THOMAS MORE

Deloitte.

[redacted]

STAGE DELOITTE

Energy Consumption Dashboard

Realisatie Document

Nathan Neve
Bachelor student Toegepaste Informatica Keuzerichting Digital Innovation

2024 - 2025

Inhoudstafel

1. Voorwoord

Tijdens mijn vorige baan bij Cartamundi vroeg ik me af hoe de efficiëntie van bepaalde processen verbeterd kon worden. Na hierover in gesprek te gaan met mijn leidinggevende, zijn we gestart met een onderzoeksproject. In dit project verzamelde ik gegevens over de verschillende stappen in het productieproces, analyseerde deze en rapporteerde mijn bevindingen.

Deze ervaring liet me inzien wat voor impact data kan hebben en heeft me gemotiveerd om Toegepaste Informatica te gaan studeren aan de Thomas Morehogeschool in Geel. Tijdens mijn opleiding heb ik me volop ingezet om mezelf te ontwikkelen binnen de wereld van data, onder andere door doelbewust te kiezen voor uitdagende projecten binnen mijn afstudeerrichting Digital Innovation. Zo kwam ik bij een project voor het bedrijf Togaether voor het eerst in aanraking met Microsoft Fabric. Die eerste ervaring gaf me het vertrouwen om te solliciteren bij Deloitte voor een stage rond Fabric.

Tijdens deze stage kon ik mijn statistische kennis uit mijn vorige opleiding, mijn werkervaring en de technische vaardigheden die ik tijdens mijn studie heb opgedaan, combineren om een mooi eindresultaat na te streven. Dit realisatiedocument geeft een overzicht van wat ik heb bereikt tijdens mijn dertien weken durende stage bij Deloitte Belgium.

2. Inleiding

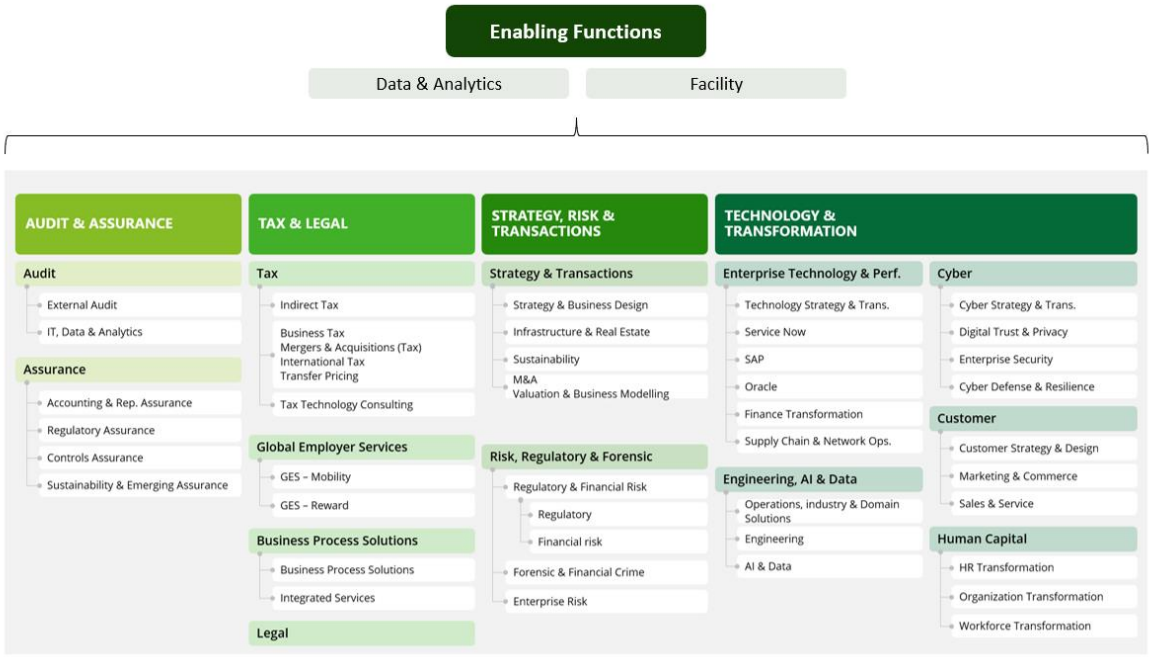
In deze inleiding volgt een korte voorstelling van mijn stagebedrijf. Aangezien Deloitte een zeer grote en diverse organisatie is, acht ik enige kadering noodzakelijk. Daarnaast licht ik mijn stageopdracht toe.

2.1. Het Stagebedrijf: Deloitte Belgium

Deloitte werd opgericht in 1845, toen William Welch Deloitte zijn eerste kantoor opende in Londen. In eerste instantie richtte het bedrijf zich uitsluitend op audit, maar sindsdien is het uitgegroeid tot een wereldwijde organisatie die actief is in tal van sectoren en regio's (Deloitte, Our heritage, 2025).

Vandaag is Deloitte het grootste dienstverlenende bedrijf ter wereld, actief in meer dan 150 landen. Het heeft projecten uitgevoerd voor meer dan 90% van de Fortune 500-bedrijven. Deloitte opereert momenteel in uiteenlopende domeinen, van audit en juridisch advies tot managementstrategieën, technologie en fiscaliteit (Deloitte, Who we are, 2025).

In het fiscaal jaar 2024 behaalde Deloitte Belgium een omzet van 819 miljoen euro, een groei van 4.4% in vergelijking met het voorgaande fiscaal jaar. De Belgische tak van Deloitte richt zich ook sterk op duurzaamheid en slaagde erin om dat jaar 31% van zijn CO2-uitstoot te reduceren. In België heeft Deloitte bijna 6.000 medewerkers in dienst en telt het 236 partners (Deloitte, 2024).



Figuur 1: Vereenvoudigd organigram van Deloitte Belgium

Ik loop stage binnen de afdeling '*Enabling Functions*'. Deze afdeling omvat alle teams die projecten uitvoeren voor Deloitte zelf, en dus niet voor externe klanten. Binnen deze context werk ik mee in het Data & Analytics-team, dat bestaat uit een vijftal vaste medewerkers. Het doel van het Data & Analytics-team is om data om te zetten in waardevolle inzichten en zo interne afdelingen te ondersteunen bij het nemen van datagedreven beslissingen. Het team houdt zich bezig met het extraheren, transformeren

en optimaliseren van data uit diverse bronnen. Het eindresultaat is vaak een dashboard voor andere teams binnen Deloitte.

De klant van mijn stageopdracht is het Facility-team van Deloitte Belgium. Zij houden zich voornamelijk bezig met het beheren van de werkomgeving, waaronder kantoorinfrastructuur, logistiek, veiligheid en duurzaamheid. Hun doel is om ervoor te zorgen dat alle werknemers op een comfortabele en duurzame manier hun werk kunnen doen. Om geïnformeerde beslissingen te kunnen nemen, doen zij beroep op het data & analytics-team.

2.2. De Stageopdracht

Het Facility-team van Deloitte Belgium heeft behoefte aan een vernieuwd dashboard dat het verbruik van energie, gas, water en andere energietypes van alle kantoren visualiseert. Gebruikers moeten in staat zijn om deze data te analyseren en te filteren op kantoorlocatie en tijdsperiode. Hierdoor kunnen verbruikspatronen en trends worden vastgesteld. Daarnaast is er vraag naar het voorspellen van verbruik en het analyseren van de impact van weersomstandigheden daarop.

Een bijkomende uitdaging is het moderniseren van de dataverzameling. Momenteel worden verbruiksgegevens verzameld via scripts die informatie van websites scrapen. Deze methode is gevoelig voor fouten en veranderingen aan de website. Daarom is het beter om deze aanpak te vervangen door API's, die betrouwbaarder, efficiënter en toekomstbestendiger zijn.

Sinds de introductie van Microsoft Fabric stappen steeds meer datagedreven organisaties over van visualisatietools zoals Tableau en Qlik naar Microsoft Fabric in combinatie met Power BI. Ook binnen Deloitte is deze trend merkbaar: het Facility-team maakt momenteel gebruik van een dashboard in Qlik Sense, maar wil migreren naar een meer geïntegreerde en krachtige oplossing binnen het Microsoft-ecosysteem.

Het doel van deze migratie is niet alleen technisch, maar ook inhoudelijk. Door over te stappen naar Power BI, kan het bestaande dashboard niet alleen worden gereconstrueerd, maar ook inhoudelijk worden verbeterd. De overstap biedt de mogelijkheid om zowel de gebruikerservaring als de visualisaties te verbeteren.

Mijn stageopdracht bestaat uit drie grote delen:

1. **Dataverzameling** – Het ontwikkelen van een robuuste en schaalbare oplossing voor het vergaren van verbruiksdata, bij voorkeur via API's. Momenteel zijn bijna alle scripts die data ophalen verouderd.

2. **Dashboardontwikkeling** – Het ontwerpen en realiseren van een interactief Power BI-dashboard binnen Microsoft Fabric, waarmee gebruikers op een intuïtieve manier inzichten kunnen verkrijgen in het energie- en middelenverbruik. Het huidige dashboard bevat niet meer de juiste en actuele data.

3. **Voorspellingen** – Uitvoeren van statistische analyses om voorspellingen te doen over het verwachte verbruik en uitstoot, met aandacht voor locatie, tijd en hoeveelheid. Hierbij wordt gekeken naar historische trends om patronen te herkennen.

Ten slotte wordt het dashboard gevalideerd door de gerapporteerde data te vergelijken met de officiële rapportage aan NSE (Deloitte North South Europe). Hierdoor kan de nauwkeurigheid van mijn dashboard worden gecontroleerd. Dit jaar vindt er naast de rapportage ook een milieuaudit plaats, wat een goed moment is om mijn uitstootberekening te valideren.

3. Analyse

In dit hoofdstuk beschrijf ik mijn planning, de werkwijze van Deloitte op het gebied van Agile en de gebruikte tools. Ten slotte beschrijf ik hoe de communicatie met de klant is verlopen en hoe ik wireframes heb ingezet om de klant een duidelijk beeld te geven van het beoogde eindresultaat.

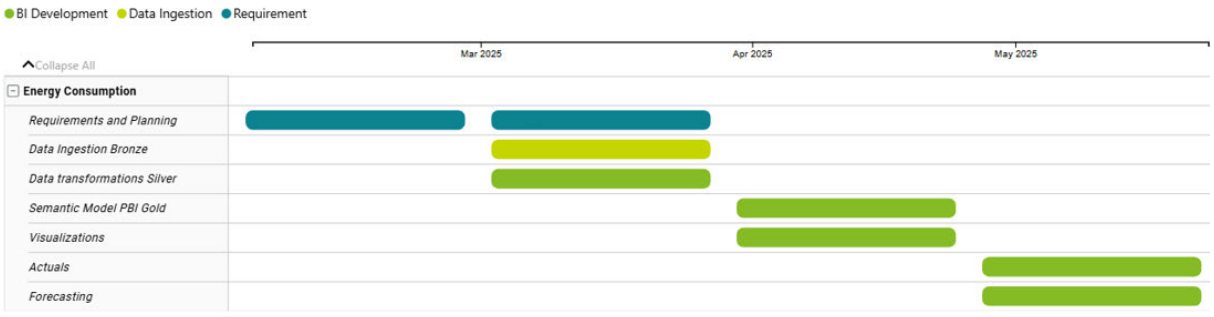
3.1. Planning

Binnen het Data & Analytics team wordt gewerkt volgens de Agile methode. Dit houdt in dat we in korte iteraties (sprints) toewerken naar opleveringen, regelmatig feedbackmomenten inplannen en flexibel inspelen op de veranderende behoeften van de klant. Deze aanpak stimuleert samenwerking, transparantie en continue verbetering gedurende het project.

Toch zijn er enkele belangrijke verschillen tussen de manier waarop Agile wordt toegepast binnen het DnA-team en de wijze waarop we werken tijdens schoolprojecten. De sprints binnen het DnA-team zijn langer en lopen over een volledige maand. Dit maakt het makkelijker om af te stemmen met de klant. Er ligt binnen het DnA-team minder nadruk op dagelijkse stand-up meetings, in plaats daarvan vinden deze slechts twee keer per week plaats. Door de kleinere schaal van de projecten en dus minder ontwikkelaars op hetzelfde project, is er minder behoefte aan dagelijkse stand-ups.

De andere sprint ceremonies waren wel vergelijkbaar met wat we gewend zijn en vonden één keer per sprint (en dus maand) plaats. Tijdens de sprint planning gaven we een score aan nieuwe user stories op basis van complexiteit. Deze complexiteitsscore komt overeen met het aantal uren dat verwacht wordt aan de taak te worden besteed.

In de sprint retrospective werden de opgeleverde resultaten besproken, evenals hoe teamleden zich voelden. Feedback werd verzameld aan de hand van positieve en negatieve punten, ook is er de mogelijkheid om verbeteringen te suggereren. De sprint review vond niet op een vast moment in de sprint plaats. Ik had de vrijheid om deze zelf in te plannen, afhankelijk van de voortgang van het project en mijn behoefte om af te stemmen met de klant.



Figuur 2: Grafiek van mijn sprints geplaatst in de tijd

Figuur 2 toont hoe mijn sprints overeenkomen met de maanden van het jaar, om zo duidelijkheid te creëren in de planning. Wanneer ik bijvoorbeeld aangeef dat het dashboard klaar is aan het einde van sprint 3, weten zowel ik als de klant als ikzelf dat dit overeenkomt met eind mei.

Er zou gesteld kunnen worden dat mijn aanpak elementen van het watervalmodel bevat, aangezien ik begin met de dataverzameling voordat ik begin met de visualisatiefase.

In werkelijkheid was deze aanpak een bewuste keuze. Ik anticipeerde erop dat er mogelijk vertraging zou ontstaan bij een van de databronnen. Door eerst de dataverzameling grondig aan te pakken, kon ik in een later stadium toch verder werken aan de visualisaties met de beschikbare data, zonder dat het hele project stilviel.

3.1.1. Fase 1: Requirements & Planning

In de eerste fase van mijn stage, nog voor de start van mijn opdracht, volgde ik interne opleidingen over ethiek op de werkvloer en het Agile-framework. Daarnaast bestudeerde ik de richtlijnen voor het omgaan met vertrouwelijke data. Vervolgens begon ik met het opstellen van mijn planning in Azure DevOps en het uitschrijven van user stories, waarvan de samenvatting in de onderstaande tabellen te vinden is.

Allereerst plande ik de modernisering van de bestaande data-integratie. Idealiter zou dit via API's gebeuren, maar wanneer deze niet beschikbaar waren, onderzocht ik de mogelijkheid om headless Selenium-scripts in Microsoft Fabric Spark Notebooks te gebruiken. Wanneer ik een duidelijk beeld zou hebben van de structuur van mijn dataset, plande ik het opzetten van mijn datamodel in de vorm van een sterschema en het uitwerken van wireframes.

	Titel	Geschatte uren werk	Sprint
Requirements & Planning	Maken van een datamodel	6	1
	Onderzoeken van headless selenium	6	1
	Onderzoeken welke energieleveranciers API's aanbieden	15	1
	Planning DevOps, schrijven van user stories	8	1
	Maken van wireframes	8	1
	Volgen van interne training	10	1
	Schrijven van het plan van aanpak	14	1

Figuur 3: Vereenvoudigde planning fase 1 - requirements en planning

Daarna bundelde ik al mijn werk in een plan van aanpak. Ik kreeg de keuze om dit uit te werken in een PowerPointpresentatie of in een tekstdocument, maar besloot beide te gebruiken: de PowerPoint om mijn aanpak toe te lichten aan mijn mentor en begeleider, en het tekstdocument om dieper in te gaan op de inhoud.

3.1.2. Fase 2: Dataverzameling

In de tweede fase van mijn stage plan ik de dataverzameling. Dit houdt in dat ik de data-integratie voor alle databronnen volledig automatiseer. Dit werk voer ik uit in de maand maart.
Allereerst laad ik de data in een 'landing area' in Microsoft Fabric, gestructureerd in tabellen per leverancier. Vervolgens transformeer ik deze tabellen naar views per energietype. Deze aanpak vormt de beste basis om de data verder te modelleren in een sterschema.

Naast de verbruiksdata verzamel ik ook een dataset met weergegevens, SLP's en CO_2-conversiefactoren. SLP's zijn factoren waarmee ik het verbruik voor elektriciteit en gas voorspel.

	Titel	Geschatte uren werk	Sprint
Dataverzameling	[redacted] automatiseren data integratie	15	1
	[redacted] automatiseren data integratie	10	1
	[redacted] automatiseren data integratie	10	1
	Weather: automatiseren data integratie	18	1
	[redacted] automatiseren data integratie	15	1
	[redacted] automatiseren data integratie	10	1
	District Verwarming & Mazout: automatiseren data integratie	10	1
	CO_2-conversiefactoren: automatiseren data integratie	15	1
	SLP: automatiseren data integratie	15	1
	Omvormen dataset per leverancier naar per energietype	40	1

Figuur 4: Vereenvoudigde planning fase 2 - dataverzameling

3.1.3. Fase 3: Dashboardontwikkeling

In de derde fase van mijn stage focus ik op het front-endgedeelte de ontwikkeling van het dashboard. Dit begint met het opzetten van een performant datamodel. Daarna start ik met het ontwikkelen van afzonderlijke dashboardpagina's per energietype. Daarnaast is er gevraagd om een overzichtspagina te maken waarop alle verbruiksgegevens gevisualiseerd worden, zodat de klant deze kan gebruiken voor rapportage aan NSE (Deloitte North South Europe). Dit werk zou verricht moeten zijn in de maand april.

Zodra deze onderdelen zijn afgerond, plan ik de berekening van de CO_2-uitstoot op basis van de verbruiksgegevens, met als doel deze visueel weer te geven. Dit doe ik aan de hand van mijn eerder verzamelde conversiefactoren.

