



UNIVERSIDADE D
COIMBRA

FACULDADE
DE CIÊNCIAS
E TECNOLOGIA

DEPARTAMENTO
DE ENGENHARIA
INFORMÁTICA

Digital Bubble

User Manual

Databases project, 3rd year, 2nd semester

André Filipe Costa Colaço, 2020220301, uc2020220301@student.uc.pt

Sancho Amaral Simões, 2019217590, uc2019217590@student.uc.pt

20 de maio de 2022



To test the *REST API*, you can use a *REST* client such as *postman*, as mentioned in the installation manual.

Endpoints available:

- *User*

- **Registration**

Anyone can register as a buyer, “role”: 2. But only admins can register as admin or seller, “role”: 0 and “role”: 1, respectively.

```
req  POST    127.0.0.1:8080/digitalbubble/user
{
  "user_name": username,
  "password": password,
  "first_name": first name,
  "last_name": last name,
  "email": email,
  "phone_number": phone number,
  "tin": tin,
  "state": state,
  "city": city,
  "zip_code": zip code,
  "street_name": street name,
  "house_no": house number,
  "role": role number
}
res {user_id} or if an error occurs {"error": error description}
```

- **Login**

After a successful login, it is necessary to include the obtained token in the section “*Authorization/Bearer Token*” of the *HTTP* request.

```
req  PUT      127.0.0.1:8080/digitalbubble/user
{
  "user_name": username,
  "password": password
}
res {token} or if an error occurs {"error": error description}
```



- **Product**

- **Create new product**

Only sellers can create a new product. There are three types of products: computer “type”: 0, television “type”: 1 and smartphone “type”: 2.

```
req    POST    127.0.0.1:8080/digitalbubble/product
{
    "name": product name,
    "price": price,
    "stock": stock,
    "description": description,
    "category": category,
    "type": type, (Not necessary for a generic product)
    "cpu": computer cpu, (computer)
    "gpu": computer gpu, (computer)
    "size": television size, (television)
    "technology": technology, (television)
    "model": model, (smartphone)
    "operative_system": system (smartphone)
}
res {product_id} or if an error occurs {"error": error description}
```

- **Update existing product**

Updates a product with a new version.

```
req    PUT    127.0.0.1:8080/digitalbubble/user/{product_id}
{
    "name": product name,
    "price": price,
    "stock": stock,
    "description": description,
    "category": category,
    "cpu": computer cpu, (computer)
    "gpu": computer gpu, (computer)
    "size": television size, (television)
    "technology": technology, (television)
    "model": model, (smartphone)
    "operative_system": system (smartphone)
}
res if an error occurs {"error": error description}
```



- **Order**

- **Make purchase**

Notifies buyer and seller when a purchase is fully processed.

```
req    POST    127.0.0.1:8080/digitalbubble/order
res
{
    "cart": [product1_id, quantity1],
            [product2_id, quantity2],
            ...,
            [productN_id, quantityN]
}
res {order_id} or if an error occurs {"error": error description}
```

- **Leave rating/feedback**

Leave textual feedback and a rating on a product that you bought.

```
req    POST    127.0.0.1:8080/digitalbubble/rating/{product_id}
res
{
    "rating": rating,
    "comment": comment
}
res if an error occurs {"error": error description}
```

- **Leave comment/question**

Leave a comment or a question on a product that you bought.

```
req    POST    127.0.0.1:8080/digitalbubble/questions/{product_id}
POST    127.0.0.1:8080/digitalbubble/questions/{product_id}/
            {parent_question_id}
{
    "question": question
}
res {question_id} or if an error occurs {"error": error description}
```



- **Information**

- **Consult product information**

Get the generic information, price history, average rating and comments of a product.

req **GET** 127.0.0.1:8080/digitalbubble/rating/{product_id}

res

```
{
  "name": name,
  "description": description,
  "category": category,
  "prices": [current_price_date - current_price, prev_price_date
prev_price, (...)],
  "rating": average rating,
  "comments": [comment1, comment2, (...)]
}
```

or if an error occurs {"error": error description}

- **Statistics for the last 12 months**

Get the sales details by month, over the last 12 months.

req **GET** 127.0.0.1:8080/digitalbubble/report/year

res

```
{
  "month": month,
  "orders": number of orders,
  "total_values": total value
}
```

or if an error occurs {"error": error description}

- **Notification**

- **Notifications for the user**

Get all the notifications

req **GET** 127.0.0.1:8080/digitalbubble/notifications

res

```
{
  [
    {
      "description": description,
      "id": id,

```



```
    "title": title,  
    "user_id": user_id},  
  
    (...)  
  ]  
}
```

or if an error occurs {"error": error description}