



AZ-300T02

Module 2: Implementing and Managing Application Services

Ahmad Majeed Zahoory



1



Module 02: Evaluating and Performing Server Migration to Azure

Lesson 01: Deploying Web Apps

2

Web App Features

Service included in the App Service offering, which provides:

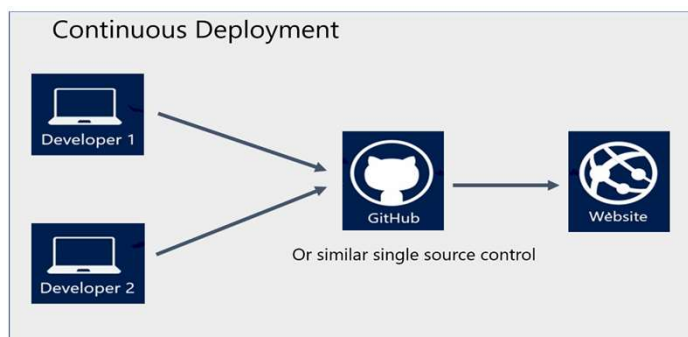
- **Multiple languages and frameworks**
- **DevOps optimization**
- **Global scale with high availability**
- **Connections to SaaS platforms and on-premises data**
- **Security and compliance**
- **Application templates**
- **Visual Studio integration**
- **API and mobile features**
- **Serverless code**

3

What is Continuous Development?

Automatic publishing of updates to source control code:

- **Visual Studio Team Services**
- **Bitbucket**
- **GitHub**
- **Git**
- **and many others...**



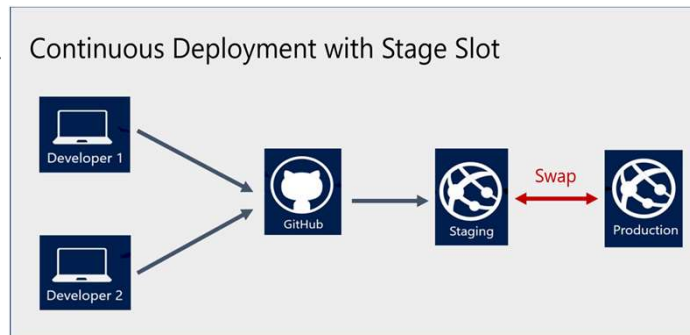
4

Staging Environments in App Service

A separate deployment slot of App Service services:

- **Supported by web apps, mobile apps, and API apps**
- **Requires a higher pricing tier:**
 - Standard
 - Premium
 - Isolated

- **Offers a number of benefits:**
 - Additional validation of app changes
 - Slot warming before swap
 - Support for auto-swap
 - Immediate rollback



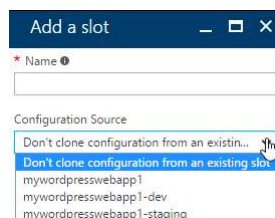
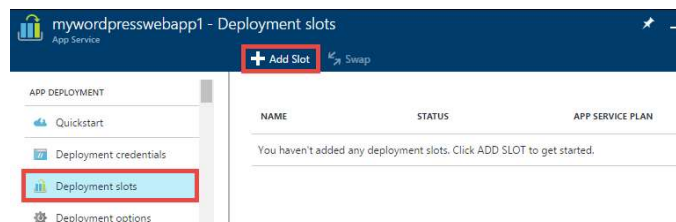
5

Add a Deployment Slot

The operation can be performed directly from the Azure portal

Slot cloning:

- **For the first slot:**
 - you can clone the production slot
- **For subsequent slots:**
 - you can clone any existing slots

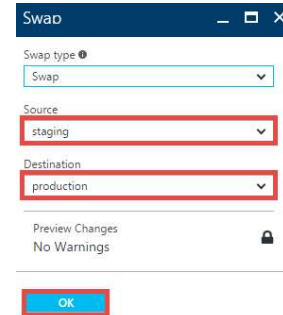


6

Swap Deployment Slots

There are three types of slot swapping settings:

- **Settings that are not swapped:**
 - publishing endpoints, custom domain names, SSL certificates and bindings, scale settings, WebJob schedulers.
- **Settings that are swapped:**
 - general settings, handler mappings, monitoring, diagnostics, and WebJobs.
- **Settings that are configurable (either swapped or not swapped):**
 - App settings and connection strings.

