|  |  |  |
| --- | --- | --- |
| Verslag tweede iteratie | December 9  2011 | |
| In dit verslag worden de vorderingen van het IKE project van groep 3 besproken. Dit verslag gaat in de op tweede iteratie. | |  |

Gijs Kuijer

Mathijs Lagendijk

Roeland Oosterloo

Inhoudsopgave

[Inleiding 2](#_Toc312153998)

[Programma van eisen 3](#_Toc312153999)

[Ontwerp 4](#_Toc312154000)

[Implementatie 6](#_Toc312154001)

[Evaluatie 8](#_Toc312154002)

[Conclusies en aanbevelingen 9](#_Toc312154003)

# Inleiding

Zoveel mensen, zoveel wensen. Deze spreuk gaat voor bijna alles op, zo ook voor muziek. Altijd al is er een enorme diversiteit aan muziek beschikbaar geweest. Vandaag de dag is dat niet anders. Wat wel is veranderd, is dat de muziek veel toegankelijker is geworden. Doordat bijna alle muziek online te luisteren en kopen is, is de keuze overweldigend geworden. Om iets van orde aan te brengen in deze chaos zijn we een systeem aan het maken dat advies geeft over mogelijk interessante muziek aan de hand van de input van de gebruiker.

De onderzoeksvraag om dit probleem op te lossen is als volgt geformuleerd: Wat is de beste manier om een advies te doen over muziek gegeven een voorkeur voor een bepaalde artiest?

Om dit probleem om te lossen moest om te beginnen kennis worden opgedaan over zogenaamde recommender systemen, systemen die een advies geven. Omdat we zelf niet alle informatie bezitten om iets te doen met een naam van een artiest zijn de beschikbare online datasources onderzocht om te bepalen of deze bruikbaar zijn voor ons onderzoek.

Een van de randvoorwaarde is dat we er van uit gaan dat de gebruiker geen super nieuwe artiesten zoekt. Omdat ons systeem getraind is op een dataset en daarna niet meer bij leert kent deze alleen de artiesten die hem geleerd zijn.

In dit rapport zal als eerst het programma van eisen worden besproken. Daarna komt het ontwerp, gevolgd door de implementatie en als laatst de conclusies en aanbevelingen.

# Programma van eisen

Dit project moet ervoor gaan zorgen dat ook mensen die weinig muziek kennis hebben en niet vaak bij concerten komen, ook nieuwe muziek kunnen vinden die zij leuk vinden. Aangezien mensen er niet van houden om hele pagina’s met vragen te gaan beantwoorden om vervolgens eindelijk een advies te krijgen, zal het product dat uit dit project naar voren komt ook simpel en duidelijk moeten zijn. Tevens zal er met weinig informatie van de gebruiker een hoop gedaan moeten worden, zodat de gebruiker altijd iets vindt wat hij of zij leuk vindt. Al deze kenmerken zijn opgenomen in het programma van eisen. Hieronder is een puntsgewijze lijst te vinden met eisen.

* Er moet een duidelijke interface zijn die makkelijk te bedienen is.
* De interface moet weinig informatie nodig hebben en toch goed kunnen functioneren.
* De interface moet snel zijn zodat mensen niet te lang hoeven te wachten en afhaken.
* De interface moet functionaliteit bieden die nu nog niet te vinden is op het web zodat mensen ook naar de website toe getrokken worden.
* De interface moet professionaliteit uitstralen waaruit naar voren komt dat er aandacht is besteed aan het bedenken van het geheel.
* Naast het advies moet er ook makkelijk meer informatie opgevraagd kunnen worden over de artiest, zodat er niet naar een andere website gegaan hoeft te worden om verder te zoeken.
* Ook oudere mensen moeten de website kunnen gebruiken, dit betekent dat er niet te gecompliceerde teksten en functies moeten zijn en alles in rustige stappen verloopt die ook voor ouderen nog te volgen zijn. Ook betekent dit dat de database waaruit advies gegeven wordt ook oudere muziek moet kunnen herkennen.

# Ontwerp

Bij het ontwerpen van ons recommender systeem hebben wij per iteratie een ontwerp van ons systeem gemaakt. Hierdoor hebben wij aan het einde van elke iteratie een volledig eindproduct wat het ontwerp van de desbetreffende iteratie weerspiegeld.

