Phase 1: Term Project for Spring 2020 Semester

University of Puerto Rico at Mayaguez, *Computer Engineering Department*
Prof. Manuel Rodríguez Martínez

Disaster Resource Locator API

March 31st, 2020

TEAM:

Victor Nazario
victor.nazario@upr.edu

Kiara Rodríguez
kiara.rodriguez18@upr.edu

Wilfredo Aponte
wilfredo.aponte@upr.edu

How to execute the program and consume API

It's important to know that already a **local database configuration** is in place for the project, thus the user **ideally would** have a working version of MySQL installed on the machine. The MySQL server **should** be reachable in the following direction jdbc:mysql://localhost:3306/terremotos. Below is a screenshot of datasource configuration in Intellij Idea. For this first phase we're database testing, thus user and password both should be **root**. The program may show side effects if a datasource is not configured, thus it's highly recommended.

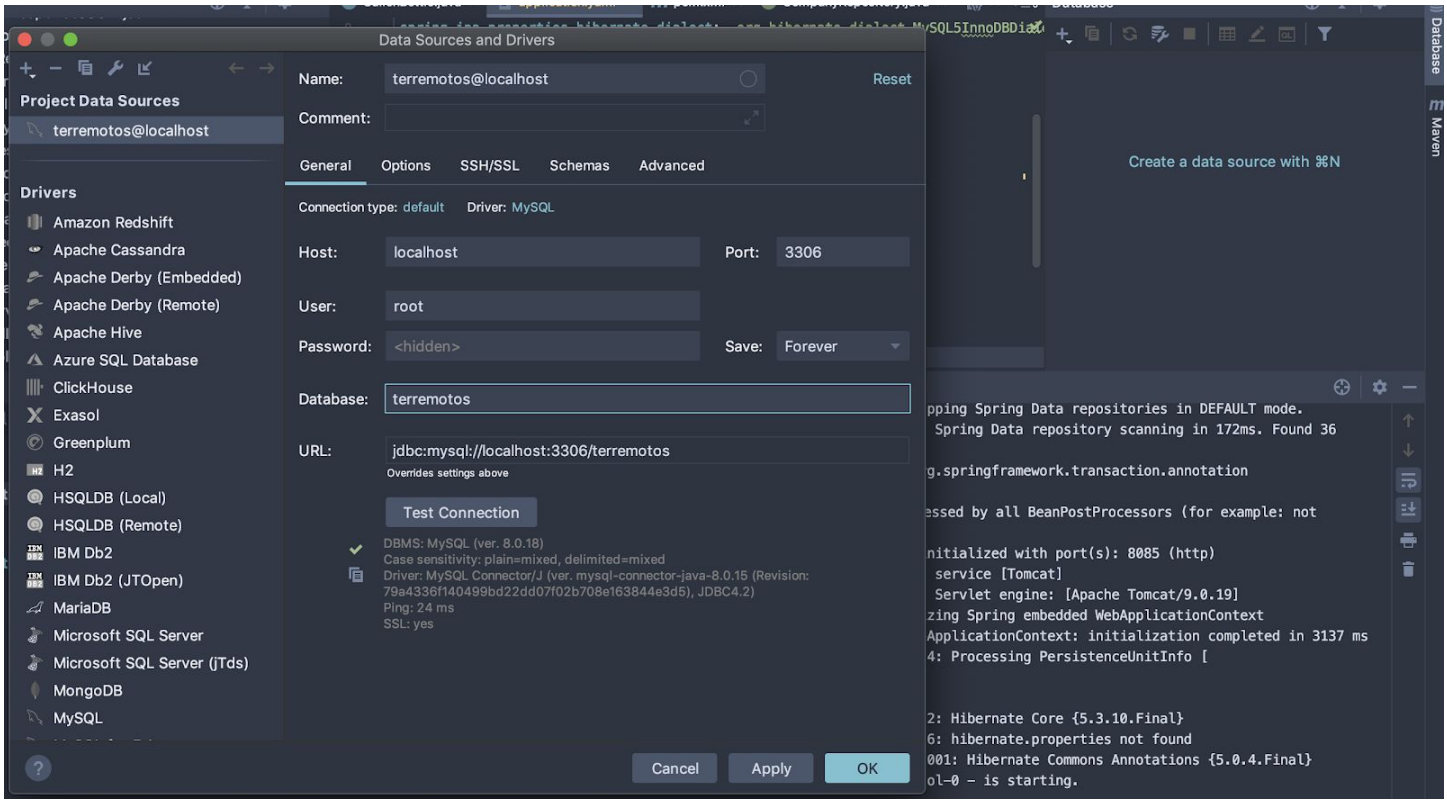

Illustration #2: The configuration needed to execute the program. A database named terremotos needs to be created on a MySQL Server. Located at port 3306 with both user and password being root.

API Documentation

Every api should provide a list of all end-points both for consuming and persisting data. Thus, for us it was important to provide a page where all endpoints could be found. After execution(running) of the project, the end-user can enter the following link in a website and see all available end-points and models. Which is the preferred way for entity modeling: http://localhost:8085/terremotos/api/swagger-ui.html#/

