

# LEESWIJZER

## SOFTWARELICENTIEMODEL



<b>Datum</b>	14-12-2022
<b>Versie</b>	1.0
<b>Status</b>	Final
<b>Auteur</b>	Patrick van Nieuwburg

## Versie

Versie	Datum	Auteur(s)	Wijzigingen	Status
0.1	14-12-2022	PvN	Context en projectaanpak	Afgerond
0.2	15-12-2022	PvN	Leerdoelen beargumenteren	Afgerond
1.0	06-01-2023	PvN	Aanvullen leerdoelen en toevoegen conclusie en reflectie	Afgerond

## Verspreiding

Versie	Datum	Aan
0.2	20-12-2022	Mark Mestrom
1.0	10-01-2023	Jeroen van Rijckevorsel
1.0	10-01-2023	Inleveren portfolio – Mark Melstroom

## Inhoudsopgave

1.	Voorwoord .....	4
2.	Context en opdracht .....	5
3.	Projectaanpak en proces .....	6
4.	Competenties .....	8
4.1.	Analyse .....	8
4.2.	Advies .....	8
4.3.	Ontwerp.....	8
4.4.	Implementatie.....	8
4.5.	Management.....	9
4.6.	Onderzoeken .....	9
4.7.	Communicatie .....	10
4.8.	Leervermogen.....	10
5.	Conclusie.....	11
5.1.	Reflectie.....	11
6.	Verwijzingen .....	13

## 1. Voorwoord

Voordat het laatste semester van start ging, ben ik samen met de productmanager (PM) en de CEO van TBWB aan tafel gaan zitten om te bespreken of er een project is waarbij ik mijn dagelijkse werkzaamheden kan combineren met het afstudeerproject.

De eerste keuze viel op een mobile applicatie die ontwikkeld zou worden voor een bestaande klant van TBWB. Uiteindelijk heeft de klant besloten het project niet aan te nemen en zijn we opnieuw gaan brainstormen naar wat de mogelijkheden zijn.

Bij TBWB zijn we op dit moment bezig met de ontwikkeling van een standaardproduct. Bij de verkoop van onze huidige projecten staan de kosten van software vaak in de schaduw van de hardware implementaties. Daarom hebben we besloten om te kijken wat de impact is als we ervoor kiezen een softwarelicentiemodel te implementeren in ons standaardproduct.

De personen die ik wil bedanken voor het mogelijk maken van deze afstudeeropdracht zijn:

- Bart Friederichs; de CEO van TBWB voor de mogelijkheid om binnen de bedrijfsuren aan mijn afstudeeropdracht te werken.
- Jeroen van Rijckevorsel; de PM van TBWB voor het leveren van de opdracht en het meedenken over de werking en implementatie van de applicatie.
- Mark Mestrom; mijn docentbegeleider van Fontys voor het geven van feedback, het organiseren van de tweewekelijkse meetings en het begeleiden van het proces.

Bovengenoemde samenwerkingen hebben geleid tot het afronden van mijn portfolio dat ik ga toelichten in onderstaande leeswijzer. Dit document dient als leidraad van mijn ontwikkeling het afgelopen semester.

## 2. Context en opdracht

Er is een volledige beschrijving van de opdracht te vinden in het [projectplan](#). In de hieronder geschreven alinea's zal ik de samenvatting van deze beschrijving in het kort toelichten.

Mijn project is uitgevoerd binnen TBWB, een bedrijf dat zich specialiseert in het automatiseren van logistieke processen. Ze helpen bedrijven met de automatisering van orderpickprocessen, integratie van automatische installaties en alles wat met het beheer van het magazijn te maken heeft.

Binnen TBWB verkopen wij de hardware om het magazijn te automatiseren en de software om het magazijn te beheren. TBWB koopt twee softwarepakketten, één voor een warehouse management system (WMS), dit is het systeem dat bijhoudt hoeveel voorraad er van een bepaald artikel aanwezig is en waar deze zich in het magazijn bevindt. De andere is het warehouse control system (WCS). Deze software stelt TBWB in staat het artikel door het magazijn heen te bewegen door gebruik te maken van de geleverde hardware.