Voordat wij zijn begonnen met het ontwerpen van ons systeem hebben wij eerst overlegd hoe wij het systeem zouden gaan maken. Na veel overleg zijn wij tot de conclusie gekomen dat wij het beste een web applicatie kunnen bouwen die gebruikers via een website kunnen gebruiken. Natuurlijk hebben wij ook andere opties overwogen. Bij het maken van bijvoorbeeld een desktop applicatie vonden wij echter het nadeel dat deze applicatie dan eerst gedownload zou moeten worden en vervolgens zou moeten worden geinstalleerd voordat deze zou werken, terwijl de gebruiker juist snel muziek wil zoeken. Mede om deze reden denken wij dat een web applicatie de snelste manier is voor een gebruiker om muziek te ‘zoeken’.

In de eerste iteratie hebben wij nagedacht over hoe wij ons systeem visueel zouden weergeven en hoe wij data zouden gaan vergaren. Wij hebben besloten om ons systeem simpel, maar krachtig in elkaar te zetten. Wij hebben dit gedaan door een zoekmachine te ontwerpen die vergelijkbaar is met de bestaande zoekmachines zoals google en yahoo, maar in plaats van webpagina´s geeft ons systeem artiesten terug. Hier komt gelijk het tweede punt waar wij bij het ontwerp van de eerste iteratie over nagedacht hebben, namelijk wat wij nu eigenlijk gaan aanraden.

Wij hebben gekozen om artiesten terug te geven als zoekresultaten. De reden hiervoor is dat wij artiesten niet te specifiek, maar ook niet te algemeen vinden. In bijvoorbeeld een muziekgenre kan een groot bereik aan muziek zitten; er kunnen twee compleet verschillende artiesten uitkomen die hele andere muziek maken. Wanneer dit gebeurd is er natuurlijk geen sprake van een goed werkend recommender systeem. Aan de andere kant vonden wij de titels van individuele nummers te specifiek om als resultaat terug te geven. Wanneer wij nummers terug geven als zoekresultaat komt er een enorme dataset als resultaat terug. Hierdoor zal de zoekende gebruiker veel werk moeten verrichten om de data te doorkruisen. Het is dus veel gemakkelijker om artiesten als resultaat terug te geven en deze te doorzoeken. Artiesten zijn makkelijker te doorzoeken dan nummers, maar ook niet zo algemeen als een muziekgenre. Natuurlijk moet er uiteindelijk wel muziek gevonden worden, daarom hebben wij ons systeem zo ontworpen dat de albums van een uitgekozen artiest opgehaald kunnen worden en dat deze kunnen worden doorkruist zodat de gebruiker uiteindelijk toch muziek te horen krijgt.

Bij het ontwerpen hebben wij besloten om enkele bestaande muziekdatabases te gebruiken en deze aan elkaar te koppelen. Dat wil zeggen dat de informatie van de databases word samengevoegd waarbij de ene database de andere aanvult met data die nog niet is vergaard. Dit hebben wij gedaan omdat wij zoveel mogelijk informatie van een artiest willen hebben om deze in eerste instantie te kunnen gebruiken voor het produceren van recommendations en in tweede instantie om deze informatie aan de gebruiker weer te geven. Wij gebruiken de databases van LastFM en MusicBrainz omdat de deze makkelijk zijn aan te spreken en omdat zij een grote up to date database hebben. Naast de data die bovenstaande databases bevatten (artiestnamen, geboortedata, albums, nummer) willen wij ook dat de gevonden muziek af gespeeld kan worden binnen ons systeem. Gelukkig hebben wij hiervoor ook een bestaande database gevonden genaamd GrooveShark. Dit is één van de dingen waar wij in onze tweede iteratie mee bezig zijn geweest.

In de tweede iteratie lag de focus van ons ontwerp op het verstevigen van ons recommender systeem, daarnaast wilden wij ook de mogelijkheid creeren om gevonden muziek binnen ons systeem af te spelen. Onze eerste gedachte lag bij het trainen van een neuraal netwerk om op deze manier de goede resultaten terug te geven. Na het gesprek bij onze eerste iteratie, waar wij andere ideen hebben gehoord en waar het bleek dat een neuraal netwerk erg lastig zou gaan worden hebben wij besloten om eerst een andere manier te gebruiken en ons ontwerp bij te stellen.

De manier waarop wij nu ons recommender systeem hebben ontworpen is door het leggen van links tussen artiesten. Deze worden gelinkt aan de hand van de tags die zij hebben. Door de hoeveelheid tags die ze gemeen hebben word de sterkte van de link bepaald en bij het zoeken worden de artiesten met de sterkste links teruggegeven. Het neuraal netwerk is echter nog niet van de baan en wij willen dit alsnog in een latere interatie implementeren.