	Titel	Geschatte uren werk	Sprint
Dashboard-ontwikkeling	Semantisch model Power BI	25	2
	Visualisaties: overzichtspagina	16	2
	Visualisaties: pagina elektriciteit	10	2
	Visualisaties: pagina verwarming	10	2
	Visualisaties: pagina facturen	10	2
	Visualisaties: pagina laadbeurten	10	2
	Visualisaties: pagina consolidatie	15	2
	Filters dashboard	6	2
	CO2 uitstoot berekening en grafiek	20	2

Figuur 5: Vereenvoudigde planning fase 3 - dashboardontwikkeling

3.1.4. Fase 4: Voorspellingen

In de vierde en laatste fase van mijn stage richt ik mij op het data science-gedeelte en ontwikkel ik modellen om het toekomstige verbruik te voorspellen. Daarnaast bouw ik een model dat inschat wanneer er een tekort aan mazout zal optreden. Dit omdat er tijdig bestellingen moeten geplaatst worden. Zodra het grootste deel van mijn werk is afgerond, begin ik met het schrijven van dit realisatiedocument.

	Titel	Geschatte uren werk	Sprint
Voorspellingen	Voorspellingen verbruik op basis van de historiek	24	3
	Voorspelling tekort aan mazout	18	3
	Schrijven van het realisatiedocument	55	3

Figuur 6: Vereenvoudigde planning fase 4 – Voorspellingen

3.2. Gebruikte tools

3.2.1. Microsoft Fabric

Microsoft Fabric is een data-analyseplatform dat organisaties helpt bij het beheren, analyseren en visualiseren van data. Het platform werkt naadloos samen met Microsoft Power BI, waarmee we onze data willen visualiseren. Fabric is gebouwd op Microsoft OneLake, een platform vergelijkbaar met OneDrive, maar dan specifiek ontworpen voor het opslaan en beheren van data op schaal. Hierdoor hoeft data slechts op één plaats opgeslagen te worden, wat het platform bijzonder aantrekkelijk maakt voor bedrijven.

Binnen Microsoft Fabric kunnen verschillende componenten worden aangemaakt, elk met een eigen functie. Hieronder licht ik deze kort toe:

- **Pipeline**: Een pipeline maakt het mogelijk om verschillende stappen of processen (zoals data-invoer, transformaties en outputs) aan elkaar te koppelen tot één geautomatiseerde workflow. Daarnaast kun je pipelines plannen zodat deze op vooraf bepaalde tijdstippen of onder bepaalde voorwaarden automatisch worden uitgevoerd.

- **Dataflow**: Een Dataflow is een low-code tool binnen Microsoft Fabric waarmee je op een eenvoudige manier data kunt integreren en transformeren (Microsoft, 2024). Dataflows zijn geoptimaliseerd voor gebruiksgemak, maar draaien niet op een gedistribueerde compute-engine zoals Spark notebooks dat doen. Dit betekent dat ze goed werken voor eenvoudige transformaties, maar minder geschikt zijn voor complexe bewerkingen. Zelf heb ik Dataflows voornamelijk gebruikt voor het integreren van Excel-bestanden vanuit OneDrive en voor het uitvoeren van eenvoudige transformaties, zoals het filteren en herschikken van kolommen.

- **Spark Notebook**: Notebooks zijn de component die ik het meest heb gebruikt binnen Microsoft Fabric. Ze draaien op een gedistribueerde compute-engine gebaseerd op Apache Spark, wat ideaal is voor complexe transformaties. In notebooks schrijven we doorgaans Python-code, bijvoorbeeld met behulp van PySpark of Pandas. Het schrijven van code biedt ons de flexibiliteit om vrijwel alles te doen wat we willen. Dit brengt ook de verantwoordelijkheid met zich mee om onze code zo efficiënt mogelijk te schrijven, aangezien alles dat we uitvoeren in Fabric capaciteit verbruikt en dat betekent kosten.

- **Environment**: Een environment in Microsoft Fabric dient als item voor al je hardware- en software-instellingen. Binnen een environment kun je verschillende Spark-runtimes selecteren, je compute-resources configureren en bibliotheken installeren vanuit openbare repositories of een lokale directory. Het handige is dat je de libraries voor je Spark-notebooks niet telkens opnieuw hoeft te installeren, en ze zelfs kunt delen met andere gebruikers en notebooks.

- **Lakehouse**: Een lakehouse in Microsoft Fabric combineert de voordelen van een data lake en een data warehouse. Het biedt een oplossing voor het opslaan, beheren en analyseren van zowel gestructureerde als ongestructureerde data. Elk lakehouse heeft een SQL-analytics endpoint, waardoor het mogelijk is om direct op de tabellen te rapporteren en SQL-query's uit te voeren (Microsoft, 2025).

Microsoft Fabric is, naast het bedrijf zelf, de reden waarom ik gesolliciteerd heb op deze stage. Ik wil hier na mijn stage professioneel mee verder werken. Omdat Fabric een nieuw platform is, merk ik dat er veel interesse is in profielen die er ervaring mee hebben. Als voorbereiding op mijn stage behaalde ik het Fabric Analytics Engineer Associate certificaat (Microsoft, n.d.).

3.2.2. Microsoft Power BI

Power BI is een business intelligence (BI) tool van Microsoft waarmee gebruikers data kunnen visualiseren en dashboards kunnen bouwen en delen. Power BI biedt krachtige functies zoals datamodellering, AI-gestuurde analyses en integratie met andere Microsoft-producten zoals Excel en Azure, maar vooral Microsoft Fabric, waarbinnen ik mijn data transformeer en opsla.

Het beste voorbeeld van de naadloze integratie van Power BI in Microsoft Fabric is de mogelijkheid om rechtstreeks dashboards te bouwen op basis van datasets die in Fabric zijn opgeslagen. In tegenstelling tot andere BI-tools en ook Power BI vóór de introductie van Fabric, wordt meestal gebruik gemaakt van importmode (Microsoft, 2025). We kunnen in Fabric gebruikmaken van Direct Lake. Dit betekent dat de data niet eerst geïmporteerd moeten worden in een Power BI-model, maar direct gebruikt worden vanuit OneLake in Microsoft Fabric (Microsoft, 2025). Dit zorgt niet alleen voor een gemakkelijker ontwikkelingsproces, maar ook voor minder duplicatie van data.

3.2.3. Qlik Sense

Het huidige dashboard dat ik moet vervangen is ontwikkeld in Qlik Sense. Dit platform biedt ongeveer dezelfde mogelijkheden op het gebied van visualisaties, maar mist de integratie met Microsoft Fabric, waar ik mijn ETL-processen zal uitvoeren. Bovendien biedt Qlik Sense geen ondersteuning voor het inplannen van API-calls of het uitvoeren van complexe transformaties. Zelfs eenvoudige taken, zoals het uitpakken van een JSON-bestand, vereisen in Qlik Sense vaak omslachtige oplossingen of extra scripting. Dit maakt het minder geschikt voor meer geavanceerde workflows.

3.2.4. Robotgestuurde procesautomatisering (RPA)

Het Data & Analytics team beschikt voor een groot deel van de leveranciers over een RPA. Deze code gaat op een geautomatiseerde manier de verbruiksgegevens ophalen bij de verschillende energieleveranciers. Deze RPA's zijn geschreven in Selenium. RPA's werken door een browser te openen en de handelingen van een menselijke gebruiker stap voor stap na te bootsen op een website. Vaak is het doel om op een exportknop te klikken, waarna er bijvoorbeeld een Excel-bestand wordt gedownload. Ze werken dus via de front-end van de website en niet via een API of de back-end, wat

het proces gevoelig maakt voor veranderingen. Stel dat een leverancier zijn website verandert, zou mijn RPA niet meer werken.

3.2.5. Azure DevOps

Azure DevOps is het DevOps-platform van Microsoft dat door ons team wordt gebruikt om samenwerking te structureren. Binnen het Data & Analytics team maken we gebruik van verschillende functionaliteiten van Azure DevOps. Zo gebruiken we het platform tijdens planning poker-sessies om story points toe te kennen aan user stories en taken. Daarnaast biedt Azure DevOps ondersteuning voor het organiseren van sprint retrospectives, waarbij teamleden feedback kunnen geven en verbeterpunten kunnen aanbrengen. Ook gebruiken we dit platform voor het beheren van backlogs, het plannen van sprints en het opvolgen van de voortgang via Kanban borden.

3.2.6. Mokkup.ai

Mokkup.ai is een online tool waarmee je snel wireframes kunt maken. Het stelt me in staat om al in een vroege fase van het ontwikkelingsproces feedback te verzamelen van stakeholders. Wanneer zij bepaalde visualisaties of gegevens missen, kan ik hiermee vroeg in het ontwikkelingsproces rekening houden, wat de kans op onnodig werk verkleint later in mijn stage.

Op figuur 7 is een gewogen beslissingsmatrix (WDM) te zien waarin ik verschillende wireframingtools heb vergeleken. Mokkup.ai kwam hierbij als beste optie naar voren. De tool is specifiek ontworpen voor het maken van dashboards. Daarnaast is het platform gratis te gebruiken en gebruiksvriendelijk.

Criteria	Gewicht	Mokkup.ai	Figma	Balsamiq	Adobe XD
Gebruiksgemak	6	9	8	7	7
Gericht op dashboards	5	9	6	5	6
Snelheid van prototyping	4	9	7	6	6
Samenwerkings-mogelijkheden	3	8	9	6	7
Prijs	2	8	6	8	5
Integratiemogelijkheden	1	8	7	4	5
Totale gewogen score		**151**	125	104	111

Figuur 7: weighted decision matrix (WDM) van wireframing tools

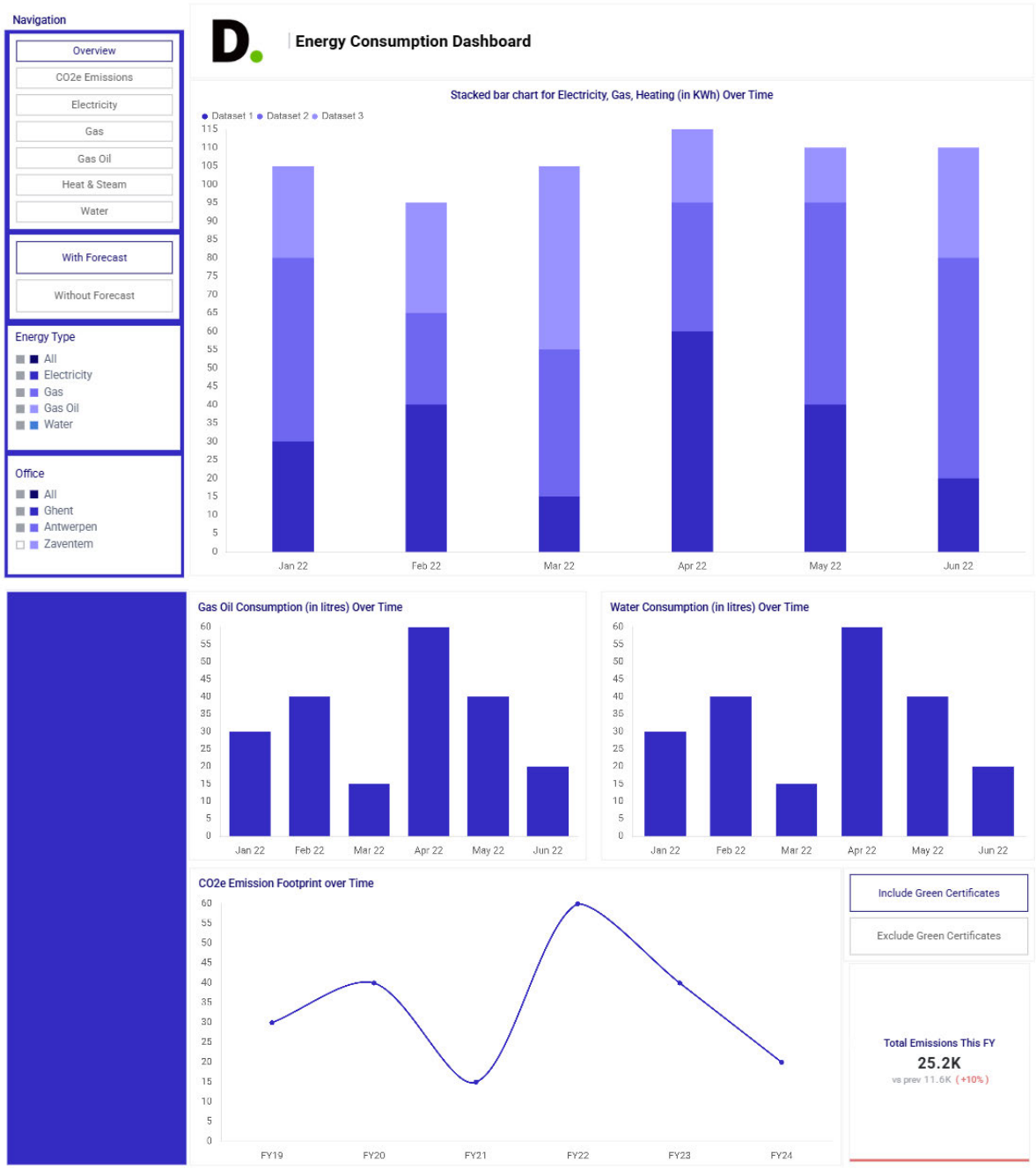
3.2.7. Bizagi

Om alle processen, die data verzamelen in functie van visualisatie te documenteren, maak ik gebruik van Bizagi. Dit is een Business Process Management (BPM) tool die helpt bij het modelleren en optimaliseren van bedrijfsprocessen. Met Bizagi kan ik workflows creëren, de efficiëntie van processen verbeteren en ervoor zorgen dat alle stappen duidelijk gedocumenteerd zijn. Hierdoor wordt het eenvoudiger te zien welke stappen efficiënter kunnen.

3.3. Wireframes

In de zesde week van mijn stage had ik een meeting met mijn stagementor, buddy en de klant. Tijdens deze meeting had ik de gelegenheid om mijn wireframes te presenteren, wat de klant een beter beeld gaf van wat ik wilde realiseren. De klant maakte van de gelegenheid gebruik om feedback te geven, zodat ik de nodige aanpassingen kon doen. Hieronder toon ik alle schermen en de aanpassingen die ik heb aangebracht op basis van de feedback van de klant.

3.3.1. Pagina Overzicht



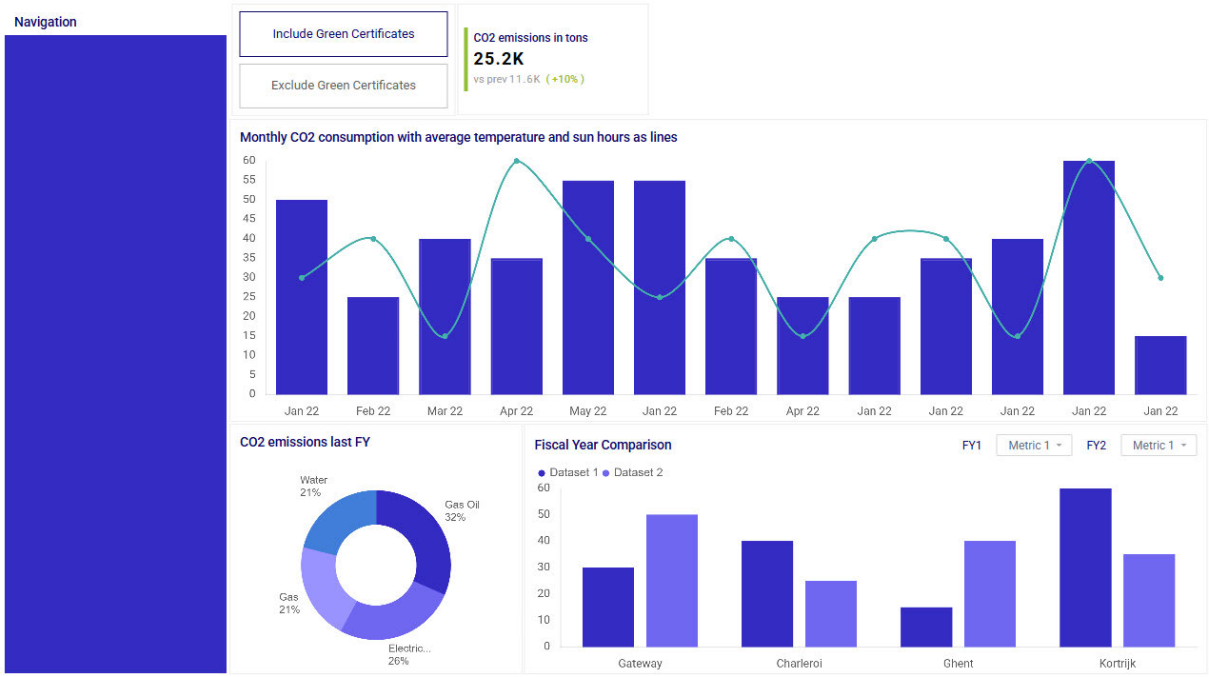
Figuur 8: wireframe van de overview pagina

De overzichtspagina is bedoeld om snel inzicht te geven in het energieverbruik per energietype en per kantoor. Aan de linkerkant bevinden zich filters waarmee gebruikers kunnen selecteren op kantoor en energietype. Dit maakt het dashboard interactief en gebruiksvriendelijk.

Oorspronkelijk was mijn idee om een gestapelde staafdiagram te gebruiken voor elektriciteit, gas en district verwarming, aangezien deze energietypes allemaal in megawattuur worden uitgedrukt. Al snel bleek echter dat het verbruik van gas en district verwarming slechts een fractie is van het elektriciteitsverbruik, waardoor deze energietypes in een gestapelde staafdiagram nauwelijks zichtbaar zouden zijn. Daarom werd besloten om alle energietypes hun eigen bar chart te geven. Daarnaast werd besloten om op de x-as de fiscale jaren weer te geven, met de mogelijkheid om via een drilldown naar maanden of zelfs dagen in te zoomen.

Het originele dashboard in Qlik bevat veel bar charts, wat doorgaans niet ideaal is, aangezien het herhaald gebruiken van dezelfde visualisatievorm de leesbaarheid belemmert. Op verzoek van de klant werd deze grafiek op de overzichtspagina toegevoegd. Daarna ga ik in een bar chart het verbruik van water en mazout tonen. Tot slot wil ik een berekening tonen van de CO_2-uitstoot die wordt gegenereerd door het energieverbruik in de vorm van een line chart.

