Recall from Section 3.1 that SQS is a system for supporting automated workflows; multiple components

can communicate with messages sent and received via SQS . An example showing the use of message

queues is presented in Section 4.7. Figure 11.8 shows the actions available for a given queue in SQS.

The following steps can be used to create a queue, send a message, receive a message, and delete a

message, and delete the queue in C#:

1. Authenticate an SQS connection:

NameValueCollection appConfig =

ConfigurationManager.AppSettings;

AmazonSQS sqs = AWSClientFactory.CreateAmazonSQSClient

(appConfig["AWSAccessKey"], appConfig["AWSSecretKey"]);

2. Create a queue:

CreateQueueRequest sqsRequest = new CreateQueueRequest();

sqsRequest.QueueName = "MyQueue";

CreateQueueResponse createQueueResponse =

sqs.CreateQueue(sqsRequest);

String myQueueUrl;

myQueueUrl = createQueueResponse.CreateQueueResult.QueueUrl;

3. Send a message:

SendMessageRequest sendMessageRequest =

new SendMessageRequest();

sendMessageRequest.QueueUrl =

myQueueUrl; //URL from initial queue

sendMessageRequest.MessageBody = "This is my message text.";

sqs.SendMessage(sendMessageRequest);

4. Receive a message:

ReceiveMessageRequest receiveMessageRequest =

new ReceiveMessageRequest();

receiveMessageRequest.QueueUrl = myQueueUrl;

ReceiveMessageResponse receiveMessageResponse =

sqs.ReceiveMessage(receiveMessageRequest);

5. Delete a message:

DeleteMessageRequest deleteRequest =

new DeleteMessageRequest();

deleteRequest.QueueUrl = myQueueUrl;

deleteRequest.ReceiptHandle = messageRecieptHandle;

DeleteMessageResponse DelMsgResponse =

sqs.DeleteMessage(deleteRequest);

6. Delete a queue:

DeleteQueueRequest sqsDelRequest = new DeleteQueueRequest();

sqsDelRequest.QueueUrl =

CreateQueueResponse.CreateQueueResult.QueueUrl;

DeleteQueueResponse delQueueResponse =

sqs.DeleteQueue(sqsDelRequest);