Chapter 1: Enterprise Architecture Concepts

Enterprise Architecture

Business Architecture  Application Architecture  Data Architecture  Technology Architecture
EA Framework

Purpose Driven
- General Purpose
- Domain Specific

Industry Model
- TOGAF
- Zachman
- Federal EA

Processes, activities of the business execution
Collected, organized, distributed information
Software app, tools, etc.
Infrastructure like computer, network, etc.
• Customer members demand architecture standards ...
  • Customer members select TAFIM as preferred starting point...
  • DoD Information Systems Agency (DISA) donate TAFIM as base

• TOGAF first published

• TOGAF 7 – Technical Edition

• TOGAF 9 Enterprise Edition

• The Interoperable Enterprise Business Scenario first published

• TOGAF 8 – Enterprise Edition
  First TOGAF Certification

• TOGAF 9.1
<table>
<thead>
<tr>
<th>Scope</th>
<th>Data</th>
<th>Function</th>
<th>Network</th>
<th>People</th>
<th>Time</th>
<th>Motivation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Investor</td>
<td>List of Values</td>
<td>List of Precedence</td>
<td>List of Locations</td>
<td>List of Organizations</td>
<td>List of Events</td>
<td>List of Business Goals</td>
</tr>
<tr>
<td></td>
<td>Important for Business</td>
<td>Important to Business</td>
<td>Important to Business</td>
<td>Important to Business</td>
<td>Important to Business</td>
<td>Important to Business</td>
</tr>
<tr>
<td>Enterprise Model</td>
<td>e.g., Entity-Relationship Diagram</td>
<td>e.g., Data Flow Diagram</td>
<td>e.g., Application Architecture</td>
<td>e.g., Organizational Chart</td>
<td>e.g., Business Plan</td>
<td>e.g., Business Plan</td>
</tr>
<tr>
<td>Information System Model</td>
<td>e.g., Data Model</td>
<td>e.g., Data Flow Diagram</td>
<td>e.g., Distributed System Architecture</td>
<td>e.g., Resource Interface Structure</td>
<td>e.g., Business Process</td>
<td>e.g., Business Process</td>
</tr>
<tr>
<td>Technology Model</td>
<td>e.g., Data Design</td>
<td>e.g., Structure Chart</td>
<td>e.g., System Architecture</td>
<td>e.g., System Design/Technology Architecture</td>
<td>e.g., Business Process</td>
<td>e.g., Business Process</td>
</tr>
<tr>
<td>Components</td>
<td>e.g., Data Flow</td>
<td>e.g., Program</td>
<td>e.g., Network Security Architecture</td>
<td>e.g., Technology Design/Technology Architecture</td>
<td>e.g., Business Process</td>
<td>e.g., Business Process</td>
</tr>
<tr>
<td>Subcontractor</td>
<td>e.g., Data Flow</td>
<td>e.g., Program</td>
<td>e.g., Network Security Architecture</td>
<td>e.g., Technology Design/Technology Architecture</td>
<td>e.g., Business Process</td>
<td>e.g., Business Process</td>
</tr>
<tr>
<td>Functioning System</td>
<td>e.g., Data</td>
<td>e.g., Function</td>
<td>e.g., Network</td>
<td>e.g., Organization</td>
<td>e.g., Schedule</td>
<td>e.g., Strategy</td>
</tr>
</tbody>
</table>
Chapter 2: Principles and Patterns

```
Component
  + operation()

ConcreteComponent
  + operation()

Decorator
  - component
  + operation()

ConcreteDecorator
  + operation()
```
### Creational patterns

These design patterns deal with object creational mechanisms in a suitable manner under a given situation

