Aditya Varma

adityav@microsoft.com

Abstract

In this document we will provide a step-by-step guidance on how to prepare the Environments to get them ready for conducting the Instructor Led Microsoft Cloud for Health Trainings

MIcrosoft Cloud for Healthcare Training Environments Preparation

Step by Step Preparation Guide

Contents

[Introduction 2](#_Toc88003234)

[Microsoft Cloud for Healthcare in a Day Training Requirements 3](#_Toc88003235)

[Preparation Flow 4](#_Toc88003236)

[Stage 1: One time steps to prepare the tenant 4](#_Toc88003237)

[Task 1: Create Users and Assign Roles and licenses 4](#_Toc88003238)

[Task 2: Create Security Groups and Add users to the Security Group 6](#_Toc88003239)

[Task 3: Create the Dynamics 365 CE Environments 7](#_Toc88003240)

[Stage 2: Create the Master MCH Environment 7](#_Toc88003241)

[Task 1: Create Dynamics 365 CE Environment 8](#_Toc88003242)

[Task 2: Create Customer Self Service Portal in the Environment 11](#_Toc88003243)

[Task 3: Configure Dynamics 365 for Marketing 13](#_Toc88003244)

[Task 4: Configure Omnichannel for Customer Service 16](#_Toc88003245)

[Task 5: Deploy Microsoft Cloud for Healthcare 18](#_Toc88003246)

[Task 6: Give the Administrators Omni-channel security role 23](#_Toc88003247)

[Stage 3: Recurring Tasks to run before every Training 25](#_Toc88003248)

[Task 1: Reset all Dynamics 365 CE Environments 27](#_Toc88003249)

[Task 2: Copy Master onto each of the environments 28](#_Toc88003250)

[Task 3: Turn Off Administration Mode 28](#_Toc88003251)

[Task 4: Change Password of users 29](#_Toc88003252)

[Task 5: Manually Configure Omni Channel for Customer Service on all organizations 29](#_Toc88003253)

[Task 6: Configure Dynamics 365 for Marketing 32](#_Toc88003254)

[Task 7: Install Dynamics 365 Customer Self Service Portal 35](#_Toc88003255)

[Task 8: Run UpdateChannelURLs.exe to update Channel URL on the environment 37](#_Toc88003256)

[Task 8: Test if presence is working accurately 37](#_Toc88003257)

[Conclusion 37](#_Toc88003258)

# Introduction

The Microsoft Cloud for Healthcare solution is composed of various integrated Microsoft technologies (Dynamics 365, Microsoft 365, and Azure) that are tailored specifically for healthcare organizations. To learn more on how to use and set up and the many components of Microsoft Cloud for Healthcare, please refer to the following article on Microsoft Docs: [Set up and configure Microsoft Cloud for Healthcare](https://docs.microsoft.com/en-us/industry/healthcare/configure-cloud-for-healthcare).

The Microsoft Cloud for Healthcare has specific product requirements to light up each scenario and feature. These Dynamics, Azure, and Microsoft 365 pre-requisites are outlined in the table below, some of which are optional and will not be included in this learning path.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scenario | Feature | Description | Dynamics 365 pre-requisites | Azure pre-requisites | Microsoft 365 pre-requisites |
| **Enhance patient engagement** | Patient Outreach | Organize and automate marketing and outreach to patients. | Dynamics 365 Marketing | Azure Health Bot service\* | - |
| Patient Service Center | Monitor and manage patient interactions and communications. Patients can schedule appointments, view care information, and communicate with healthcare staff. | Dynamics 365 Customer Service, Omnichannel for Customer Service, Dynamics 365 Customer Service Insights\* | - | - |
| Patient Access | Patients can schedule appointments, view care information, and communicate with healthcare staff. | Power Apps portals | Azure API for FHIR\* |  |
| **Empower health team collaboration** | Care Management | Provider personnel can create, personalize, and enable new care plans for patients, as well as manage the appropriate care team members. |  | Azure API for FHIR\* | Microsoft Teams\* |
| Home Health | Schedule visits for patients in their own homes. | Dynamics 365 Field Service and Dynamics 365 Customer Service | Azure API for FHIR\* | Microsoft Teams\* |
| Virtual Visits | Schedule and conduct virtual visits remotely with patients. |  | Azure API for FHIR\* | Microsoft Teams |
| Internal health team collaboration | Clinicians and staff can collaborate internally on schedules, documents, tasks, and so on. |  | Azure API for FHIR\* | Microsoft Teams\* |
| **Improve clinical and operational data insights** | Clinical analytics | Access and securely share actionable data to help improve patient care. |  | Azure API for FHIR\*, Azure Health Bot service\* |  |
| Operational analytics | Gain actionable insights to optimize operations. |  | Azure API for FHIR\*, Text Analytics\* |  |
| Data interoperability | Create new healthcare systems of engagement by connecting data from multiple systems of record. |  | Azure API for FHIR\* |  |

