

DGMR - Rekentool

Het ontwikkelen van een rekentool webapplicatie voor het berekenen van het isolerend vermogen van een constructie bij DGMR Software.

OZAN PALALI

Haagse Hogeschool



MIJN GEGEVENS

Naam : Ozan Palali
Studentnummer :
Telefoonnummer :
Privé emailadres :

AFSTUDEERBEDRIJF

Naam bedrijf : DGMR
Afdeling bedrijf :
Adres :
Postcode : Den Haag
Telefoon bedrijf :
Bedrijfsmentor :
Email bedrijfsmentor :

ONDERWIJSTINSTELLING

SLB'er : Anneke Wieman
Begeleidend examiner : Gerda in 't Veld
Expert examiner : Dinesh Ramawadh

REFERAAT

Dit document is geschreven naar aanleiding van de uitgevoerde afstudeeropdracht bij DGMR Software. Deze afstudeeropdracht maakt onderdeel uit van de opleiding HBO-ICT Software Engineering (SE) aan de Haagse Hogeschool.

Tijdens de afstudeerperiode van zeventien weken is er een webapplicatie ontwikkeld die het mogelijk maakt om het isolerend vermogen van een constructie uit te rekenen aan de hand van materiaalgegevens die worden ingevoerd door een gebruiker. Hiermee kunnen gebruikers registreren, inloggen, projecten beheren, berekeningen opslaan, berekeningen inladen, berekeningen exporteren naar PDF en berekeningen exporteren naar XML.

DESCRIPTOREN

- AngularJs
- ASP.NET Core
- Javascript
- API
- ERD
- JWT Authenticatie
- N-Tier design pattern
- UML
- Azure
- Database
- DGMR Rc-Tool
- DGMR Software
- Rc-waarde
- Rekentool

VOORWOORD

Tijdens mijn afstuderen bij DGMR Software heb ik de gelegenheid gekregen om naast mijn afstudeeropdracht ook aan mijn afstudeerdocumenten te werken. Ik heb de opdracht die ik toegewezen heb gekregen als erg interessant en uitdagend ervaren. In het verleden heb ik gewerkt voor een IT-bedrijf dat zich ook bezighoudt met ASP.NET en AngularJS. Alleen heb ik nooit echt de gelegen gehad om een webapplicatie met ASP.NET Core en AngularJS van de grond af aan op te bouwen. Bij DGMR Software heb ik daar wel de gelegenheid voor gekregen. Deze opdracht brengt ook veel zelfstandigheid en verantwoordelijkheid met zich mee. Op grond van mijn vertrouwen in mijn kennisniveau en ervaring in ASP.NET en AngularJS, ben ik ervan overtuigd dat ik een eindproduct zal opleveren dat voldoet aan de kwaliteit en functionaliteitseisen. Door deze verantwoordelijkheid en zelfstandigheid ben ik erg gemotiveerd en enthousiast. Ook heeft wat ik oplever daadwerkelijk toegevoegde waarde voor het bedrijf waardoor ik de groei en het succes kan bevorderen. Dat geeft mij een goed gevoel. Behalve mijzelf is er ook een stagiaire van mijn school daar werkzaam. Dit zorgt voor een goede sfeer en maakt het leuker om hier af te studeren. Verder zijn de begeleiders en medewerkers erg vrolijk en zorgen zij voor een leuke werkomgeving.

INHOUDSOPGAVE

1	Inleiding.....	1
2	DGMR	2
3	Probleemstelling	3
4	Doelstelling.....	4
5	Werkzaamheden	4
5.1	Introductie & inwerken	4
	Scrum	5
	Ontwerpen	5
	Backlog	8
	Planningsschema.....	10
5.2	Sprint 1 Initiële opzet	11
5.3	Sprint 2 Projecten realisatie	16
5.4	Sprint 3 Database conversie	20
5.5	Sprint 4 Rc -waarde berekening.....	25
5.6	Sprint 5 Rc berekening verbeteren en opslaan	31
5.7	Sprint 6 Rc berekening ophalen en Exporteren PDF.....	33
5.8	Sprint 7 XML export en publish	40
6	Evaluatie.....	46
6.1	Producten	47
	Plan van aanpak	47
	Requirements.....	47
	Ontwerprapport.....	47
	Databaseconversie	47
	Database	47
	DGMR Rctool.....	48
	Definition of Ready.....	48
	Definition of Done	48
6.2	Beroepstaken.....	48
	A1 Analyseren probleemdomein	48
	A3 Vergaren en analyseren van requirements.....	48
	C6 Ontwerpen Software.....	48
	D14 Realiseren van software	49
	D16 Realiseren & gebruiken database	49
	Gc Kritisch, onderzoekend en methodisch werken.....	49
	GF Leren Leren	49

7	Bibliografie	51
	Bijlagen.....	52
	Bijlage A - Afstudeerplan	52
	Bijlage B - Plan van aanpak	52
	Bijlage C – Ontwerprapport	52
	Bijlage D - Defintion of Done	52
	Bijlage E - Defintion of Ready.....	52
	Bijlage F – Conversie Scripts.....	52

LIJST VAN FIGUREN

Figuur 1	Organogram DGMR.....	2
Figuur 2	Organogram DGMR Software	3
Figuur 3	Globale database ontwerp.....	7
Figuur 4	Strokenplanning	10
Figuur 5	Sprint 1 backlog.....	11
Figuur 6	Links tussen Javascript controllers en HTML.....	12
Figuur 7	webapplicatie ontwerp 1	14
Figuur 8	sprint 2 backlog	16
Figuur 9	Rollen diagram	17
Figuur 10	webapplicatie ontwerp sprint 2	18
Figuur 11	ERD 1	19
Figuur 12	Sprint 3 backlog.....	20
Figuur 13	ERD 2	21
Figuur 14	Dataconversie Papa Parse	22
Figuur 15	Dataconversie Upload CSV	22
Figuur 16	Dataconversie spaties corrigeren.....	23
Figuur 17	Dataconversie HTTP requests	23
Figuur 18	Sprint 4 backlog.....	25
Figuur 19	ERD 3	26
Figuur 20	Warmte overgangsweerstanden.....	27
Figuur 21	Sprint 5 backlog.....	31
Figuur 22	ERD 3	32
Figuur 23	Sprint 6 backlog.....	34
Figuur 24	ERD 4	36
Figuur 25	Ingeladen materialen en waardes.....	37
Figuur 26	PDF export schets.....	38

Figuur 27 sprint 7 backlog	40
Figuur 28 Laden constructie	41
Figuur 29 Calculatie opslaan in project	41
Figuur 30 Nieuwe project & calculatie	42
Figuur 31 Rekenpagina.....	43
Figuur 32 XML Export javascript	44

1 INLEIDING

Dit afstudeerdossier bevat de informatie die gerelateerd is aan de werkzaamheden en resultaten bij DGMR Software. Tijdens het afstuderen wordt een webpagina gerealiseerd die het mogelijk maakt om het isolerend vermogen van een constructie uit te rekenen door middel van het selecteren van materialen.

In hoofdstuk 2 wordt de algemene achtergrondinformatie van DGMR beschreven. Ook wordt hierin beschreven wat de verschillende softwareproducten zijn die DGMR Software aanbiedt en waar ze voor dienen. In hoofdstuk 3 wordt beschreven hoe de webapplicatie in elkaar zit en welke keuzes er zijn gemaakt om het te realiseren. Ook wordt er een en ander uitgelegd over de infrastructuur en de beveiliging. In hoofdstuk 4 en 5 wordt beschreven hoe het probleem ontstaan is en hoe eraan gewerkt gaat worden om dit op te lossen. Hoofdstuk 6 gaat over de leerdoelen die zijn opgesteld. Verder wordt in hoofdstuk 7 de databasestructuur uitgelegd. Hierin wordt onder andere vermeld wat de relaties tussen de verschillende entiteiten zijn. In hoofdstuk 8 beginnen de werkzaamheden per sprint. Tijdens de eerste week van de afstudeerperiode is er ingewerkt. Tijdens het inwerken zijn er requirements verzameld, een globale ontwerp diagram gemodelleerd, een plan van aanpak gemaakt en een initiële backlog opgesteld. Daarnaast zijn ook de Definition ready en Definition of Done gedefinieerd.

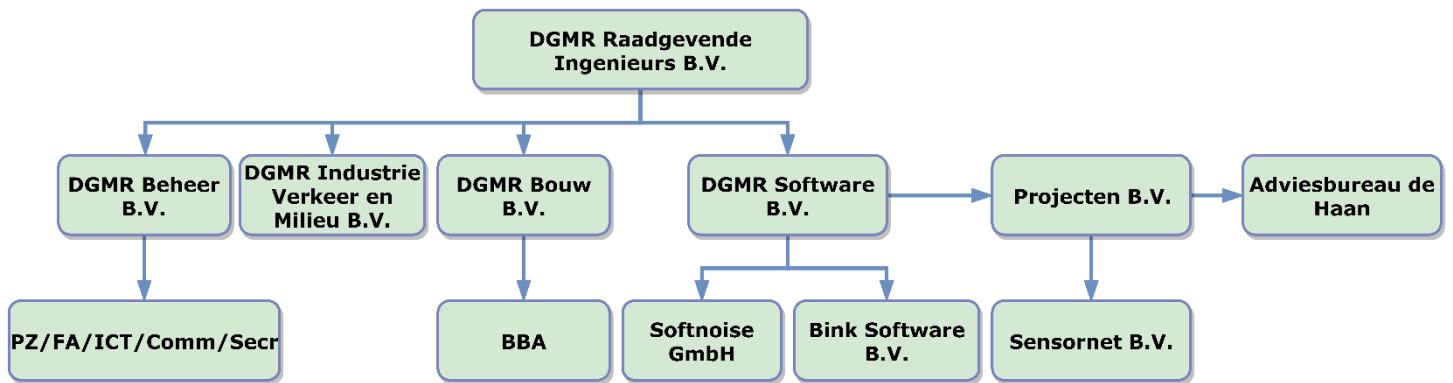
Tijdens sprint 1 is de initiële opzet van de webapplicatie uitgevoerd. Hierbij is het inloggen en registreren van gebruikers gerealiseerd. In sprint 2 is de projectenfunctionaliteit gerealiseerd. Hiermee kunnen gebruikers hun projecten aanmaken en beheren. Vervolgens is er in sprint 3 gewerkt aan de databaseconversie om ervoor te zorgen dat de database gevuld kan worden aan de hand van een CSV-bestand. Aan de hand hiervan kunnen beheerders van de webapplicatie de database vullen met materiaal gegevens door middel van het uploaden van een CSV-bestand. Tijdens sprint 4 is het zover om de daadwerkelijke berekening te implementeren. Hierbij kan er gekeken worden naar de oude SBR-Rekentool voor de verschillende voorwaardes waaraan de berekening moet voldoen. Tijdens sprint 5 is het opslaan van deze calculaties gerealiseerd. Daarnaast zijn de berekeningen uitgebreid en verbeterd. In sprint 6 is er gewerkt aan het inladen van de opgeslagen calculaties, zodat gebruikers hun opgeslagen berekeningen kunnen terughalen. Verder wordt er in deze sprint ook gewerkt aan de PDF-export functionaliteit.

Tot slot bevat hoofdstuk 9 de evaluatie en reflectie.

2 DGMR

Algemene achtergrondinformatie over het bedrijf.

DGMR is een adviesbureau met ruim 200 ingenieurs en adviseurs. DGMR is ervan overtuigd dat mensen het beste tot hun recht komen in een duurzame, veilige en gezonde omgeving. Ze leveren oplossingen die doeltreffend zijn voor de samenleving (zie Figuur 1 Organogram DGMR).



Figuur 1 Organogram DGMR

Het bedrijf is opgericht in 1980 door de ingenieurs Van Dalfsen, Gies, Meerdink en Van Rangelrooij. Dit deden ze omdat ze kwaliteit met creativiteit wilden verbinden. DGMR gebruikt zijn kennis om met oplossingen te komen die nadrukkelijk betrekking hebben tot duurzaamheid, veiligheid en gezondheid.

DGMR Software is een B.V. binnen de DGMR holding. Deze afdeling combineert softwareontwikkeling en kennis op het gebied van milieu, bouw installatie en duurzaamheid (zie *Figuur 2 Organogram DGMR Software*)

DGMR Software heeft de specialistische kennis in huis om software te ontwikkelen op de gebieden industrielawaai, weg- en railverkeerslawaaai, lichtkwaliteit, geur, arbeidsplaats, brandveiligheid, bouw fysica, installatie, duurzaam bouwen en energieprestatie.

De rijksoverheid, maar ook vele adviesbureaus, gemeenten, regionale uitvoeringsdiensten, architectenbureaus en installatiebedrijven maken gebruik van de standaardsoftware van DGMR Software. Ze hebben onder andere de volgende pakketten ontwikkeld: Geomilieu, ENORM en BINK. Dit zijn de drie grootste pakketten van DGMR Software. Geomilieu en Bink zijn omgevingen, deze bestaan uit meerdere applicaties. ENORM is weer een los staande applicatie.

Geomilieu is een softwarepakket dat tools biedt voor het analyseren en berekenen van de milieubelasting als gevolg van geluid en luchtkwaliteit. Het programma is in staat om geluidsberekeningen voor wegverkeerslawaaai, industrielawaai, en railverkeerslawaaai uit te voeren. Naast geluidsberekeningen is Geomilieu ook in staat om luchtkwaliteitsberekeningen en geurberekeningen uit te voeren. De berekende resultaten worden vervolgens gepresenteerd en geanalyseerd.

ENORM is een desktopapplicatie die op eenvoudige en snelle wijze energieprestatieberekeningen volgens de energieprestatienorm NEN 7120 (NEN, 2011) uitvoert. ENORM beschikt over uitgebreide mogelijkheden voor het gebruik van kwaliteitsverklaringen, het maken van rekenvarianten en het rapporteren van de invoergegevens en resultaten. ss

BINK is een rekenomgeving waarin onder andere de volgende berekeningen gemaakt kunnen worden: warmteverlies, EPC (energieprestatiecoëfficiënt), brandoverslag, dynamische simulatie en

daglichtberekeningen. Ook een 3D-gebouw tekensoftware maakt deel uit van deze omgeving. Met BINK 8 worden leidingberekeningen gemaakt voor gas, verwarming, ventilatie en riolering.



Figuur 2 Organogram DGMR Software

De hiërarchie binnen DGMR Software is te zien in *Figuur 2 Organogram DGMR Software* hierboven. Bovenaan het organogram is natuurlijk de directeur wie in dit geval Dhr. Schmidt is. Als directeur is dhr. Schmidt uiteraard verantwoordelijk voor de belangrijke beslissingen binnen DGMR Software. Hieronder valt het releaseschema voor diverse producten, dus wanneer welke software wordt gereleased. Dhr. Schmidt beslist ook hoe de middelen binnen DGMR Software worden toegepast, dus wie waaraan moet werken. Dhr. Schmidt stuurt hiernaast ook de visie van het bedrijf en maakt strategische keuzes aan de hand van de doelmarkt. Daarnaast is Dhr. Schmidt verantwoordelijk voor de hoeveelheid omzet die gegenereerd moet worden. Dhr. Bosch is zoals vermeld in het afstudeerplan de productmanager en tegelijkertijd ook de bedrijfsmentor. Dhr. Bosch is verantwoordelijk voor het releaseschema van de BINK-desktopapplicatie. Daarbij ligt de prioriteit van Dhr. Bosch grotendeels bij BINK Software. Als afstudeerder ligt de rol bij developer/tester onder de productmanager.

3 PROBLEEMSTELLING

In het verleden is de SBR Rekentool webapplicatie ontwikkeld door Bink Software (dit bedrijf is later overgenomen door DGMR Software) echter heeft DGMR Software daar de rechten niet meer voor, omdat het ontwikkeld is voor een bedrijf dat failliet is gegaan en is verkocht. De rechten van de SBR-Rekentool zit dan ook bij dat bedrijf. SBR wat staat voor Stichting Bouw Research was een bedrijf die zich richtte op burgerlijke en utiliteitsbouw (B&U) bouw van woningen, kantoren en dergelijke. Dit bedrijf was in 2013 gefuseerd met CURnet. CURnet stond bekend voor het maken van aanbevelingen en richtlijnen voor de GWW-sector (Grond-,

Weg- en Waterbouw). De SBR-rekentool was ontwikkeld door Bink Software voor SBRCURnet wat de fusering is van SBR en CURnet. SBRCURnet is eind 2017 failliet verklaard. Eind 2015 is er een nieuwe directeur aangesteld om op zoek te gaan naar een nieuwe financieringsconstructie. Dat was niet gelukt.

Met de SBR-rekentool konden isolatie installatiebedrijven het isolerend vermogen van een constructie berekenen op basis van verschillende aspecten en producten zoals bijvoorbeeld isolatiemateriaal. Het isolatievermogen van een constructie wordt aangeduid met de R_c -waarde. De SBR-rekentool voldoet niet meer aan de eisen van DGMR Software. Dit komt omdat de huidige rekentool een onaantrekkelijke en onoverzichtelijke design/lay-out bevat. Daarnaast is de oude SBR-rekentool gebouwd in PHP. Dit is een probleem, omdat de huidige kennis en ervaring van het ontwikkelteam van DGMR Software niet meer gericht is op PHP.

4 DOELSTELLING

De nieuwe DGMR Rekentool genaamd DGMR Rc-tool moet ontwikkeld worden met ASP.NET en AngularJS, omdat dit de programmeertalen zijn waar DGMR Software tegenwoordig goede kennis over beschikt. De DGMR Rc-tool is hierbij de nieuwe situatie. De SBR-Rekentool is de oude situatie. Verder moet er een klantenportaal ontwikkeld worden waarmee betalende klanten kunnen inloggen en registreren. Daarbij hoort ook een SQL-database dat ontworpen en gerealiseerd dient te worden. Om deze database te vullen met de benodigde materiaal gegevens moet er een database conversie geschreven worden. De berekeningen die van toepassing zijn voor de te ontwikkelen software worden door wiskundige en natuurkundige adviseurs binnen DGMR van tevoren gespecificeerd. De klanten zijn de werknemers van installatiebedrijven, adviseurs en producenten van bouwmaterialen, architecten, bouwkunde studenten en werktuigbouwkunde studenten. Adviseurs en producenten van bouwmaterialen kunnen betalen voor de DGMR Rc-tool als zij behoefte hebben aan projecten beheer en het kunnen opslaan en inladen van calculaties. Tevens kunnen zij ook gratis gebruik maken van de DGMR Rc-tool, mits ze alleen maar een R_c -waarde berekening moeten maken. Bouwkunde en werktuigbouwkunde studenten kunnen een gratis account verkrijgen van de school waar ze in studeren. Dit is mogelijk, omdat dhr. Bosch connecties heeft met deze opleidingen. Dhr. Bosch is namelijk een parttime docent bij de Hogeschool van Rotterdam. Vanuit deze positie kan Dhr. Bosch vervolgens in contact komen met studenten en andere docenten.

Met het uitvoeren van deze afstudeeropdracht wordt een webapplicatie met front-end zowel als back-end ontwikkeld met daarbij de bijbehorende database en klantenportaal, die DGMR Software verder kan uitbreiden met andere constructies.

5 WERKZAAMHEDEN

5.1 INTRODUCTIE & INWERKEN

Tijdens de introductie is er kennis gemaakt met de faciliteiten en medewerkers van DGMR-Software. Verder zijn er requirements verzameld door middel van de stakeholders te interviewen en is er een initiële backlog opgesteld met epics en userstories. Daarnaast is er een plan van aanpak opgesteld. Ook moet er een database conversie plaatsvinden. Alle benodigde data zit namelijk nog in de oude SBR-rekentool gemaakt door BINK Software. Deze moesten allemaal geconverteerd worden naar de nieuwe database. De oude data is beschikbaar in csv formaat. Er moest dus een conversie geschreven worden die de Excel bestanden omzet naar data die past in de nieuwe databasestructuur. Tijdens het inwerken zijn ook de Defintion of Ready en Definition of Done gedefinieerd. Daarnaast is er ook een globale database ontwerp gemodelleerd (zie Figuur 3 Globale database ontwerp). Het inwerken heeft in totaal 1 week geduurd. Verder is er alvast een Visual Studio project aangemaakt, zodat de werkzaamheden in sprint 1 gelijk kunnen beginnen. Tot slot is voor versiebeheer besloten om GIT te gebruiken van Azure DevOps in combinatie met Visual Studio Team Services.

SCRUM

De software-ontwikkelmethode die zal worden toegepast is SCRUM. Deze ontwikkelmethode is gekozen om inzicht te krijgen in de voortgang van het project en omdat niet alle requirements direct vanaf het begin duidelijk zijn. Ook is er gekozen voor SCRUM, omdat er tijdens de afstudeeropdracht veel belangrijke veranderingen worden verwacht. De veranderingen die kunnen worden verwacht zijn veranderingen in het ontwerp, verandering van de prioriteiten door de stakeholders en mogelijke nieuwe userstories en taken van de stakeholders. Bovendien is er ook voor SCRUM gekozen, omdat de aanpakken voor sommige userstories nog onduidelijk zijn. Met SCRUM kan er snelwerkende producten opgeleverd worden per increment. Vervolgens kan hier feedback op gegeven worden tijdens de sprint review. Dit zorgt ervoor dat het project wordt geleid naar een goed eindresultaat. Bij het realiseren van de DGMR Rekentool gaat er gewerkt worden in sprints van twee weken. Er is gekozen voor twee weken, omdat er anders te weinig gerealiseerd is voor de review en feedback aan het einde van de sprint. De stakeholders zijn Richard Schmidt; de product owner en Hans Bosch; de product manager. In het begin van de week van een sprint wordt de sprint planning gehouden. De sprint review staat gepland op het einde van een sprint met Richard Schmidt de product owner en Hans Bosch de product manager. Het einde van een sprint kan op een maandag of dinsdag zijn, dat hangt af van de beschikbaarheid van de stakeholders. Vervolgens zal er op dezelfde dag een retrospective plaatsvinden met dhr. Bosch en dhr. Schmidt. Verder is er besloten om in plaats van een daily standup een weekly meeting te houden vanwege kennis en ervaring met dotNet en AngularJs. Door deze ervaring en kennis is er minder begeleiding nodig bij de realisatie. Samen met dhr. Bosch is er uiteindelijk besloten dat het houden van een wekelijkse meeting tijd efficiënter is dan een dagelijkse meeting gegeven de omstandigheden. De omstandigheden zijn dat er verwacht wordt tegen weinig problemen aan te lopen vanwege ervaring en kennis met het realiseren van dotNet en AngularJs applicaties. Hierdoor is er dus minder tijd nodig om te bespreken tegen wat voor problemen er wordt verwacht aan te lopen. De weekly meeting zal een meeting zijn van circa 10 tot 15 minuten wanneer de sprint op de helft is. Hierin wordt besproken wat er is gedaan in de vorige helft van de sprint en tegen wat voor obstakels er verwacht wordt tegen aan te lopen. Bij het begin van elke sprint zal er eerst een deelontwerp gemaakt worden of er zal een bestaand ontwerp verder uitgewerkt worden met het oog op de sprint. Ook zal er aan het begin van de sprint de Definition of Ready doorlopen worden om te verifiëren of de sprint klaar is om gestart te worden. Tevens zal ook de Definition of Done doorlopen worden aan het einde van elke user story om ervoor te zorgen dat elke user story goed wordt afgerond.

