**Transformation**

The ER-model needs to be converted to a relational database schema. So that it can be used in SQLite. So below is a summary of how this conversion can be done.

1. Tables need to be created for all entities.
2. So DDL-commands `CREATE TABLE` can be used to make the tables for lower higher-level and lower-level entities.
3. Add the corresponding attributes to their respective tables.
4. Add primary key constraints in each table. In our ER-model the next attributes are primary keys (they are chosen based on the fact that they have unique values): GameID, SystemID, DSID, PublisherID, SubfranchiseName, ConsoleLineID. There is one composite Key added consisting of UserID and Function. To make an disjointed isa-hierarchy possible.
5. Add foreign key constraints based on cardinality ratios and the specified constraints in the ER-model.
6. DDL-command `NOT NULL` is used for all attributes, since for our ER-model each column of all tables needs to be not empty.
7. Rename each primary key and foreign key with the general notation ( PK_ *primary-key-name* and FK_*foreign-key-name*).

**Posteriori normalisation**

Our current system won't give feedback when entering conflicting data. According to our ER some entries are mandatory, but this isn't reflected in our DDL SQLite. There can for example be multiple presidents having worked for a development studio, but there is not a constraint for the faulty data of multiple presidents working at the same time. You actually want to have a degree constraint in the ER-model to set a limit, but that can not really be added, since then there must be a relation between DevelopmentStudio and the children tables of the isa-hierarchy (President, GameDesigner and DevelopmentDirector). Regarding the relational database schema, it has the same problem as well.

In real life there could also be multiple development studios working on the same game and often development directors are linked towards development studios and games directed, which are not reflected in our ER diagrams and our code as well, but the world of game distribution and development is unfortunately for our database too complex to be represented for this assignment.

Due to its use as a reference for a foreign key. The column called 'Function' looks pretty useless in the 3 tables of our ISA-hierarchy since all the entries have the same value for the function (For example all entries in the president table have the value: Function = 'President').

**Normal Form Violation:**

- The 'Function' column of the children tables of the ISA-hierarchy is an Atomic one, since the attribute within each of the entities has only one possible value.
- Not every constraint is a consequence of domain and key constraints (For example the earlier described concurrent president problem).