# Microsoft Cloud for Healthcare in a Day Training Requirements

For conducting Microsoft cloud for healthcare in a Day training, we need to provide every trainee the following:

1. User Account that is a
   1. Dynamics 365 Service Administrator
   2. Application developer

with following licenses:

* 1. Dynamics 365 Customer Engagement license
  2. Dynamics 365 Customer Service digital Messaging add-on
  3. Microsoft Cloud for Heathcare Addon
  4. Office 365 E3+
  5. Power BI (Free)

1. Dynamics 365 CE Environment with the following applications:
   1. Dynamics 365 for Sales
   2. Dynamics 365 for Customer Service Enterprise
   3. Dynamics 365 for Field Service
   4. Dynamics 365 for Marketing
   5. Omni-channel for Customer service
   6. Microsoft Cloud for healthcare suite of applications
   7. Customer Self Service Portal app
2. Resource access in Azure Subscription in the same tenant so that Azure Healthbots and other Azure services can be installed. Azure resources that will be consumed:
   1. Azure Healthbot

# Preparation Flow

The preparation flow is in three stages as follows:

1. One time step by step of running our scripts for creating users and Azure AD :
   1. Create users and assign roles
   2. Create Security Groups
   3. Adding users to those teams.
   4. Creating the Dynamics 365 CE Environments.
2. One time step by step of creating the Master Environment using Solution Center.
   1. Install all the apps from solution center
   2. Give the Global Administrators who are setting up the Environments the Omnichannel Agent Security Role
3. Step by step of running the following to prep environments before each training:
   1. Recurring scripts before every Training:
      1. Reset the Dynamics 365 CE Environments
      2. Copy the Master Environment to the Dynamics 365 Environments
      3. Remove the Dynamics 365 CE Environments from Admin Mode
      4. Set a New password for the users
   2. Manual Post Installation Steps
      1. Install Omni-Channel on the Dynamics 365 Environments
      2. Install Marketing on the Dynamics 365 Environments
      3. Install Customer Self Service Portal on the Dynamics 365 Environment
      4. Run the WebURL Update script to make presence work

# Stage 1: One time steps to prepare the tenant

The following are the Tasks that need to be run once to prepare the tenant.

1. Create Users and Assign roles and licenses
2. Create Azure AD security groups and add one user per security group
3. Create Dynamics 365 CE environments. We need to create one environment per user for training.

## Task 1: Create Users and Assign Roles and licenses

Run the following powershell script to create users after editing and setting your tenant domain and password for the users. In the example, we are creating mchiaduser01 to mchiaduser100.

In order to run the scripts the following Powershell Modules need to be installed from Powershell command line:

1. Azure AD:

PS> Install-module -name AzureAD

1. Microsoft.PowerApps.Administration.Powershell

PS> Install-Module -Name Microsoft.PowerApps.Administration.PowerShell -RequiredVersion 2.0.1

**Create-Bulk-O365-User.ps1**

------------------------------------------------------------------------------------------------------------------------------------------

Connect-AzureAD

$userSequenceNoStart = 1;

$userSequenceNoEnd = 100;

$orgUserPassword = "[YOURSELECTEDPASSWORD]"

$UserDomainName = "@[YOURTENANTDOMAIN].onmicrosoft.com"

$UserFirstName = "mch"

$UserLastName = "iaduser"

$password = New-Object "Microsoft.Open.AzureAD.Model.PasswordProfile"

$password.ForceChangePasswordNextLogin = $False

$password.Password = $orgUserPassword

for($i = $userSequenceNoStart; $i -le $userSequenceNoEnd; $i++)

{

############### User Creation################################

if($i -lt 10)

{

Write-Output("Creating user with display name " + "$UserFirstName $UserlastName 0$i");

$displayName = $UserFirstName + " " + $UserLastName + "0" + $i

$lastName = $UserLastName + "0" + $i

Write-Output("displayName is $displayName");