Voor het beheren van het project is Azure DevOps gebruikt. Hiermee kunnen er makkelijk sprints gepland worden door eerst een backlog aan te maken en de backlog items te slepen naar de bijbehorende sprint. Zo kan er gemakkelijk per sprint een overzicht gemaakt worden. De backlog items zijn userstories en epics. Epics hebben userstories en userstories hebben taken. Ook is Azure DevOps gebruikt voor versiebeheer van de webapplicatie. Bij Azure DevOps horen ook een aantal best practices waaraan gehouden zal worden. Eén daarvan is het gebruik maken van versiebeheer, zodat de source code van de webapplicatie altijd online te benaderen is. Nog een best practice waaraan gehouden zal worden, is dat de stakeholders zich actief gaan betrekken bij het ontwikkel proces. Dit wordt gedaan bij de sprint reviews, sprint retrospectives, sprint planningen en weekly meetings. Tijdens de sprint planning is het de bedoeling dat userstories uit de backlog worden meegenomen in de sprint en taken krijgen. Daarnaast worden er ook uren toegekend aan taken om een inschatting te geven hoelang een taak duurt om te realiseren of implementeren.

ONTWERPEN

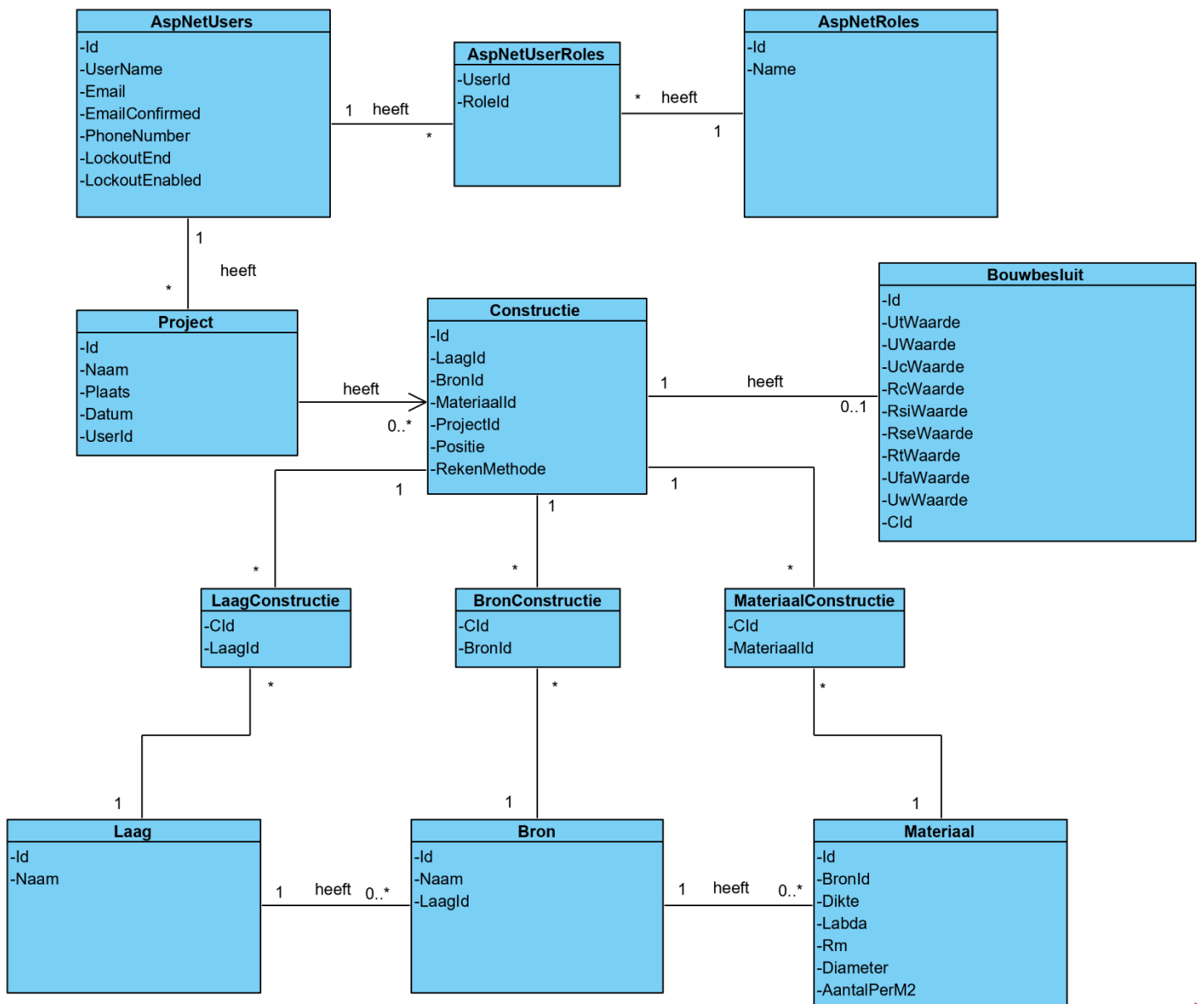
Zoals eerder vermeld zal er aan het begin van elke sprint ook een deelontwerp gemaakt worden en/of een bestaand ontwerp verder uitgebreid worden wat betrekking heeft tot de sprint. Wanneer de database tijdens een sprint uitgebreid of gewijzigd moet worden, zal dit gemodelleerd worden met het Entity-Relationship diagram (ERD). Een ERD is een model of diagram dat het mogelijk maakt om inzicht te geven in een conceptueel, logische of fysiek datamodel. Voor de DGMR Rc-Tool gaat er gebruik gemaakt worden van een fysiek datamodel. Een fysiek datamodel representeert hoe het model in de database wordt geïmplementeerd.

Hierin worden o.a. de relaties, foreign keys, primary keys, tabel structuren, kolomnamen en datatypes weergegeven. Er is gekozen voor een fysiek datamodel, omdat een fysiek datamodel het meest overeenkomt met een databasestructuur. Het ERD is gekozen in plaats van een klassendiagram, omdat het visueel aantrekkelijker, overzichtelijker en gebruiksvriendelijker is dan een klassendiagram voor het visualiseren van een databasestructuur. Wanneer er in een sprint een wijziging moet plaatsvinden aan de databasestructuur zal er een deelontwerp gemodelleerd worden. Dit deelontwerp is een onderdeel van het gehele database ontwerp. In de laatste sprint - die betrekking zal hebben tot wijzigingen aan de databasestructuur, zullen alle deelontwerpen één geheel databaseontwerp vormen. Wanneer er tijdens een sprint een wijziging of uitbreiding gaat plaatsvinden in de structuur van de webapplicatie, zal dit afgebeeld worden in de vorm van een klassendiagram. Voor het weergeven van de infrastructuur van de webapplicatie, wordt er wel gebruik gemaakt van klassendiagrammen. De reden hiervoor is omdat met een klassendiagram de structuur van de webapplicatie overzichtelijk genoeg weergegeven kan worden. Daarnaast is het ook gebruiksvriendelijk. Wanneer er in een sprint een wijziging moet komen in de structuur van de webapplicatie, zal het webapplicatie ontwerp tijdens de ontwerpfase uitgebreid worden.

Zoals vermeld in 5.1 Introductie & inwerken is er een globaal databaseontwerp gemodelleerd. Dit is gedaan in de vorm van een analyse klassendiagram. De reden waarom er een globale database ontwerp gemodelleerd is, is om alvast een overzicht te hebben van hoe de database er ongeveer uit komt te zien in de aankomende sprints (zie *Figuur 3 Globale database ontwerp*). Hieronder zal er nog één en ander toegelicht worden over de globale database ontwerp van figuur 3.

Een AspNetUser heeft meerdere rollen en AspNetRoles heeft meerdere AspNetUsers. Dit is de reden waarom er een koppeltabel staat tussen deze entiteiten. Verder kunnen AspNetUsers projecten aanmaken waar ze gemaakte constructies in kunnen opslaan. Een constructie heeft een bouwbesluit wanneer een constructie is opgezet met materialen en de Rc-waarde is berekend. Verder heeft elke constructie meerdere lagen bronnen en materialen. Dat geldt ook andersom. Lagen, bronnen en materialen kunnen in meerdere constructies voorkomen. Constructie heeft tussentabellen met laag, bron en materiaal omdat deze een meer op meer relatie hebben met elkaar.

- AspNetUsers zijn de gebruikers accounts.
- AspNetRoles zijn de rollen.
- AspNetUserRoles zijn de rollen die bij de users horen.
- Projecten kunnen 0 of meer constructies bevatten die gebruikers hebben opgeslagen. Een gebruiker kan meerdere projecten aanmaken en dus ook meerdere constructies per project.
- Constructies hebben een bouwbesluit wanneer de constructie is opgezet met lagen, bronnen en materialen.
- Bouwbesluit heeft alle waardes die zijn uitgerekend d.m.v. materialen, bronnen en lagen.
- Per laag kan je een bron kiezen. Deze bron heeft vervolgens weer materialen die erbij horen.
- Sommige materialen hebben ook een diameter en een aantal per vierkante meter.



Figuur 3 Globale database ontwerp

Ook zijn tijdens deze sprint sequentie diagrammen gemodelleerd om de registratie en inlog functionaliteit overzichtelijk te maken. Deze zijn te vinden in Bijlage C – Ontwerpproject.

BACKLOG

Tijdens het inwerken is eerst de initiële backlog aangemaakt. Deze is in de loop van het project zo nodig bijgewerkt of aangevuld door onder andere voortschrijdend inzicht. De backlog items zijn in volgorde van prioriteit gerubriceerd. De userstory met de hoogste prioriteit staat op nummer 1. Voordat de backlog items werden gedefinieerd zijn eerst de epics opgesteld. Aan de hand van de epics zijn de backlog items samengesteld. Elke backlog item hoort bij een epic. Voor het vergaren van de epics en userstories zijn er interviews gehouden met dhr. Bosch en dhr. Schmidt. Ook is het plan van aanpak en de afstudeerplan gebruikt voor het bepalen van de backlog items. Nadat ervan overtuigd te zijn dat de backlog zo volledig mogelijk was in dit stadium van de afstudeeropdracht, is er gevraagd aan dhr. Bosch en dhr. Schmidt om deze te prioriteren. Aan de hand van de verkregen informatie zijn vervolgens de backlog items en epics geprioriteerd. Deze priorisering is te zien in de tabellen hieronder. De reden waarom er gaten zitten tussen de nummering van de userstories en taken is, omdat als er in Azure Devops een userstory verwijderd wordt, de userstory nummer ook verwijderd wordt binnen het Azure Devops project. Dus het constant verwijderen en aanmaken van userstories is de oorzaak voor deze gaten in de nummering van de userstories.

Tijdens de realisatie van deze webapplicatie, kan er niet direct begonnen worden aan de userstory met de hoogste prioriteit in sprint 1. Dit komt doordat niet alle userstories gelijk gerealiseerd kunnen worden. De reden hiervoor is het feit dat veel userstories afhankelijk zijn van andere userstories. Om bijvoorbeeld userstory met prioriteit nummer 1 te kunnen realiseren moet eerst userstory met prioriteit nummer 2 gerealiseerd worden. Ook is in de derde tabel waarin de links worden gelegd tussen de epics en userstories, te zien dat sommige userstories dubbel genoteerd staan. Dit komt omdat sommige userstories bij meerdere epics horen. Zie tabellen hieronder.

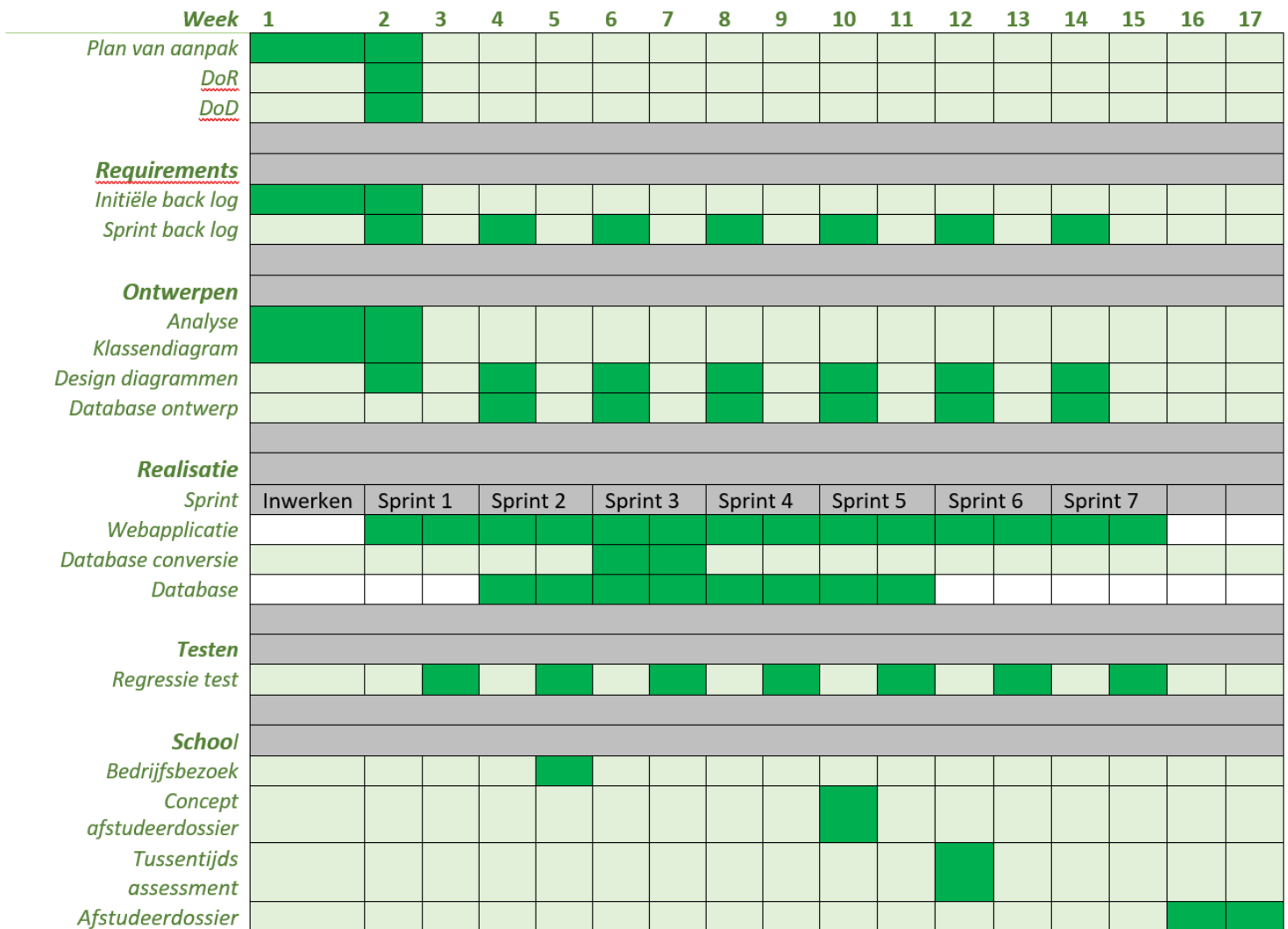
Nr.	Epics
1	Als product owner wil ik een back-end in ASP.NET Core die projecten en materialen opslaat in de database of ophaalt uit de database, zodat ik een API heb waar ik vanuit de AngularJS front-end requests naar toe kan sturen.
2	Als product owner wil ik een front-end gebouwd in AngularJS die verschillende berekeningen doet, zodat ik een dynamische en gestructureerde webapplicatie heb waarmee gebruikers de R_c -waarde van een constructie kunnen berekenen.
3	Als product owner wil ik een Azure SQL Database, zodat ik klantgegevens, constructies, materialen en projecten in kan opslaan.

Prioriteit	Userstory nr.	Backlog items
1	45	Als product owner wil ik een rekenscherf voor de berekening van de R_c -waarde voor spouwmuren, zodat de isolatievermogen van de spouwmuren berekend kunnen worden.
2	23	Als product owner wil ik een home pagina in AngularJS met een selectie van constructies waar klanten berekeningen mee kunnen uitvoeren, zodat klanten een overzicht hebben van de beschikbare constructies.
3	20	Als product owner wil ik dat klanten kunnen registreren en inloggen, zodat klanten projecten kunnen aanmaken.
4	21	Als product owner wil ik dat klanten eerst hun email confirmeren voordat ze toegang krijgen tot de rekentool, zodat ze geen nep accounts opgeven.

5	24	Als product owner wil ik dat ingelogde klanten projecten kunnen aanmaken met plaats, datum, rekenmethode en constructie, zodat betalende klanten verschillende constructies in een project kunnen opslaan.
6	50	Als product owner wil ik de materiaalselectie die gebruikt is voor de berekeningen opslaan in de database, zodat gebruikers hun berekeningen kunnen opslaan en inladen
7	22	Als product owner wil ik onderscheid tussen betalende en niet betalende klanten d.m.v. rollen, zodat ik projecten van betalende klanten kan opslaan en van niet-betalende klanten niet.
8	17	Als product owner wil ik dat beheerders csv-bestanden met materiaalgegevens kunnen importeren, zodat de database gemakkelijk uitgebreid kan worden met nieuwe materiaalgegevens
9	18	Als product owner wil ik dat klanten hun bouwbesluit van een constructie kunnen exporteren naar een PDF bestand, zodat klanten gemakkelijk een overzicht kunnen printen.
10	13	Als product owner wil ik dat klanten hun bouwbesluit van een constructie kunnen exporteren naar een XML-bestand, zodat klanten eventueel hun berekeningen kunnen overzetten naar andere systemen.
11	38	Als product owner wil ik dat de Admin LTE 3 html template wordt toegepast, zodat binnen het bedrijf dezelfde template wordt aangehouden.
12	54	Als product owner wil ik dat gebruikers tijdens het opslaan van een calculatie een project kunnen aanmaken, zodat gebruikers calculaties kunnen opslaan wanneer ze nog geen project hebben.

Epic	Backlog item
1	45, 20, 21, 24, 50, 22
2	45, 23, 24, 50,18, 13, 38, 54
3	17

PLANNINGSSCHEMA

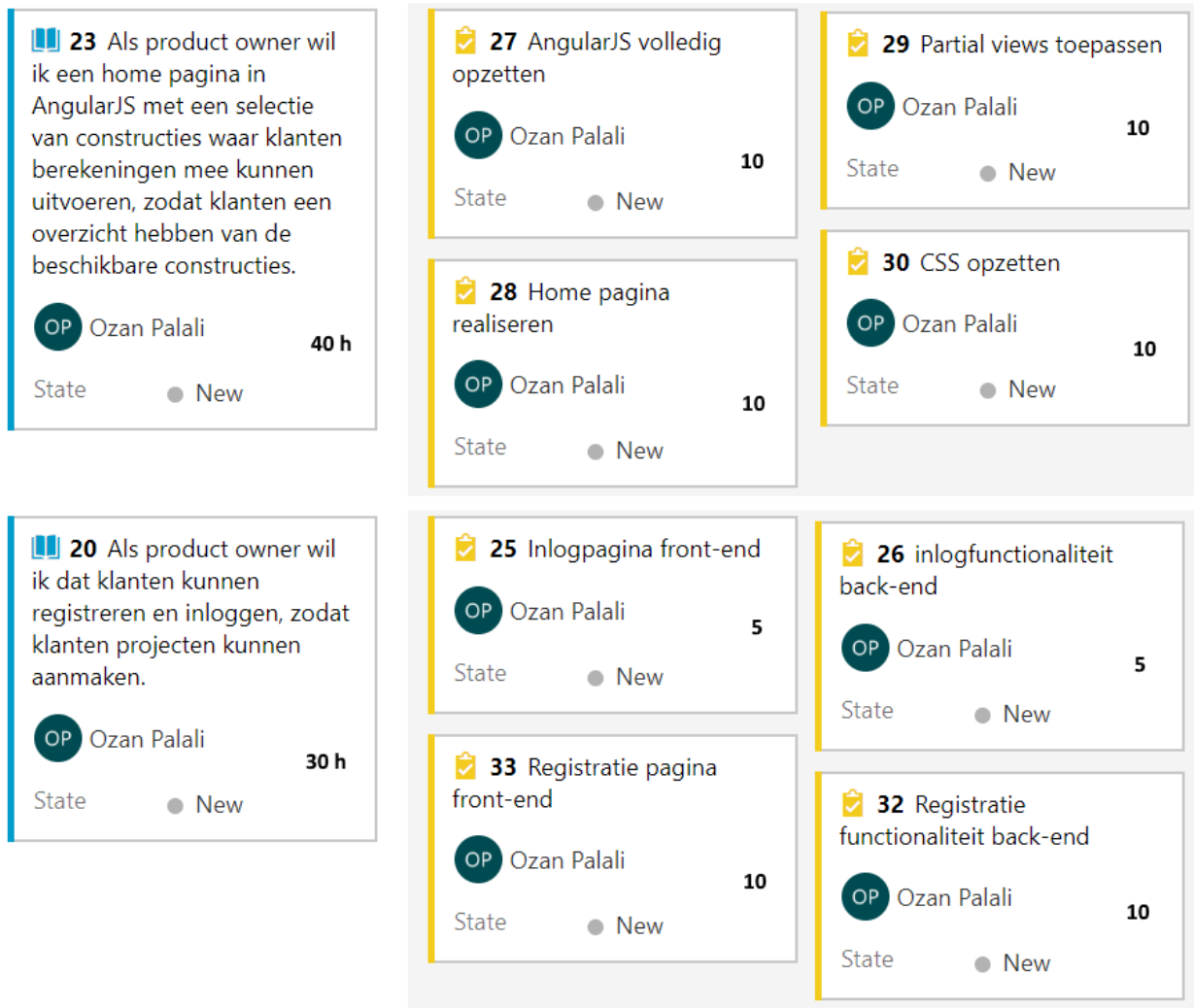


Figuur 4 Strokenplanning

5.2 SPRINT 1 INITIËLE OPZET

REQUIREMENTS & PLANNING

Tijdens de eerste sprint planning is er besloten om de authenticatie en autorisatie te realiseren met de bijbehorende front-end pagina's, controllers en services. Daarbij hoort natuurlijk ook de back-end ingericht te worden. Daarnaast is er uiteraard ook besloten om de front-end te realiseren. Het opzetten van AngularJs, realiseren van de back-end inlogfunctionaliteit en back-end registratie functionaliteit waren de belangrijkste werkzaamheden in deze sprint.



Figuur 5 Sprint 1 backlog

ONTWERP

Tijdens de ontwerpfase is de eerste versie van het webapplicatie ontwerp ontworpen. Binnen dit project is de client-side de front-end en de server-side de back-end.

