

Casus opdracht “video-on-demand platform FLETNIX”

Version: 7-11-2017

Inleiding

In dit document is het ontwerp te vinden op grond waarvan het eindproduct voor de beroepstaak “het maken en beheren van een kleine client/server database” kan worden gerealiseerd. Het eindproduct bestaat uit een geïmplementeerde relationele database en een aantal gerealiseerde rapporten voor eindgebruikers.

Er is een aantal opdrachten geformuleerd die moeten worden uitgevoerd om tot het eindproduct te komen.

De gehele casus is een groepsopdracht. Groepen mogen uit maximaal uit 2 personen bestaan.

Casusbeschrijving

Deze casus gaat over de nieuwe video-on-demand internetplatform "FLETNIX", een concurrent van Netflix¹. Tot nu toe waren de founders van het platform volledig gericht op het realiseren van de streaming functionaliteit van het platform. De rest van de functionaliteit bestaat nog niet. Omdat de applicatie binnenkort live moet gaan, hebben zij jouw hulp nodig voor de ontwikkeling en implementatie van het vereiste datamodel en voor het verkrijgen van management informatie middels nog te realiseren rapportages.

De films zijn georganiseerd in genres, ook de directors en de belangrijkste cast van elke film zijn bekend. Films kunnen deel uitmaken van een filmreeks (bijvoorbeeld *Matrix* en *Matrix Reloaded*). Geregistreerde gebruikers betalen een tarief om films online te mogen kijken. Dit tarief verschilt per film.

Om films aan te mogen bieden, betalen de platform managers maandelijks vaste licentiekosten aan de copyright-houders. Uiteraard is de CEO geïnteresseerd in een hoge opbrengst. Dit betekent dat hij zoveel populaire films als mogelijk aan wil bieden. Films die niet vaak gekeken worden veroorzaken meer kosten dan opbrengst. De CEO bekijkt dus de statistieken over de gekozen films op een maandelijks basis en beslist welke films blijven en welke films eruit gaan.

De webapplicatie waarop de applicatiebeheerder filmgegevens kan wijzigen, een financial manager filmprijzen kan aanpassen en facturen kan genereren en waarop gebruikers online films kunnen bestellen en bekijken maakt geen deel uit van deze opdracht. Dat leer je in de course WebTech.

Use Case beschrijvingen

In bijlage A vind je alleen de *Use Case* beschrijvingen voor de management rapportages. In een *Use Case* beschrijving wordt ieder proces tot in detail gespecificeerd. Dit is onderdeel van een *functioneel ontwerp* (wat moet de applicatie doen).

Logisch Relationeel Schema

In bijlage B vind je een *Logisch Relationeel Schema*. Dit hoort bij het *technisch ontwerp* (hoe moet de applicatie de dingen doen en opslaan).

¹ www.netflix.com

Opdrachten

Hieronder vind je opdrachten waarin je in stappen de database voor het FLETNIX platform maakt en daarbij een applicatie voor eindgebruikers ontwikkelt.

Opdracht 1: Maak database

- A. Implementeer de database die in het Logisch Relationeel Schema (LRS) is beschreven. Stel CREATE TABLE statements op voor alle tabellen die in het LRS zijn beschreven en voer deze uit.

**Bewaar de scripts van opdracht 1A tot en met 1D in een bestand genaamd
OPDRACHT1_DDL.SQL**

- B. Probeer de volgende constraints (business rules) te definiëren in de CREATE TABLE statements:
- Het geslacht (gender) van een persoon (Person) moet 'M' (male) of 'F' (female) zijn.
 - Het publicatie jaar (publication_year) van een film (Movie) moet tussen 1890 en het huidige jaar zijn.
 - De aanmelddatum (subscription_start) van een klant (Customer) mag niet na de datum van afmelding (subscription_end) zijn.
 - De gebruikersnaam (user_name) van een klant (Customer) moet uniek zijn.
 - De datum (watch_date) waarop een film (Movie) bekeken is moet tussen de aanmelddatum (subscription_start) en afmelddatum (subscription_end) van de klant (Customer) liggen.