$userUPN = $UserFirstName + $UserLastName + "0" + $i + $UserDomainName

Write-Output("userUPN is $userUPN");

$newUser = New-AzureADUser -DisplayName $displayName -PasswordProfile $password -UserPrincipalName $userUPN -AccountEnabled $true -UsageLocation US -MailNickName "NotSet" -GivenName $UserFirstName -Surname $lastName

}

else

{

Write-Output("Creating user with display name " + "$UserFirstName $UserlastName $i");

$displayName = $UserFirstName + " " + $UserLastName + $i

Write-Output("displayName is $displayName");

$userUPN = $UserFirstName + $UserLastName + $i + $UserDomainName

Write-Output("userUPN is $userUPN");

$newUser = New-AzureADUser -DisplayName $displayName -PasswordProfile $password -UserPrincipalName $userUPN -AccountEnabled $true -UsageLocation US -MailNickName "NotSet" -GivenName $UserFirstName -Surname $UserLastName$i

}

Write-Output "Object Id of created users " $newUser.ObjectId

Write-Output("User with display name " + $userUPN + "successfully created....");

############### License Assignement ################################

Write-Output("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

Write-Output("Assigning licenses to user " + "$UserFirstName$i$UserDomainName");

$planName= "ENTERPRISEPREMIUM","FLOW\_FREE","POWER\_BI\_STANDARD","DYN365\_ENTERPRISE\_PLAN1"

$License = New-Object -TypeName Microsoft.Open.AzureAD.Model.AssignedLicense

$SkuIds = (Get-AzureADSubscribedSku | Where-Object { $\_.SkuPartNumber -in $planName }).SkuID

$SkuIds = $SkuIds.split(' ')

ForEach ($SkuId in $SkuIds)

{

$License.SkuId = $SkuId

$LicensesToAssign = New-Object -TypeName Microsoft.Open.AzureAD.Model.AssignedLicenses

$LicensesToAssign.AddLicenses = $License

Set-AzureADUserLicense -ObjectId $userUPN -AssignedLicenses $LicensesToAssign

Write-Output("Assigned license [$SkuId] to user " + "$userUPN");

}

############### Admin Role Assignment################################

Write-Output("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

$D365roleName="Dynamics 365 Administrator"

$AppDevroleName="Application developer"

Write-Output("Assigning [Dynamics 365 Admin Role] to user " + $userUPN);

$D365role = Get-AzureADDirectoryRole | Where-Object {$\_.displayName -eq $D365roleName}

Add-AzureADDirectoryRoleMember -ObjectId $D365role.ObjectId -RefObjectId (Get-AzureADUser | Where-Object {$\_.UserPrincipalName -eq $userUPN}).ObjectID

Write-Output("[Dynamics 365 Admin Role] assigned to user " + $userUPN + " successfully...");

Write-Output("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

Write-Output("Assigning [Application developer] to user " + $userUPN);

$AppDevRole = Get-AzureADDirectoryRole | Where-Object {$\_.displayName -eq $AppDevroleName}

Add-AzureADDirectoryRoleMember -ObjectId $AppDevRole.ObjectId -RefObjectId (Get-AzureADUser | Where-Object {$\_.UserPrincipalName -eq $userUPN}).ObjectID

Write-Output("[App Dev Role] assigned to user " + $userUPN + " successfully...");

Write-Output("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

## Task 2: Create Security Groups and Add users to the Security Group

Run the following powershell script to create AAD Security Groups with name “MCH IAD SG 1” … “MCH IAD SG 100”. Set your $admin1 variable with the appropriate UPN of user who needs to be added as owner for each of the AAD Security Group being created. Replace **[YOURTENANTNAME]** for each of the users getting added to the AAD Security Group.

**CreateAADSG.ps1**

-------------------------------------------------------------------------------------------------------------------------------------

Connect-AzureAD

$aadsgnamebase = "MCH IAD SG"

$admin1 = Get-AzureADUser -ObjectId "admin1@[YOURTENANTNAME].onmicrosoft.com"

$adminid = $admin1.ObjectId

for($i = 1; $i -le 100; $i++)

{

$sgdisplayname = "$aadsgnamebase $i"

$sgdisplayname

$aadsg = New-AzureADGroup -DisplayName $sgdisplayname -MailEnabled $false -SecurityEnabled $true -MailNickname "NotSet"

