

Step 3 : Project Proposal, Outline, ERD

Reviewed Draft 2 - Design HTML Interface

+ DML SQL

CS 340 Group 126

San Davran, Young-Joon Park

Feedback by the peer reviewers draft 2:

- **Review by Sangtawun Miller**

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?
 - a. Yes, the schema presents a model that follows the data outline + ER diagram.
- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?
 - a. Yes, there is consistency in naming across the document
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?
 - a. Yes, the schema is clear and easy to read (no crossing lines)
- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?
 - a. Yes, the Performances intersection table has two FKs, the Order_items intersection table has two FKs
- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?
 - a. I did not catch any non-normalized issues, but I think that the credit_card_num column under Orders could potentially have repeatable data because the same customer would be more likely than not to use the same credit card
- Is the SQL file syntactically correct? This can be easily verified by using PHPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)
 - a. The SQL file is syntactically correct
- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?
 - a. Yes, my only comment is that credit_card_num could be INT instead of VARCHAR if we do not plan to include any characters in the sample data
- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

- a. Yes, the PKs and FKs are correctly defined and appropriate CASCADE operations are declared
- In the SQL, are relationship tables present when compared to the ERD/Schema?
 - a. Yes, relationship tables are present
- In the SQL, is all example data shown in the PDF INSERTED?
 - a. Yes, all example data is shown (sample data in all the entity tables + intersection tables; I thought it was a nice touch that they included clean tables alongside the SQL output)
- Is the SQL well structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?
 - a. The SQL is well structured but I'd recommend adding more comments throughout the document

• Review by Tony Le

- Schema follows the database outline exactly.
- There is consistency in naming convention, ie entities are plural, attributes are not.
- Schema is clear and easy to read.
- Intersection table correctly formed to facilitate M:N relationships.
- Sample data looks correct, no normalization problems.
- SQL file is correct, was able to add to my database fine.
- Data types are appropriate considering descriptions of attributes
- Primary and foreign keys are correctly defined
- Relationship tables are present in Schema
- Example data are inserted using INSERT statements
- Yes the sql is well structured and well-commented

• Review by Miles Wedemeyer

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

Yes. Looks good!

- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Yes these all look okay to me! Plurality, capitalization, naming are all consistent

- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

yes. There is no criss-crossing

- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

Yes

- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

The tables do not seem to have any normalization issues

- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

Yes. Ran on my end!

- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

Yes. Maybe things like cell phone number can be slightly optimized (like use varchar(12) instead of varchar(50)) but everything looks pretty good to me

- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

Yup!

- In the SQL, are relationship tables present when compared to the ERD/Schema?

Yes.

- In the SQL, is all example data shown in the PDF INSERTED?

yes.

- Is the SQL well structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?

Looks well structured. I personally think the amount of commenting is fine but you probably want to add some more for the grading rubric. It's pretty light on comments

● Review by Bashar Akkary

- Yes, the schema matches both the ER diagram and the database outline.
- There is consistency of naming throughout the different parts of the draft, with correct pluralization/singularization for entities and attributes, and correct use of capitalization.
- Yes, the schema is easy to read
- The two intersection tables present are properly formed.
- There are no normalization issues as far as I can tell
- The SQL is syntactically correct
- Yes and yes.
- Yes
- Yes
- The SQL is well structured, I would personally add more comments though.
- Overall, awesome work!

Overall, the feedback from the reviewers and the TA indicated that there were no problems with our draft. A few comments added, but no significant change otherwise.

Feedback by the peer reviewers draft 1:

Notes left by team members are written in blue.

Changes to be applied have been changed to red.

1. Review by Kaitlyn Hornbuckle

1. I really enjoy the interstellar-style of this database. I am a little **confused on what problem** the database is solving - is it to make transactions easier for their customers or just to have available on the owner's website?

2. The overview has some numerical facts, such as, "The database currently has around 200 most popular movies." However, I think including the number of current customers that come can be helpful to give the reader an idea of what the size of the database will be. - **Rough estimate of customers was included in the original draft**

3. Most entities can be stored in a list. However, I would suggest making all their **names plural** since there will be multiple of them. Also, instead of having Performance, it might be a good idea to convert that to a rating property in Movies instead.

- **performance is there to convert the M:M relationship between Actor and Movie to two 1:M relationship.**

4. The outline is very helpful, but the reasons for each relationship is unclear and needs more explanation.

5. Since it's unclear what relationships each is for, I can't tell which relationships are correctly formatted. Also, it doesn't look like there is a M:M relationship.