Het volgende punt waar ons ontwerp van de tweede iteratie verschilt van het ontwerp van de eerste iteratie is het afspelen van muziek. Bij het ontwerpen hebben wij besloten dat het toch erg handig is als de muziek gelijk kan worden afgespeeld. Hierdoor kan de gebruiker die aan het zoeken is de muziek direct beluisteren en belissen of hij/zij het leuk vind of dat hij/zij verder moet zoeken.

Het ontwerp van de derde iteratie is zodanig in elkaar gezet dat we de punten uit de voorgaande iteraties willen verstevigen. Zo is aan het ontwerp van de derde iteratie een rating systeem toegevoegd. Dit systeem willen wij gaan gebruiken om de sterkte van de eerdergenoemde links tussen artiesten (die wij gebruiken om recommendations te maken) te kunnen beinvloeden. Op deze manier kan ons systeem als het ware leren wat de juiste recommendations zijn en dus betere recommendation geven. Ook zijn op deze manier nieuwe artiesten veel makkelijker toe te voegen en te integreren in het systeem.

Zoals eerder gezegd willen wij het neurale netwerk toch proberen te integreren in het systeem. Wij hebben dit dan ook in het ontwerp van de derde iteratie gezet. In feite hebben wij nu al een neuraal netwerk door het gebruik van de links tussen de artiesten en de ratings die deze links kunnen beinvloeden. We willen echter toch proberen om het systeem te verbeteren door gebruik te maken van de technieken die wij bij de colleges geleerd hebben en daarom staat het (verbeteren van) het neurale netwerk bovenaan het lijstje in ons ontwerp van de derde iteratie.

# Implementatie

Bij de implementatie van het product moeten er op vele dingen gelet worden. Aangezien dit een product van enige omvang is, is de planning een zeer belangrijk punt waar men niet zomaar overheen kan stappen. Zo hebben ook wij een duidelijke planning over wat we in welke iteratie willen doen.

Zo hebben we in de eerste iteratie veel tijd besteed aan het online krijgen van het product en het zorgen dat we een basis recommender systeem hadden. Hierbij hebben we eerste gebruikt gemaakt van LastFM en het recommender systeem dat daarin gebouwd is.

In de tweede iteratie hebben we vooral tijd besteed aan het koppelen van Grooveshark aan ons systeem. Dit is een systeem waarmee elk nummer dat wij aanraden ook weer online beluisterd kan worden. Dit is een grote aanwinst voor ons systeem zodat mensen ook makkelijk kunnen luisteren of ze ons advies ook echt leuk vinden. Hierdoor kunnen we ook makkelijker erop vertrouwen dat de gebruikers ons gaan helpen het advies wat wij geven te beoordelen. Hoe en of we deze beoordelingsfunctie gaan maken is nog niet besloten, maar de mogelijkheden open houden is altijd goed. Wat we ook in de tweede iteratie hebben gedaan is het bouwen van een tabel waarin alle artiesten uitgezet zijn tegen alle artiesten en er op de connecties tussen de artiesten een bepaalde rating wordt neergezet. Op deze manier kunnen we makkelijk zoeken naar artiesten die bij andere artiesten passen. Hiermee krijgen we dan ons eigen recommender systeem. Helaas is deze tabel nog niet af aangezien het maken ervan afhing van een enorme dataset die lastig te doorlopen is.

Hoe we ons systeem hebben opgebouwd is niet heel makkelijk uit te leggen. De programmeertaal keuze was redelijk makkelijk. Aangezien we een recommender systeem moesten maken, wil je ook dat die makkelijk toegankelijk is voor gebruikers. En wat is er nu makkelijker dan het internet. Dus aangezien we het systeem op internet gingen bouwen is er gekozen voor PHP als achterliggend systeem om alle informatie op te halen, als voorkant is er HTML en Javascript gebruikt om een mooie dynamische website te maken. Om alles met elkaar te verbinden is AJAX gebruikt zodat de pagina’s nooit ververst hoeven worden. Dit alles is zo geprogrammeerd om de site zo snel mogelijk te laten werken en de gebruikers het makkelijk te maken om de site door te werken.

De achterkant van de site waar alle informatie wordt opgehaald is gebouwd met PHP en haalt de informatie op van MusicBrainz en LastFM door middel van JSON. Deze acties duren het langst van alle acties op de website. Dit komt door MusicBrainz en LastFM omdat deze redelijk traag zijn in het geven van informatie. Hier is dus helaas niks aan te doen.

Van de achterkant van de site gaat de informatie in JSON objecten naar de voorkant, waar Javascript dan door middel van AJAX de informatie ophaalt en de HTML instuurt en alles zichtbaar maakt. Dit zorgt voor een totaal dynamische website waar geen enkele pagina ververst hoeft te worden.