De DGMR Rc-tool webapplicatie bestaat uit een AngularJs front-end en een ASP.NET Core back-end. De back-end is verantwoordelijk voor het opvangen van http-requests. Met deze requests kunnen dan eventueel berekeningen of verwerkingen gedaan worden. Bij een POST-request wordt er data opgeslagen in de database door middel van repositories of de UserManager en RoleManager. Bij een GET request kan op dezelfde manier als bij een POST data opgehaald worden vanuit de repositories of UserManager en RoleManager. De RoleManager en UserManager worden aangeroepen via de service laag van de back-end. Deze managers communiceren met de automatisch gegenereerde ASP.NET tabellen van de database. Voor deze tabellen zijn er dan ook geen repositories nodig om met de database te communiceren. De tabellen waarvoor dit geldt zijn: AspNetUsers, AspNetUserRoles en AspNetRoles (zie Figuur 3 Globale database ontwerp).

De http-requests komen vanuit de front-end. De html-pagina's van de webapplicatie hebben gelinkte javascript controllers. Deze links staan gedefinieerd in de rekentool_App.js. Hierin is te zien welke controller bij welke HTML-pagina hoort. Zo is in Figuur 6 Links tussen Javascript controllers en HTML hieronder te zien dat bijvoorbeeld de registerController bij de register.html hoort en wat de URL is naar deze pagina.

```
$stateProvider
  .state('dashboard', {
    templateUrl: '/dashboard.index.html',
    controller: 'dashboardController'
  })
  .state('home', {
    url: '/',
    parent: 'dashboard',
    templateUrl: '/Views/home.html',
    controller: 'homeController'
  })
  .state('register', {
    url: '/register',
    templateUrl: '/Views/register.html',
    controller: 'registerController',
    name: 'register'
  })
  .state('login', {
    url: '/login',
    templateUrl: '/Views/login.html',
    controller: 'loginController',
    name: 'login'
  });
```

Figuur 6 Links tussen Javascript controllers en HTML

Daarnaast hebben controllers bijbehorende services of één service. In de controllers staan alle javascript en eventuele jQuery codes. Via de service worden alle http-requests verstuurd naar de back-end API. Deze splitsing van controllers, services en repositories in de back-end is ontworpen om separation of concern te bereiken.

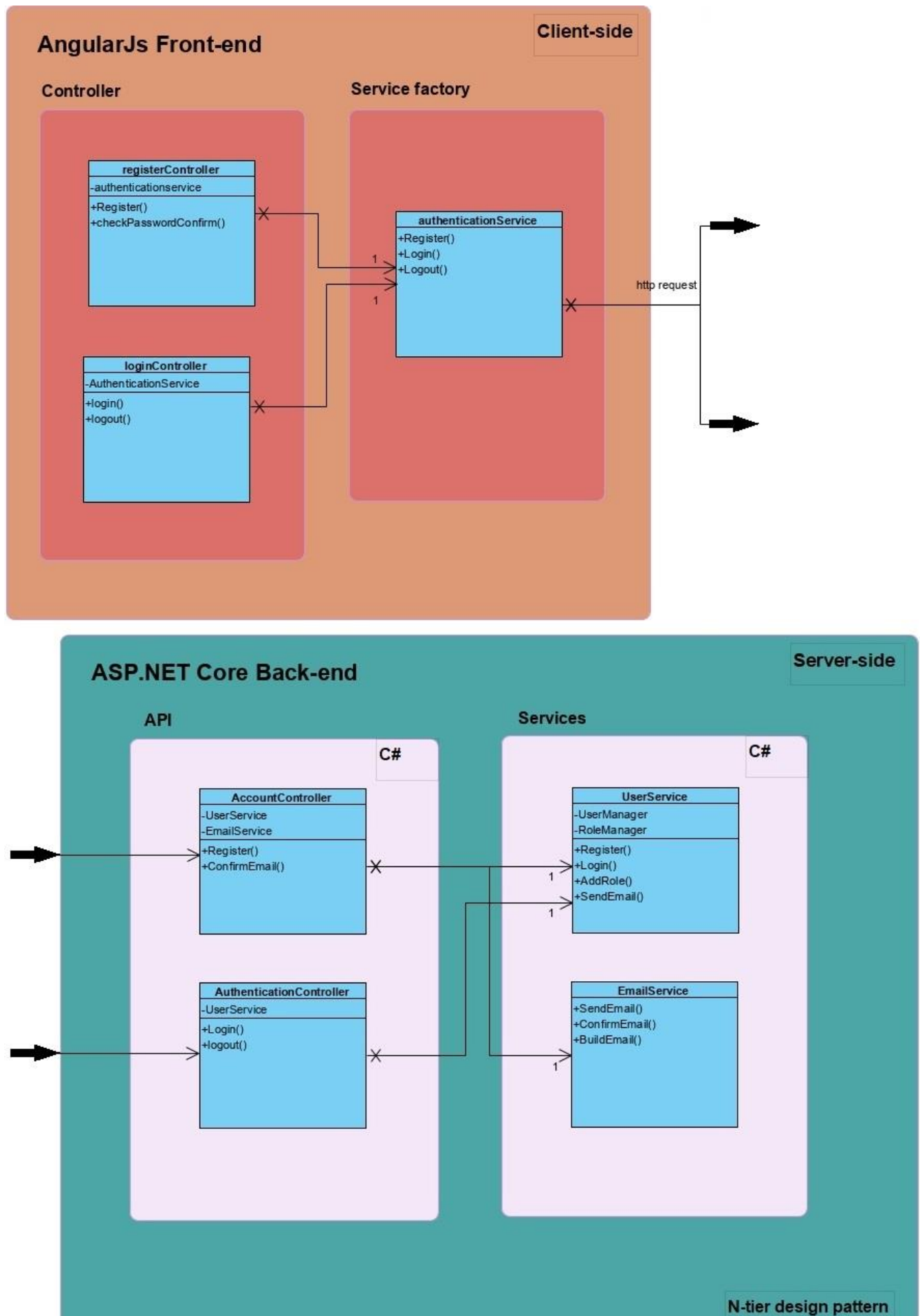
INFRASTRUCTUUR

Om ervoor te zorgen dat in de nieuwe situatie de DGMR-Rekentool webapplicatie goede code-kwaliteit beschikt is ervoor gekozen om de separation of concern (SoC) design principle toe te passen. Dit design principle zorgt ervoor dat de webapplicatie in verschillende secties wordt verdeeld. Elke sectie heeft vervolgens zijn eigen verantwoordelijkheid (zie Figuur 7 webapplicatie ontwerp 1). Separation of concern zorgt voor flexibiliteit in de webapplicatie, hierdoor is het makkelijker uit te breiden waardoor er ook minder code aangepast hoeft te worden wanneer er nieuwe functionaliteit gebouwd wordt. Daarnaast voorkomt het dat een bestaande functionaliteit niet meer werkt na een aanpassing in de code. Naast flexibiliteit zorgt het ook voor herbruikbaarheid. Zo kan er bijvoorbeeld in de toekomst een smartphone applicatie gemaakt worden voor de DGMR-Rekentool die weer de back-end API kan gebruiken van de webapplicatie.

Om deze design principle toe te passen is er besloten om de N-Tier closed layered structure design pattern te implementeren. Naast het feit dat hierdoor flexibiliteit bereikt kan worden verhoogt het ook de security van de webapplicatie. De security wordt verhoogd, doordat er gebruik gemaakt wordt van lagen. Hoe meer lagen hoe groter de afstand tussen de gebruiker en de database. Daarbij kan ook elke laag apart beveiligd worden. Door de N-tier design pattern is de webapplicatie ook schaalbaar, omdat per laag middelen kunnen worden toegevoegd zonder dat het invloed heeft op andere lagen. Het is een closed layered structure omdat elke laag alleen maar de laag onder zich kan aanroepen. In Figuur 7 webapplicatie ontwerp 1 zijn de lagen duidelijk in beeld gebracht. Zo is te zien dat op de front-end twee lagen zijn genaamd controller en service factory. In de controller laag worden alle front-end functionaliteiten gerealiseerd. Verder is de service factory een methode die aangeroepen kan worden vanuit de controller, deze is verantwoordelijk voor het versturen en ophalen van data naar en van de back-end. Verder bevat de back-end nu twee lagen en zal er in de aankomende sprints een derde laag bijkomen genaamd repositories. Deze laag wordt dan rechts van de back-end services laag geplaatst. De repositories-laag is dan verantwoordelijk voor het ophalen en opslaan van data uit en in de database. Daarnaast is de services laag aansprakelijk voor eventuele berekeningen of ophalen van diverse id's die nodig kunnen zijn in de repositories. De API laag is alleen verantwoordelijk voor het opvangen en doorsturen van requests die vanuit de front-end komen.

SECURITY

Voor security is de Identity Core van ASP.NET Core gebruikt. Dit is een ingebouwde security system van ASP.NET. Dit is gecombineerd met JWT Authenticatie, omdat voor een AngularJS client alleen Identity Core niet voldoende is. Met normale Identity van ASP.NET is het mogelijk om webpages te genereren met Razor markup. Deze zijn alleen niet van toepassing, omdat het geen normale HTML client is maar een AngularJS client. Om alsnog zoveel mogelijk gebruik te kunnen maken van de ingebouwde security van ASP.NET is niet Identity, maar Identity Core gebruikt. Hiermee kunnen de standaard user tabellen in de lokale database gegenereerd worden. Vervolgens kan met JWT Authenticatie op basis van token authenticatie geauthentiseerd en geautoriseerd worden. De back-end is verantwoordelijk voor het genereren van authenticatie tokens. Deze worden vervolgens opgeslagen in de localstorage van de front-end. Wanneer er een geauthentiseerde of geautoriseerde request moet worden gestuurd, wordt deze token met de request verstuurd. Vervolgens wordt de token op de back-end gecontroleerd.



Figuur 7 webapplicatie ontwerp 1

REALISATIE

Tijdens deze sprint is er voornamelijk tijd besteed aan het opzetten van de webapplicatie en implementeren van de authenticatie en autorisatie functionaliteit. Bij het opzetten van de webapplicatie is een ASP.net Core project aangemaakt in Visual Studio. Voor de authenticatie en autorisatie is er voor gekozen om token-based authenticatie te gebruiken in combinatie met ASP.net Identity Core, ASP.net en een AngularJS front end. Met token-based authenticatie is de client niet afhankelijk van specifieke authenticatie functionaliteit. De token wordt gegenereerd door de server en heeft ook weer de functionaliteit om de token te kunnen valideren en vervolgens te kunnen authentifieren. De token die wordt gegenereerd en gevalideerd is een JSON Web Token (JWT).

REVIEW

Aan het einde van de sprint is de review gehouden. Tijdens deze review kwamen dhr. Bosch en dhr. Schmidt samen om de resultaten te beoordelen. Er is een kleine demo gegeven van de functionaliteiten en uiterlijk van de webapplicatie. Ze waren tevreden over de voortgang van de webapplicatie. Daarnaast zijn alle userstories en taken afgerond. Verder hebben ze aangegeven wat ze de volgende keer wilden zien. Ze hadden ook een aantal opmerkingen over de html template. Voor deze webapplicatie is er gebruik gemaakt van een gratis html template. Een ander project waar DGMR Software aan werkt, gebruikt ook een html template. Het was beter als er voor de DGMR Rekentool dezelfde template gebruikt kon worden. Het implementeren van de nieuwe html template was een nieuwe user story voor de volgende sprint. De functionaliteiten die ze de volgende keer wilde zien waren de email confirmatie voor registrerende klanten, functionaliteit voor het aanmaken en opslaan van projecten en een onderscheid tussen betalende en niet betalende klanten. Met de email confirmatie functionaliteit, krijgen gebruikers die registreren een email naar hun emailadres waarmee zij hebben geregistreerd. In deze email moeten zijn op een link drukken om hun account te activeren om zo toegang te krijgen tot de webapplicatie.

RETROSPECTIVE

Bij de sprint retrospective was er besproken wat er goed ging. Dat was namelijk het netjes afronden van alle taken en usestories die van tevoren gedefinieerd waren tijdens de sprint planning. De uren waren goed van tevoren ingepland voor de taken. Hierdoor was er geen tijdnoed.

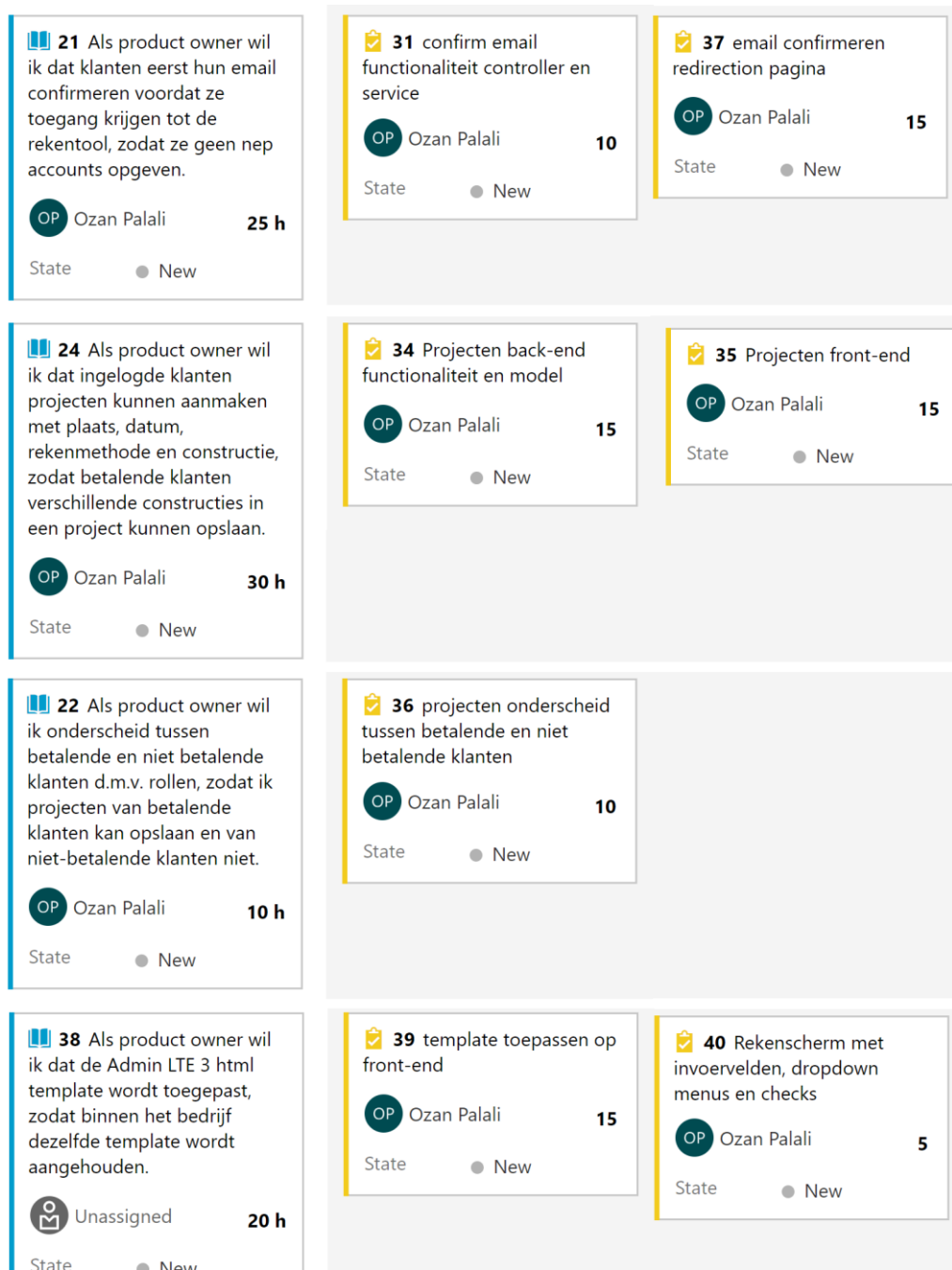
Wat er beter kon was het beter onderzoeken welke packages er nodig waren voor de webapplicatie. Wanneer er in een webapplicatie in Visual Studio gebruik gemaakt wordt van kant en klare functionaliteiten die geschreven zijn door andere mensen of bedrijven, wordt er gebruik gemaakt van de zogenaamde NuGet packages. Deze packages kunnen dan door iedereen gedownload en gebruikt worden via Visual Studio NuGet Package Manager. Een probleem wat zich hier voordeed was het downloaden van de package. Omdat er zoveel packages gedeeld worden via de package manager is het makkelijk om de verkeerde package te downloaden. Dit kan voor ongewenste problemen en gevolgen zorgen, zoals het niet werken van de functionaliteiten waarvoor de packages nodig waren.

Om zulke problemen te voorkomen gaat er voortaan eerst goed gekeken worden naar welke NuGet package er echt nodig is. Daarnaast moet er ook gecontroleerd worden of de auteur van de package klopt.

5.3 SPRINT 2 PROJECTEN REALISATIE

REQUIREMENTS & PLANNING

Tijdens de sprint planning is er besloten om de project functionaliteit te realiseren met daarbij de bijbehorende front-end pagina's. Daarnaast moet er ook een nieuwe HTML en CSS-template gebruikt worden die ook intern in het bedrijf wordt gebruikt namelijk: Admin LTE. Het toepassen van deze template zorgt ervoor dat in de toekomst wanneer de DGMR Rekentool wordt uitgebreid, het makkelijker uit te breiden is vanwege de aanwezige kennis over de Admin LTE-template. Verder moet er een onderscheid komen tussen betalende en niet-betalende klanten d.m.v. rollen. Ook moet het mogelijk zijn dat gebruikers die een nieuw account hebben geregistreerd, eerst hun email moeten confirmeren voordat ze toegang krijgen tot de rekentool. Ten slotte moet alvast een eerste opzet komen van een rekenscherf met diverse invoervelden.