Alle projecten zijn nu bijna volledig klantspecifiek. Op dit moment bouwen we aan een standaardproduct, dit is een standaardisatie van het WMS dat we kunnen inzetten bij nieuwe klanten zonder dat er klantspecifieke wijzigingen aan te pas komen. De afgelopen jaren is er een grote switch waarneembaar in het softwarelandschap van het kopen van een stuk software naar de implementatie van licentiemodellen in de software. Het leek ons dan ook een goed idee om te kijken naar wat een softwarelicentiemodel voor TBWB kan betekenen.

Om het project onderzoekend te kunnen uitvoeren heb ik onderstaande hoofdvraag opgesteld.

“Hoe kan een softwarelicentiemodel TBWB helpen om een maandelijkse/jaarlijkse inkomstenstroom te genereren zonder onze klanten in de weg te zitten met hun dagelijkse werkzaamheden.”

Het beantwoorden van bovenstaande hoofdvraag zal resulteren in een minimum viable product (MVP) waarmee ik de uitkomst van deze vraag kan aantonen aan de PM, stakeholders en natuurlijk de beoordelaars van mijn project. Door goed te kijken naar de wensen van alle stakeholders en deze te bundelen en compromissen te sluiten wil ik een applicatie opleveren waar alle stakeholders zich in kunnen vinden.

### 3. Projectaanpak en proces

In dit hoofdstuk wil ik een totaalbeeld geven van de aanpak die ik heb gekozen voor het project en hoe ik tot bepaalde besluiten ben gekomen.

Het project is uitgevoerd door gebruik te maken van de agile/scrum methode. Deze methode geeft het totale project flexibiliteit waardoor het onderzoek, de documentatie en de implementatie synchroon naast elkaar door kunnen lopen. Dit resulteert in een project waarbij eenvoudig kan worden gereageerd op nieuwe inzichten, wensen en eisen zodat er op tijd bijgestuurd kan worden waar nodig.

Bij de start van het onderzoek lag mijn focus voornamelijk op de wensen en eisen van de stakeholders. Ik heb de eerste sprints me dan ook voornamelijk gericht op het principe van licentiemodellen. Wat is een licentiemodel? Welke smaken zijn er? En welk licentiemodel is geschikt voor TBWB? Nadat ik die fase van mijn onderzoek had afgerond, kon ik beginnen met het opstellen van een aantal vragen die ik aan alle stakeholders wilde stellen tijdens een interview.

Mijn project kent een viertal stakeholders, de CEO van het bedrijf, de PM, het hoofd van de afdeling verkoop en het hoofd van de afdeling service. Door aan alle vier de stakeholders dezelfde vragen te stellen in mijn interviews creëer ik voor mezelf de mogelijkheid om alle wensen en eisen van hen naast elkaar te leggen en hier mijn functionele en niet functionele eisen uit te filteren. Een aantal wensen van de stakeholders waren bijvoorbeeld dat het licentiemodel de klant niet mag belemmeren in zijn dagelijkse werkzaamheden en dat de klant zelf de mogelijkheid moet hebben om het aantal gebruikerslicenties te verhogen of te verlagen.

Nadat ik alle eisen heb opgesteld, heb ik een meeting georganiseerd voor alle betrokkenen. Hier heb ik de huidige status van mijn onderzoek besproken en heb ik feedback gevraagd met betrekking tot alle eisen die ik had opgesteld. De stakeholders waren het veelal wel eens over de eisen, echter was er wat discussies tussen de stakeholders die een ander antwoord hadden gegeven op de vragen van het interview. Deze onenigheden heb ik allemaal kunnen weghalen door mijn keuzes te beargumenteren en dit heeft geresulteerd in het feit dat ik verder kon gaan met mijn ontwerp.

De enige feedback die ik heb meegekregen is dat de afdeling verkoop graag wilde weten hoe dit softwaremodel er voor TBWB uit zou zien. De afdelingsleider wilde bijvoorbeeld graag weten welke percentages en bedragen we zouden moeten doorrekenen aan de klant. Ik heb hem na de presentatie aangesproken om te vragen of hij het eens was met het voorstel voor het licentiemodel dat ik heb gekozen en dat leek hem een prima idee. Ik heb hem verteld dat ik na mijn project graag met hem wil meedenken over prijzen en percentages maar dat dit voor nu buiten mijn projectkader valt.