$mvmsg = Get-AzureADGroup -SearchString $sgdisplayname

$sgid = $mvmsg.ObjectId

if($i -lt 10)

{

$user = Get-AzureADUser -ObjectId "mchiaduser0$i@[YOURTENANTNAME].onmicrosoft.com"

$userid = $user.ObjectId

}

else

{

$user = Get-AzureADUser -ObjectId "mchiaduser$i@[YOURTENANTNAME].onmicrosoft.com"

$userid = $user.ObjectId

}

Add-AzureADGroupMember -ObjectId $sgid.ToString() -RefObjectId $userid.ToString()

Add-AzureADGroupOwner -ObjectId $sgid.ToString() -RefObjectId $admin1id.ToString()

}

## Task 3: Create the Dynamics 365 CE Environments

Run the following script the create MCHInADay\_01 to MCHInaADay\_100 environments with Dynamics 365 CE for Sales, Customer Service and Field Service applications

**CreateMCHEnvironments.ps1**

-------------------------------------------------------------------------------------------------------------------------------------

$username = "admin1@[YOURTENANTNAME].onmicrosoft.com"

$password = ConvertTo-SecureString "[ADMINPASSWORD]" –ASPlainText –Force

Add-PowerAppsAccount -Endpoint "prod" -Username $username -Password $password

$DateTimestamp = Get-Date -Format "MMddyy"

for($i = 1; $i -le 100; $i++)

{

if($i -le 9)

{

$displayname = "MCHInADay\_0$i"

$domainname = "MCHInADay-0$i-$DateTimestamp"

}

else

{

$displayname = "FSIInADay\_$i"

$domainname = "FSIInADay-$i-$DateTimestamp"

}

Write-Host $displayname

Write-Host $domainname

New-AdminPowerAppEnvironment -DisplayName $displayname -DomainName $domainname -LocationName unitedstates -EnvironmentSku Sandbox -ProvisionDatabase -CurrencyName USD -LanguageName 1033 -SecurityGroupId "[YOURPRIVATESECURITYGROUPID]"

}

--------------------------------------------------------------------------------------------------------------------------------------

At this stage the tenant should have the following:

1. All users with appropriate administrative privileges and licenses created
2. All Azure AD Security Groups created with 1 user each
3. All baseline Dynamics 365 CE environments created and added to a private AAD Security Group so that they are not visible to mchiaduser users

# Stage 2: Create the Master MCH Environment

In this stage we will create an MCH environment that will be used as the master Environment to copy onto all the Environments created in Stage 1/Task 3. The process of creating the master Environment is as follows:

### Task 1: Create Dynamics 365 CE Environment

1. Goto <https://admin.powerplatform.com>
2. Click on New

Graphical user interface, application

Description automatically generated

1. Enter the following information:
   1. Name = Name for your Master Environment
   2. Type = Sandbox
   3. Region = Unites States – Default or any other region where MCH is available for your usage
   4. Create a database for this environment = Yes

Graphical user interface, text, application, email

Description automatically generated

1. Click Next
2. Select the following appropriately:
   1. Language = English
   2. URL = To your choosing
   3. Currency = USD ($)
   4. Automatically deploy these applications:
      1. Select Customer Service
      2. Select Field Service
      3. Select Sales
   5. Select Security Group to your private security group so that it is not visible to mchiadusers.

Graphical user interface, application, Teams

Description automatically generated

1. Click **Save**

The Environment preparation will be kicked off and will take a few minutes to be ready.

Graphical user interface, text, application, chat or text message, email

Description automatically generated

Once completed it will be change state to Ready.

Graphical user interface, text, application, email

Description automatically generated

### Task 2: Create Customer Self Service Portal in the Environment

1. Go to <https://make.powerapps.com>

Graphical user interface, application, Teams

Description automatically generated

1. Click on **Environment** on top Right corner
2. Select the Master Environment created in previous step
3. Then Click on **“+ Create**” on Left Nav Bar

Graphical user interface, application, Teams, PowerPoint

Description automatically generated

1. Then search for “**Customer self-service**”
2. Select the Customer self-service Portal and start creation process

Graphical user interface, application

Description automatically generated

1. Click Create
2. Wait for Portal Creation to complete.

Graphical user interface, application

Description automatically generated

With this step the pre-requisite of portal is completed.

