University of Puerto Rico
Mayagüez Campus
Electrical and Computer Engineering Department

Disaster Site Resources Locator - Phase 2

Gabriel Cantres Rosario
Erik Gutiérrez Cataquet
Julian Quiñones Pérez
Zhaotian Wu Meléndez
ICOM5016 - 116
3/31/2020

Note: Phase 2 changes to the Entity Relation Diagram Include:

Removing foreign keys from all entities
Correcting ISA symbol
Splitting Ternary Relations
Payment connects relation solely to Supplier
Resource - Order cardinality fix and table addition
Attributes like address made compound

---

1. **User Accounts** is the building block for all other accounts. Given that the majority of attributes(*Username, Names, email, phones, etc.)* within supplier, requester, and administrator are the same, personal information is stored in the user account while specific IDs and locations are stored in the **Supplier** and **Requester** tables.

2. **Suppliers** can perform a variety of functions. The entity does not contain many attributes as it "inherits" the majority of information from the **User Accounts** table. Among the many things a **supplier** can do, a supplier can register as a **Company** employee (see **Company** table). **Suppliers**' main role is to *stock* **Resources** and *post* **Listings** which announce the availability of a **Resource** to be reserved or purchased. **Suppliers** *receive* **Payments** from **Orders** placed by **Requesters** after *responding* to a **Request.** This is the purchase or donation mechanic of the application, wherein a **Requester** reserves a good either with or without cost.

3. **Company** is where the corporate details of the company being represented (supplier registered as company) reside. This table is a way for employees to identify as a representative to a specific company. When **Payments** are made, the **Company** (not the **Supplier**) receives the payment from the **Requester**. All **Listings** and responses to **Requests** however, are handled by the **Supplier** who is a company employee if they do not operate independently.

4. **Requesters** represent the clients seeking the various resources tracked by the application. They *post* **Requests** to let **Suppliers** know they are seeking a certain **Resource**, from which they receive a notification and a reply when their resource becomes available and a **Supplier** wishes to transact. Then, the **Requester** can *place* an order for the requested **Resource** in addition to *sending* the pertinent payment, if applicable. **Requesters** also help let **Suppliers** know what kind of resources are in demand through the **Requests**.

5. **Listings** announce the availability and sale or donation of a resource. Listings display which resource is being supplied, who is supplying the resource, the date of the listing post, location of the resource, and how much of the specific resource is being sold. Listings can only be posted by **Suppliers**. When a specific resource is Requested by a **Requester**, the supplier is notified to respond to this **Request.** Once the **Request** is accepted, the **Requester** can place an **Order** which displays the completion of the transaction. In the case that the **Listing** is of a large quantity, and a **Requester** only purchases an amount less than the entirety of the stock, the *quantity* in the listing is reduced. In the case that all of the stock is purchased, *quantity* in the listing becomes 0 and the listing is no longer displayed.

6. **Requests** announce the need for a resource by a **Requester** user. The table displays the status of the **Request** (accepted, pending, etc.), the resource needed (rid), the *quantity* of the resource needed and the *Date* in which the request was posted. **Requests** can only be posted by **Requesters**, and are

accepted/rejected by **Suppliers**, who own the resources solicited in the **Request**. The acceptance of a **Request** allows the **Requester** to place an **Order,** through which the transaction of the resource takes place.

7. An **Order**, as mentioned previously, is created when a **Requester** places it through an accepted **Request. Order** acts as a log of all the information pertinent to a transaction: the **Requester's** ID, the **Supplier's** ID, **Payment** ID, **Resource** ID being exchanged/donated, quantity of the resource being donated, and the status of the **Order,** which tracks the exchange of the named resources and details whether its being processed, is fully completed, etc.

8. **Resource** is the table with all the information pertinent to an existing resource. This includes: its ID (rid), its name, its **Supplier's** ID, and the department (rtype) the item belongs to. The department it belongs to leads us to the rest of the resource's information related to the various **Resource** subtypes (ISA): e.g. Fuel, Clothing, Medication, Batteries, etc. Each of these departments contain their own attributes describing the specifications of each kind: e.g. Water brand, size. Fuel type, quantity, etc.

9. **Payment** is the table containing the specifics to fulfill an **Order's** payment. It has attributes for the **Requester** ID of who submitted the payment, the **Supplier** ID it is addressed to, the amount (price) being paid, the payment type (visa card#, exp date, sec code), and the account details of the **Supplier** to which the payment is being sent.

10. **Administrator** accounts are in charge of registering and managing other types of user accounts. These contain higher privileges to make changes in the application. However, the relationships of the administrator are not included in the ER since they are not detailed in the current phase of the project.