Het ontwerp heb ik gemaakt door gebruik te maken van het C4 model. Dit model biedt de mogelijkheid om steeds dieper in de softwarearchitectuur te duiken op vier verschillende detail levels. De systeemcontext, een container diagram, een component diagram en een code/klassen diagram. Hoe verder ik in de stappen kom, hoe meer ik ga inzoomen op de techniek die gebruikt wordt voor de applicatie.

Het ontwerp van mijn applicatie is terug te vinden in het [ontwerpdocument](#). In het kort bevat mijn applicatie twee delen. Een online deel (licentieservice en licentiedashboard) en een deel wat bij de klant op de server komt te staan (licentiemanager). Deze applicatie bevat drie belangrijke componenten. Een email component om de klant op de hoogte te stellen van zijn licentiestatus. Een certificaat component om een nieuw certificaat te creëren wanneer de gebruikerslicenties worden verhoogd of verlaagd. En een watchdog component wat de verbinding status monitort tussen de licentieservice en de licentiemanager.

Nadat mijn eisen helder waren en het ontwerp was voltooid, kon ik starten met het opstellen van mijn userstories. Ik heb ervoor gezorgd dat al mijn userstories zijn gekoppeld aan mijn eisen en dat er een link ligt met mijn Jira omgeving. Zo voorkom ik dat er eisen worden vergeten en kan ik tracken wat mijn status is met betrekking tot het creëren van mijn MVP. Dit is te bekijken in mijn [analysedocument](#) op pagina veertien en vijftien.

Nadat ik de gehele analyse heb afgerond, ben ik gestart met het ontwerpen van mijn applicatie. Ik ben begonnen met de uitwerking van de licentieservice en het licentie manager dashboard. De opzet van de licentieservice is ook te gebruiken voor de licentiemanager, dus het was van belang dat de initiële code goed in elkaar zat.

Een aantal voorbeelden waar ik tegenaan ben gelopen tijdens het maken van de code zijn de correcte encryptie van het certificaat en hoe ik het certificaat aan de serverkant kan opvragen. Voor de encryptie was er in eerste instantie gekozen om het hele document te encrypten. Zo is er niks voor de klant aan te passen omdat er alleen maar een hashcode te zien is. Maar als er naar de juiste vorm van encryptie gekeken wordt, resulteert dit in het feit dat dit certificaat met een publieke sleutel beveiligd wordt en met een privésleutel wordt uitgelezen. Zo heeft de klant de beschikking over de privésleutel en dat is iets wat niet wenselijk is. Vervolgens is er gekozen voor signing. Zo is het certificaat te lezen en wordt te data samen met de privésleutel tot een hash omgeschreven. Dit certificaat wordt dan bij de klant met de privésleutel ontcijferd en gecontroleerd op eventuele aanpassingen.

Het tweede voorbeeld was dat ik een manier moest vinden om het certificaat bij de licentiemanager te krijgen. Dit heb ik uiteindelijk opgelost door een api-call te doen naar de licentieservice die het certificaat in de content meestuurt naar de licentiemanager. Hier wordt de content bekeken en kan deze gebruikt worden in de applicatie.

## 4. Competenties

### 4.1. Analyse

Voordat er een goede analyse gemaakt kon worden, heb ik eerst [interviews](#) afgenomen met de stakeholders van mijn project. Al deze interviews zijn gebundeld en uitgewerkt in het [analysedocument](#). Door deze interviews samen te voegen was het heel snel zichtbaar welke kant ik op moest gaan met de ontwikkeling van de applicatie. Ik heb door middel van deze interviews dan ook mijn functionele en niet functionele eisen opgesteld welke ik daarna weer heb gekoppeld aan mijn userstories. Voor deze eisen en stories is een [acceptatietestplan](#) geschreven om de juiste werking te testen. Door gebruik te maken van de Moscow-methode heb ik bepaald wat mijn Minimum Viable Product (MVP) wordt voor het maken van mijn Proof of Concept (PoC).

### 4.2. Advies

Na het onderzoeken van de verschillende licentiemodellen heb ik in een advies beschreven wat ik TBWB zou adviseren. Hierin staat wat voor soort licentiemodel passend is en hoe we dit model moeten inrichten voor onze klanten. Dit advies staat beschreven in het [onderzoeksdocument](#) in hoofdstuk vier.