### Task 3: Configure Dynamics 365 for Marketing

1. Next Go to <https://admin.powerplatform.com>
2. Select your master Environment by clicking on environment

Graphical user interface, text, application

Description automatically generated

1. Click on Resources drop down on Left Nav Bar

Graphical user interface, text, application, email

Description automatically generated

1. Select Dynamics 365 apps

Graphical user interface, text, application

Description automatically generated

1. Then click triple dots on an unconfigured Dynamics 365 Marketing Application and click Manage.

Note: In order to see the marketing applications in your tenant, your tenant will need the necessary Dynamics 365 Marketing application licenses.

Graphical user interface, application

Description automatically generated

1. Configure Dynamics 365 Marketing Application on your environment by selecting your Master Organization from drop down and select “Use your own webserver” option and click Continue.

Graphical user interface, text, application, email

Description automatically generated

1. Enter an address and click setup

Graphical user interface, text, application, email

Description automatically generated

1. Wait for the marketing application to complete installation.

Graphical user interface, text, application

Description automatically generated

1. Once completed Dynamics Marketing application will show completed status. This step will take several hours.

Background pattern

Description automatically generated

### Task 4: Configure Omnichannel for Customer Service

1. Go to the <https://admin.powerplatform.com>
2. Expand Resources on the left navigation bar and select Dynamics 365 apps.

Graphical user interface, text, application

Description automatically generated

1. Search for Omni, select Omnichannel for Customer Service, and click Manage.

Graphical user interface, application

Description automatically generated

1. Click OK to navigate to the Omnichannel for Customer Service administration center.

Graphical user interface, application

Description automatically generated

1. In the new window, click + Add environment.

Graphical user interface, application, website

Description automatically generated

1. Select your environment from the drop-down menu and click Next.

Graphical user interface, text

Description automatically generated

1. Under Chat, toggle Yes to add Chat.

Application

Description automatically generated with low confidence

1. Click Next all the way to Confirmation and then click Finish. You only need to enable the chat channel for set up.

A picture containing timeline

Description automatically generated

Omnichannel set up will take several hours to complete. Note that all pre-requisite license configurations should be completed before deploying Microsoft Cloud for Healthcare in your environment.

### Task 5: Deploy Microsoft Cloud for Healthcare

In this task, we will deploy all the Microsoft Cloud for Healthcare capabilities.

1. While logged into your Microsoft 365 tenant, open a new tab, and navigate to <https://solutions.microsoft.com> to access the Microsoft Cloud Solution Center.
2. Click the **Microsoft Cloud for Healthcare** button to begin setup.

Graphical user interface, text, application

Description automatically generated

1. All the available Microsoft Cloud for Health capabilities are displayed.

Graphical user interface, application, PowerPoint

Description automatically generated

1. Select any of the Microsoft Cloud for Healthcare capabilities to open the capabilities page. Select All from the “Filter by capability” dropdown.

Graphical user interface, application

Description automatically generated

1. All the solutions are displayed

Graphical user interface, application

Description automatically generated

1. Select “Add all Healthcare solutions” checkbox

Graphical user interface, application

Description automatically generated

1. Click the **Deploy** button on upper right

*Note: Health assistant is not a required Healthcare solution to complete any lab in this series. If you do want to deploy Health assistant, Sales Insights is a required license.*

1. In Additional Components, select both **Sample data** and **Codeable concepts** to include them in installation. This is important as the lab content is based off this data. Click **Next**.

Graphical user interface, text, application, email

Description automatically generated

1. Select your Environment from the drop-down menu, enter a Name for the deployment, accept the Terms of service, and click Next.

Graphical user interface, text, application

Description automatically generated

1. You will now see that all pre-deployment dependencies are installed and configured. Click Deploy to complete the deployment process.

Graphical user interface, application, email

Description automatically generated

1. Congratulations! Deployment is underway. The Microsoft Cloud for Healthcare deployment will take several hours to complete. You can monitor this screen to check the deployment status.

Graphical user interface, text, application

Description automatically generated

1. Healthcare apps will also appear in the Apps page in [Power Apps](https://make.powerapps.com/) as they are deployed.

Graphical user interface, application

Description automatically generated

### Task 6: Give the Administrators Omni-channel security role

In this task we will give the global administrator accounts being used to setup all the environments the “Omnichannel agent” security role in order to facilitate in testing whether presence functionality works at the end of the preparation of MCH in a day environments

1. Go to <https://admin.powerplatform.com>
2. Select the master environment

Graphical user interface

Description automatically generated with medium confidence

1. Click on “See All” under users

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Select your administrator user and select Manage Roles in the User form



1. Add the following security roles:
   1. Healthcare User

Graphical user interface, text, application

Description automatically generated

* 1. Omnichannel agent

Graphical user interface

Description automatically generated

1. Click **OK**.

The security roles are added to the user accounts that are being used to setup the environments. Later when the master environment is copied on to all the organizations, these administrative users will automatically have the security roles and can test if presence works after all the setup steps are completed.