6. All entities **need to be plural**.

2. Review by Sean Madden

• **Does the overview describe what problem is to be solved by a website with DB back end?**

Yes, the overview describes that Gorak, an entrepreneurial Gorkian, is looking to profit from the trend of human movies from Earth becoming popular among Gorkian high society. Due to Gorkian laws on xeno-preservation, direct contact between Gorkians and humans is prohibited, so Gorak's business model involves infiltrator agents fulfilling orders for human movies through a galaxy-wide website that uses a database to store information on titles, prices, stock, and actors.

• **Does the overview list specific facts?**

Yes, the overview lists specific facts such as the popularity of human movies among Gorkian high society, the laws on xeno-preservation in Gorkia, Gorak's business model of using infiltrator agents to fulfill orders for human movies through a website, and the fact that the database currently has around 200 popular movies but is growing every week. Additionally, Gorak estimates that his customer base will number in the 10,000s due to the expenses involved in transporting the original blu-ray back to Gorkia Prime.

·Are at least four entities described and does each one represent a single idea to be stored as a list?

Yeah, there are definitely at least four entities described in that information. Each one represents a single idea that could be stored in a list. However, I did notice a couple of minor issues. For example, in the `Order_item` table, the `order_id` and `movie_id` are both set as `auto_increment` and `primary keys`, but I think the `order_id` should just be a foreign key since it is already a primary key in the `Order` table. Also, in the `Performance` table, the `actor_id` is set as a foreign key, but there is no `actor_id` attribute in the `Actor` table, it's `actor_first` and `actor_last`. Other than that, it looks good to me.

- the two FKs are composite keys that make the PK of `Order_item`. There also is an actor ID so we are not sure if Madden just missed that it was there.

·Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?

Yeah, the outline of entity details seems to be thorough. It explains what each entity represents and lists all of their attributes, including their datatypes and constraints. It also describes the relationships between the different entities, like how a customer can have multiple orders or how a movie can be related to multiple performances.

One minor thing I noticed is that in the "Order" entity, the "ship_or_pickup" attribute is listed as a BIT datatype, but it's not clear what that represents exactly. It might be helpful to add a brief explanation of what that attribute is used for. Also, in the "Performance" entity, it's not clear if the "actor_id" is a primary key or a foreign key, it would be better if the primary key is written as PK and foreign key is written as FK.

- BIT datatype was there for boolean as a boolean data type does not exist in sql

- A foreign key can be a primary key as well. The `actor_id` and `movie_id` form a composite key, hence they are both FK and PK.

·Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

So, the relationships between the different entities in the database design look good to me. For instance, there's a clear one-to-many relationship between an order and the employee handling it - meaning one employee can handle multiple orders, but each order can only be handled by one employee.

Additionally, there are a couple of many-to-many relationships in the design. [Like, the relationship between movies and orders - which is connected through an "order_item" table. And then there's also a relationship between movies and actors, which is connected through the "performance" table.](#) Overall, I think the ERD presents a logical view of how all the different entities in the database are related to each other. It lists all the entities and their relationships in a way that makes sense, in my opinion

- This review noted that M:M was handled by the 2 entities.

·Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

The naming consistency is maintained throughout the overview, entities, and attributes. Capitalization is used appropriately for naming the entities, such as "Movie", "Actor", "Order", and "Customer". However, it's worth noting that the [entities are not plural in their naming.](#)

Suggestions for improvement

I noticed that the Order_item table has both the order_id and movie_id set as auto-increment and primary keys. It might make things a bit clearer if the order_id was just a foreign key, since it's already a primary key in the Order table. Also, to make the data stored even more easy to understand, it might be helpful to use consistent naming conventions throughout the ERD. Like for example, using plural forms for entities and singular forms for attributes. Just a thought, but it could help reflect the nature of the data being stored better. Great draft overall!

3. Review by Xiaoqi Wu

My initial post was missing some parts, so I updated my response.

Hello, San and Young-Joon,

The project

- Does the overview describe what problem is to be solved by a website with DB back end?

- Yes, the overview describe that the customer orders movies from the store, and the store employee fulfill the order. It makes sense that the store needs a DBMS to manage their customers, orders, employees, and more.

- Does the overview list specific facts?