Als het niet mogelijk is om een constraint te definiëren, geef dan aan waarom niet.

- C. Probeer nog minimaal 3 voor de hand liggende (en dus realistische) constraints te vinden, beschrijf deze en realiseer deze indien mogelijk.
- D. Pas de DDL-statements aan zodat ook de update- en deleteregels (ON UPDATE, ON DELETE) worden geïmplementeerd.
- E. Test alle constraints die in opgave B en C gecreëerd zijn. Toon door middel van INSERT-statements voor het toevoegen van voorbeeldpopulatie (goede populatie en tegenvoorbeelden) aan dat de geïmplementeerde constraints uit opgave B en C correct geïmplementeerd zijn.

**Bewaar dit script met goede populatie en tegenpopulatie als
OPDRACHT1_TESTSCRIPT.SQL**

- F. Leg minimaal de volgende voorbeeldpopulatie (zie voor een begin ook bijlage B) aan:
- Alle records voor de movies "Matrix" en "Matrix Reloaded". Gebruik de YouTube link naar de trailer van de films als URL. Tabellen: Genre, Movie, Person, Movie_Cast (3 acteurs), Movie_Genre.
 - Minimaal 5 andere movies (waarvan minimaal 2 met cast en directors). Zorg ervoor dat alle movies verschillende prijzen hebben. Tabellen: Genre, Movie, Person, Movie_Cast, Movie_Directors, Movie_Genre
 - 10 Customers met watchhistories. De customers hebben alle 'The Matrix' in de afgelopen week gezien. Verder heeft elke customer nog 3 andere movies in de laatste twee weken gezien. Tabellen: Customer, Watchhistory

Bewaar dit script met voorbeeldpopulatie als OPDRACHT1_INSERTSCRIPT.SQL

Opdracht 2: Breid het bestand van films uit

De internationale movie database ([IMDB](#)) bevat informatie over (bijna) alle films die ooit zijn geproduceerd. De informatie wordt ook digitaal ter beschikking gesteld. Op Onderwijs Online vind je een backup van de data die afkomstig is van IMDB.

Het is de bedoeling data deze gegevens in jouw database terecht komt. Doorloop hiervoor **Digitale Workshop 4 “Installeren van de IMDB database in SQL Server 2016 Developer”**.

Het is verstandig om eerst de IMDB data te analyseren en een plan te bedenken welke gegevens naar welke tabellen gekopieerd moeten worden.

Zorg van tevoren dat de hele FLETNIX database leeg is en daarna met een script de gegevens in de juiste tabellen terecht komen. Hier en daar moet je ervoor zorgen dat informatie die in jouw model bestaat, maar niet in de IMDB database, op een goede manier aangevuld wordt (e.g. de omschrijving van een film; hier kun je bijvoorbeeld de tekst “Description of ” gecombineerd met de naam van de film gebruiken). Hier en daar moet je misschien ook uitgaan van bepaalde standaardwaardes (prijs per movie). Dit wordt ook *data conversion* genoemd. Voorzie je DML script van gepast commentaar.

Zorg dat je later gemakkelijk kunt switchen tussen de voorbeeldpopulatie uit opdracht 1 F en deze IMDB populatie! Je hoeft niet beide populaties tegelijkertijd in de database te hebben.

- A. Maak de delete statements die de gehele FLETIX database in de juiste volgorde leegmaakt.
- B. Schrijf het conversiescript voor de Parent-tabellen (domein-tabellen) van FLETNIX
 - Person
 - Movie
 - Customer
 - Genre
- C. Schrijf het conversiescript voor het vullen van de Child-tabellen.

