**Лабораторная работа № 1**

**Создание первого Windows Azure приложения**

***Цель работы***

Целью лабораторной работы является практическое освоение создания приложений Windows Azure.

***Аппаратура и программные инструменты, необходимые для лабораторной работы***

* Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.
* Доступ к сети Интернет.
* Наличие аккаунта Windows Azure.

***План работы:***

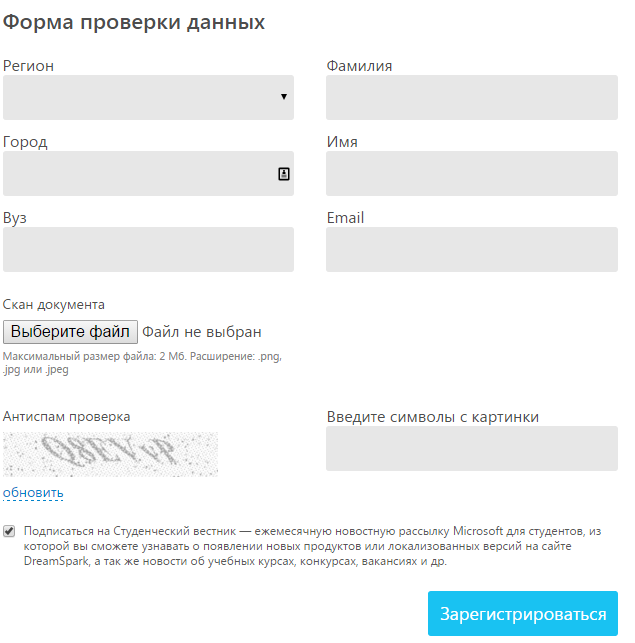
1. Получение бесплатного аккаунта Microsoft Windows Azure.
2. Создание проекта в Visual Studio с поддержкой Azure SDK
3. Создание модели данных для элементов в Table Storage.
4. Создание Веб роли для отображения гостевой книги.
5. Организация очереди рабочих элементов для обработки в фоне.
6. Проверка

**Содержание работы**

***1. Получение аккаунта Microsoft Windows Azure.***

* Первый вариант:

1. Перейдите на сайт <http://www.dreamspark.ru/requests>
2. Заполните форму проверки данных, загрузив скан-копию студенческого билета:



1. Дождитесь получения ответа (может занять несколько дней).
2. Далее следуйте инструкциям –

<http://www.dreamspark.ru/content/documents/AlreadyDreamspark.pdf>

* Второй вариант:

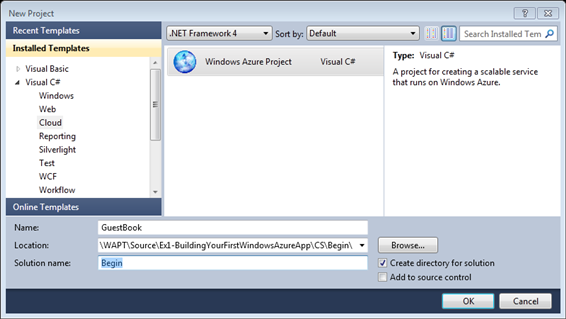
1. Следуйте инструкциям – <http://www.dreamspark.ru/content/documents/ForStudents.pdf>

* Третий вариант (менее предпочтительный):

1. Перейдите на сайт – <https://azure.microsoft.com/ru-ru/>
2. Перейдите по ссылке «Бесплатное пробное использование»
3. Заполните формы с контактной информацией и отправьте запрос на получение триального доступа на общих основаниях (**Важно**: для отправки запроса потребуется ввести данные банковской карты).

