Department of Electrical and Computer Engineering

University of Puerto Rico

Mayagüez Campus

Members: Luis Liz, Yeniel Díaz, Jorge Vega

**Term Project – Backend System for Disaster Site Resources Locator**

**Entity Relationship Diagram Report**

The following are descriptions that illustrate the behaviour and interactions of the entities within the Entity Relationship diagram.

**Tables:**

Resources: This table holds information on all resource names, their resource id and a foreign key linking them to a specific resource category.

Resource Details: Holds details of the resource such as availability (purchased, reserved, available), quantity, price per unit, location, and the supplier.

Water: Contains information of all resources that are water bottles or water gallons.

Medications: Contains information of all resources that are medication.

Baby Foods: Contains information of all resources that are baby foods.

Canned Foods: Contains information of all resources that are canned foods.

Dry Foods: Contains information of all resources that are dry foods.

Ice: Contains information of all resources that are ice.

Fuel: Contains information of all resources that are fuel.

Medical Device: Contains information of all resources that are medical devices.

Tools: Contains information of all resources that are tools.

Clothing: Contains information of all resources that are clothings.

Power Generators: Contains information of all resources that are power generators.

Batteries: Contains information of all resources that are batteries.

Heavy Equipment: Contains information of all resources that are heavy equipment.

Users: Represents every user and holds their basic information: Name, date of birth, email and password, plus holds a foreign key to specify the user’s category (supplier, user, admin).

User Category: Holds all the categories a user can fall under. A user can be a requester (normal user), supplier or administrator.

Request: Each entry represents a request from a user for a set of resources, holds the information on when the request was posted, the location where its requested to be brought, the user requesting, the user that supplied the requested resource (initially null) and the date at which it was supplied (initially null).

Resources per Request: Relationship table that contains all the resources that belong to a request in the request table.

Payment Info: Holds the payment information a user will use to pay for supplies, or to receive from selling supplies. Information like: card number, card type, card provider, and card date of expiration

Transaction: Each entry represents an exchange of goods between users, whether it was reserving resources or purchasing them. Holds information such as: purchaser, supplier, resource being exchanged, quantity of the resource being exchanged, the date of the transaction and the amount of money paid for the resource (can be zero in the case of reservation).

**Relationships:**

* User-Request:
  + User Makes Request: 1 to Many with total participation on the Many side. User is the identifying entity and request is a weak entity. Users can make multiple requests and requests cannot exist without a user.
  + User Supplies Request: 1 to Many. A user can supply many requests
* User has Payment\_Info: 1 to Many total participation on the Many side. A user can have various payment infos and payment info cannot exist without belonging to a user
* User belongs to User\_Category: Many to 1. A user has to identify as a supplier, person seeking supplies or administrator. This allows there to be consistency for the type of user.
* User is supplier for Resource\_Details: 1 to Many. A user can provide various resources and the details table is the one that separates the resources by supplier location, etc.
* Transaction keeps track of Resource: Many to 1. A resource can participate in many transactions but each transaction can only be used to exchange one resource.
* Transaction-Payment\_Info:
  + Payment Info payes Transaction: 1 to Many. Each transaction for a resource with a price greater than 0 has a payment info that will pay the price of the resource
  + Payment\_Info gets payed in Transaction: Many o 1 Each transaction for a resource with a price greater than 0 has a payment info that will receive the price of the resource
* Request requests Resources: Many to Many. A resource can be requested many times and each request can request many resources.
* Resource has Resource\_Details: 1 to 1. Each resource record has exactly one resource detail record that describes the details for that resource.
* Resource-Resource\_Category: Many to 1. Every single resource category table (water, medications, baby foods, canned foods, dry foods, ice, fuel, medical devices, tools, clothings, power generators, batteries and heavy equipment) has this relationship to resources. Every resource has this relationship with exactly one of these category tables.