3.3.2. Pagina CO_2-uitstoot



Figuur 9: wireframe van de CO_2-uitstoot Pagina

Mijn intentie voor de CO_2-pagina was om de invloed van het weer op de uitstoot te visualiseren. Daarnaast wilde ik met een donut chart laten zien welke kantoren de meeste CO_2 uitstoten. Ook was het plan om de CO_2-trend weer te geven met een clustered bar chart, waarbij de x-as de kantoren zou weergeven en de y-as de uitstoot.

Na overleg met de klant en mijn stagementor werd besloten om deze pagina te verwijderen. In plaats daarvan zou de CO_2-uitstoot een eigen dashboard krijgen. Bijkomend diende ik alle benodigde berekeningen uit te voeren om de CO_2-uitstoot van energieverbruik te berekenen.

3.3.3. Pagina Elektriciteit, Gas en Facturen

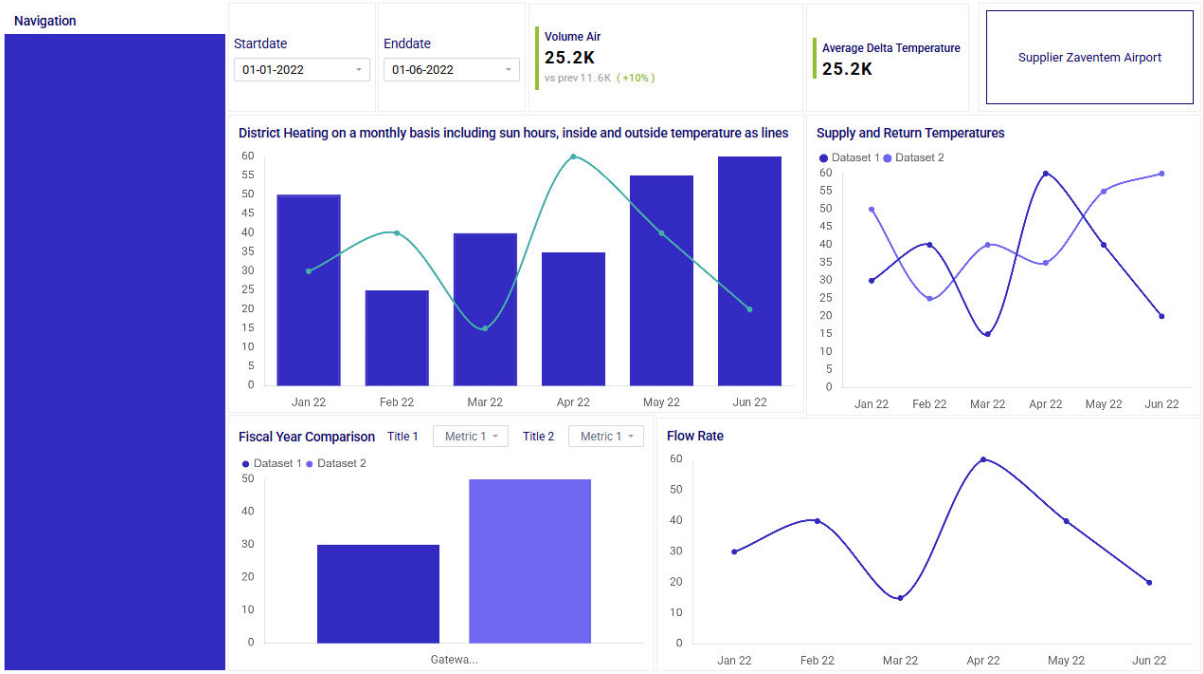
Figuur 10: wireframe van de elektriciteit pagina

De wireframes voor de gas- en elektriciteitspagina bevatten grotendeels dezelfde visualisaties. Een belangrijk verschil is het taartdiagram, dat specifiek op deze pagina wordt gebruikt om het piek- en niet-piekverbruik te vergelijken. Daarnaast is er op deze pagina een filter toegevoegd waarmee gebruikers het energietype kunnen selecteren.

Bovenaan bevinden zich KPI's die het totale energieverbruik weergeven. Verder visualiseer ik het energieverbruik in combinatie met de gemiddelde temperatuur en zonuren in een combo chart, om mogelijke correlaties te tonen. Onderaan wordt een clustered column chart gebruikt om het energieverbruik tussen verschillende kantoren of perioden te vergelijken.

Onderaan toon ik een gestapelde staafdiagram waarmee gebruikers de kosten van facturen kunnen bekijken, opgesplitst naar energie, transport, distributie en taksen. We hebben besloten om deze visualisatie een eigen pagina te geven, met filters op kostencategorie, factuurnummer en energietype. Verder hebben we ook besloten om de consumptiegegevens van de zonnepanelen toe te voegen op de pagina van elektriciteit. Daarnaast geef ik ook weer hoeveel CO_2 wordt uitgespaard door de energieproductie van de zonnepanelen.

3.3.4. Pagina District Verwarming



Figuur 11: wireframe van district verwarming pagina

De pagina voor stadsverwarming toont KPI's voor verbruik en de delta van de temperatuur die wordt teruggegeven door de luchthaven. Daarnaast visualiseer ik de verbruiksgegevens in relatie tot de buitentemperatuur. Ook wordt een clustered bar chart weergegeven, per kantoor en per fiscaal jaar.

Tijdens mijn analyse van de dataset zag ik de mogelijkheid om meer gegevens te tonen, zoals de 'flow rate' (de hoeveelheid lucht die door het systeem circuleert) en de aanvoer- en aflevertemperaturen. De klant gaf echter aan dat ze deze grafieken niet zouden gebruiken, dus heb ik ervoor gekozen ze niet op te nemen in het eindresultaat.

3.3.5. Pagina Mazout

Figuur 12: wireframe van mazout pagina

Op de mazoutpagina toon ik het aantal verbruikte liters als KPI bovenaan. Later heb ik ervoor gekozen om daarnaast ook aan te geven of er een kans is op een brandstoftekort. Verder visualiseer ik, zoals op de andere schermen, een combo chart met het verbruik per fiscaal jaar (op verzoek van de klant) en het verband met de gemiddelde temperatuur en zonuren.

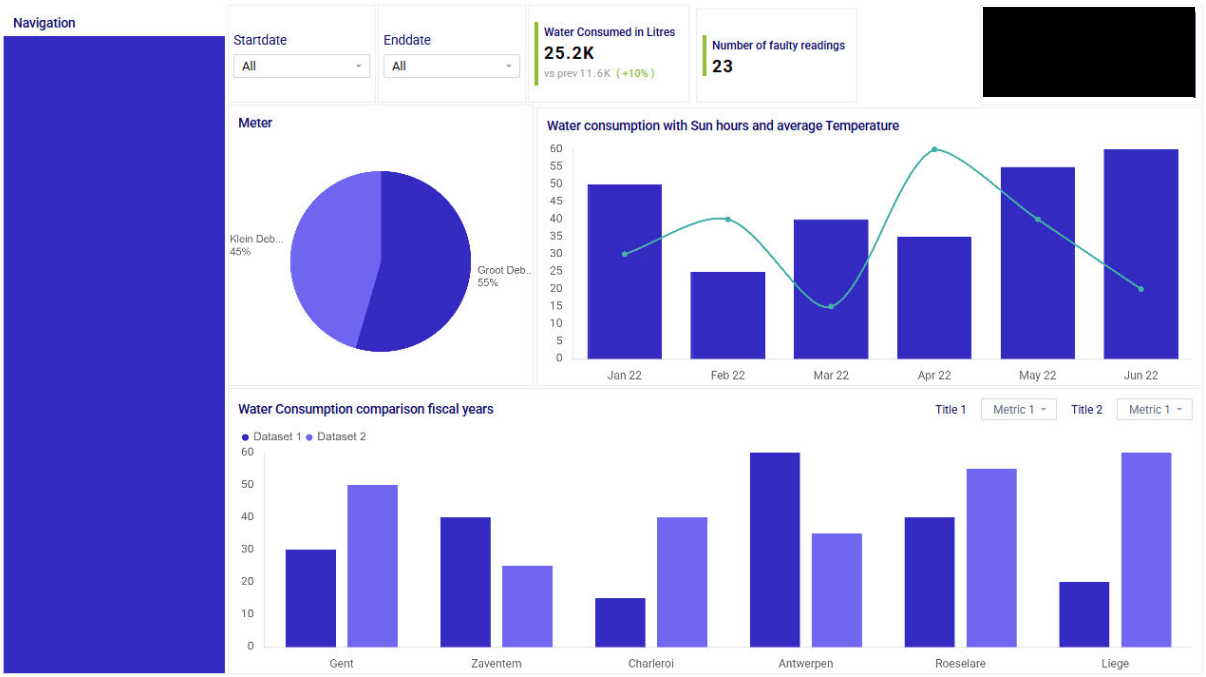
Tot slot toon ik onderaan een clustered column chart die het verbruik per fiscaal jaar en kantoor visualiseert. Het enige kantoor dat nog mazout gebruikt, is [redacted] Mazout wordt op termijn uitgefaseerd.

3.3.6. Pagina Water

Op de waterverbruikspagina zie je bovenaan opnieuw KPI's die het verbruik weergeven. Daarnaast leek het me een goed idee om de mogelijkheid te bieden om te filteren tussen een begin- en einddatum voor de weergegeven data. Verder gebruik ik een taartdiagram dat visualiseert hoeveel van het waterverbruik onder klein en groot debiet valt.

Zoals op de andere pagina's, verwerk ik ook hier een clustered column chart per fiscaal jaar en kantoor, die het verbruik weergeeft. Bovendien wordt een combo chart gepresenteerd waarin het verbruik is gekoppeld aan de gemiddelde temperatuur en zonuren.

De klant gaf feedback dat het filteren op tijd beter vormgegeven kan worden met checkboxes per fiscaal jaar. Daarnaast vonden ze het taartdiagram dat het verbruik voor klein en groot debiet toont, niet noodzakelijk. Ook het aantal foutieve lezingen van de sensoren, waarvan we de data uitlezen, werd als overbodig beschouwd.



Figuur 13: wireframe van water pagina

3.4. Conclusie Analysefase

Na de onboarding bij Deloitte begon ik meteen met het opstellen van mijn planning en het uitschrijven van user stories in Azure DevOps. Achteraf gezien bleek mijn planning goed ingeschat. Aan het einde van de sprints bleef alleen het werk over dat geblokkeerd was door externe factoren. Dit werk nam ik mee naar de volgende sprint, maar nooit had ik het gevoel dat ik te weinig of te veel werk had.

Naast het opstellen van mijn planning heb ik ook onderzoek gedaan naar de meest geschikte wireframingtools. De overige tools hoefden niet onderzocht te worden, aangezien Microsoft Fabric de belangrijkste reden was waarom ik op deze stage heb gesolliciteerd.

Ten slotte heb ik veel tijd besteed aan het begrijpen van wat de klant precies wilde en toetste ik dit aan de hand van een presentatie met wireframes van het beoogde resultaat. De grootste aanbevelingen waren het weglaten van overbodige visualisaties en het toevoegen van een filter op fiscaal jaar.

4. Realisatie

In dit hoofdstuk geef ik een overzicht van wat ik tijdens mijn stage heb gerealiseerd en licht ik toe hoe ik daarbij te werk ben gegaan. Ik begin met een toelichting over hoe ik mijn dataset heb samengesteld en welke bronnen ik daarvoor heb geïntegreerd. Vervolgens toon ik de methode die ik heb gebruikt om voorspellingen te doen op basis van de verbruiksgegevens. Tot slot presenteer ik het eindresultaat: het dashboard, waarbij ik de gekozen visualisaties en functionaliteiten verder toelicht.

4.1. Data Engineering

De grootste uitdaging bij het samenstellen van een dataset voor het dashboard is de verscheidenheid aan databronnen. Sommige kantoorgebouwen worden gedeeld met andere bedrijven, waardoor Deloitte geen invloed heeft op de keuze van energieleverancier. Hierdoor ontstaat een versnipperd datalandschap, met informatie uit diverse leveranciers, systemen en bestandsformaten.

Het integreren van al deze bronnen vereist het samenbrengen van uiteenlopende kolomnamen en datatypes. Bovendien verschilt de kwaliteit van de data per bron, wat het noodzakelijk maakt om extra aandacht te besteden aan het opschonen en standaardiseren van de gegevens voordat ik visualisaties kan maken of verbruiksvoorspellingen kan doen.

Figuur 14 geeft een overzicht van welke leverancier verantwoordelijk is voor welk energietype per kantoor.

API	
Excel	
RPA	

Office	Elektriciteit	Gas	Water	District-verwarming	Mazout	EV	Zonne-energie
[redacted]							
Boitsfort							

Figuur 14: tabel van energieleveranciers per kantoor

Er zijn wel enkele belangrijke kanttekeningen bij de tabel (figuur 14):

- Omdat Extension en Old Terminal beide deel uitmaken van het Brusselse Gateway-kantoor, hebben we besloten om deze te beschouwen als één kantoor.

- Er wordt geen onderscheid gemaakt tussen oude en nieuwe locaties van eenzelfde stad. Het Antwerpse kantoor is recentelijk verhuisd naar de Keyserlei, maar om de vergelijking eenvoudig te houden, worden beide locaties als één kantoor beschouwd.

- Er zijn twee kantoren met zonnepanelen: [redacted] Helaas was het niet mogelijk om de gegevens van het [redacted] kantoor te verkrijgen.

- De kantoren in [redacted] worden beheerd door een externe partij, die alle data via een Excel-bestand aan het Facility-team doorgeeft. Aangezien er veel fouten in deze data zaten en de formaten van de Excel-bestanden bijna jaarlijks veranderen, hebben we besloten deze kantoren niet te integreren in het dashboard. Om de integratie van deze data automatisch en duurzaam te kunnen programmeren, zijn consistente Excel-formaten nodig. Ik heb ervoor gezorgd dat alle logica zo dynamisch mogelijk is. Als er bijvoorbeeld wordt besloten om [redacted] als waterleverancier voor een van deze kantoren te gebruiken, hoeft er geen extra code geschreven te worden om het te integreren in het dashboard.

Naast energieleveranciers zijn er nog een aantal andere databronnen die ik nodig heb. Ik heb CO2-conversiefactoren, het aantal zonuren en de gemiddelde temperatuur voor al mijn kantoren nodig. Verder heb ik ook nood aan de synthetische belastingprofielen (SLP's), dit zijn factoren die ik integreer om het verbruik van gas en elektriciteit te voorspellen.

4.1.1. Medaillonarchitectuur

Om te begrijpen hoe ruwe data wordt omgezet in tabellen die klaar zijn voor visualisaties, moet de architectuur waarmee het Data & Analytics team werkt toegelicht worden. De medaillonarchitectuur is een patroon voor dataverwerking, waarbij data door verschillende lagen gaat:

Bronzen Laag (Bronze Layer): Hier komt de rauwe data binnen, we kunnen ook al basistransformaties doen. Dit doen we om de data leesbaar en bruikbaar te maken voor de volgende lagen. Voor sommige databronnen heb ik gekozen om hier data te aggregeren per dag in plaats van per kwartier, om opslagkosten te besparen. Het einddoel van mijn data in de bronzen laag is een tabel per energieleverancier vorm te geven.

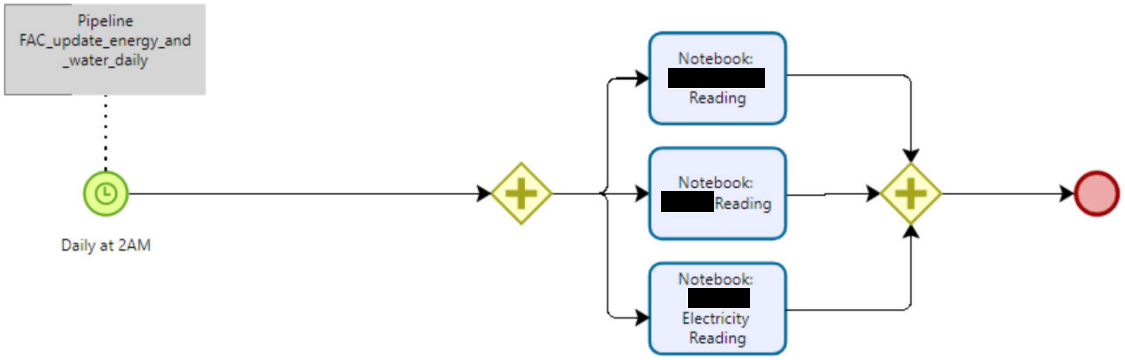
Zilveren Laag (Silver Layer): In deze laag worden de data verder opgeschoond en verrijkt. Complexere transformaties en aggregaties worden uitgevoerd om de data te structureren en te normaliseren. Mijn data in de zilveren laag bestaan uit views per energietype.

Gouden Laag (Gold Layer): Dit is de laatste laag waar de data volledig opgeschoond, geoptimaliseerd en klaar zijn voor visualisatie en analyse. In deze laag giet ik mijn views van de zilveren laag in dimensietabellen en een fact tabel (Geeksforgeeks, 2024).

4.1.2. Waterleveranciers

Er zijn twee waterleveranciers voor de kantoren van Deloitte Belgium, namelijk [redacted] Beide leveranciers bieden API's aan, waardoor het ontwikkelen van de integratie relatief eenvoudig was vergeleken met andere leveranciers. Elke nacht om 2 uur worden beide notebooks uitgevoerd en wordt onze dataset bijgewerkt. Voor beide leveranciers kijk ik naar de meest recente meting in mijn dataset

en vraag vervolgens de periode tussen dat tijdstip en nu op. Hierdoor zorgen we ervoor dat, mocht één van de platformen niet werken, we deze metingen alsnog verkrijgen.



Figuur 15: BPMN-model integratie data van waterleveranciers

[redacted]

Figuur 16: flowchart van de integratielogica voor Oktow

Figuur 16 toont de vier API-calls die ik maak om de gegevens van [redacted] te verkrijgen. Eerst ontvang ik een refresh token, gevolgd door een access token. Hierna haal ik de ID's van alle meters van de kantoren op. Ten slotte gebruik ik deze ID's in mijn laatste API-call om de metingen op te vragen. Het aanmaken van de refresh token vereist gebruikersnaam en wachtwoord, het aanmaken van de access token niet.

