Kyle Bell
Jeremy Tsang

Project Step 2 Draft Version: ERD & Schema

(A) Fixes based on Feedback from Step 1

We did not receive any feedback but we made the following changes:

- Switch naming conventions from camel case to underscores.
- Rename the composite entities shipTrooperManifest and shipDroidManifest to ships_troopers and ships_droids respectively.
- Change loadOuts to loadouts since loadout is a single word.
- Change the relationship between loadouts and troopers from M:M to M:1.
- In the Database Outline for ship_troopers indicate that together ship_id and trooper_id form the primary key for composite entity ship_troopers.
- In the Database outline for ship_droids indicate that together ship_id and droid_id form the primary key for composite entity ship_droids.

(B) Project Outline and Database Outline - Updated Version*

we have not received any feedback yet but we have included some updates

Project Title: Empire Employees
Team: Rakghoul Serum

Without proper organization and resolve the galaxy will turn to chaos, or worse, become under the control of rebel scum. Empire Employees manages over 1 billion Stormtroopers along with supporting droids and vehicles. To deploy a specific number of troops to ensure effective defense of each garrison spread such large cosmic distances it is critical to always have the correct number of supplies and personnel available. Empire Employees does this by allowing the users current troop, ship, and droid counts for each garrison and providing sufficient warnings when said counts drop below an acceptable level needed for adequate defense. For some ships, such as tie fighters, there may only be 1 trooper on board whereas for larger ships, such as star destroyers, may have as much as 9,700 Stormtroopers. With our easy to use online interface, you can get a taste of how easy and efficient our software is to meet all your Stormtrooper relocation management needs!

Entity Tables:

troopers - an individual soldier of the Galactic Empire that upholds peace and prosperity for all. They have a garrison and load out assignments that are related to the garrisons and loadouts entities respectively.

- id: int, auto_increment, unique, not NULL, PK
- garrison: int, FK

- loadout: int, not NULL
- Relationship: a 1:M relationship between garrisons and troopers, a M:M relationship between troopers and ships (implemented with the ships_troopers composite entity), and a 1:M relationship between loadouts and troopers.

ships - The Galactic Empire has ships in order to transport troopers and droids throughout the galax. Each ship has a ship type and is related to troopers and droids.

- id: int, auto_increment, unique, not NULL, PK
- type: varchar(255), not NULL
- Relationship: a M:M relationship between ships and troopers, a M:M relationship between ships and droids.

droids - Robust, top of the line droids help repair ships for our glorious fleet. They have a droid type, and have a relationship with ships.

- id: int, auto_increment, unique, not NULL, PK
- type: varchar(255), not NULL
- Relationship: a M:M relationship between droids and ships implemented with the ships_droids composite entity.

loadouts - Each unique loadout contains a proprietary blaster as well as an option for a detonator in order for each trooper to get the job done right their own way! A loadout has a relationship with a trooper.

- id: int, auto_increment, unique, not NULL, PK
- blaster: varchar(255), not NULL
- detonator: bool, not NULL
- Relationship: a M:M relationship between troopers and loadouts

garrisons - Our Empire keeps the galaxy safe with garrisons throughout the galaxy. Each of them has a name and a maximum capacity

- id: int, auto_increment, unique, not NULL, PK
- Name: varchar(255), not NULL
- capacity: int, not NULL
- Relationship: a 1:M relationship is implemented between troops and garrisons as a FK inside of troops.

Relationship Tables:

ships_troopers - Relationship between ships and troopers.

- ship_id: int, not NULL, PK, FK
- trooper_id: int, not NULL, PK, FK
- Relationship: facilitates M:M relationship between ships and troopers.

ships_droids - Relationship between ships and droids.

- ship_id: int, not NULL, PK, FK
- droid_id: int, not NULL, PK, FK
- Relationship: facilitates M:M relationship between ships and droids.

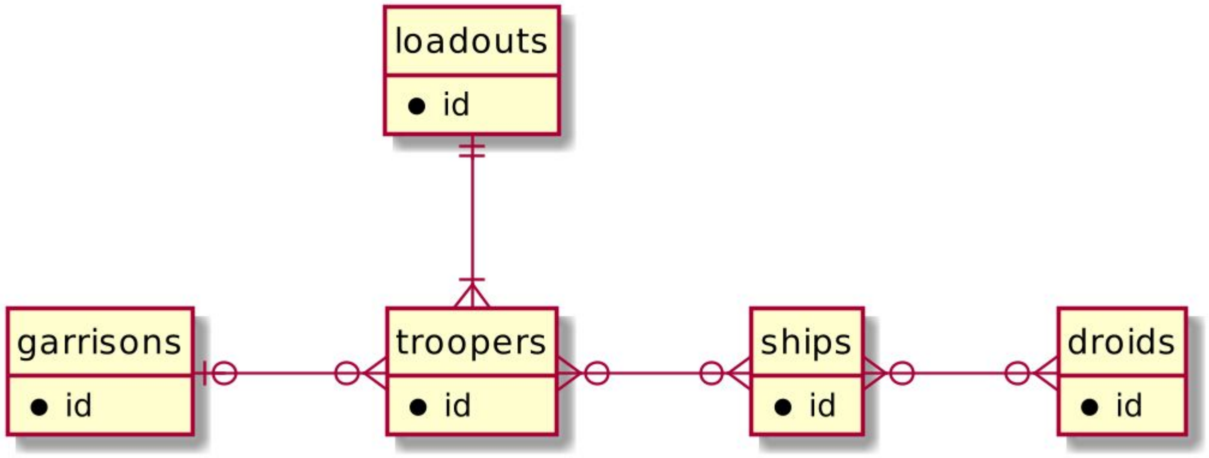
Team Assignments:

We will be implementing all of the entities described above:

- troopers
- loadouts
- garrisons
- ships
- droids
- ships_droids
- ships_troopers

- Kyle will be responsible for the code and webpages for the following entities:
  - ships_droids (M:M relationship)
  - droids
  - garrisons
  - ships
- Jeremy will be responsible for code and webpages for the following entities:
  - ships_troopers (M:M relationship)
  - troopers
  - loadouts

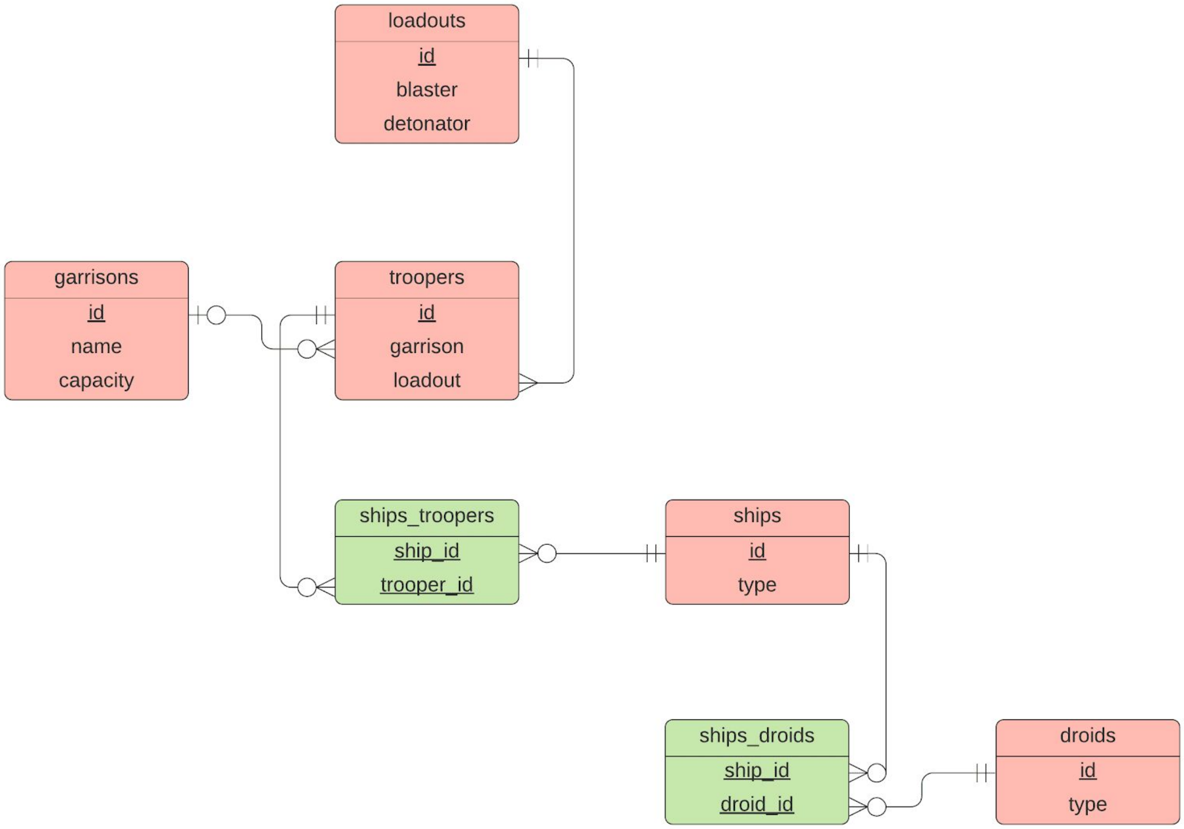
(C) Entity-Relationship Diagram



Kyle Bell
Jeremy Tsang

(D) Schema

Schema



loadouts
id
blaster
detonator
garrisons
id
name
capacity
troopers
id
garrison
loadout
ships_troopers
ship_id
trooper_id
ships
id
type
ships_droids
ship_id
droid_id
droids
id
type