00:00:00.000 --> 00:00:02.600
Anouk Wolters
Ja oké, Dat is dat.

00:00:05.530 --> 00:00:13.120
Anouk Wolters
Ja, zou je Misschien wat meer kunnen vertellen over wat je net zei waar je mee bezig bent op dit moment aan bij [de bank]?

00:00:14.780 --> 00:00:36.470
Data Engineer
Ja ja [de bank] heeft dus een [data science team] en dat team bouwt modellen machine modellen om te detecteren om eigenlijk misdaad te detecteren. Of nou ja, wat voor malifide praktijken dan ook binnen binnen de bank en hun klanten.

00:00:38.430 --> 00:01:08.480
Data Engineer
En om die modellen de productie te brengen, hebben ze ook een ander team, ML team, dus het transaction monitoring machine learning team, dat de modellen dus ja productie brengt. En dat betekent data pipelines bouwen en streamlinen en nou op een gegeven moment heeft dat ML team heeft bedacht van hé Omdat Iedereen net iets andere code schrijft voor datapipelines en voor de features die jullie gebruiken, willen we graag naar een centraal systeem gaan.

00:01:08.540 --> 00:01:10.950
Data Engineer
En hebben ze de features store bedacht, dus Dat is een.

00:01:12.480 --> 00:01:16.500
Data Engineer
store waarin eigenlijk alle features die worden gebruikt voor een model.

00:01:18.090 --> 00:01:34.420
Data Engineer
Independently kunnen worden beschreven en ook alle sources waar ze dan vanaf hangen dat die heel duidelijk zijn en dat een feature eigenlijk op die manier helemaal op zichzelf kan leven en dus los van het model, dus Dat is model agnostic in in die zin.

00:01:35.820 --> 00:02:05.450
Data Engineer
Nou die features store bestaat, dus de library bestaat, Maar daar moeten de datapipelines nog naar worden gemigreerd en en mij was gevraagd om binnen een pillar, een van de modellen daarvan te nemen en die data pipeline te migreren naar de feature store. Wat dat betekent, is dat ik eigenlijk een deel van de documentatie van het model lees om te begrijpen wat voor sources er worden gebruikt.

00:02:07.470 --> 00:02:19.710
Data Engineer

En vervolgens ben gaan kijken met het data Science, dus hoe we de beste manier uitwerken, beste kunnen aanpakken hoe veel features er zijn en dat soort dingen. En vervolgens is het gewoon de features implementeren.

00:02:21.090 --> 00:02:51.660
Data Engineer
En ja, Dat is eigenlijk waar ik nu een beetje klaar mee ben. Er wordt nog een beetje revisie gedaan op een aantal dingen en We zijn hier vooral bezig met het testen. Dus het testen van nou die migratie heb gemaakt, wat is de impact die dat heeft op onze modellen is er heel veel veranderd, want Het is toch weer nieuwe code geschreven. Code draait weer net iets anders dan In de oude pipeline, dus daar worden Misschien net iets andere aannames gedaan. Je hebt net iets ander code beschikbaar

00:02:51.780 --> 00:02:58.870
Data Engineer
per periode dat soort dingen dus ja, daar zijn we, nu gaan naar het kijken en.

00:02:59.590 --> 00:03:05.100
Data Engineer
Ja dat, Dat is een beetje het verhaal. Daar komen nog een andere modellen bij en wat We zijn.

00:03:05.810 --> 00:03:10.990
Data Engineer
En Samen met twee aan de data Engineers gaan we nog een aantal modellen migreren.

00:03:12.490 --> 00:03:15.940
Data Engineer
Kan de komende tijd? Ja, en dat zou.

00:03:16.760 --> 00:03:45.250
Data Engineer
Heeft ja op vele manieren hetzelfde werk blijven en net iets anderede stakeholders, andere data Scientists die het model hebben gebouwd en en wat ook leuk is, is hopelijk dat we straks minder features hoeven te schrijven van dat veel modellen binnen de bank gebruiken Natuurlijk ook een beetje dezelfde features. Iedereen wil voor een klant weten hoeveel transacties ze hebben gedaan in de huidige maand of ja, of ze een bepaalde activiteit doen die een beetje.

00:03:46.810 --> 00:03:59.640
Data Engineer
Ja of een coffeeshop hebben of iets dergelijks dat soort dingen als een joint account hebben. Dat zijn allemaal dingen die waarschijnlijk over modellen heen getrokken kunnen worden en daar is de features store perfect voor. Dus dat dat hoef je dan niet opnieuw te schrijven dat daar.

00:04:00.380 --> 00:04:02.400
Data Engineer
Ja, Dat is een beetje waar ik me mee bezig houd.

00:04:03.260 --> 00:04:08.320
Anouk Wolters
Oké en Als ik het goed begrijp, heb je dit eerst voor een bepaald model gedaan, klopt dat?

00:04:08.890 --> 00:04:10.000
Data Engineer
Ja klopt. Klopt.

00:04:10.050 --> 00:04:18.420
Anouk Wolters
En zou je wat meer over dat model kunnen vertellen? In hoeverre ben jij op de hoogte van de hele ontwikkeling van dat model?

00:04:19.700 --> 00:04:28.970
Data Engineer
Ja vrij beperkt in die zin heb het model zelf niet ontworpen. Ik weet dat het model in grote lijnen doet. Het idee is dat de.

00:04:31.220 --> 00:04:56.570
Data Engineer
Er is een rule-bases model, dat al bestaat dat op basis van een aantal regels een alert stuurt als ze als er als ze denken dat een klant risicovol is en aan de hand daarvan kan dan vervolgens een CCD analist kijken van Hey ja, Misschien moeten we toch nog iets dieper naar deze klant kijken? Laten we, die klant reviewen en dat soort dingen, Maar daar kwamen er heel veel.

00:04:58.210 --> 00:04:59.360
Data Engineer
false positives uit.

00:05:00.820 --> 00:05:10.290
Data Engineer
En Er zijn ook heel veel unknown unknowns tenminste. Dat is het idee, dus daar is het model eigenlijk voor gebouwd.

00:05:11.040 --> 00:05:25.400
Data Engineer
Het model probeert dus eigenlijk de filteren in wat er al is getriggerd en ook nog eens te kijken van is er al is er iets wat we wat ja, wat het rule-based model Misschien heeft gemist? Is er een klant Misschien wel risicovol waar we een trigger moet sturen.

00:05:27.440 --> 00:05:30.810
Data Engineer
En ja, ik geloof dat het een ja en die is in.

00:05:32.040 --> 00:05:39.400
Data Engineer
Dat is een beetje wat Ik weet of het model in grote lijnen de details. En ja, die heb ik niet echt voor. Ja.

00:05:40.000 --> 00:05:40.510
Anouk Wolters
Nee.

00:05:40.100 --> 00:05:42.890
Data Engineer
Heb ik niet echt mee bezig houden Natuurlijk, en.

00:05:43.930 --> 00:05:46.190
Data Engineer
Want Dat is voor mij iets minder relevant.

00:05:46.830 --> 00:05:49.690
Data Engineer
Ik zal even kijken of er nog iets speciaals is, een model.

00:05:51.300 --> 00:06:01.110
Data Engineer
Waarschijnlijk wordt het een van de eerste modellen die elke dag gaat draaien, waar veel modellen tot nu toe dan maandelijks draaien. Dat is ook nog een stukje.

00:06:01.870 --> 00:06:03.900
Data Engineer
Wat speciaal is, en.

00:06:05.140 --> 00:06:06.270
Data Engineer
Ja, nee, Dat is het wel.

00:06:07.050 --> 00:06:07.450
Anouk Wolters
OK.

00:06:09.580 --> 00:06:20.830
Anouk Wolters
Dus voor het werk dat jij doet Omdat migreren te doen met welke andere Mensen en rollen werk je daarin Samen en hoe ziet dat er een beetje uit?

00:06:19.410 --> 00:06:19.730
Data Engineer
Ja.