In het oorspronkelijke dashboard is het niet gelukt om [redacted] te integreren, aangezien de laatste API-call een geneste JSON teruggeeft. Dit is zeer lastig te interpreteren in Qlik, terwijl het in een Spark Notebook eenvoudig is. Al deze logica wordt in één notebook uitgevoerd en is robuust. Wanneer er een nieuwe meter aan een kantoor wordt toegevoegd, zullen deze metingen automatisch in het dashboard verschijnen.

Op figuur 15 zien we als onderste activiteit een notebook die elektriciteitslezingen verzamelt via het [redacted]-platform. Dit doen we omdat het ons niet gelukt is om de integratie van [redacted] te automatiseren. Dit komt omdat er (nog) geen API beschikbaar is en de site afgeschermd is door een Captcha. In de kantoren [redacted] zijn er sensoren van [redacted] op de transformatoren. De officiële meter staat voor de transformatoren, daarom gebruiken we de elektriciteitsgegevens van [redacted] alleen wanneer we geen data van [redacted] hebben.

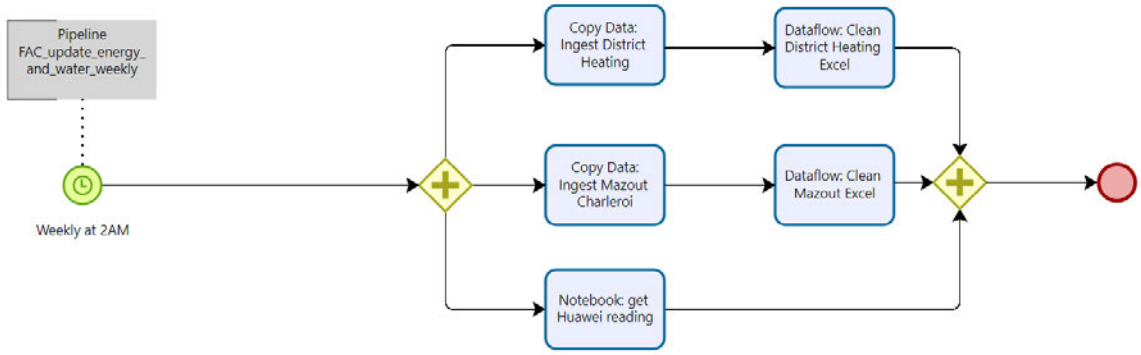
SMARTVATTEN

[redacted] is een soortgelijk platform als [redacted] dat zich richt op het monitoren en beheren van waterverbruik. [redacted]

[redacted] biedt ook API's aan, wat het ontwikkelingsproces vereenvoudigt. Voor authenticatie gebruiken we een API-key. Vervolgens vragen we alle meters op, waarbij er één meter per kantoor is. Ten slotte vraag ik de metingen op van elke meter. De logica voor [redacted] is dus vergelijkbaar met die van [redacted] met uitzondering van de authenticatie.

4.1.3. Districtverwarming

Districtverwarming is een systeem waarbij warmte van één gebouw via geïsoleerde leidingen naar meerdere gebouwen wordt gedistribueerd. In het hoofdkantoor van Deloitte Belgium, Gateway in Zaventem, wordt dit systeem gebruikt voor verwarming. Hierbij wordt koude lucht van het kantoor naar de luchthaven afgevoerd, terwijl warme lucht van de luchthaven terug naar het kantoor vloeit. Dit proces draagt bij aan een efficiënte en milieuvriendelijke warmtevoorziening, waardoor de ecologische voetafdruk van het gebouw verlaagt.



Figuur 17: BPMN-model integratie districtverwarming, mazout en zonnepanelen

De data van districtverwarming worden door het Facility-team manueel ingegeven in een Excel. Er is momenteel geen manier om dit te automatiseren zonder Excel. Aangezien deze Excel niet dagelijks wordt bijgewerkt, worden de stappen om de data te integreren wekelijks uitgevoerd. In figuur 17 zien we dat de Excel wordt ingeladen in ons lakehouse met een ‘copy data activity’. Vervolgens gebruiken we een dataflow om deze data op te schonen. Omdat de data handmatig worden ingevoerd, bevatten de brondata fouten en zullen er waarschijnlijk in de toekomst nog meer fouten in verschijnen. Daarom zorg ik er bijvoorbeeld voor dat punten en komma’s, die door mensen verkeerd worden ingevoerd, worden gestandaardiseerd en correct verwerkt. Hierna gieten we de data in een tabel.

4.1.4. Mazout

Het Deloitte-kantoor in [redacted] stookt nog steeds mazout voor verwarming. Net zoals bij districtverwarming, worden de data handmatig ingevoerd in een Excel-bestand. Een paar keer per jaar worden er enkele duizenden liters mazout geleverd, en deze leveringen worden vervolgens ingevoerd in het Excel-bestand.

In figuur 17 zien we dat de data-integratie op dezelfde manier gebeurt als bij districtverwarming. Eerst laden we het Excel-bestand in het lakehouse van de bronzen laag met behulp van een copy data activity. Vervolgens schonen we de data op en plaatsen deze in een tabel.

4.1.5. Zonne-energie

De kantoren in [redacted] maken gebruik van zonne-energie. Het kantoor [redacted] het [redacted] voor het monitoren van de zonnepanelen. Het kantoor in [redacted] wordt echter niet door Deloitte beheerd, waardoor we geen zicht hebben op de energieproductie. Voor het opvragen van gegevens van het [redacted] is er uitgebreide documentatie beschikbaar, zelfs een Python-library: [redacted]

```
1  def query_monthly(user: str, password: str, timestamp: datetime):
2      '''
3      Function that queries [redacted] and returns the KPI's for that month for all our solar installations.
4      It also returns the plants, because we want to know what plants correspond to which KPI's.
5      '''
6      [redacted]
```

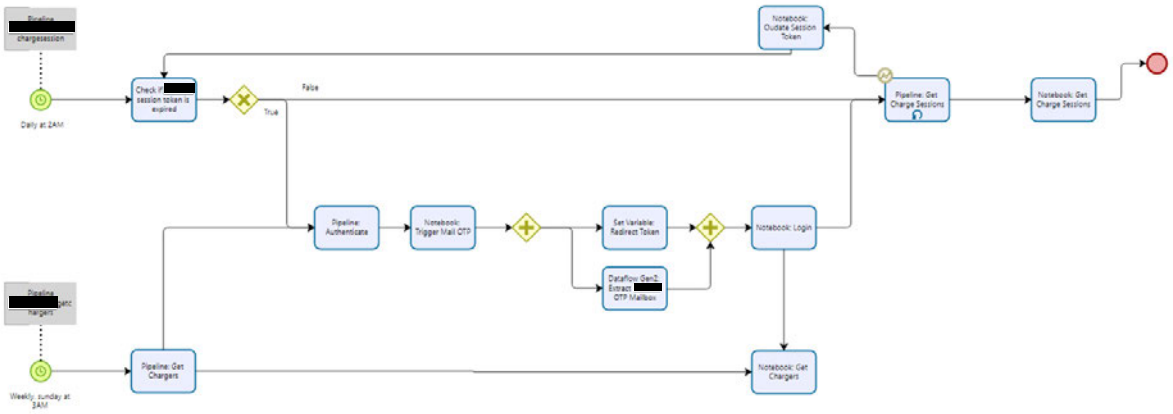
Figuur 18: functie voor het maandelijks opvragen van gegevens van zonnepanelen met de [redacted]

Op figuur 18 is te zien hoe eenvoudig het is om data op te vragen via de Python-library. Eerst maken we een client object aan met gebruikersnaam en wachtwoord. Vervolgens kunnen we dit object gebruiken om alle plantcodes op te vragen, wat de unieke codes zijn voor elk kantoor. Daarna vragen we alle productiegegevens van de zonnepanelen op.

Dit is een goed voorbeeld van een robuuste code. Als een ander kantoor gebruik gaat maken van het [redacted], zal dat kantoor ook een unieke code krijgen. Deze code ontvangen we vervolgens met de [redacted] Daarna gebruiken we deze code om alle productiegegevens op te vragen. Een ontwikkelaar moet dit dus niet herschrijven om nieuwe kantoren automatisch op te nemen in onze dataset. Na het ontvangen van onze waarden, controleren we op duplicaten en schrijven we deze weg in onze delta tabel.

4.1.6. Oplaadbeurten

Deloitte Belgium maakt gebruik van het [redacted] voor het monitoren en beheren van oplaadbeurten van elektrische wagens. In 2021 maakten ze nog gebruik van het [redacted] Deze historische data heb ik eenmalig geïntegreerd in Fabric. Hierna ben ik aan de slag gegaan met het integreren van het [redacted] bleek veruit het moeilijkste platform om te integreren.



Figuur 19 BPMN-model integratie[redacted]

De moeilijkheid bij het integreren van data uit het[redacted] was de authenticatie. [redacted] vereist multi-factor authenticatie (MFA), wat zeer moeilijk te automatiseren is, maar zeker niet onmogelijk. Allereerst heb ik een tabel aangemaakt voor het opslaan van de sessietoken. Aangezien de logica om te authenticeren complex is, willen we onze token bewaren om onnodige authenticatie te vermijden. Naast de sessietoken houden we ook een vervaldatum bij, zodat we als eerste stap kunnen bepalen of we opnieuw moeten authenticeren of niet. Als onze token geldig is, kunnen we de meest recente oplaadbeurten tot en met gisteren opvragen en opslaan in de tabel met de notebook “get charge session”. Als dit faalt, wat kan als we de API al te vaak hebben opgeroepen, zullen we de authenticatiepipeline opnieuw uitvoeren en een nieuwe sessie starten.

De eerste stap in onze authenticatiepipeline is ervoor zorgen dat er een e-mail wordt verzonden met een eenmalig wachtwoord (OTP). We doen dit door een POST-request te sturen naar de website van het[redacted] waarbij we wachtwoord en gebruikersnaam meesturen. Het is cruciaal dat bij het versturen van een verzoek de parameter ‘allow_redirects’ op false staat. Hierdoor wordt er geen automatische doorverwijzing uitgevoerd, wat ons in staat stelt om daarna nog een POST-request te sturen en aan te geven dat we via een OTP-mail willen authenticeren.

```python
login_url = 'fake_ev_charging_platform.lms'
headers = {
    'Content-Type': 'application/x-www-form-urlencoded'
}

data = {
    'emailField': 'documentation@realizationdocument.com',
    'passwordField': 'fakepassword',
    'Login': 'Log in'
}

# Create a session
session = requests.Session()

# Send POST, allow redirects is set to false, so we can inspect the response manually.
# If we set allow redirects to True, we will get a 200 and no redirect token in de headers.
# We need this token to trigger the MFA with the next API call.
response = session.post(login_url, data=data, headers=headers, allow_redirects=False)
# This needs to be 302
print(f"Response code: {response.status_code}")

# Extract cookies
PHPSESSID = session.cookies.get("PHPSESSID")
SERVERID = session.cookies.get("SERVERID")
print(f"PHPSESSID: {PHPSESSID}")
print(f"SERVERID: {SERVERID}")
```

Figuur 20 Code om te authenticeren op [redacted]

```python
url = f'fake_ev_charging_platform/FallBackOTPEmailSubmit?token={token}'
headers = {
    'Content-Type': 'application/x-www-form-urlencoded'
}

# Set the cookies
cookies = {
    'PHPSESSID': f'{PHPSESSID}',
    'SERVERID': f'{SERVERID}'
}

response = requests.get(url, headers=headers, cookies=cookies, allow_redirects=False)
# This needs to be 302
print(f"Response code: {response.status_code}")
```

Figuur 21: Code om een OTP mail te laten sturen [redacted]

Aan de hand van deze cookies kunnen we het volgende POST-request sturen, ditmaal om aan te geven dat we willen aanmelden met een OTP. Omdat we dezelfde cookies gebruiken weet de server dat het om dezelfde sessie gaat.

Deze sessietoken gaan we met een ‘set variable’ activiteit persistent maken in de pipeline (Deloitte, 2025). Hierdoor kunnen we de token verder gebruiken om in te loggen. Daarna introduceren we een pauze van 1 minuut om zeker te zijn dat de mail is verstuurd. Parallel aan deze activiteit gaan we de OTP extraheren uit onze mailbox. Met een dataflow is het mogelijk om te verbinden met een Microsoft Exchange-server. Hier kunnen we de e-mail met de OTP ophalen en gebruiken voor de verdere authenticatie. Ten slotte gaan we in de laatste notebook van onze pipeline authenticeren met behulp van de OTP. Deze laatste API-call geeft ons een sessie-id, vervaldatum en server-id, die we persistent gaan maken in een delta tabel.

Nu we al het nodige hebben verzameld, kunnen we de benodigde data van het [redacted] extraheren en opslaan in Microsoft Fabric. Een groot deel van deze zaken waren nice-to-haves:

Type	Prioriteit	Frequentie
[redacted]	Must Have	Dagelijks
[redacted]	Must Have	Wekelijks
[redacted]	Nice To Have	Wekelijks
[redacted]	Nice To Have	Wekelijks
[redacted]	Nice To Have	Wekelijks

Figuur 22 tabel met type data[redacted]

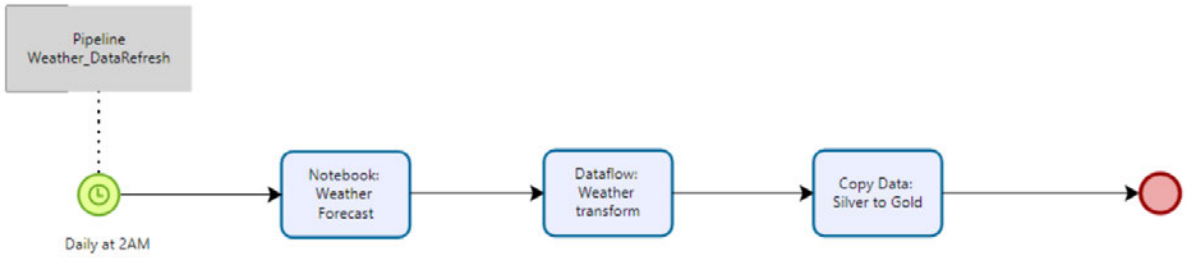
4.1.7. Elektriciteit en Gas

[redacted] de hoofdleverancier van elektriciteit en gas voor alle kantoren van Deloitte Door de aanwezigheid van een captcha op de website van [redacted] kon ik het inlogproces niet automatiseren. Daarnaast bood [redacted] geen publieke API aan waarmee de verbruiksgegevens automatisch opgevraagd konden worden. Om dit te omzeilen, heb ik geprobeerd de data rechtstreeks uit de backend van de website op te halen. Dit is een aanpak die bij andere leveranciers is gelukt. Bij [redacted] bleek dit echter niet mogelijk, doordat de captcha-verificatie niet te omzeilen bleek het ophalen van data. Hierdoor moest ik de integratie handmatig uitvoeren. Op termijn wil het team opnieuw gebruikmaken van RPA. Hiervoor heeft [redacted] de IP-adressen van Deloitte op een whitelist gezet en zou captcha-verificatie niet meer nodig zijn. Daarnaast zou Engie deze zomer een API ontwikkelen, wat een betere oplossing zou zijn.

4.1.8. Temperatuur en Zonne-uren

Om de correlatie tussen het weer en de productie van zonne-energie en het energieverbruik voor verwarming te visualiseren, heb ik weerdata nodig. Hiervoor gebruik ik Open-Meteo, een gratis open-source API (Open-Meteo, n.d.). Open-Meteo maakt gebruik van lokale en internationale weerdiensten en stelt deze beschikbaar via een zeer gebruiksvriendelijke API. Ze bieden een breed scala gegevens aan, zoals neerslag, wind en temperatuur. Voor dit dashboard heb ik gekozen voor zonne-uren en de gemiddelde temperatuur.

Data & Analytics beschikt over een tabel met alle Deloitte-kantoren en hun coördinaten. Met behulp van deze coördinaten heb ik de historische gegevens van zonne-uren en temperatuur voor alle kantoren tot vier jaar terug in de tijd opgebouwd.



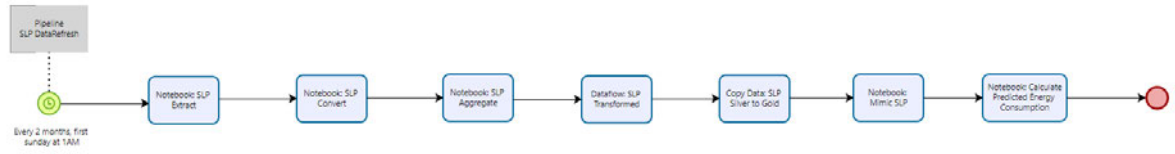
Figuur 23 BPMN-model integratie weerdata

Op figuur 23 zien we het dagelijkse proces om de weerdata en voorspellingen up-to-date te houden. De eerste notebook verwijdert alle rijen die data bevatten na de huidige datum. Vervolgens wordt de API opgeroepen om de ontbrekende historische gegevens op te halen, vanaf de meest recente datum tot en met vandaag. Ten slotte wordt de API aangeroepen voor weersvoorspellingen, tot twee weken in de toekomst. Dit doen we uiteraard voor alle kantoren van Deloitte Belgium. Vervolgens gaan we met een

dataflow de weerdata naar silver pushen en doen we een aantal transformaties. Als laatste kopiëren we de data van zilver naar goud, aangezien er geen transformaties meer nodig zijn.

4.1.9. Synthetische Lastprofielen (SLP's)

Synthetische lastprofielen (SLP's) zijn gestandaardiseerde profielen die het energieverbruik van een typische gebruiker over een bepaalde periode weergeven. Ik gebruik ze om het energieverbruik te voorspellen wanneer er geen gedetailleerde meetgegevens beschikbaar zijn (Synergrid, 2025). Concreet zijn het factoren op dagelijks niveau, de som van alle factoren van een jaar is 1. Om te voorspellen hoe het verbruik verdeeld is over een jaar, vermenigvuldigen we het jaarverbruik van het vorige jaar met de factor, zo weten we het verbruik voor die dag. SLP's worden enkel gemaakt voor elektriciteit en gas.