With this the one time action of creating a Master Environment completed. This master environment will be used to copy over all the Training environments before every training for faster reset and deployment using scripts. This Master Environment should not be touched so that it remains in a clean state throughout. Keep it up to date by installing the updates from Solution Center whenever Microsoft Cloud for Healthcare solutions are updated.

# Stage 3: Recurring Tasks to run before every Training

In this stage, we will walk through the tasks we need to run to prepare the environments before every training. The recurring steps do the following actions:

1. Reset all the Dynamics 365 CE Environments using powershell script to wipe out all the changes done by trainees in the previous training.
2. Copy the Master Environment created in Stage 2 onto all the environments using powershell script so that Master Environment data and configuration is copied onto each environment and each environment is brought to ready state.
3. Run powershell script to bring each environment out of admin mode. After a copy is done, each environment defaults to admin mode only. We need to remove it from admin mode so that users can log in.
4. Reset password for all mchiaduser1 … mchiaduser100 using powershell script so that previous trainees are restricted from accessing the tenant.
5. Manually install omni-channel for customer service on each environment via UI. There is no API for this task.
6. Manually install Dynamics 365 for Marketing on each environment via UI. There is no API for this task.
7. Install a new Dynamics 365 Customer Self Service Portal
8. Change the WebRedirectURL for each environment to fix the wrong webredirecturl due to copy of the master environment onto each environment

Tasks 1 to 4 are automated using powershell scripts. Each of the powershell scripts read data from a csv file and run accordingly. The format of the csv file needs to be as follows:

Example: MCHInADay\_1to100.csv

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FriendlyName | EnvironmentId | Label | Notes | SecurityGroupName | SecurityGroupId | EnvUserUPN | DomainName | OrgUrl |
| MCHInADay\_01 | 58a09e0d-e06c-46ce-b4e6-0205f4a1ae9c | Start Point Backup | This is the start point backup | Mch in a day 1 | 6d2b2c7b-e87a-4b46-ae15-26f805bc3ceb | mchIADUser01@[YOURTENANTNAME].onmicrosoft.com | Mchinaday-01 | <https://MCHInADay-01>-[MMDDYY].crm.dynamics.com |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

EnvironmentId can be captured using Kusto Query (to be edited to query your tenant):

OrganizationDetails

| where TenantId == "YOURTENANTID"

| where FriendlyName contains "MCHInaDay"

| where TIMESTAMP >= datetime(2021-11-08 00:00)

| project FriendlyName, BAPEnvironmentId,"Start Point Backup", "This is the start point backup", "In a Day User", SecurityGroupGUID, "mchiadUser01@[YOURTENANTNAME].onmicrosoft.com", DomainName, strcat("https://",DomainName,".crm.dynamics.com")

| order by FriendlyName asc

| summarize any(\*) by BAPEnvironmentId

SecurityGroupName and SecurityGroupID can be obtained from O365 by exporting AAD Security Groups.

Graphical user interface, text, application

Description automatically generated

## Task 1: Reset all Dynamics 365 CE Environments

In this task we will reset all the Dynamics 365 CE environments using powershell script. The powershell script reads a data from a csv file and based on that data resets each of the environments.

Step1\_ResetEnvironments.ps1

-----------------------------------------------------------------------------------------------------------------------------------------

$username = "admin1@[YOURTENANTNAME].onmicrosoft.com"

$password = ConvertTo-SecureString "[PASSWORD]" –ASPlainText –Force

Add-PowerAppsAccount -Endpoint "prod" -Username $username -Password $password

$DateTimestamp = Get-Date -Format "MMddyy"

#### This is the full path to your csv file #####

Import-Csv C:\PS\IAD\MCH\MCHInADay\_1to100.csv | ForEach-Object {

$FriendlyName = $\_.FriendlyName

$EnvironmentID = $\_.EnvironmentId

$securitygroupId = $\_.SecurityGroupId

$securitygroupName = $\_.SecurityGroupName

$envUserUPN = $\_.EnvUserUPN

$domainName = $\_.DomainName

$url = "$domainName-$DateTimestamp"

