Azure

236504 – winter 2016/17

Boris van Sosin, Marina Minkin, Nitsan Pri Hadash, Ariel Yehezkeli

SI CALOREM NON TOLERAS E CVLINA EXI
We’re going to talk about:

• Microsoft Azure Cloud Computing.
• WeMos.
Microsoft Azure
Cloud Computing
What Is Cloud Computing?
Advantages of Cloud Computing

• “Unlimited resources”
• Scale on demand
  • #clients
  • #services
• Out of the box infrastructure
• Redundant (geographical)
• Less deployment management
Azure Features

• Integrated into Visual Studio

• Many services in one place

• App Service
  • Has API for both tables and custom APIs
  • General computing- because Arduino computational resources are limited

• Connects to both Windows Phone and WeMos/NodeMCU (and Arduino with Ethernet shield)
The Azure portal

• Azure control panel: https://portal.azure.com/
• Each team was required to activate its Azure pass.
• You need Visual Studio 2015 or newer. You can get it from MSDNAA: https://csms.cs.technion.ac.il:4423/
Azure App Service

WEB APPS
Web apps that scale with your business

MOBILE APPS
Build Mobile apps for any device

LOGIC APPS
Automate business process across SaaS and on-premises

API APPS
Easily build and consume APIs in the cloud
Mobile Service Software Stack

Client

- App
- Proxy
- JSON / XML
- HTTP

Cloud

- Service
- Controller
- Controller
- JSON / XML
- HTTP
Table Controller Tutorial

<table>
<thead>
<tr>
<th>NAME</th>
<th>PUBLISHER</th>
<th>CATEGORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile App</td>
<td>Microsoft</td>
<td>Web + Mobile</td>
</tr>
<tr>
<td>Mobile Apps Quickstart</td>
<td>Microsoft</td>
<td></td>
</tr>
<tr>
<td>Mobile Engagement</td>
<td>Microsoft</td>
<td>Web + Mobile</td>
</tr>
<tr>
<td>aiMobile Mobile Acceleration -BYOL</td>
<td>aiScaler</td>
<td>Compute</td>
</tr>
<tr>
<td>Convertigo Mobility Platform</td>
<td>Convertigo</td>
<td>Compute</td>
</tr>
<tr>
<td>aiMobile Mobile Acceleration -HOURLY</td>
<td>aiScaler</td>
<td>Compute</td>
</tr>
<tr>
<td>Good Enterprise Mobility Server</td>
<td>BlackBerry</td>
<td>Compute</td>
</tr>
<tr>
<td>ManageEngine Mobile Device Manager Plus</td>
<td>ManageEngine</td>
<td>Compute</td>
</tr>
<tr>
<td>Keyhub</td>
<td>iQuest</td>
<td>Compute</td>
</tr>
</tbody>
</table>
Web App + SQL

Microsoft

Enjoy secure and flexible development, deployment, and scaling options for your web app plus a SQL database.

Create
1. Connect a database

- You already have a data connection
- SQLite enabled, not recommended for production use. Click here to create a SQL Azure data connection.

2. Create a table API

- To store data in your backend, you need a table. Pick a backend language below and create a TodoItem table API.

   Backend language: C#

   Download

   Once you've downloaded your personalized server project, extract it and open in Visual Studio. Right-click the project and select "Publish" to host the code in your mobile backend. The TodoItem table will be created automatically using Entity Framework.

3. Configure your client application

- CREATE A NEW APP
- CONNECT AN EXISTING APP
// Summary
// Provides application-specific behavior to supplement the default Application class.
// Summary
// This MobileServiceClient has been configured to communicate with the Azure Mobile Service and
// Azure Gateway using the application key. You're all set to start working with your Mobile Service!

public static MobileServiceClient MobileService = new MobileServiceClient("https://arduinomatcha.azurewebsites.net");
Try Mobile Apps

1. **Insert a TodoItem**
Enter some text below and click Save to insert a new todo item into your database

2. **Query and Update Data**
Click refresh below to load the unfinished TodoItems from your hosted service
Try Mobile Apps

1. **Insert a TodoItem**
Enter some text below and click Save to insert a new todo item into your database

2. **Query and Update Data**
Click refresh below to load the unfinished TodoItems from your hosted service
API Apps Tutorial