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Abstract factory</td>
<td>Provides an interface that can create families of related classes</td>
</tr>
<tr>
<td>2</td>
<td>Builder</td>
<td>Separates complex object construction from its representation</td>
</tr>
<tr>
<td>3</td>
<td>Factory method</td>
<td>An interface used to create an object without specifying the exact class to be created</td>
</tr>
<tr>
<td>4</td>
<td>Prototype</td>
<td>Creates fully initialized objects by copying or cloning an existing object</td>
</tr>
<tr>
<td>5</td>
<td>Singleton</td>
<td>A class whose only a single instance can exist</td>
</tr>
</tbody>
</table>

### Structural Patterns

These design patterns are the best practice used to identify a simple way to realize relationships between entities in a given situation

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Adapter</td>
<td>Converts the interface of a class into another interface client except by wrapping its own interface around that of an already existing class</td>
</tr>
<tr>
<td>7</td>
<td>Bridge</td>
<td>Separates an object’s interface from its implementation so that the two can vary independently</td>
</tr>
<tr>
<td>8</td>
<td>Composite</td>
<td>Composes zero or more similar objects in a tree structure so that they can be manipulated as one object to represent part or whole hierarchies</td>
</tr>
<tr>
<td>9</td>
<td>Decorator</td>
<td>Dynamically adds/overrides behavior in an existing object</td>
</tr>
<tr>
<td>10</td>
<td>Facade</td>
<td>Provides a unified interface to a set of interfaces in a subsystem</td>
</tr>
<tr>
<td>11</td>
<td>Flyweight</td>
<td>Efficient sharing of a large number of similar objects</td>
</tr>
<tr>
<td>12</td>
<td>Proxy</td>
<td>An object acting as a placeholder for another object to control access to it and reduce complexity</td>
</tr>
</tbody>
</table>
**Behavioral patterns**

They identify common communication patterns between objects and increase the flexibility in carrying out the communication between them

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Chain of responsibility</td>
<td>Decouples the request sender from the request receiver by chaining more than one receiver object in a way that if one receiver object does not handle the request, it passes on to the next until the request has been handled</td>
</tr>
<tr>
<td>14</td>
<td>Command</td>
<td>Encapsulates a request/action/event along with parameter as an object</td>
</tr>
<tr>
<td>15</td>
<td>Interpreter</td>
<td>Specifies a way to interpret/evaluate sentences in a given language</td>
</tr>
<tr>
<td>16</td>
<td>Iterator</td>
<td>Sequentially accesses the elements present in an aggregate object without exposing its underlying representation/structure</td>
</tr>
<tr>
<td>17</td>
<td>Mediator</td>
<td>Promotes loose coupling by encapsulating (and often centralizing) the communication/interaction between various objects</td>
</tr>
<tr>
<td>18</td>
<td>Memento</td>
<td>Provides the capability to snapshot the object’s internal state in order to restore to this state later</td>
</tr>
<tr>
<td>19</td>
<td>Observer</td>
<td>An observable object sends events to many observing objects; sort of defines a one-to-many dependency between objects</td>
</tr>
<tr>
<td>20</td>
<td>State</td>
<td>Allows the changes in the object’s behavior based on the change in its state</td>
</tr>
</tbody>
</table>

**Abstract Factory Pattern**

```
Abstract Factory
      /   \
  Create Product A()  Create Product B()
      |     |
  △←                             ←△
    create                      create

Concrete Factory 1
CREATE Product A()  CREATE Product B()

Concrete Factory 2
CREATE Product A()  CREATE Product B()

Concrete Product A1
CREATE 1 Product A

Concrete Product A2
CREATE 1 Product B

Concrete Product B1

Concrete Product B2
```
Builder Pattern

Director
- builder : Builder
- Construct Product()

Contains reference to builder

Builder
- Build Part A()
- Build Part B()
- Get Results()

Note
builder. Build Part A();
builder. Build Part B();
Obj = builder. GetResult();

Concrete Builder
- Build Part A()
- Build Part B()
- Get Results()

create

Concrete Product

Adapter Pattern

Client

Adapter

Adapter

Client

DB Adapter

Oracle DB Driver
Chapter 3: Distributed Computing
### Distributed Computing

![Diagram of distributed computing](image)

