

Bài 3. Trao đổi dữ liệu sử dụng giao thức AMQP (RabbitMQ)

Yêu cầu. Viết chương trình (bằng 1 ngôn ngữ tùy chọn: C#, Java, python) thực hiện

- Producer: gửi dữ liệu qua một RabbitMQ broker. Cài đặt RabbitMQ broker trên máy local.
- Đóng gói dữ liệu bằng JSON. Ví dụ:

```
{ "id":11, "packet_no":126, "temperature":30, "humidity":60, "tds":1100, "pH":5.0 }
```
- Consumer: nhận dữ liệu từ broker, bóc tách các trường và hiển thị dữ liệu nhận được.

Yêu cầu nộp bài:

Viết báo cáo mô tả ngắn gọn gồm: code chính của chương trình, *chụp ảnh màn hình thực hiện chương trình và kết quả ở các yêu cầu.*

Submit: file báo cáo, file code

Bài làm

Source code:

```

producer.py X
producer.py > ...
1  import pika
2  import json
3
4  MESSAGE = {
5      "id":11,
6      "packet_no":126,
7      "temperature":30,
8      "humidity":60,
9      "tds":1100,
10     "pH":5.0
11 }
12 EXCHANGE_NAME = 'sensor_data'
13
14 connection = pika.BlockingConnection(pika.ConnectionParameters(host='localhost'))
15 channel = connection.channel()
16 channel.exchange_declare(exchange=EXCHANGE_NAME, exchange_type='fanout')
17 message = json.dumps(MESSAGE)
18 channel.basic_publish(exchange=EXCHANGE_NAME, routing_key='', body=message)
19
20 print(" [x] Sent %r" % message)
21 connection.close()

```

```

producer.py  receiver.py X
receiver.py > ...
1  import pika
2
3  EXCHANGE_NAME = 'sensor_data'
4
5  connection = pika.BlockingConnection(pika.ConnectionParameters(host='localhost'))
6  channel = connection.channel()
7  channel.exchange_declare(exchange=EXCHANGE_NAME, exchange_type='fanout')
8  result = channel.queue_declare(queue='', exclusive=True)
9  queue_name = result.method.queue
10 channel.queue_bind(exchange=EXCHANGE_NAME, queue=queue_name)
11 print(' [*] Waiting for logs. To exit press CTRL+C')
12
13 def callback(ch, method, properties, body):
14     print(" [x] %r" % body)
15
16 channel.basic_consume(queue=queue_name, on_message_callback=callback, auto_ack=True)
17 channel.start_consuming()

```

Kết quả:

```
PS G:\School\IoT\Homework\Homework3> python producer.py
[x] Sent '{"id": 11, "packet_no": 126, "temperature": 30, "humidity": 60, "tds": 1100, "pH": 5.0}'
PS G:\School\IoT\Homework\Homework3> python producer.py
[x] Sent '{"id": 11, "packet_no": 126, "temperature": 30, "humidity": 60, "tds": 1100, "pH": 5.0}'
PS G:\School\IoT\Homework\Homework3> 
```

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

```
PS G:\School\IoT\Homework\Homework3> python receiver.py
[*] Waiting for logs. To exit press CTRL+C
[x] b'{"id": 11, "packet_no": 126, "temperature": 30, "humidity": 60, "tds": 1100, "pH": 5.0}'

```