Na het afronden van mijn analyse heb ik een [presentatie](#) gegeven voor alle stakeholders. In deze presentatie heb ik mijn functionele en niet functionele eisen besproken en heb ik mijn idee van de applicatie geadviseerd.

### 4.3. Ontwerp

Na het onderzoeken van bruikbare technieken zijn er vier diagrammen opgesteld welke de software architectuur in kaart brengt. Op basis van het C4 model heb ik een [systeemcontext](#), [container](#)- en [componentdiagram](#) opgesteld. Deze zijn met uitleg terug te vinden in het [ontwerpdocument](#) onder hoofdstuk twee. Na het opstellen van deze drie diagrammen zijn het [klassendiagram](#) en de use cases opgesteld, deze zijn ook terug te vinden in het [ontwerpdocument](#) in hoofdstuk drie.

### 4.4. Implementatie

Om een idee te krijgen van hoe de gebruikerslicenties beheerd moeten worden en hoe deze moeten interfacen op het WMS is er een PoC ontwikkeld. Hierin kan zowel de klant, als de afdeling service en de afdeling administratie inloggen om gebruikerslicenties toe te voegen. Daarnaast kunnen er handmatig certificaten gegenereerd worden en kunnen de facturen geaccordeerd worden.

Er zijn voor de [emailservice](#) en de [certificaatservice](#) prototypes gemaakt om de techniek te onderzoeken alvorens deze geïmplementeerd werden in mijn PoC. De licentieservice en het licentiedashboard zijn parallel naast elkaar ontwikkeld, zodat de functionaliteit gelijk getest kon worden door de gehele applicatie. Er zijn [service- en unittesten](#) ontwikkeld voor het testen van de api-calls en de lossen functionaliteit.

Op het moment van schrijven ben ik nog bezig de serverkant van de applicatie te ontwikkelen. Ik laat u allen dan ook graag de implementatie van mijn applicatie zien op 2 februari 2023 bij mijn examenzitting.



## 4.5. Management

De keuze voor het beheren van mij softwarecode was al vastgelegd door het softwareteam. Zo wordt er gebruik gemaakt van de Atlassian suit met producten zoals Jira en Bitbucket.

Jira is gebruikt om de userstories vast te leggen en om de sprints te plannen, Bitbucket is de [git-repository](#) waar de branches, pullrequest en het versiebeheer van de code bijgehouden worden. In het [ontwikkelpocesdocument](#) staan voorbeelden en wordt er extra uitleg gegeven over dit proces.

Daarnaast is er ook nog een [definition of done](#) opgesteld om ervoor te zorgen dat kwaliteit van de code blijft behouden voordat er gepushed wordt naar de master branch. De ontwikkelde code wordt gecontroleerd door een medeontwikkelaar binnen het softwareteam. Dankzij het vierogenprincipe wordt de kwaliteit van de code behouden en wordt de kans op bugs verkleind.

De codestructuur wordt uniform gehouden door dezelfde 'formatter settings' te gebruiken als in de andere projecten van TBWB. Hiervoor wordt gebruik gemaakt van een [DotSettings](#) file die wordt ingeladen in Visual studio. Wanneer er een save doet op je project zal alle code worden geformatteerd volgens het settingsbestand.

## 4.6. Onderzoeken

In het eerste deel van het onderzoek ligt de focus op de verschillende licentiemodellen en hoe deze toegepast kunnen worden op wensen van TBWB. Bij de afronding van dit deel van het onderzoek kan de analyse plaatsvinden om de richting van het project te bepalen door feedback van de stakeholders. De interviews zijn uit het onderzoek gehaald omdat deze beter bij de bedrijfsanalyse passen. Dit onderzoek en advies is terug te vinden in het [onderzoeksdocument](#) in hoofdstuk drie en vier.

In het tweede deel van het onderzoek zijn de technieken onderzocht die nodig zijn voor het maken van de applicatie. Een belangrijk deel van het project was het veilig genereren van een certificaat. Het certificaat wordt bij ons op de server gegenereerd en naar de server van de klant gestuurd, er moet dus een encryptie plaatsvinden om deze data te beveiligen op aanpassingen van buitenaf.