### Forms of Transparency

<table>
<thead>
<tr>
<th>#</th>
<th>Type</th>
<th>Objective to hide</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Access</td>
<td>difference in data representation and how resource is accessed</td>
</tr>
<tr>
<td>2</td>
<td>Migration</td>
<td>move a resource to another location</td>
</tr>
<tr>
<td>3</td>
<td>Location</td>
<td>where a resource is located</td>
</tr>
<tr>
<td>4</td>
<td>Replication</td>
<td>share a resource by several competitive users</td>
</tr>
<tr>
<td>5</td>
<td>Relocation</td>
<td>move a resource to another location, while in use</td>
</tr>
<tr>
<td>6</td>
<td>Concurrency</td>
<td>share resource by several competitive users</td>
</tr>
<tr>
<td>7</td>
<td>Persistence</td>
<td>whether a resource is in memory or on disk</td>
</tr>
<tr>
<td>8</td>
<td>Failure</td>
<td>resource’s failure and recovery process</td>
</tr>
</tbody>
</table>

### Failure Rate Over Time

![Graph showing failure rate over time](image)
Chapter 4: Software Development Life Cycle
Microsoft Azure
20% Linux
1,000 Linux VMs to choose from

CodePlex membership has more than tripled

1,100,000+

30,000 lines of code & 10,000+ engineering hours in support of Hadoop projects

SUSE - Microsoft Alliance customers

>1,000

WebMatrix

>1M downloads

WPF Windows Forms ASP.NET (4 & 5) ASP.NET 5 Universal Windows Apps

.NET Framework 4.6

.NET Core 5

CoreCLR

.NET Native runtime

Fully-featured and integrated .NET runtime and libraries for Windows

Modular and optimized .NET runtimes and libraries

Shared

Runtime Components

RyuJIT, GC, SIMD

Compilers

.NET Compiler Platform (Roslyn)

Languages innovation

NuGet packages

.NET Core 5 Libraries

.NET Framework 4.6 Libraries

1996 - ASP

2005 - ASP.NET2

2012 - Web API

2016 - ASP.NET Core

2002 - ASP.NET

2008 - MVC

2015 - vNext
Chapter 5: Enterprise Practices in Software Development
Check out the project

Stage Fixes

Commit
Commit to move the file from staging to permanent storage of Git
Modify files in the working directory
Staging the modified file by adding snapshots in staging area
<table>
<thead>
<tr>
<th>File</th>
<th>Date</th>
<th>Size</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>.template.config</td>
<td>28-Feb-2017 12:31</td>
<td></td>
<td>File folder</td>
</tr>
<tr>
<td>src</td>
<td>27-Feb-2017 7:30</td>
<td></td>
<td>File folder</td>
</tr>
<tr>
<td>EAWebApplication1.sln</td>
<td>27-Feb-2017 7:30</td>
<td>2 KB</td>
<td>Microsoft Visual Studio</td>
</tr>
<tr>
<td>global.json</td>
<td>27-Feb-2017 7:30</td>
<td>1 KB</td>
<td>JSON File</td>
</tr>
</tbody>
</table>
Chapter 6: Layered Approach to Solution Architecture

<table>
<thead>
<tr>
<th>Advantages</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Web service (Web API, WCF, Web Services)</strong></td>
<td><strong>Shared library (POGOS)</strong></td>
</tr>
<tr>
<td>Technology agnostic – any consumer can access</td>
<td>No Internet connection is required</td>
</tr>
<tr>
<td>Code cannot be decompiled</td>
<td>No serialization overhead</td>
</tr>
<tr>
<td>Easy upgrades or bug fixes</td>
<td>No risk of forgery</td>
</tr>
<tr>
<td>Scalable</td>
<td>Higher performance as native code is called</td>
</tr>
<tr>
<td>Centralized deployments</td>
<td>There is no downtime of service impacting the consumer</td>
</tr>
</tbody>
</table>
Tenant Management System