**Bewaar dit script met queries als
OPDRACHT2_CONVERSIESCRIPT.SQL**

Opdracht 3: Maak queries voor het genereren van overzichten

Maak SQL Select-queries voor onderstaande informatiebehoeften. Dit dient twee doelen. Op deze manier leer je de database goed kennen. Een deel van de queries kun je straks als basis voor de realisatie van rapportages gebruiken. Maak queries voor de volgende informatiebehoefte; toon daarbij de kolommen die tussen de blokhaken staan:

- A. Alle films gesorteerd naar genre [movie title, publication year, genre]
- B. Alle movies die tussen 1990 en 2010 geproduceerd zijn.
- C. Alle klanten die op dit moment actief zijn (i.e. subscription_end moet leeg zijn) [customer lastname, firstname, subscription_start]
- D. De cast uit alle Terminator movies uit het jaar 1991 [id, title, firstname, lastname, role]
- E. Alle movies waarin de acteur "Arnold Schwarzenegger" een rol speelt [movie title, publication year]
- F. Alle gebruikers met openstaande kosten [Customer lastname, firstname, total price]
Maak een View voor deze informatiebehoefte.
- G. Toon 100 movies die tot nu toe het minst bekeken zijn, gesorteerd naar het aantal keren dat ze gekeken werden. Dit houdt ook 0 keer in [movie title, number of times watched].
Maak een View voor deze informatiebehoefte.
- H. Alle movies die in de afgelopen twee maanden het meest bekeken zijn, gesorteerd naar het aantal keren dat ze gekeken werden. Toon alleen movies die minimaal één keer bekeken zijn [movie title, publication_year, number of times watched].
Maak een View voor deze informatiebehoefte.
De sortering kun je niet binnen de view doen, laat die buiten de view.
- I. Alle movies die meer dan 8 genres hebben [title, publication_year]
- J. Alle vrouwen die in *Horror* movies en *Family* movies gespeeld hebben [firstname, lastname].
- K. De director die tot nu toe de meeste films geproduceerd heeft [firstname, lastname].
- L. Alle Genres en het percentage dat de films uit het bepaalde genre uitmaken t.o.v. het totale aantal films [genre, percentage], gesorteerd op meest populaire genre.
Maak een View voor deze informatiebehoefte. Je mag ook eerst één of meerdere (hulp-)views maken om de informatiebehoefte op te lossen.
- M. Gebruikers [mail_adress] en het gemiddelde aantal films die elke gebruiker per dag kijkt. Toon alleen gebruikers die gemiddeld 2 of meer films per dag kijken, met het grootste gemiddelde bovenaan.
Maak een View voor deze informatiebehoefte. Je mag ook eerst één of meerdere (hulp-)views maken om de informatiebehoefte op te lossen.

Bewaar dit script met queries en views als OPDRACHT3_SELECTSCRIPT.SQL
--

Opdracht 4: Breid het bestaande model uit

De founders van het platform willen graag onder meer van movies bij gaan houden welke awards er zijn gewonnen. Daarvoor hebben ze onderstaand overzicht ontworpen. Dit overzicht genaamd 'Overview movie awards per year' vind je hieronder.

Ieder jaar (*year*) worden deze *awards* uitgereikt op een andere locatie. Daarvan wordt alleen de *country* vastgelegd. Personen die aan de productie van deze film hebben meegewerkt kunnen genomineerd worden en dus ook kans maken om te award te winnen voor diverse *categories*. Onderstaande ontwerp bevat tussen haakjes de identificatie van de *movie* en de *person*.

Overview movie awards per year

Academy Awards 2000 in The Netherlands

Best Film Editing	The Matrix (207992)	Zach Staenberg (550813)	W	Won
	The Insider (160492)	William Goldenberg (276884)	N	Nominated
		Paul Rubell (950001)	N	Nominated
		David Rosenbloom (68042)	N	Nominated
Best Sound Effects Editing	The sixt sense (303564)	Andrew Mondschein (54796)	N	Nominated
	Fight club (112290)	Ren Klyce (351397)	N	Nominated
		Richard Hymns (950002)	N	Nominated
	The Matrix (207992)	Dane Davis (950003)	W	Won
Best Visual Effects	Star Wars: Episode I - the phantom menace (313474)	Ben Burt (165453)	N	Nominated
	Star Wars: Episode I - the phantom menace (313474)	John Knoll (351981)	N	Nominated
		Rob Coleman (191712)	N	Nominated
	The Matrix (207992)	John Gaeta (950004)	W	Won
		Janek Sirrs (540853)	W	Won
		Steve Courtley (950005)	W	Won
		Jon Thum (950006)	W	Won

