Logotipo

Descripción generada automáticamente

**UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA**

*La Universidad Católica de Loja*

**ÁREA TÉCNICA**

**INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN**

**Proyecto Integrador 2021**

**Asignatura:**  Fundamento de Bases de Datos

**Integrantes:**  Adrián Alessandro Rivera Cueva

**Docente:**  Ing. Nelson Oswaldo Piedra Pullaguari

**Octubre 2021 – Febrero 2022**

## Introducción

La normalización de una base de datos (BB DD) es fundamental para que la misma sea consistente y reaccione de la mejor manera ante anomalías lógicas. Por eso para el proyecto de Practicum 1.1 (Proyecto Integrador) en lo que corresponde a Fundamentos de Bases de Datos, hemos procedido a normalizar la BB DD, a partir de la tabla universal que se nos entregó, e identificar las dependencias funcionales para facilitar dicho proceso.

La tabla universal se nos la entregó en un archivo CSV, en el cual se pueden almacenar información de cualquier tipo, prácticamente como su fuera una tabla enorme, y estos datos son entendidos por un sin número de programas como por ejemplo Excel o cualquier procesador de hojas de cálculo, y también en un bloc de notas, aunque aquí los datos no se van a ver tan legibles, más si se tiene una cantidad astronómica de data. En dicho CSV lo introducimos en Excel para el análisis de los campos que contenía y que información llevaba. Luego de este proceso se puede decir que comenzamos formalmente el proceso de identificación de demencias funcionales, de tablas y el proceso de normalización de la Base de Datos.

## Desarrollo

Como ya es lo dijo en la Introducción en primer lugar debemos hallar las dependencias funcionales, para eso extraemos todos los campos que existen en el archivo entregado, de los cuales salieron los siguientes campos: index, budget, genres, homepage, id, keywords, original\_language, original\_title, overview, popularity, production\_companies, production\_countries, release\_date, revenue, runtime, spoken\_languages, status, tagline, title, vote\_average, vote\_count, cast, crew, director. Luego de obtenidos los campos, hicimos un desglose en una tabla de todos los atributos y sus características.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Movie | | | | | | | | |
| 0FN | index | budget | genres | homepage | **id** | keywords | original\_language | original\_title | overview |
|  | popularity | production\_companies | production\_countries | release\_date | revenue | runtime | spoken\_languages | status | tagline |
|  | title | vote\_average | vote\_count | cast | crew | director |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Entity | Property | Dominio | Optional | MultiValuated | Type | Comment |
| Movie | Id | Int(11) | Not Null | False | Primary Key |  |
| Movie | Index | Varchar(5) | Not Null | False |  |  |
| Movie | Budget | Varchar(12) | Not Null | False |  |  |
| Movie | Genres | Varchar (75) | Not Null | True |  |  |
| Movie | Homepage | Varchar (150) |  | False |  |  |
| Movie | Keyword | Varchar (100) |  | True |  | Palabras claves que identifican la película |
| Movie | Original\_Language | Varchar (2) | Not Null | False |  |  |
| Movie | Original\_Title | Varchar (75) | Not Null | False |  |  |
| Movie | Overview | Varchar (150) |  | False |  |  |
| Movie | Popularity | Double (25,6) | Not Null | False |  |  |
| Movie | Release\_Date | Date | Not Null | False |  |  |
| Movie | Revenue | Int (10) | Not Null | False |  |  |
| Movie | Runtime | Double (4,1) | Not Null | False |  | Tiempo de la película en minutos. |
| Movie | Status | Varchar (8) | Not Null | False |  |  |
| Movie | Tagline | Varchar (100) |  | False |  |  |
| Movie | Title | Varchar (75) | Not Null | False |  |  |
| Movie | Vote\_Average | Double (2,1) | Not Null | False |  |  |
| Movie | Vote\_Count | Int (10) | Not Null | False |  |  |
| Movie | Cast | Varchar (125) | Not Null | False |  |  |
| Movie | Director | Varchar(30) |  | False |  |  |
| Production Company | Id | Int(11) | Not Null | False | Primary Key |  |
| Production Company | Name | Varchar (12) | Not Null | False |  |  |
| Production Country | Iso\_3166\_1 | Varchar (2) | Not Null | False | Primary Key |  |
| Production Country | Name | Varchar (12) | Not Null | False |  |  |
| Spoken Languages | Iso\_639\_1 | Varchar (2) | Not Null | False | Primary Key |  |
| Spoken Languages | Name | Varchar (12) | Not Null | False |  |  |
| Crew | Id | Int(11) | Not Null | False | Primary Key |  |
| Crew | Credit\_Id | Varchar (25) | Not Null | False | Primary Key |  |
| Crew | Name | Varchar (20) | Not Null | False |  |  |
| Crew | Gender | Varchar (1) | Not Null | False |  | Género, 0 es desconocido, 1 mujer y 2 hombre |
| Crew | Job | Varchar (30) | Not Null | False |  |  |
| Crew | Department | Varchar (10) | Not Null | False | Foreign Key |  |