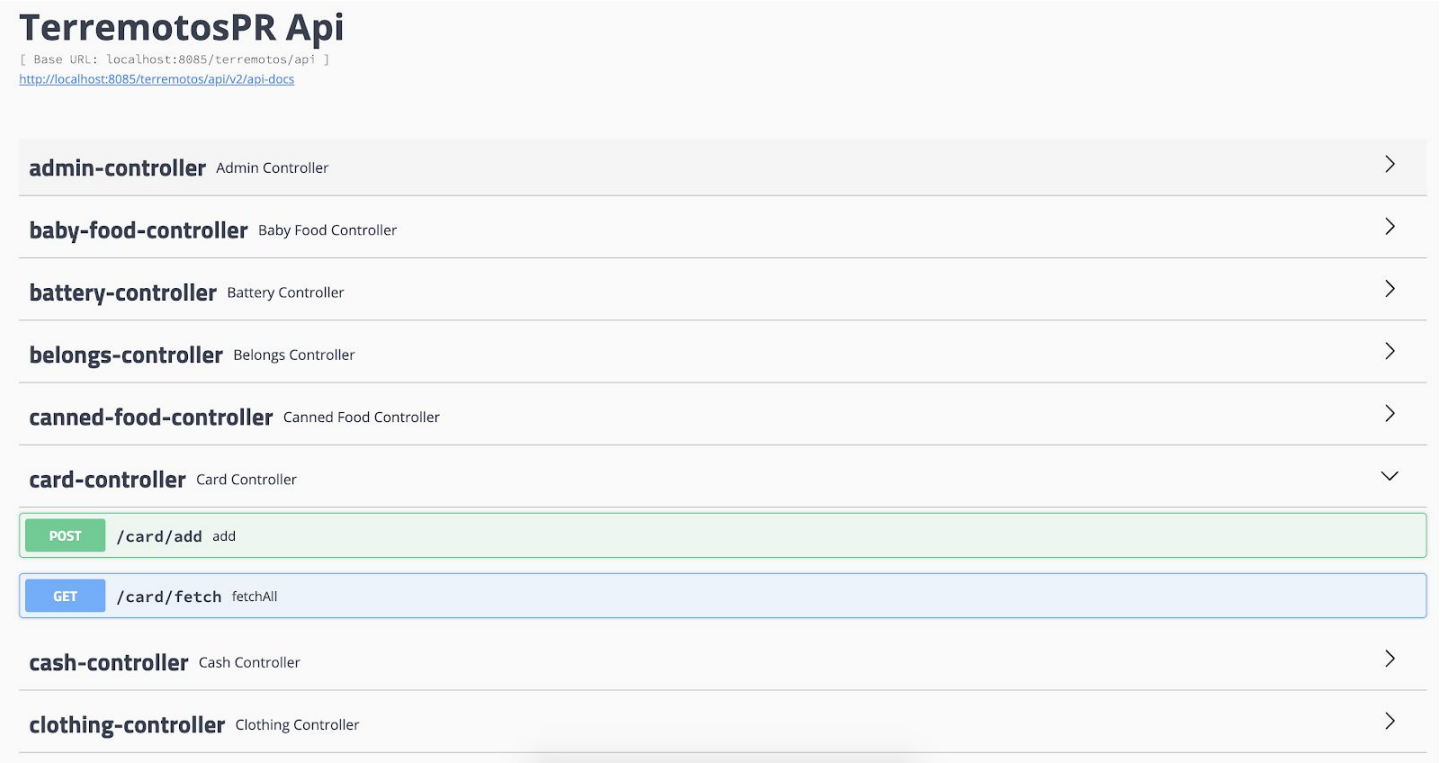

Illustration #2: List of all end-points available on the Rest Api, clicking each end-point provides data of parameters it accepts as well as response codes.

Programming language and significant dependencies

Programming language:

- Java SE 8

Dependencies:

- SpringBoot 2.1.5 for application and service creation
- SpringBoot Data JPA 2.1.5 for datasource persistence and repository management
- HikariCP 2.4.3 for JDBC connection pooling
- MySQL Connector 2.1.5 for JDBC drivers

ER Description

After consideration, the specified functionality can be achieved with the following entities. Each entity is accompanied by a short description of what it represents and its attributes.

User Entity: The User entity is the parent class of all other user-based entities. Provides the common attributes of all users and more specialized entities can inherit from it. The entity is composed of the following attributes:

- id - represents identifier of a user and primary key
- email - represents the email address for the particular user.
- Name: a composite attribute representing a user's complete name with the attributes:
    - first_name - represents the first name of an user.
    - last_name - represents the last name of a user.
- Address - a composite attribute representing an user's address, with the attributes:
    - line_1 - represents the main address line of the particular user.
    - line_2 - represents the secondary address line of the particular user.
    - city - represents the city the user's located in.
    - country - represents the country the user's located in.
    - region - represents the region the user's city is located in.
    - zipcode - represents the zipcode the user's city is located in.

Admin Entity: Represents a special type of user that has privileges to modify application content. It inherits from the user entity, thus it's to be understood that it possesses all the fields of User.

Consumer Entity: Represents a special type of user that has the need to acquire resources. Consumer is a User entity, therefore, it inherits the User entity's attributes. In addition it has the following attributes:

- type - represents the consumer's type, in case the consumer could qualify for a discount. The possible types are civilian, veteran, elderly or disabled.
- payment_method - is a multi-valued attribute that represents the payment methods owned by the consumer.
- phone - is a multi-valued attribute that represents the phone numbers corresponding to a consumer.

Supplier Entity: Represents a special type of user that owns or supplies resources. Supplier is a User entity, therefore, it inherits the User entity's attributes. In addition it has the following attributes:

- comp_id - this represents a one-to-one relationship between company and supplier called **Works for,** if the supplier works for a company. This attribute is a foreign key which references the id from a company.
- position - he supplier's position in the company if the supplier works for a company.

Manages Entity: Represents the many-to-many relationship between the admins and the users' accounts that are managed. The primary key of this entity is composed of the primary key of the user (User's id) and the primary key of the admin user (Admin's id). The attribute describing this relationship is status, where the admin can indicate if the user's account status is active, inactive or archived.

Company Entity: Represents the possible workplace of a supplier. The relationship **Work for** is one-to-many since one company can have many suppliers related to it. We decided to relate a company to a supplier by foreign key. An independent supplier wouldn't have a company related to them. The attributes that composed is entity are:

- company_id - the company's identifier (used as primary key)
- cName - the company's name
- cLocation - the company's location

Supplies Entity: Represents the many-to-many relationship between supplier and resources. The primary key of this entity is composed of the primary key of the resource (resource_id) and the primary key of the supplier (supplier_id).