### API Results

<table>
<thead>
<tr>
<th>NAME</th>
<th>PUBLISHER</th>
<th>CATEGORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>API App</td>
<td>Microsoft</td>
<td>Web + Mobile</td>
</tr>
<tr>
<td>API management (preview)</td>
<td>Microsoft</td>
<td>Web + Mobile</td>
</tr>
<tr>
<td>Cognitive Services APIs (preview)</td>
<td>Microsoft</td>
<td>Intelligence + analytics</td>
</tr>
<tr>
<td>API Connection (preview)</td>
<td>Microsoft</td>
<td></td>
</tr>
<tr>
<td>Bing Maps API for Enterprise</td>
<td>Bing Maps</td>
<td>Web + Mobile</td>
</tr>
<tr>
<td>API App Test Bench (preview)</td>
<td>Microsoft</td>
<td></td>
</tr>
<tr>
<td>Appcelerator Arrow API Builder</td>
<td>Appcelerator</td>
<td>Compute</td>
</tr>
<tr>
<td>CipherPoint Eclipse Data Security API</td>
<td>CipherPoint</td>
<td>Compute</td>
</tr>
<tr>
<td>Cloud service</td>
<td>Microsoft</td>
<td>Compute</td>
</tr>
<tr>
<td>Customer Insights (preview)</td>
<td>Microsoft</td>
<td>Intelligence + analytics</td>
</tr>
<tr>
<td>SlashDB Unlimited</td>
<td>vt.enterprise</td>
<td>Compute</td>
</tr>
<tr>
<td>DreamFactory</td>
<td>Bitnami</td>
<td>Compute</td>
</tr>
<tr>
<td>SlashDB Cloud Edition</td>
<td>vt.enterprise</td>
<td>Compute</td>
</tr>
<tr>
<td>STRATO Blockchain Individual Instance</td>
<td>BlockApps</td>
<td>Compute</td>
</tr>
</tbody>
</table>

**Turnkey solution for publishing APIs to external and internal and modern API gateways for existing backend services**

- Provide API documentation and an interactive console.
- Throttle rate limit and quota your APIs.
- Monitor health of your APIs and quickly identify errors.
- Bring modern formats like JSON and REST to existing APIs.
- Connect to APIs hosted anywhere on the Internet.
- Gain analytic insights on how your APIs are being consumed.
- Manage your service via the Azure portal; REST API.

**Useful Links**

- Service Overview
- Getting Started
- Videos
- Documentation
Create your first API

Open the publisher portal and click ‘Import API’ from the dashboard. Use the link to the Calculator API Swagger definition below to create an example API. If you would prefer, you can manually define your API operation by operation instead.

Publisher portal
Calculator API Swagger definition

Make an API call from the Developer portal

The developer portal is the public-facing experience containing API documentation and the ability for developers to subscribe and get credentials for making calls. The built-in console allows you to make your first call without writing a single line of code.

Developer portal
APIs in developer portal

Review analytics

Once you have an API up and running you can check its usage and health via the Analytics tab on the publisher portal.

Publisher portal

Learn more

There are many more features that Azure API Management service provides you here are some links to get you started:

Get started with API Management
Customizing the developer portal
Full documentation
Import API

From clipboard

From file

From URL

Specification document URL

http://calcpi.cloudapp.net/calcapi.json

Specification format

- WADL
- Swagger
- WSDL

New API  Existing API

Web API URL suffix

Last part of the API’s public URL. This URL will be used by API consumers for sending requests to the web service.

Web API URL scheme

- HTTP
- HTTPS

This is what the URL is going to look like:

https://arduinomatchamangement.azure-api.net

Products (optional)

Add this API to one or more existing products.

Save  Cancel
Users

Current: Pending verification

ADD USER  INVITE USER

SELECT ALL  ADD TO GROUP  REMOVE FROM GROUP

Administrator
arduinomaster2@gmail.com via Azure

Administrators  Developers

Nitsan Pri Hasah
pnitsan@gmail.com via Basic auth

Developers

Subscribed to any product  Search users

EXPORT  DELETE  BLOCK

active

active
Administrator

This developer account is in **active** state and can be used to access all of the APIs it has subscriptions for.

- **Identity provider**: Azure
- **User name**: Administrator
- **Email**: arduinomaster2@gmail.com
- **Registered since**: 11/17/2015

### Subscriptions

**ADD SUBSCRIPTION**

<table>
<thead>
<tr>
<th>Subscription</th>
<th>Product</th>
<th>State</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscription name: Starter (default)</td>
<td>Starter</td>
<td>Active</td>
<td>[CANCEL]</td>
</tr>
<tr>
<td>Primary key: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>Show</td>
<td>Regenerate</td>
<td></td>
</tr>
<tr>
<td>Secondary key: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>Show</td>
<td>Regenerate</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subscription name: Unlimited (default)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary key: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>Show</td>
<td>Regenerate</td>
<td></td>
</tr>
<tr>
<td>Secondary key: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>Show</td>
<td>Regenerate</td>
<td></td>
</tr>
</tbody>
</table>

### Subscription requests

No results found.
APIs - Basic Calculator

- Summary
- Settings
- Operations
- Security
- Issues
- Products

ADD API TO PRODUCTS

No results found.
APIs - Basic Calculator

No results found.