TMS
Tenant Management

The main purpose of the system is to provide functionality to schedule and conduct jobs.
Chapter 7: SOA Implementation with .NET Core
**Business analysis and optimization service:**
Analyze real-time business data and plan business optimization

**Interaction services:**
Enable collaboration between people, processes, and information

**Process services:**
Orchestrate and automate business processes

**Information Services:**
Manage data and content, in a unified manner

**Enterprise Service Bus:**
Support interconnectivity between services

**Partner services:**
Connect with business partners

**Business application services:**
Provide robust, scalable, and secure services environment

**Access services**
Facilitate interaction with existing information and data assets

**Infrastructure hosting and support service:**
Ensure availability and performance

**SCA Domain**

Composite X

Service

Composite Y

Component A

Composite Z

Reference

Promote

implementation

Composite A
Chapter 8: Cloud-Based Architecture and Integration with .NET Core
<table>
<thead>
<tr>
<th>Features</th>
<th>Virtual machines</th>
<th>Cloud services</th>
<th>Azure App Services</th>
<th>Service Fabric</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure managed OS</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Quick deployment</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Shared storage, which enables easy scaling</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Deployment slots to keep multiple environments</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Operating system and patches updated by Azure</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Deploy code from git</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Deploy code from TFS</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Access to Azure storage services</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Support for different languages like ASP.NET, Node.js, PHP, and Python</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>SSL support</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Remote Access to servers</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Integrated Monitoring support</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Azure App Service

Web Apps

Mobile Apps

API Apps

Logic Apps
App Service
Host your web and mobile applications, REST APIs, and more in Azure

Subscription
Visual Studio Ultimate with MSDN

View
Resource Group

Search

Create App Service
Host your web and mobile applications, REST APIs, and more in Azure

Hosting

Services

Web App Name

Subscription
Visual Studio Ultimate with MSDN

Resource Group
Default-Storage-NorthEurope

App Service Plan

Microsoft account
ovaismehboob@hotmail.com
Configure App Service Plan

An App Service plan is the container for your app. The App Service plan settings will determine the location, features, cost and...

App Service Plan
HelloWorld20161231081245Plan

Location
South Central US

Size
S3 (4 cores, 7 GB RAM)

HelloCloudChapterWebApp - Web Deploy

Publish method: Web Deploy

Server: hellolochapterwebapp.scm.azurewebsites.net:443

Site name: HelloCloudChapterWebApp

User name: $HelloCloudChapterWebApp

Password: 

Save password

Destination URL: http://hellolochapterwebapp.azurewebsites.net

Validate Connection
Web apps on Linux leverage the power of Docker containers to let you use custom containers from Azure Container Registry, Docker Hub, a private container registry, or use one of our default containers provided by App Service.

**Image source**
- [Build] Docker Hub | Private registry

**Runtime Stack**
- .Net Core v1.0
  - Node.js
    - Node.js 8.10
    - Node.js 6.11
    - Node.js 4.3.0
    - Node.js 4.4.7
  - PHP
    - PHP 5.6.23
    - PHP 7.0.8
  - .Net Core
    - .Net Core v1.0

---

**APP DEPLOYMENT**

- Quickstart
- Deployment credentials
- Deployment slots
- Deployment options
- Continuous Delivery (Preview)
When a post is created (Preview)

No additional information is needed for this step. You will be able to use the outputs in subsequent steps.

How often do you want to check for items?

- Frequency
  - Minute

- Interval
  - 3

When a post is created (Preview)

No additional information is needed for this step. You will be able to use the outputs in subsequent steps.

How often do you want to check for items?

- Frequency
  - Minute
- Interval
  - 3


Post a tweet

Tweet text

We published a new blog post [Title](URL) and the link is [URL]

Show advanced options

Connected to ovaismehboob. Change connection.
Integrate with your favorite services by notifying them when events happen in your project.