00:06:22.450 --> 00:06:27.240
Data Engineer
Na de eerste is voor mij dus wel belangrijk om de lijst van features te krijgen.

00:06:28.650 --> 00:06:41.930
Data Engineer
En ja daarvoor heb ik Natuurlijk de data scientists nodig en zij hebben Natuurlijk een voor model validation een lijst met features en daar ook de bijbehorende descriptions van.

00:06:43.400 --> 00:06:48.630
Data Engineer
Maar vaak. Jij hebt gewoon vaak dat de description best wel ambigu kan zijn.

00:06:49.540 --> 00:06:53.800
Data Engineer
Dus dan ga je toch nog vaak met samenzitten van. Hé, wat bedoel je hier nou precies?

00:06:54.450 --> 00:07:00.740
Data Engineer
En dus dat, ja, dat zijn de eerste Mensen waar ik eigenlijk een meest contact contact mee heb gehad.

00:07:01.540 --> 00:07:13.020
Data Engineer
En daarnaast hadden we dus ook nog al een pipeline staan, dus dat het een goed om hem tussentijds dat te testen en kijken of onze data nog een beetje klopt en een beetje dezelfde kant op werken.

00:07:13.700 --> 00:07:21.830
Data Engineer
Ja ik nog steeds de data scientists voor nodig en aan de andere kant dan Omdat de features store nee is, hè? Dus de nieuwe data frame work.

00:07:23.730 --> 00:07:35.220
Data Engineer
Bestaat nog niet alles. Er zijn dan wat features die Misschien ja iets minder goed werken of iets wat je graag zou willen. Voorbeeld daarvan is dat je bijvoorbeeld in plaats van Als je data opvraagt voor maand.

00:07:35.930 --> 00:07:43.610
Data Engineer
dat je data krijgt die valide is aan het einde van de maand in plaats van het begin van de maand dat die optie bestaat in ieder geval.

00:07:44.570 --> 00:07:49.840
Data Engineer
Dus dat zijn dingen waar je dan met team ML over gaat zitten, want zij hebben de features store geschreven.

00:07:52.440 --> 00:08:02.440
Data Engineer
Even kijken met wie nog meer hè? Dat zijn in hoofdlijnen wel de twee kern Mensen dus een beetje de ml Engineers en de MLOps kant van dingen en het data Science dus.

00:08:03.260 --> 00:08:03.550
Anouk Wolters
Ja.

00:08:04.550 --> 00:08:04.920
Data Engineer
Ja.

00:08:07.070 --> 00:08:09.030
Anouk Wolters
Oké en en.

00:08:10.620 --> 00:08:19.540
Anouk Wolters
Even kijken als jij toen jij begon aan aan dit project om die features store op te zetten. Hoe ging zo'n project? Hoe ziet dat eruit?

00:08:17.190 --> 00:08:17.490
Data Engineer
Ja.

00:08:20.640 --> 00:08:28.070
Data Engineer
Het begint eerst met een heel lang wachten tot je access hebt tot alles. Maar ja, dat hoort erbij. Even kijken.

00:08:30.380 --> 00:09:01.030
Data Engineer
Ja, wat ik als eerste het gedaan is, eigenlijk gewoon kijken naar de oude data pipeline en eigenlijk zo snel mogelijk met een voorstel komen, want dan ja of doorgaan kon gaan of afgeschoten kon worden. En ja, dan begrijpen waarom het afgeschoten wedhet. In dit geval was het, dus Ik had het eerste idee om heel veel eigenlijk hetzelfde doen als in de data pipeline maar dat kan natuurlijk lijken als een soort van knowhow van de feature store waar alles dan alsnog in elkaar vermengd zou worden, terwijl je een feature zo Independent mogelijk wil houden.

00:09:11.820 --> 00:09:21.060
Data Engineer
Dus ja, Dat is een. Dat is mijn eerste voorstel vervolgens. Ja merkte ik dat ik te weinig wist over de feature store en gelukkig is er iemand vanuit Team ML

00:09:21.950 --> 00:09:26.210
Data Engineer
iemand waar je altijd langs kan, dus ja, eigenlijk gewoon een paar weken lang.

00:09:26.960 --> 00:09:39.650
Data Engineer
Er zitten veel vragen gesteld en We hadden een vragen uurtje per week. En ja, Ik heb die in een paar weken langer bezig gedomineerd, Maar dat op een gegeven moment helpt het dan heel erg. Ja, dan begrijp je van het hele proces.

00:09:40.850 --> 00:09:52.180
Data Engineer
En toen ben ik met een nieuw voorstel van komen en en met een voorstel gekomen voor het testen Omdat het Het was. Het is waarschijnlijk het grootste model qua aantal features, dus 250 ofzo.

00:09:52.950 --> 00:10:12.750
Data Engineer
En en dat betekende wel dat ik ja, Ik was ook gelijk een beetje bang van Als ik gewoon blindelings dingen gaat schrijven Zonder tussentijdse test te doen. Op een gegeven moment is het straks december komen we erachter dat helemaal niet klopt met de oude piepen. Als ze die toch hebben, dan kunnen we tussentijds testen. Nou, dat vond Iedereen ook fijn, dus dat hebben we gedaan.

00:10:13.390 --> 00:10:22.040
Data Engineer
En ja, dus Dat is een beetje het beginnen. Vervolgens was het eigenlijk gewoon ja door door schrijven en.

00:10:22.820 --> 00:10:41.800
Data Engineer
Want gewoon feature voor feature ga je dan gewoon schrijven en en testen en dan kijk je wat er fout is en het fijne is dat features kunnen heel goed gecategoriseerd worden. Dus Als je een fout in een categorie maakt dan Als je die oplossen, los je dat vaak ook voor de hele categorie op. Dus Dat is zo.

00:10:42.490 --> 00:10:52.920
Data Engineer
En ja dus daar, daar hebben ze tegen een ander heel veel bij geholpen, dus gewoon heel veel vragen kunnen stellen over wat precies de features Store is en wat die kan, wat hij niet kan.

00:10:54.360 --> 00:11:00.770
Data Engineer
Ja voorstellen kunnen doen van wat ik zou willen van de feature store en wat de data scientists Misschien nog zouden willen van de feature store.

00:11:01.530 --> 00:11:03.050
Data Engineer
Dus ja, Dat is het een beetje.

00:11:04.500 --> 00:11:14.580
Anouk Wolters
Oké, en toen jij deze opdracht kreeg had had je werd er dan ook bepaalde criteria of requirements aan je meegegeven.

00:11:15.340 --> 00:11:22.440
Data Engineer
En, wat bedoel je van de Van het resultaat van de opdracht?

00:11:22.270 --> 00:11:22.590
Anouk Wolters
Ja.

00:11:25.050 --> 00:11:26.010
Data Engineer
Ja Dat is.

00:11:27.360 --> 00:11:28.670
Data Engineer
Nee, Dat was nogal vaag.

00:11:30.320 --> 00:11:37.530
Data Engineer
Kijken wat het volgens mij de derde migratie en was hiervoor een project gedaan met een andere daad indien er.

00:11:38.760 --> 00:11:44.610
Data Engineer
En dat het dat dat nogal lang geduurd, want hij was de eerste. Dus wel, ja, dan heb je een beetje de show. Ja.

00:11:45.490 --> 00:11:59.790
Data Engineer
Ja, Het is een beetje ongelukkig is dus dan en Hij heeft met alle onduidelijkheden dan te maken gehad en ik dan iets minder, maar nog steeds wel wat ik merkte is dat wat Iedereen eigenlijk op een gegeven moment is dat we weinig visie was om.

00:12:01.100 --> 00:12:13.010
Data Engineer
Hoe duur is om migratie zou worden, dus in termen van hoeveel tijd het zou kosten en wat voor ingewikkelde ja, ingewikkelde dingetjes er allemaal in zitten en.

