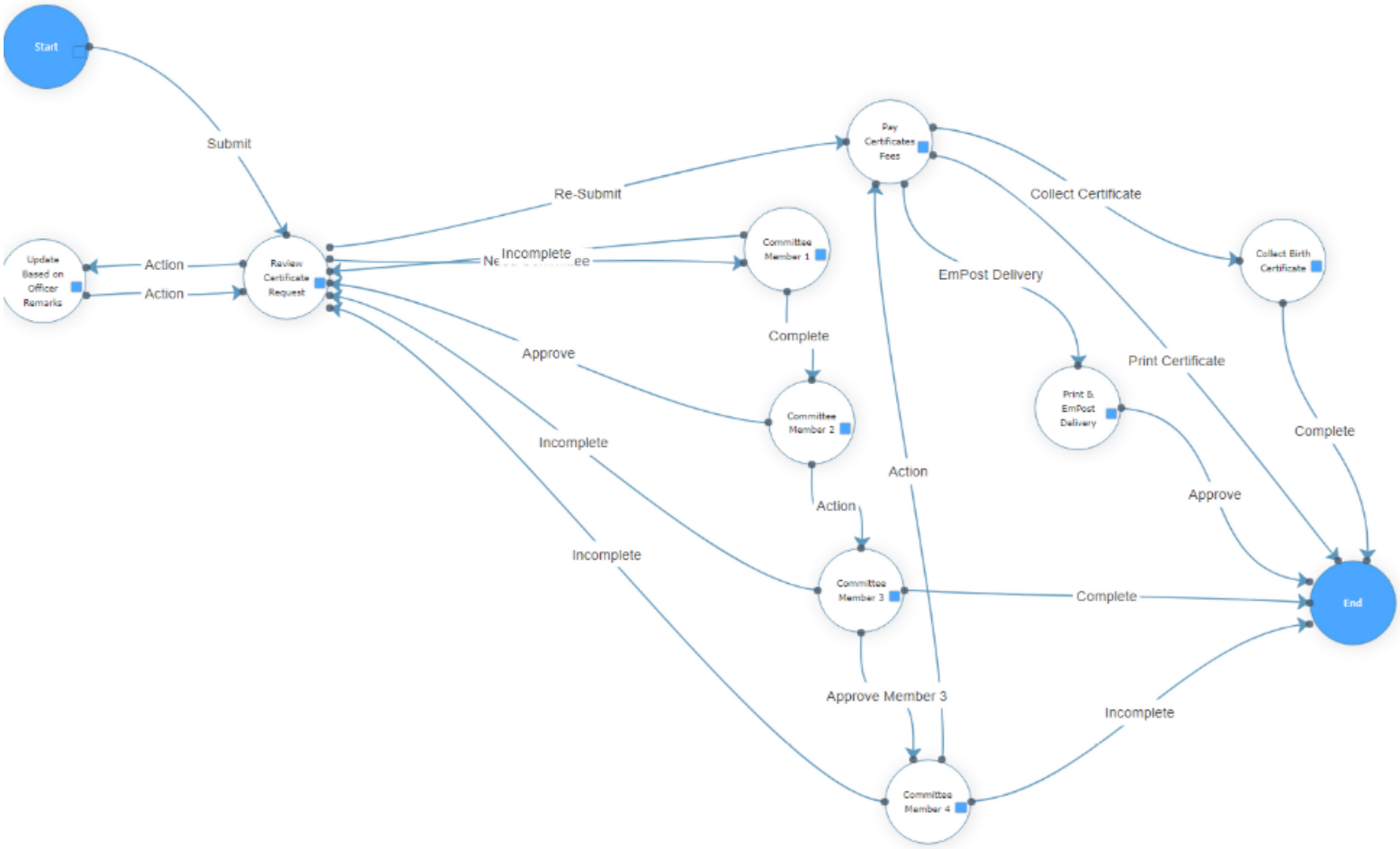
netways

Process Automation and Configuration Engine - PACE



netways

Prepared for

Microsoft Appsource

30-May-24

Version 1.3 Final

Prepared by

Netways Presales Team

Change Record

Date	Author	Version	Change Reference
20/05/2022	UAE Presales Team	1	Creation
23/05/2022	UAE Presales Team	1.1	Content Update
25/05/2022	Ali Mostafa	1.2	Adding Product description and Features
OBJ OBJ 30/05/2022	OBJ OBJ Ali Mostafa	OBJ OBJ 1.4	OBJ OBJ OBJ OBJ

Abbreviations

Abbreviation	Explanation
PACE	Process Automation and Configuration Engine
D365	Dynamics 365
CRM	Customer Relationship Management
C#	C (Sharp) is a programming language
DMS	Document Management System
OOB	Out of the Box

Table of Contents

What is PACE?