+ Create subscription
Service
Select a service to integrate with. Discover more integrations

Web Hooks
Provides event communication via HTTP

**Supported events:**
- All events

**Supported actions:**
- Post via HTTP

Learn more about this service
NEW SERVICE HOOKS SUBSCRIPTION

Trigger
Select an event to trigger on and configure any filters.

Trigger on this type of event
Build completed

Remember that selected events are visible to users of the target service, even if they don't have permission to view the related artifact.

FILTERS
Build Definition
[Optional]

Build Status
[Optional]
Web Hooks (Post via HTTP)

**Summary**
- Request
- Response
- Event

✅ **Succeeded**

Sent at: Sunday, January 22, 2017 5:21:29 PM

**Message**

*Bug #5 (Some great new idea!)* created by Jamal Hartnett.

---

<table>
<thead>
<tr>
<th>NAME</th>
<th>PUBLISHER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function App</td>
<td>Microsoft</td>
</tr>
</tbody>
</table>
### Choose a template

<table>
<thead>
<tr>
<th>Language</th>
<th>Scenario</th>
<th>Function Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlobTrigger-CSharp</td>
<td></td>
<td>A C# function that will be run whenever a blob is added to a specified container</td>
</tr>
<tr>
<td>BlobTrigger-JavaScript</td>
<td></td>
<td>A JavaScript function that will be run whenever a blob is added to a specified container</td>
</tr>
<tr>
<td>EventHubTrigger-CSharp</td>
<td></td>
<td>A C# function that will be run whenever an event hub receives a new event</td>
</tr>
<tr>
<td>EventHubTrigger-JavaScript</td>
<td></td>
<td>A JavaScript function that will be run whenever an event hub receives a new event</td>
</tr>
<tr>
<td>HttpTrigger-JavaScript</td>
<td></td>
<td>A JavaScript function that will be run whenever it receives an HTTP request</td>
</tr>
<tr>
<td>ManualTrigger-CSharp</td>
<td></td>
<td>A C# function that is triggered manually via the portal &quot;Run&quot; button</td>
</tr>
<tr>
<td>ManualTrigger-JavaScript</td>
<td></td>
<td>A JavaScript function that is triggered manually via the portal &quot;Run&quot; button</td>
</tr>
</tbody>
</table>

### Code (run.csx)

```
using System;

public static void Run(string myQueueItem, TraceWriter log)
{
    log.Info("C# Queue trigger function processed: {myQueueItem}");
}
```
Application Logging (Blob)

Off  On

Level
Error

Storage Settings
Storage not configured

* Name
blobwebappchap9
.core.windows.net

Performance
Standard  Premium

Replication
Locally-redundant storage (LRS)
Locally-redundant storage (LRS)
Zone-redundant storage (ZRS)
Geo-redundant storage (GRS)
Read-access geo-redundant storage (RA-GRS)

<table>
<thead>
<tr>
<th>NAME</th>
<th>PUBLISHER</th>
<th>CATEGORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Insights</td>
<td>Microsoft</td>
<td>Web + Mobile</td>
</tr>
</tbody>
</table>
Chapter 9: Microservices Architecture

A monolithic application puts all its functionality into a single process...

... and scales by replicating the monolith on multiple servers

A microservices architecture puts each element of functionality into a separate service...

... and scales by distributing these services across servers, replicating as needed.
Typical Microservice Architecture
Typical Serverless Architecture

- CRUD on Customer
- GetPotentialCustomers
- GetAllCustomers

Document Management System
- Documentum
- SharePoint services
- GoogleDrive / Dropbox
- (based on) Azure Blob
Hello, world!
These are some of the top level contents..