- stock - amount supplied from the specified resource

Phone Entity: Represents a user's phone number, every registered user should have a phone number associated to its account. The phone is represented as an entity instead of an attribute of User because there are scenarios when we want to store more than one number. The entity is composed of the following fields.

- phone_id - the natural identifier of the entity, represents the record number (primary key)
- phone - a string representation of the user's phone.
- consumerId - the consumer id associated to the phone number

PaymentMethod Entity: Represents a user's payment method towards a particular payment transaction on the application. The entity contains the following fields.

- pm_id - identifies uniquely the payment method in the database (primary key)
- consumerId - the id of the user effectuating a payment.

PayPal Account Entity: Represents a payment method available to pay for a resource. This entity inherits from PaymentMethod, therefore all attributes in PaymentMethod are available for this entity. Its primary key is given by the payment method's primary key (pm_id). The attribute describing this entity is pp_account, which represents the username or account number from PayPal.

Cash Entity: Represents a payment method available to pay for a resource. Cash entity inherits from PaymentMethod, therefore all attributes in PaymentMethod are available for this entity. Its primary key is given by the payment method's primary key (pm_id). The attribute describing this entity is amount_available, which represents the amount of cash the user has available.

Card Entity: Represents a payment method available to pay for a resource. Card entity inherits from PaymentMethod, therefore all attributes in PaymentMethod are available for this entity. Its primary key is given by the payment method's primary key (pm_id). The attribute card_number, represents the card number (Visa, MasterCard, etc.). The other attribute card_expdate attribute represents the expiration date on the card.

Payment Entity: Represents the payment billed to a consumer. Its primary key is called payment_id. The consumer's primary key (Consumer's id) is added as foreign key to identify the owner of the payment. This represents **Owns** (the relationship one-to-many with Total Participation). The other attribute of this entity is purchase_total, which indicates the total price of the resources ordered.

Pays Entity: Represents the one-to-many relationship between the payment and the orders to be paid. The primary key of this entity is composed of the primary key of the payment (payment_id) and the primary key of the order placed (order_id). The other attribute is payment_date, which saves when an order was paid.

PlacedOrder Entity: Represent a user's order to acquire certain resource(s). The entity contains the following elements. This keeps a foreing key in **Belongs** to maintain ownership of the order.

- order_id - represents the natural identifier of a record in the table.
- date - a representation of when was the order placed.
- consumerId - keeps record of the owner of the order, the client that made the order.

Reservation Entity: Represents a reservation of resources placed by a user. Its primary key is called reservation_id. The consumer's primary key (id) is passed as a foreign key in this entity to represent the relationship **Places**. This relationship is one-to-many with total participation in the many side, since a consumer can place many reservations. The other attribute in this entity is reservation_date, which represents the date a reservation was placed.

Reserves Entity: Represents the one-to-many relationship between a reservation and the resources that can be reserved. The primary key of this entity is composed of the primary key of the reservation (reservation_ id) and the primary key of the resource (rid). The attribute describing this entity is rquantity, which indicates the quantity of a resource that will be reserved if possible.

Belongs Entity: Represents the many-to-many relationship between the resources and the orders placed. The primary key of this entity is composed of the primary key of the resource (resource_id) and the primary key of the order placed (order_id).

- quantity - amount of resources
- final_price - final price of order, after tax and any discount

Base Resources Entity: Resource entity is the parent class of all other resource type entities. It provides the common attributes of all resources and more specialized entities can inherit from it. Its attributes are:

- id - resource identifier and primary key
- available - indicates whether the resource is available or not
- brand - resource brand
- name - resource name
- category - indicates what type of resource and to which category belongs
- description - brief description of what the product is or does
- Location - a composite attribute that indicates the resource's location by coordinates, with the attributes:
    - latitude - latitude of the location
    - longitude - longitude of the location
- price - resource listed price

Water Entity: Represents the water resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- type - represents the type of water. It can be purified, alkaline, mineral or spring.
- potable - to indicate if the water is potable or not
- packaged_quantity - defines the quantity of individual units packaged

Small Bottle Entity: Represents the small water bottles resource. It's a Water entity, therefore, it inherits the Water entity' attributes. Its primary key equals the Resource entity's primary key (rid). The other attribute of this entity is the size of the bottles in ounces.