PACE (Process Automation and Configuration Engine) is a Netways accelerator that works Add-On to Dynamics 365 Customer engagement to streamline the design of State Machine workflows.
It is a Managed CRM solution which includes a library of reusable actions, plugins and UI widgets that enhance the following Dynamics 365 features

- Business Process Management.
- Automatic assignment and Task Management.
- Escalation and Delegation.
- Document Management

PACE play a key role in the digital transformation of a firm on top of Dynamics, it performs the following

- Empowering employees by allowing users to have full insights about service requests from submission till closing, it helps users as well to validate required information and documents and take the right decision with best user experience.
- Engaging Customer: PACE add external stakeholders like customers, vendors, whether they are individuals or corporate in the process, by assigning them tasks or re-submit missing information.
- Optimize operations: PACE reduces the costs of operation maintenance or improving services by providing a No Code/Low Code intuitive interface throughout the design of your business processes.

Who Can Configure PACE?

Analysts, business users, and developers can use and configure PACE since it's user-friendly. It is a low code/ no-code tool that depends on CRM entities to prepare the needed workflows. An analyst or a business user can easily and abstractly create starting and ending points of a flow, along with different stages between the initiation and the end of a process.

What Are PACE Features?

PACE has a list of features that make business process flows more flexible in configuration.

Find below the most important features:

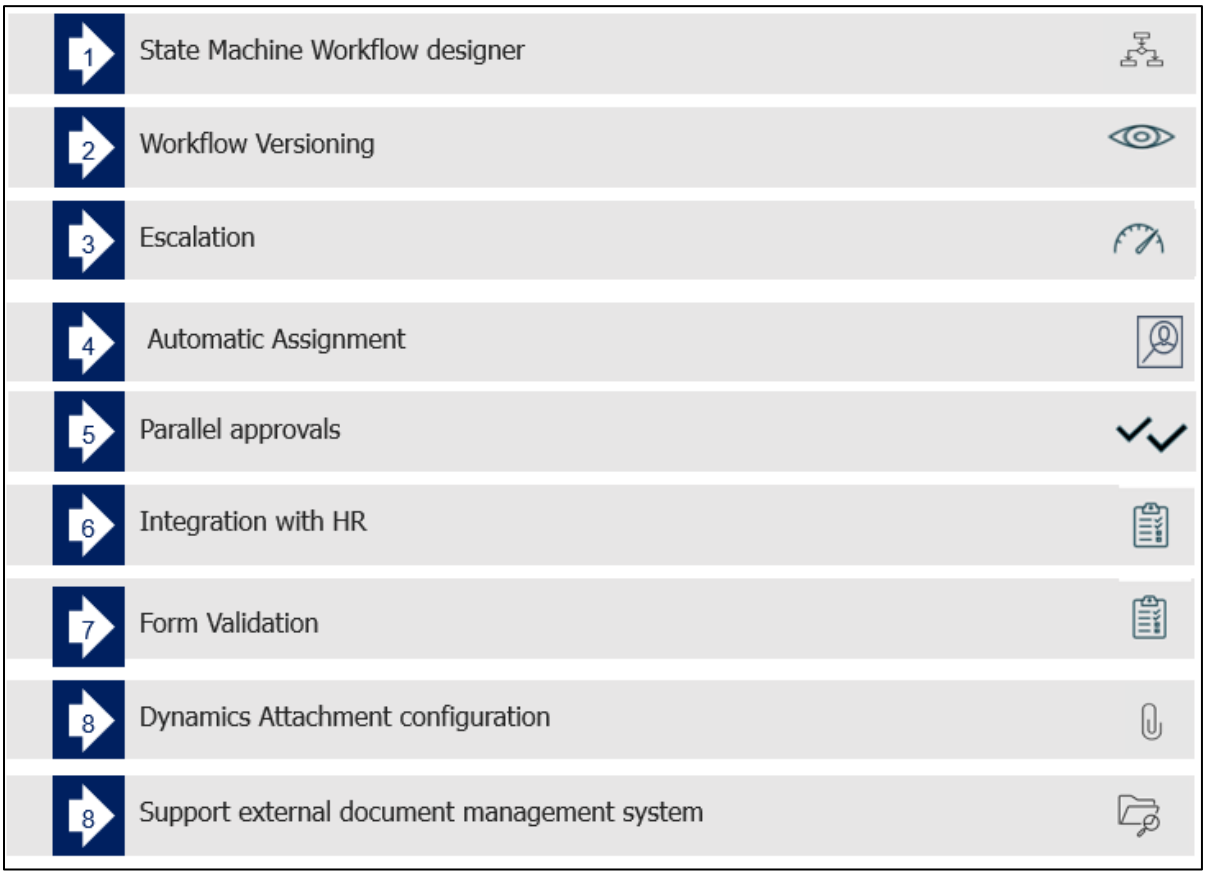


Figure 1: PACE Features

1. State Machine Workflow designer that supports human based Actions like Form decisions and event-based actions.
2. Workflow Versioning
3. Escalation
    - Configure Escalation and SLA on Stage and Task Level.
    - Escalation Matrix support.
4. Automatic Assignment
    - Users, Teams, or Queues
    - Based on Condition
    - Random Assignment to Team members
    - Assignment based on users' utilization.
    - Based on a Field in the form
5. Parallel approvals or voting for users in a single team or multiple teams.
6. Integration with HR to sync users' calendar, holidays and day offs, so that the engine excludes them from automatic assignment
7. Form Validation:
    - Users cannot submit if some configured fields are not filled
    - Users cannot select a decision unless the required fields for that action are filled.
    - Comments are required during submission based on a condition or criteria.
    - Some Actions or decisions can be hidden based on a condition or criteria.
8. Support for Internal Employees and external stakeholders like customers, vendors, contractors, and service providers.
9. Dynamics Attachment configuration: You can define attachments required for each service. Note that attachments show on forms like fields not as annotations.

10. Support for External document management system. PACE provides standard APIs to integrate with any document management system, so that documents are secured externally, and only Task owners can access them.
11. Integration with Dynamics Business Process Flow to map current sage with BPF Stage.

PACE Components

In this section we will list down the D365 CE components that have been added to PACE.

1. Plugins
   1.1. **Netways.Apps.Plugins.NewRequestTypePlugin** : run when defining new request Type, it creates some reusable fields on the request, and automatically register 2 plugins on selected entity.
   1.2. **Netways.Apps.Plugins.OnCreateRequest :** Run when creating a new request, it sets the corresponding workflow version and sets initial stage. It creates a record as well in the ntwapp_request entity.
   1.3. **Netways.Apps.Plugins.OnUpdateRequest:** Run whenever Action is made on a request, it moves the record to a new stage, create a task, and assign both task and record to corresponding user/team.
2. Code Activities
   2.1. **NextStageWorkflow**: it used whenever a Stage should change based on external event like Invoice or a payment. Then you can call this activity to ask PACE to move current record to next stage assuming that there is one available action to follow. Example: a process is waiting for payment, and there is no other available action. This Code Activity use only one parameter, it is an "EnityReference" on the request itself.

2.2. **MoveNextStageWorkflow**: like NextStageWorkflow, but when using this activity, can have a stage with more than one available action. In that case there is additional parameter should be provided. You should set "MultipleOptions" to True and provide the sequence number of the action you want to take.

2.3. **MoveNextWithBPFWorkflow** : Like MoveNextStagWorkflow but this Code Activity can move a corresponding BPF process assigned to current record if available.

2.4. NextStageWorkflow

3. Web Resources

3.1. **ntwapp_DialogSubmit**: it checks current stage in the Action Modal Form, and list down available actions, then save the selected decision in the "ntwapp_actiontaken" on the request record.

3.2. **ntwapp_onSubmitRequest**: runs when the Submit Button is clicked, it launches the Modal Form, it includes the display rule script as well, that is responsible to show or hide the submit button.

3.3. **ntwapp_WorkflowDrawScript**: it includes all scripts related to the Workflow Designer.

3.4. **ntwapp_WorkflowDraw**: it includes the HTML code need in the Workflow Designer.

3.5. **ntwapp_jsplumb_demo.js**: it is an open-source library used for Workflow designer.

3.6. **ntwapp_stepper_index.html:** it includes the HTML code need in the Stepper Widget that ou can embed in a Request Form instead of Dynamics 365 CE Business process flow stepper, as D365 stepper still can allow users to back, while PACE stepper Widget doesn't.

3.7. **ntwapp_stepper_script.js** : it includes the JavaScript code need in the Stepper Widget

3.8. **ntwapp_stepper_style.css**: it includes the CSS code need in the Stepper Widget

3.9. **ntwapp_stepper_style_rtl.css**: it includes the CSS code need in the Stepper Widget in RTL mode.

How is PACE Working in the Background?

The main entity of PACE is "**Request Type**", and the purpose of the Request Type Entity is to link the services/requests to D365 Entities.

When you define a new Service in the "**Request Type**" entity, a Plugin called "NewRequestTypePlugin" in the Assembly **Netways.Apps.Plugins.dll** runs and performs the follwing.