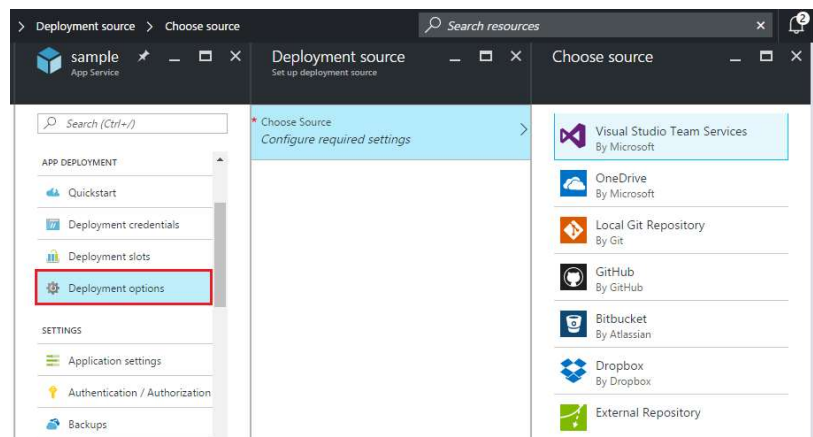


7

Enabling Continuous Deployment

Configurable from the Azure portal:

- VSTS
- OneDrive
- Local Git
- GitHub
- Bitbucket
- Dropbox
- External Repository

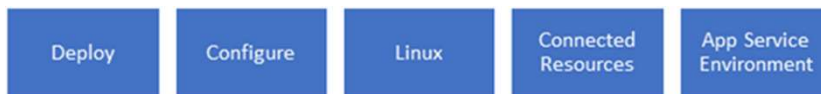


8

Web App Templates

Azure Resource Manager templates for:

- **Deploying a Windows-based web app:**
- **Configuring a Windows-based web app:**
- **Deploying and configuring a Linux-based web app**
- **Deploying web apps with connected resources (e.g. an Azure SQL Database)**
- **Creating App Service Environment v2**



9

Module 02: Implementing and Managing Application Services

Lesson 02: Managing Web Apps

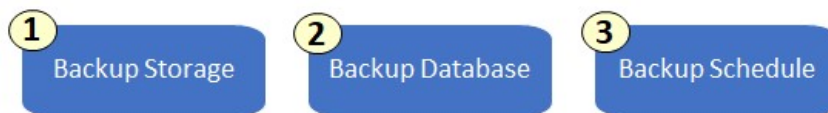


10

Backup Your App

To configure Azure web app backups, provide:

- **1. Backup storage which designates backup destination:**
 - Azure Storage account and container
- **2. Backup database (in scenarios where the web app uses a back-end database):**
 - The database must be referenced in the connection strings of the web app
 - The backup supports Azure SQL Database, Azure Database for MySQL, Azure Database for PostgreSQL, and MySQL in-app
- **3. Backup schedule (in order to automate backups)**



11

Restore a Backup

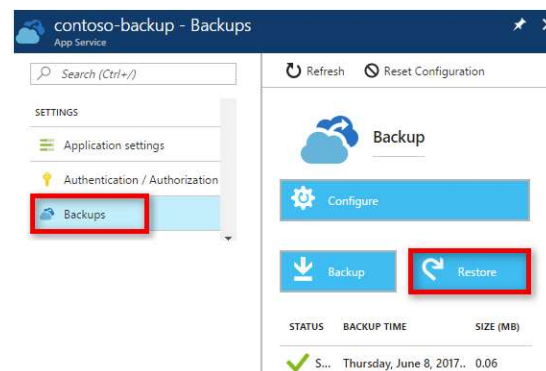
Available from the Azure portal

Included in higher pricing tiers:

- **Standard**
- **Premium**

Requires designating:

- **Restore source:**
 - Any backup of the current web app
 - Any web app backup in Azure Storage
- **Restore destination:**
 - An existing web app
 - A new web app



12

Restore a Snapshot

Automatic, incremental shadow copies of a web app:

- **Requires the Premium pricing tier**
- **Is a subject to a number of constraints:**
 - You can only restore to the same app or to a slot belonging to that app.
 - App Service stops the target app or target slot during the restore.
 - App Service keeps three months' worth of snapshots for platform data recovery purposes.
 - You can only restore snapshots from the last 30 days.
- **Offers several benefits:**
 - No file copy errors due to file locks.
 - No storage size limitation.
 - No configuration required



Select Backup to Restore

Select from either a Backup on the app or zip file of a valid backup from a storage container.