Learn more..
Chapter 10: Security Practices with .NET Core
namespace Microsoft.AspNetCore.Identity.EntityFrameworkCore
{
    ...public abstract class IdentityDbContext<TUser, TRole, TKey, TUserClaim, TUserRole, TUserRole, TKey, TUserLogin, TUserRole, TKey, TUserToken, TKey> : DbContext
    {
        ...public IdentityDbContext(DbContextOptions<DbContext> options, DbSet<TUserClaim> roleClaims = null, DbSet<TUserRole> roles = null, DbSet<TUserLogin> userLogins = null, DbSet<TUserRole> userRoles = null, DbSet<TUserToken> userTokens = null)
        {
            ...protected override void OnModelCreating(ModelBuilder builder)
            {
                ...public partial class ApplicationDbContext
                {
                    ...public virtual DbSet<ApplicationUser> ApplicationUsers
                    {
                        get;
                        set;
                    }
                    ...public virtual DbSet<ApplicationRole> ApplicationRoles
                    {
                        get;
                        set;
                    }
                    ...public virtual DbSet<ApplicationClaim> ApplicationClaims
                    {
                        get;
                        set;
                    }
                }
            }
        }
    }
}

---

Migrations

C# 20160911065352_Initial.cs
C# ApplicationDbContextModelSnapshot.cs
Tell us about your website

Site URL

https://testsiteEA.com

Next
<table>
<thead>
<tr>
<th>App ID</th>
<th>App Secret</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Display Name</th>
<th>Namespace</th>
</tr>
</thead>
<tbody>
<tr>
<td>EA</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>App Domains</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Privacy Policy URL</th>
<th>Terms of Service URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privacy policy for Login dialog and App Details</td>
<td>Terms of Service for Login dialog and App Details</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contact Email</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:ovalemehboob@yahoo.com">ovalemehboob@yahoo.com</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apps for Pages</td>
</tr>
</tbody>
</table>

---

**Log in.**

Use a local account to log in.

<table>
<thead>
<tr>
<th>Email</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- [ ] Remember me?

Log in

Register as a new user?
Forget your password?

---

**Register.**

Associate your Facebook account.

Association Form

You're successfully authenticated with Facebook. Please enter an email address for this site below and click the Register button to finish logging in.

<table>
<thead>
<tr>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

Register
Manage your account.
Change your account settings

Password: [ Change ]
External Logins: [ Manage ]
Phone Number: Phone Numbers can be used as a second factor of verification in two-factor authentication. See this article for details on setting up this ASP.NET application to support two-factor authentication using SMS.

Two-Factor Authentication... There are no two-factor authentication providers configured. See this article for setting up this application to support two-factor authentication.
[ Enable ] [ Disable ]

Verify Phone Number.
Add a phone number.

Code

Submit

Manage your account.
Your phone number was added.
Change your account settings

Password: [ Change ]
External Logins: [ Manage ]
Phone Number: Phone Numbers can be used as a second factor of verification in two-factor authentication. See this article for details on setting up this ASP.NET application to support two-factor authentication using SMS.

Two-Factor Authentication... There are no two-factor authentication providers configured. See this article for setting up this application to support two-factor authentication.
[ Enable ] [ Disable ]
Resource Owner (End User) → Access

Retrieve Token

Client (Web Application, Native Application, etc.) → Authorization Server

Resource Server (Web API, Web Services, etc.)
Welcome to IdentityServer4

IdentityServer publishes a discovery document where you can find metadata and links to all the endpoints, key material, etc.

Here are links to the source code repository, and ready to use samples.
Login

Local Login

Username

Password

Remember My Login

Login

MVC Client is requesting your permission

Uncheck the permissions you do not wish to grant.

Personal Information

- Your user identifier (required)

User profile

- Your user profile information (first name, last name, etc.)

User roles

Remember My Decision

Yes, Allow  No, Do Not Allow
Web API Client is requesting your permission

Uncheck the permissions you do not wish to grant.