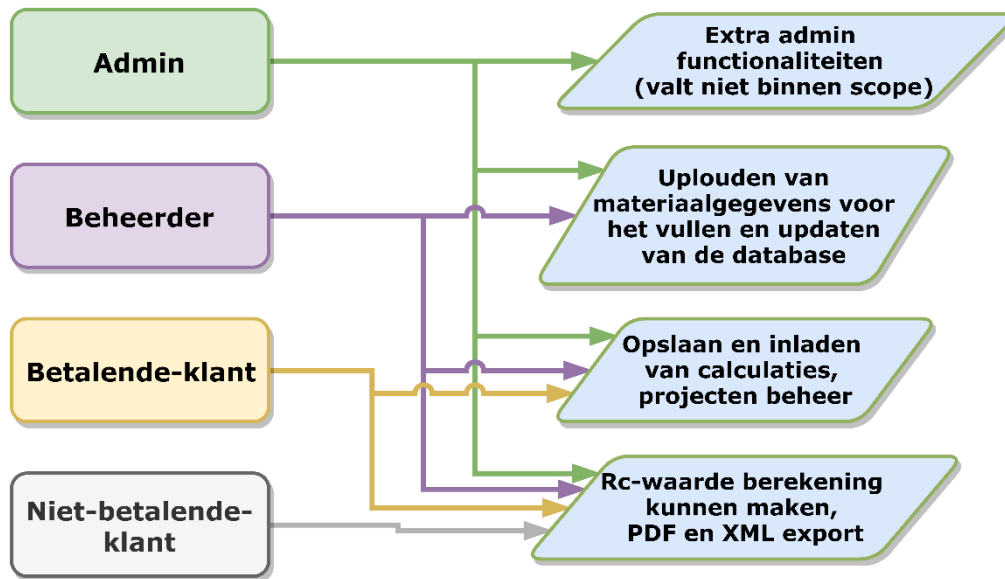


Figuur 8 sprint 2 backlog

ONTWERP

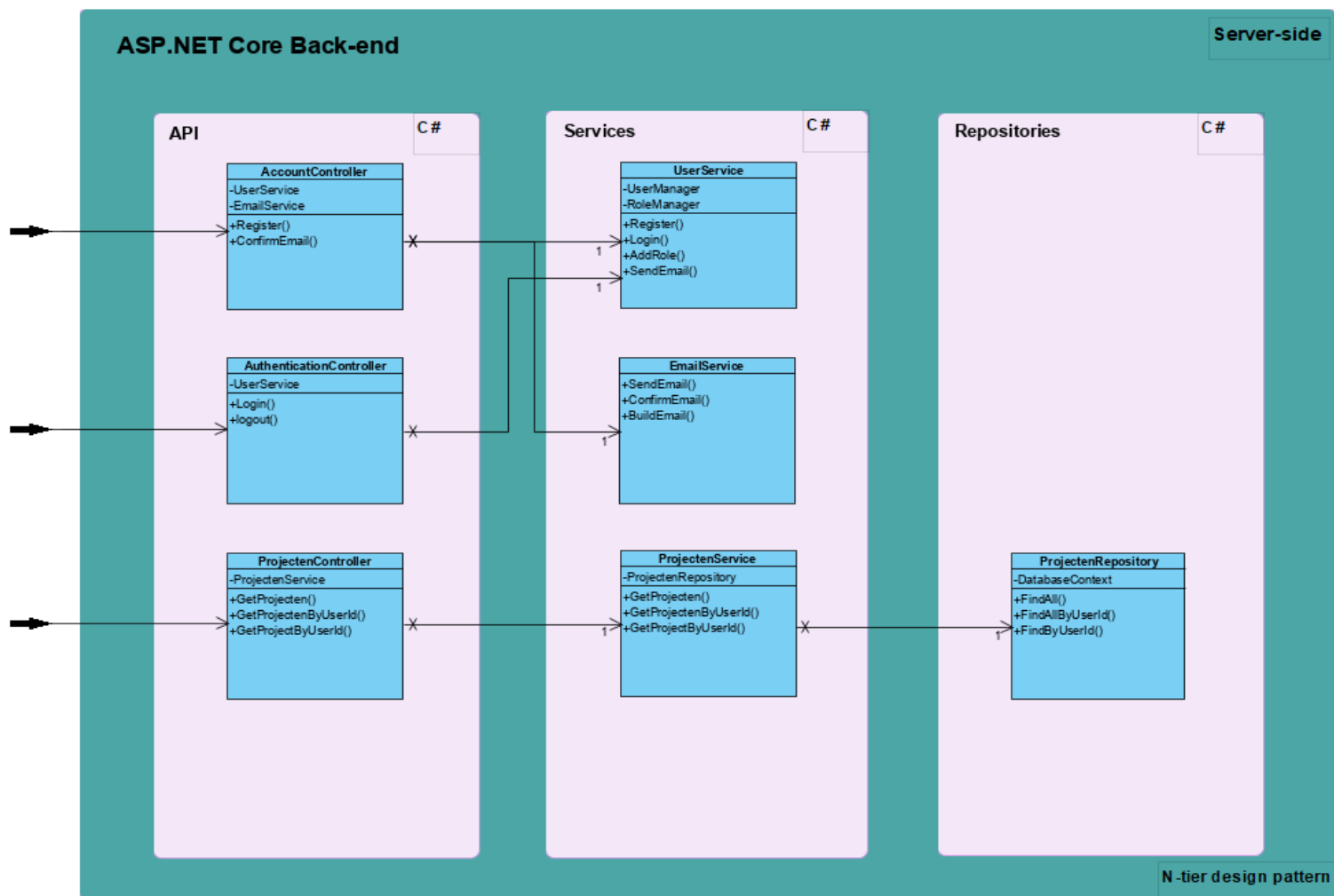
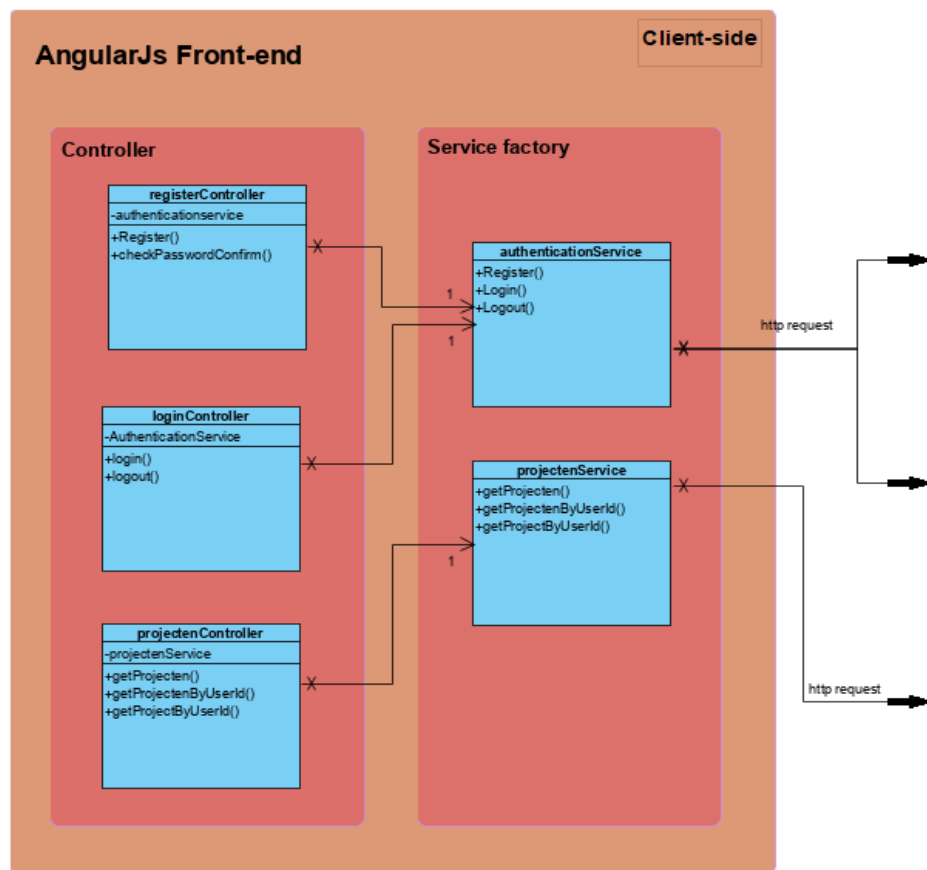
Naast het uitbreiden van de webapplicatie ontwerp is tijdens deze sprint ook het eerste ERD (Entity-Relationship-Diagram) deelontwerp van de database ontworpen. Dit model is bedoeld om het fysieke datamodel inzichtelijk te maken. Het is een visuele weergave van de entiteiten, relaties en regels die gelden of aanwezig zijn in de database. Ten opzichte van het webapplicatie ontwerp van de vorige sprint is nu te zien in *Figuur 10 webapplicatie ontwerp sprint 2* dat de repository laag is toegevoegd aan de back-end. De repositories zijn zoals eerder vermeld verantwoordelijk voor het ophalen en opslaan van data in de database.

Verder is er ook een ontwerp gemodelleerd om een overzicht te creëren van de verschillende rollen binnen de webapplicatie (zie *Figuur 9 Rollen diagram*).

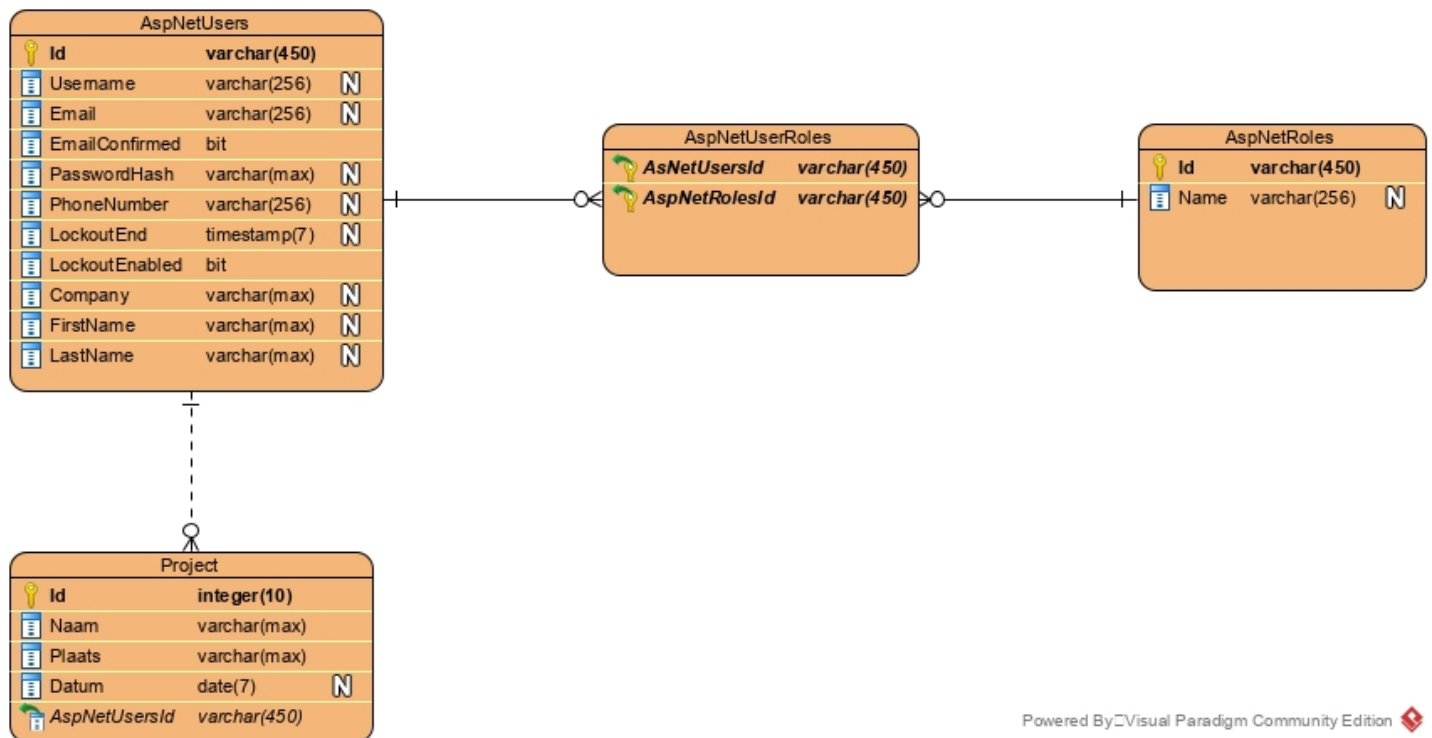


Figuur 9 Rollen diagram

In dit diagram hierboven is te zien dat de admin rol toegang heeft tot alle functionaliteiten binnen de webapplicatie. Onder de admin rol is de beheerder rol. Deze rol heeft toegang tot alle functionaliteiten behalve die van de admin. De beheerder is verantwoordelijk voor de materiaalgegevens in de database. De beheerder kan daarom dus materiaalgegevens uploaden via de webapplicatie. Hieronder valt vervolgens de betalende-klant. Een betalende klant kan ook alles wat een niet-betalende-klant kan. Daarbij kan een betalende-klant ook eigen projecten aanmaken en deze beheren. Ook kan een betalende-klant calculaties opslaan in projecten en deze vervolgens inladen. Een niet-betalende-klant kan een Rc-waarde berekening maken en daarbij ook deze exporteren naar PDF en XML.



Figuur 10 webapplicatie ontwerp sprint 2



Figuur 11 ERD 1

REALISATIE

Bij de realisatie is eerst de nieuwe html template genaamd adminLTE toegepast. Hier was er in het begin problemen mee, omdat de login pagina een andere lay-out bevat dan de dashboard pagina. Wanneer een gebruiker inlogt komt die op de dashboard pagina terecht. Om ervoor te zorgen dat de dashboard pagina en de login pagina verschillende lay-outs kunnen hebben moest er gewerkt worden met stateProvider van AngularJS in plaats van routeProvider. Deze twee providers zijn functionaliteiten van AngularJS zelf. Deze zorgen ervoor dat de URL's worden gelinkt met de html en javascript controllers van de webapplicatie. Nu was in de vorige sprint de routeProvider geïmplementeerd. Dit moest nu veranderd worden naar de stateProvider, omdat er met stateProvider parent en child relaties gelegd kunnen worden tussen de verschillende pagina's en met routeProvider niet. Deze parent en child relatie is van essentieel belang zodat er verschillende lay-outs toegepast kunnen worden binnen de webapplicatie. Verder is in deze sprint de projecten functionaliteit gerealiseerd. Nu kunnen betalende klanten via een CRUD (Create, Read, Update, Delete) pagina hun projecten beheren. De projecten worden opgeslagen en opgehaald vanuit de database.

REVIEW

Toen alle taken af waren, de regressie tests uitgevoerd waren en alle userstories voldeden aan de Definition of Done, is de sprint review gehouden. Tijdens de review is er een demo gegeven van de gerealiseerde functionaliteiten. Hier is vervolgens feedback op gegeven door dhr. Schmidt. Dhr. Schmidt vond dat de voortgang erg vlot ging en was tevreden over de uitgevoerde werkzaamheden. De beschreven userstories zijn correct gerealiseerd. Na de feedback is er besproken wat er voor de volgende sprint gerealiseerd moest worden. Naast feedback is er ook besproken dat er een code review gehouden moet worden in een van de aankomende sprints. De code review wordt samen met een programmeur van DGMR Software gehouden om de kwaliteit van de code te bepalen.

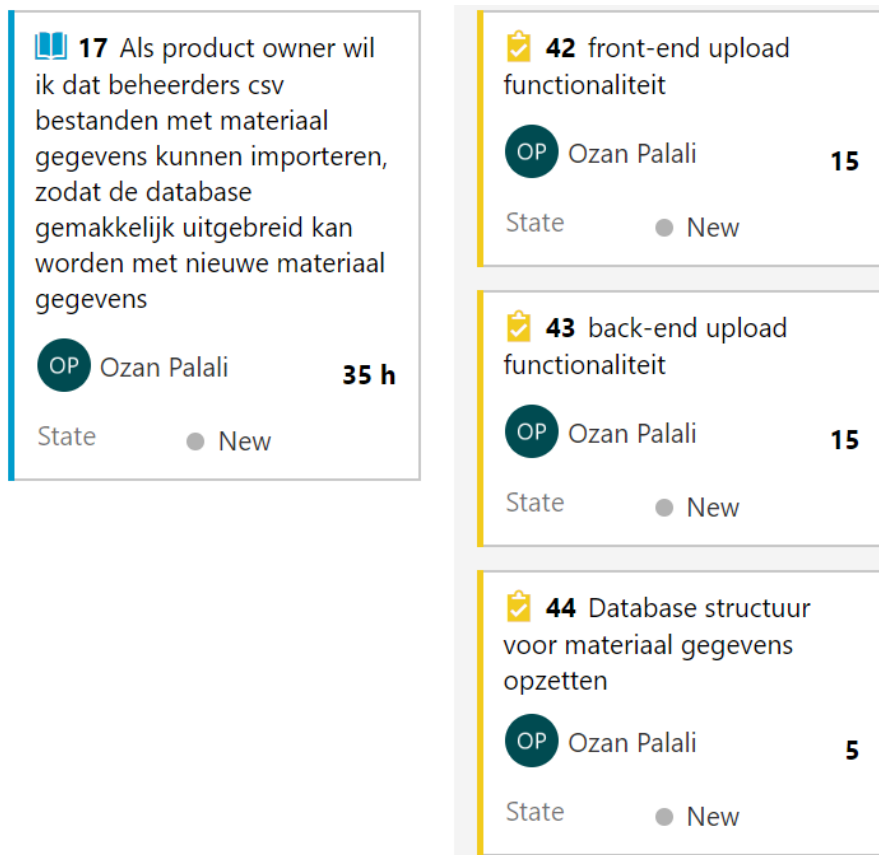
RETROSPECTIVE

Tijdens de retrospective is er besproken wat er goed ging, minder goed ging en hoe er voor gezorgd kan worden dat het de volgende keer beter gaat. Als eerste werd verteld door dhr. Schmidt en dhr. Bosch dat de realisatie van de webapplicatie erg vlot gaat. Ze vonden het vlot, omdat het realiseren van projectenbeheer, authenticatie en autorisatie in sprint 2 al zijn afgerond. Wat iets minder goed ging was punctualiteit. Om de punctualiteit te verbeteren is er afgesproken om voortaan eerder aanwezig te zijn.

5.4 SPRINT 3 DATABASE CONVERSIE

REQUIREMENTS & PLANNING

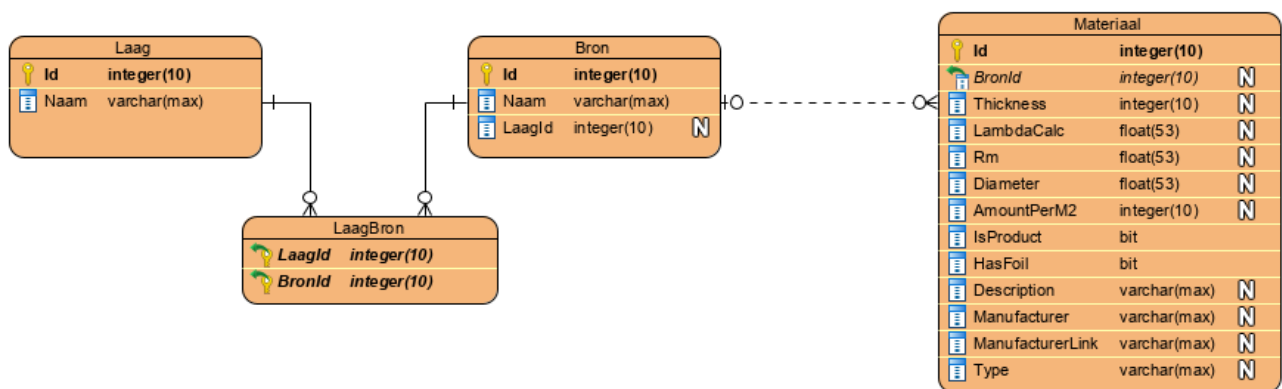
Tijdens sprint 3 moest de database conversie uitgevoerd worden. Omdat het moeilijk was om in te schatten hoelang de database conversie precies zou duren is er besloten om geen verdere userstories op te nemen voor deze sprint. Zodra deze userstory af was kon er besproken worden wat er verder gedaan kon worden tijdens de sprint. Verder is er besloten dat de database conversie niet iets eenmalig is maar een functionaliteit die vaker gebruikt kan worden door beheerders van de webapplicatie. Dit is van belang, omdat materiaalgegevens kunnen wijzigen. Deze wijzigingen moeten dan doorgevoerd kunnen worden in de webapplicatie.



Figuur 12 Sprint 3 backlog

ONTWERP

Nadat de Definition of Ready doorlopen was, is er direct begonnen aan de ontwerpfase. Voor het deelontwerp van deze sprint zijn de tabellen voor de database conversie gemodelleerd. De data die uit het csv-bestand komen, worden in deze tabellen opgeslagen (zie Figuur 13 ERD 2). Tijdens het ontwerpen was er eerst geen tussentabel gemodelleerd tussen Laag en Bron. Dit is er later pas bijgekomen. Over de relatie tussen laag en bron was er namelijk niet goed genoeg nagedacht. Aan de hand van de requirements die verzameld waren en de oude SBR-rekentool is er besloten om een één op meer relatie te modelleren van laag naar bron. Deze informatie vanuit de requirements waren niet voldoende om deze relatie vast te leggen. Wanneer er nog eens goed gekeken werd naar het csv-bestand met alle materiaalgegevens en relaties, is er gebleken dat de relatie tussen laag en bron een meer op meer relatie moest worden. Deze fout in het ontwerp heeft tijdens het realiseren voor vertraging gezorgd.



Figuur 13 ERD 2

REALISATIE

De data die geconverteerd moest worden zit allemaal in één CSV bestand. CSV staat voor comma separated values. Een CSV bestand is een soort database indeling. CSV bestanden kunnen geopend en bewerkt worden met Microsoft Excel. De data die in het CSV bestand zit, moest geconverteerd worden naar de lokale Microsoft SQL Server database van de webapplicatie. De database conversie scripts die opgeleverd moeten worden, zijn in deze sprint opgebouwd. Deze scripts zijn geschreven in Javascript in de materialsController van de front-end. Deze scripts zijn onderdeel van de upload functionaliteit. Het realiseren van de upload functionaliteit ging in het begin erg vlot. Wat een uitdaging was, was het ophalen van alle verschillende bronnen en lagen uit één csv-bestand. De bronnen en lagen moesten beiden in een aparte lijst opgeslagen worden. Verder moest ervoor gezorgd worden dat de twee lijsten geen dubbele gegevens, spaties, lege regels, hoofdletters of komma's bevatten. Nadat de lagen bronnen en materialen zodanig aangepast zijn dat ze de database in kunnen, worden ze via http requests verstuurd naar de back-end. Deze worden vervolgens via de back-end opgeslagen in de database. Eerst moest alle data uit het CSV bestand ingeladen worden en aan javascript lijsten toegekend worden in de materialsController. Om dit te bereiken is een javascript library genaamd Papa Parse gebruikt. Deze library maakt het mogelijk om onder andere CSV bestanden in te laden en toe te kennen aan lijsten in Javascript (zie Figuur 14 Dataconversie Papa Parse). Wanneer een beheerder het CSV bestand uploadt (zie Figuur 15 Dataconversie Upload CSV), begint de upload functionaliteit bij de \$scope.upload functie. Vervolgens wordt het bestand toegekend aan de "file" variabele. Daaropvolgend wordt Papa parse aangeroepen, die er weer voor zorgt dat het geüploade bestand aan de "csvArray" variabele toegekend wordt.

```

$scope.upload = function () {

    $scope.uploaded = false;
    $scope.failedUpload = false;

    var file = document.getElementById("csvFile");

    var uploadedFile = [];

    if (file.value.length > 0) {

        var csvFile = file.files[0];

        Papa.parse(csvFile, {
            download: true,
            header: true,
            encoding: "ISO-8859-1",
            skipEmptyLines: true,
            error: function (err, csvFile, inputElem, reason) { },
            complete: function (results) {
                uploadedFile.push(results.data);

                var csvArray = uploadedFile[0];
            }
        });
    }
}

```

Figuur 14 Dataconversie Papa Parse

Upload csv bestand

Choose File No file chosen

Upload

Figuur 15 Dataconversie Upload CSV

In Figuur 16 Dataconversie spaties corrigeren, wordt er voor het eerst door de data van het CSV bestand gelust. In deze lus worden alle spaties en lege velden die in het CSV bestand zitten gevuld. Deze lege velden en spaties moesten vervangen worden, omdat één materiaal vaak meerdere rijen bevat in het CSV bestand waarvan alleen de dikte van het materiaal verschilt. In de `forEach` lus van Figuur 16 Dataconversie spaties corrigeren, worden al deze spaties gedetecteerd en verbeterd.

Vervolgens wordt er weer door de data van het CSV bestand heen gelust, alleen dit keer met de gecorrigeerde spaties. In deze lus worden alle lagen, bronnen en materialen van elkaar onderscheiden en worden ze aan aparte lijsten toegekend. Verder worden alle kommagetallen gecorrigeerd, zodat deze correct worden opgeslagen in de database. Daarnaast worden ook nog eens alle overgebleven spaties en lege regels gecorrigeerd. Wanneer de materialen, lagen en bronnen gereed zijn om opgenomen te worden in de database, worden deze verstuurd naar de back-end. In Figuur 17 Dataconversie HTTP requests is te zien dat de materialen (materials), lagen (layers) en bronnen (sources), door middel van HTTP requests naar de back-end worden verstuurd.

```

csvArray.forEach(function (item, index) {

    if (item.type === "" && item.manufacturer === "" && item.manufacturer_link === "" && item.supplier === ""
        && item.walltie_amount === "" && item.walltie_diameter === "" || item.type === " ") {

        item.article_number = csvArray[index - 1].article_number;
        item.density = csvArray[index - 1].density;
        item.description = csvArray[index - 1].description;
        item.ean_code = csvArray[index - 1].ean_code;
        item.has_foil = csvArray[index - 1].has_foil;
        item.heat_capacity = csvArray[index - 1].heat_capacity;
        item.manufacturer = csvArray[index - 1].manufacturer;
        item.manufacturer_link = csvArray[index - 1].manufacturer_link;
        item.material_link = csvArray[index - 1].material_link;
        item.rvalue = csvArray[index - 1].rvalue;
        item.supplier = csvArray[index - 1].supplier;
        item.thermal_conductivity = csvArray[index - 1].thermal_conductivity;
        item.type = csvArray[index - 1].type;
        item.walltie_amount = csvArray[index - 1].walltie_amount;
        item.walltie_diameter = csvArray[index - 1].walltie_diameter;
    }
    else if (item.type === "" && item.manufacturer === "" && item.manufacturer_link === ""
        && item.supplier === "" || item.type === " ") {

        item.article_number = csvArray[index - 1].article_number;
        item.density = csvArray[index - 1].density;
        item.description = csvArray[index - 1].description;
        item.ean_code = csvArray[index - 1].ean_code;
        item.has_foil = csvArray[index - 1].has_foil;
        item.heat_capacity = csvArray[index - 1].heat_capacity;
        item.manufacturer = csvArray[index - 1].manufacturer;
        item.manufacturer_link = csvArray[index - 1].manufacturer_link;
        item.material_link = csvArray[index - 1].material_link;
        item.rvalue = csvArray[index - 1].rvalue;
        item.supplier = csvArray[index - 1].supplier;
        item.thermal_conductivity = csvArray[index - 1].thermal_conductivity;
        item.type = csvArray[index - 1].type;
        item.thickness = csvArray[index - 1].thickness;
    }

});

```

Figuur 16 Dataconversie spaties corrigeren

```

var _postMaterials = function (materials) {

    return $http({
        method: "post",
        url: "api/materials",
        headers: { 'Content-Type': 'application/json'},
        data: JSON.stringify(materials)
    });
};

var _postLayers = function (layers) {

    return $http({
        method: "post",
        url: "api/layers",
        headers: { 'Content-Type': 'application/json' },
        data: JSON.stringify(layers)
    });
};

var _postSources = function (sources) {

    return $http({
        method: "post",
        url: "api/sources",
        headers: { 'Content-Type': 'application/json' },
        data: JSON.stringify(sources)
    });
};

```

Figuur 17 Dataconversie HTTP requests

De volledige conversiescripts kunnen teruggevonden worden in Bijlage F – Conversie Scripts.