Restore source ⓘ App backup Storage Snapshot (Preview)

Snapshot (Preview) ⓘ

Snapshot taken at Monday, October 23, 2017, 10:41:49 PM PDT

13



Module 02: Implementing and Managing Application Services

Lesson 03: App Service Security

14

Application Security

Customers are responsible for:

- **Developing, deploying, and managing App Service code and content in a secure way**
- **Protecting apps from cyber threats such as:**
 - SQL Injection
 - Session hijacking
 - Cross-site-scripting
 - Application level Man-In-the-Middle (MITM) attacks

15

App Service Authentication

Relies on a trust relationship with an identity provider (IdP):

- **You select an identity provider for your App Service app**
- **The identity provider handles authentication and generates authentication token**
- **The App Service app validates token assertions and authorizes access to its resources**
- **Cookies, JWTs, and bearer tokens validate access for the duration of a session**

Directs authentication requests to an authentication endpoint

Includes built-in supports for the most popular identity providers

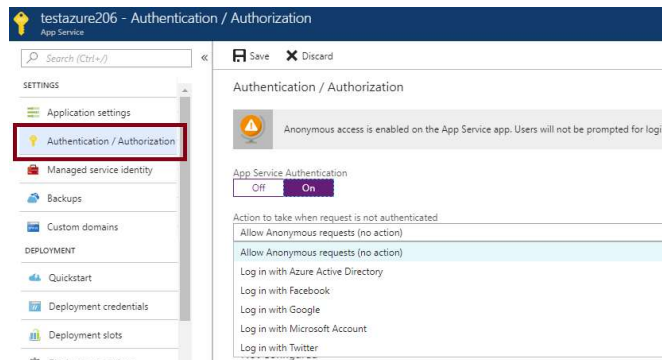


16

Authentication Providers

Configurable directly from the Azure portal:

- **Relies on the Authentication/Authorization feature**
- **Uses federated identity**
- **Offers built-in support for:**
 - **Azure AD**
 - **Facebook**
 - **Google**
 - **Microsoft Account**
 - **Twitter**



17

Module 02: Implementing and Managing Application Services

Lesson 04: Serverless Computing Concepts



18

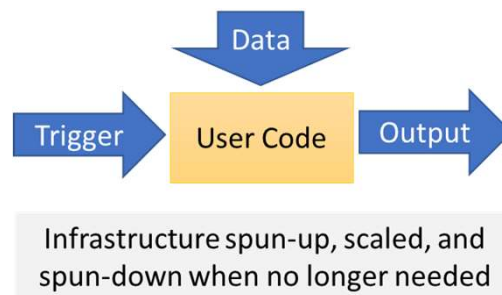
Serverless Computing

Abstracts servers, infrastructure, and operating systems

Reacts to events and triggers in near-real time

Offers a range of benefits:

- **Eliminates management overhead**
- **Allows developers to focus on business logic**
- **Implements flexible scaling**
- **Reduces costs**



19

Serverless Applications

- Compute:
 - **Azure Functions**
- Cloud Messaging:
 - **Event Grid**
 - **Service Bus**
- Workflow Orchestration:
 - **Logic Apps**

20

Module 02: Implementing and Managing Application Services

Lesson 05: Managing Azure Functions

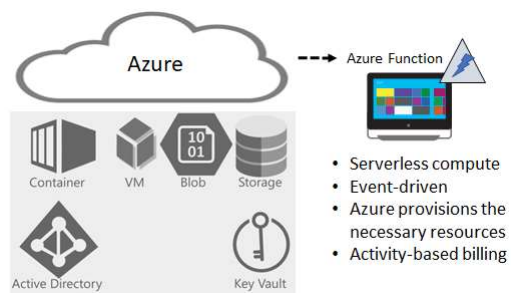


21

Overview of Azure Functions

Serverless compute service:

- **Eliminates the need to explicitly provision or manage infrastructure**
- **Enables running code on-demand or in response to a trigger**
- **Incurs charges only when active**



22

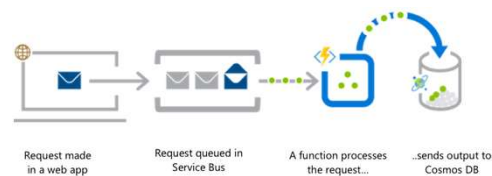
Features of Azure Functions

- **Support for a range of programming languages:**
 - C#, F#, Node.js, Python, PHP, batch, bash, or any executable.
- **Pay-per-use pricing model**
- **Support for custom dependencies:**
 - NuGet and NPM-based libraries.
- **Integration with the most popular OAuth providers:**
 - Azure AD, Facebook, Google, Twitter, and Microsoft Account.
- **Integration with other Azure services and SaaS apps.**
- **Flexible development:**
 - Directly from the Azure portal
 - Via continuous integration through GitHub, VSTS, and other supported development tools.
- **Open-source: available on GitHub.**
- **Ease of code reuse:**
 - Developers can reuse their functions in multiple applications.