- As a human being, I found that it is very interesting to read this project's background. In the problem section, there are specific facts that the database has around 200 movies, but the quantity has been increasing weekly. Meanwhile, there will be more than 10,000 customers later on. It will be better if the overview could also provide how many employees they have.

- Are at least four entities described and does each one represent a single idea to be stored as a list?

- There are 7 entities described. Movie, Employee, Order, Order_item, Customer and Actor are representing single idea to be stored as a list. For Performance, this entity doesn't have any other attribute besides primary key and the foreign key. I would suggest to **add additionally attributes to the performance entity, such as rating or views.**

- Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?

- **Each entity is missing its purpose.**

- Suggestions to the customer_id, employee_id, order_id which could be added unique to their attributes.

- The entities relationship are shown on the project.

- Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

- The 1:M relationships are correctly formulated. For example, there is a 1:M relationship between Order and Employee. In other words, 1 employee can process many orders, but each order can be only processed by 1 employee.

- There are two M:M relationship in database design. One is between Movie and Order, while using Order_item as the intermedia table. The other one is between Movie and Actor, while using Performance.

- The ERD is logical and lists all of the entities as described.

- Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Naming consistency is within the naming of overview, entities and attributes. Entities are use of capitalization of naming, **but they are not plural**, such as movie, actor, order, and customer.

Good luck with your project!

4. Review by Sunny Patel
Hi Sam and Young-Joon,

I want to start by saying that I really enjoyed reading your outline! Especially the overview, it was unique, creative, and similar to how many science-fiction movies are structured, the irony being that in this science fiction world you've created movies are somewhat scarce! Please find my full review below, which some contains comments and suggestions on how to improve this draft for final submission.

·Does the overview describe what problem is to be solved by a website with DB back end?

The team did an excellent job of creating a unique, fun to read science-fiction story about how one entrepreneur is looking to capitalize on a business opportunity created by a political situation. The overview does a great job of establishing a profile of our entrepreneur, and setting the backstory of the situation that has lead this entrepreneur towards starting this buying/selling movie business, along with the business model that will facilitate operations. My only suggestion for the overview is to **beef up the commentary on what exact problem this DB backend is going to solve**. From the overview I understand that this DB will support his website, but maybe go into more detail about **how essential this website is to running smooth business operations in the movie resell business**. I think by doing so the reader will get a much better idea of how building this DB backend/CRUD website is solving a future, or foreseeable problem rather than a present problem Gorak (the entrepreneur) is facing.

·Does the overview list specific facts?

The overview does a great job of listing some key facts about the scope of operations this DB backend will help support. For example, the authors mention that there are currently 200 popular movies in Gorak's store and potentially there will be 10,000+ customers if all goes well. These figures give us an important parameter to access the amount of storage essential entities in this DB will hold (customers and movies are centrally important entities in the ERD design). My only suggestion here is to **add more estimates or forecast facts about some of the other entities as well**. For example, how many employees does Gorak have right now? How many does he plan on hiring if his business grows? How many orders does he currently have vs. forecast for the future? Incorporating facts for these key questions will further help define the scope of this DB.

·Are at least four entities described and does each one represents a single idea to be stored as a list?

In total the group chooses to make seven entities: employee, order, customer, actor, performance, movie, and order item. In essential each entity represents a critical part of the business that any operator would want to store information about and track. Inside of these entities each input/data point looks to correctly represent a single idea as well. For example, one data point in Customer will represent one person, while one data point in Movie will represent one movie. Since the number of entities is over four, and each entity represents a single idea stored as a list, these requirements determined by the project specifications are fulfilled. Great Job! My only suggestion here is to make the entity **names themselves**

plural. Instead of customer use Customers, I believe this is the common practice instructed in the module explorations.

Since the number of entities is over four, and each entity represents a single idea stored as a list the requirements determined by the project specifications are fulfilled. Great Job!

·Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?

The database outline does a great job at listing the attributes within each entity. Each attribute has a datatype specified (int, varchar, or date), and defines any constraints if any (not NULL mostly). My only notable observation on attributes is the naming convention of some entities includes the entity name as a pre-fix. For example, every attribute in customers has customers_ in front, the same for employees as well. I'm not 100% sure, but **I think this can pre-fix naming convention can be dropped**, and singular names can be used like you did in other entities. For example, instead of customer_city you can just name the attribute city. As far as descriptions of relationships between entities are concerned, it was well described with no changes needed! The final comment I would like to point out is that each entity in the outline is **missing a description/details of what purpose that entity serves**. Make sure you add a sentence or two for each entity, so you don't lose points in the final submission!

·Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

As far as I can tell all of the 1:M relationships are correctly formulated, it makes logical sense that some entities such as Employee, Customer are singularly connected to multiple instances of another entity like Order, and vice versa. Additionally, there are looks to be two M:M relationships, one between Movie and Actor, and another between Order and Movie. These M:M relationships also make logical sense. For example, multiple instances of Order will be connected to multiple instances of Movie, and vice versa. Finally, I would say that this ERD presents a plausible/logical view of the operations this database will support. These entities and their relationships seem comprehensive enough to support the operations of limited movie growing business, with flexibility to connect in more entities if needed.

·Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

As mentioned above there were some inconsistencies noted. To recap, the **entity names are not plural, and instead they are singular**. For example, instead of Customer (singular – Bad), it should be Customers (Plural – Good). Additionally, while each attribute in the entities is singular - which is good, maybe consider **dropping the entity name as a pre-fix** to each attribute. For example, in Customer entity the attribute customer_first can simply be renamed as first_name. It will hopefully serve the same purpose

when fully building out the database and look much cleaner. Finally, it looks like every Entity name was capitalized as needed, and every attribute name was lowercase as intended. Great job! Make these minute changes to get full points on the final version.

One final note, it looks like you **left an internal team comment** just below the Outline header. Make sure you remove that as well before the final submission!

Actions based on the feedback:

1. Added short description to entities to clarify what they are
2. Changed entity names in the description section to plural
3. Change entity names in the ERD to plural
4. order_id and movie_id changed to FK in Order_item
5. Added more attribute candidates under Movies entity (will consider implementing as the project develops)
6. Added more backstory to clearly explain the goal of the database
7. Added more estimates and forecast for the growth of Gorak's business that the database must support
8. Dropped prefixes in attribute names
9. The composite key issue that was pointed out by the TA feedback has been worked out. Composite key kept
10. ERD diagram inconsistency pointed out by the TA corrected

The reasons for actions that were NOT taken are commented in blue in the above section.

Upgrades to the draft version:

1. Added short description to entities to clarify what they are
2. Changed entity names in the description section to plural
3. Change entity names in the ERD to plural
4. order_id and movie_id changed to FK in Order_item

5. Added more attribute candidates under Movies entity (will consider implementing as the project develops)
6. Added more backstory to clearly explain the goal of the database
7. Added more estimates and forecast for the growth of Gorak's business that the database must support
8. Dropped prefixes in attribute names
9. Deleted ship_or_pickup attribute in Orders entity
10. Replaced screenshot of ERD
11. Added a detailed schema
12. Updated Movies.price from int to Decimal (16,2)
13. Updated ON DELETE CASCADE to select foreign keys (in Order_items, Performances)
14. Updated ERD to be consistent with outline

PROJECT OUTLINE

A) Overview

Gorak is an entrepreneurial Gorkian, an interstellar species faring from Gorkia Prime. In Gorkia, human movies from Earth have recently grown in popularity among Gorkian high society. With figures like Gelon Gusk collecting primitive blu-rays containing human cinema, Gorak has made a company to profit from the trend.

As part of the Gorkian policy of xeno-preservation, Gorkians are strictly prohibited when it comes to directly contacting humans in their original form. Therefore, specially trained Gorkian agents have infiltrated human society disguised in human form. The only direct contact permitted between Gorkians and humans is to be done in disguise until the Gorkian congress passes first contact protocols. However, the Gorkian senate is in political deadlock, and Gorak expects that passing the protocols will take a long time.

Thus, Gorak made the following business model:

Gorak's store wants to build a galaxy-website where a potential customer can look up the title, price, stock, and actors in a given human movie (the database currently has around 200 most popular movies, but is growing every week). If the customer wants to buy one or more blu-ray discs of one or more movies, she can order the products online by inputting her galactic-credit card number, products, and quantities. Once the order is received, one Gorkian infiltrator agent employee is assigned to that order. The order is then fulfilled by this agent and is shipped back to the address of a customer in Gorkia. Because of the expenses involved in transporting the original blu-ray back to Gorkia Prime, Gorak estimates that his customer base will number in the 10,000s.