**Primera Forma normal**

Como Production Company, Production Country, Spoken Languages y Crew son campos compuestos se los separo cada uno de sus campos para cumplir la primera forma normal y también se separa en tuplas ya que estos atributos también a su vez son campos multivaluados. Por ende, en una entidad separada.

Como cast y genres son campos multivaluados se separa de en tuplas separadas, sin perder la relación con la clave primaria.

La tabla tendría un aspecto así:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Movie** | | | | | | | | | | | | | | | |
| index | budget | homepage | **id** | keywords | original\_language | original\_title | overview | popularity | release\_date | revenue | runtime | status | tagline | title | director |
| **Movie** | | | | | | | | | | | | | | | |
| vote\_average | vote\_count | cast | **idCompany** | name | **iso\_3166\_1** | name | **iso\_639\_1** | name | idPerson | **creditId** | name | gender | job | deparment |  |
|  |  |  | **production\_companies** | | **production\_countries** | | **spoken\_languages** | | **crew** | | | | | |  |

**Segunda Forma Normal**

Se identifica las dependencias funcionales completas para que ya no existan dependencias funcionales parciales.

***Movie***

IdMovie → Index, Budget, Homepage, Original\_Language, Original\_Title, Overview, Popularity, Release\_Date, Revenue, Runtime, Status, Tagline, Title, Vote\_Average, Vote\_Count, Director,

***Company***

IdCompany → Name

***Country***

Iso\_3166\_1 → Name

***Languages***

Iso\_639\_1 → Name

***Credit***

Credit\_Id → idPerson, name, genre, Job, Department (Existe Dependencia Transitiva)

Credit\_Id → idPerson, Job, Department

***Person***

idPerson→ Name, Gender

***Actor***

idActor 🡪 Name

**Claves Candidatas que determinan una tabla al completo**

IdMovie, Iso\_3166\_1 → **Production\_Countries**

IdMovie, Iso\_639\_1 → **SpokenLanguages**

IdMovie, Id\_company → **Production\_Companies**

IdMovie, creditId → **Crew**

idMovie, idActor 🡪 **Cast**

En la búsqueda de dependencias pudimos identificar en total once entidades principales, las cuales disponen de sus respectivos atributos y llaves primarias.

**Resultado de haber aplicado la segunda forma normal en la base de datos.**

Aquí se separa las dependencias parciales que teníamos en la primera forma normal y se crea las tablas con las dependencias funcionales encontradas en la parte anterior

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Movie | | | | | | | | | | | | | |
| 2FN | index | budget | homepage | **id** | keywords | original\_language | original\_title | overview | popularity | release\_date | revenue | runtime | status |
|  | tagline | title | vote\_average | vote\_count | cast | director |  |  |  |  |  |  |  | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | | |
|  | **companies** | |  | **countries** | |  | **languages** | |  | **crew** | | | | | |
|  | **id** | name |  | **iso\_3166\_1** | name |  | **iso\_639\_1** | name |  | **creditId** | idPerson | name | gender | | |
|  |  |  |  |  |  |  |  |  |  | job | deparment |  |  | | |
|  | **production\_companies** | |  | **production\_countries** | |  | **spoken\_languages** | |  |  |  |  |  | | |
|  | **idMovie** | **idCompany** |  | **iso\_3166\_1** | **idMovie** |  | **iso\_639\_1** | **idMovie** |  |  |  |  |  | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | | |
|  | **Genres** | |  | **Genres\_Movies** | |  | Actor | |  |  |  |  |  | | |
|  | **idGenre** | **name** |  | **idGenre** | **idMovie** |  | **idActor** | name |  |  |  |  |  | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | | |
|  | Cast | |  |  |  |  |  |  |  |  |  |  |  | | |
|  | **idMovie** | **idActor** |  |  |  |  |  |  |  |  |  |  |  | | |