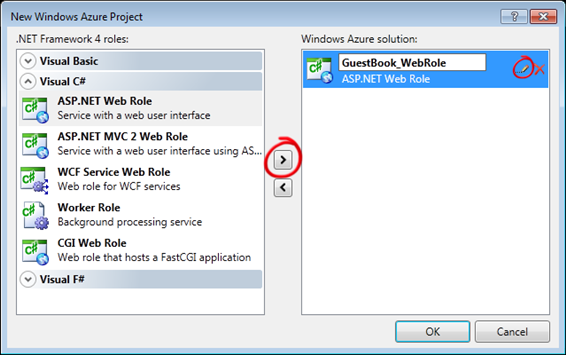
***2. Создание проекта в Visual Studio с поддержкой Azure SDK***

1. Инициируйте сценарий создания нового проекта в **Microsoft Visual Studio 2010**.
2. В диалоговом окне **New Project** развернитеузел **Visual C#** и в списке **Installed Templates** выберите **Cloud.**
3. В списке **Templates** выберите **Windows AzureCloud Service** (соответствующие программные инструменты должны быть предварительно загружены, иначе – загрузите их, воспользовавшись вашим аккаунтом Microsoft Windows Azure).
4. Введите Name «**GuestBook**», имя solution «**Begin**». Затем выберите расположение внутри папки **Ex1-BuildingYourFirstWindowsAzureApp**. Убедитесь что опция **Create directory for solution** выбрана и нажмите **OK** чтобы создать проект.



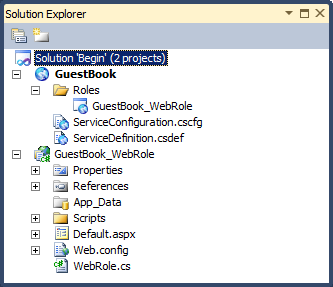
Создание нового проектаWindows Azure Cloud Service

1. В диалоге **New Cloud Service Project** разверните узел **Visual C#** и выберите **ASP.NET Web Role**. Переместите выбранную роль в проект. Выберите роль в проекте, нажмите на изображение «карандаш» и введите имя **GuestBook\_WebRole**. Нажмите **OK** для завершения.



Добавление ролей в проект.

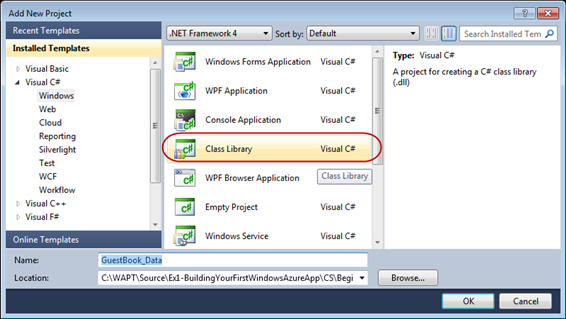
1. Обратите внимание на структуру проекта в **Solution Explorer**.



Структура проекта в Solution Explorer.

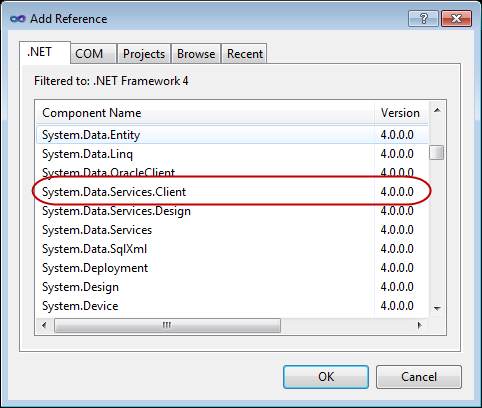
***3. Создание модели данных для элементов в Table Storage.***

1. В **Solution** Explorerнажмитеправойкнопкоймышипо **Begin,** выберите **Add | New Project**.
2. В диалоге **Add New Project,** разверните узел **Visual C#** в списке **Installed Templates**, выберите категорию **Windows** ивыделите **Class Library** в списке шаблонов. Убедитесь что выбран **.NET Framework 3.5** или выше. Введите имя **GuestBook\_Data** и нажмите **OK**.

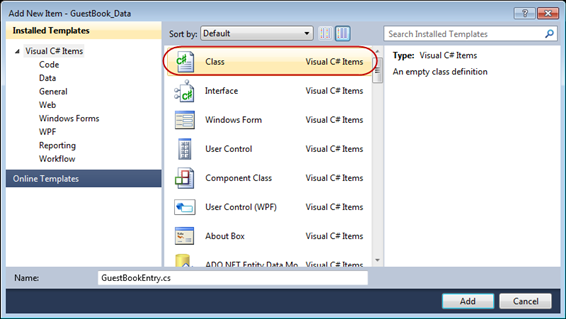


Создание библиотеки классов.