Write-Host $url

$resetRequest = [pscustomobject]@{

FriendlyName = $FriendlyName

DomainName = $url

Purpose = "MCHIAD"

BaseLanguageCode = 1033

Currency = [pscustomobject]@{

Code = "USD"

Name = "USD"

Symbol = "$"

}

SecurityGroupId = $securitygroupId

Templates = @()

}

Reset-PowerAppEnvironment -EnvironmentName $EnvironmentID -ResetRequestDefinition $resetRequest -WaitUntilFinished $false

}

------------------------------------------------------------------------------------------------------------------------------------------

## Task 2: Copy Master onto each of the environments

In this task we will copy over the master environment over each of the environments listed in the csv file. The powershell walks through each environment in the csv file and copies over the master onto that environment.

Step2\_CopyEnvironments.ps1

------------------------------------------------------------------------------------------------------------------------------------------

$username = "admin1@[YOURTENANTNAME].onmicrosoft.com"

$password = ConvertTo-SecureString "[PASSWORD]" –ASPlainText –Force

Add-PowerAppsAccount -Endpoint "prod" -Username $username -Password $password

#Source is YOUMASTERENVIRONMENT. To be updated by Ravindran

$souceEnvironmentId = "[YOURMASTERENVIRONMENT\_ID]"

Import-Csv C:\PS\IAD\MCH\MCHInADay\_1to100.csv | ForEach-Object {

$FriendlyName = $\_.FriendlyName

$EnvironmentID = $\_.EnvironmentId

$securitygroupId = $\_.SecurityGroupId

$securitygroupName = $\_.SecurityGroupName

$envUserUPN = $\_.EnvUserUPN

$copyToRequest = [pscustomobject]@{

SourceEnvironmentId = $souceEnvironmentId

TargetEnvironmentName = $FriendlyName

TargetSecurityGroupId = $securitygroupId

CopyType = "FullCopy"

}

Copy-PowerAppEnvironment -EnvironmentName $EnvironmentID -CopyToRequestDefinition $copyToRequest

}

------------------------------------------------------------------------------------------------------------------------------------------

## Task 3: Turn Off Administration Mode

In this task each of the environments listed in the csv file are removed from Administration Mode. This enables all users to be able to log into the environment.

Step3\_TurnOffAdminMode.ps1

------------------------------------------------------------------------------------------------------------------------------------------

$$username = "admin1@[YOURTENANTNAME].onmicrosoft.com"

$password = ConvertTo-SecureString "[PASSWORD]" –ASPlainText –Force

Add-PowerAppsAccount -Endpoint "prod" -Username $username -Password $password

Import-Csv C:\PS\IAD\MCH\MCHInADay\_1to100.csv | ForEach-Object {

$FriendlyName = $\_.FriendlyName

$EnvironmentID = $\_.EnvironmentId

Write-Host "---------------------------------"

Write-Host "Kicking off Disable of Admin Mode: EnvironmentID: " $EnvironmentID "; TargetEnvName: " $FriendlyName

Set-AdminPowerAppEnvironmentRuntimeState -EnvironmentName $EnvironmentID -RuntimeState "Enabled" -WaitUntilFinished $true

}

------------------------------------------------------------------------------------------------------------------------------------------

## Task 4: Change Password of users

In this task we will change the password for all the users so that previous training passwords are reset.

Step4\_SetPassword.ps1

------------------------------------------------------------------------------------------------------------------------------------------

Connect-AzureAD

$newpassword = ConvertTo-SecureString "[NEWPASSWORD]" -AsPlainText -Force

Import-Csv C:\PS\IAD\MCH\MCHInADay\_1to100.csv | ForEach-Object {

$envUserUPN = $\_.EnvUserUPN

$user = Get-AzureADUser -ObjectId $envUserUPN

Write-Host "Setting Password for " $user.UserPrincipalName " ObjectID = " $user.ObjectId " To password: " $newpassword

Set-AzureADUserPassword -ObjectId $user.ObjectId -Password $newpassword -ForceChangePasswordNextLogin $False -EnforceChangePasswordPolicy $False

}

------------------------------------------------------------------------------------------------------------------------------------------

## Task 5: Manually Configure Omni Channel for Customer Service on all organizations

The omnichannel for customer service needs to be configured after the master organization is copied over each of the organizations. The configurations of master organization will not work on the copied organization. These steps have to be configured manually as there are no API endpoints available to automate this task.

