Mustafa A. Hakkoz - 150117509 23.12.2019

Muhammed Adem E. Demirkuş - 150115003

**DATABASE SYSTEMS PROJECT STEP-5**

**CHANGE REQUEST**

* **Subject**

This project is a business applicable database which is designed to represent **an** **online TV/movie streaming** **service** like Amazon Prime Video, Netflix, Hulu etc.

 ***CANNAL | Premium Streaming Service***

“Cannal” is an online streaming service offering live and on-demand TV and movies, with affordable prices for home sets and mobile devices. Along with his licensed library and on-growing originals, Cannal offers extensive streaming experience by virtue of highly customizable add-ons and features.

* **Summary**

For the first step we decided on subject and scope of the project firstly. Then we determined problem statements, system constraints and use cases, to have an overall design on tables and attributes. We stated all of thought process in our step-1 report with details.

And for second step, we designed tables, attributes and relationships of database and drew an entity-relationship diagram.

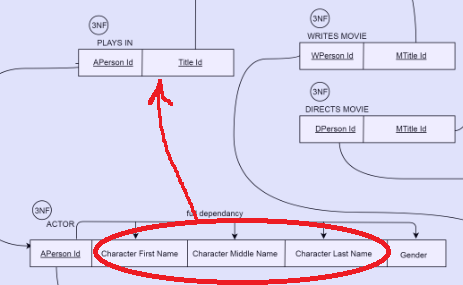
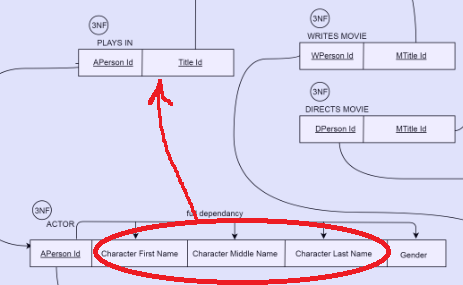
And before implementing Relational Model for third step, we changed some attributes and tables of step-2 to get a more improved structure. We submitted a change request form for step-3 with our relational model.

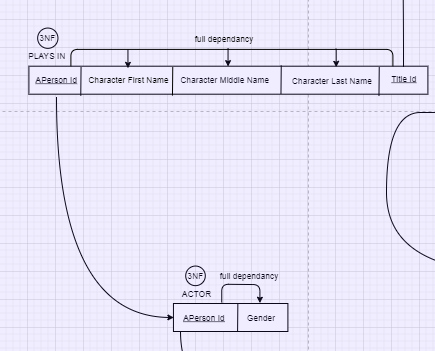
For step-4 we created a physical database in SQL Server. Some features of step-4 were missing at that phase and now at step-5, we’re submitting a more general change request form for step-4 and step-5.

* Updated version of ERD can be found as “step5-UpdatedERD.png” in our step-5 submission folder.
* Updated version of RD can be found as “step5-UpdatedRD.png” in our step-5 submission folder.
* Updated version of physical database can be found as “step5-UpdatedDB.sql” in our step-5 submission folder.
* **Objective**

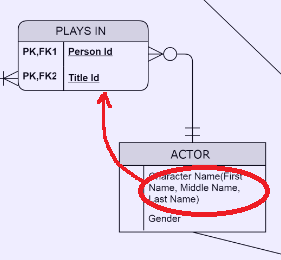
Our changes include some corrections for previous steps and improvements after step-4 database implementations. As correction, we fixed and added some of forgotten attributes. And as improvement we removed unnecessary attributes and added some new features. We also insert new data to physical database get enough population for web application. All of them are explained below.

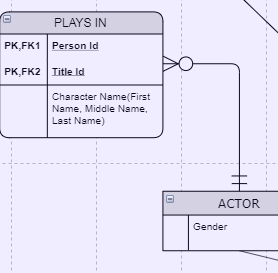
* **Diagram Corrections for Step-5**
  + **Actor relation in Catalog section:** We moved the attributes “Character First Name”, “Character Middle Name” and “Character Last Name” from the relation “Actor” to the relation “Plays In”.