Figuur 24 BPMN-model integratie SLP's

De eerste stap voor het integreren van de SLP's is het downloaden ervan via de website van de VREG. Op de site van de VREG zijn verschillende wijzigingen te vinden, zoals het niet langer opstellen van SLP's voor gas en het groeperen van verschillende verbruikerstypes (VREG, 2025). Deze veranderingen maken het lastig om een tabel te maken met één kolom per energietype, wat de factor is die we gebruiken voor de voorspellingen. Daarnaast zijn er twee jaren waarvoor corrupte Excel-bestanden op de website staan. Daarom heb ik logica moeten schrijven om handmatig een corrupte file in te lezen. Na het uitvoeren van de eerste notebook hebben we alle Excel-bestanden met SLP's gedownload in onze Microsoft Fabric Lakehouse. Met behulp van BeautifulSoup filteren we de relevante download-URL's (Beautifulsoup, n.d.).

Met de tweede en derde notebooks converteren we de gedownloade bestanden in de formaten XLSX, XLS en XLSB naar CSV. Vervolgens aggregeren we de CSV-bestanden tot één tabel. We schonen alle verschillende datatypes op, waarbij vooral de datums en 'datetimes' in verschillende formaten een uitdaging vormden. Vervolgens kopiëren we de data naar zilver en goud, in zilver specifiëren we datatypes. De volgende notebook, "mimic SLP", gaat SLP's namaken voor alle andere energietypes dan elektriciteit en gas. Voor elke dag van verbruik in onze historiek geven we een percentage van het jaarverbruik, genormaliseerd naar 1.

Deze pipeline draait één keer per twee maanden, aangezien de SLP's slechts één keer per jaar worden bijgewerkt. Het kan echter voorkomen dat de SLP's van het huidige of zelfs het vorige jaar worden aangepast. Daarom downloaden we alle SLP's opnieuw, voor het geval dat. Hieronder vinden we axioma's van de SLP's.

$$\sum_{alle\ dagen} Factor = 1 \qquad Factor = \frac{Verbruikdag}{Totaal\ jaarverbruik}$$

Een belangrijke kanttekening is dat niet alle SLP's mogen gebruikt worden voor eender welk kantoor. Het juiste SLP wordt bepaald aan de hand van het type aansluiting (VREG, n.d.):

- **S21** is het SLP voor huishoudelijke verbruikers met een nacht-dagverhouding < 1,3 en wordt ook toegekend aan alle mensen met een enkelvoudige meter omdat de dag-nachtverhouding hier niet nader te bepalen is.
- **S22** is het SLP van een huishoudelijke verbruiker met een nacht-dagverhouding >= 1,3. Mensen met accumulatieverwarming (met exclusief nachtmeter) worden doorgaans binnen dit profiel genomen.
- **S11** is het SLP voor niet-huishoudelijke afnemers met een aansluitingsvermogen < 56 kVA.
- **S12** is het SLP voor niet-huishoudelijke afnemers met een aansluitingsvermogen > 56 kVA.
- **S18** is het SLP voor vlakke verbruikers.
- **S19** is het SLP voor openbare verlichting.

Uit bovenstaande voorbeelden afkomstig van de website van de VREG kunnen we afleiden dat de volgende SLP's gebruikt moeten worden voor de volgende kantoren:

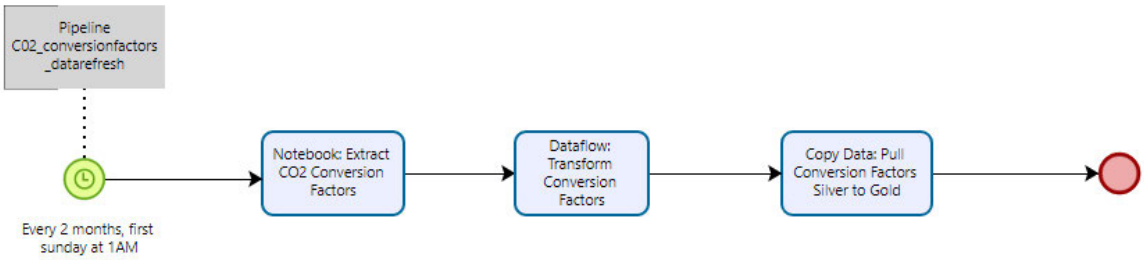
Kantoor	SLP Gas	SLP Elektriciteit
Gateway		
Gent		
Oostkamp		
Kortrijk		
Leuven		
Hasselt		
Antwerpen		
Liège		
Roeselare		
Charleroi		
Boitsfort		

Figuur 25 Tabel van de kantoren en corresponderende SLP

Merk op dat Charleroi als enige kantoor S11 heeft als SLP voor elektriciteit. Dit komt omdat het een aansluitingsvermogen heeft dat kleiner is dan 56 kVA.

4.1.10. CO2-conversiefactoren

De extractie van CO2-conversiefactoren gebeurt op dezelfde manier als de synthetische lastprofielen. Het grootste verschil is dat we geen historiek hoeven bij te houden. Om de twee maanden surfen we naar de URL die de conversiefactoren bevat. De URL bevat het jaar als parameter, elke twee maanden bezoeken we dus de URL met het volgende jaar als parameter. Als deze URL ons HTML teruggeeft, betekent dit dat de nieuwe conversiefactoren zijn gepubliceerd. Deze integreren we dan, analoog aan de SLP's, in Microsoft Fabric (Department for Energy Security and Net Zero, 2024).
Vervolgens gaan we deze conversiefactoren opschonen in de zilveren laag. Sommige conversiefactoren zijn gebaseerd op kWh, terwijl ik mijn consumptiedata in MWh bijhoud. Dit pas ik aan met behulp van de dataflow. Als laatste stap kopieer ik de data van zilver naar goud met een 'copy data'-activiteit.



Figuur 26 BPMN-model integratie CO2-conversiefactoren

De berekening van de uitstoot is redelijk eenvoudig. We filteren de CO2-conversiefactorentabel totdat we alleen de energietypes en factoren overhouden die relevant zijn voor ons. Vervolgens voeg ik de tabel samen met de energieconsumptietabellen via een 'LEFT JOIN'. Hierna vermenigvuldig ik de consumptie met de conversiefactor en krijg ik als resultaat de uitstoot in kilogram.

$$uitstoot = consumptie * conversiefactor$$

4.2. Data Science

In dit deel licht ik toe hoe ik de consumptie van alle energietypes voor alle kantoren heb voorspeld. De klant moet budgetten opstellen en kunnen inschatten hoeveel het verbruik zal zijn in het volgende fiscale jaar. Verder toon ik hoe ik uitschieters heb gecorrigeerd.

4.2.1. Voorspelling van consumptie

In de voorgaande paragrafen wordt beschreven hoe we SLP's gaan downloaden of berekenen op basis van historische gegevens. Op figuur 24 zien we dat nadat de SLP's zijn gedownload of berekend, we de voorspellingen kunnen berekenen in notebook "predict SLP energy consumption". Deze notebook bevat bijna duizend regels, dus zal ik de belangrijkste delen toelichten.

Figuur 27 flowchart van logica notebook voorspellen consumptie

Als eerste stap in onze notebook verzamelen we de jaarlijkse verbruiksgegevens voor alle jaren en energietypes. Op figuur 28 zien we hoe ik dit gedaan heb voor gas en elektriciteit.

```
1   # yearly consumption [redacted]
2   [redacted] = spark.sql("""
3       SELECT d.year, e.energy, o.office_name, SUM(f.[redacted] AS sum_consumption
4       FROM fct_FAC_Energy f
5       LEFT JOIN dim_FAC_Date d
6       ON f.dateID = d.dateID
7       LEFT JOIN dim_FAC_EnergyType e
8       ON f.energytypeID = e.energytypeID
9       LEFT JOIN dim_FAC_Office o
10      ON f.officeID = o.officeID
11      -- is_prediction needs to be false ofcourse, we calculate the predictions on the actual consumption data
12      WHERE f.engie_consumption IS NOT NULL AND f.is_prediction IS FALSE
13      GROUP BY d.year, e.energy, o.office_name
14      ORDER BY d.year DESC
15  """)
16
17  display[redacted]
```

Figuur 28 Spark-SQL code om jaarlijkse [redacted] te berekenen

Nu we onze jaarlijkse verbruiken kennen, kunnen we voorspellen hoeveel de consumptie in de volgende fiscale jaren zal bedragen. Door het axioma dat de som van alle dagelijkse factoren altijd 1 bedraagt, betekent dit dat de som van alle verbruiken in de volgende fiscale jaren altijd gelijk zal zijn aan het huidige jaarlijkse verbruik. Met die reden introduceer ik een richtingscoëfficiënt en een intercept met doel de SLP's te vermenigvuldigen met een corrigerende factor, zodat de voorspelde waarden rekening houden met een stijgende of dalende trend.

$$m = \frac{n\sum_{i=1}^{n}(x_i y_i) - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} y_i x_i^2 - (\sum_{i=1}^{n} x_i)^2} \qquad b = \frac{\sum_{i=1}^{n} y_i - m\sum_{i=1}^{n} x_i}{n}$$

De bovenstaande formules tonen hoe ik de richtingscoëfficiënt (m) en het intercept (b) bereken. Dit doe ik per energietype en kantoor, aangezien trends sterk kunnen verschillen per kantoor. De richtingscoëfficiënt bereken ik niet aan de hand van het verbruik van de afgelopen drie jaar. De totale verbruiksgegevens van deze jaren tellen als mijn punten.

Vervolgens bereken ik de correctiefactoren voor het huidige jaar en de volgende jaren op basis van een dataframe met energieverbruik voor een specifiek energietype en kantoor over drie jaar, en de richtingscoëfficiënt (m) en het intercept (b) van de trend. Eerst wordt het verbruik van vorig jaar opgehaald. Vervolgens wordt het voorspelde verbruik voor het huidige jaar, het volgende jaar en het jaar daarna berekend met de formule (y = mx + b). De correctiefactoren worden berekend als de verhouding van het voorspelde verbruik van het huidige jaar ten opzichte van vorig jaar, het volgende jaar ten opzichte van het huidige jaar, en het jaar daarna ten opzichte van het volgende jaar. De functie geeft deze correctiefactoren terug. Op figuur 29 zien we wat ik heb beschreven in code.

```python
def calculate_slope_and_intercept(df):
    """
    function that calculates the slope and intercept of the energy consumption trend.
    input: takes a dataframe of 1 distinct energytype and 1 distinct office with the sums of consumption throughout 3 years.
    output: m (slope), b (intercept)
    """
    consumption_df = df.toPandas()

    # Extract x and y values
    x = consumption_df["year"]
    y = consumption_df["sum_consumption"]

    # Calculate sums
    n = len(x)
    sum_x = x.sum()
    sum_y = y.sum()
    sum_xy = (x * y).sum()
    sum_x2 = (x ** 2).sum()

    # Calculate slope (m)
    m = (n * sum_xy - sum_x * sum_y) / (n * sum_x2 - sum_x ** 2)

    # Calculate intercept (b)
    b = (sum_y - m * sum_x) / n

    return m, b
```

```python
def calculate_trend_factors(df, m, b):
    """
    function that calculates the factors of correction for the current year and the next years.
    input: takes a dataframe of 1 distinct energytype and 1 distinct office with the sums of consumption throughout 3 years.
           m (slope of the trend)
           b (intercept of the trend)
    output: factors of correction for the current year and the next years
    """
    y_last_year = df.where(col("year") == (this_year - 1)).select("sum_consumption").collect()[0][0]
    y_this_year = m * this_year + b
    y_next_year = m * next_year + b
    y_next2_year = m * (next_year + 1) + b

    factor_current_year = y_this_year / y_last_year
    factor_next_year = y_next_year / y_this_year
    factor_next2_year = y_next2_year / y_next_year

    return factor_current_year, factor_next_year, factor_next2_year
```

Figuur 29 belangrijkste functies berekening rico, intercept en trend factoren

Op figuur 29 is te zien wat het resultaat is van onze berekeningen. We hebben een dataframe met per kantoor, jaar en energietype de som van het verbruik, de trendfactor en de voorspelde totale jaarconsumpties voor volgende jaren.

Figuur 30: Tabel met trendfactoren

4.2.2. Correctie van uitschieters

De volgende stap is het vermenigvuldigen van de voorspelde jaarconsumpties met de SLP-profielen, zodat we per dag een voorspelling van het verbruik verkrijgen. Vervolgens corrigeer ik eventuele uitschieters. Een uitschieter definieer ik als een waarde die hoger is dan de gemiddelde dagelijkse voorspelling plus drie keer de standaarddeviatie. Deze drempelwaarde heb ik zelf bepaald, de uitschieters zijn vaak 100 keer groter dan het gemiddelde en zijn dus makkelijk te identificeren. De drempelwaarde wordt als volgt berekend:

$$Drempelwaarde = \mu + 3\sigma$$

Waarbij μ de gemiddelde dagelijkse voorspelling is en σ de standaarddeviatie van de dagelijkse voorspellingen is. Waarden die deze drempel overschrijden, worden vervangen door μ. Tot slot tel ik het aantal gecorrigeerde voorspellingen. De standaarddeviatie berekenen we zo:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

De standaarddeviatie (σ) berekenen we door het gemiddelde verschil te nemen tussen de dagelijkse voorspellingen en hun gemiddelde, waarbij we eerst elk verschil kwadrateren, de kwadraten optellen, en daar vervolgens de wortel van nemen. Op deze manier kan ik zelf bepalen wat uitschieters zijn en vanaf wanneer ze gecorrigeerd moeten worden. Zo verwijder ik ongewenste uitschieters zonder al te veel invloed te hebben op mijn berekening. Dit zorgt ervoor dat de voorspellingen betrouwbaarder zijn en beter aansluiten bij de werkelijke consumptie.

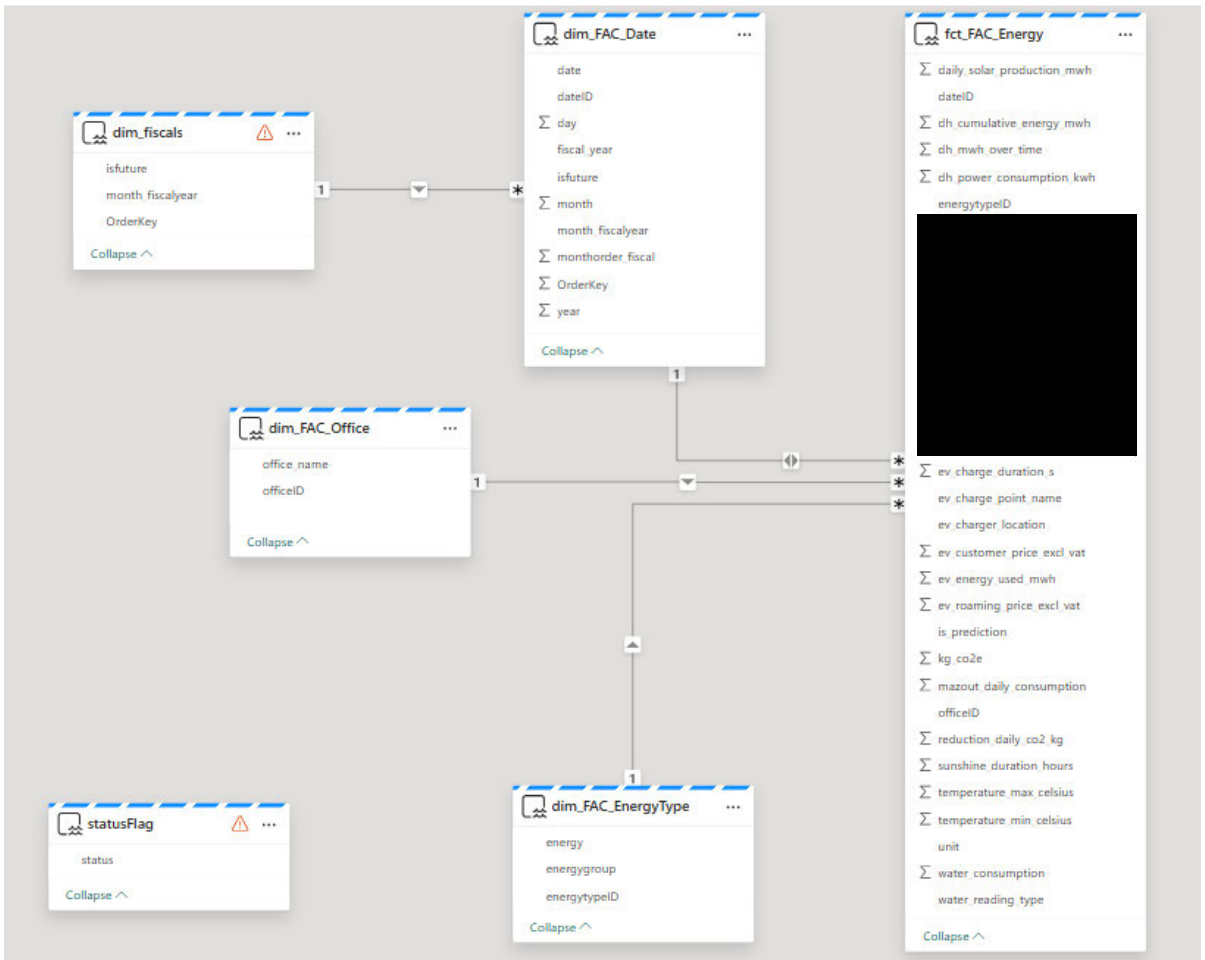
4.3. Data Visualisatie

In dit deel toon ik mijn eindproduct: het dashboard. Dit dashboard biedt een overzicht van de historische en voorspelde consumptie van alle energietypes voor alle kantoren. Het stelt de klant in staat om budgetten op te stellen en nauwkeurig in te schatten hoeveel het verbruik zal zijn in het volgende fiscale jaar.