00:12:13.690 --> 00:12:33.260
Data Engineer

Dus in die zin dat er Misschien meer criteria kunnen zijn. Want ja, Dat was ook niet echt een bepaalde tijd aan verbonden, Omdat men ook gewoon niet wist van hebben. Ja, Het is gewoon een beetje een unknown, het is de grootste features set tot nu toe, Het is een nieuwe library en We hebben geen idee hoe lang het precies gaat duren.

00:12:33.980 --> 00:12:49.380
Data Engineer
Dus qua criteria was dat eigenlijk gewoon criteria die ook wel eens vacatures dan gewoon nette code schrijven, En ja, Dat is Misschien wel de belangrijkste. En ja betrouwbaar qua data engineering.

00:12:50.640 --> 00:12:51.380
Anouk Wolters
Ja oké.

00:12:52.120 --> 00:12:58.890
Anouk Wolters
En nu het bijna ten einde loopt of ten einde loopt, hoe wordt er dan geëvalueerd of.

00:12:59.990 --> 00:13:03.200
Anouk Wolters
Uiteindelijk is voldaan aan de opdracht.

00:13:05.040 --> 00:13:06.210
Data Engineer
Dat is een goede vraag.

00:13:07.290 --> 00:13:09.050
Data Engineer
Dus een definition of done, bedoel je?

00:13:09.850 --> 00:13:10.200
Anouk Wolters
Sorry.

00:13:10.760 --> 00:13:13.650
Data Engineer
Een definition of done, dus wanneer ben je klaar of?

00:13:13.470 --> 00:13:13.930
Anouk Wolters
Ja.

00:13:14.450 --> 00:13:14.920
Data Engineer
Oké.

00:13:16.530 --> 00:13:22.080
Data Engineer
Dat is, Dat is nog niet helemaal duidelijk. We hebben wel wat.

00:13:23.830 --> 00:13:53.790
Data Engineer
Ja wat we nu bijvoorbeeld doen is dat we testen met de features die dus nu gecalculeerd zijn en die gebruiken om het model te trainen en te kijken wat voor impact heeft. En Maar dat doe ik niet zelf. Dat doet de data scientist. Het is dan met de features die Ik heb berekend en hij komt dan bij mij terug met feedback van hé, deze features zijn toch wel iets minder dan dan we zouden willen zou je hier nog een keer na kunnen kijken, dus de laatste weken hebben bijvoorbeeld een daar daarvoor een aantal features.

00:13:53.840 --> 00:13:58.490
Data Engineer
Moeten aanpassen en dan nog een keer moeten draaien. En Ik denk dat dan.

00:14:00.270 --> 00:14:14.460
Data Engineer
Ja, We hebben het team is ook nu dus met twee andere data Engineers geworden en Het is echt een team aan het worden, dus We hebben nu ook sinds kort een scrum master en met hem gaan we ook iets meer de komende weken. Echt een definition of done.

00:14:15.710 --> 00:14:27.700
Data Engineer
definiëren, want wat er nu op lijkt is eigenlijk gewoon dat we dus een model gaan trainen. En Als de data Scientist tevreden is met het resultaat en dat hij niet teveel afwijken van wat er voorheen zagen, daar zit het allemaal goed.

00:14:30.080 --> 00:14:39.850
Data Engineer
Maar Misschien dat we dat dus Samen met het nieuwe team en scrum master en product owner wat harder kunnen maken voor de toekomst, dat we gewoon echt een iets duidelijkere définitions dan kunnen hebben.

00:14:40.470 --> 00:14:50.000
Anouk Wolters
Ja en je zegt dus dat er op basis van de features straks in model getraind gaat worden en op basis daarvan wordt gekeken of het goed genoeg is.

00:14:51.200 --> 00:15:00.250
Anouk Wolters
Maar als jij zelf bezig bent met die features hoe, hoe kun je dan weten of inschatten wanneer het goed goed is om een model erop te trainen?

00:15:01.360 --> 00:15:03.020
Data Engineer
Ja Dat is.

00:15:04.230 --> 00:15:15.560
Data Engineer
Dat hebben we dus voor het begin gedaan, door dus tussentijds te testen met de Legacy Pipeline. Dus de pipeline die er al stond en dat hebben we ongeveer. Ik denk zo een.

00:15:16.570 --> 00:15:38.000
Data Engineer
170 features gedaan en het idee wat we dan hadden is dan om te kijken van hé hoeveel verschillende waardes dus absoluut verschillen zijn. Er hoeveel zijn dat er? En als dat onder de 1% is dan dan zijn we op zich tevreden Omdat je toch binnen de twee frameworks heb je toch te maken met iets ander data die je beschikbaar hebt, periode waarvoor je berekend

00:15:38.820 --> 00:16:10.300
Data Engineer
En dus onder de 1% zijn we ook richting eigenlijk het einde toe, hebben we besloten. Van ja oke het heeft niet per se heel veel. We hebben nu de hele features het heeft niet per se heel veel zin om al die losse testen nog te schrijven, kunnen we eigenlijk gewoon beter gewoon doorpakken met de impactanalysei, dus ook waar het nu aan het doen zijn. En dus eigenlijk hebben we aan het begin zoals van die test gebruiken om ons bij de hand te nemen van hé, je gaat nog steeds het goede kant op die fase en ja, en Als je dus niet de goede kant op.

00:16:10.350 --> 00:16:16.820
Data Engineer
Dan is dat gelijk te zien in test en. Dan kan je vragen stellen van hé, wat hebben jullie gedaan hebben, wat hebben wij gedaan en waar gaat het mis?

00:16:18.540 --> 00:16:29.260
Data Engineer
En dan nu is het eigenlijk meer doorpakken en impactanalyses doen dan met een beetje vertalen van hé. De methode tot nu toe was goed, dus dan ja, waarschijnlijk gaan we de juiste kant op.

00:16:30.110 --> 00:16:32.050
Anouk Wolters
En wat houdt zo'n impact analysis in?

00:16:32.790 --> 00:17:02.930
Data Engineer
Ja, Dat is dus wat ik net uitleg, Dat is dus wat het moet. Het trainen van een model met de berekende features. En ja, Ik heb nu bijvoorbeeld net 10 features gekregen van hé, ik zou je Misschien nog nog eens naar kunnen kijken van Dit is toch wel een groot verschil met wat we wat we zouden verwachten en en ja dat hoe groot het verschil is. Ja, dat wordt bepaald door de data scientists dus de tevredenheid van Van.

00:16:34.890 --> 00:16:35.270
Anouk Wolters
Het.

00:17:03.090 --> 00:17:04.520
Data Engineer
De data scientists in die zin.

00:17:05.020 --> 00:17:05.400
Anouk Wolters
Oké.

00:17:07.070 --> 00:17:24.920
Anouk Wolters
En je zegt, klopt het dan dat het oude zeg Maar de oude situatie wordt gezien als dat dat optimaal is en dat alles waarvan de nieuwe situatie afwijkt dat het een afwijking is, of zou het ook kunnen dat de nieuwe situatie bepaalde opzichten ook verbetering te weeg brengt?

00:17:23.900 --> 00:17:24.340
Data Engineer
Ja.

00:17:26.090 --> 00:17:34.620
Data Engineer
Nee, Het is zeker dat laatste. Dat is ook de overwegingen om op een gegeven moment die unit tests die één op één test ja achterwege te Laten.

00:17:36.950 --> 00:17:46.900
Data Engineer
Ja, de nieuwe situatie kan in sommige gevallen ook wel echt beter zijn. Dat hebben we ook een paar keer gemerkt dat dat we bijvoorbeeld fouten tegenkwamen In de oude pipeline.

00:17:48.290 --> 00:17:53.210
Data Engineer
Dus ja, het kan zeker zijn dat de nieuwe situatie beter is, maar.

00:17:54.060 --> 00:17:59.250
Data Engineer
We hebben de oude pipeline niet helemaal losgelaten, puur om om dat ja.

