**Return Order Management System**

1. Component-processing-service
   1. Checks for Repair or Replacement
2. Packaging-Delivery-service
   1. Determine the packaging and delivery charge
3. Authorization-service
   1. JSON Web Token (JWT) authorization
4. Return-order-service
   1. Login
   2. Provide detail for return order
   3. View
   4. Confirm processing



### **Return-Order Microservice**

**ProcessRequest— Name {String username}, ContactNumber { long userContactNumber },**

**DefectiveComponentDetail {**ComponentType (Integral / Accessory), String ComponentName, int quantity}, Date systemDate;

**ProcessResponse ---- RequestId** {String requestId}**, ProcessingCharge**

{float processingCharge} **PackagingAndDeliveryCharge{**float packageAndDeliveryCharge} **DateOfDelivery** dateOfDelivery **;**

**REST End Points**

* + GET: http://localhost:port/ProcessDetail (Input: ProcessRequest | Output: ProcessResponse)
  + POST: http://localhost:port/CompleteProcessing (Input: RequestId, CreditCardNumber, CreditLimit, ProcessingCharge) | Output: string response of the success of operation)

### **Package-Delivery Microservice**

**PackagingItem— {int item, int accesory, int protectiveSheath}**

**DeliveryItem ---- {float deliveryChargeForItem, float deliveryChargeForAccessory}**

**REST End Points**

* + GET: http://localhost:port/ GetPackagingDeliveryCharge (Input: ComponentType, Count | Output: PacakagingAndDeliveryCharge )

**packageAndDeliveryCharge**

Input ---if(accessories){

Return (Price\_of\_accesorries \* quantity) + delivery\_charge\_of\_accessories;

}else if(integral){

Return (Price\_of\_interal \* quantity) + delivery\_charge\_of\_integral;

}

**ProcessingCharge**

GenericInterface

Integral/Repair implements GenericInterface{

1 day = 100

5 days = 500;

}

Accesory/Replacement implements GenericInterface{

1 day = 150;

2 days = 300;

}

**DeliveryDate**

if(accessory){

date +=2days;

}else if(integral){

Date +=5 days;

}

**RequestId**

Generate random id;

### **Authorization Microservice**

|  |  |
| --- | --- |
| Return Order Management System | Authorization Microservice |
| **Security Requirements**   * Create JWT * Have the token expired after specific amount of time say 30 minutes * Has anonymous access to get the token detail | |

### **Return-Order Microservice**

POST -- <http://localhost:8080/ROM/auth/signup>

ReqBody-

{

“username”:”Jahnabi”,

“email”:[jahnabi@gmail.com](mailto:jahnabi@gmail.com),

“password”:”pass123”

}

Response –

{

“errorCode”:”0”,

“errorMessage”:” Suggessful”

}

POST -- [http://localhost:8080/ ROM /auth/signin](http://localhost:8080/PMS/auth/signin)

ReqBody-

{

“username”:”Jahnabi”,

“password”:” pass123”

}

Response –

{

“username”:”jahnabi”,

“email”:[jahnabi@gmail.com](mailto:jahnabi@gmail.com),

“roles”:[“ROLE\_USER”],

“accessToken”:” eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJQYXJpd2VzaDEiLCJpYXQiOjE2NTE4MTAwMjcsImV4cCI6MTY1MTgxMDA3N30.NYDq5NttSvIx1N\_-h46B29zo5O0RhVxBOYbzyIV0c9l1TpIx-pci2l8J5uxNNEVwz-emXkan9Lo285h-epL0EQ”,

“tokenType”:” Bearer”

}