iPasalWebService

iPasalWebservice is the REST API built using Spring framework.

# User Authentication and Authorization

Users are given 3 different roles in ipasal.

* Admin : responsible for overall management of data
* Data Admin: responsible for adding records into the application
* Customers: those who consumes the application resources

The Authentication and Authorization in iPasal is handled by Spring Security implementing JSON Web Token (JWT). iPasal has custom filters that handles the authentication and authorization. The JWTAuthenticatonFilter is responsible for the authentication process. During the authentication process, when the user successfully logs in using their credentials, a JSON Web Token is returned to the client. Client stores JSON Web Token locally. Whenever the client requests the protected resource, JWT is sent in authorization header using Barer Schema along with the request.

When iPasal gets the client request, JWT is validated before processing the request. The request is rejected if the validation fails. JWTAuthorizatonFilter is responsible for the authorization process in iPasal.

# Product Services

1. Add a new product
2. Accessing products
3. Updating product
4. Delete a product

# Category Services

1. Adding category

# Database

## Architecture

Tables

User: table that contains the user details

Username: user name used by the user

Password: hashed password of the user

f\_name: first name of the user

m\_name: middle name of the user

l\_name: last name of the user

email: email id of the user

phone: phone number of the user

role: the role specifies the role of the user

parent\_id: id of the Admin(parent) who has creared the data admin (child)

Role: specifies the role of the user

role\_id: id of the role

role\_name: role name

Products: contains the details about a product

Product\_id: id of a particular product

Product\_name: name of the product

Unit: unit of each product

Rate: cost of the product per unit

Available\_items : total number of items that are available in our inventoty

Description: short description of the product

User\_id: the id of the user(usually data admin) who added the product

Deleted: flag that represents that the item is deleted or not

0 => not deleted

1 => deleted

Featured : flag represents if the product is featured in the home page or not

0 => not featured

1 => featured

Category: specifies the category of a product

Category\_id: id of category

Category\_name: name of the category

Parent\_id: parent category id

Deleted : flag that represents the deleted status of category

0 => not deleted

1 => deleted

Featured: Flag that represents the featured status of the catgory

0 => not featured

1 => featured in the home page

(note: based on our template, only 5 categories can be set featured at once)

Product\_category: relation between the product and category

(defines the products of a specific category)

Product\_id: id of the product

Category\_id: id of category

Product\_images: it contains the images for a product

Image\_id : primary id for the image

Product\_id : id of the product

Image: image name

Customers: contains the details about a customer ( customer who has signed up for making order)

customer\_id : primary id of the customer

username: username of customer

password: hashed password of customer

first\_name: customer’s first name

middle\_name: customer’s middle name

last\_name: customer’s last name

address: customer’s address

city: customer’s city

email\_address: customer’s email address

phone: contact number of customer

role\_id: role of the customer (always 3)

Orders : contains the orders information

( an order id is generated for a particular checkout, an order may contains multiple products information)

Order\_id : primary key for each order

Order\_date : date of order

Shipping\_details\_id : id of the shipping details

Payment\_details\_id: id of the payment details

Order\_status: contains the id that defines the order status

Ordered\_by : id of the customer who has placed the order

Order\_details: It contains the product information of the order

Order\_details\_id: primary key for order detail

Product\_id: id of the product

Order\_id : id of the order to which the product is related to

Quantity: quantity of the products ordered

Order\_status : status of the order

Status\_id: status reference

Status: status name

Payment\_details: contains the details of payment

Payment\_details\_id : primary key for each payment

Payment\_type: type of the payment

Card\_number: credit card number in case of payment by credit card/visa card

Status: defined if the payment is pending or terminated

0 => pending

1=> completed

Payment\_type: contains the type of the payment

Payment\_id: payment reference

Payment\_name: payment name

Shipping\_details: ontains the shipping address where the ordered product has to be delivered.

Shipping\_details\_id: primary key of shipping details

Address: shipping address

City: city for shipping

Zipcode: zip code of the shipping city

Phone: contact number of receiver

Shipping\_id: shpping type id

Customer\_id: id of customer who has ordered the item

Shipping\_type: type of shipping that determines the time (days) of delivery

Shipping\_id: id of shipping type

Type: type name

Approx.\_days: approximation days required for delivery