<Problem to solve and numerical facts, entities noted in parentheses>

Gorak's first step is to design a database that can support his website and operations. Gorak envisions facilitating order (Orders and Order_items) fulfillment by using his database to let customers (Customers) pick an available movie (Movies) and look at the general information of the movie (Actors, Performances, Movies) from his site, submit an order, and assign that order to an employee (Employees, who is a Gorkian infiltrator). The database also has to support the growth of Gorak's business. Gorak expects that the website will support up to 10,000 movies and 100,000 customers by the end of the year.

B) Database Outline:

Customers:

Description: a customer who put in an order to Gorak's business. The customer is located in Gorkia prime and is using Gorak's galactic-website to put in orders for Gorak's business to fulfill.

- Attributes:
 - id : int, auto_increment, not NULL, PK;
 - first : VARCHAR(50), not NULL;
 - last : VARCHAR(50), not NULL;
 - street : VARCHAR(50), not NULL;
 - city : VARCHAR(50), not NULL;
 - state : VARCHAR(50), not NULL;
 - phone : VARCHAR(15), not NULL;
- Relationships:
 - M : 1 relationship between Customer and Order (FK customer_id in Order)

Employees:

Description: an Gorkian infiltrator hired by Gorak to procure ordered blu-ray discs. One employee is assigned to one order from one customer and is in charge of filling out that order.

- Attributes:
 - id : int, auto_increment, not NULL, PK;
 - first : VARCHAR(50), not NULL;
 - last : VARCHAR(50), not NULL;
 - street : VARCHAR(50), not NULL;
 - city : VARCHAR(50), not NULL;
 - state : VARCHAR(50), not NULL;
 - phone : VARCHAR(50), not NULL;
- Relationships:
 - M : 1 relationship between Employee and Order (FK employee_id in Order)

Orders:

Description: an order that the customer has filled out. The junction table Order_items indicate what and how many items an order contains.

- Attributes:
 - order_id : int, auto_increment, not NULL, PK;
 - customer_id : int, FK
 - employee_id : int, FK
 - order_date : DATETIME, not NULL;
 - credit_card_num : VARCHAR(16), not NULL;
 - order_fulfilled : BIT, not NULL; – this is a BOOL value
- Relationships:
 - 1 : M relationship between Order and Customer (FK customer_id in Order)
 - 1 : M relationship between Order and Employee (FK employee_id in Order)
 - M : 1 relationship between Order and Order_item (FK order_id in Order_item)

Order_items:

Description: linked to an order, an order item contains one movie and the quantity of that movie that has been ordered. It is linked to an order_id, allowing customers to buy multiple movies and multiple copies of the same movie. **Junction table between Movies and Orders.**

- Attributes:

- order_id : int, auto_increment, not NULL, PK, FK;
 - Composite key
- movie_id : int, auto_increment, not NULL, PK, FK;
 - Composite key
- quantity : int;
- Relationships:
 - 1 : M relationship between Order_item and Order (FK order_id in Order_item)
 - 1 : M relationship between Order_item and Movie (FK movie_id in Order_item)
- Deletion of order_id and movie_id cascades to delete Order_items entry

Movies:

Description: a human movie that Gorak's business offers on the galactic-website for his store. Xeno-preservists in the Gorkian congress love the Avatar series.

- Attributes:
 - movie_id : int, auto_increment, not NULL, PK;
 - title : VARCHAR(100), not NULL;
 - stock : int, not NULL;
 - price : DECIMAL(16, 2), not NULL;
- Possible additional attributes:
 - main_genre: VARCHAR(32)
 - ratings: int;
- Relationships:
 - M : 1 relationship between Movie and Order_item (FK movie_id in Order_item)
 - M : 1 relationship between Movie and Performance (FK movie_id in Performance)

Performances:

Description: a junction table indicating a performance of an actor in a movie. The entity is there to get rid of the M:M relationship between Movies and Actors. **Junction table between Movies and Actors.**

- Attributes:
 - performance_id : int, auto_increment, not NULL, PK;
 - movie_id : int, FK
 - actor_id : int, FK
- Relationships:

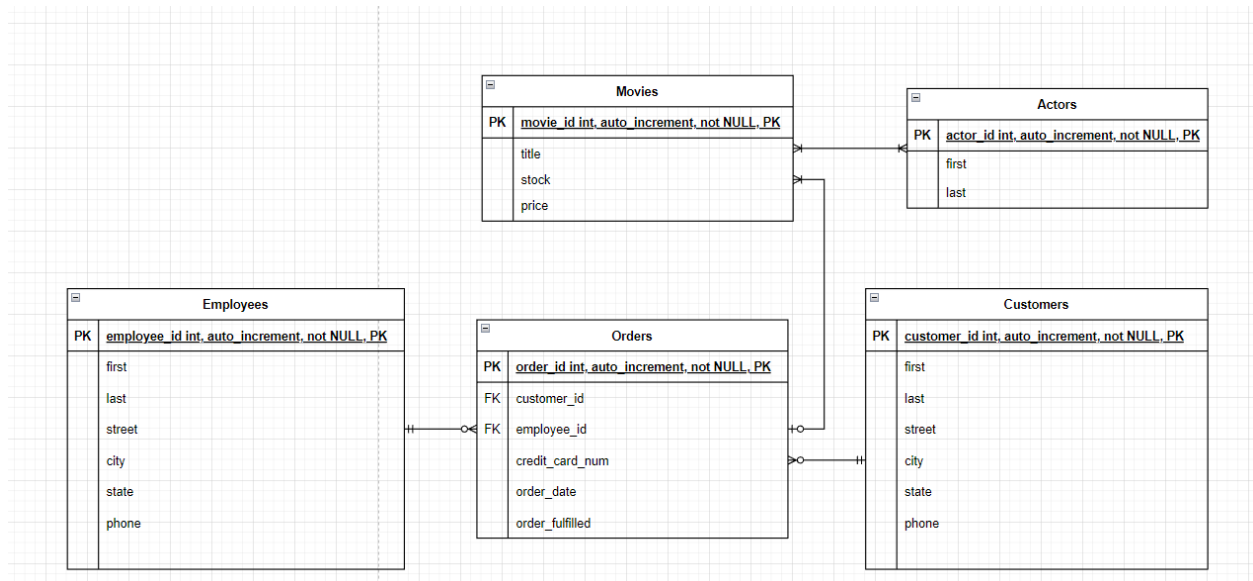
- 1 : M relationship between Performance and Movie (FK movie_id in Performance)
- 1 : M relationship between Performance and Actor (FK actor_id in Performance)
- Deletion of movie_id and actor_id cascades to delete Performances entry

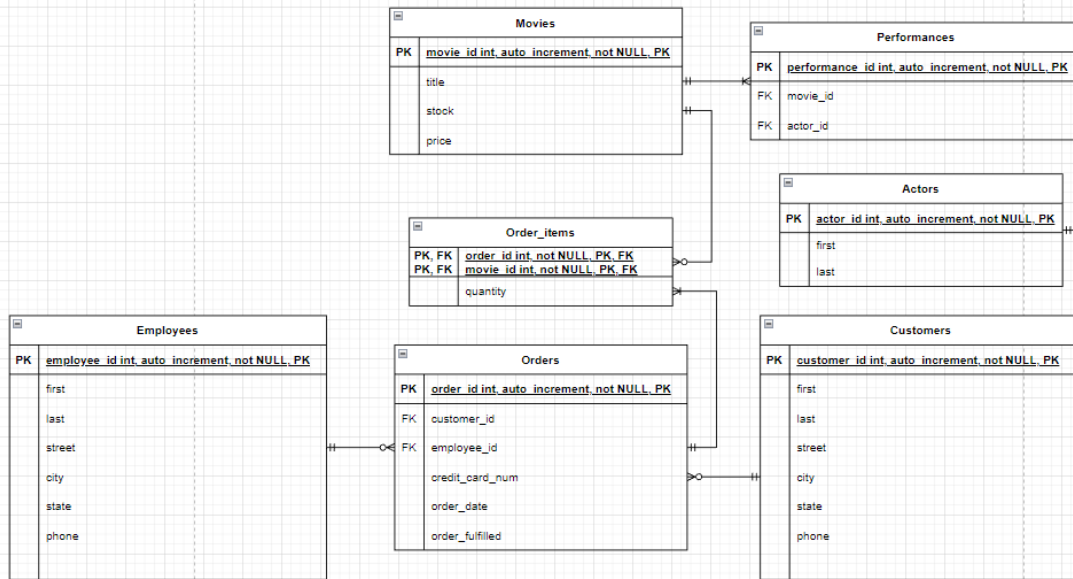
Actors:

Description: an Earthar actor who performs in a movie. There are rumors that a famous Hollywood actor is a Gorkian agent.