**Resultado de haber aplicado la tercera forma normal en la base de datos.**

Aquí solo se separa la depencia transitiva que había en la tabla que antes se llamaba crew ahora credits. Los datos que dependían de idPerson se puso en otra tabla llamada person. Por ultimo se cambio el nombre de la tabla movies\_crew por crew, estos cambios de nombre se dan porque después de normalizar se analizó que esos nombres eran los más sensatos para lo que representa.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Movie | | | | | | | | | | | | | | | | | | | | | | | | | | |
| index | budget | | homepage | | **id** | | keywords | | original\_language | | original\_title | | overview | | popularity | | release\_date | | revenue | | runtime | | | status | |
| tagline | title | | vote\_average | | vote\_count | | cast | | director | |  | |  | |  | |  | |  | | |  | | |  | | |
|  | |  | |  | |  | |  | |  | |  | |  | |  | |  | |  | | |  | | |  |
| **companies** | | | |  | | **countries** | | | |  | | **languages** | | | |  | | **credits** | | | | | | | | |
| **id** | | name | |  | | **iso\_3166\_1** | | name | |  | | **iso\_639\_1** | | name | |  | | **creditId** | | idCrew | | | deparment | | | job |
|  | |  | |  | |  | |  | |  | |  | |  | |  | |  | |  | | |  | | |  |
| **production\_companies** | | | |  | | **production\_countries** | | | |  | | **spoken\_languages** | | | |  | | **Person** | | | | | | | |  |
| **idMovie** | | **idCompany** | |  | | **iso\_3166\_1** | | **idMovie** | |  | | **iso\_639\_1** | | **idMovie** | |  | | **idPerson** | | name | | | gender | | |  |
|  | |  | |  | |  | |  | |  | |  | |  | |  | |  | |  | | |  | | |  |
| **Genres** | | | |  | | **Genres\_Movies** | | | |  | |  | |  | |  | | **Crew** | | | | |  | | |  |
| **idGenre** | | **name** | |  | | **idGenre** | | **idMovie** | |  | |  | |  | |  | | **creditId** | | **idMovie** | | |  | | |  |

Todo esto lo podemos ver plasmado en el siguiente esquema entidad-relación:

**Esquema entidad-relación**

Gráfico, Diagrama

Descripción generada automáticamente

Una vez obtenido este esquema, revisamos si todas las entidades tenían dependencias totales y además que sus campos sean valores atómicos, una vez revisado todo lo anterior pudimos decir que la BB DD estaba en Primera Forma Normal (1FN), por ende, también en Segunda Forma Norma (2FN) y por último en Tercera Forma Normal (3FN). Una vez finalizado el proceso de Normalización procedimos hacer el esquema relacional.

**Esquema Relacional**

Diagrama

Descripción generada automáticamente

**Estrategias de introducción de datos.**

* En primer lugar, cargamos el CSV en la base de datos
* Una vez que tenemos la tabla general vamos separando los campos compuestos y multivaluados en tablas aparte.
* Para la separación de estos campos de usa tablas temporales donde se aloja los atributos descompuestos con el idMovie.
* Luego se separó la dependencia parcial en una tabla y luego se creó la tabla de relación con movie que contiene las llaves primarias de las dos tablas

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Modelo Físico de las Tablas

Imagen que contiene Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza baja Imagen que contiene Escala de tiempo

Descripción generada automáticamente

Escala de tiempo

Descripción generada automáticamente con confianza media Texto

Descripción generada automáticamente con confianza baja

Tabla

Descripción generada automáticamente con confianza media Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Texto, Tabla

Descripción generada automáticamenteTexto, Tabla

Descripción generada automáticamente

**Consultas a la Base de Datos**

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## Conclusión

Para concluir el presente informe, debemos decir que el proceso de Normalización realizado por mi persona me ha parecido satisfactorio, ya que he podido desglosar todos los atributos que se encontraba en la tabla universal que a su vez se encontraba almacenada en un archivo CSV, fuimos encontrando sus entidades y las dependencias funcionales. Hay que decir que el encontrar en primer lugar las dependencias funcionales facilito mucho el proceso de normalización de la base de datos. Y una vez normalizada la BB DD procedimos a realizar el esquema E/R, en donde pulimos algunos detalles para finalmente realizar el esquema relacional.

Finalmente hicimos un arduo trabajo de limpieza de datos, el cual nos extrajo mucho esfuerzo para lograr tener los datos que eran textos legibles y los JSON formateados para que las funciones las puedan leer, con todo esto realizado es en donde se puede decir que ha concluido nuestro trabajo, que nos ha dejado mucha sabiduría para el resto de nuestra vida profesional como ingenieros en computación.