****

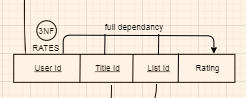
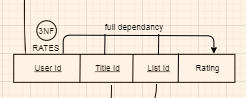
****

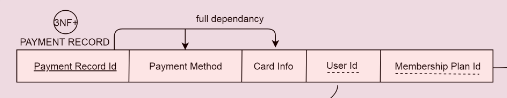
**We also updated it in ERD:**

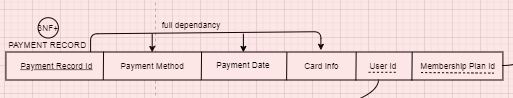
****

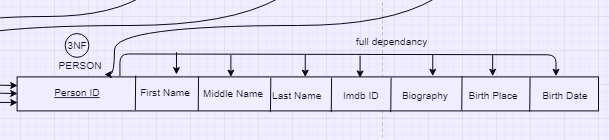
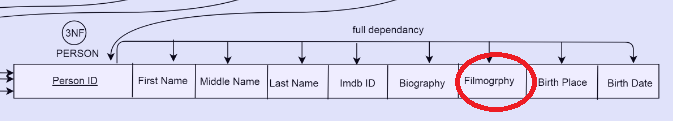
****

* + **Rates relation in User Interface section:** We had an attribute “Rating” in the relation “Rates” in ERD but we forgot to put it in RD. So we updated RD like below:

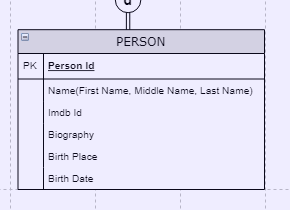
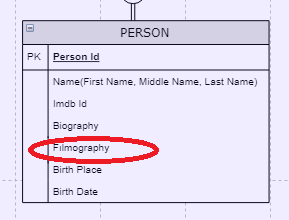
****

* + **Payment Record relation in Membership section:** We had an attribute “Payment Date” in the relation “Payment Record” in ERD but we forgot to put it in RD. So we updated RD like below:

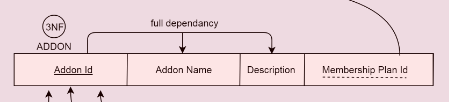
****

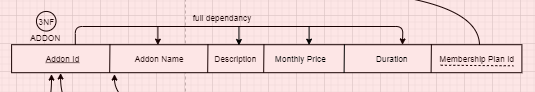
* **Diagram Improvements for Step-5**
* **Person relation in Catalog section:** We removed the attribute “Filmography” of the relation “Person” since we can retrieve previous titles of a certain person by using views in step5.

**We also updated it in ERD:**

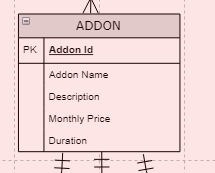


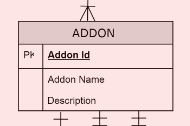
* + **Addon relation in Membership section:** We figured out when we were implementing physical database that we should add two attributes “Monthly Price” and “Duration” to the relation “Addon” just like “Membership Plan”. So we updated both RD and ERD:

****



**We also updated it in ERD:**





* **SQL changes for step-5**
  + **Data Inserton in Catalog section:** We insert data to all of unpopulated tables in the section “Catalog” (all tables in this section besides “Title”) to implement web application with enough data at step-5.

* + **Fixing the Trigger “InsertAmount”:** Before fixing it, local variable “@id” was assigned to “userId” so when we are testing it in step-4 demo, it didn’t update more than one Payment for a specific user. So we changed it by assigning to “PaymentRecordId” instead of “userId”.
  + **UserName in the Table “\_User”:** We made the attribute “UserName” unique.