Arquivo para Configuração do RabbitMQ no projeto Java.  
1) Colocar o arquivo no pacote Configuration do projeto Java

import org.springframework.amqp.core.AmqpAdmin;  
import org.springframework.amqp.core.Binding;  
import org.springframework.amqp.core.DirectExchange;  
import org.springframework.amqp.core.Queue;  
import org.springframework.amqp.rabbit.connection.ConnectionFactory;  
import org.springframework.amqp.rabbit.core.RabbitTemplate;  
import org.springframework.amqp.support.converter.Jackson2JsonMessageConverter;  
import org.springframework.amqp.support.converter.MessageConverter;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.context.annotation.Bean;  
import org.springframework.stereotype.Component;  
import jakarta.annotation.PostConstruct;  
  
@Component  
public class MQConfig {  
 @Autowired  
 private AmqpAdmin amqpAdmin;  
 private Queue queue;  
 private Queue queue (String queueName){  
 return new Queue(queueName, true, false, false);  
 }  
 private DirectExchange createDirectExchange(){  
 return new DirectExchange("ecommercermq");  
 }  
  
 @PostConstruct  
 private void Create (){  
 this.queue = new Queue("fila-ecommerce");  
 // Create the direct exchange  
 DirectExchange directExchange = createDirectExchange();  
 // Create the binding  
 Binding binding = new Binding(queue.getName(), Binding.DestinationType.*QUEUE*,  
 directExchange.getName(), queue.getName(), null);  
 amqpAdmin.declareQueue(queue);  
 amqpAdmin.declareExchange(directExchange);  
 amqpAdmin.declareBinding(binding);  
 }  
  
 @Bean  
 public RabbitTemplate rabbitTemplate(ConnectionFactory connectionFactory,  
 MessageConverter jsonMessageConverter) {  
 RabbitTemplate template = new RabbitTemplate(connectionFactory);  
 template.setMessageConverter(jsonMessageConverter);  
 return template;  
 }  
  
 @Bean  
 public MessageConverter jsonMessageConverter() {  
 return new Jackson2JsonMessageConverter();  
 }  
}

2) Adicionar a seguinte dependência:

<dependency>  
 <groupId>org.springframework.amqp</groupId>  
 <artifactId>spring-rabbit</artifactId>  
</dependency>  
  
No service escolhido adicionar:  
  
@Autowired  
private RabbitTemplate rabbitTemplate;

.....

.....

.....

rabbitTemplate.convertAndSend("fila-ecommerce", objetoJava);

3) Colocar os parâmetros de configuração no application.properties

spring.rabbitmq.host=10.136.38.50  
spring.rabbitmq.username=admin  
spring.rabbitmq.password=admin  
spring.rabbitmq.port=5672  
spring.rabbitmq.virtual-host=tvecommercermq

4) Atenção!!  
O objeto a ser colocado na fila precisa estar serializado

public class RegistroPagamento implements Serializable {  
     private static final long serialVersionUID = 1L;

........

5) Dentro da classe service para poder receber o objeto da mensagem adicionar a seguinte linha (repare que o objeto na passagem de parametro do método deve ser igual à estrutura do objeto recebido na fila):

@RabbitListener(queues = "fila-ecommerce")  
private void receberdaFila(RegistroPagamento registroPagamento){  
 System.out.println(registroPagamento);  
}