Wanneer de database gevuld is met de gegevens van de bronnen, lagen en materialen worden daaropvolgend de relaties gelegd. De relaties worden gelegd door eerst door alle materialen heen te lopen en vervolgens per materiaal de bijbehorende bron en laag op te halen. Per materiaal wordt daarna een LaagBron aangemaakt (zie Figuur 13 ERD 2). De laagbron wordt per materiaal toegevoegd aan een lijst. Wanneer de loop klaar is worden de laagbronnen opgeslagen in de database. Op deze manier kunnen nu per laag en per bron de bijbehorende materialen opgehaald worden.

REVIEW

Met deze sprint is uiteindelijk de database conversie gerealiseerd. Met de functionaliteit dat is gerealiseerd tijdens deze sprint kan er een csv-bestand geüpload worden via de front-end die vervolgens de database vult met materiaalgegevens. Naast dat de database wordt gevuld worden ook de relaties tussen de materiaalgegevens gelegd. Elke keer wanneer een csv-bestand wordt geüpload worden eerst automatisch alle bestaande materiaalgegevens en relaties verwijderd uit de database voordat de database wordt gevuld met nieuwe materiaalgegevens en relaties. Deze functionaliteit werd gedemonstreerd tijdens de review. Tijdens de review waren dhr. Schmidt en dhr. Bosch tevreden over deze functionaliteit, omdat het namelijk zorgde voor gebruiksvriendelijkheid. Beheerders van deze rekentool hoeven in de toekomst dan alleen nog maar een csv-bestand te uploaden om de database bij te werken met nieuwe en of gewijzigde materiaalgegevens. Zonder al te veel kennis te beschikken over de databasestructuur.

RETROSPECTIVE

Wat er goed ging was communicatie tijdens deze sprint, wanneer er onduidelijkheden waren, werd er een en ander opgehelderd door dhr. Bosch en dhr. Schmidt. Dit ging om de CSV upload functionaliteit. Dit kwam omdat de inhoud van het CSV bestand niet altijd even helder was. Het CSV bestand bevatte namelijk veel termen en afkortingen waar nog niet bekend mee was. Om ervoor te zorgen dat het CSV bestand goed werd geconverteerd naar de huidige database, moest ervoor gezorgd worden dat deze onduidelijkheden verduidelijkt werden. Dankzij de goede communicatie waren de onduidelijkheden verduidelijkt.

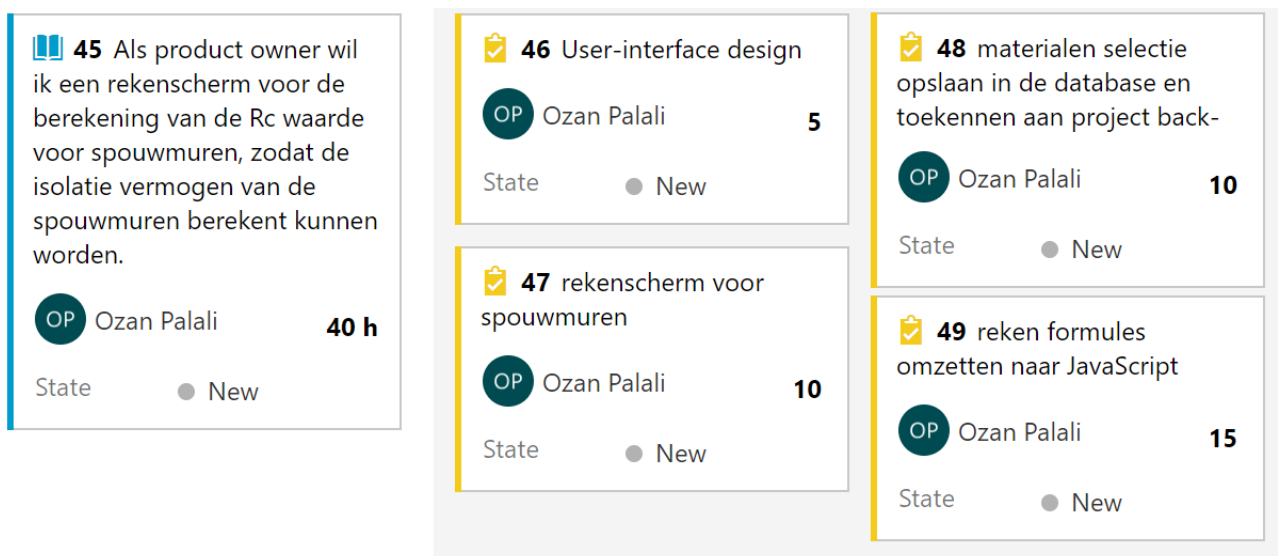
Wat er ook goed ging was het inlezen van het csv-bestand. Deze had namelijk veel aanpassingen nodig voordat het opgeslagen kon worden in de database. Deze aanpassingen zoals het corrigeren van spaties en lege regels waren erg vlot doorgevoerd.

Wat er minder goed ging was dat de database conversie toch langer duurde dan verwacht, waardoor er geen additionele userstories opgenomen konden worden gedurende de sprint. De reden dat het langer duurde was, omdat de functionaliteiten moeilijk te realiseren waren aan de hand van het ontwerp en er een gebrek was aan vaardigheden en kennis met het opzetten van een meer op meer relatie.

5.5 SPRINT 4 RC -WAARDE BEREKENING

REQUIREMENTS & PLANNING

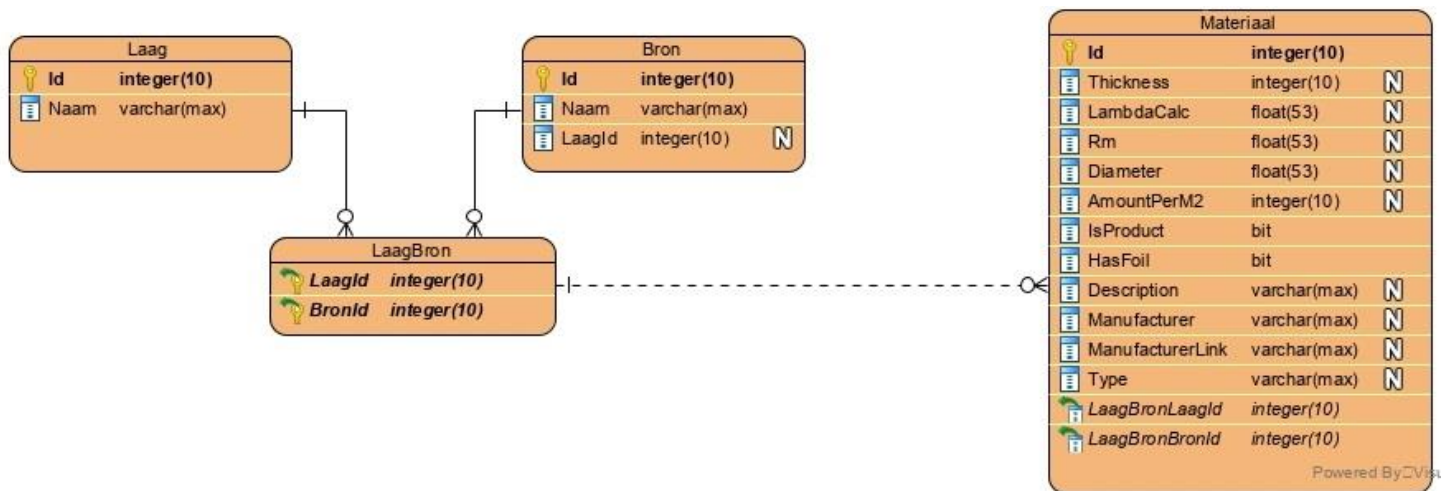
Dhr. Schmidt en dhr. Bosch wilden eerst een userinterface design zien voordat er aan de R_c -waarde berekening pagina werd gewerkt. In de userinterface moet te zien zijn waar de verschillende waardes, invoervelden en drop down menu's komen te staan. Naast de userinterface moeten ook de rekenformules vanuit de oude rekentool die in PHP geschreven zijn omgezet worden naar Javascript. Verder moet er samen met een programmeur van DGMR Software gekeken worden naar de kwaliteit van de code. De kwaliteit van de code wordt beoordeeld tijdens een code review met dhr. Biemond. Vervolgens moet ook de rekenpagina gerealiseerd worden.



Figuur 18 Sprint 4 backlog

ONTWERP

Voor het berekenen van de R_c -waarde was het deelontwerp van Figuur 13 ERD 2 gebruikt van de vorige sprint. Dat deelontwerp had een één op meer associatie van bron naar materiaal zodat er per bron en laag bijgehouden kon worden welke materialen daarbij hoorden. Dit gewenste resultaat was helaas niet mogelijk met het deelontwerp van de vorige sprint. Met dat deelontwerp konden er namelijk alleen maar materialen bijgehouden worden per bron en niet per bron en per laag. Hierdoor was er bij het selecteren van een bron bij een laag een te grote materialen lijst die alle materialen weergaven die bij de geselecteerde bron hoorde. Om ervoor te zorgen dat er wel per bron en per laag materialen bijgehouden konden worden, moest de associatie van Bron naar Materiaal verplaatst worden naar van LaagBron naar materiaal (zie Figuur 19 ERD 3). Achteraf was dit een logische oplossing. Alleen was ervan uitgegaan dat het mogelijk was met het deelontwerp van de vorige sprint, wat dus niet mogelijk was.



Figuur 19 ERD 3

REALISATIE

Voor het berekenen van de R_c -waarde moet het volgende stappenplan doorlopen worden. De informatie die is verkregen om deze stappenplan te verwerken in dit verslag is tot stand gekomen uit onderzoek naar hoe een R_c -waarde berekent wordt. Hiervoor zijn er documenten beschikbaar gesteld door dhr. Bosch om deze berekening beter te begrijpen.

1. Berekenen van de warmteweerstand van elke laag.
2. Berekenen van de totale warmteweerstand R_T .
3. Berekenen van de warmtedoorgangscoefficiënt U_T .
4. Bepalen van de toeslagfactor ΔU .
5. 3% controle ΔU .
6. Bepalen van de warmtedoorgangscoefficiënt U_c .
7. Berekenen van de warmteweerstand R_c .
8. Presentatie van gegevens.

In de tabel hieronder wordt er een voorbeeld spouwmuur constructie getoond met de geselecteerde materialen en diktes. Het stappenplan hierboven wordt vervolgens stap voor stap doorlopen om de R_c -waarde berekening van de webapplicatie te simuleren, aan de hand van de gegevens uit deze tabel.

Laag	Materiaal	Dikte (mm)	λ_{Calc} (W/m ² K)
1. Binnenspouwblad	Kalkzandsteen	100	0,90
2. Isolatie	Isolatiemateriaal	170	0,035
3. Spouwanker	Spouwanker RVS Diameter: 4mm, aantal per m ² : 4 stuks	30	15
4. Spouw	Spouw (Sterk geventileerd)	30	
5. Buitenspouwblad	Metselwerk	100	1,0

STAP 1: BEREKENEN VAN DE WARMTEWEERSTAND VAN DE LAGEN

Eerst moet per materiaal laag de warmte weerstand (R_m) bepaald worden. Deze wordt als volgt bepaald:

$$R_{m,i} = \frac{d_i}{\lambda_{calc,i}} \quad [(m^2K)/W]$$

Waarbij geldt dat d_i de dikte is van het materiaal en λ_{Calc} de warmtegeleidingscoëfficiënt.

STAP 2: BEREKENEN VAN DE TOTALE WARMTEWEERSTAND

Wanneer vervolgens de warmteweerstand van alle materiaal lagen zijn berekend, wordt de totale warmteweerstand (R_T) uitgerekend.

Hier is te zien dat eerst de R_{si} wordt opgeteld en vervolgens alle warmteweerstanden (R_m) van elke materiaal laag. Als laatste wordt de R_{se} opgeteld. R_{si} en R_{se} zijn warmte overgangsweerstanden, die van toepassing zijn op oppervlakten die in contact komen met lucht. Hierbij geldt R_{si} voor binnenwerkse oppervlakten (in dit geval binnenspouwblad) en R_{se} voor buitenwerkse oppervlakten (in dit geval buitenspouwblad). Wanneer alle oppervlakten van een materiaal een ander materiaal raken is de overgangsweerstand 0. De waarde van de overgangsweerstanden zijn afhankelijk van de warmtestroomrichting. Deze waarden kunnen uit Figuur 20 Warmte overgangsweerstanden hieronder afgeleid worden. Omdat een spouwmuur altijd horizontaal staat gelden de waarden voor de overgangsweerstanden van de kolom genaamd horizontaal.

Warmteovergangsweerstand $m^2 \cdot K/W$	Richting van de warmtestroom		
	Omhoog	Horizontaal	Omlaag
R_{si}	0,10	0,13	0,17
R_{se}	0,04	0,04	0,04

OPMERKING 1 De warmteovergangsweerstanden zijn van toepassing op oppervlakten in contact met lucht. Raken oppervlakten een ander materiaal is de warmteovergangsweerstand = 0.

OPMERKING 2 De waarden voor de warmteovergangsweerstand van binnenwerkse oppervlakten zijn berekend voor $\varepsilon = 0,9$ en met h_{r0} bij 20 °C. De waarde waarden voor de warmteovergangsweerstand van buitenwerkse oppervlakten zijn berekend voor $\varepsilon = 0,9$, met h_{r0} bij 10 °C en $v = 4$ m/s.

Figuur 20 Warmte overgangsweerstanden

$$R_T = R_{si} + \sum_i (R_{m,i}) + R_{se} \quad [(m^2K)/W]$$

R_T wordt dan als volgt bepaald:

$$R_T = R_{si} + R_{m,1} + R_{m,2} + R_{m,4} + R_{m,5} + R_{se} \quad [(m^2K)/W]$$

Omdat $R_{m,4}$ in deze situatie een spouw is wordt deze verder aangeduid met R_{cav} . Verder wordt laag 3 (de spouwanker) niet meegenomen in de totale warmteweerstand berekening, omdat hier andere regels voor gelden. In Figuur 20 Warmte overgangsweerstanden is te zien dat de R_{se} voor horizontale oppervlaktes zoals die van een spouwmuur 0,04 is, maar omdat de spouw van deze spouwmuur sterk geventileerd is geldt er R_{se} is 0,13 en niet 0,04. Ook is te zien dat bij de optelling de $R_{m,5}$ 0 is. Wanneer een spouw sterk geventileerd is, wordt de warmteweerstand van de buitenste laag buiten beschouwing gelaten. Dit is de reden waarom $R_{m,5}$ 0 is.

$$R_T = R_{si} + \frac{d_1}{\lambda_{calc;1}} + \frac{d_2}{\lambda_{calc;2}} + R_{cav;sv} + \frac{d_5}{\lambda_{calc;5}} + R_{se} \quad [(m^2K)/W]$$

$$R_T = 0,13 + \frac{0,1}{0,9} + \frac{0,170}{0,035} + 0,13 + 0 + 0,13 = 5.35825396825 \quad [(m^2K)/W]$$

STAP 3: BEREKENEN VAN DE WARMTEDOORGANGSCOËFFICIËNT VAN TOTALE CONSTRUCTIE U_T

De totale warmtedoorgangscoefficiënt van een constructie U_T drukt de hoeveelheid warmte uit die per seconde (s) per vierkante meter (m^2) en per graad temperatuurverschil (K) tussen de ene en de andere zijde van een wand(constructie) doorgelaten wordt.

$$U_T = \frac{1}{R_T} = \frac{1}{5.35825396825} = 0.18662795864 \quad [W/(m^2K)]$$

STAP 4: BEPALEN VAN DE TOESLAGFACTOR ΔU

ΔU is de toeslagfactor voor eventuele convectie, ankers en regenwater/drainage in $W/(m^2K)$.

$$\Delta U = \Delta U_a + \Delta U_{fa} + \Delta U_r \quad [W/(m^2K)]$$

U_a is toeslagfactor voor convectie (in deze situatie 0)

U_{fa} is toeslagfactor bevestigingshulpmiddelen

U_r is toeslagfactor voor een omgekeerd dak (in deze situatie 0)

De formule voor ΔU_{fa} is:

$$\alpha_{fa} = \left(0,8 \times \frac{d_{fa}}{d_{iso}}\right) \times \frac{(n_{fa} \times \lambda_{fa} \times A_{fa})}{d_{iso}}$$

De n_{fa} hierboven is het aantal spouwankers per vierkante meter en de A_{fa} is de diameter van de spouw in meters. In dit geval is het aantal spouwankers 4 en de diameter 0.004 in meters. d_{iso} is de dikte van de isolatielaag in meters. d_{fa} is de indringingsdiepte van het bevestigingshulpmiddel in meters, wat voorlopig gelijk zal zijn aan d_{iso} .

$$\alpha_{fa} = \left(0,8 \times \frac{0,170}{0,170}\right) \times \frac{\left(4 \times 15 \times \frac{1}{4} \times \pi \times 0,004^2\right)}{0,170} = 0,00354815$$

$$\Delta U_{fa} = \alpha_{fa} \times \left(\frac{R_1}{R_T}\right)^2 = 0,00354815 \times \left(\frac{4,85714}{5,35825396825}\right)^2 = 0.00291552403 \text{ W}/(m^2K)$$

De hierboven weergegeven waarde R_1 is in het voorbeeld de warmteweerstand van de isolatie laag (tweede laag), $R_{m;2}$.

$$\Delta U = \Delta U_a + \Delta U_{fa} + \Delta U_r \quad [W/(m^2K)]$$

$$\Delta U = 0 + 0,00291552403 + 0 = 0,00291552403 \text{ W}/(m^2K)$$

STAP 5: 3% CONTROLE ΔU

Bij de 3% controle geldt dat wanneer ΔU kleiner is dan 3% van U_T , dat ΔU 0 wordt. Als dit niet het geval is blijft ΔU onaangepast.

$$U_T = 0,18662795864 \text{ [W/(m}^2\text{K)]}$$

$$\Delta U = 0,00291552403 \text{ [W/(m}^2\text{K)]}$$

$$3\% * U_T = 0,00559884 \text{ [W/(m}^2\text{K)]}$$

$$0,00291552 < 0,00559884 \text{ dus } \Delta U = 0 \text{ [W/(m}^2\text{K)]}$$

Hierboven is te zien dat ΔU kleiner is dan 3% van U_T , dus geldt $\Delta U = 0 \text{ [W/(m}^2\text{K)]}$.

STAP 6: BEREKENEN VAN U_c

U_c is de warmtedoorgangscoefficiënt van een onderdeel.

f_{prac} is de praktijk-prestatiefactor voor ondoorschijnende constructiedelen, waarvoor geldt $f_{\text{prac}} = 1$.

$$U_c = \frac{U_t}{f_{\text{prac}}} + \Delta U \quad [\text{form 8.4}]$$

$$f_{\text{prac}} = 1$$

$$U_c = \frac{0,1912684}{1} + 0 = 0,1912684 \text{ [W/(m}^2\text{K)]}$$

STAP 7: BEREKENEN VAN DE WARMTEWEERSTAND R_c

$$R_c = \frac{1}{U_c} - R_{\text{si}} - R_{\text{se}} = \frac{1}{0,1912684} - 0,13 - 0,13 = 4,96825 = 4,97 \text{ [(m}^2\text{K)/W]}$$

Deze complete berekening van de R_c -waarde is een uitgebreid voorbeeld van hoe de R_c -waarde wordt berekend in de webapplicatie. Echter zijn alle berekeningen situatie afhankelijk en kunnen er verschillende aspecten een rol spelen. Zo zou er voor een zwak geventileerde spouw andere regels gelden en dat geldt ook voor wanneer bijvoorbeeld een isolatielaag een reflecterende folie bevat. Deze berekening is over het algemeen altijd van toepassing op de berekening van de R_c -waarde, alleen zijn de materiaalgegevens en diktes telkens dynamisch, wat voor verschillende resultaten zorgt.

CODE REVIEW

Met dhr. Biemond is er gekeken naar de kwaliteit van de code. Volgens dhr. Biemond is de structuur van de code duidelijk en overzichtelijk. Verder moest er opgelet worden voor dubbele code, omdat er op enkele plaatsen nog dubbele code te zien was. Er moest ook opgelet worden voor de async methodes van ASP.NET, omdat er in de code daar veel gebruik van wordt gemaakt. Het gebruik van async methodes is een goede functionaliteit van ASP.NET en is het zeer aangeraden om hier gebruik van te maken, aangezien dit de performance van de webapplicatie verbetert. De performance wordt verbeterd doordat methodes en taken asynchroon uitgevoerd kunnen worden als er async voor staat. Wat dhr. Biemond duidelijk heeft gemaakt is dat het niet altijd goed is als een methode async gemaakt wordt, omdat sommige async methodes aan bepaalde voorwaarden moeten voldoen. Wanneer niet aan deze voorwaarden voldaan zijn kunnen er veel problemen ontstaan. Om ervoor te zorgen dat er goed gebruik wordt gemaakt van de async functionaliteit raad dhr. Biemond aan om eerst naar de Microsoft documentatie te kijken. Daarnaast moet er rekening gehouden worden met de URL requests die gestuurd worden naar de back-end. Wanneer de webapplicatie online komt te staan verandert namelijk de URL. Hier moet een dynamische oplossing voor komen wat ervoor zorgt dat de URL's zowel online als lokaal functioneren.

REVIEW

Tijdens de review is samen met dhr. Schmidt en dhr. Bosch gekeken naar de voortgang. De R_c berekening pagina was qua functionaliteit en design volledig gerealiseerd. Het enige wat nog niet functioneerde volgens wens, was de daadwerkelijke R_c -waarde berekening. Om dit goed te krijgen moet er met veel aspecten en voorwaarden rekening gehouden worden. Deze aspecten en voorwaarden staan geformuleerd in de Nederlandse technische afspraak NTA 8800 Energieprestatie van gebouwen – bepalingmethode. Verder vonden dhr. Schmidt en dhr. Bosch dat de pagina van de R_c -waarde berekening er mooi en modern uit zag (zie Figuur 31 Rekenpagina). Tijdens de planning van deze sprint was er ook afgesproken dat de selectie van de materialen voor de berekening opgeslagen moeten kunnen worden als het berekenen van de R_c -waarde eerder klaar was. Omdat het berekenen van de R_c -waarde veel aandacht en nauwkeurigheid vergt, is er besloten om de taak voor het opslaan van berekeningen mee te nemen naar de volgende sprint. Dat geldt ook voor de taak voor het verbeteren van de R_c -waarde berekening. Userstory nummer 45 wordt dus samen met de resterende taken meegenomen naar de volgende sprint.