00:18:00.430 --> 00:18:19.630
Data Engineer
Als je zoveel features hebt die je dus los van elkaar moet gaan implementeren, is toch wel handig Als je gewoon gaandeweg nog een beetje richting ja weet dat je de juiste richting op gaat en dat hoeft dus niet perfect zijn, hebben we gelijk ja gezegd van. Dat hoeft dus niet helemaal een op een te kloppen, Maar we willen wel zien dat het.

00:18:21.050 --> 00:18:24.420
Data Engineer
Weet je de juiste gedachten heeft, Dat is wat we voorheen hadden.

00:18:26.520 --> 00:18:40.260
Data Engineer
En echt grote verschillen ga je Natuurlijk altijd vragen stellen en daar komt bijvoorbeeld vaak uit. Dan hebben we hier een fout gemaakt In de features store, maar ook vaak genoemd. Hé In de Legacy pipeline staat dit, maar klopt dit wel? En dus ja.

00:18:41.640 --> 00:18:48.950
Anouk Wolters
Oké en als kun je Misschien een voorbeeld geven van in hoeverre wat een verbetering zou kunnen zijn.

00:18:51.040 --> 00:18:52.520
Data Engineer
Ja even kijken.

00:18:54.360 --> 00:19:13.110
Data Engineer
Ja, We hebben bijvoorbeeld een situatie gehad waarin de definitie van de feature als we die direct implementeren dat we andere resultaten zien. Maar dat was dus een fout In de description, en dus dat de description dus ambigu is. En description is Natuurlijk onderdeel van de deliverable van het model. Die gaat naar model validatie, dus dat.

00:19:13.780 --> 00:19:17.630
Data Engineer
Ja, dat moet aangepast worden en wat we ook zagen, is dat.

00:19:19.390 --> 00:19:35.990
Data Engineer
We gebruiken data met de historische data van klanten, Maar we hebben niet historie tot 10 jaar terug ofzo om dat zeg maar ook te vangen, deed de legacy pipeline opvullen op basis van bepaalde.

00:19:38.720 --> 00:20:06.970
Data Engineer
Wat de datum was of als er gaten zijn In het In de In de data source dat hij ook werden opgevuld. En nou proberen we dus proberen wij dus aan de features store kant dat de implementeren, dus Zonder dus direct op de source zelf. Maar dan zagen we bijvoorbeeld hé, de manier waarop jullie gaten vullen, dat klopt niet helemaal. Je hebben bijvoorbeeld klanten die eigenlijk pas in 2021 bestaan. Neem je al mee in 2018

00:20:08.160 --> 00:20:11.160
Data Engineer
Dat klopt niet helemaal, dus dat zouden jullie moeten kijken.

00:20:13.450 --> 00:20:14.940
Data Engineer
Ja, dus dat soort dingen eigenlijk?

00:20:15.910 --> 00:20:17.520
Anouk Wolters
Oké en zijn er dan ook?

00:20:18.630 --> 00:20:27.360
Anouk Wolters
Bepaalde soorten checks die dan altijd doet, of een proces die je doorloopt om dit soort fouten eruit te halen. Of kom je dat toevallig tegen?

00:20:28.210 --> 00:20:41.740
Data Engineer
Nou, dat komt dus vooral te geven, dus de door de unit test dus stellen we een half miljoen klanten en we doen een unit test en. We hebben dus meer dan vijfduizend mismatches.

00:20:43.120 --> 00:21:12.130
Data Engineer
Ja dan ja dan moeten we gewoon gaan kijken, dus voor dan pakken we gewoon. En ja, Misschien moet ik even bij zeggen, alle features zijn op customer niveau dus op party ID dus elke elke eigenschap van het Van een bepaalde customer, dus als er bijvoorbeeld vijfduizend mismatches zijn, gaan we kijken van Hey pak een party ID doe direct op de source dezelfde berekening die in de feature description staat en de description die de data scientist heeft gegeven,

00:21:13.330 --> 00:21:43.080
Data Engineer
En wat zie je dan? En Als je dan zeg maar Als de feature store gelijk heeft, dan gaan we terug naar de de data scientists van Hé, jullie moeten hier even naar kijken, want we zien dat we hier toch echt doen wat jullie zeggen en dat we niet hetzelfde krijgen als dat jullie wat je wat In de Legacy pipeline staat krijgen. En dan gaan zij dat nou doorvoeren In de Legacy pipeline om er dan bijvoorbeeld vervolgens achter te komen dat er te veel klanten zijn in 2017 die helemaal niet bestaan en pas in 2021.

00:21:44.500 --> 00:21:47.880
Data Engineer
Dus Dat is een beetje de unit test en in die zin zeg maar.

00:21:49.410 --> 00:21:50.120
Data Engineer
Als er.

00:21:51.110 --> 00:21:57.840
Data Engineer
Ja, zo kan wandelstok geweest om een beetje op gang te houden, kijken ons scherp te houden.

00:21:59.110 --> 00:21:59.860
Data Engineer
Gaandeweg.

00:22:00.640 --> 00:22:04.790
Anouk Wolters
Oké oké en worden. Die unit test dan ook gedaan.

00:22:05.840 --> 00:22:10.710
Anouk Wolters
Of zijn die ook gedaan in eerste instantie, toen het model werd ontwikkeld, want blijkbaar.

00:22:11.500 --> 00:22:17.190
Anouk Wolters
Klopte er dan dingen niet In de data die eerst instanties gebruikt Als ik het goed begrijp.

00:22:17.800 --> 00:22:23.430
Data Engineer
Dat weet ik niet, Ik heb ze niet gezien dat. Ja, Dat is voor voordat ik voor mijn tijd zeg maar.

00:22:25.670 --> 00:22:28.640
Data Engineer
Van de code die Ik heb gezien waren die is volgens mij niet.

00:22:30.070 --> 00:22:38.150
Data Engineer
Wat we wel hebben gedaan, is aan het einde nog. Ja, wij hebben de legacy pipeline eigenlijk helemaal niet getest in die zin.

00:22:40.530 --> 00:23:10.420
Data Engineer
Aan het einde hebben we wel nog kiezen we dan 3 periodes waarvoor we dus aan het einde nog een keer unit test doen. Daarom niet Alleen het aantal mismatches, maar ook de correlatie tussen de features te berekenen. En als ze echt zeg maar grote red flags zijn om die dan nog de te op te sporen we doen, dan zeg Maar de vroegste periode en laatste periode ergens midden in een periode toetsen. Maar van de Legacy pipeline zelf heb ik eigenlijk geen.

00:23:12.330 --> 00:23:24.930
Data Engineer
Nee, Ik heb helemaal geen unit test daar gezien, Maar het kan Natuurlijk zijn wat Ik heb gemerkt. Is dat bijvoorbeeld een ander team Misschien een pipeline heeft en dan wel unit tests heeft? Dat is dat eigenlijk Misschien een beetje van Van de pillar af ook.

00:23:25.610 --> 00:23:29.410
Anouk Wolters
Wil je dat ze dan dezelfde datum data gebruiken in die teams?

00:23:28.980 --> 00:23:54.810
Data Engineer
Nou, nee, ik bedoel meer van je bijvoorbeeld iemand In de transaction monitoring pillar die elke model heeft gebouwd die ook data pipeline heeft. Dat zij bijvoorbeeld wel unit tests

hebben voor een data line en maar nu niet toevallig In deze pillar dat dat daarvoor niet is gekozen. Dus want Ik heb volgens mij wel van een aantal modellen wat wat test gezien.

00:23:49.780 --> 00:23:50.020
Anouk Wolters
Ja.

00:23:55.470 --> 00:23:56.010
Anouk Wolters
Oké.

00:23:56.810 --> 00:23:57.100
Data Engineer
Ja.

00:23:57.530 --> 00:24:00.590
Anouk Wolters
Dat kan er gewoon per case per model verschillen.

00:24:01.050 --> 00:24:02.930
Data Engineer
Nou ja, daar lijkt het wel op. Ja.

00:24:02.990 --> 00:24:03.270
Anouk Wolters
Ja.

00:24:04.510 --> 00:24:06.380
Anouk Wolters
Oké oké.