BAFTA Awards 2000 in New Zealand

Best Sound	The Matrix (207992)	David Lee (950007)	W	Won
		John Reitz (950008)	W	Won
		Gregg Rudloff (950009)	W	Won
		David Campbell (950010)	W	Won
		Dane Davis (950011)	W	Won
		Ben Burt (165453)	N	Nominated
Best Cinematography	The Matrix (207992)	Tom Bellfort (950012)	N	Nominated
		Bob Beemer (950013)	N	Nominated
		Bill Pope (950014)	N	Nominated
		Conrad Hall (950015)	W	Won
Best Production Design	The Matrix (207992)	Owen Paterson (464716)	N	Nominated
	Sleepy Hollow (304862)	Rick Heinrichs (303143)	W	Won
Best Editing	The Matrix (207992)	Zach Staenberg (550813)	N	Nominated
	American Beauty (13789)	Tariq Anwar (950016)	W	Won
		Christopher Greenbury (283753)	W	Won

- A) De opdracht is uiteindelijk om een model in 3NV te maken om bovenstaande informatie in op te kunnen slaan.
Stel op basis van dit overzicht eerst 1 tabel op in 0NV waarin alle gegevens uit het overzicht voorkomen.
Pas vervolgens de normalisatiestappen toe en geef duidelijk de tussenstappen in dit proces aan (dus zowel de structuur in 0NV, 1NV, 2NV en in 3NV weergeven met daarin duidelijk alle functionele afhankelijkheden opgenomen).

Hint: het model in 1NV zal een tabel bevatten waarbij de sleutel de kolommen Award, Year, Category, Movie (movie id) en Person (person id) betreft. Controleer of je dat zo hebt!

De FLETNIX database moet worden uitgebreid met het model dat in 3NV ontworpen is. Hiervoor is het noodzakelijk om aan te sluiten bij het bestaande model. Er is immers al een Person, een Country en een Movie tabel. Pas, indien nodig, het model in 3NV aan zodat het aansluit bij de tabellen die er al zijn. Noem dit model 3NV_aangepast. Mogelijk moet je hiervoor identificaties (primary keys) en relaties (foreign keys) wijzigen. Check hiervoor ook de bestaande populatie van de database.

Bewaar de uitwerkingen (0NV, 1NV, 2NV, 3NV en 3NV_aangepast) in één document genaamd OPDRACHT4_NORMALISATIE.DOCX (of PDF)
--

- B) Implementeer het model (dus het aangepast 3NV model) in SQL Server door het maken van een DDL script.
- C) Populeer de nieuwe database met de ontbrekende data conform deze twee overzichten.

Bewaar dit script met DDL en INSERTs als OPDRACHT4_DDL.SQL

Opdracht 5: Ontwerp Management-rapportages

Voor de FLETNIX database dienen een aantal rapportages gerealiseerd te worden, o.a. om te meten hoe het met het bedrijf FLETNIX er voor staat.

Voordat gerealiseerd gaat worden moet van iedere rapportages echter eerst een schets gemaakt worden zodat concreter wordt waar het rapport aan moet voldoen.

Laat de ontwerpen van feedback voorzien door de docent voordat je verder gaat met opdracht 5.

