1. Business Rules:

* Each user can create multiple posts.
* A user must be logged in to create, edit, or delete their own posts.
* Each post represents a unique clothing item and contains specific attributes.
* Each clothing item post must have a title, photo, color, type, material, short description, the username of the creator, and a creation date.
* Only the user who created a post can modify or delete it.

2. Identified Entities:

* User
* ClothingPost (represents a clothing item)
* ClothingType (predefined types)
* Role
* Color
* material

**3. Defined Attributes:**

User:

* Username (Primary Key)
* password
* role\_id (Foreign Key referencing Role)

Role:

* role\_id (Primary Key)
* role (e.g., "user," "guest," "admin")
* ~~permission (e.g., "view," "create\_delete", “create\_delete\_universalDelete)~~

ClothingPost:

* post\_id (Primary Key)
* username (Foreign Key referencing User)
* title
* photo\_url (URL or path for the photo of the item)
* color\_id (Foreign Key referencing Colors table)
* clothing\_type\_id (Foreign Key referencing ClothingType)
* material\_id (Foreign Key referencing materials table)
* description (a short description of the item)
* created\_at (date and time of creation)

ClothingType:

* clothing\_type\_id (Primary Key)
* type(e.g., "Shirt," "Jacket," "Pants")
* ~~season (e.g., "Summer," "Winter") - to help categorize clothing suitable for different seasons~~
* ~~style (e.g., "Casual," "Formal," "Athletic") - provides an additional categorization of the clothing type~~

Colors:

* Color\_id (Primary Key)
* Color (purple, red, blue, etc)

Materials

* Material\_id (primary key)
* Material (e.g., cotton, polyester, etc)

4. Relationships:

* User to ClothingPost: One-to-Many (A user can create multiple posts, but each post is linked to only one user).
* User to Role: many-to-One (One user can only have one role, but one role can have many users)
* ClothingPost to ClothingType: One-to-many (each post contains only one piece of clothing, and a piece of clothing can only be one type, eg it can’t be both a shirt and pant) but one type can have many posts
* Colors to ClothingPost: many-to-one: one color can be associated with many posts, but a post can only have one color
* Materials to ClothingPost: many-to-one: one material can be associated with many posts, but a post can only have one material

**DATABASE CREATE SCRIPT:**

-- Create the Roles table

CREATE TABLE Role (

role\_id INT AUTO\_INCREMENT PRIMARY KEY,

role VARCHAR(50) NOT NULL

);

-- Create the Users table

CREATE TABLE User (

username VARCHAR(50) PRIMARY KEY,

password VARCHAR(255) NOT NULL,

role\_id INT NOT NULL,

FOREIGN KEY (role\_id) REFERENCES Role(role\_id)

);

-- Create the Colors table

CREATE TABLE Colors (

color\_id INT AUTO\_INCREMENT PRIMARY KEY,

color VARCHAR(50) NOT NULL

);

-- Create the Materials table

CREATE TABLE Materials (

material\_id INT AUTO\_INCREMENT PRIMARY KEY,

material VARCHAR(50) NOT NULL

);

-- Create the ClothingTypes table

CREATE TABLE ClothingType (

clothing\_type\_id INT AUTO\_INCREMENT PRIMARY KEY,

type VARCHAR(50) NOT NULL

);

-- Create the ClothingPosts table

CREATE TABLE ClothingPost (

post\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL,

title VARCHAR(100) NOT NULL,

photo\_url VARCHAR(255) NOT NULL,

color\_id INT NOT NULL,

clothing\_type\_id INT NOT NULL,

material\_id INT NOT NULL,

description TEXT NOT NULL,

created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (username) REFERENCES User(username),

FOREIGN KEY (color\_id) REFERENCES Colors(color\_id),

FOREIGN KEY (clothing\_type\_id) REFERENCES ClothingType(clothing\_type\_id),

FOREIGN KEY (material\_id) REFERENCES Materials(material\_id)

);

**DATABASE POPULATE SCRIPT:**

-- Create the Roles table

CREATE TABLE Role (

role\_id INT AUTO\_INCREMENT PRIMARY KEY,

role VARCHAR(50) NOT NULL

);

-- Create the Users table

CREATE TABLE User (

username VARCHAR(50) PRIMARY KEY,

password VARCHAR(255) NOT NULL,

role\_id INT NOT NULL,

FOREIGN KEY (role\_id) REFERENCES Role(role\_id)

);

-- Create the Colors table

CREATE TABLE Colors (

color\_id INT AUTO\_INCREMENT PRIMARY KEY,

color VARCHAR(50) NOT NULL

);

-- Create the Materials table

CREATE TABLE Materials (

material\_id INT AUTO\_INCREMENT PRIMARY KEY,

material VARCHAR(50) NOT NULL

);

-- Create the ClothingTypes table

CREATE TABLE ClothingType (

clothing\_type\_id INT AUTO\_INCREMENT PRIMARY KEY,

type VARCHAR(50) NOT NULL

);

-- Create the ClothingPosts table

CREATE TABLE ClothingPost (

post\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL,

title VARCHAR(100) NOT NULL,

photo\_url VARCHAR(255) NOT NULL,

color\_id INT NOT NULL,

clothing\_type\_id INT NOT NULL,

material\_id INT NOT NULL,

description TEXT NOT NULL,

created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (username) REFERENCES User(username),

FOREIGN KEY (color\_id) REFERENCES Colors(color\_id),

FOREIGN KEY (clothing\_type\_id) REFERENCES ClothingType(clothing\_type\_id),

FOREIGN KEY (material\_id) REFERENCES Materials(material\_id)

);