00:24:08.060 --> 00:24:12.740
Anouk Wolters
Kijken die die features data die wordt gebruikt.

00:24:13.760 --> 00:24:18.150
Anouk Wolters
Weet je hoe die in eerste instantie is verzameld en waar in de bank dat gebeurt?

00:24:19.280 --> 00:24:21.890
Data Engineer
Je hebt gebeurt door kaap.

00:24:23.330 --> 00:24:29.580
Data Engineer
Ik weet niet waar de afkorting voor staat en super afkortingen en het data enabeling.

00:24:30.870 --> 00:24:38.300
Data Engineer
En zij verzamelen bijvoorbeeld. Ik weet bijvoorbeeld voor transacties dat er vanuit verschillende databronnen transacties zijn.

00:24:40.070 --> 00:24:48.220
Data Engineer
En zij verzamelen die data bronnen, doen transformaties op die data bronnen om uiteindelijk op een transactie in die dataset te komen.

00:24:51.370 --> 00:25:17.480
Data Engineer
Ja en zij regelen ook dat er bronnen zijn die curators zijn, dus dat zijn echt de Gold Standard bronnen, maar Er zijn ook een aantal bronnen die raw zijn, dus dat zijn nog net geen goald standard zijn, en die je gewoon die eigenlijk het liefst liever niet in je features Store hebt, want je wilt eigenlijk de beste databronnen in je feature store hebben Omdat ja als Iedereen die data gaat gbruiken, dan gebruikt iedereen gewoon gold standard data. Dat zou heel mooi zijn.

00:25:17.740 --> 00:25:19.630
Anouk Wolters
Wat is dat precies? Gold standard data?

00:25:20.500 --> 00:25:21.840
Data Engineer
Nou dus.

00:25:23.900 --> 00:25:25.010
Data Engineer
Hoe leg ik dat uit?

00:25:26.130 --> 00:25:29.640
Data Engineer
Ja God standard is eigenlijk, Je moet het zo zien.

00:25:31.080 --> 00:25:34.860
Data Engineer
Er is data bijvoorbeeld.

00:25:37.420 --> 00:26:07.210
Data Engineer
Hoe heet het systeem? Ik geloof dat het systeem DAAL heet en en daar komt dan data uit, bijvoorbeeld voor de transacties en maar zijn 3 transactiesystemen die die data verzamelen en daar zitten ook verschillende transacties in. En nou, het liefst wil je dat allemaal in één één set hebben, dus het data enabling team gaat een logica bedenken om die daar te zetten, zeg maar te formuleren. En wat er dan In de goal standard folders

00:26:07.270 --> 00:26:10.680
Data Engineer
Komt is dan die uiteindelijk In de geïntegreerde set.

00:26:11.560 --> 00:26:13.290
Data Engineer
Die wij dan vervolgens kunnen gebruiken.

00:26:14.890 --> 00:26:24.620
Data Engineer
En, wat een eigenschap van de Gold standaard is dat het zo dicht mogelijk blijft bij eigenlijk ja, de originele bronnen.

00:26:25.970 --> 00:26:30.160
Data Engineer
Zo weinig mogelijk verschilt van het originele bron.

00:26:31.810 --> 00:26:41.060
Data Engineer
Ja, Dat is eigenlijk een beetje het idee van een gold standard, set dat eigenlijk zo dicht mogelijk bij de realiteit zit, maar ook de data zo netjes mogelijk is geformuleerd in die zin.

00:26:41.510 --> 00:26:42.270
Anouk Wolters
Ja oké.

00:26:43.030 --> 00:26:46.040
Anouk Wolters
Oké, even kijken toen.

00:26:47.180 --> 00:26:56.650
Anouk Wolters
Als je bezig bent met dat migreren van de features, is het dan de bedoeling dat alle features die in personele model waren ook gemigreerd worden of doe je ook nog een soort?

00:26:57.430 --> 00:27:03.370
Anouk Wolters
check up van welke zijn er ook features die je eigenlijk Misschien liever beter niet kan gebruiken?

00:27:05.150 --> 00:27:16.820
Data Engineer
Het zou kunnen, Maar dat is niet per se onderdeel van mijn takenpakket. Het is iets meer iets wat de data scientist zou moeten doen, dus dan kan ik me voorstellen dat je features Selection doet en.

00:27:17.600 --> 00:27:26.290
Data Engineer

En wellicht een nog gebeurt, zeg maar na deze stad dus na de migratie, maar daar heb ik in principe geen zicht op, dus Dat is.

00:27:27.530 --> 00:27:40.840
Data Engineer
Ja, Ik kan me voorstellen ik het model is al gevalideerd door model validation, dus Ik kan me voorstellen dat dat soort van het einde is, dus alle features die daaruit komen. Dat zijn de features die worden geïmplementeerd en worden meegenomen naar productie.

00:27:41.550 --> 00:27:47.770
Data Engineer
Ja maar alle feature selection opties of overwegingen. Daar hou ik me niet mee bezig.

00:27:47.930 --> 00:27:48.390
Anouk Wolters
Nee OK.

00:27:49.420 --> 00:27:50.350
Anouk Wolters
Oké duidelijk.

00:27:54.960 --> 00:27:55.730
Anouk Wolters
Het kijken.

00:28:01.730 --> 00:28:11.400
Anouk Wolters
Zijn er bepaalde risico's of mogelijk fouten die ze zouden kunnen voordoen bij de migratie en.

00:28:12.920 --> 00:28:18.590
Anouk Wolters
Heb jij, heb je daar, ben je daar bewust van of heb je daar zicht op van wat? Wat zijn nou risico's die zich kunnen voordoen?

00:28:23.030 --> 00:28:26.440
Data Engineer
Ja Er zijn, Er zijn wel een aantal risico's.

00:28:27.590 --> 00:28:43.830
Data Engineer
We hebben het een paar keer gehad over de kwaliteit van de data, dus zoals sommige datasets, wat ik daar zei, dat er gaten zijn inderdaad gezegd dat een voorbeeld daarvan is het een klant bestaat in augustus 2017.

00:28:45.420 --> 00:28:52.990
Data Engineer

September niet bestaat en in oktober ook niet, maar dan in november wel weer wel weer bestaat Zonder dat er iets per se is veranderd in zijn eigenschappen.

00:28:54.050 --> 00:29:02.580
Data Engineer
Zijn dat soort dingen dus vooral data quality is volgens mij risico geweest. Wat Ik weet is dat.

00:29:04.920 --> 00:29:24.000
Data Engineer
we er wel scherp op zijn In de zin van dat toen dat voorkwam in de transaction set heeft een van data science leden een voorstel gedaan om eigenlijk een een transformatie toedoen van de transaction set, een voorstel naar data enabling van hey, Dit is wat we allemaal zien wat er mis is.

00:29:26.190 --> 00:29:56.880
Data Engineer
Ja, we, we zouden het graag zo opgelost zien worden. Zijn jullie daarvan op de hoogte? Willen jullie dat meenemen, dus ik ik. Dat zijn een beetje de risico's waar ik een beetje mee te maken heb en bijvoorbeeld nog een risico is dat Als het model straks dagelijks moet draaien, waar de features het nu niet helemaal opgebouwd zijn. Maar ja, dat komt dus nog wel is nog wel de vraag, maar of je wat je nu hebt Als je hem maandelijks draait is zodat je.

00:29:57.650 --> 00:30:28.560
Data Engineer
Als je in december 2021 features draait voor januari 2021, dan kan je in principe hetzelfde resultaat verwachten Omdat je precies dezelfde data selecteer als die je toendertijd had kunnen selecteren. Als je dat dagelijks doet, wordt het een ander verhaal. Als je bijvoorbeeld s middags wil berekenen of de customer een native is of niet. Nou, dat kan zijn dat diegene in Nederland woont en werkt maar Misschien een half uur of een, weet ik veel.

00:30:28.600 --> 00:30:38.110
Data Engineer
5 uur later is dat Misschien aangepast en en is die customer niet meer native.