RETROSPECTIVE

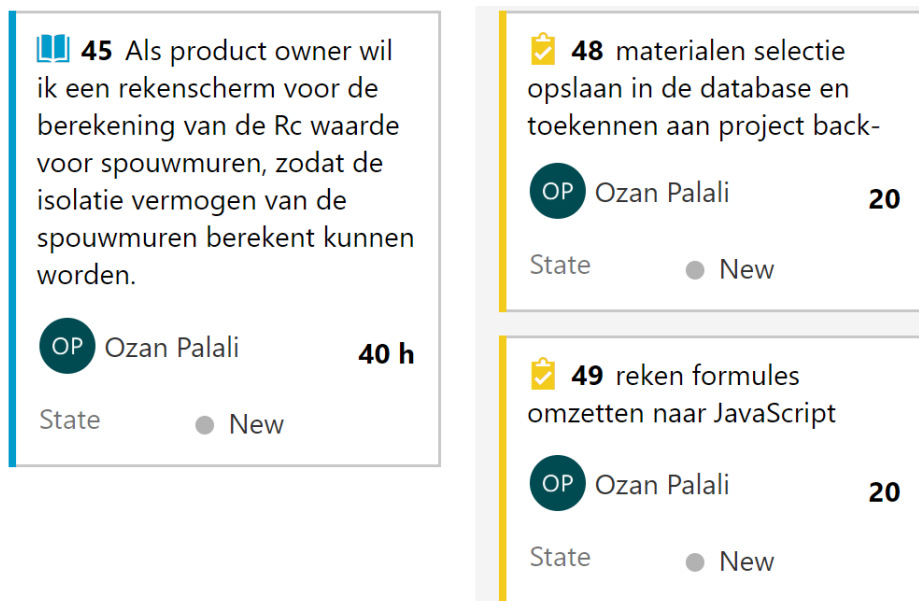
Dhr. Schmidt en dhr. Bosch waren erg tevreden over de voortgang van de webapplicatie. Dhr. Schmidt vond ook aan de hand van de code review met dhr. Biemond dat de code een goede structuur heeft. Visueel vonden ze ook dat de R_c -waarde bereken pagina er modern uit zag en voldeed aan de verwachtingen. Alleen de daadwerkelijke R_c -waarde stond nog niet op de juiste plek. Daar moest nog een goede plek voor bedacht worden.

Wat er minder goed ging was de R_c -waarde berekening. Bij deze berekening horen namelijk veel regels zoals dat wanneer een bepaalde laag bijvoorbeeld sterk geventileerd is dat andere lagen niet meer van toepassing zijn bij het berekenen van de R_c -waarde. Dit is een klein voorbeeld van de soort regels die verwerkt moeten worden in de berekening. Zo zijn er nog talloze andere regels waar de berekening aan moet voldoen. Deze regels moeten nog goed verwerkt worden in de code om ervoor te zorgen dat de R_c -waarde correct berekend wordt.

Om ervoor te zorgen dat het de volgende keer beter gaat, gaat er gebruik gemaakt worden van de Nederlandse technische afspraak NTA 8800 Energieprestatie van gebouwen – bepalingmethode en een validatie document van DGMR Software voor het berekenen van de R_c -waarde in verschillende scenario's.

REQUIREMENTS & PLANNING

Tijdens de sprint planning was er besloten om niet te veel userstories op te nemen i.v.m. het inleveren van het concept afstudeerdossier. Wat belangrijk was, was het verbeteren van de R_c -waarde berekening en het opslaan ervan. Om de berekening te verbeteren moest er nog eens goed gekeken worden naar de oude sbr-rekentool. Daarnaast moest er ook gebruik gemaakt worden van het validatie document voor het berekenen van de R_c -waarde van DGMR Software. Hierin staan stapsgewijs de berekeningen die doorlopen moeten worden voor het bepalen van de R_c -waarde. Naast het validatie document is ook de Nederlandse technische afspraak NTA 8800 Energieprestatie van gebouwen – bepalingsmethode beschikbaar gesteld om verdere onduidelijkheden te verhelpen.

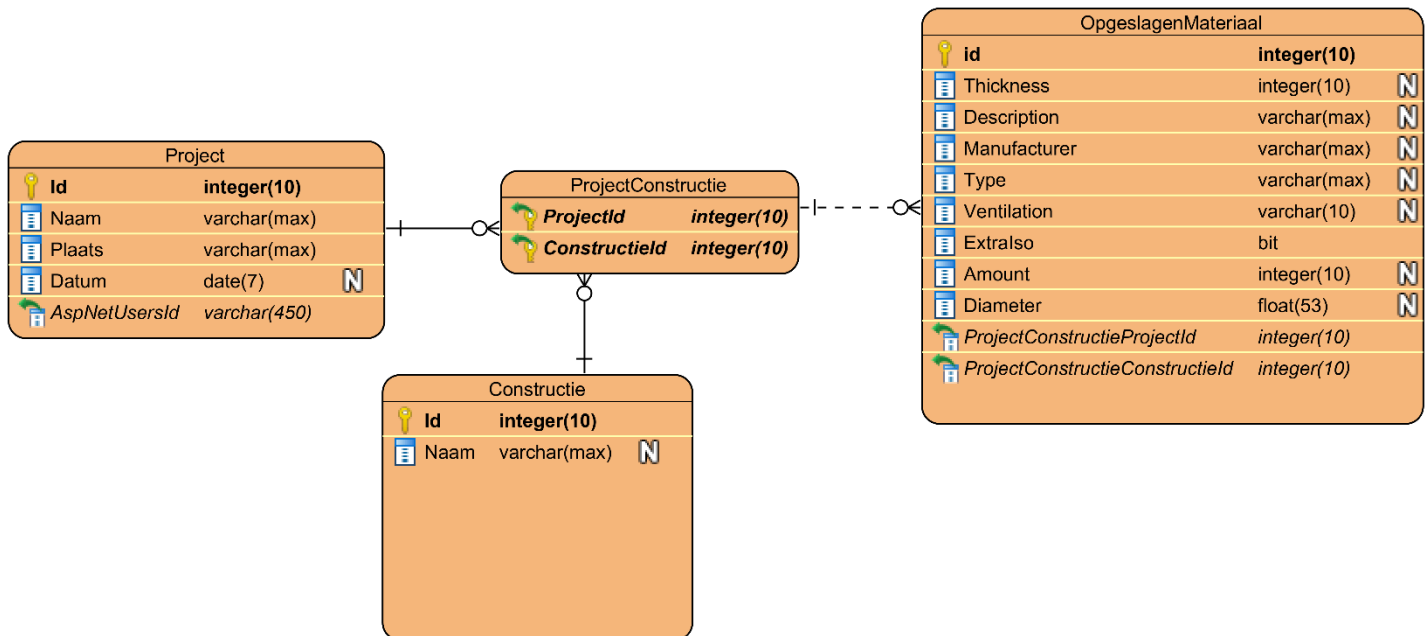


Figuur 21 Sprint 5 backlog

ONTWERP & REALISATIE

Tijdens het realiseren is de R_c -waarde berekening van de vorige sprint verbeterd. Ook is hierbij een deelontwerp gemaakt om de functionaliteit voor het opslaan van berekeningen te kunnen realiseren (zie Figuur 22 ERD 4). Voor het opslaan van berekeningen worden eigenlijk alleen de gekozen materialen en ingevulde diktes opgeslagen in de database. Dit wordt gedaan, zodat wanneer calculaties worden ingeladen, er alleen nog maar de javascript functie voor het berekenen van de R_c -waarde aangeroepen moet worden. Deze javascript functie zorgt er vervolgens voor, dat alle berekende waardes weer ingevuld en berekent zijn als hoe ze eerst waren, voordat de materialen werden opgeslagen. De geselecteerde materialen en ingevulde/geselecteerde diktes kunnen alleen opgeslagen worden, wanneer alle verplichte velden zijn ingevuld.

Verder zijn alle aanpassingen en verbeteringen die nog gedaan moesten worden aan de R_c -waarde berekening uitgevoerd. Zo wordt er nu rekening gehouden met alle regels die gelden voor het berekenen van het isolerend vermogen van een spouwmuur.



Figuur 22 ERD 4

REVIEW

De voortgang van de webapplicatie verliep nog steeds goed volgens dhr. Schmidt. Verder zag dhr. Schmidt direct dat er bij het opslaan van berekeningen nieuwe projecten aangemaakt moeten kunnen worden. Dit is handig voor wanneer gebruikers nog geen projecten hebben aangemaakt, maar wel hun berekeningen willen opslaan. Nu is het alleen nog mogelijk berekeningen in bestaande projecten op te slaan. Deze nieuwe user story mocht van dhr. Schmidt in de product backlog en hoefde niet de volgende sprint gelijk af. Verder moest er een aantal cosmetische aanpassingen gemaakt worden aan de R_c -waarde bereken pagina zoals het juist weergeven van een wiskundige waardes. Daarnaast moet de R_c -waarde rekenscherf op een pagina passen. Ook vond dhr. Schmidt dat er een about pagina moest komen met belangrijke informatie over de R_c -waarde berekening.

RETROSPECTIVE

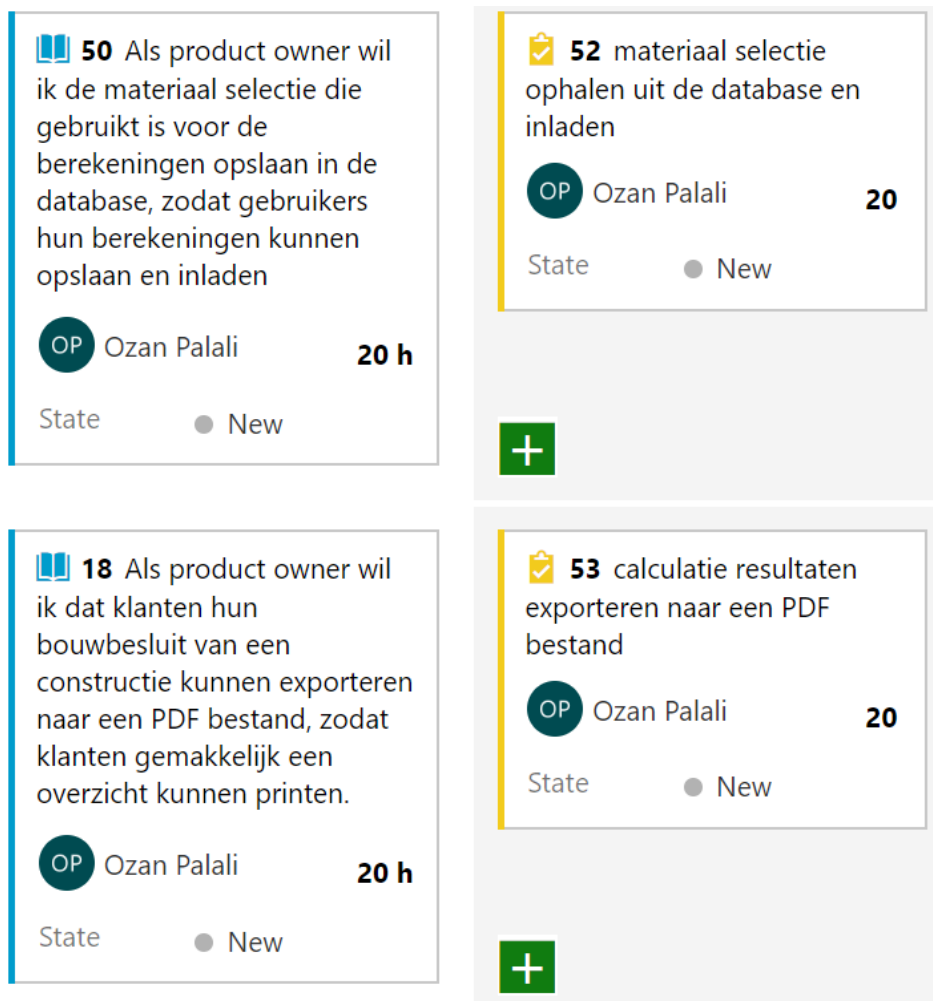
Wat er goed ging tijdens de sprint was het afronden van het afstudeerdossier voor de conceptbespreking. Verder ging het begrijpen van de berekening vanuit het validatie document beter dan verwacht. De formules konden bijna een op een overgenomen worden in de code. Daarnaast heeft het analyseren van de oude SBR-rekentool ook veel geholpen. Vanuit de oude tool kon gemakkelijk gezien worden welke functionaliteiten in de nieuwe DGMR-rekentool ontbraken.

Wat er minder goed ging was het niet kunnen afronden van alle taken door tijdsnood. Doordat er veel tijd was besteed aan het afstudeerdossier. Wat een verbeterpunt voor de volgende keer kan zijn, is eerst aan de sprint werkzaamheden werken en het afstudeerdossier zo nodig thuis verder afronden.

REQUIREMENTS & PLANNING

Tijdens de sprint planning is er een nieuwe userstory gemaakt. Dit is userstory nummer 56. De aanleiding hiervoor was, omdat tijdens de sprint review dhr. Schmidt er van overtuigd was dat het handig was om projecten aan te kunnen maken tijdens het opslaan van een calculatie. Volgens dhr. Schmidt is dit een gebruiksvriendelijkere methode dan waarbij eerst een nieuwe pagina geopend moet worden om vervolgens een nieuw project aan te maken, voordat gebruikers een materiaalselectie kunnen opslaan. Echter is de prioriteit van deze nieuwe userstory nog niet hoog genoeg om in de huidige sprint opgenomen te worden. Wel moest tijdens deze sprint het ophalen en inladen van materiaalselecties geïmplementeerd worden. Ook vond dhr. Schmidt dat het nu tijd was om de pdf-export functionaliteit te realiseren. De aanleiding hiervoor was, omdat de pdf-export nu de hoogste prioriteit had van de resterende userstories. Naast de userstories moeten ook een aantal cosmetische aanpassingen gedaan worden aan de rekenpagina. Deze aanpassingen zijn: het afgerond weergeven van sommige waardes, de juiste eenheden achter waardes weergeven en een disclaimer tekst.

Een belangrijk aandachtspunt is dat in dit document de woorden calculatie en materiaalselectie gebruikt worden, echter wilde dhr. Bosch dat dit in de webapplicatie constructie genoemd wordt. De reden hiervoor is, omdat wat er feitelijk opgehaald wordt wanneer een gebruiker opgeslagen materiaalselecties wilt inladen, een complete constructie is. Hierbij geldt ook dat een calculatie hetzelfde is als materiaalselecties. Daarbij is de reden waarom het materiaalselectie of calculatie genoemd wordt in dit document, omdat in het ontwerp een constructie een andere entiteit is dan een calculatie of materiaalselectie. Een materiaalselectie of calculatie is een verzameling van de entiteit "OpgeslagenMateriaal" (zie Figuur 24 ERD 5) en constructies in het ontwerp zijn de verschillende constructies waarvoor gebruikers een calculatie voor kunnen maken. Dus de reden waarom in dit document de terminologie materiaalselecties of calculatie wordt gebruikt is om een duidelijke onderscheid te hebben tussen materiaalselecties of calculatie en constructies



Figuur 23 Sprint 6 backlog

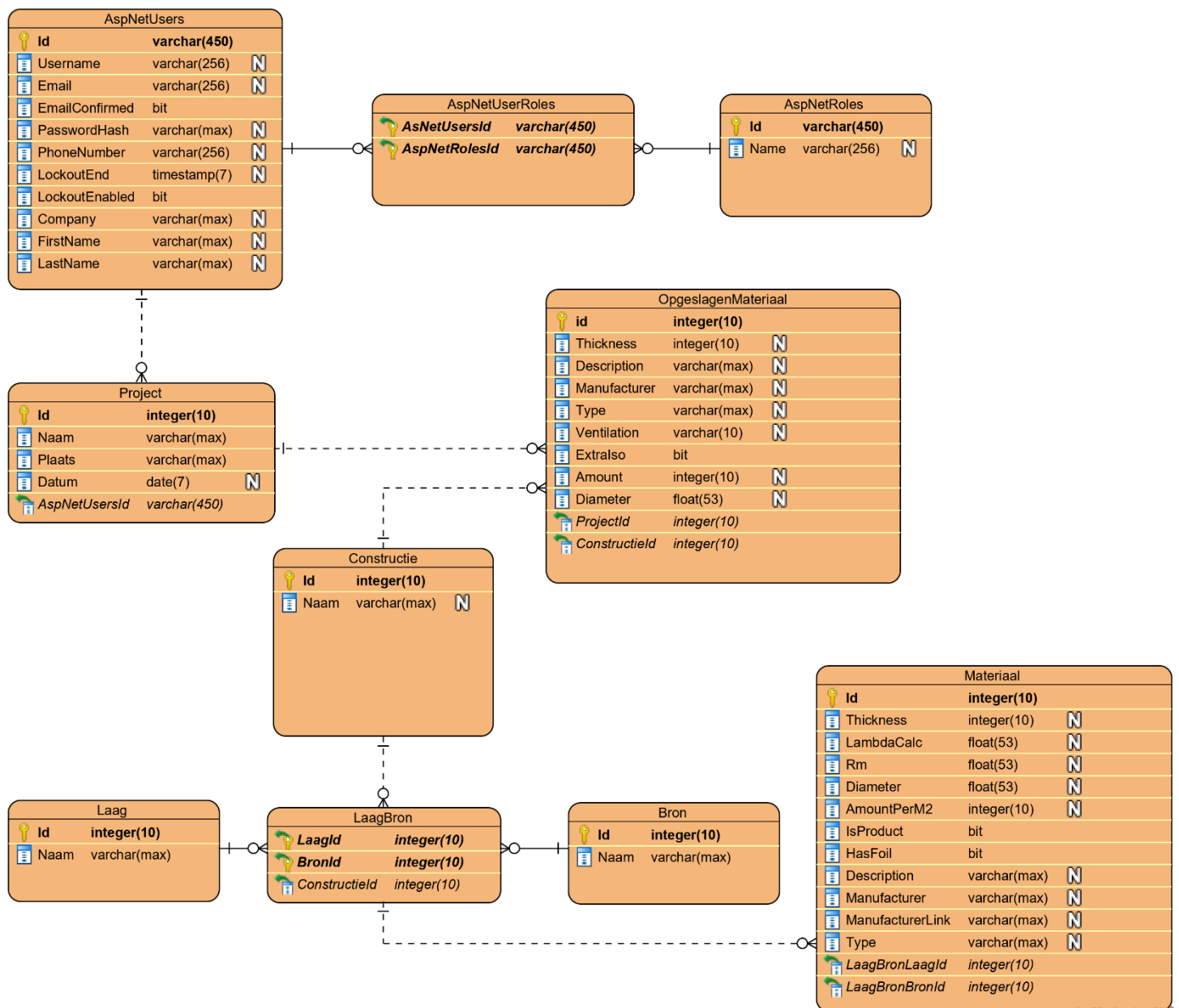
ONTWERP

In deze sprint zijn de laatste wijzigingen doorgevoerd die betrekking hebben tot de database, omdat na deze sprint geen userstories meer overblijven waarvoor de database aangepast of uitgebreid moet worden. Tijdens de ontwerpfasen in de vorige sprint waren de "ProjectConstructie" en "OpgeslagenMateriaal" tabellen gemodelleerd in Figuur 22 ERD, echter kon er van deze twee tabellen één tabel gemaakt worden. De aanleiding tot deze ontwerpkeuze was om de complexiteit van het ontwerp te verminderen. Door het reduceren van het aantal tabellen in het ontwerp, is het overzichtelijker en beter te begrijpen.

In de voorafgaande sprints zijn alle deelontwerpen gemodelleerd, echter worden tijdens deze sprint alle deelontwerpen samengevoegd om de complete databaseontwerp te vormen. Nu zal er het een en ander uitgelegd en samengevat worden aan de hand van het complete databaseontwerp. De entiteit `AspNetUser` heeft meerdere rollen en `AspNetRoles` heeft meerdere `AspNetUsers`. Dit is de reden waarom er een koppeltabel staat tussen deze entiteiten. Verder kunnen `AspNetUsers` projecten aanmaken waar ze gemaakte berekeningen in kunnen bewaren door de bijbehorende materialen selectie op te slaan. Echter kunnen `AspNetUsers` alleen projecten opslaan en laden als ze over de betalende-klant rol beschikken. Wanneer er een berekening opgeslagen moet worden, wordt de selectie van materialen met de ingevoerde waardes die gebruikt zijn bij het berekenen van de R_c -waarde opgeslagen. De `OpgeslagenMateriaal` tabel houdt dan bij welke opgeslagen materialen bij welke project en constructie horen. Daarnaast heeft elke constructie meerdere lagen bronnen en materialen. Dat geldt ook andersom. Lagen, bronnen en materialen kunnen in meerdere constructies voorkomen. Laag en bron hebben daarom een tussentabel genaamd `LaagBron`. Via

Laagbron kan bijgehouden worden welke lagen bij constructies horen en welke bronnen bij deze lagen horen en vervolgens welke materialen bij deze lagen en bronnen horen. Hieronder wordt nog eens uitgelegd waar elke tabel voor bedoeld is.

- AspNetUsers zijn de gebruikers accounts.
- AspNetRoles zijn de rollen.
- AspNetUserRoles zijn de rollen die bij de users horen.
- Projecten kunnen door gebruikers aangemaakt worden.
- Constructies zijn de verschillende constructies waar gebruikers de R_c -waarde van kunnen berekenen door middel van lagen bronnen en materialen te kiezen.
- Opgeslagen materiaal bevat de informatie van een materiaal die geselecteerd was voor een bepaalde constructie en project.
- Een constructie bevat lagen. Een laag kan bijvoorbeeld een isolatielaag zijn van een spouwmuur constructie.
- Een bron is het merk van het materiaal. Dus door wie een bepaald materiaal geleverd wordt.
- Per laag kan een bron gekozen worden deze bron heeft vervolgens weer materialen die erbij horen.



Figuur 24 ERD 5

REALISATIE

Voordat er aan de PDF-export gewerkt kon worden, moest er eerst een schets gemaakt worden van hoe de PDF eruit komt te zien (zie Figuur 26 PDF export schets). Toen de PDF-schets eenmaal goedgekeurd was, is er vervolgens onderzoek gedaan naar welke javascript library er gebruikt kon worden voor het genereren en exporteren van PDF-bestanden. Hierbij is er onder andere intern bij DGMR Software gevraagd naar advies bij programmeurs. Als advies werd een javascript library genaamd PDF.js en een javascript library genaamd jsPDF aangeraden door een van de programmeurs. Uit nadere zelfstandige onderzoek bleek dat PDF.js geen PDF-bestanden genereert maar enkel een library is die het mogelijk maakt om PDF-bestanden weer te geven op een webpagina. Dit was niet de gewenste functionaliteit die nodig was. Na deze library geëlimineerd te hebben is er gekeken naar de jsPDF javascript library. jsPDF is een gratis library die te downloaden is via GitHub. Op de github pagina van jsPDF zijn er voorbeelden, tutorial's en demo's. Nadat deze doorgenomen waren, is er besloten om gebruik te maken van de jsPDF library. Deze library maakte het namelijk mogelijk om de gewenste functionaliteit te realiseren. Door diverse voorbeelden en demo's te volgen via de github pagina van jsPDF, is uiteindelijk de functionaliteit voor het genereren en exporteren van PDF-bestanden gerealiseerd. Voor het

genereren van PDF-bestanden wordt er eerst door middel van de jsPDF library een pdf-document aangemaakt in de code. Vervolgens wordt er een layout afbeelding in de achtergrond geplaatst van het document. Daarna worden alle teksten en rekenwaardes op de juiste locatie geplaatst in het document. Wanneer een gebruiker nu op de knop export pdf klikt op de rekenpagina, wordt er automatisch een PDF gegenereerd met de materialen en diktes die hij of zij heeft ingevoerd en de waardes die zijn uitgerekend door de rekentool.