- A) Maak een ontwerp van het rapport dat de use case, zoals beschreven in bijlage A, ondersteunt. Je mag je ontwerp deels baseren op de tabel in opdracht 4 maar zorg dat de use case ondersteund wordt.
- B) Voor het management van FLETNIX zijn onder andere onderstaande KPI's interessant:
1. De omzet die over een bepaalde periode (jaar – maand) gemaakt wordt (betaald versus nog niet betaald). Men wil die informatie graag per land zien maar ook per genre zodat duidelijk wordt of men meer budget aan reclame (in bepaalde landen) moet besteden en of men meer films van een bepaald genre moet inkopen.
 2. De geografische spreiding van klanten en inzicht in de abonnementsvormen per land. Men wil die informatie hebben zodat inzichtelijk wordt welke regio's de afgelopen jaren sterk in opkomst zijn. Je mag ervan uitgaan dat de abonnementsvormen niet veranderen (we houden immers geen historie bij).
 3. Het aantal customers per abonnementsvorm dat in een periode (jaar) lid is geworden versus het aantal dat het abonnement heeft opgezegd. Men wil gevoel krijgen of FLETNIX nog steeds groeiende is, constant blijft of dat het aantal gebruikers juist afneemt. Men hoopt op een jaarlijkse groei van 6%. Het zou mooi zijn als het inzichtelijk gemaakt kan worden of die groei bereikt is/wordt.
 4. Men heeft het gevoel dat de groei van abonnees in de leeftijdscategorie 50 - 75 jarigen achter blijft ten opzichte van de leeftijdscategorie 25 - 50. Men wil dit gevoel kunnen toetsen door één of meerdere overzichten te krijgen.

Ontwerp voor elk van deze KPI's / informatiebehoeften een rapport waarbij de KPI overzichtelijk getoond wordt.

Geef voor elk te realiseren rapport:

- een schets met een indeling van de informatie, baseer daarbij zoveel mogelijk op de bestaande gegevens
- de visualisatie vorm (zoals beschikbaar in Microsoft Power BI)
 - i. Zorg dat je daarbij minstens 3 verschillende visualisatievormen gebruikt (map, table, pie, stacked column etc.)
- de te gebruiken SQL query/queries
- een toelichting van de aanwezige informatie-elementen, geef dus ook aan hoe eventuele totalen berekend worden en hoe de gegevens gegroepeerd zijn.

Let op: er wordt hier slechts het ontwerp gevraagd, realiseer het dus nog niet. Gebruik voor dit ontwerp een ontwerp- / tekentool.

Bewaar alle ontwerpen in één document genaamd OPDRACHT5_ONTWERPEN.DOCX (of PDF)
--

Opdracht 6: Bouw Management-rapportages

Bouw de rapportages uit opdracht 5 in Microsoft Power BI volgens de in opdracht 5 gemaakte ontwerpen.

Het is mogelijk dat je tijdens de realisatie bedenkt dat je van de specificaties wilt afwijken. Maak in dat geval een aantekening op de specificatie en pas alle specificaties aan het eind aan om ze in overeenstemming te brengen met het gerealiseerde rapport.

Denk ook aan de afwerking: zorg voor nette rapport titels en structuur.

Maak van de gerealiseerde rapporten schermafdrucken en plaats deze in een Word document. Geef daarbij duidelijk aan wat welk rapport is.

Bewaar de schermafdrucken van de rapporten genaamd OPDRACHT6_RAPPORTEN.DOCX
--

Lever ook de sourcefiles van de gerealiseerde rapporten in.