00:30:39.200 --> 00:30:51.990
Data Engineer
Dus de replicability van de data is kan kan nogal een beetje verdwijnen en daar zijn we nu ook gesprek over aan het voeren hoe we dat ja kunnen ja minimaliseren of eigenlijk kunnen.

00:30:52.640 --> 00:31:18.370
Data Engineer
Weghalen dat risico, ik hou me niet zozeer bezig met bijvoorbeeld dingen zoals fairness of eigenlijk meer dingen die met modelleren te maken hebben, dus mijn risico is eigenlijk ja ja, dingen die ik moet gewoon de juiste data aan het model geven, zoals om het operationeel zo zo goed mogelijk te Laten gaan. En dat zijn dit zijn de meeste de twee. Zo zie ik een beetje tegen komen.

00:31:19.540 --> 00:31:21.770
Anouk Wolters
Oké en en.

00:31:23.210 --> 00:31:26.130
Anouk Wolters
Even kijken, dus je zei dat Als je vanaf een maand.

00:31:26.800 --> 00:31:29.220
Anouk Wolters
Maandelijks naar dagelijks bijvoorbeeld gaat.

00:31:30.830 --> 00:31:33.980
Anouk Wolters
Moet ik het dan zo zien dat de data die gebruikt wordt?

00:31:35.720 --> 00:31:38.290
Anouk Wolters
Over de tijd steeds aangevuld wordt of veranderd wordt.

00:31:39.490 --> 00:31:53.090
Data Engineer
Ja, Dat is hoe de tabellen nu werken, is dat nou stel Ik ben een klant, dan heb ik een party ID en daar staat mijn naam bij en mijn achternaam bijvoorbeeld in april 2021.

00:31:54.670 --> 00:31:57.590
Data Engineer
En dat is dan een regel als er iets verandert.

00:31:59.090 --> 00:32:00.950
Data Engineer
In juni, dan komt daar een regel bij.

00:32:01.350 --> 00:32:01.690
Anouk Wolters
Ja.

00:32:02.440 --> 00:32:21.430
Data Engineer
Die validity van die eerste regel die eindigt Omdat er nieuwe regel bij komt. Stel, Ik ben getrouwd met iemand en ik krijg die de achternaam van diegene, dus mijn achternaam verandert dus dan in in juli is er een nieuwe regel met de nieuwe achternaam en een nieuwe validity date.

00:32:23.310 --> 00:32:47.920
Data Engineer
Dus Dat is een beetje. Het gebeurt dus ook data wordt in die zin op die manier aangepast, waardoor je dus wat ik zei Als ik dus de feature weer wil berekenen voor april 2021, Toen ik

nog een ander achternaam had en dan kan dat nog steeds in december 2022, puur Omdat dat In de tabel nog steeds beschikbaar is.

00:32:49.190 --> 00:32:50.050
Anouk Wolters
Oké oké.

00:32:50.940 --> 00:32:52.620
Anouk Wolters
En en.

00:32:53.560 --> 00:33:00.160
Anouk Wolters
Hoe gaat? Hoe wordt er dan omgegaan met mogelijke veranderingen in data distributies over tijd.

00:33:03.340 --> 00:33:06.280
Data Engineer
Ja dus drift, bedoel je van?

00:33:06.400 --> 00:33:06.820
Anouk Wolters
Ja.

00:33:08.350 --> 00:33:11.490
Data Engineer
Ja, nee, Dat is ook iets waar ik me niet mee bezig houdt.

00:33:16.730 --> 00:33:36.150
Data Engineer
Maar ja, Ik kan me wel voorstellen dat Als je dus features hebt, dan op een gegeven moment feature drift wil berekenen. Nou, Dit is dus een moment waar we dus naar een feature set naar toe werken die naar productie kan, dus Dat is een soort van nul punt en kan ik me voorstellen dat je vanaf daar een soort berekening gaat doen.

00:33:37.360 --> 00:33:40.420
Data Engineer
Ja periodiek van oké, hoe is dat veranderd?

00:33:42.180 --> 00:34:05.390
Data Engineer
Over tijd dus en daar zijn ook dingen. Ik heb een aantal teams al gehoord of een dashboard voor feature drift. Ik weet niet of die gebaseerd is op oude pipelines of nieuwe pipelines, Maar ik denk dat het ja het qua ideeën theoretisch gezien dan niet verandert Omdat je bent gegaan en de features door ja Misschien in je code haal je de data ergens anders ook, Maar dat zal waarschijnlijk niet heel veel veranderen.

00:34:06.890 --> 00:34:11.120
Anouk Wolters
Oké, want jij zorgt dus uiteindelijk dus voor dat nulpunt dan van die features.

00:34:10.800 --> 00:34:14.330
Data Engineer
Dat is wel inderdaad. Ja, Dat is van een beetje waar we naartoe werken.

00:34:15.990 --> 00:34:27.060
Data Engineer
En een nul punt is dan een beetje een gek woord, want Als de featuree klopt voor nu, dan hoort hij ook te kloppen voor januari 2021 en dat testen we ook, zeg maar, dus ja, Dat is inderdaad.

00:34:27.850 --> 00:34:29.190
Data Engineer
een beetje waar we nu mee bezig zijn?

00:34:29.860 --> 00:34:30.320
Anouk Wolters
Dank u.

00:34:31.310 --> 00:34:31.790
Anouk Wolters
Oké.

00:34:35.160 --> 00:34:35.810
Anouk Wolters
Even kijken.

00:34:40.790 --> 00:34:44.800
Anouk Wolters
Hoort er bij die feature store ook een bepaalde.

00:34:46.070 --> 00:34:51.660
Anouk Wolters
Monitoring van of ik nog wel allemaal klopt wat er per features staat bijvoorbeeld.

00:34:55.760 --> 00:34:57.010
Data Engineer
Wat bedoel je met klopt?

00:34:59.370 --> 00:35:04.060
Anouk Wolters
Nou, Als ik goed begrijp, heb je per features dus data die erbij hoort.

00:35:04.920 --> 00:35:07.460
Anouk Wolters
En dat over tijd wordt dat dan aangevuld?

00:35:09.130 --> 00:35:11.080
Anouk Wolters
Wordt het ook gemonitord of?

00:35:12.250 --> 00:35:21.700
Anouk Wolters
Nou ja of bijvoorbeeld nieuwe data daar mist of op een gegeven moment of er fouten kunnen optreden, is daar een manier voor Omdat het kunnen detecteren?

00:35:22.490 --> 00:35:27.500
Data Engineer
Er zijn verschillende scenario's, dus.

00:35:28.890 --> 00:35:36.370
Data Engineer
Één is dus ja, gegeven dus die van de tabel geeft altijd historisch opnieuw berekenen kan je hetzelfde resultaat verwachten.

00:35:39.100 --> 00:36:08.420
Data Engineer
Er zijn checks die je dus mee kan geven, bijvoorbeeld Als je een bepaalde distributie verwacht van de data Als je bepaalde een bepaalde mean verwacht of standard deviation of Als je verwacht in ieder geval zoveel van. Ik weet niet of je of je grade expectations kent. Dat is een library die waarmee je dus over tabellen bepaalde dingen kan asserten te dus je kan zeggen van ik verwacht in ieder geval dit en kan je In de features store ook meegeven, dus aangeeft van hé ik verwacht deze

00:36:08.470 --> 00:36:12.600
Data Engineer
Distributie dus als het zo blijft en.

00:36:13.440 --> 00:36:15.090
Data Engineer
Ik weet niet of de.

00:36:20.050 --> 00:36:26.680
Data Engineer
Het ja, het kan Natuurlijk zijn dat. Ja, Dat is ook iets voor waar we een paar keer over hebben gehad. Dat Als je.

00:36:27.920 --> 00:36:44.030
Data Engineer
Stel jouw model gebruikt data van periode 12 tot periode 61 en en jij zet dus op een nul punt. Zet jij dus die die die harde eis van: Ik wil een gemiddelde van 24,2.