Nadat userstory 18 het genereren en exporteren van PDF-bestanden voldeed aan de Definition of Done en de regressie test was uitgevoerd, is er gewerkt aan userstory 50, het inladen van materiaalselecties uit de database. Hiervoor moest een select field komen op de rekenpagina waarin er opgeslagen calculaties geselecteerd konden worden. Wanneer een gebruiker een opgeslagen calculatie selecteert, wordt deze meteen ingeladen. Bij het inladen worden alle velden in de rekenpagina ingevuld met de waardes en materialen die opgeslagen zijn (zie Figuur 25 Ingeladen materialen en waardes).

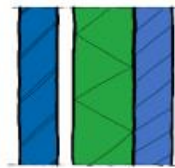
Spouwmuur

Laag	Bron	Materiaal	Dikte (mm)	λ_{Calc} (W/m ² ·K)	Rm (m ² ·K/W)
Binnenspouwblad	xella	Silka CS12	100	0.75	0.13333
Isolatie	sbr-referentiedetails	Speciaal cellulair glas	100	0.05	2
Extra Isolatie	isover	Mupan Plus	140	0.033	4.24242
Spouwankers	sbr-referentiedetails	Spouwanker RVS	Diameter: 5 Aantal: 5	17	
Spouw		Zwak geventileerd	30		0.16
Buitenspouwblad	sbr-referentiedetails	Buitenspouwblad (baksteen; gemetseld)	24	1.2	0.02
Totale constructie dikte:			394		

Rsi (m ² ·K/W):	0.13	Ut (m ² ·K/W):	0.149
Rse (m ² ·K/W):	0.04	ΔU (W/m ² ·K):	0.001
Rt (m ² ·K/W):	6.726	Uc (W/m ² ·K):	0.149
ΔUfa (W/m ² ·K):	0.001	Rc (m ² ·K/W):	6.522

Rc bb: 6.5 m²·K/W

Figuur 25 Ingeladen materialen en waardes



Laag	Bron	Materiaal	Dikte (mm)	Calc (W/m ² *K)	Rm (m ² *K/W)
Binnenspouwblad	xella	Silka CS12	100	0.75	0.1333
Isolatie	rockwool	Rockfit 433 BP	200	0.035	5.7143
Extra isolatie	rockwool	Rockfit 433 HP ? = 0,033	175	0.033	5.303
Spouwankers	sbr-referentiedetails	Spouwanker verzinkt staal (niet zonder meer toe te passen) Diameter: 6 Aantal: 6		50	
Spouw		Sterk geventileerd	28		0.33
Buitenspouwblad	nen1068	baksteen metselwerk 1200	35	0.7	0
Totale constructie dikte:			538		

Rsi	0.13 m ² *K/W	Ut	0.085 m ² *K/W
Rse	0.13 m ² *K/W	U	0 W/m ² *K
Rt	11.741 m ² *K/W	Uc	0.085 W/m ² *K
Ufa	0.003 W/m ² *K	Rc	11.481 m ² *K/W

RC = 11.5

m²*K/W

Figuur 26 PDF export schets

REVIEW

Nadat alle regressietesten waren uitgevoerd en de userstories voldeden aan de Definition of Done is de sprint review gehouden.

Omdat dit het laatste moment is voor de opdrachtgevers om verbeteringen en aanpassingen door te geven, is er een presentatie gehouden op een grote scherm binnen DGMR Software. Ook is deze presentatie gehouden om de leerdoel uit de beroepstaak leren leren te behalen. In deze presentatie zijn de volgende punten aan de orde gekomen.

Tijdens de review is ondanks dat het PDF-design goedgekeurd was, toch besloten om het PDF design te wijzigen. Dhr. Bosch vond toch een blauwe achtergrond in de tabellen van het PDF toepasselijker dan een

groene achtergrond. Verder waren de kwadraten van de diverse eenheden in het PDF-document niet correct genoteerd. Deze stonden genoteerd als een normale twee. Deze correcties moeten aan het begin van de volgende sprint toegepast worden.

Voor het ophalen en inladen van materiaalselecties, deed de functionaliteit die is gerealiseerd exact wat die moest doen. Namelijk het ophalen van een materiaalselectie die eerder was opgeslagen en vervolgens het inladen ervan op de rekenpagina. Het enige wat niet goed was, was de manier waarop deze functionaliteit werd gestart. Om een materiaalselectie in te laden moet een gebruiker eerst op een checkbox drukken om de knop voor het inladen en de selectfield voor het kiezen van een calculatie zichtbaar te maken. Dan pas kan de gebruiker een calculatie inladen. Wanneer de knop en de selectfield voor het inladen zichtbaar is, wordt de knop en de selectfield voor het opslaan onzichtbaar. Dit vond dhr. Schmidt te ingewikkeld voor gebruikers. Hij vindt het beter als alle knoppen direct zichtbaar zijn op de webpagina. Daarnaast wilt dhr. Schmidt dat de selectfield voor het kiezen van een calculatie in een popup komt, die pas tevoorschijn komt wanneer er op het "Constructie laden" knop geklikt wordt. In deze popup kunnen gebruikers vervolgens een opgeslagen calculatie selecteren en op ophalen klikken in de popup. Voor het opslaan van een calculatie in een project wilt dhr. Schmidt net als bij het inladen, dat er eerst op een "Opslaan in project" knop geklikt wordt voordat de popup met de selectfield tevoorschijn komt. In deze popup kunnen gebruikers vervolgens hun project kiezen waarin ze de calculatie in willen opslaan en de calculatie een naam geven.

Na afloop van de presentatie was er feedback gevraagd over de presentatie vaardigheden. Er werd verteld door dhr. Schmidt en dhr. Bosch dat er te snel werd gepraat tijdens het presenteren, waardoor het soms niet duidelijk was wat er precies werd bedoeld. Verder verliep de presentatie soepel volgens de opdrachtgevers. Daarnaast werden eventuele onduidelijkheden verhelderd.

RETROSPECTIVE

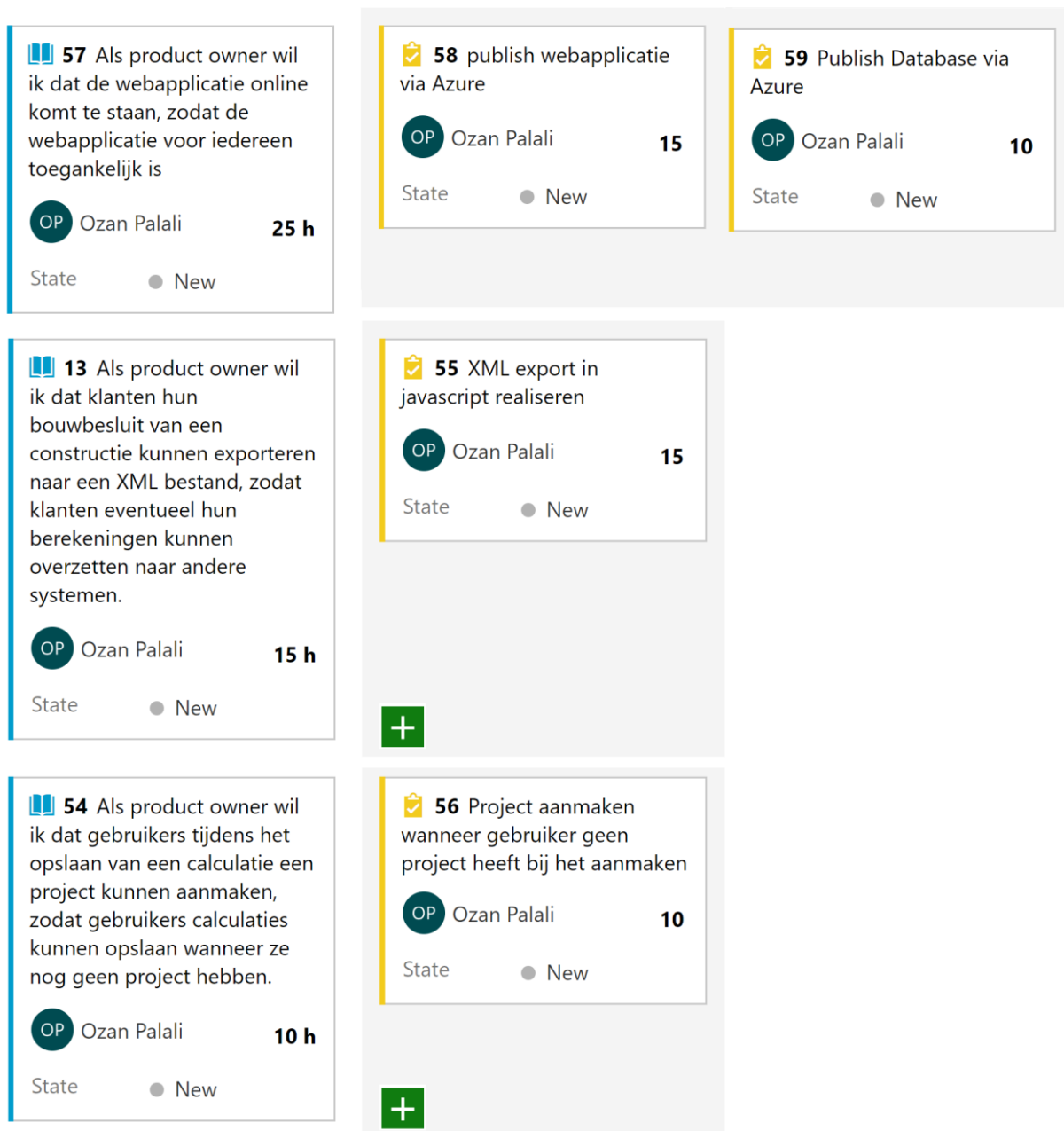
Wat er goed ging tijdens deze sprint was de planning en uitvoering van de taken. De uren voor de taken waren goed ingeschat. Hierdoor was er genoeg tijd over om aan dit afstudeerdossier te werken.

Tijdens deze sprint is er verder niet tegen andere problemen aangelopen dan genoemd in de sprint review.

5.8 SPRINT 7 XML EXPORT EN PUBLISH

REQUIREMENTS & PLANNING

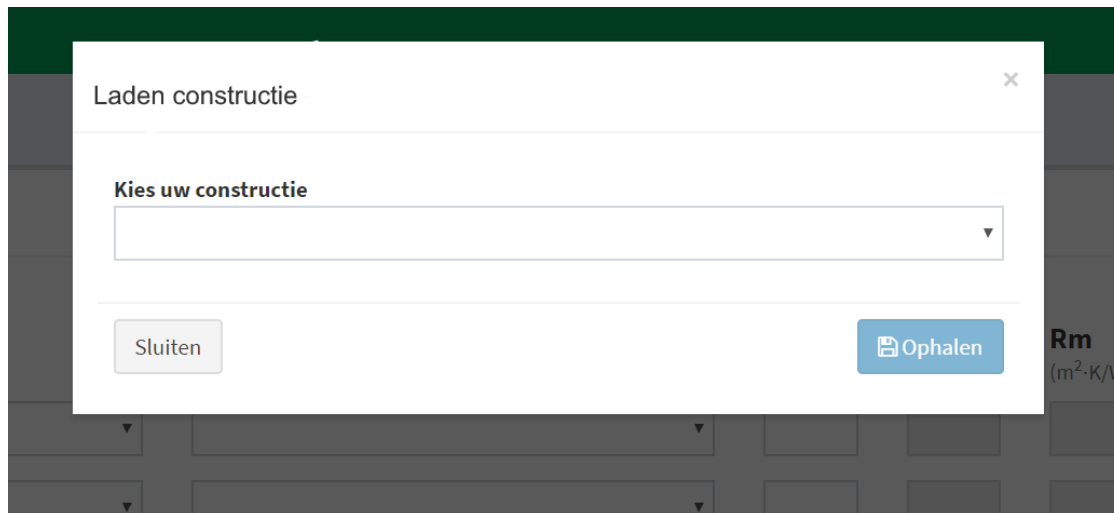
Naast de verbeterpunten van de vorige sprint review, zijn ook de laatste userstories opgenomen en taken ervoor aangemaakt. Aangezien dit de laatste sprint is, zijn er verder geen userstories meer overgelaten in de product backlog. De verbeterpunten zijn de popup velden en de diverse PDF-verbeteringen en aanpassingen zoals vermeld in de review van sprint 6. Bij deze sprint is er ook een nieuwe userstory aangemaakt. De webapplicatie moet namelijk online gezet worden samen met de database. Deze nieuwe userstory is userstory nummer 57. Deze userstory is erbij gekomen, omdat dhr. Schmidt en dhr. Bosch de applicatie uitgebreid willen testen.



Figuur 27 sprint 7 backlog

ONTWERP & REALISATIE

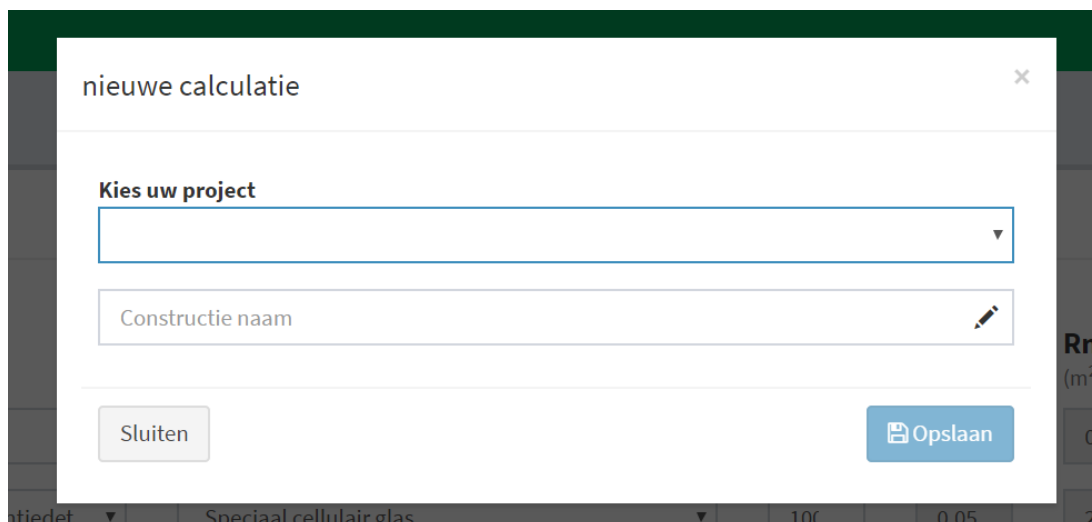
Voor deze sprint was het niet nodig om het webapplicatie ontwerp uit te breiden of aan te passen. Dit geldt ook voor het ERD-ontwerp. Als eerste is er gewerkt aan de verbeterpunten die benoemd waren in de vorige sprint review. Het wijzigen van de achtergrond kleur van de PDF-layout was simpelweg met de desktopapplicatie Paint voor elkaar gekregen. Verder waren de eenheden van de waardes gecorrigeerd. Voor het inladen en opslaan van de calculaties, zijn de wijzigingen ervoor toegepast zoals Dhr. Schmidt dat had beschreven. Nu zijn de twee knoppen voor het ophalen en opslaan van calculaties direct zichtbaar rechtsonder in de webpagina boven de twee export knoppen (zie Figuur 31 Rekenpagina). Wanneer er nu op constructie laden wordt geklikt, krijgt de gebruiker een popup te zien boven in de webpagina. (zie Figuur 28 Laden constructie).



Figuur 28 Laden constructie

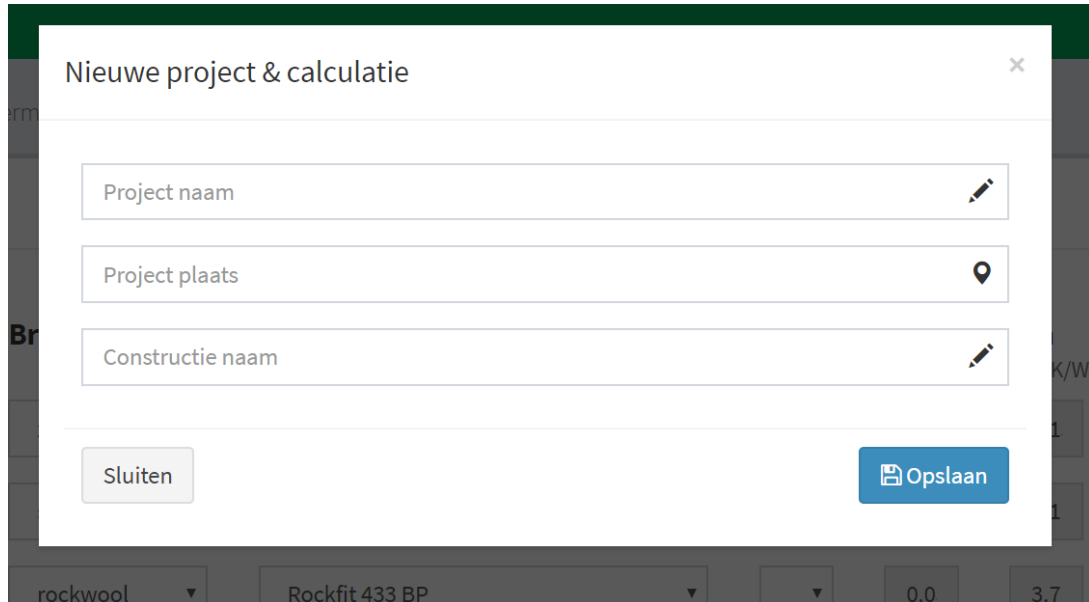
In deze popup kunnen gebruikers hun eerder opgeslagen calculaties inladen op de rekenpagina door een calculatie te selecteren en vervolgens op ophalen te drukken.

Hetzelfde geldt ook bij het opslaan van een calculatie. De gebruiker drukt op de knop “Opslaan in project” en vervolgens komt er een popup tevoorschijn waarin de gebruiker het project kan kiezen waar de calculatie in opgeslagen moet worden. Daarbij kan de gebruiker de calculatie ook een naam geven (zie Figuur 29 Calculatie opslaan in project).



Figuur 29 Calculatie opslaan in project

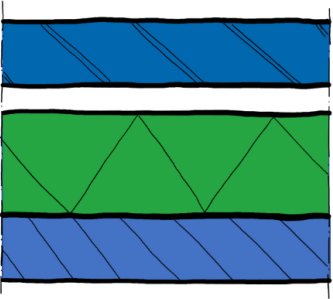
Naast de verbeterpunten van de vorige sprint is ook gelijk userstory nummer 54 geïmplementeerd, het kunnen opslaan van een calculatie wanneer de gebruiker geen project heeft. Wanneer de gebruiker nog geen project heeft waarin die zijn calculatie in kan opslaan, detecteert de webapplicatie dat. Dit wordt gedaan door in de database te checken of de id van huidige gebruiker voorkomt in één van de projecten. Als dat niet het geval is krijgt de gebruiker nu een iets andere popup te zien dan in Figuur 29 Calculatie opslaan in project. In deze nieuwe popup kan de gebruiker gelijk een nieuw project aanmaken en tegelijkertijd ook de calculatie erin opslaan (zie Figuur 30 Nieuwe project & calculatie).



Figuur 30 Nieuwe project & calculatie

Bij het online zetten van de webapplicatie en database werd er geholpen door dhr. Biemond. Het online zetten van de database ging vrij vlot. Daarbij waren geen onverwachte obstakels. In tegenstelling tot het online zetten van de database, was het online krijgen van de webapplicatie wel lastig. Dit kwam doordat de webapplicatie was gebouwd in twee verschillende projecten binnen Visual Studio. De reden hiervoor was om de front-end en back-end gescheiden te houden door voor de back-end en de front-end een aparte project aan te maken. Dit was echter onnodig, omdat deze scheiding ook plaats kon vinden binnen één project. Doordat voor het online zetten van de webapplicatie één project vereist was binnen Visual Studio, is er uiteindelijk voor gekozen om deze twee projecten te combineren en één project van te maken. Hiermee was het probleem opgelost en kon de webapplicatie online gezet worden. De URL voor de webapplicatie is: <https://dgmrc-tool.azurewebsites.net/>.

Laag	Bron	Materiaal	Dikte (mm)	λ_{calc} (W/m ² .K)	R _m (m ² .K/W)
Binnenspouwblad	xella	YTONG Blokken G4/600	150	0.16	0.9375
Isolatie	recticel	Eurowall 21	90	0.021	4.28571.
Extra Isolatie	isoover	Mupan Ultra XS	138	0.032	4.3125
Spouwankers	sbr-referentiedetails	Spouwanker RVS	Diameter 5 Aantal 5		
Spouw		Sterk geventileerd	24		0.33
Buitenspouwblad	sbr-referentiedetails	Buitenspouwblad (baksteen; gemetseld)	24	1.2	0
Totale constructie dikte:			426		



Spouwmuur

R_{si} (m².K/W) :

0.13

U_t (m².K/W) :

0.099

R_{se} (m².K/W) :

0.13

ΔU (W/m².K) :

0.002

R_{te} (m².K/W) :

10.126

U_c (W/m².K) :

0.1

ΔU_{fa} (W/m².K) :

0.002

R_c (m².K/W) :

9.711

R_c bb: 9.7 m².K/W

Om dit te realiseren is er eerst op google gezocht hoe gebruikers een XML-bestand kunnen downloaden via de webapplicatie. Bovendien moest er ook achterhaald worden hoe de structuur van de XML eruit moet zien. Hiervoor moest er informatie verkregen worden van een programmeur die werkt aan de BINK-applicatie. Dhr. Álvarez is een van de programmeurs die verantwoordelijk is voor BINK. Dhr. Álvarez heeft vervolgens een voorbeeld XML-bestand overgedragen door middel van een export functionaliteit vanuit de BINK-applicatie. Dit voorbeeld XML-bestand dient als basisstructuur voor de te realiseren XML-export functionaliteit.

```
var xmltext = '<?xml version="1.0" encoding="ISO-8859-1"?>
<BinkNode ModelVersion = "9.0.74.0" FileGuid = "{71BA25F8-3CE1-4DB2-920F-E58EFC871792}">
  <Construction>
    <Construction Accumulation="1" AdjacentToGround="false" Area="0" Area_Formula="L*W" Articulation="0">
  </Construction>
  <Material>
    <Material Description="" + $scope.selectedBibMaterial.description + ` ` Id="1" Indication="Bib" />
    <Material Description="" + $scope.selectedIsoMaterial.description + ` ` Id="2" Indication="Iso" />
    <Material Description="" + $scope.selectedExtraIsoMaterial.description + ` ` Id="3" Indication="ExtraIso" />
    <Material Description="" + $scope.selectedAnkMaterial.description + ` ` Id="4" Indication="Ank" />
    <Material Description="" + $scope.selectedSpouw.name + ` ` Id="5" Indication="Buitenspouw" />
    <Material Description="" + $scope.selectedBubMaterial.description + ` ` Id="6" Indication="Bub" />
  </Material>
  <Materiallayer>
    <Materiallayer Description="" + $scope.selectedBibMaterial.description + ` ` Id="43" IdCorrelation="Bib" />
    <Materiallayer Description="" + $scope.selectedIsoMaterial.description + ` ` Id="44" IdCorrelation="Iso" />
    <Materiallayer Description="" + $scope.selectedExtraIsoMaterial.description + ` ` Id="45" IdCorrelation="ExtraIso" />
    <Materiallayer Description="" + $scope.selectedAnkMaterial.description + ` ` Id="46" IdCorrelation="Ank" />
    <Materiallayer CavityHasFoil="" + $scope.selectedExtraIsoMaterial.hasFoil + ` ` Description="" Id="47" IdCorrelation="ExtraIso" />
    <Materiallayer Description="" + $scope.selectedBubMaterial.description + ` ` Id="48" IdCorrelation="Bub" />
  </Materiallayer>
</BinkNode>';
```

Vervolgens wordt er een XML document gevormd en een naam aan toegekend. Tot slot wordt dit bestand gedownload wanneer de gebruiker op de export XML-knop klikt op de webpagina. Daarbij kan een XML-document alleen maar gedownload worden, wanneer alle waardes op de rekenpagina correct zijn ingevuld. Dit geldt ook voor de PDF-export.