Add API to product

- Starter
- Unlimited

Save  Cancel
Create your first API

Open the publisher portal and click 'Import API' from the dashboard. Use the link to the Calculator API Swagger definition below to create an example API. If you would prefer, you can manually define your API operation by operation instead.

Publisher portal
Calculator API Swagger definition

Make an API call from the Developer portal

The developer portal is the public-facing experience containing API documentation and the ability for developers to subscribe and get credentials for making calls. The built-in console allows you to make your first call without writing a single line of code.

Developer portal
APIs in developer portal

Review analytics

Once you have an API up and running you can check its usage and health via the Analytics tab on the publisher portal.

Publisher portal

Learn more

There are many more features that Azure API Management service provides you, here are some links to get you started:

Get started with API Management
Customizing the developer portal
Full documentation
ArduinMatchaOrganization API

APIs

Basic Calculator
Arithmetics is just a call away!

Echo API
Basic Calculator

Arithmetics is just a call away!

Add two integers

Produces a sum of two numbers.

Try it

Request URL

https://arduinomatchamanagement.azure-api.net/add?a=(a)&b=(b)

Request parameters

a
First operand. Default value is 51.

b
Second operand. Default value is 49.

Request headers

Ocp-Apim-Subscription-Key string
Subscription key which provides access to this API. Found in your Profile.

Request body

Code samples

C# Java JavaScript ObjC PHP Python Ruby
Basic Calculator

Add two integers

Produces a sum of two numbers.

Query parameters

a  
51

b  
49

[ Add parameter ]

Headers

Ocp-Apim-Trace  true

Ocp-Apim-Subscription-Key  [obfuscated]

[ Add header ]

Authorization

Subscription key  Primary-ec7f...

[ Remove header ]

Request URL

https://arduinomatchmanagement.azure-api.net/add?a=51&b=49
Request URL
https://arduinomatchamangement.azure-api.net//add?a=51&b=49

HTTP request
GET https://arduinomatchamangement.azure-api.net//add?a=51&b=49 HTTP/1.1
Host: arduinomatchamangement.azure-api.net
Ocp-Apim-Trace: true
Ocp-Apim-Subscription-Key: ***************

Send

Response
200 OK
Response latency
47 ms
Response content
Pragma: no-cache
Ocp-Apim-Trace-Location: https://apiimgmtsttktiasyrwaak2as.blob.core.windows.net/apiinspectorcontainer/Eolr6Olulps-mX5FfVkg2-457s2015-07-08sr-basiag-a0JPlmYT%2Bmg1085jih57K0%2B7wQaP5OLVY%2F5L5q1k8Xs%3D&se-2016-11-18T15%3A32%3A35Z&sp-r&traceId=22059047ced14218929261e2978c9993
Cache-Control: no-cache
Date: Thu, 17 Nov 2016 15:32:53 GMT
X-AspNet-Version: 4.0.30319
Once you copied your code from the developer portal, you may experience an error with the HttpUtility. To handle it, you have to right click your project, then select ‘add’ and ‘reference’, then choose System.Web and Save.
Publishing

- Don’t forget to publish your service!!!
Other Features

File Storage

- [https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-how-to-use-files](https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-how-to-use-files)

Blob Storage

- [https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-how-to-use-blobs](https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-how-to-use-blobs)
WeMos
System structure

• There are several ways to integrate WeMos into your project. Some of them are illustrated here

Today we will focus on this case
Wemos Client with Client Server Example
JSON

- JavaScript Object Notation
- A lightweight data-interchange format
- “Self-describing”, easy to parse in code, human-readable

```json
{"employees": [
    {
        "firstName": "John", "lastName": "Doe"
    },
    {
        "firstName": "Anna", "lastName": "Smith"
    },
    {
        "firstName": "Peter", "lastName": "Jones"
    }
]}
```
Creating Function App - Screenshots
Addind ESP8266 To the Arduino IDE

Open the Preferences menu in the Arduino IDE

Paste this URL:
Addind ESP8266 To the Arduino IDE

1. Click on Boards Manager...
2. Select esp8266
3. Install

3- reboot the machine
Running the Sample sketch

- In Tools -> Boards choose WeMos D1 R2 & mini
- File -> ESP8266HTTPClient -> BasicHttpClient
- Make sure to fix the AP details, and the server address (lines 34, 47 respectively)- don’t forget to replace https with https
- To send objects to the server, modifications to the http header are needed. After http.begin() add the following (copy-paste might ruin the apostrophes):
  - http.addHeader("Content-Type", "application/json");
  - http.addHeader("x-functions-key", "JpnxoZbrnhYD2MBvavafniqHAYB61uwZ45URiPljNljjunLrrPlWA=="); // the key here is the master key (marked in the screenshot)
  - Replace http.GET() with http.POST({JSON*string*comes*here})