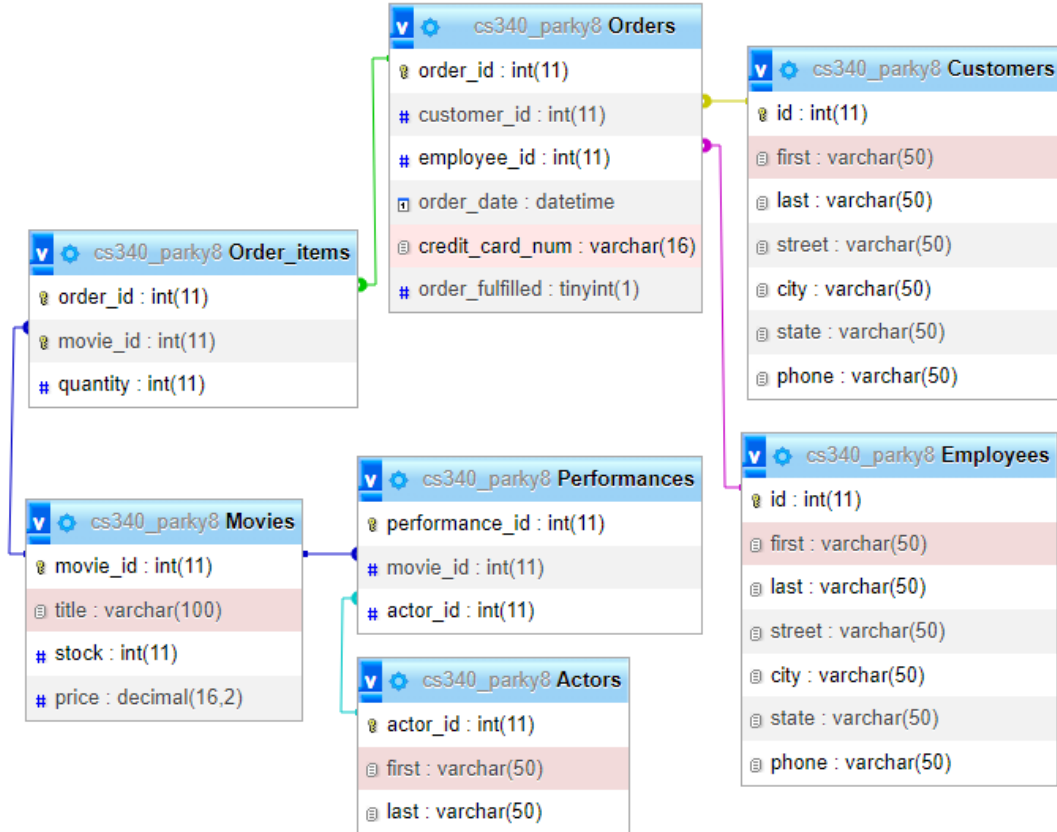
- Attributes:
 - actor_id : int, auto_increment, not NULL, PK;
 - first : VARCHAR(50), not NULL;
 - last : VARCHAR(50), not NULL;
- Relationships:
 - M : 1 relationship between Actor and Performance (FK actor_id in Performance)

C) Entity-Relationship Diagram, Resolution of M to N Relationship:





D) Schema:



E) Sample Data:

```
MariaDB [cs340_park8]> select * from Customers;
```

id	first	last	street	city	state	phone
1	YJ	Park	2415 Chestnut St	Philadelphia	PA	1234567890
2	San	Davran	1231 Maple St	New York City	NY	0987654321
3	Tupac	Shakur	9807 Oak St	Los Angeles	CA	3216540987

```
3 rows in set (0.000 sec)
```

Customers						
id	first	last	street	city	state	phone
1	YJ	Park	2415 Chestnut	Philadelphi a	PA	123456789 0

			St			
2	San	Davran	1231 Maple St	New York City	NY	0987654321
3	Tupac	Shakur	9807 Oak St	Los Angeles	CA	3216540987

```
MariaDB [cs340_park8]> select * from Employees;
+-----+-----+-----+-----+-----+-----+-----+
| id | first | last | street | city | state | phone |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Erling | Haaland | 3211 Oakview St. | Manchester | MA | 1111111111 |
| 2 | Harry | Kane | 3453 Carnaby St | London | TN | 2222222222 |
| 3 | Darwin | Nunez | 1315 Bristol St | Liverpool | UK | 3333333333 |
+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.000 sec)
```