4.3.1. Dimensioneren van mijn Dataset

Om mijn data efficiënt te visualiseren, giet ik mijn dataset in een sterschema. Hiervoor moet ik mijn data opdelen in één feitentabel en meerdere dimensietabellen. Alle numerieke data gaan in de feitentabel en alles wat de numerieke data beschrijft hoort in dimensietabellen (Microsoft, 2024). In mijn model heb ik een dimensietabel voor tijd, waarin ik een kalender maak met dag, jaar, fiscaal jaar, enzovoort. Hieraan hangt nog een dimensietabel met een nummering voor de maanden en fiscale jaren, aangezien dit niet automatisch te sorteren is in Power BI. Daarom kan dit datamodel ook een Snowflake-model genoemd worden.

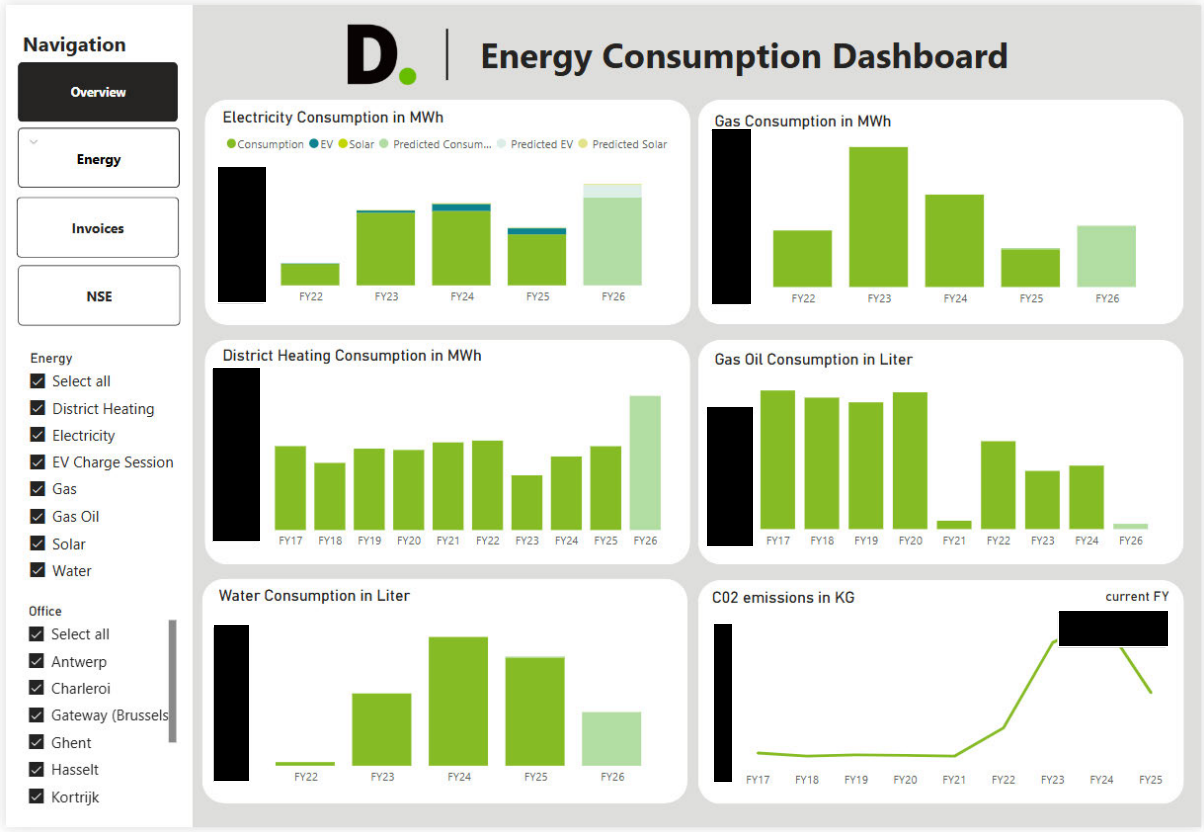
Verder heb ik nog een dimensietabel voor de kantoornamen en energietypes. De 'statusFlag'-tabel bevat een boolean die gebruikt wordt voor het voorselecteren van bijvoorbeeld een energietype in mijn visualisaties. De rode gevarendriehoeken op figuur 30 zijn er omdat het niet gebruikelijk is om views te gebruiken in plaats van tabellen in een datamodel. Aangezien deze tabellen niet veel data bevatten, vormt dit echter geen probleem.



Figuur 31 Datamodel Energieconsumptie Dashboard

4.3.2. Pagina Overzicht

Op figuur 32 zien we de overzichtspagina van het Energy Consumption Dashboard. Linksboven bevindt zich een navigatiebalk met een dropdownmenu voor alle energietypes, zoals te zien op figuur 34. Hieronder zijn er filters per energietype en kantoor. Het doel van deze pagina is om een overzicht te bieden van alle kantoren, energietypes, voorspellingen en CO2-uitstoot. Met behulp van de filters kunnen we de CO2-uitstoot per kantoor weergeven. Op figuur 33 zien we hoe de overzichtspagina er uitziet met alleen data van het Gentse kantoor.



Figuur 32 Pagina Overzicht

Figuur 34 Dropdown navigatie

Figuur 33 Pagina overzicht gefilterd op Gent

4.3.3. Energiepagina's

Als eerste energiepagina hebben we elektriciteit, wat veruit het grootste energietype is. Op deze pagina tonen we de data van de energieleverancier [redacted] zonnepaneelproductiegegevens van het [redacted] en de oplaadbeurten van elektrische wagens afkomstig van het [redacted].

Bovenaan zien we een aantal Key Performance Indicators (KPI's):

- Het totale energieverbruik exclusief oplaadbeurten in MWh
- Het totale energieverbruik van de oplaadbeurten in MWh
- De totale energieproductie van de zonnepanelen in MWh
- Hoeveel CO2 in kg we uitsparen door de energieproductie van zonnepanelen

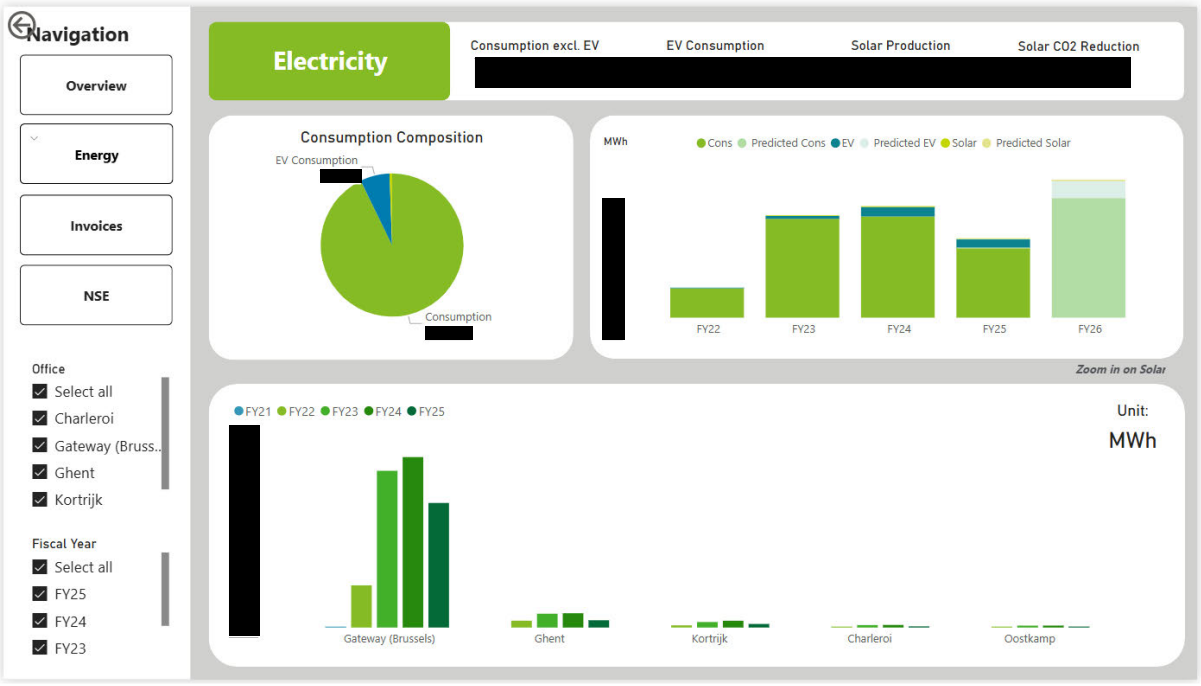
KPI's maak ik aan de hand van measures in Power BI, wat zorgt voor dynamische en interactieve visualisaties die automatisch worden bijgewerkt wanneer de onderliggende data veranderen. Measures worden geschreven aan de hand van de DAX-programmeertaal.

Dit is een voorbeeld van de measure achter de eerste KPI, totale energieverbruik:

```
kpi_total_electricity_consumption =
var _calc = SUM(fct_FAC_Energy[redacted])
RETURN COALESCE(_calc, 0)
```

De COALESCE-expressie zorgt ervoor dat onze eerste waarde, de verbruiksgegevens, wordt teruggegeven. Als er geen cijfers beschikbaar zijn, wordt een 0 teruggegeven. Dit voorkomt dat er een lege waarde (blank) wordt weergegeven, wat de visualisatie minder professioneel zou maken.

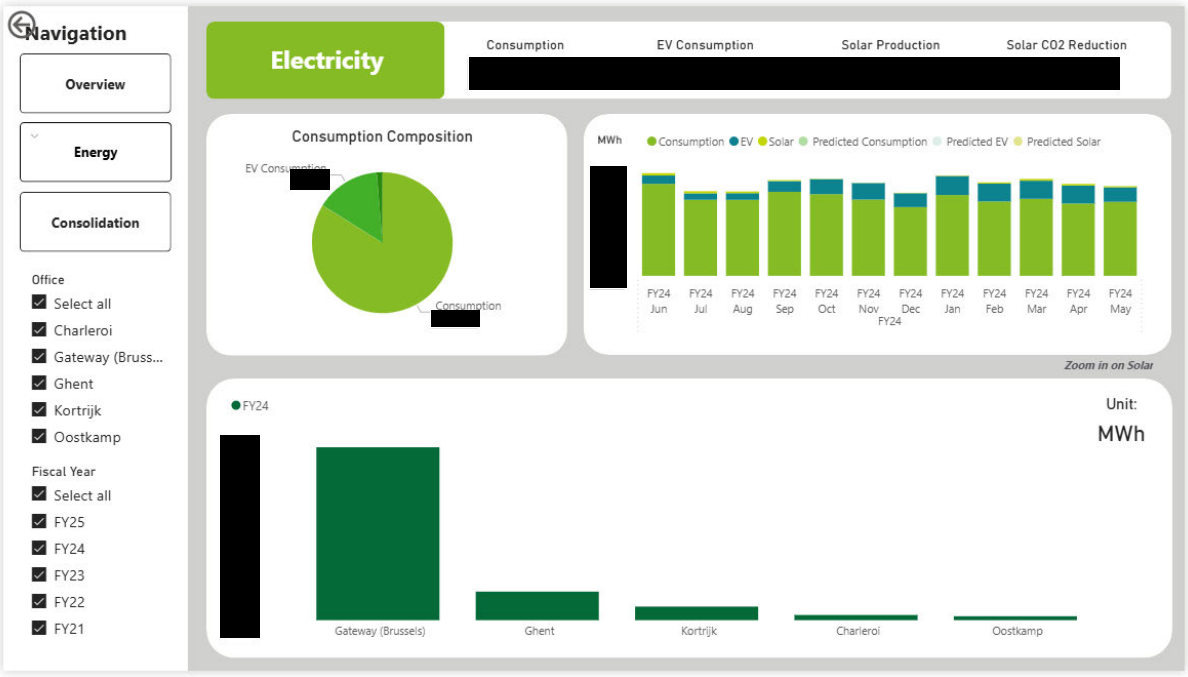
Hieronder zien we de samenstelling van het elektriciteitsverbruik in een taartdiagram. Daarnaast zien we in een gestapelde staafdiagram het verbruik en voorspelde verbruik per fiscaal jaar. Merk op dat het totale energieverbruik stijgt, wat te wijten is aan de oplaadbeurten van elektrische wagens. Let ook op dat voor FY25 een deel van onze data ontbreekt vanwege het niet kunnen automatiseren van de integratie.



Figuur 35 Pagina Elektriciteit

Onderaan zien we een gestapeld staafdiagram, waar het verbruik wordt vergeleken per kantoor. Verder heb ik deze pagina interactief gemaakt door het mogelijk te maken om een drill-down te doen op een fiscaal jaar. Hierdoor kunnen we nog gedetailleerder inzicht krijgen in de verbruiksgegevens.

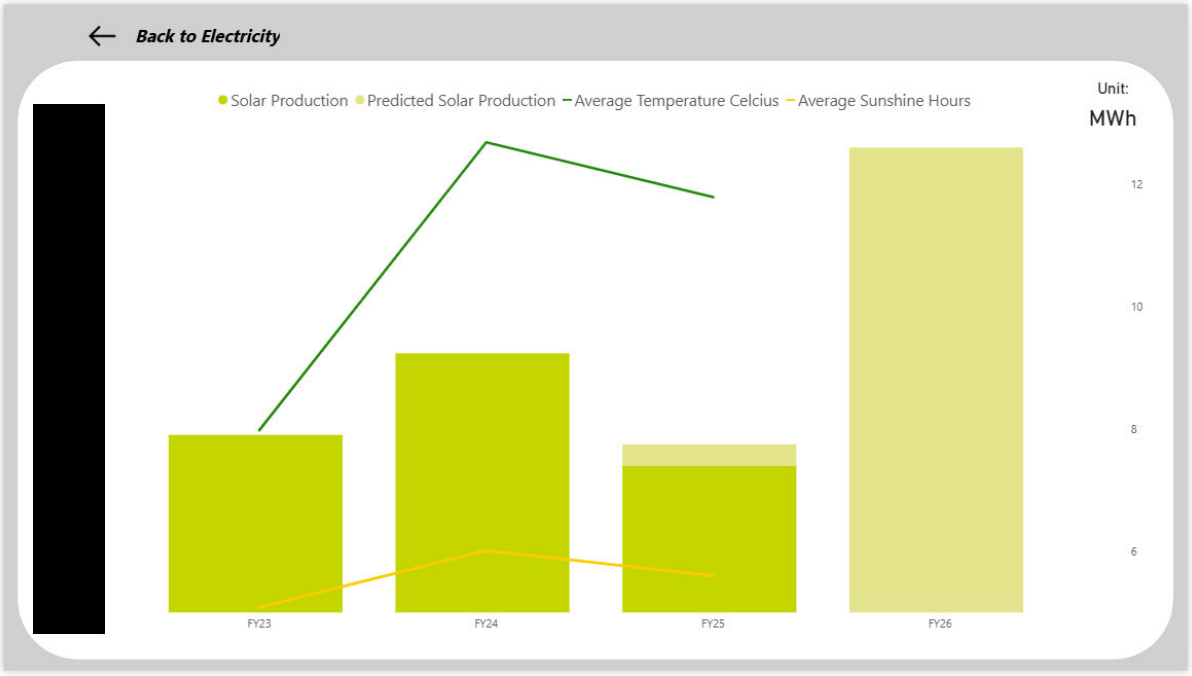
Op figuur 36 zien we een drill-down op fiscaal jaar 24. Merk op dat alle KPI's en grafieken veranderen en gegevens worden weergeven specifiek voor dat fiscale jaar.



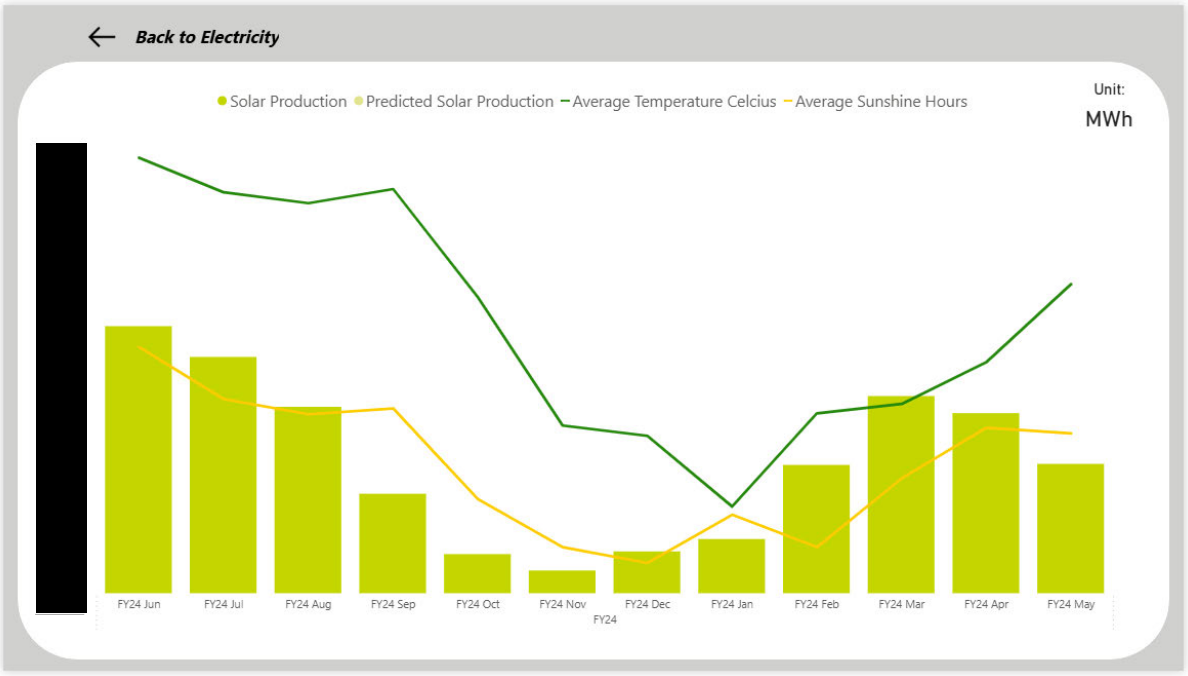
Figuur 36 Drill-down elektriciteitspagina

Omdat zonne-energie nauwelijks zichtbaar is in de visualisaties, heb ik onder het gestapeld staafdiagram een knop toegevoegd met de tekst "Zoom in on Solar". Wanneer we op deze knop klikken, wordt er een gestapeld staafdiagram getoond met alleen de productiegegevens van zonnepanelen te zien op figuur 36. Dit doen we omdat we het facility-team willen uitdagen om de productie van zonne-energie te verhogen.