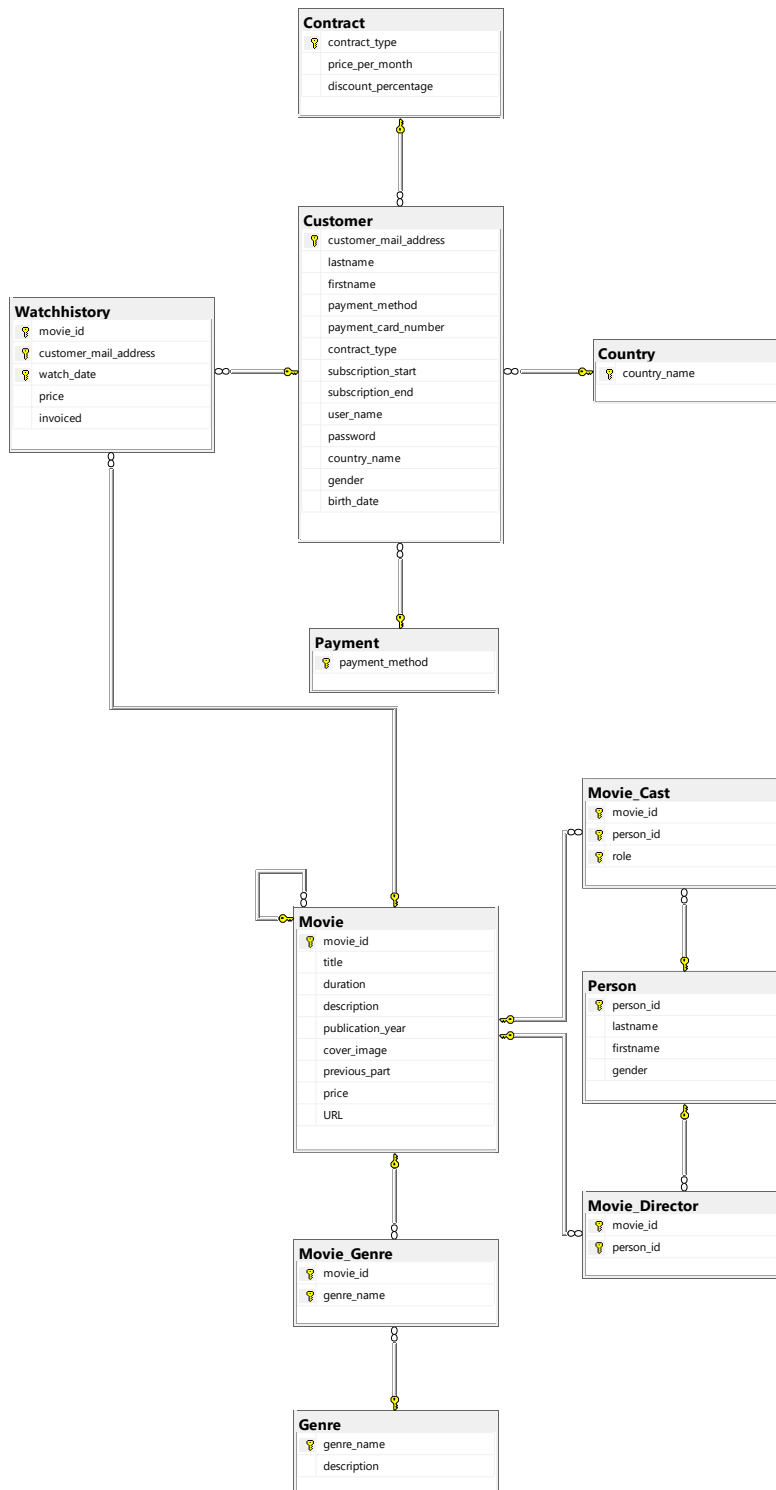
Bewaar de sourcefiles van de rapporten genaamd OPDRACHT6_RAPPORTEN.PBIX
--