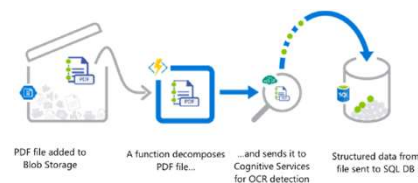
23

Azure Functions (Examples)

- **Web app backend**



- **Real-time file processing**



- **Automation of scheduled tasks**

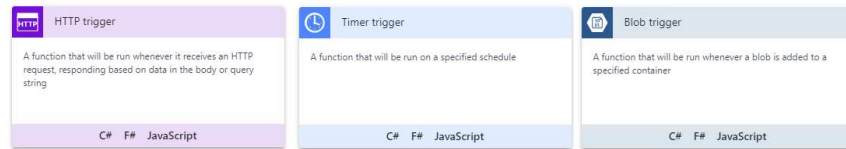


24

Function Templates

Simplify creating functions for a specific trigger type:

- **HTTP**
- **Timer**
- **Blob**
- **Event Hub**
- **GitHub**
- **Webhook**
- **Queue**

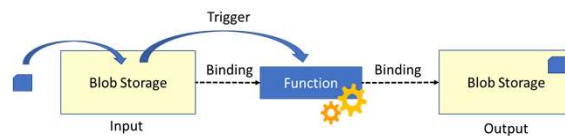


25

Bindings

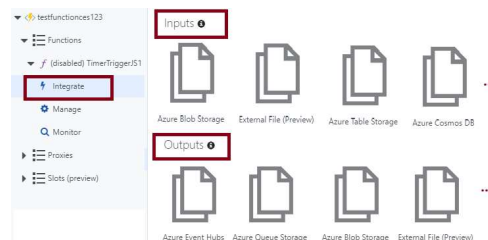
- Link functions to other services as:

- **Inputs**
- **Outputs**



- Configurable in the Azure portal:

- **Accessible via the Integrate link**



26

Module 02: Implementing and Managing Application Services

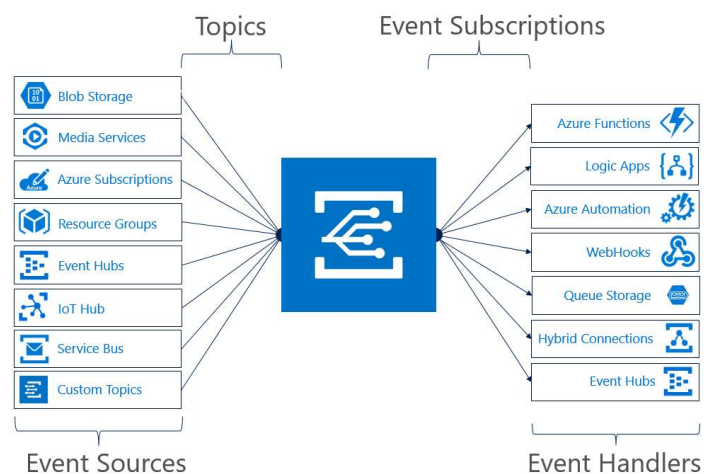
Lesson 06: Managing Event Grid



27

Event Grid Concepts

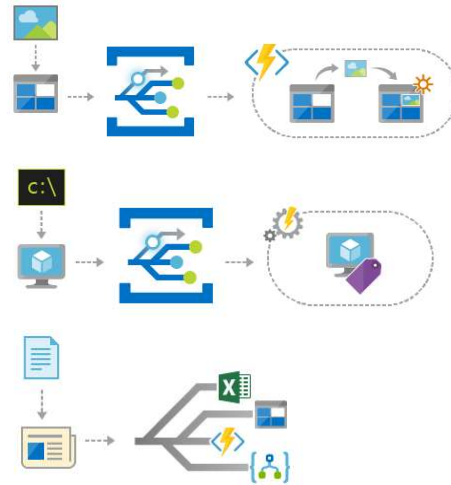
- Event Sources
- Topics
- Event Subscriptions
- Event Handlers