1. Удалите файл класса по умолчанию. Нажмите правой кнопкой по **Class1.cs** и выберите **Delete**. Нажмите **OK**.
2. Добавьте ссылку на библиотеку .NET для ADO.NET в проект **GuestBook\_Data**. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_Data**, выберите **Add Reference**, затем выберите закладку **.NET**, выделите компонент **System.Data.Service.Client** и нажмите **OK**.
3. Аналогично добавьте ссылку библиотеку **Microsoft.WindowsAzure.StorageClient**



1. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookEntry.cs** и нажмите **Add**.



1. Откройте файл **GuestBookEntry.cs**, добавьте в начало файла

using Microsoft.WindowsAzure.StorageClient;

1. Измените объявление класса**GuestBookEntry**

public class GuestBookEntry :

Microsoft.WindowsAzure.StorageClient.TableServiceEntity{

}

1. Добавьте конструктор по умолчанию

public GuestBookEntry(){

PartitionKey = DateTime.UtcNow.ToString("MMddyyyy");

// Row key allows sorting, so we make sure the rows come back in time order.

RowKey = string.Format("{0:10}\_{1}", DateTime.MaxValue.Ticks - DateTime.Now.Ticks, Guid.NewGuid());

}

1. Добавьте свойства

public string Message { get; set; }

public string GuestName { get; set; }

public string PhotoUrl { get; set; }

public string ThumbnailUrl { get; set; }

1. Сохраните файл **GuestBookEntry.cs**.
2. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookDataContext.cs** и нажмите **Add**.
3. Откройте файл **GuestBookDataContext.cs**, добавьте в начало файла

using Microsoft.WindowsAzure;

using Microsoft.WindowsAzure.StorageClient;

1. Измените объявление класса**GuestBookDataContext** и добавьте конструктор

public class GuestBookDataContext : TableServiceContext{

public GuestBookDataContext(string baseAddress, StorageCredentials credentials)

: base(baseAddress, credentials)

{ }

}

1. Добавьте свойство

public class GuestBookDataContext : TableServiceContext{

...

public IQueryable<GuestBookEntry> GuestBookEntry{

get{

return this.CreateQuery<GuestBookEntry>("GuestBookEntry");

}

}

}

1. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookDataContext.cs** и нажмите **Add**.
2. Откройте файл **GuestBookEntryDataSource.cs**, добавьте в начало файла

using Microsoft.WindowsAzure;

using Microsoft.WindowsAzure.StorageClient;

1. Далее измените класс

public class GuestBookEntryDataSource{

**private static CloudStorageAccount storageAccount;**

**private GuestBookDataContext context;**

}

1. Добавьте конструктор

public class GuestBookEntryDataSource

{

private static CloudStorageAccount storageAccount;

private GuestBookDataContext context;

**static GuestBookEntryDataSource(){**

**storageAccount = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");**

**CloudTableClient.CreateTablesFromModel(**

**typeof(GuestBookDataContext),**

**storageAccount.TableEndpoint.AbsoluteUri,**

**storageAccount.Credentials);**

}

}

1. Добавьте конструктор для класса **GuestBookDataEntrySource**

public GuestBookEntryDataSource(){

this.context = new GuestBookDataContext(storageAccount.TableEndpoint.AbsoluteUri, storageAccount.Credentials);

this.context.RetryPolicy = RetryPolicies.Retry(3, TimeSpan.FromSeconds(1));

}

1. Добавьте методы

**public IEnumerable<GuestBookEntry> Select()**{

var results = from g in this.context.GuestBookEntry

where g.PartitionKey == DateTime.UtcNow.ToString("MMddyyyy")

select g;

return results;

}

**public void UpdateImageThumbnail(string partitionKey, string rowKey, string thumbUrl)**{

var results = from g in this.context.GuestBookEntry

where g.PartitionKey == partitionKey && g.RowKey == rowKey

select g;

var entry = results.FirstOrDefault<GuestBookEntry>();

entry.ThumbnailUrl = thumbUrl;

this.context.UpdateObject(entry);

this.context.SaveChanges();