Het recommender systeem wat we nu nog gebruiken is een systeem van LastFM waar je voor een bepaald genre de beste artiesten kan opvragen. Dit werkt nog niet super goed en vandaar dat we zelf zijn gaan bouwen aan een recommender systeem. Dit wilde we doen door middel van een tabel die artiesten aan andere artiesten linkt. Deze tabel maken gaat vanzelf lukken maar hiervoor is een goed algoritme nodig. Het huidige algoritme moet dus nog een beetje aangepast worden zodat het maken van een tabel wel goed gaat lukken. Een eerste resultaat van het algoritme is namelijk nog niet uitgekomen op een correct resultaat. Om deze tabel te maken gebruiken we Java. Omdat het niet op internet hoeft te gebeuren heeft dit de voorkeur omdat het dan vele malen sneller gaat. We slaan deze tabel dan op in een tekst bestand, en deze kan PHP dan weer uitlezen. De methodes om de tabel uit te lezen zijn al klaar en maken gebruik van Object Oriented PHP ,zoals alle PHP in de website, en lezen de tabel al in aan het begin van het gebruik zodat de toegang naar de tabel sneller wordt op het moment dat er ook daadwerkelijk artiesten opgezocht moeten worden.

In de volgende iteratie van het project plannen we om de tabel en het recommender systeem volledig werkend te krijgen. Ook willen we voor extra informatie gaan zoeken op blogs, twitter en facebook. Dit zorgt er dan voor dat we wat meer meningen bij de adviezen kunnen laten zien.

Uiteindelijk krijgen we dus een recommender systeem dat via PHP alle informatie ophaalt. Waarbij adviezen verstrekt worden op basis van een tabel die is samengesteld uit een flinke dataset. Verder wordt er meta informatie opgehaald van LastFM en MusicBrainz zodat we ook meer kunnen afbeelden dan alleen de artiest naam. Dit alles wordt dan verwerkt door Javascript met Ajax en afgebeeld op de site zonder dat er een pagina ververst hoeft te worden. Dit alles zorgt voor een redelijk snelle site die makkelijk toegankelijk is voor gebruikers via het internet.

# Project evaluatie

Tot nu toe loopt het project redelijk. Het gaat nog niet super vlot en snel. Dit heeft meerdere redenen. Één van de redenen is zeker dat het project nieuw is en we nog heel erg op ontdekkingstour zijn over het hoe en wat bij het project.

Een tweede is zeker dat het vakgebied waarover het project nog in ontwikkeling is. Het interactief advies geven via websites is iets dat nog niet zo lang bestaat en er moet dus ook nog een perfecte manier gevonden worden om het te bewerkstelligen. Hier doen we natuurlijk onze uiterste best voor, maar op een dusdanig kort tijdbestek is het lastig dit te realiseren.

Een derde factor die er zeker voor zorgt dat het nog niet heel soepel loopt is waarschijnlijk dat er naast het project nog Algoritmiek loopt waar ook zeker meer dan 10 uur tijd in steekt per 2 weken zodat er weer minder tijd overblijft om na project tijd nog even verder te gaan en de finishing touch op dingen te geven. We hopen natuurlijk dat dit positief omslaat en ons tijd geeft om nog meer te bereiken in het project.

Ondanks deze drie factoren moeten we toch zeggen dat we redelijk positief kunnen zijn van het resultaat tot nu toe. Er staat een zeer nette website met nu nog een basis recommender systeem. We zijn echter zeer dichtbij een oplossing voor een geavanceerd recommender systeem waar nog niet alle opties van bekend zijn. In volgende iteraties is dit nog een prachtig onderwerp om verder op te bouwen.

# Conclusies en aanbevelingen

Deze iteratie heeft ons zeker weer wat mooie nieuwe functionaliteiten gebracht. Het zorgt ervoor dat we weer met goede moed richting de volgende iteratie kunnen gaan, waarbij we de laatste eindjes aan ons systeem kunnen knopen en nog verder kunnen gaan om de bronnen van het internet te gaan verkennen. We kunnen de conclusie trekken dat deze iteratie netjes is verlopen, al een stuk minder chaotisch dan de eerste, en dat ook de organisatie op de project middagen al beter verloopt waardoor er meer effectief werk verzet kan worden.

Voor volgende iteratie is het de bedoeling om binnen de eerste week toe te werken naar een geavanceerd recommender systeem. Waarna we doorgaan met het toevoegen van extra meta informatie aan de website. Dit kan gaan gebeuren door middel van Facebook of twitter of dergelijke andere media.