28

Event Grid Examples

- Serverless application architecture
- Ops automation
- Application integration



29



Module 02: Implementing and Managing Application Services

Lesson 07: Managing Service Bus

30

Queues

Azure Service Bus Queues provide:

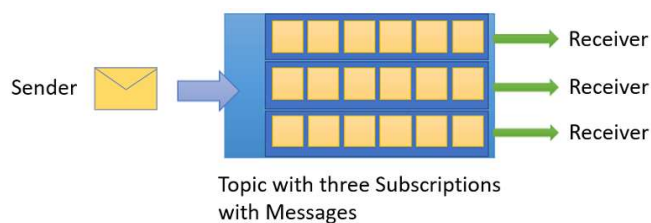
- **Asynchronous, brokered messaging**
- **Temporal decoupling of message senders and receivers**
- **Structured message processing**
- **Publish/subscribe capabilities**
- **First In, First Out (FIFO) message delivery**



31

Topics and Subscriptions

- One-to-many communication in a publish/subscribe pattern
- Useful for scaling large number of recipients
- Each published message is available to topic-registered subscriptions
- Subscriptions use filters to designate messages to receive



32

Module 02: Implementing and Managing Application Services

Lesson 08: Managing Logic App



33

Logic Apps

- Implement scalable integrations and workflows
- Include a visual designer to model workflows
- Offer built-in connectors for cloud and on-premises services and apps
- Provide a range of benefits, including:
 - **Getting started quickly from templates.**
 - **Saving time by designing complex processes using easy to understand design tools.**
 - **Implementing patterns and workflows seamlessly, that would otherwise be difficult to implement in code.**
 - **Customizing your logic app with your own custom APIs, code, and actions.**
 - **Connecting and synchronizing disparate systems across on-premises and the cloud.**

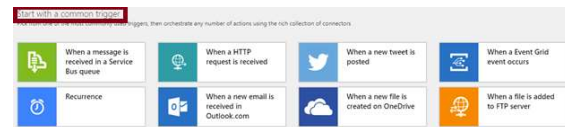
34

Implementing Logic Apps

From the Azure portal:

- **To create a logic app, provide:**

- Name
- Subscription
- Resource Group
- Location



- **To configure a logic app:**

- Use the Logic Apps Designer to:
 - Add triggers
 - Add connectors
 - Leverage templates

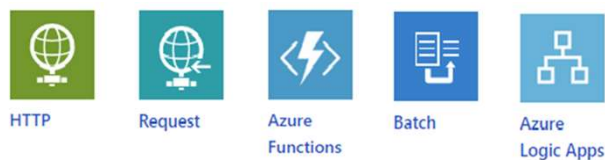


35

Built-in Triggers and Actions

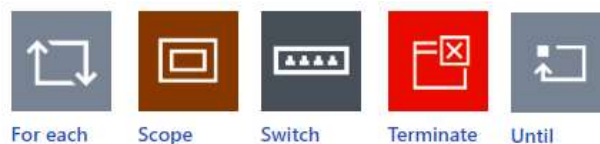
- **Triggers:**

- Recurrence
- HTTP
- Request
- Azure Functions
- Batch
- Logic Apps



- **Actions:**

- Condition
- For each
- Scope
- Switch
- Terminate
- Until



36

Managed Connectors

200+ built-in connectors, including:

- **Managed API connectors:**
 - Azure Blob Storage, Office 365, Dynamics, Power BI, OneDrive, Salesforce, and SharePoint Online.
- **On-premises connectors:**
 - SQL Server, SharePoint Server, Oracle DB, Twitter, Salesforce, Facebook, and file shares.
- **Integration account connectors:**
 - Require a paid-for integration account
 - Transform and validate XML
 - Encode and decode flat files
 - Process B2B messages
- **Enterprise connectors:**
 - Incur extra cost
 - Provide access to enterprise systems:
 - Support SAP and IBM MQ.