Gallon Bottle Entity: Represents the gallon bottle of water resource. It's a Water entity, therefore, it inherits the Water entity's attributes. Its primary key equals the Resource entity's primary key (rid).

Fuel Entity: Represents the fuel resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attribute is type, which represents the types of fuels.

Gasoline Entity: Represents the gasoline resource. This entity inherits its attributes from the entity Fuel, which inherits from the Resource entity. Therefore, its primary key is given by the resource's primary key (rid). The other attributes that compose this entity are: type (Regular or Premium), octane and size(in liters).

Diesel Entity: Represents the diesel resource. This entity inherits its attributes from the entity Fuel, which inherits from the Resource entity. Therefore, its primary key is given by the resource's primary key (rid). The other attribute that composes this entity is size(in liters).

Propane Entity: Represents the propane resource. This entity inherits its attributes from the entity Fuel, which inherits from the Resource entity. Therefore, its primary key is given by the resource's primary key (rid). The other attribute that composes this entity is size(in pounds).

Canned Food Entity: Represents the canned food resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- type - represents the type of canned food. It can be fruit, vegetable, meat, soup, beans, milk or pasta
- size - size of the can in ounces
- exp_date - the expiration date of the canned food

Dry Food Entity: Represents the dry food resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- type - represents the type of dry food. It can be meat, fish, vegetables, fruits, nuts or pasta.
- size - size of dry food bag or container
- exp_date - the expiration date of the dry food

Baby Food Entity: Represents the dry food resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- flavor - represents the flavor of the baby food. It can be fruit, vegetable, variety or other.
- size - size of baby food container
- exp_date - the expiration date of the baby food

Ice Entity: Represents the ice resource and inherents its attributes from the Resource entity. Therefore, its primary key is given by the resource's primary key(rid). Its other attributes are the size in pounds.

Battery Entity: Represents a battery, it inherits all attributes from the base resource, but includes an attribute to identify which type of battery the record references, this attribute is given as an enumeration value.

Tool Entity: Represents a tool resource, in addition to all the inherited fields from the Resource the entity possesses the following attributes not inherited from the base resource.

- type - An Enumeration representing the tool type.
- weight - a representation of the tool's weight.

Heavy Equipment Entity: represents the heavy equipment resource. It inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). The other entity attributes are

- type - represents the type of heavy equipment. It can be kitchenware, furniture, canopy or camping gear.
- size - size of equipment in pounds

Power Generator Entity: Represents the power generator resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- type - represents the type of the generator. It can be an inverter, portable, house, solar, battery or quiet.
- power - power it generates in Watts
- fuel_type - represents the type of fuel the generator uses. It can be gasoline, dual fuel, propane or diesel.

MedicalDevices Entity: Represents the medical device resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- type - represents a type of the medical device. It can be a respirator, pacemaker, infusion pump, thermometer, gloves, bandages, alcohol, mask, catheter and others.

Medications Entity: Represents the medications resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- medType - represents a type of the medication
- size - represents the size or quantity of a medication depending on the dosage form
- dosageForm - represents the dosage form. Dosage examples are: tablets, gel, cream, liquid.

Clothing Entity: Represents the medications resource. This entity inherits its attributes from the entity Resource, therefore, it's primary key is given by the resource's primary key (rid). Its other attributes are:

- size - represents the size of the clothes.
- material - represents cloth material, it can be for example cotton, jean, silk and others.
- type - represents the type of cloth. It can be a shirt, underwear, shorts, jackets, pants, dress and others.
- gender - it represents the gender of the item or if it is unisex.