REVIEW

Tijdens de laatste sprint review van deze afstudeeropdracht is er door dhr. Bosch en dhr. Schmidt gevalideerd of de aanpassingen en verbeteringen van de vorige sprint review zijn afgerond. Ook hebben zij gekeken naar de gerealiseerde/geïmplementeerde userstories en taken van deze sprint. Deze voldeden aan alle eisen die zij hadden gesteld tijdens de vorige sprint review.

RETROSPECTIVE

Wat er goed ging was het feit dat er hulp werd verleend bij het publishen van de webapplicatie en de database door dhr. Biemond. Deze hulp is erg op prijs gesteld, hierdoor kon er namelijk meer aandacht besteed worden aan de andere userstories en taken en kon de afstudeeropdracht succesvol afgerond worden.

Wat er minder goed ging tijdens deze sprint, was dat er te weinig tijd was besteed aan dit afstudeerverslag. Dit kwam omdat de userstories en taken van deze sprint veel tijd vergden. Omdat het de laatste sprint was moest het zeker zijn dat de afstudeeropdracht voldeed aan de eisen die dhr. Bosch en dhr. Schmidt hadden gesteld.

Om dit probleem voortaan te voorkomen, zal er in de toekomst beter rekening gehouden worden met het aantal userstories en de hoeveelheid sprints die nog over zijn. Hierdoor kan er dan beter voorruit gekeken worden, waardoor er een betere planning gemaakt kan worden voor de sprints.

6 EVALUATIE

Toen ik begon met mijn afstudeeropdracht was het nog niet helemaal duidelijk welke functionaliteiten ik in welke sprints zou realiseren. Tijdens het maken van het plan van aanpak heb ik een globale planning gemaakt in de vorm van een strokenschema (zie Figuur 4 Strokenplanning). Ik heb geprobeerd om mij hier zoveel mogelijk aan te houden. Dat was uiteindelijk ook grotendeels gelukt. De originele planning. Kon grotendeels aangehouden worden. In de planning was aangegeven dat er in sprints 2 tot en met 5 gewerkt zou worden aan de database. Dit was echter van sprints 1 tot en met 7. Omdat er met scrum wordt gewerkt, was het mogelijk om een beetje af te wijken van de originele planning. De reden waarom het realiseren van de database afweek van de globale planning was, omdat de database elke sprint aangepast, verbeterd of uitgebreid moest worden. De database werd tegelijkertijd met de webapplicatie gerealiseerd. Vanaf sprint 1 tot en met sprint 6 ging het om de lokale Microsoft SQL Server database en in sprint 7 werd deze database gepubliceerd op Azure. Verder verliep het proces van de afstudeeropdracht volgens planning.

Ik ben tevreden over de volgorde van de uitvoering van de werkzaamheden. De afstudeeropdracht verliep grotendeels volgens plan en alle deelontwerpen vormde in de laatste sprint een compleet geheel en daar ben ik erg tevreden over. Waar ik ontevreden over ben is mijn gebrek aan gebruik van creativiteit tijdens het uitvoeren van de werkzaamheden.

Als ik opnieuw zou moeten beginnen aan deze afstudeeropdracht, zou ik meer mijn creativiteit gebruiken en hiermee meer betrokken zijn bij de totstandkoming van de beslissingen, als het gaat om bijvoorbeeld hoe de webapplicatie eruit komt te zien. Daarbij had ik ook graag met eigen ideeën willen komen in plaats van enkel de ideeën van dhr. Bosch en dhr. Schmidt te implementeren. Zo zou ik meer het gevoel hebben dat ik een actieve bijdrage heb geleverd aan de vorm van de webapplicatie op het gebied van bijvoorbeeld gebruiksvriendelijkheid en/of toegankelijkheid. Dit is ook iets wat ik te horen kreeg tijdens de eindevaluatiegesprek met dhr. Bosch en dhr. Schmidt.

Doordat er nog een aantal onduidelijkheden waren aan het begin van de afstudeeropdracht over bijvoorbeeld hoe de database conversie uitgevoerd moest worden of hoe de R_c -waarde precies berekent moest worden, heeft het feit dat er met scrum is gewerkt erg veel geholpen. Zo heb ik gedurende de afstudeeropdracht steeds nieuwe kennis opgedaan die betrekking hebben tot deze onduidelijkheden, waardoor ik de requirements steeds beter kon specificeren en de sprints beter kon indelen.

Tijdens mijn eindevaluatie gesprek met dhr. Bosch en dhr. Schmidt kreeg ik te horen dat ik erg snel kan praten en dat ik hierdoor vergeet rekening te houden met het feit dat mensen mij dan niet meer goed kunnen volgen. Dit is iets wat ik vaker hoor in het dagelijks leven. Ik heb vaak het gevoel dat wanneer ik iets aan het uitleggen ben, ik niet duidelijk genoeg ben. Hierdoor vertel ik veel extra informatie, waardoor ik sneller ga praten wat weer als gevolg er juist voor zorgt dat ik onduidelijk wordt. Dit probleem deed zich voor bij een aantal sprint reviews en weekly meetings. Dit heb ik nu ingezien. Ik zal er in de toekomst aan werken om alleen de hoofdlijnen van een onderwerp te vertellen, wanneer ik iets aan het uitleggen ben.

6.1 PRODUCTEN

PLAN VAN AANPAK

Het eerste waaraan gewerkt was toen er begonnen was met het afstuderen bij DGMR Software was de plan van aanpak. Bij het maken van de plan van aanpak is er uiteraard rekening gehouden met de beschikbare tijd voor de afstudeeropdracht. De input voor het tot stand komen van de plan van aanpak, waren diverse interviews met de opdrachtgevers en natuurlijk het afstudeerplan. Het plan van aanpak heeft dhr. Bosch en dhr. Schmidt inzicht gegeven in mijn aanpak en planning en in de manier waarop ik de tussenproducten zal opleveren. Hierdoor waren zij voorbereid op wat ze van mij konden verwachten. De plan van aanpak gaf een overzicht van de op te leveren producten en diende hierbij ook als een overeenstemming tussen mij en de opdrachtgevers over wat er allemaal opgeleverd moest worden.

REQUIREMENTS

Voor het verzamelen van de requirements zijn er gesprekken en interviews gehouden met de opdrachtgevers. Ook is het plan van aanpak en afstudeerplan gebruikt bij het opstellen van de requirements. De formulering van de requirements heb ik in de loop van de tijd eventueel meer in detail gespecificeerd. Dit kwam doordat ik gedurende de afstudeeropdracht steeds nieuwe kennis opdeed die betrekking had tot onduidelijkheden van de requirements.

ONTWERPRAPPORT

Het ontwerprapport bevat gedetailleerde diagrammen die de functionaliteiten en relaties van de Rekentool weergeeft. De database diagrammen moesten vaak verbeterd worden, omdat ik vaak achteraf erachter kwam dat ik het ontwerp beter had kunnen aanpakken. Het ontwerp leek goed te zijn voordat ik startte met het realiseren echter eenmaal aan het coderen zag ik dat het ontwerp te complex was en dat het eenvoudiger kon. Bovendien kwam ik er soms ook achter dat het ontwerp aangepast moest worden, omdat de te realiseren functionaliteit niet mogelijk was met het gemodelleerde ontwerp tijdens het programmeren. Uiteindelijk kwam tijdens de laatste sprint alles mooi bij elkaar en werd het totale plaatje compleet met de laatste ERD van de database (zie Figuur 24 ERD 5).

DATABASECONVERSIE

De databaseconversie was een uitdagende proces. Er moest rekening gehouden worden met veel aspecten om dit eenmaal goed te krijgen. De volledige databaseconversie scripts kunnen gevonden worden in Bijlage F – Conversie Scripts. Zoals vermeld in de Realisatie van sprint 3 moest er eerst kennis opgedaan worden over het CSV-bestand met alle materiaalgegevens dat geïmporteerd moes worden. Dit bestand bevatte namelijk termen en afkortingen waar ik nog niet bekend mee was. Om ervoor te zorgen dat het CSV bestand goed werd geconverteerd naar de huidige database, moest ervoor gezorgd worden dat deze onduidelijkheden verduidelijkt werden. Dankzij de goede communicatie met dhr. Bosch en dhr. Schmidt tijdens de sprint (Sprint 3 Database conversie), waren deze onduidelijkheden snel verholpen.

DATABASE

Het realiseren van de database was eigenlijk het enige wat niet helemaal verliep volgens de globale planning van Figuur 4 Strokenplanning. Dit kwam omdat vanaf sprint 1 de database eigenlijk al gedeeltelijk moest functioneren. Deze moest functioneren voor het opslaan van de gebruikersgegevens wanneer een gebruiker een nieuwe account registreerde. De reden waarom dit niet in de globale planning stond was, omdat er geen rekening was gehouden met de werkzaamheden die uitgevoerd zouden worden in sprint 1 (Sprint 1 Initiële opzet). In de globale planning staat dat er van sprint 2 tot en met sprint 5 aan de database gewerkt zal worden,

echter moest er elke sprint aan de database gewerkt worden. De database werd namelijk elke sprint aangepast, verbeterd of uitgebreid en in sprint 7 werd deze online gezet.

DGMR RCTOOL

Ik ben tevreden over de door mij gerealiseerde webapplicatie. Ook zijn mijn opdrachtgevers tevreden over het eindresultaat. Zo kreeg ik complimenten van dhr. Schmidt over de stabiliteit van de webapplicatie. Dhr. Schmidt gaf aan dat tijdens zijn carrière, hij weinig applicaties niet heeft kunnen laten crashen en deze webapplicatie was daar één van. Verder vond hij het een prima webapplicatie, zeker gezien de beschikbare tijd, leercurve die vereist is voor .NET applicaties en de input die ik heb gekregen.

DEFINITION OF READY

De defintion of ready speelde een belangrijke rol tijdens het ontwikkelproces. Voordat een sprint gestart kon worden, werd eerst de lijst uit de defintion of ready doorlopen om er zeker van te zijn of de sprint wel gestart kon worden. Dit zorgde ervoor dat de kwaliteit van de userstories hoog was, wat als gevolg voor minder onduidelijkheden zorgde.

DEFINITION OF DONE

De defintion of done is een checklist die nagelopen moest worden wanneer een userstory af was. Dit zorgde ervoor dat mijn gerealiseerde functionaliteiten aan de kwaliteitseisen voldeden. Zo werd er onder andere gegarandeerd dat elke userstory getest werd en indien nodig een code refactor plaatsvond. Het nalopen van deze checklist vond ik erg handig, omdat dit mij het vertrouwen gaf dat ik daadwerkelijk een userstory had afgerond.

6.2 BEROEPSTAKEN

A1 ANALYSEREN PROBLEEMDOMEIN

Deze beroepstaak is voldaan door het opbouwen van de plan van aanpak (zie Bijlage B - Plan van aanpak). Tijdens het opstellen van dit afstudeerdossier is de probleemstelling verder uitgebreid (zie Probleemstelling). Door de stakeholders te interviewen, is duidelijk geworden wat de aanleiding was voor deze afstudeeropdracht. Verder is er uitgelegd hoe de huidige situatie eruit zag.

A3 VERGAREN EN ANALYSEREN VAN REQUIREMENTS

Aan deze beroepstaak is voldaan door diverse interviews en gesprekken te houden met de stakeholders. Tijdens de introductie en inwerken is in de kop Backlog de aanpak voor het vergaren en analyseren van de requirements beschreven. Verder heb ik mijzelf georiënteerd op het gebied van R_c-waarde berekeningen.

C6 ONTWERPEN SOFTWARE

Door het opstellen van het ontwerprapport in Bijlage C – Ontwerprapport, is deze beroepstaak voldaan. Per sprint zijn er deelontwerpen gemodelleerd van de database, wat uiteindelijk één complete databaseontwerp is. Daarnaast zijn er ook sequentie diagrammen en klassendiagrammen van de webapplicatiestructuur gemodelleerd. Ook heb ik tijdens deze afstudeeropdracht zoals beschreven in de kop Infrastructuur van Introductie & inwerken, een design principle en een design pattern geïmplementeerd in de webapplicatie ontwerpdiagrammen (zie Figuur 7 webapplicatie ontwerp 1 en Figuur 10 webapplicatie ontwerp sprint 2). Deze webapplicatieontwerpen zijn gemodelleerd in UML. Ook zijn er twee sequentie diagrammen gemaakt (zie Bijlage C – Ontwerprapport).

D14 REALISEREN VAN SOFTWARE

Door het ontwikkelen en opleveren van de DGMR Rctool webapplicatie is deze beroepstaak voldaan. Vanaf Sprint 1 Initiële opzet tot en met Sprint 7 XML export en publish, is er gewerkt aan deze webapplicatie. Er is zowel een front-end als een back-end gerealiseerd. Voor versiebeheer is GIT gebruikt in combinatie met Azure DevOps (zie Scrum in Introductie & inwerken). In de realisatie kop van elke sprint wordt in detail beschreven hoe de functionaliteiten tot stand zijn gekomen. Door werkervaring, kennis die opgedaan is op school en mijn ervaring tijdens mijn derdejaars stage, beschikte ik over de basiskennis die nodig was voor het realiseren van deze webapplicatie.

D16 REALISEREN & GEBRUIKEN DATABASE

De beroepstaak realiseren & gebruiken database is voldaan door de database te ontwerpen en te realiseren. In elke sprint die betrekking heeft tot een aanpassing, verbetering of uitbreiding van de database, wordt tijdens het ontwerpen de verschillende keuzes die zijn gemaakt voor het ontwerp van de database in detail uitgelegd. In de kop Ontwerpen van Introductie & inwerken is duidelijk gemaakt hoe de database gemodelleerd gaat worden. Vanaf Sprint 1 Initiële opzet tot en met Sprint 6 Rc berekening ophalen en Exporteren PDF, heb ik gewerkt aan de lokale Microsoft SQL Server database. In sprint 7 heb ik deze met behulp van dhr. Biemond gepubliceerd op Azure.

GC KRITISCH, ONDERZOEKEND EN METHODISCH WERKEN

Met het toepassen van SCRUM is deze beroepstaak behaald (zie Scrum in Introductie & inwerken). Tijdens het uitvoeren van de werkzaamheden, ben ik methodisch te werk gegaan. In de kop Introductie & inwerken heb ik uitgelegd over de verschillende methodes die ik gebruikt hebt tijdens het werken aan de afstudeeropdracht. Daarnaast heb ik kritisch gekeken naar welke design principle en design pattern ik het beste kon gebruiken voor het modelleren van de webapplicatieontwerpen (zie Infrastructuur in Sprint 1 Initiële opzet). Verder heb ik onderzoek gedaan naar hoe een Rc-waarde precies berekend wordt (zie Realisatie in Sprint 4 Rc-waarde berekening).

GF LEREN LEREN

De beroepstaak leren leren is behaald door het op te stellen van een leerdoel. Deze leerdoel is SMART geformuleerd. Verder is er van deze leerdoel een tabel gemaakt om zo een goed overzicht te hebben over de verschillende SMART aspecten. Aan deze leerdoel is gewerkt tijdens een sprint review. Naast het opstellen van een leerdoel, heb ik ook actief gezocht naar oplossingen voor problemen die ik bijvoorbeeld had tijdens het realiseren van de webapplicatie. Hiervoor ben ik naar diverse forums geweest zoals StackOverflow en heb ik tutorials gevolgd, om zo problemen van de code op te lossen.

Specifiek	Verbeteren van de presentatie en demonstratie van de gerealiseerde functionaliteiten van een increment.
Meetbaar	Dit leerdoel is te meten door na het geven van een demo tijdens een sprint review, naar feedback te vragen over de gepresenteerde en gedemonstreerde increment.
Acceptabel	Het is belangrijk om je werkzaamheden of product goed te kunnen presenteren, zodat alles wat geïmplementeerd of gerealiseerd is ter sprake komt. Dit is belangrijk, omdat dan duidelijk wordt wat er precies is gedaan.
Realistisch	Een demo voorbereiden voor een sprint review.
Tijdgebonden	Binnen de periode van het afstuderen.

BEGRIPPENLIJST

Begrip	Definitie
Back-end	Dit is de achtergrond van de webpagina waar de gebruiker niet direct toegang tot heeft. Hier wordt bijvoorbeeld data opgeslagen in de database of opgehaald.
Checkbox	Een aankruisvakje op een webpagina.
Convectie	De beweging van een stof (meestal lucht of een vloeistof) die bij verwarming uitzet en opstijgt, en vervolgens afkoelt en weer daalt. Hierdoor ontstaat een circulatie die tot warmtetransport aanleiding geeft.
Daily standup	Een dagelijkse bijeenkomst van maximaal 15 minuten waarbij werkzaamheden op elkaar worden afgestemd en mogelijke obstakels worden vermeld.
Epic	Een grote userstory. Een epic bestaat meestal uit meerdere userstories
Front-end	Dit is wat de gebruiker ziet op een webpagina. Dus de html, CSS en javascript.
Materiaalselectie	Een selectie van materialen. Wordt ook calculatie genoemd binnen deze afstudeeropdracht.
Product backlog	Een verzameling van userstories.
R_c -waarde	Het isolerend vermogen van een constructie.
Rekenscherm	Een webpagina waar de R _c -waarde van een constructie wordt berekent.
Requirement	Een requirement is een gedocumenteerde fysieke of functionele behoefte waar een ontwerp, product of proces aan moet voldoen.
SCRUM	Een software ontwikkelmethode waarin met sprints gewerkt wordt, om op een effectieve en flexibele manier software te ontwikkelen.
Selectfield	Een veld op een webpagina waarin je een bepaalde waarde kan selecteren.
Sprint	Een tijdsperiode die steeds even lang duurt waarin een gedeelte van het eindproduct wordt gerealiseerd.
Sprint backlog	Een verzameling van userstories voor een sprint.
Sprint planning	Een planning die gemaakt wordt voor een sprint.
Sprint retrospective	Tijdens de retrospective wordt er besproken wat er goed ging tijdens afgelopen sprint, wat er beter kon en hoe het de volgende keer beter kan.
Sprint review	Een review die gehouden wordt aan het einde van de sprint waarbij de belanghebbende feedback kunnen geven.
Userstory	Een korte beschrijving van wat een belanghebbende wil.

7 BIBLIOGRAFIE

- Belzen, T. v. (2017, 05 03). *Doek valt voor SBRCURnet*. Opgehaald van Cobouw:
https://www.cobouw.nl/bouwbreed/nieuws/2017/05/doek-valt-voor-sbrcurnet-101247899?_ga=2.178899874.728712345.1578446731-311272485.1578446731
- DGMR Software B.V. (2019). *Algemene informatie over DGMR Software B.V.* (DGMR) Opgeroepen op September 4, 2019, van DGMR Software: <https://dgmsoftware.nl/>
- FullStack Mark. (2018, 05 23). *JWT Authentication with ASP.NET Core*. Opgehaald van fullstackmark:
<https://fullstackmark.com/post/13/jwt-authentication-with-aspnet-core-2-web-api-angular-5-net-core-identity-and-facebook-login>
- Gupta, S. (2012, juni 19). *Difference between Conceptual, Logical and Physical Data Models*. Opgehaald van uksanjay.blogspot: <http://uksanjay.blogspot.com/2012/06/difference-between-conceptual-logical.html>
- Hall, J. (2019, 09 12). *jsPDF*. Opgehaald van github: <https://github.com/MrRio/jsPDF>
- Krzyczkowski, D. (sd). *Azure DevOps as a Central Hub for Managing DevOps Best Practices*. Opgehaald van predica: <https://predica.pl/blog/azure-devops/>
- nen. (2011, 09 01). Opgehaald van nen: <https://www.nen.nl/web/file?uuid=bc96cb69-7b2a-483b-a323-e7b089876947&owner=661f2b5a-9c3b-4e79-a9f9-5420b26f8bf4>
- Stackify. (2017, 05 19). *What is N-Tier*. Opgehaald van stackify: <https://stackify.com/n-tier-architecture/>
- Wikipedia. (2019, 10 10). *Bouw*. Opgehaald van Wikipedia: <https://nl.wikipedia.org/wiki/Bouw>
- Wikipedia. (2019, 12 27). *Entity–relationship model*. Opgehaald van Wikipedia:
https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model
- Wikipedia. (2019, 1 3). *Entity-relationshipmodel*. Opgehaald van <https://nl.wikipedia.org/>:
<https://nl.wikipedia.org/wiki/Entity-relationshipmodel>
- Wikipedia. (2019, 09 09). *U-waarde*. Opgehaald van Wikipedia: <https://nl.wikipedia.org/wiki/U-waarde>
- Wikipedia. (2020, 01 05). *Multitier architecture*. Opgehaald van Wikipedia:
https://en.wikipedia.org/wiki/Multitier_architecture

BIJLAGEN

BIJLAGE A - AFSTUDEERPLAN

BIJLAGE B - PLAN VAN AANPAK

BIJLAGE C – ONTWERPRAPPORT

BIJLAGE D - DEFINITION OF DONE

BIJLAGE E - DEFINITION OF READY

BIJLAGE F – CONVERSIE SCRIPTS