37

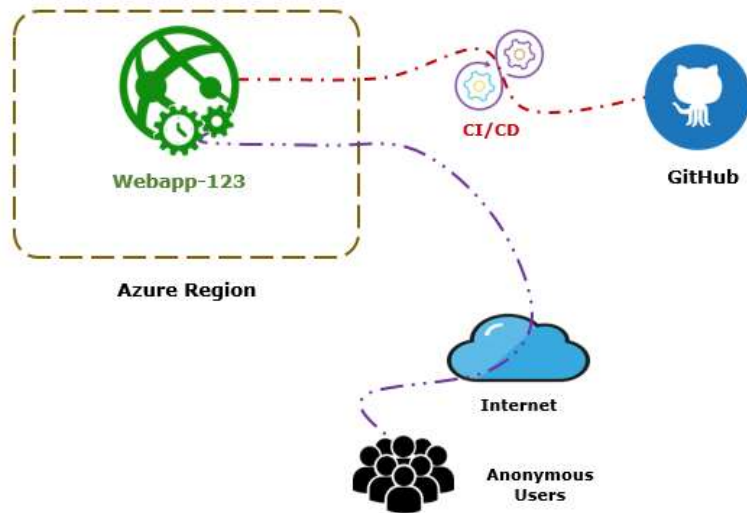
LAB [300TO02-M02-01]

1. Deploy Secure Restricted Website with Continuous Integration & Continuous Delivery.



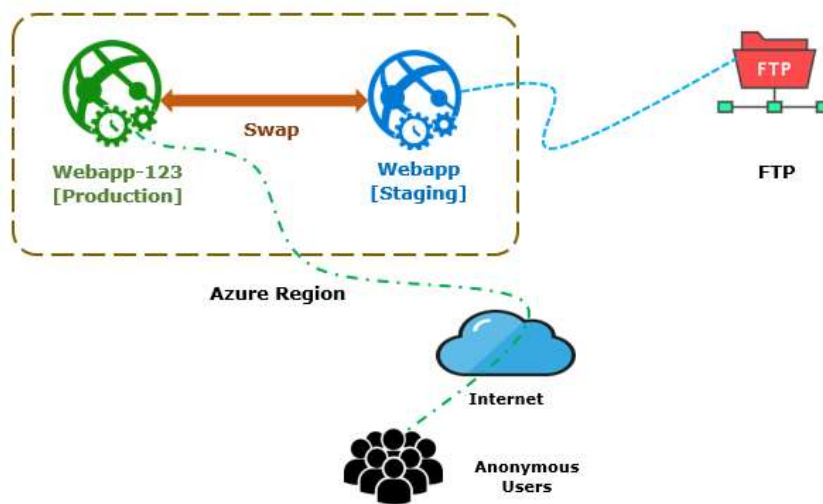
38

Part A



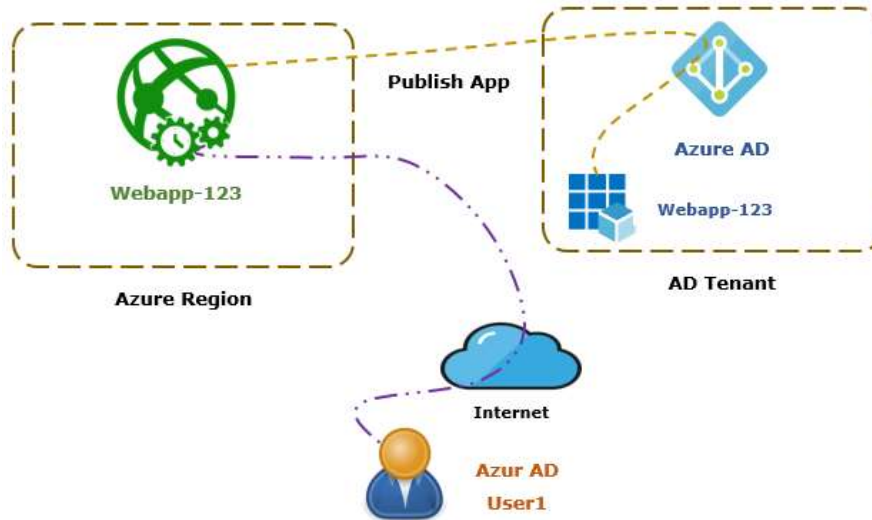
39

Part B



40

Part C



41

LAB [300TO02-M02-01]

1. Deploy Secure Restricted Website with Continuous Integration & Continuous Delivery.

a. Services, Tools & Code used

- i. Azure App Service
- ii. Azure Web App
- iii. HTML Code (3)
- iv. GitHub
- v. FTP
- vi. Azure Web App Slot
- vii. Azure AD Application Integration

Duration: 40 mnts.



42



43