SQS Queue Troubleshooting Guide

Common Error: "The specified queue does not exist"

This error means the SQS queue hasn't been created yet. The updated code now handles this automatically!

Solution 1: Automatic Queue Creation (Recommended)

The application is now configured to **automatically create the queue** if it doesn't exist.

Configuration

In application.yml:

```
aws:
  sqs:
    work-order-queue: work-order-queue
    auto-create-queue: true  # ✅ Enabled by default
```

What Happens on Startup

1. Application starts
2. `@PostConstruct` initialize() runs
3. Checks if queue exists
4. If not found → Creates it automatically
5. Caches the queue URL for future use

Logs You'll See

PROF

```
INFO  - Initializing SQS queue connection for: work-order-queue
WARN  - Queue 'work-order-queue' does not exist
INFO  - Creating new queue: work-order-queue
INFO  - Successfully created queue: work-order-queue with URL:
https://sqs.us-east-1.amazonaws.com/.../work-order-queue
INFO  - Successfully initialized queue URL: https://...
```

Solution 2: Manual Queue Creation

If you prefer to create the queue manually:

Option A: AWS Console

1. Go to AWS SQS Console

2. Click "Create queue"
3. Choose "Standard Queue"
4. Name it: `work-order-queue`
5. Keep default settings
6. Click "Create queue"

Option B: AWS CLI

```
# Create the queue
aws sqs create-queue --queue-name work-order-queue

# Verify it was created
aws sqs list-queues | grep work-order-queue

# Get the queue URL
aws sqs get-queue-url --queue-name work-order-queue
```

Option C: Terraform

```
resource "aws_sqs_queue" "work_order_queue" {
  name                       = "work-order-queue"
  delay_seconds              = 0
  max_message_size           = 262144
  message_retention_seconds = 345600
  receive_wait_time_seconds = 20

  tags = {
    Environment = "production"
    Service     = "billing-service"
  }
}
```

PROF

Troubleshooting Steps

1. Verify AWS Credentials

```
# Check if credentials are set
echo $AWS_ACCESS_KEY
echo $AWS_SECRET_KEY

# Test AWS CLI works
aws sts get-caller-identity

# List SQS queues
aws sqs list-queues
```

2. Check Application Logs

Look for these log messages:

✅ **Success:**

```
INFO  - Successfully initialized queue URL: https://sqs.us-east-
1.amazonaws.com/.../work-order-queue
DEBUG - Polling messages from SQS queue: work-order-queue
```

❌ **Queue doesn't exist:**

```
WARN  - Queue 'work-order-queue' does not exist
INFO  - Creating new queue: work-order-queue
```

❌ **Permission denied:**

```
ERROR - User: arn:aws:iam::123456789012:user/myuser is not authorized to
perform: sqs:GetQueueUrl
```

3. Verify IAM Permissions

Your AWS user/role needs these permissions:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sqs:GetQueueUrl",
        "sqs:CreateQueue",
        "sqs:ReceiveMessage",
        "sqs:SendMessage",
        "sqs:DeleteMessage",
        "sqs:GetQueueAttributes"
      ],
      "Resource": "*"
    }
  ]
}
```

PROF

4. Check Region Configuration

Make sure your queue is in the same region as configured:

```
aws:
  region: us-east-1  # ⚠ Must match where you created the queue
```

List queues in specific region:

```
aws sqs list-queues --region us-east-1
```

5. Test Queue Manually

Send a test message via API:

```
curl -X POST http://localhost:8080/api/test/send-test-message
```

Send via AWS CLI:

```
# Get queue URL
QUEUE_URL=$(aws sqs get-queue-url --queue-name work-order-queue --query
'QueueUrl' --output text)

# Send message
aws sqs send-message \
  --queue-url $QUEUE_URL \
  --message-body '{
    "workOrderId": "70e6215d-b5c6-4896-987c-144a6333a2b3",
    "customer": {
      "Id": "70e6215d-b5c6-4896-987c-144a6312a2b3",
      "name": "Test User",
      "email": "test@example.com"
    },
    "vehicle": {
      "Id": "80e6215d-b5c6-4896-987c-144a6312a2b4",
      "make": "Toyota",
      "model": "Camry",
      "year": 2020
    },
    "parts": [],
    "services": []
  }'
```

PROF

Advanced Configuration

Disable Auto-Create

If you want to manually manage queues:

```
aws:
  sqs:
    auto-create-queue: false
```

Set Queue Attributes

If you create the queue manually, consider these settings:

```
aws sqs set-queue-attributes \
  --queue-url $QUEUE_URL \
  --attributes '{
    "VisibilityTimeout": "300",
    "ReceiveMessageWaitTimeSeconds": "20",
    "MessageRetentionPeriod": "345600"
  }'
```

Add Dead Letter Queue (DLQ)

For production, add a DLQ for failed messages:

```
# Create DLQ
aws sqs create-queue --queue-name work-order-queue-dlq

# Get DLQ ARN
DLQ_ARN=$(aws sqs get-queue-attributes \
  --queue-url $(aws sqs get-queue-url --queue-name work-order-queue-dlq
--query 'QueueUrl' --output text) \
  --attribute-names QueueArn \
  --query 'Attributes.QueueArn' \
  --output text)

# Configure redrive policy
aws sqs set-queue-attributes \
  --queue-url $QUEUE_URL \
  --attributes "{
    \"RedrivePolicy\": \"
{\\\"deadLetterTargetArn\\\":\\\"$DLQ_ARN\\\",\\\"maxReceiveCount\\\":\\
\"3\\\"}\"
  }"
```

Environment-Specific Configuration

Development (LocalStack)

Use LocalStack for local development:

```
aws:
  region: us-east-1
  credentials:
    access-key: test
    secret-key: test
  endpoint: http://localhost:4566  # LocalStack endpoint
  sqs:
    work-order-queue: work-order-queue
    auto-create-queue: true
```

Production

```
aws:
  region: us-east-1
  credentials:
    access-key: ${AWS_ACCESS_KEY}
    secret-key: ${AWS_SECRET_KEY}
  sqs:
    work-order-queue: prod-work-order-queue
    auto-create-queue: false  # Create manually for production
```

Error Messages and Solutions

Error	Cause	Solution
`QueueDoesNotExist`	Queue not created	Enable `auto-create-queue: true` or create manually
`AccessDenied`	No SQS permissions	Add IAM permissions
`InvalidClientTokenId`	Wrong AWS credentials	Verify AWS_ACCESS_KEY and AWS_SECRET_KEY
`SignatureDoesNotMatch`	Wrong secret key	Double-check AWS_SECRET_KEY
`Unknown queue`	Wrong region	Check region in config matches queue location

Health Check

Verify everything is working:

```
```

```
# Check service is running
curl http://localhost:8080/api/health

# Check application logs
tail -f logs/spring.log | grep -i sqs

# Monitor queue in real-time
watch -n 2 'aws sqs get-queue-attributes \
  --queue-url $(aws sqs get-queue-url --queue-name work-order-queue --
query QueueUrl --output text) \
  --attribute-names All'
```

Support

If you still have issues:

1. ✅ Check all credentials are set
2. ✅ Verify IAM permissions
3. ✅ Ensure region is correct
4. ✅ Look at application logs
5. ✅ Try manual queue creation
6. ✅ Test with AWS CLI first

The auto-create feature should handle most scenarios automatically! 🎉