

Zadatak A – Rešenje.

Listing1 – *QueueHelper* implementacija.

```
namespace StudentService_Data
{
    public class QueueHelper
    {
        public static CloudQueue GetQueueReference(String queueName)
        {
            CloudStorageAccount storageAccount =
CloudStorageAccount.Parse(CloudConfigurationManager.GetSetting("DataConnectionString"));
            CloudQueueClient queueClient = storageAccount.CreateCloudQueueClient();
            CloudQueue queue = queueClient.GetQueueReference(queueName);
            queue.CreateIfNotExists();

            return queue;
        }
    }
}
```

Listing 2 – Implementacija *Run* metode *WorkerRole* klase.

```
public override void Run()
{
    CloudQueue queue = QueueHelper.GetQueueReference("vezba");
    // This is a sample worker implementation. Replace with your logic.
    Trace.TraceInformation("ImageConverter_WorkerRole entry point called",
"Information");

    while (true)
    {
        CloudQueueMessage message = queue.GetMessage();
        if (message == null)
        {
            Trace.TraceInformation("Trenutno ne postoji poruka u redu.",
"Information");
        }
        else
        {
            Trace.TraceInformation(String.Format("Poruka glasi: {0}",
message.AsString), "Information");
            ResizeImage(message.AsString);
            queue.DeleteMessage(message);
            Trace.TraceInformation(String.Format("Poruka procesuirana: {0}",
message.AsString), "Information");
        }

        Thread.Sleep(5000);
        Trace.TraceInformation("Working", "Information");
    }
}
```