Naast de productie van zonne-energie en voorspelde zonne-energie tonen we ook de gemiddelde temperatuur en zonne-uren. Op figuur 38 kunnen we een duidelijke correlatie zien tussen alle weergegeven waarden.

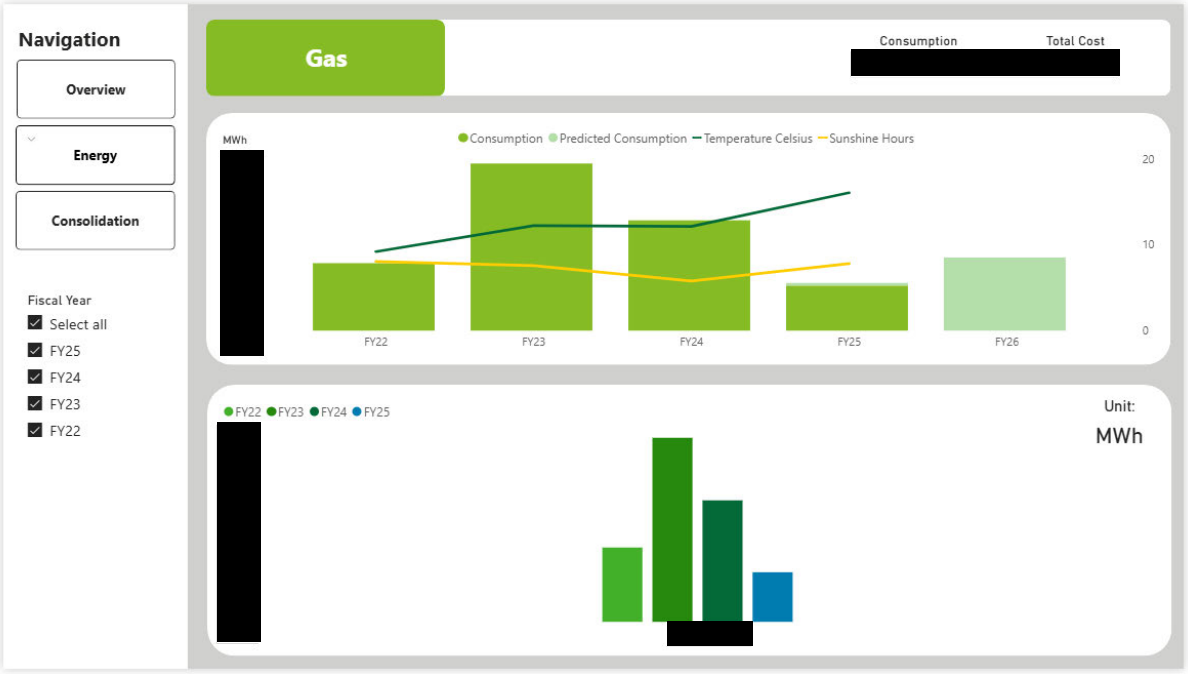


Figuur 37 Zoom op zonne-energie

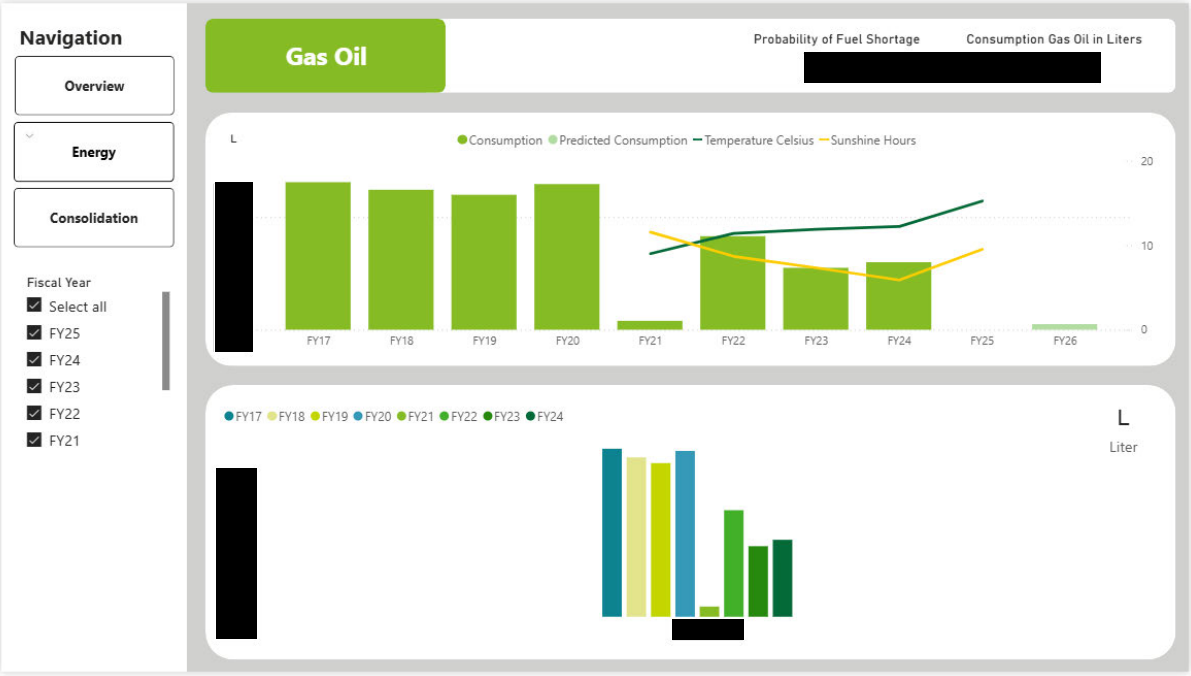


Figuur 38 Drill-down FY24 zonne-energie

De pagina's voor energietypes gas, mazout, districtverwarming en water volgen dezelfde lay-out als die van elektriciteit.



Figuur 39 Pagina gas



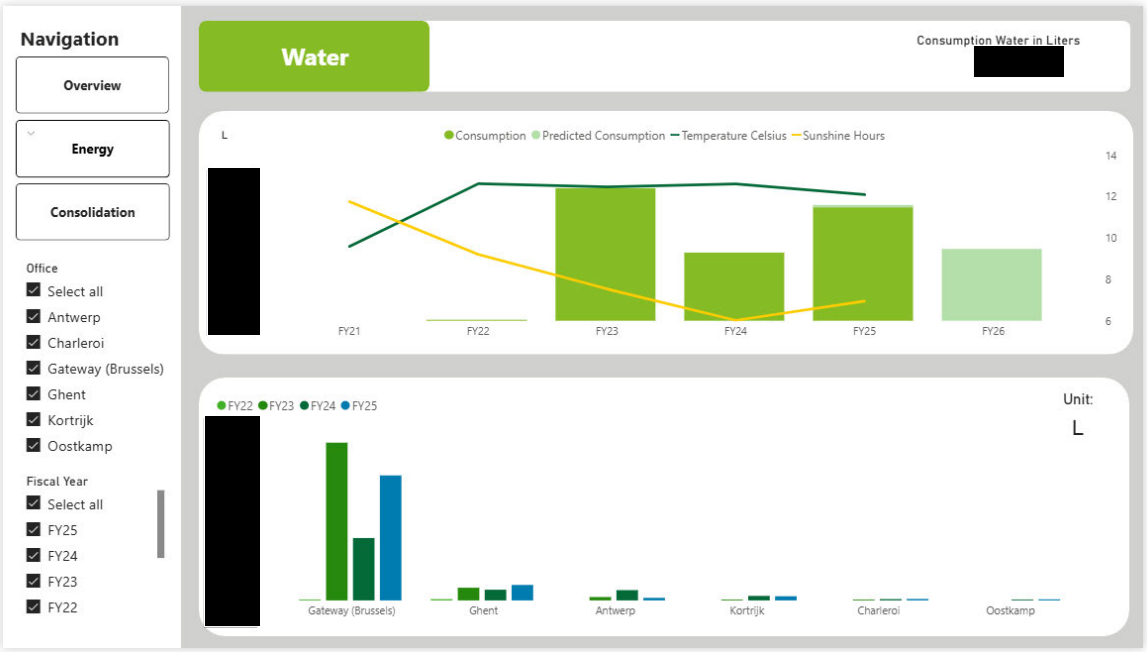
Figuur 40 Pagina Mazout

Op figuur 40 zien we een opmerkelijke KPI, namelijk het risico op een brandstoftekort. Aangezien de verbruiksvoorspellingen allemaal al gebeurd zijn, is het eenvoudig om deze measure te schrijven.

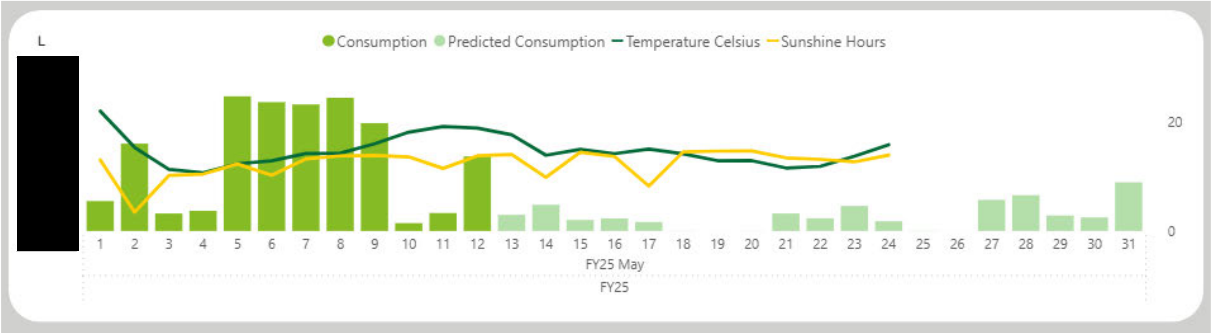
In onderstaande measure berekenen we of de effectieve consumptie hoger ligt dan de voorspelde consumptie. Als dit het geval is dan is de kans klein dat er een energietekort is en geven we ‘Low’ terug. Als de voorspelde consumptie hoger is dan de werkelijke consumptie dan geven we ‘High’ terug.

```
mazout_fuel_shortage = IF(
    [Sum Consumption Mazout] < CALCULATE(
        SUM(fct_FAC_Energy[engie_futureconsumption]),
        dim_FAC_EnergyType[energy] = "Mazout"
    ),
    "High",
    "Low"
)
```

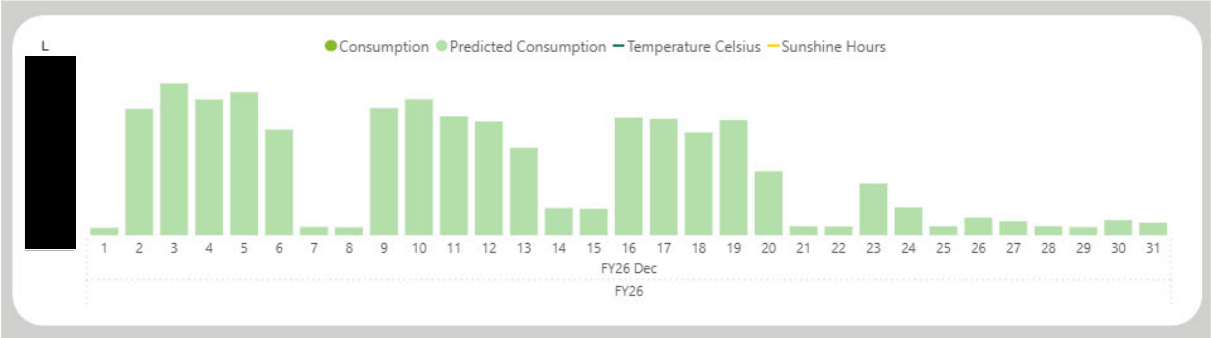
Figuur 41 Pagina districtverwarming



Figuur 42 Pagina water



Figuur 43 Drill-down water huidige datum



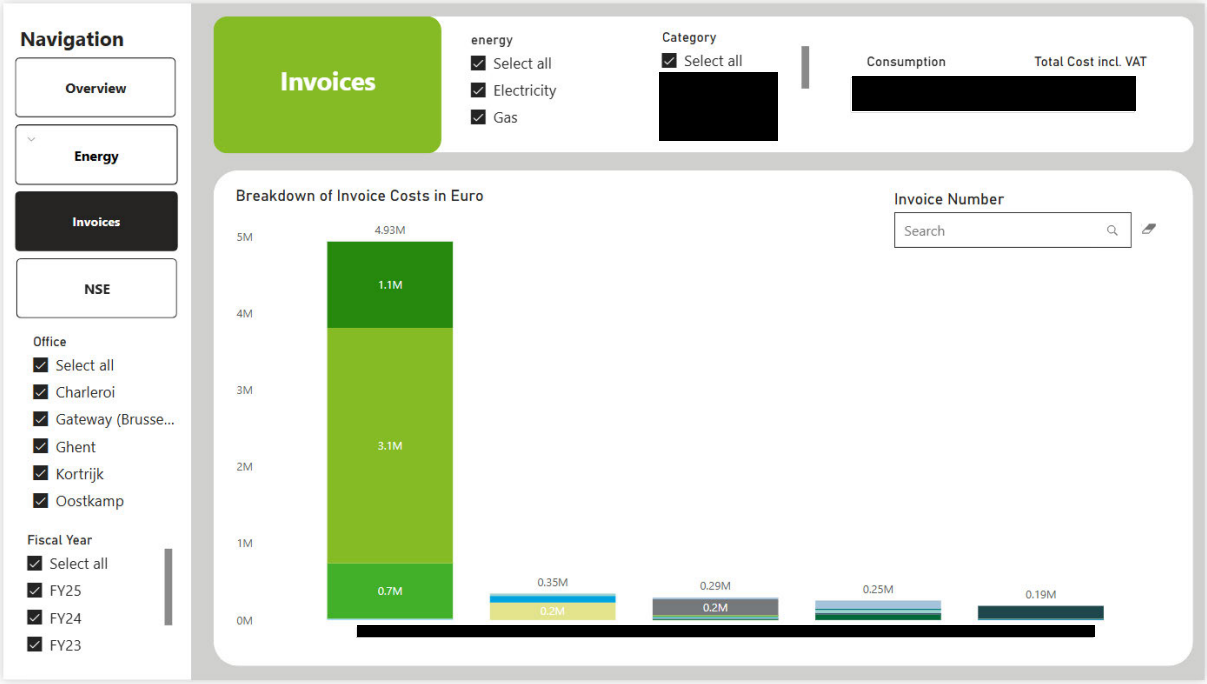
Figuur 44 Drill-down water december FY26

Op figuur 43 zien we een drill-down op dagelijks niveau voor het staafdiagram van de waterpagina. Bij het schrijven van dit document is het 13 mei. Merk op dat de weervoorspellingen tot en met 24 mei gaan, twee weken vooruit.

Op figuur 44 zien we een drill-down naar de toekomst, specifiek naar december in fiscaal jaar 26. Merk op dat mijn lineaire regressieanalyse voorspelt dat er veel minder water zal worden verbruikt in de periode tussen Kerst en Nieuwjaar.

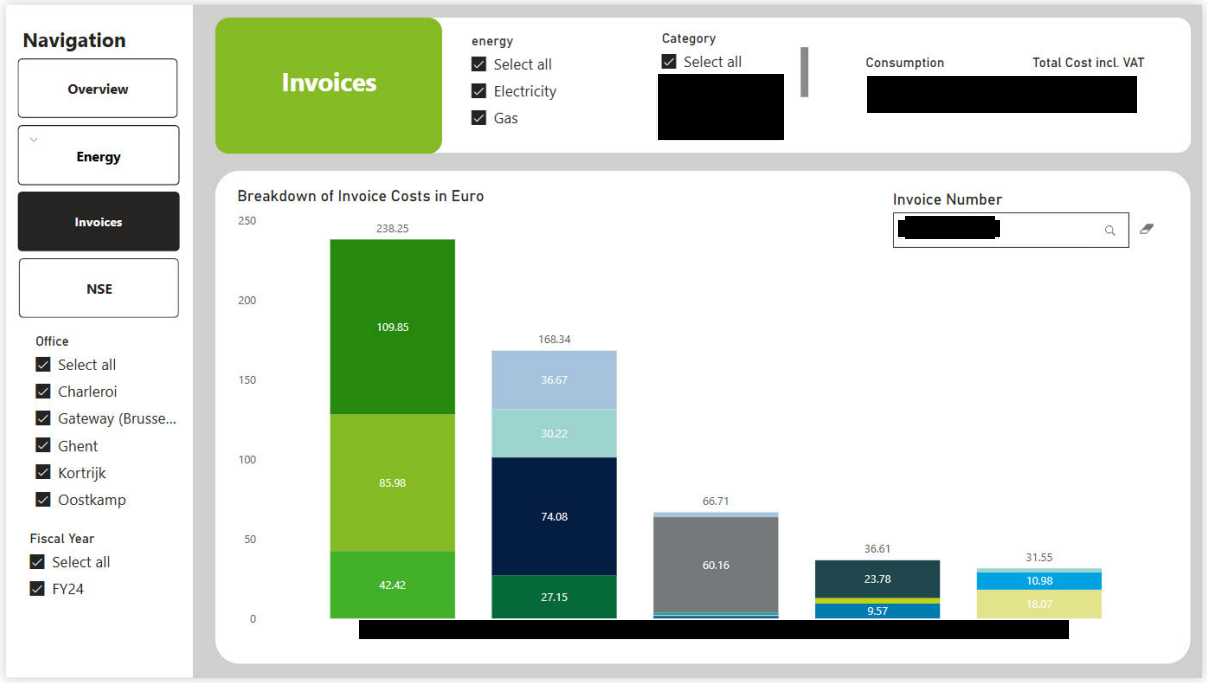
4.3.4. Pagina Facturen

Op figuur 45 zien we een breakdown van alle facturen [redacted], voor elektriciteit en gas. We kunnen filteren per kantoor, fiscaal jaar, energietype en categorie. Deze categorieën zijn rechtstreeks overgenomen van de facturen. We zien hiernaast ook de totale energieconsumptie en kost.