00:36:45.770 --> 00:37:04.060
Data Engineer
Het kan inderdaad zijn dat als jij in periode 75 komt en de feature wil berekenen dat die dat dat dus niet meer geldt. het seintje wat daar uitkomt, is puur dat de feature niet wordt berekend, dus de feature wordt dan niet opgeslagen, dus als daar niet aan die eis.

00:37:04.680 --> 00:37:12.080
Data Engineer
Als je niet aan die eis wordt voldaan, dan wordt de feature niet berekend en Dat is dan je directe sein van hé. Er is iets mis, dus dan moet je ermee aan de slag.

00:37:13.090 --> 00:37:15.720
Anouk Wolters
En voor wie is dat? Zijn dan, wie gaat er dan naar kijken?

00:37:16.560 --> 00:37:19.810
Data Engineer
Hoogstwaarschijnlijk, want Dat is volgens mij

00:37:21.030 --> 00:37:43.300
Data Engineer
Dat zal waarschijnlijk team ML zijn, want zij draaien de features in productie, dus daar krijg je het seintje binnen. Maar het elke feature heeft ook een owner, dus degene die hem heeft gebouwd, dus dan kan ik me voorstellen dat de owner dan vervolgens een een berichtje op teams krijgen of een mailtje krijgt van hé, er gaat iets mis met feature hier. Want ja, er wordt niet meer voldaan aan jouw expectation.

00:37:44.640 --> 00:37:45.340
Anouk Wolters
Ja oké.

00:37:47.140 --> 00:37:58.750
Anouk Wolters
En is er, dan geven jullie hier standaard aantal checks mee aan die features Store of is dat iets waar jij dan ook mee bezig houdt, of is dit al bedacht hoe?

00:38:00.010 --> 00:38:05.370
Data Engineer
Nou, Het is, Het is iets wat in theorie kan. Ik heb het In de praktijk nog niet heel veel gezien In de features.

00:38:05.750 --> 00:38:06.040
Anouk Wolters
OK.

00:38:08.520 --> 00:38:08.960
Data Engineer
Ja.

00:38:10.320 --> 00:38:11.830
Data Engineer
Het is, Het is ook iets.

00:38:12.790 --> 00:38:43.040
Data Engineer
Iets waar ik wel gewoon het begin met de data scientists over heb gehad van hé, want Dat is. Dat is één van de mogelijke manieren om te testen waar ik aan dacht is oké, je kan direct testen op de data die je dus eerst had, maar je kan ook zeggen, Hey ik, Ik heb bepaalde eisen Voor bepaalde features qua distributie of gemiddelde, of dat in ieder geval zoveel van bepaalde waarde in moet zitten. En Als je dat als test gebruikt. Maar dat was niet per se.

00:38:44.030 --> 00:38:50.740
Data Engineer
Iets wat ze nodig vonden, dus dan, ja, voor mij, Het is niet aan mij om dan verder te beslissen van Dat is waar de feature aan moet voldoen.

00:38:53.450 --> 00:38:55.650
Anouk Wolters
Nee snap ik en denk je dat er?

00:38:58.200 --> 00:39:00.340
Anouk Wolters
Stel, er wordt zoiets niet gedetecteerd.

00:39:01.890 --> 00:39:04.440
Anouk Wolters
Wat zou er dan mogelijk impact daarvan kunnen zijn?

00:39:08.760 --> 00:39:09.670
Data Engineer
Hé.

00:39:14.220 --> 00:39:16.820
Data Engineer
Ja een mogelijke impact.

00:39:18.250 --> 00:39:24.510
Data Engineer
Het kan Natuurlijk zijn dat je model dan net iets iets schever gaat werken, richting een bepaalde feature toe.

00:39:25.930 --> 00:39:26.860
Data Engineer
Dus stel.

00:39:29.170 --> 00:39:30.940
Data Engineer
Normaal gesproken duurt het.

00:39:31.690 --> 00:39:43.950
Data Engineer
Nou, stel je hebt een keertje die berekent hoe lang nadat iemand klant is geworden, doen ze een transactie. Als het gemiddelde daarvan in plaats van het 3 dagen opeens.

00:39:46.520 --> 00:40:07.050
Data Engineer
25 dagen is, dat je Misschien die Mensen die die ja 25 Natuurlijk best wel rustig aan en en Als je dan bij iemand kijkt die na een dag al een transactie doet dan ga je Misschien eerder denken dat je dat diegene een risicovolle klant is, omdat is geld heel snel verplaatst.

00:40:05.300 --> 00:40:05.680
Anouk Wolters
Ja.

00:40:07.780 --> 00:40:10.140
Data Engineer
Hè, dus Dat is iets wat in theorie zou kunnen gebeuren.

00:40:11.920 --> 00:40:33.620
Data Engineer
Dus ja ik, Ik weet niet hoeveel kwaliteit verder kwaliteitschecks Er zijn. Bij bij team ML van feature drift en dergelijke, maar stel zoiets zou ja, want ik bedoel, Ik weet ook niet hoe lang ze wachten met kijken wat we echt drift is. Dus wat voor thresholds daar zit is het één of twee maanden, 3 maanden half jaar, dat weet ik niet.

00:40:18.280 --> 00:40:18.680
Anouk Wolters
Oké.

00:40:35.280 --> 00:40:41.540
Data Engineer
En dus, maar stel dat wordt niet gedetecteerd. Dat is dan denk ik theoretisch gezien, iets wat ik zou kunnen gebeuren.

00:40:42.300 --> 00:40:44.160
Anouk Wolters
Ja ok snap ik ook.

00:40:45.150 --> 00:40:45.750
Anouk Wolters
En kijk.

00:40:50.590 --> 00:40:51.320
Anouk Wolters
Een.

00:40:59.020 --> 00:41:00.210
Anouk Wolters
Wat is uiteindelijk?

00:41:00.980 --> 00:41:03.020
Anouk Wolters
Wie is uiteindelijk verantwoordelijk voor.

00:41:04.630 --> 00:41:07.030
Anouk Wolters
De features Store en het onderhoud daarvan.

00:41:08.200 --> 00:41:11.490
Data Engineer
Dat is dus het MLOps team, dus Team ML.

00:41:14.080 --> 00:41:20.030
Data Engineer
Ja, zij zijn verantwoordelijk voor het onderhouden van de features Store. Er is nog wel, Er zijn wel gesprekken.

00:41:20.760 --> 00:41:40.320
Data Engineer
Dat is allemaal nieuw is en Er zijn ook ja de de soms ja, zij bouwen de features In de toekomst zelf niet. Wij doen Natuurlijk nu de migratie, Omdat het een soort van hé jongens jullie hebben allemaal de pipelines, die moeten allemaal gemigreerd worden In de toekomst. Nieuwe modellen worden gewoon direct In de features store gebouwd worden door de data Scientists.

00:41:42.420 --> 00:41:46.650
Data Engineer
Ja, dus je kan je nog afvragen wat het In de toekomst gaat betekenen.

00:41:47.400 --> 00:41:50.980
Data Engineer
Ja qua ownership, maar in principe is dat dit momenteel Team ML.

00:41:51.830 --> 00:41:52.410
Anouk Wolters
Ja oké.

00:41:53.290 --> 00:41:54.020
Anouk Wolters
Duidelijk.

00:41:58.780 --> 00:41:59.620
Anouk Wolters
Even kijken.

00:42:05.090 --> 00:42:08.960
Anouk Wolters
En, heb je in je werk de laatste maanden bepaalde.

00:42:09.670 --> 00:42:14.150
Anouk Wolters
Bepaalde processen gebruikt of workflows die binnen de bank worden gebruikt.

00:42:20.030 --> 00:42:20.740
Data Engineer
Nee.

00:42:23.650 --> 00:42:26.850
Data Engineer
Nee, eigenlijk niet, Er is. Er is een niet echt iets wat opvalt.

00:42:28.480 --> 00:42:38.930
Data Engineer
Ja want het team het [data science team] zelf gebruikt bijvoorbeeld geen bepaalde methodiek qua scrum of sprints of iets dergelijks .