Employees						
id	first	last	street	city	state	phone
1	Erling	Haaland	3211 Oakview St	Manchester	MA	1111111111
2	Harry	Kane	3453 Carnaby St	London	TN	2222222222
3	Darwin	Nunez	9807 Oak St	Los Angeles	CA	3333333333

```
MariaDB [cs340_park8]> select * from Orders;
+-----+-----+-----+-----+-----+-----+
| order_id | customer_id | employee_id | order_date | credit_card_num | order_fulfilled |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 2 | 2022-01-11 00:00:00 | 1111222233334444 | 0 |
| 2 | 1 | 1 | 2022-01-12 00:00:00 | 2222333344445555 | 1 |
| 3 | 2 | 2 | 2022-01-01 00:00:00 | 3333444455556666 | 1 |
| 4 | 2 | 3 | 2022-01-04 00:00:00 | 4444555566667777 | 0 |
| 5 | 1 | 2 | 2022-01-04 00:00:00 | 5555666677778888 | 0 |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

Orders					
order_id	customer_id	employee_id	order_date	credit_card_num	order_fulfilled
1	1	2	2022-01-11	1111222233334444	0
2	1	1	2022-01-12	2222333344445555	1
3	2	2	2022-01-01	3333444455556666	1

4	3	3	2022-01-04	4444555566667777	0
5	2	2	2022-01-04	5555666677778888	0

```
MariaDB [cs340_parky8]> select * from Order_items;
+-----+-----+-----+
| order_id | movie_id | quantity |
+-----+-----+-----+
|         1 |         2 |         2 |
|         1 |         3 |         4 |
|         2 |         1 |         8 |
|         3 |         2 |         5 |
|         4 |         1 |         1 |
|         5 |         2 |         3 |
+-----+-----+-----+
6 rows in set (0.000 sec)
```

Order_items		
order_id	movie_id	quantity
1	3	4
1	2	2
2	1	8
3	2	5
4	1	1
5	2	3

```
MariaDB [cs340_parky8]> select * from Movies;
+-----+-----+-----+-----+
| movie_id | title           | stock | price |
+-----+-----+-----+-----+
|         1 | Aliens          |      3 | 12.99 |
|         2 | The Purge       |     13 |  2.99 |
|         3 | The Black Phone |     21 |  5.99 |
+-----+-----+-----+-----+
3 rows in set (0.000 sec)
```

Movies			
movie_id	title	stock	price
1	Aliens	3	12.99

2	The Purge	13	2.99
3	The Black Phone	21	5.99

```
MariaDB [cs340_parky8]> select * from Performances;
```

```
+-----+-----+-----+
| performance_id | movie_id | actor_id |
+-----+-----+-----+
| 1 | 2 | 1 |
| 2 | 2 | 2 |
| 3 | 2 | 3 |
| 4 | 2 | 4 |
| 5 | 1 | 5 |
| 6 | 1 | 6 |
| 7 | 1 | 7 |
| 8 | 1 | 8 |
| 9 | 3 | 9 |
| 10 | 3 | 10 |
| 11 | 3 | 11 |
+-----+-----+-----+
```

```
11 rows in set (0.000 sec)
```

Performances		
performance_id	movie_id	actor_id
1	2	1
2	2	2
3	2	3
4	2	4
5	1	5
6	1	6
7	1	7
8	1	8
9	3	9
10	3	10
11	3	11
12	3	1

```

Database changed
MariaDB [cs340_parky8]> select * from Actors;
+-----+-----+-----+
| actor_id | first      | last      |
+-----+-----+-----+
| 1        | Ethan      | Hawke     |
| 2        | Lena       | Headey    |
| 3        | Max        | Burkholder|
| 4        | Adelaide   | Kane      |
| 5        | Sigourney  | Weaver    |
| 6        | Michael    | Biehn     |
| 7        | Carrie     | Henn      |
| 8        | Paul       | Reiser    |
| 9        | Mason      | Thames    |
| 10       | Jeremy     | Davies    |
| 11       | Rebecca    | Clarke    |
+-----+-----+-----+
11 rows in set (0.001 sec)

```

Actors		
actor_id	first	last
1	Ethan	Hawke
2	Lena	Headey
3	Max	Burkholder
4	Adelaide	Kane
5	Sigourney	Weaver
6	Michael	Biehn
7	Carrie	Henn
8	Paul	Reiser
9	Mason	Thames
10	Jeremy	Davies
11	Rebecca	Clarke