In eerste instantie is er gekozen om voor encryptie van het gehele certificaat te gaan. Echter bleek dat deze versleuteld werd met een publieke sleutel en gedecodeerd werd met een privésleutel. Na feedback van het team bleek dat dit niet correct was omdat we dan onze privésleutel moesten versturen naar de klant. Hierdoor is de keuze gevallen op signing. Hier wordt het certificaat ondertekend met onze privésleutel. Het certificaat en de sleutel wordt omgezet naar een Hash die vervolgens door de andere kant wordt gedecodeerd met de publieke sleutel. In hoofdstuk vijf van het [onderzoeksdocument](#) wordt dieper ingegaan op de encryptie.

Na mijn onderzoek van de mailserver was het type server kiezen een makkelijke taak. Er is eigenlijk maar één server die voor mij het mailverkeer kan verzorgen en dat is de SMTP-server. Het hoofddoel van deze server is om een mail op te stellen en te verzenden naar het opgegeven emailadres.

#### **4.7. Communicatie**

Het grootste deel van de communicatie verloopt via de Zoom-meeting van het ontwikkelteam. Wij zitten de gehele dag online in dezelfde meeting zodat we snel kunnen schakelen op het moment dat er vragen zijn. Elke dag om kwart over acht wordt er een [stand-up](#) gehouden waar het team zijn voortgang van de dag ervoor bespreekt en er komt aan bod/wordt gekeken naar wat de verwachtingen zijn van de huidige dag.

Wanneer nodig wordt er een meeting ingepland met een medewerker van het bedrijf om bepaalde zaken zoals architectuur en beveiliging door te spreken. Dit was de mogelijkheid om extra informatie te verzamelen voor het onderzoek of implementatie.

Er zijn met alle stakeholders ook een-op-een sessies gehouden om de vragen van de [interviews](#) voor te leggen. Nadat ik de analyse had voltooid, is er een bijeenkomst geweest met alle stakeholders om mijn plan voor te leggen.

In het onderzoekstraject is voornamelijk met mijn bedrijfsbegeleider gesproken om te kijken of zijn wensen overeenkwamen met de uitkomsten uit het onderzoek. Zodra de ontwikkeling van de applicatie begon, is er elke week een [demo](#) aan het gehele softwareteam gegeven. In deze demo's werd er feedback ontvangen en verwerkt in de applicatie voor de volgende demo.

#### **4.8. Leervermogen**

Er zijn drie belangrijke leerpunten geweest voor dit project. Hoe werkt encryptie? Hoe worden er vanuit de applicatie mails verstuurd? En hoe kan ik mijn Nederlands taalgebruik beter onder de knie krijgen?

Bij de eerste twee punten is de kennis door middel van het onderzoek en prototypes verworven, maar voor het Nederlandse taalgebruik is de hulp ingeschakeld van een goede kennis. Zij is lerares Nederlands en heeft samen met mij, bij het maken van het projectplan, heel duidelijk aangegeven waar ik op moest letten. Deze verbeteringen zijn terug te zien in dit [document](#).

## 5. Conclusie

Het doel van het project was om onderzoek te doen en een PoC te ontwikkelen om te kijken hoe er bij TBWB gebruik gemaakt kan worden van een softwarelicentiemodel. Voor het project was de feedback van de stakeholders van belang om een MVP op te stellen voor mijn PoC. De belangrijkste punten uit het interview heb ik verwerkt in userstories en geïmplementeerd in mijn applicatie.

Het doel om te kijken hoe een softwarelicentiemodel geïmplementeerd kan worden is binnen het traject volbracht. Zo is er door middel van een certificaat, die beveiligd is voor fraude, rekening gehouden met de veiligheid van de applicatie. Door gebruik te maken van de mailserver is er een manier gevonden om de klant op de hoogte te houden van zijn licentiestatus en is er een optie gekomen voor de afdeling service van TBWB om een certificaat te genereren zodra er een verbindingsprobleem ontstaat tussen de licentieservice en de licentiemanager. Echter blijft er wel ruimte voor verbetering, dit ga ik met het softwareteam oppakken zodra we de applicatie gaan doorontwikkelen voor in het standaardproduct.