00:42:39.550 --> 00:42:45.220
Data Engineer
En dus Het is gewoon aan mijzelf om een bepaalde methode te bedenken en daar aan.

00:42:46.880 --> 00:42:47.830
Data Engineer
Aan te houden.

00:42:49.400 --> 00:42:57.350
Data Engineer
Ik deed de migratie samen met een van de nieuwe data Scientistsat bij de bank, dus Samen hebben we gewoon iets bedacht wat voor ons werkt.

00:42:58.110 --> 00:43:04.530
Data Engineer
Daar hebben we ons aan gehouden. Nu gaat dat dus wel verandering. Ga daar wat verandering in komen. Met de nieuwe scrum master en het nieuwe team.

00:43:05.200 --> 00:43:06.000
Data Engineer
En.

00:43:07.050 --> 00:43:09.750
Data Engineer
Maar In de laatste maanden, niet.

00:43:10.290 --> 00:43:24.330
Anouk Wolters
Oké en voor het migreren weet jij dan vanuit je achtergrond welke stappen je daarvoor moet doorlopen of heb je dat heel erg on job bedacht of daar achter gekomen? Hoe wat er allemaal voor nodig is?

00:43:26.990 --> 00:43:51.290
Data Engineer
Nou ja, Toen ik begreep wat het betekent om een feature te bouwen In de features Store. Welke bronnen hoe je bij je bronnen kan en welke bronnen de legacy pipeline gebruikte? Ja, had ik niet per se heel veel meer nodig dan dat om het proces op te zetten. Ik heb nog wel een vraag hier en daar gesteld over mogelijke test libraries.

00:43:52.460 --> 00:43:55.880
Data Engineer
Ja, die heb ik dan ook gevonden, dus dan.

00:43:56.770 --> 00:44:00.580
Data Engineer
Ja, ja, dat ja verder. Verder ging het allemaal redelijk vanzelf.

00:44:01.190 --> 00:44:01.530
Anouk Wolters
Oké.

00:44:04.990 --> 00:44:08.100
Anouk Wolters
Even nog naar mijn vraag, Ik denk dat ik er wel bijna doorheen dan namelijk.

00:44:11.480 --> 00:44:12.340
Anouk Wolters
Had jij?

00:44:14.120 --> 00:44:20.790
Anouk Wolters
In jouw opdracht nog bepaalde kwaliteitseisen Quality Insurance waaraan je moest voldoen.

00:44:27.030 --> 00:44:27.650
Data Engineer
Ja.

00:44:31.980 --> 00:44:37.150
Data Engineer
Ik kan Alleen bedenken dat ik schone code moet schrijven die begrijpelijk is.

00:44:38.940 --> 00:44:39.430
Data Engineer
Want.

00:44:40.230 --> 00:45:09.470
Data Engineer
Als je bijvoorbeeld kijkt naar de Legacy Pipeline code ja, die is begrijpelijk binnen het team van data scientists, Omdat zij er Natuurlijk allemaal aan zitten, maar In de features store is dat een ander verhaal, want Iedereen leest dat en Iedereen moet dat in één keer kunnen begrijpen om te kunnen zeggen, oh, ik kan deze feature gebruiken of niet, ik hoef niet eenzelfde te gaan schrijven. Dat één ding verder ja wist ik een beetje vanuit mezelf. Als je gewoon zorgvuldig moet zijn.

00:45:10.300 --> 00:45:17.390
Data Engineer
Dus heb ik gewoon een aantal checks ingebouwd, is vooral testen was een belangrijk punt van met werken met.

00:45:19.490 --> 00:45:24.090
Data Engineer
version control om soort van retracability te hebben.

00:45:27.320 --> 00:45:34.140
Data Engineer
Ja dat ja dat zijn eigenlijk de voornaamste dingen, maar werd niet echt weer heel veel van mij verder gevraagd van.

00:45:34.480 --> 00:45:34.800
Anouk Wolters
OK.

00:45:36.000 --> 00:45:37.710
Data Engineer
Heeft het de kwaliteiten die we verwachten?

00:45:38.180 --> 00:45:44.130
Anouk Wolters
Ja oké en kun je wat meer vertellen over die version control nog wat, wat heb je precies geversioned?

00:45:44.940 --> 00:45:46.550
Data Engineer
O nou.

00:45:47.550 --> 00:45:50.420
Data Engineer
Ja wat ik zag, is dat het?

00:45:51.820 --> 00:46:21.830
Data Engineer
Daar werken ze in databricks en in database notebook schrijven is prima op zich, maar zodra die een beetje groot worden dan is het niet te doen. Dus sowieso ben ik gaan kijken naar een soort van middleware en Azure DevOps Git implementatie en wat we dan kunnen doen is eigenlijk een exel sheet makers van hé, dit zijn de 230 features die We hebben gebouwd en kolom met program een kolom tested een kolom met handover dus.

00:46:21.890 --> 00:46:23.460
Data Engineer
handover naar Team ML toe.

00:46:25.130 --> 00:46:45.680
Data Engineer
En wat we dan kunnen doen is eigenlijk per features branchen in Git en dus eigenlijk elke feature eigenlijk Los heeft geleefd ergens dat we heel erg precies kunnen teruggaan naar bepaalde states. Op een gegeven moment werden het wel features badges, want de 150 branches werd een beetje gek en.

00:46:46.480 --> 00:47:07.700
Data Engineer
Dus Dat is een beetje de structuur die hadden aangebracht. En ja, en dat werd wel gewaardeerd, want dat op die manier kan Iedereen ook Als ik zeg, maar morgen zijn van hé jongens, Dat is leuk, maar doei. Dat iemand anders precies kan kijken wat Ik heb gedaan en kan teruggaan naar een state waar het anders was waar het beter was.

00:47:08.830 --> 00:47:09.630
Data Engineer
Ja zoiets.

00:47:09.180 --> 00:47:09.440
Anouk Wolters
Ja.

00:47:10.390 --> 00:47:19.970
Anouk Wolters
Ja ja, en heb je verder nog een bepaalde keuze die je hebt gemaakt of aannames die heb je gedaan gedocumenteerd tijdens het hele proces.

00:47:21.210 --> 00:47:26.900
Data Engineer
Ja we, We hebben op een gegeven moment heeft ook een senior van het team.

00:47:28.910 --> 00:47:58.720
Data Engineer
Ja een beetje ondervonden dat het toch wel handig is om een beetje een soort van best practices Guide op te zetten. Dus Samen met wat andere Mensen die ook iets met de features store deden, hebben we een soort van Wiki page is geschreven waarin we alle

problemen die we tegen kwamen hebben gedefinieerd en de plek hebben gedefinieerd. Waar Mensen features die nog die wel bestaan of In de ether zijn, maar nog niet in productie zijn bij Team ML dus nog niet gepubliceerd zijn voor Iedereen dat dat ze daarvan weten, wie de owner daarvan is.

00:47:59.870 --> 00:48:06.100
Data Engineer
En Er is ook een aantal van die ja één van die best practices is dus die version control, zoals ik net zei.

00:48:07.610 --> 00:48:09.380
Data Engineer
Er zijn nog meer nog wat meer dingen.

00:48:11.340 --> 00:48:11.980
Anouk Wolters
Duidelijk.

00:48:15.480 --> 00:48:22.930
Anouk Wolters
Nou, Ik denk dat ik wel door mijn vragen heen ben. Denk je dat we nog iets hebben gemist In het interview of iets dat je zou willen toevoegen?

00:48:18.730 --> 00:48:19.110
Data Engineer
Oké.

00:48:26.790 --> 00:48:28.860
Data Engineer
Nee, niet per se. Ik denk dat we.

00:48:30.550 --> 00:48:33.090
Data Engineer
Ja, Ik denk dat we alles wel hebben gehad en.

00:48:33.760 --> 00:48:35.820
Data Engineer
Ja voor mij in ieder geval geen vragen nog.