Figuur 45 Pagina facturen

Bij het ingeven van een factuurnummer zien we de grafieken en KPI's veranderen. Deze factuur is van [redacted] van FY24 en heeft een kostprijs van [redacted] aan elektriciteit.



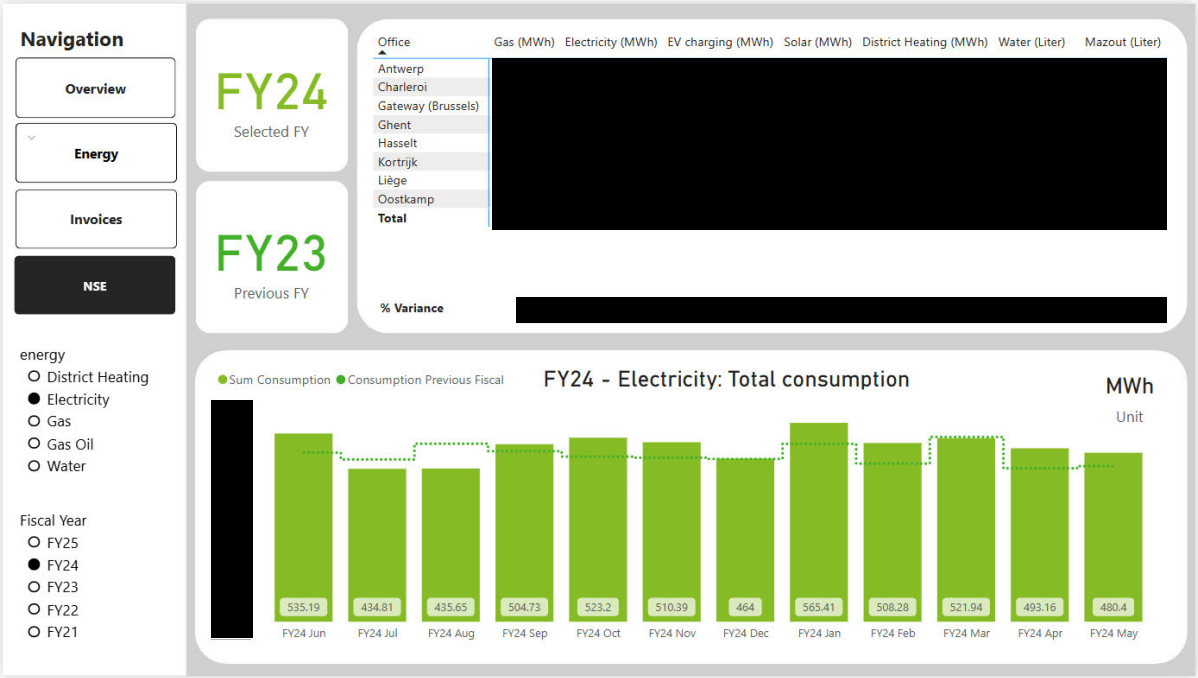
Figuur 46 Pagina facturen met een filter op factuurnummer

4.3.5. Pagina NSE

Op figuur 47 zien we de NSE-pagina, waar alle gegevens worden samengebracht. Het Facility-team rapporteert jaarlijks alle consumptiegegevens aan NSE (Deloitte North South Europe). Met behulp van deze pagina kan het Facility-team de gegevens opvragen, exporteren en rapporteren.

Aan de hand van de radio-buttons links kunnen gebruikers selecteren welk fiscaal jaar en welk energietype ze willen weergeven. In de tabel wordt de procentuele variatie weergegeven in vergelijking met het voorgaande fiscale jaar. Deze variatie heeft "conditional formatting", wat betekent dat de cellen rood of groen kleuren afhankelijk van de richting van de variatie. Merk op dat meer productie van zonne-energie als goed wordt beschouwd. Zowel de eenheid als de geselecteerde fiscale jaren veranderen afhankelijk van de selectie van de gebruiker.

Belangrijke kanttekening bij de consolidatiepagina is dat er voor elektriciteit alleen data [redacted] facturen worden gevisualiseerd. Er zijn ook data van [redacted] beschikbaar, maar omdat de sensoren die deze data verzamelen na de officiële meter staan, zit er verlies op. De NSE-sheet dient voor rapportage en hoort dus alleen exacte data te visualiseren. Vandaar dat de [redacted] lezingen niet gebruikt worden voor elektriciteit.



Figuur 47 Pagina consolidatie

5. Conclusie

Tijdens mijn stage bij Deloitte Belgium heb ik mijn statistische kennis, werkervaring en technische vaardigheden succesvol kunnen combineren om een waardevol eindresultaat te bekomen.

Het Energy Consumption Dashboard biedt een gedetailleerd overzicht van het historische en voorspelde energieverbruik voor alle kantoren in België. Dit stelt het Facility-team in staat om beter geïnformeerde beslissingen te nemen en nauwkeurige budgetten op te stellen. Voor de ontwikkeling van dit dashboard heb ik de integratie van alle verbruiksdata geautomatiseerd, evenals de verwerking van CO_2-conversiefactoren, synthetische lastprofielen en weerdata. Deze gegevens zijn bovendien ook waardevol voor andere toepassingen binnen het Data & Analytics-team.

Deze stage heeft mij niet alleen geholpen mijn technische vaardigheden verder te ontwikkelen, maar ook mijn vermogen versterkt om complexe projecten te plannen en tot een goed einde te brengen. Het resultaat is een gebruiksvriendelijk dashboard dat het Facility-team ondersteunt bij het nemen van datagedreven beslissingen en bijdraagt aan de duurzaamheidsdoelstellingen van Deloitte Belgium.

Tot slot beschouw ik de ervaring bij een groot consultancybedrijf als Deloitte als een enorme meerwaarde. De dertien weken waarin ik actief heb meegewerkt binnen de wereld van Data & Analytics hebben me ervan overtuigd dat dit de richting is waarin ik mijn carrière verder wil uitbouwen.

6. Nawoord

IT is een brede en voortdurend evoluerende sector waarin er altijd wel iemand is die ergens beter in is dan jij. Daarom is het belangrijk positieve ervaringen op te doen, zodat je zelfverzekerder wordt in je kunnen. Deze stage is daar een goed voorbeeld van. Drie jaar geleden schreef ik tijdens een Python-les mijn eerste programma "Hello World". Ik had nooit gedacht dat ik drie jaar later een dashboard zou maken voor één van de grootste consultancybedrijven ter wereld.

Mijn vorige opleiding was zeer academisch, en ik merk nu dat ik veel meer voldoening haal uit praktijkgerichte projecten. Ik ben ontzettend blij met mijn keuze voor de Thomas More hogeschool en de afstudeerrichting Digital Innovation. Met een duidelijk doel, mezelf specialiseren in de wereld van data, ben ik in 2022 begonnen aan mijn opleiding. In het tweede jaar heb ik gekozen voor de afstudeerrichting Digital Innovation, waar ik zelf mijn eigen accenten kon leggen in wat ik wilde leren. Zo kreeg ik de kans om te werken met Microsoft Fabric, iets wat niet in het traditionele curriculum voorkwam. Al doende merkte ik dat door deze eigen accenten te leggen en mezelf te onderscheiden, er meer interesse ontstond in mijn profiel. Kort na dit project koos ik ervoor om het certificaat voor Fabric te halen.

Deloitte is een bedrijf met veel boeiende, ambitieuze en professionele medewerkers werken. Ik was direct onder de indruk van de hands-on manier van werken en de toegankelijkheid van de managers.

Ik wil graag Yorick Stevens en Jonas Blom bedanken voor hun hulp bij het project en hun waardevolle professionele tips. Mijn dank gaat ook uit naar Julien Morret voor zijn rol als down-to-earth manager van het Data & Analytics team. Daarnaast wil ik mijn stagebegeleider Michiel Verboven bedanken voor het vertrouwen en de opvolging. En zeker ook Bram Heyns, Kathleen Renders en Jochen Mariën, bedankt voor jullie begeleiding de afgelopen twee jaar. Het was een luxe om zoveel één-op-één begeleiding te krijgen.

Met trots kan ik zeggen dat ik een aanbod heb gekregen van Delaware, een middelgroot consultancybedrijf dat al lang op mijn radar stond. Ik heb deze mooie kans aanvaard! Na mijn stage ben ik klaar om het leren op de schoolbanken achter mij te laten en het leren op de werkvloer aan te vatten.

7. Begrippenlijst

Begrip	Definitie
Access token	Een tijdelijk toegangsbewijs dat wordt gebruikt om beveiligde API's of systemen te benaderen namens een gebruiker of applicatie.
Agile	Een flexibele manier van werken waarbij softwareontwikkeling wordt opgedeeld in korte iteraties met regelmatige feedback en bijsturing.
API (Application Programming Interface)	Een set regels en definities waarmee softwaretoepassingen met elkaar kunnen communiceren.
Copy data activity	Een taak binnen gegevensintegratietools (zoals Microsoft Fabric) die gegevens kopieert van de ene locatie naar de andere.
Drill-down	Een functie in dashboards of rapportages waarmee je kunt inzoomen op meer gedetailleerde gegevens binnen een dataset.
ETL (Extract, Transform, Load)	Een proces in data-engineering waarbij gegevens worden opgehaald, getransformeerd en opgeslagen in een data warehouse.
Headless Selenium	Selenium is een populaire tool voor het automatiseren van webbrowser-interacties. Headless wil zeggen dat de browser geen grafische interface toont. Dit betekent dat de browser in de achtergrond draait zonder een visueel venster, wat efficiënter is voor het uitvoeren van tests en webscraping-taken. Het is vooral nuttig voor situaties waarin snelheid en resource-efficiëntie belangrijk zijn.
Intercept (snijpunt)	In een lineair model is dit het punt waar de regressielijn de y-as snijdt (bij x = 0).
JSON (JavaScript Object Notation)	Een lichtgewicht dataformaat voor gestructureerde gegevensuitwisseling, veel gebruikt bij API's.
Kanban-bord	Een visueel hulpmiddel om taken te beheren, meestal verdeeld in kolommen zoals 'Te doen', 'Bezig' en 'Gedaan'.
KPI (Key Performance Indicator)	Een meetbare waarde die aangeeft hoe effectief een individu, team of organisatie is in het bereiken van doelstellingen.
Lakehouse	Een moderne data-architectuur die de flexibiliteit van een data lake combineert met de structuur van een data warehouse.
Library	Een library (bibliotheek) in de context van softwareontwikkeling is een verzameling van voorgeschreven code die ontwikkelaars kunnen gebruiken om specifieke functionaliteiten toe te voegen aan hun applicaties zonder dat ze die code zelf hoeven te schrijven. Libraries bevatten functies, classes en methoden die herbruikbaar zijn en helpen bij het versnellen van de ontwikkelingsprocessen. Voorbeelden zijn de Python Standard Library, React.js voor JavaScript, en TensorFlow voor machine learning.
Lineaire regressieanalyse	Een statistische techniek om de relatie tussen een onafhankelijke en een afhankelijke variabele te modelleren.
Microsoft OneLake	Een centraal data-opslagsysteem binnen Microsoft Fabric, ontworpen als één uniforme datalaag voor analytics.

OLAP (Online Analytical Processing)	OLAP is een technologie die wordt gebruikt voor het snel analyseren van multidimensionale data uit een datawarehouse. OLAP-systemen zijn ontworpen om complexe queries en analyses uit te voeren, zoals trendanalyses, financiële rapportages, en andere vormen van data-analyse die grote hoeveelheden data vereisen. OLAP is essentieel voor business intelligence en helpt organisaties bij het nemen van geïnformeerde beslissingen door het bieden van snelle en interactieve toegang tot data. De tegenhanger is OLTP, wat geoptimaliseerd is voor webapplicaties, niet data-analyse.
Power BI Measure	Een Power BI Measure is een berekening die je maakt in Power BI, een business intelligence tool van Microsoft, om data te analyseren en visualiseren. Measures worden geschreven in DAX (Data Analysis Expressions) en kunnen dynamische berekeningen uitvoeren op basis van de data in je model. Ze worden vaak gebruikt om KPI's (Key Performance Indicators) te berekenen, zoals omzetgroei, gemiddelde waarden, of percentages, en kunnen worden gebruikt in rapporten en dashboards.
Refresh token	Een token waarmee je een nieuw access token kunt verkrijgen zonder opnieuw in te loggen, vaak gebruikt bij OAuth-authenticatie.
Richtingscoëfficiënt	De helling van een lijn in een grafiek; geeft aan hoe sterk en in welke richting twee variabelen samenhangen.
Snowflake-model	Het Snowflake-model is een type datawarehouse-schema dat een uitbreiding is van het stermodel. In een Snowflake-schema worden dimensionele tabellen genormaliseerd, wat betekent dat ze worden opgesplitst in meerdere gerelateerde tabellen om redundantie te verminderen en data-integriteit te verbeteren. Dit resulteert in een meer complexe structuur met meerdere niveaus van tabellen, maar kan efficiënter zijn voor bepaalde analytische queries. Het wordt vaak gebruikt in situaties waar de dimensionele data zeer gedetailleerd en complex zijn.
Standaarddeviatie	Een maat voor de spreiding of variatie van een dataset; hoe groter de standaarddeviatie, hoe meer de waarden verspreid zijn.
Sterschema	Het sterschema is een type datawarehouse-schema dat wordt gebruikt voor het organiseren van data in een manier die analyse en rapportage vergemakkelijkt. Het schema bestaat uit een centrale fact-tabel die de belangrijkste meetgegevens bevat, zoals verkoopcijfers of transacties, en wordt omringd door dimensionele tabellen die contextuele informatie bieden, zoals tijd, locatie, product, of klantgegevens. Deze dimensionele tabellen zijn direct verbonden met de fact-tabel, waardoor het schema de vorm van een ster krijgt. Het sterschema is eenvoudig en efficiënt voor het uitvoeren van queries en wordt vaak gebruikt in OLAP (Online Analytical Processing) systemen.
User story	Een korte beschrijving van een wens of behoefte van een gebruiker, vaak gebruikt binnen Agile-methodologieën.
Wireframes	Schematische weergaven van een webpagina of app-structuur, bedoeld om de lay-out en gebruikerservaring te plannen.

Referentielijst

Beautifulsoup. (n.d.). *Beautiful Soup Documentation*. Retrieved from Beautifulsoup: https://beautiful-soup-4.readthedocs.io/en/latest/

Blanc, G. (n.d.). *Pyhfs*. Retrieved from Github: https://github.com/guillaumeblanc/pyhfs

Deloitte. (2024, 10 03). *Deloitte Belgium achieves growth of +4.4% in financial year 2024*. Retrieved from Deloitte: https://www.deloitte.com/be/en/about/press-room/annual-results-fy24.html

Deloitte. (2025). *Our heritage*. Retrieved from https://www.deloitte.com/global/en/about/story/purpose-values/our-history.html.

Deloitte. (2025, 02 13). *Set Variable Activity in Azure Data Factory and Azure Synapse Analytics*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/azure/data-factory/control-flow-set-variable-activity

Deloitte. (2025). *Who we are*. Retrieved from Deloitte: https://www.deloitte.com/

Department for Energy Security and Net Zero. (2024, 10 30). *Greenhouse gas reporting: conversion factors 2024*. Retrieved from GOV.UK: https://www.gov.uk/government/publications/greenhouse-gas-reporting-conversion-factors-2024

Geeksforgeeks. (2024, Aug 12). *Difference between Fact Table and Dimension Table*. Retrieved from Geeksforgeeks: https://www.geeksforgeeks.org/difference-between-fact-table-and-dimension-table/

Microsoft. (2024, 18 12). *Quickstart: Create your first dataflow to get and transform data*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/fabric/data-factory/create-first-dataflow-gen2

Microsoft. (2024, 12 30). *Understand star schema and the importance for Power BI*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/power-bi/guidance/star-schema

Microsoft. (2025, 04 26). *Microsoft Learn*. Retrieved from Direct Lake Overview: https://learn.microsoft.com/en-us/fabric/fundamentals/direct-lake-overview

Microsoft. (2025, 08 02). *Semantic model modes in the Power BI service*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/power-bi/connect-data/service-dataset-modes-understand

Microsoft. (2025, 02 05). *Wat is een lakehouse in Microsoft Fabric?* Opgehaald van Microsoft Learn: https://learn.microsoft.com/nl-nl/fabric/data-engineering/lakehouse-overview

Microsoft. (n.d.). *Microsoft Certified: Fabric Analytics Engineer Associate*. Retrieved from Microsoft Learn: https://learn.microsoft.com/en-us/users/nathanneve-4540/credentials/2c5fc5d79ed97822?ref=https%3A%2F%2Fwww.linkedin.com%2F

[redacted]

Open-Meteo. (n.d.). *Features*. Retrieved from Open-Meteo: https://open-meteo.com/en/features

[redacted]

Synergrid. (2025). *Profielen - SLP-SPP-RLP*. Retrieved from Synergrid: https://www.synergrid.be/nl/documentencentrum/statistieken-gegevens/profielen-slp-spp-rlp

VREG. (2025). *Verbruiksprofielen en productieprofielen*. Retrieved from VREG: https://www.vlaamsenutsregulator.be/nl/verbruiksprofielen-en-productieprofielen

VREG. (n.d.). *Verbruiksprofielen en productieprofielen*. Retrieved from VREG: https://www.vlaamsenutsregulator.be/nl/verbruiksprofielen-en-productieprofielen

GEBRUIK VAN AI

Dit document is nagelezen door PairD, de AI-chatbot van Deloitte die gebruikmaakt van het GPT4-model. De begrippenlijst is gegenereerd door deze chatbot. AI is niet gebruikt voor het opstellen van dit document, maar enkel ingezet voor het verbeteren van de zinsbouw in bepaalde paragrafen.

Afbeeldingen

CONTACT

Nathan Neve | Stagiair Deloitte
R0742822@student.thomasmore.be
Tel. + 32 489 89 39 11

VOLG ONS

www.thomasmore.be
fb.com/ThomasMoreBE
#WeAreMore

THOMAS MORE