}

1. Сохраните файл **GuestBookEntryDataSource.cs**

***4. Создание Веб роли для отображения гостевой книги.***

1. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add Reference**, затем выберите закладку **.NET**, выделите компонент **System.Data.Service.Client** и нажмите **OK**.
2. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add Reference,** затем выберите закладку **Project**, выделите **GuestBook\_Data** и нажмите **OK**.
3. Удалите **Default.aspx**.
4. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add**, выделите **Existing Item.**
5. В диалоге **Add Existing Item** выберите директорию **\Source\Ex1-BuildingYourFirstWindowsAzureApp\CS\Assets**, выберите **Add.**
6. В **Solution Explorer** нажмите правой кнопкой по **Default.aspx**, выберите **View Code,** объявите следующие пространства имен

using System.IO;

using System.Net;

using Microsoft.WindowsAzure;

using Microsoft.WindowsAzure.ServiceRuntime;

using Microsoft.WindowsAzure.StorageClient;

using GuestBook\_Data;

1. В классе укажите

private static bool storageInitialized = false;

private static object gate = new Object();

private static CloudBlobClient blobStorage;

1. Найдите событие **SignButton\_Click** и добавьте следующий код

protected void SignButton\_Click(object sender, EventArgs e){

**if (FileUpload1.HasFile){**

**InitializeStorage();**

**// upload the image to blob storage**

**CloudBlobContainer container = blobStorage.GetContainerReference("guestbookpics");**

**string uniqueBlobName = string.Format("image\_{0}.jpg", Guid.NewGuid().ToString());**

**CloudBlockBlob blob = container.GetBlockBlobReference(uniqueBlobName);**

**blob.Properties.ContentType = FileUpload1.PostedFile.ContentType;**

**blob.UploadFromStream(FileUpload1.FileContent);**

**System.Diagnostics.Trace.TraceInformation("Uploaded image '{0}' to blob storage as '{1}'", FileUpload1.FileName, uniqueBlobName);**

**// create a new entry in table storage**

**GuestBookEntry entry = new GuestBookEntry() { GuestName = NameTextBox.Text, Message = MessageTextBox.Text, PhotoUrl = blob.Uri.ToString(), ThumbnailUrl = blob.Uri.ToString() };**

**GuestBookEntryDataSource ds = new GuestBookEntryDataSource();**

**ds.AddGuestBookEntry(entry);**

**System.Diagnostics.Trace.TraceInformation("Added entry {0}-{1} in table storage for guest '{2}'", entry.PartitionKey, entry.RowKey, entry.GuestName);**