BIJLAGE A: Use Case

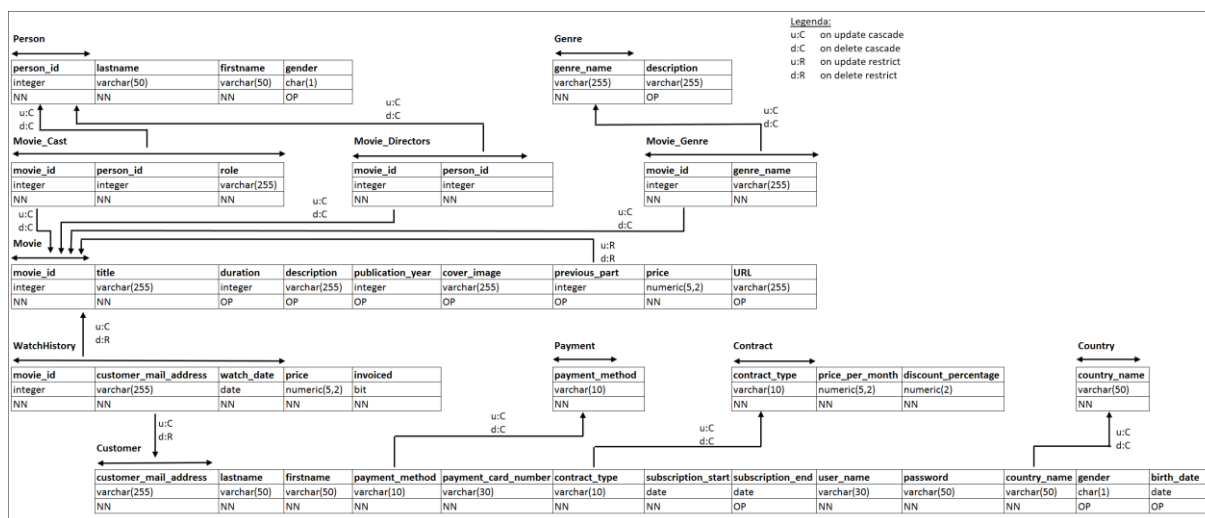
Use Case **Bekijken Overzicht nominaties en winnaars per award**

Use Case: Bekijken Overzicht nominaties en winnaars per award	
Purpose: Samenstellen en afdrukken rapport met nominaties en winnaars	
Description of use case: Inzicht verschaffen in goed beoordeelde films	
Primary actor: CEO (Chief Executive Officer) Secondary actor: -	
Stakeholders and interests: -	
Preconditions: CEO is ingelogd	
Postconditions (Success Guarantee): gewenste overzichten zijn afgedrukt	
Basic Flow (Main Success Scenario)	
Actor action	System responsibility
1. CEO kiest ervoor om Overzicht nominaties en winnaars per award te bekijken	2. Systeem toont een lijst met jaartallen
3. CEO kiest een jaartal	4. Systeem toont een overzicht met de in dat jaar uitgebrachte films met de nominaties en winnaars per fim. Ook totaalstellingen van aantal nominaties en winnaars per film worden getoond.
Alternative flows	
A3. CEO kiest geen jaartal (of alle jaartallen)	5. Systeem toont een overzicht met per jaar de uitgebrachte films met de nominaties en winnaars per film.

BIJLAGE B1: Relationeel Schema – zoals getoond in SQL Server



BIJLAGE B2: Relatieve Schema – als strokendiagram



Tabellen met voorbeeldpopulatie en betekenis (op type niveau)

Tabel Person

Voorbeeldpopulatie:

person_id	lastname	firstname	gender
83616	Wachowski	Andy	M
83617	Wachowski	Larry	M
393411	Reeves	Keanu	M
729933	Moss	Carrie-Anne	F
151786	Fishburne	Laurence	M

Betekenis:

“There is a person with person id <person_id>”

“Person <person_id> has firstname <firstname>”

“Person <person_id> has lastname <lastname>”

“Person <person_id> has the gender <gender>”

Tabel Movie

Voorbeeldpopulatie:

movie_id	title	duration	description	publication_year	cover_image	previous_part	price	URL
11	Matrix, The	122	Description of Matrix, The	1999	NULL	NULL	2.50	http ...
12	Matrix Reloaded, The	106	Description of Matrix Reloaded, The	2003	NULL	11	2.50	http ...
13	Matrix Revolutions, The	103	Description of Matrix Revolutions, The	2003	NULL	12	2.50	http ...

Betekenis:

“There is a movie with movie id <movie_id>”

“Movie <movie_id> has the title <title>”

“Movie <movie_id> has a duration of <duration> minutes”

“Movie <movie_id> has the description <description>”

“Movie <movie_id> is published in year <publication_year>”

“The cover image of movie <movie_id> is <cover_image>”

“Movie <movie_id> is the successor of movie <previous_part>”

“The standard price of movie <movie_id> is <price> euro”

“Movie <movie_id> can be watched at <URL>”

Tabel Movie_Directors

Voorbeeldpopulatie:

movie_id	person_id
11	83616
11	83617
12	83616
12	83617

Betekenis:

"Movie <movie_id> is directed by <person_id>"

Tabel Movie_Cast

Voorbeeldpopulatie:

movie_id	person_id	role
11	393411	Neo
11	729933	Trinity
11	151786	Morpheus

Betekenis:

"Person <person_id> plays the role of <role> in movie <movie_id>"

Tabel Genre

Voorbeeldpopulatie:

genre_name	description
Action	Creative works characterized by emphasis on exciting action sequences
Drama	Fictional division between comedy and tragedy
Family	Movies suitable for a wide range of age groups
Sci-Fi	Fictional movies dealing with imaginative content such as futuristic settings, futuristic science and technology, space travel, time travel, parallel universes, and extraterrestrial life

Betekenis:

"There exists a genre with genre name <genre_name>"

"Genre <genre_name> has description <description>"

Tabel Movie_Genre

Voorbeeldpopulatie:

movie_id	genre_name
11	Action
11	Sci-Fi

Betekenis:

"Movie <movie_id> is categorized as a(n) <genre_name> movie"

Tabel Country

Voorbeeldpopulatie:

country_name
The Netherlands
Belgium

Betekenis:

“There is a country named <country_name>.”

Tabel Payment

Voorbeeldpopulatie:

payment_method
Mastercard
Visa
Amex

Betekenis:

“<payment_method> is a valid payment method.”

Tabel Contract

Voorbeeldpopulatie:

contract_type	price_per_month	discount_percentage
Basic	4	0
Premium	5	20
Pro	6	40

Betekenis:

“There is a <contract_type> contract.”

“Customers with a <contract_type> contract pay a monthly fee of <price_per_month> euro.”

“Customers with a <contract_type> contract get <discount_percentage> percent discount on the standard price of a movie.”

Tabel Customer

Voorbeeldpopulatie:

customer_mail_address	last_name	first_name	payment_method	payment_card_number	contract_type	subscription_start
peter@pan.de	Pan	Peter	Mastercard	0123456789	Basic	01-oct-2014
guy@monk.nl	Threepwood	Guybrush	Visa	333-555-777	Premium	01-jan-2014

subscription_end	user_name	password	country_name	gender	birth_date
28-oct-2014	pp123	peterpass	The Netherlands	M	26-jun-1971
NULL	gm4ever	guypass	Belgium	M	03-mar-1972

Betekenis:

"There is a customer with mail address <customer_mail_address>"
"The customer with mail address <customer_mail_address> has last name <last_name>"
"The customer with mail address <customer_mail_address> has first name <first_name>"
"The customer with mail address <customer_mail_address> pays with <payment_method>"
"The customer with mail address <customer_mail_address> pays with card number <payment_card_number>"
"The customer with mail address <customer_mail_address> has a <contract_type> contract"
"The customer with mail address <customer_mail_address> is subscribed from <subscription_start>"
"The customer with mail address <customer_mail_address> is subscribed until <subscription_end>"
"The customer with mail address <customer_mail_address> can log in with user name <user_name>"
"The customer with mail address <customer_mail_address> can log in with password <password>"
"The customer with mail address <customer_mail_address> lives in <country_name>"
"The customer with mail address <customer_mail_address> has the gender <gender>"
"The customer with mail address <customer_mail_address> is born at <birth_date>"

Tabel Watchhistory

Voorbeeldpopulatie:

movie_id	customer_mail_address	watch_date	price	invoiced
11	peter@pan.de	01-oct-2014	2.50	1
11	peter@pan.de	02-oct-2014	2.50	0
12	guy@monk.nl	10-mar-2014	2.00	0

Betekenis:

"Movie <movie_id> has been watched by customer <customer_mail_address> at <watch_date>"
"The price of movie <movie_id> watched by customer <customer_mail_address> at <watch_date> was <price> at that time"
"The invoice indicator of movie <movie_id> watched by customer <customer_mail_address> at <watch_date> is <invoiced>"

*) invoiced: 1 means the invoice has been paid, 0 means the invoice is not paid (yet).