- Personal Information
- Your user identifier
- User profile
- Application Access
- Offline access
- Vendor API
  Vendor API scope

Remember My Decision

Yes, Allow  No, Do Not Allow
Chapter 11: Modern AI Offerings by Microsoft
ACS with Swarm
Simple Sum App
Simple Sum App

44 11 =

This site says...
55
**PUBLIC REPOSITORY**

habibcs/eea-spa ⭐

Last pushed: 21 days ago

---

**Short Description**

EEA Book - Sample Repo for Front End

**Docker Pull Command**

docker pull habibcs/eea-spa

---

**Full Description**

EEA Book - Sample Repo for Front End

https://www.packtpub.com/application-development/enterprise-application-architecture-net-core

**Owner**

habibcs

---

Comments (0)
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage account</td>
<td>Storage account</td>
<td>East US</td>
</tr>
<tr>
<td>Storage account</td>
<td>Storage account</td>
<td>East US</td>
</tr>
<tr>
<td>Storage account</td>
<td>Storage account</td>
<td>East US</td>
</tr>
<tr>
<td>Container service</td>
<td>Container service</td>
<td>East US</td>
</tr>
<tr>
<td>Storage account</td>
<td>Storage account</td>
<td>East US</td>
</tr>
<tr>
<td>Storage account</td>
<td>Storage account</td>
<td>East US</td>
</tr>
<tr>
<td>Virtual machine scale set</td>
<td>Virtual machine</td>
<td>East US</td>
</tr>
<tr>
<td>Load balancer</td>
<td>Load balancer</td>
<td>East US</td>
</tr>
<tr>
<td>Virtual machine</td>
<td>Virtual machine</td>
<td>East US</td>
</tr>
<tr>
<td>Network interface</td>
<td>Network interface</td>
<td>East US</td>
</tr>
<tr>
<td>Availability set</td>
<td>Availability set</td>
<td>East US</td>
</tr>
<tr>
<td>Public IP address</td>
<td>Public IP address</td>
<td>East US</td>
</tr>
<tr>
<td>Load balancer</td>
<td>Load balancer</td>
<td>East US</td>
</tr>
<tr>
<td>Virtual network</td>
<td>Virtual network</td>
<td>East US</td>
</tr>
</tbody>
</table>
azurehb@swarm-master-580D48D8-0:~$ docker info
Containers: 2
    Running: 2
    Paused: 0
    Stopped: 0
Images: 2
Server Version: 1.13.1
Storage Driver: aufs
    Root Dir: /var/lib/docker/aufs
    Backing Filesystem: extfs
    DirFs: 20
    DirsPerMnt Supported: true
Logging Driver: json-file
Cgroup Driver: cgroupfs
Plugins:
    Volume: local
    Network: bridge host macvlan null overlay
Swarm: inactive
Runtimes: runc
    Default Runtime: runc
Init Binary: docker-init
    containerd version: aa0107dbd3b7ae67d8e3c3a151115e3e9e4a7cd1
    runc version: 9df503060155d3a829be4119c815b7304dd8f
    init version: 949f6fa
Security Options:
    apparmor
Kernel Version: 3.19.0-65-generic
Operating System: Ubuntu 14.04.4 LTS
OSType: linux
Architecture: x86_64
CPUs: 2
Total Memory: 6.805 GiB
Name: swarm-master-580D48D8-0
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): false
Registry: https://index.docker.io/v1/
WARNING: No swap limit support
Experimental: false
Insecure Registries:
    127.0.0.0/0
Live Restore Enabled: false
Simple Sum App

33 + 11 = 44

This site says...

44

OK
Scale up
(fewer, larger servers)

vs

Scale out
(more smaller servers)

2002: Apache Nutch

2004: Google publishes MapReduce

2006-07: Yahoo launches Hadoop
Windows Azure Hadoop
Travel Analysis

Regional Load

Sales

Number of Trips

Average Cost Per Mile

Delayed Flights

Actual Flight Expenditures, Bu.

Budget Remaining

Flight Expense

Rush Booking Trip Class, Session

Budget Remaining

Variance to Budget

Average Cost Per Trip

($1.39M)

$723.16
Data Mining

Statistics

Machine Learning

Artificial Intelligence