**}**

**NameTextBox.Text = "";**

**MessageTextBox.Text = "";**

**DataList1.DataBind();**

}

1. Обновите метод**Timer1\_Tick**

protected void Timer1\_Tick(object sender, EventArgs e){

**DataList1.DataBind();**

}

1. Обновите событие **Page\_Load**

protected void Page\_Load(object sender, EventArgs e){

**if (!Page.IsPostBack){**

**Timer1.Enabled = true;**

**}**

}

1. Произведите изменения в методе **InitializeStorage**

private void InitializeStorage(){

**if (storageInitialized){**

**return;**

**}**

**lock (gate){**

**if (storageInitialized){**

**return;**

**}**

**try{**

**// read account configuration settings**

**var storageAccount = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");**

**// create blob container for images**

**blobStorage = storageAccount.CreateCloudBlobClient();**

**CloudBlobContainer container = blobStorage.GetContainerReference("guestbookpics");**

**container.CreateIfNotExist();**

**// configure container for public access**

**var permissions = container.GetPermissions();**

**permissions.PublicAccess = BlobContainerPublicAccessType.Container;**

**container.SetPermissions(permissions);**

**}**

**catch (WebException){**

**throw new WebException("Storage services initialization failure. "**

**+ "Check your storage account configuration settings. If running locally, "**

**+ "ensure that the Development Storage service is running.");**

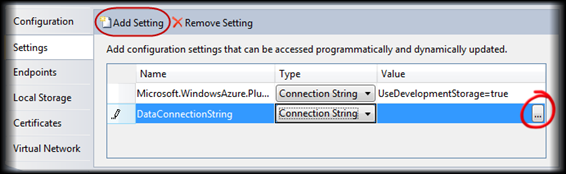
**}**

**storageInitialized = true;**

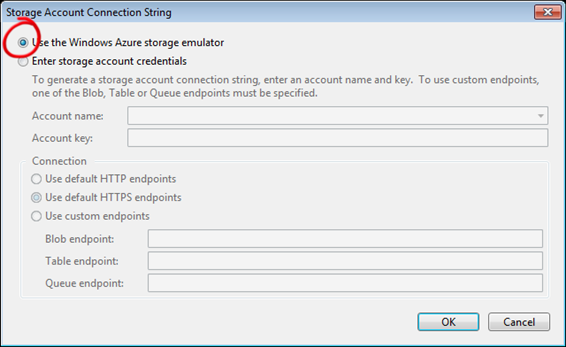
**}**

}

1. В **Solution Explorer** разверните узел **Roles** в проекте **GuestBook**. Нажмите два раза по **GuestBook\_WebRole**, откроется свойства данной роли, выберите закладку **Setting**. Нажмите **Add Setting**, наберите “DataConnectionString” в колонке **Name**, измените **Type** не *ConnectionString* и нажмите **Add** **Setting**.



1. В диалоге **Storage Connection String** выберите **Use development storage** и нажмите **OK.**



1. Сохраните изменения.
2. В проекте GuestBook\_WebRole, откройте файл Global.asax.cs
3. Объявите пространства имен

using Microsoft.WindowsAzure;

using Microsoft.WindowsAzure.ServiceRuntime;

1. Вставьте следующий код внутрь метода Application\_Start, заменив содержимое по умолчанию

void Application\_Start(object sender, EventArgs e){

**Microsoft.WindowsAzure.CloudStorageAccount.SetConfigurationSettingPublisher((configName, configSetter) =>**

**{**

**configSetter(RoleEnvironment.GetConfigurationSettingValue(configName));**

**});**

}

***5. Организация очереди рабочих элементов для обработки в фоне.***

1. В **Solution Explorer** нажмите правой кнопкой по **Default.aspx**, выберите **View Code,** объявите элемент клиента очереди

public partial class \_Default : System.Web.UI.Page{

private static bool storageInitialized = false;

private static object gate = new Object();

private static CloudBlobClient blobStorage;

**private static CloudQueueClient queueStorage;**

...

}

1. Найдите метод **InitializeStorage** и вставьте следующий код внутрь данного метода:

public partial class Default : System.Web.UI.Page{

...

private void InitializeStorage(){

...

try{

...

// configure container for public access

var permissions = container.GetPermissions();

permissions.PublicAccess = BlobContainerPublicAccessType.Container;

container.SetPermissions(permissions);

**// create queue to communicate with worker role**

**queueStorage = storageAccount.CreateCloudQueueClient();**

**CloudQueue queue = queueStorage.GetQueueReference("guestthumbs");**

**queue.CreateIfNotExist();**

}

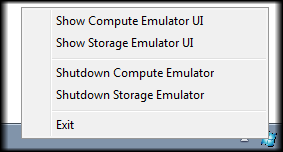
catch (WebException){

...

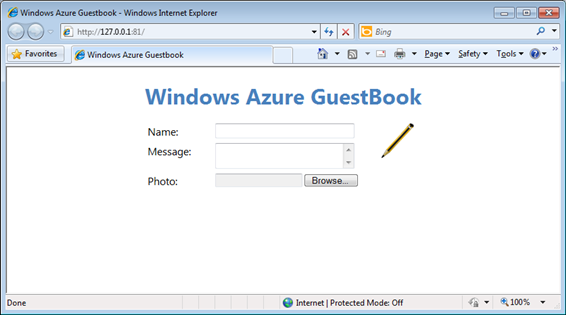
}

***6. Проверка***

1. Нажмите **F5** для запуска сервиса. Сервис запустится в development fabric. Для открытия пользовательского интерфейса необходимо нажать правой кнопкой мыши на значке в области уведомления панели задач и выбрать **Show Development Fabric UI**



1. Переключитесь на Internet Explorer для просмотра приложения GuestBook
2. Добавьте новую запись в гостевой книге



***Литература***

* 1. Windows Azure Platform Training Kit - January 2011 Update