1. Go to the <https://admin.powerplatform.com>
2. Expand Resources on the left navigation bar and select Dynamics 365 apps.

Graphical user interface, text, application

Description automatically generated

1. Search for Omni, select Omnichannel for Customer Service, and click Manage.

Graphical user interface, application

Description automatically generated

1. Click OK to navigate to the Omnichannel for Customer Service administration center.

Graphical user interface, application

Description automatically generated

1. In the new window, click + Add environment.

Graphical user interface, application, website

Description automatically generated

1. Select each environment from the drop-down menu and click Next.

Graphical user interface, text

Description automatically generated

1. Under Chat, toggle Yes to add Chat.

Application

Description automatically generated with low confidence

1. Click Next all the way to Confirmation and then click Finish. You only need to enable the chat channel for set up.

A picture containing timeline

Description automatically generated

Omnichannel set up will take several hours to complete.

## Task 6: Configure Dynamics 365 for Marketing

Dynamics Marketing needs to be installed on each organization post copy of the master organization. This step is a manual step that needs to be kicked off from UI as there are no API end points available to automate this task.

1. Next Go to <https://admin.powerplatform.com>

Graphical user interface, text, application

Description automatically generated

1. Click on Resources drop down on Left Nav Bar

Graphical user interface, text, application, email

Description automatically generated

1. Select Dynamics 365 apps

Graphical user interface, text, application

Description automatically generated

1. Then click triple dots on an unconfigured Dynamics 365 Marketing Application and click Manage.

*Note: In order to see the marketing applications in your tenant, your tenant will need the necessary Dynamics 365 Marketing application licenses.*

Graphical user interface, application

Description automatically generated

1. Configure Dynamics 365 Marketing Application on your environment by selecting your Organization from drop down and select “Use your own webserver” option and click Continue.

Graphical user interface, text, application, email

Description automatically generated

1. Enter an address and click setup

Graphical user interface, text, application, email

Description automatically generated

1. Wait for the marketing application to complete installation.

Graphical user interface, text, application

Description automatically generated

1. Once completed Dynamics Marketing application will show completed status. This step will take several hours.

## Task 7: Install Dynamics 365 Customer Self Service Portal

Since each Environment was reset and master environment copied over it. The portal applications are deleted during Environment reset. The portal configurations are copied over from master but a portal itself will need to be installed from “Customer Self Service” Portal Template. The advantage though is that since the master has all the portal solutions, the portal install will be a quick process.

1. Go to <https://make.powerapps.com>

Graphical user interface, application, Teams

Description automatically generated

1. Click on Environment on top Right corner
2. Select each Environment in training
3. Then Click on “+ Create” on Left Nav Bar

Graphical user interface, application, Teams, PowerPoint

Description automatically generated

1. Then search for “Customer self-service”
2. Select the Customer self-service Portal and start creation process and name it “Lamna Healthcare Patient Portal”

Graphical user interface, application

Description automatically generated

Lamna Healthcare Patient Portal

1. Click Create
2. Wait for Portal Creation to complete.

Graphical user interface, application

Description automatically generated

## Task 8: Run UpdateChannelURLs.exe to update Channel URL on the environment



1. Download UpdateChannelURLs.zip
2. Open Properties and Unblock the zip file
3. Extract the zip file to a folder
4. Open cmd and go to the extracted folder

Run>UpdateChannelURLs.exe /username:yourid@[YOURTENANTNAME].onmicrosoft.com /password:yourpassword /csvfile:C:\PS\IAD\MCH\MCHInADay\_1to100.csv

1. The exe will connect to each Environment listed in the csv using webapi endpoint and update the ChannelURL data to the accurate data for the environment

## Task 8: Test if presence is working accurately

1. Open Patient Service center as one of the administrators who was give “Omnichannel agent” security role
2. Check if the presence of the logged in user in Patient service center shows Available.
3. If the presence spins and changes to available, then the users’ presence check works accurately which is needed for Healthcare Bot escalation to Human Lab to work accurately.

# Conclusion

With all the above steps completed the environments should be fully setup for the Microsoft Cloud for healthcare in a Day Training to be conducted. Each user will have their own Dynamics 365 Environment with all the Microsoft Cloud for healthcare applications fully installed. The labs will then train the users on functionality and post installation configurations.

Graphical user interface, application

Description automatically generated