Doormiddel van Jira hebben wij van de usecases “Epics” gemaakt. Hierdoor konden we taken koppelen aan usecases, en dit overzichtelijk houden. Daarnaast zijn er in elke hoofdtaken meerdere subtaken ontstaan, afhankelijk van de hoeveelheid subsystemen die relevant zijn. Ik heb daarnaast de meeste branches vanuit deze Jira issues aangemaakt, waardoor de code in deze branch gekoppeld is aan de usecases en het correcte sub-issue.

4.3 OOSE P-07

De student bewaakt continu de kwaliteit van de software en het proces door o.a. reviews en gestructureerd testen en stuurt waar nodig bij

Ik geef regelmatig opmerkingen als bepaalde onderdelen missen in de op te leveren producten, en stel vragen als dingen onduidelijk zijn. Ook heb ik het proces in onze bitbucket omgeving ingesteld waarin je niet zomaar code kan toevoegen aan het git repository, maar eerst je verandering moet doorvoeren op een aparte branch. Zodra dit klaar is kan je het bij de code toevoegen, maar dit moet eerst nagekeken worden door minstens 2 anderen. Op deze manier wordt de kwaliteit van het product beter bewaakt.

Daarnaast heb ik Jenkins & Sonarqube ingesteld, waarbij tests & lint-regels nogmaals worden gecontroleerd. Dit heeft meerdere malen ervoor gezorgd dat foutieve code vroegtijdig werd opgespoord.

5 LEERDOELEN

5.1 Leerdoel 1

“Andere meningen sneller accepteren en niet altijd mijn mening willen doordrammen”

In mijn vorige projecten heb ik vaak gehad dat ik mijn mening vrij veel wilde doordrammen, en daarmee dus de groep een bepaalde richting op wilde sturen die ik had bepaald. Ik wilde na feedback van mijn vorige projectgroep hierin minder worden. Tijdens de rollenverdeling heb ik dat al gedaan om een rol te nemen die daar minder last van had: Niels pakte de teamleider rol op, en ik pakte de productowner by proxy op. Dit ook omdat de product owner nog steeds wel een mening moest meedelen, maar wel met de product owner in gedachte. Ik was hierin gelimiteerd in hoe erg ik kon doordrammen, omdat het niet altijd mijn mening of eisen waren, maar dat werkte juist erg fijn. Omdat deze eisen erbij hoorde kon ik niet zomaar mijn mening zo vertellen.

Desondanks dat heb ik niet per se alleen goede momenten gehad, ik heb bijvoorbeeld een lange discussie gehad met Gino over een implementatie van een usecase. Gelukkig is er op een gegeven punt besloten met de groep om lange discussies te vermijden, en eerst even na te denken over wat er is gezegd. Op deze manier krijgt iedereen meer tijd om zijn idee op een goede manier uit te kunnen leggen.

5.2 Leerdoel 2

Zorgen dat ik beter hulp geef als ik dingen wel weet, ipv heel erg teruggetrokken zijn

In eerdere projecten heb ik vaak mezelf teruggetrokken tijdens projecten (misschien ook als gevolg van de online-corona situatie), Bij het ISE-project ben ik daar erg op afgerekend omdat ik liever online werkte en wat meer teruggetrokken was. Dat is ook de reden waarom ik in dit project voornam om exclusief op school te werken wanneer dat mogelijk was met de groep. Ondanks dat ik thuis veel betere tools heb, merk ik dat mijn focus over de dag wel beter blijft.

Aangezien ik meestal wel aardig wat algemene kennis heb met programmeren (en in dit gebied, veel front-end development) heb ik geprobeerd om veel actiever hulp en uitleg te geven. Dit is wel gelukt, al ben ik daar misschien iets te ver in gegaan; halverwege het project kwamen er in mijn ogen veel te vragen, waardoor ik het positieve focus-effect als veel negatiever kwam te ervaren.

Gelukkig heb ik dit tijdens de 2^e sprint-retro aangekaart, en zijn de oplossingen die gevonden waren, zoals gewoon op eigen tijd hulp kunnen geven via Discord ontzettend fijn en een stuk rustgevender. Ik denk dat ik hierin een goede mix moet vinden, en ik denk dat dit een mooie stap is voor een volgend project. Dat ik daarmee nog beter meedraai in een projectgroep.

6 CONCLUSIE

Allereerst, ik vond dit een prima project om te doen, de opdracht was leuk, de druk van een externe opdrachtgever en de goeie team-band maakte dit project een leuke uitdaging door al deze factoren. Ook omdat de opdracht goed te doen was en de sfeer in het team goed was, zorgde dit voor een hele fijne leerervaring.

Wel vind ik dat we qua planning goed werk hebben geleverd, en ons ook strak aan de planning gehouden hebben. Als ik kijk naar hoe ik dit project heb gewerkt in vergelijking met het I-Project of ISE-project, dan zie ik hierin veel meer structuur en ervaring.

Kijkend naar competenties vind ik dat ik het misschien te makkelijk heb af laten gaan bij bepaalde documenten, specifiek het SDD. Ik heb wel vaak voorstellen gedaan voor beslissingen doorgegeven¹, maar dan niet nagedacht om dit bij de SDD te zetten. Dit heeft bijvoorbeeld Thijmen achteraf nog moeten toevoegen om het document compleet te maken.

Ik denk dat dit een goed leerdoel is voor het volgende project: Niet al te snel verdwalen in het programmeer-werk dat nodig is, maar mezelf remmen om aan bijbehorende documentatie te werken.

Desondanks vind ik persoonlijk het een geslaagd project.

¹ Zie bijlage "conclusie_sdd_1.png"

OPEN UP
NEW **HAN_** UNIVERSITY
OF APPLIED SCIENCES
HORIZONS.