Een voorbeeld van een van deze verbeterpunten is het implementeren van 2FA bij het inloggen voor de licentieservice, zodat we weten dat de juiste persoon de licenties beheert. Een ander punt dat nog aandacht vereist, is het uitbouwen van de licentieservice tot een klantenportaal waar de klant zijn servicecontracten kan zien en waar al zijn facturen zichtbaar zijn. Daarnaast stuurt de mailserver een mail naar de klant wanneer zijn factuur weer betaald moet worden. Dit zal uiteindelijk een automatische incasso worden, zodat er minder administratieve handelingen uitgevoerd moeten worden.

De applicatie is op dit moment werkend op mijn lokale machine, ik hoop dat ik bij de demo de licentiemanager kan tonen op een virtuele machine die de server van de klant moet vertegenwoordigen. Daarnaast zal er met betrekking tot de wet- en regelgeving nagekeken moeten worden of de middelen die we gebruiken om de klant tot betalen te dwingen wel mogen.

Voor nu zal deze applicatie als rode lijn gelden voor de doorontwikkeling van deze applicatie en heb ik hoge verwachtingen dat de stakeholders van TBWB ook overstag gaan met de implementatie van dit softwarelicentiemodel.

### 5.1. Reflectie

In mijn project proposal heb ik een aantal persoonlijke leerdoelen beschreven. Eén van deze doelen was dat ik meer wilde weten over licentiemodellen en een ander doel was hoe ik dit licentiemodel moet implementeren in een applicatie. Ik denk dat ik beide doelen heb behaald omdat er een werkende applicatie is opgeleverd die een licentiemodel implementeert en dat ik de juiste vorm qua softwarelicentiemodel heb kunnen adviseren aan TBWB.

Als ik terugkijk op mijn traject zijn er wel een paar dingen die ik anders zou doen. Vanwege het feit dat ik een lange periode van mijn project bezig ben geweest met stakeholders, hun wensen en de eisen die ik aan de applicatie heb gesteld, ben ik te laat aan de ontwikkeling van de applicatie gestart. Toen ik mezelf uiteindelijk volledig op de ontwikkeling van de applicatie had gestort kwam de deadline van het portfolio snel in zicht.

Na feedback van mijn begeleider bleven er nog een aantal punten over die ik nog moest verwerken voor het aantonen van mijn leerdoelen. Dit resulteerde in een rommelig ontwikkelingstraject van mijn applicatie. Ik was bijvoorbeeld nog niet begonnen om mijn API calls te testen, mijn testplan was nog niet helemaal compleet en mijn unittesten stonden nog niet klaar.

Nu was het advies van mijn eerste en tweede beoordelaar om alles voor het aantonen van mijn leerdoelen eerst te doen, omdat ik in de periode tussen de deadline en mijn examenzitting nog tijd had om mijn applicatie verder te ontwikkelen. Er staan nog een paar restpunten open als het gaat om de ontwikkeling van mijn applicatie. Zo moet ik nog meldingen genereren zodra het watchdog component ziet dat er geen verbinding meer is tussen de service en manager. Er moet nog een service en administratiepagina komen in mijn licentiemanager dashboard en zou ik de serverkant van mijn applicatie graag willen aantonen in een demo omgeving op een virtuele machine. Dit is natuurlijk niet per se nodig om de werking van mijn applicatie aan te tonen, maar zal uiteindelijk wel de kers op de taart zijn voor de demo ik u allen ga geven.

Zoals hierboven in mijn leeswijzer aangegeven denk ik dat ik voldoende bewijslast heb geleverd om al mijn leerdoelen aan te tonen en kan ik niet wachten om u een demo te geven van mijn applicatie.

## 6. Verwijzingen

Wikipedia-bijdragers. (2022, 16 december). *Scrum (projectmanagementmethode)*. Wikipedia.  
[https://nl.wikipedia.org/wiki/Scrum\\_\(projectmanagementmethode\)](https://nl.wikipedia.org/wiki/Scrum_(projectmanagementmethode))

The C4 model for visualising software architecture. (z.d.). <https://c4model.com>

MoSCoW – *projectmanagementsite*. (z.d.). <https://projectmanagementsite.nl/moscow/>

What is a Minimum Viable Product? (2020, 08 20).  
<https://www.techopedia.com/definition/27809/minimum-viable-product-mvp>