1. PACE creates automatically new 14 fields in the selected entity as follows.

    - **ntwapp_currentstage**: it is lookup "on stwapp_stage entity, it represents the current Stage of the record.
    - **ntwapp_workflowversion**: it is the current workflow version applied to current request. If a request has been initiated using an old workflow version, then we enhanced the workflow with a new version then new request will use new version, and old request will be keep using old version.
    - **ntwapp_isstagevalid**: the button submit in the request will not work if this Boolean field if "false", so you can create some business rules that set this field to false in case of some validation you need to apply before taking decision.
    - **ntwapp_actiontaken** : represents the last action taken on this record. Note that PACE resets this field to null upon each stage transition, then save last decision taken in the task entity.
    - **ntwapp_lastcomment**: represents the last comment taken on this record. Note that PACE resets this field to null upon each stage transition, then save last decision taken in the task entity.
    - **ntwapp_requeststatus**: it can be Draft (not submitted yet), In Progress, or Closed.
    - **ntwapp_approvalstatus**: represents the final approval status of the request,
        - if "ntwapp_requeststatus" is Closed then "ntwapp_approvalstatus" shojld be one of the follwing statuses: "Approved", "Rejected", or "Cancelled".
        - If "ntwapp_requeststatus" is in Progress, then "ntwapp_approvalstatus" is Null, and you check "ntwapp_currentstage".
        - If ""ntwapp_requeststatus"" is draft, then "ntwapp_currentstage" is "Start" and "ntwapp_approvalstatus" is still null.
    - **ntwapp_task**: it represents a look to the current task assigned on the current record.
    - **ntwapp_token**: it is 32 digits alphanumeric random auto generated string. It is used to allow users to attach documents to the current record. If the user knows the value of this field, that means he can upload files to current record. Note that "ntwapp_token" can be reset to a new random value on each transition from a stage to stage based on a setting "ntwapp_resettoken" on the stage level true, so that

users who took previous decisions on a record cannot upload files after moving to next stage.

- **ntwapp_externalcreator**: If the Workflow engage external users, then "ntwapp_external creator" is the contact who created the record from outside the system.
- **ntwapp_externalowner**: If the Workflow engage external users, then "ntwapp_ externalowner" is the contact that currently owns the record.
- **ntwapp_customer**: If the Workflow engage external users, then "ntwapp_customer" is the current external owner of the record, it is a lookup of type customer, so it can be an account or contact. If the workflow is targeting individuals then "ntwapp_externalowner" and "ntwapp_customer" will be the same, but If the workflow is targeting corporates then "ntwapp_customer" will be the account assigned to that record, and "ntwapp_externalowner" will be the current contact who belongs to that account that has been assigned to that record.
- **ntwapp_webrole**: If the Workflow engage external users, then "ntwapp_webrole" is external role assigned to the record in the current account. note that Webrole is a custom entity in PACE like system role that corresponds to system users. Note that a record can be assigned to contact "ntwapp_externalowner" or a role in corresponding account "ntwapp_webrole".

Note that the Implementer who is using PACE must add all the fields above to the Request form as hidden.

2. Registers automatically 2 Dynamics CRM Plugins called "OnCreateRequest" and "OnUpdateRequest" on that selected entity, which exist as well in the Assembly **Netways.Apps.Plugins.dll.**

    2.1.1. The Plugin **OnCreateRequest** runs whenever a new Request is created. It checks the request types associated with current entity.

        2.1.1.1. If there is only one associated request, then it set the corresponding Start Stage in "ntwapp_currentstage" and current workflow version in "ntwapp_workflowversion".

        2.1.1.2. If there is more than one request type associated with current entity, then PACE checks the fields "ntwapp_conditionfield" and "ntwapp_conditionvalue" and set the Stage and version correspondingly.

2.1.2. The Plugin **OnUpdateRequest** runs whenever the "ntwapp_action" field in the request is updated. It checks current Stage and the Action taken, then compare with configuration in "ntwapp_stage" and "ntwapp_stageaction" entities the performs the following.

2.1.2.1. Run the selected process in the Pre-Workflow configuration in Stage level.

2.1.2.2. Assign the record to the corresponding user/team based.

2.1.2.3. Calculate the SLA based in user's calendar.

2.1.2.4. Creates tasks for the user/team.

2.1.2.5. Moves the record to a new Stage.

2.1.2.6. Run the selected process in the "Post-Workflow" configuration in Stage level.

3. Once the 14 Fields have been added to the Form, the button submit will show automatically on the form, and it is using Display rules in Dynamics CRM Ribbon, the ribbon button works on all entities in the system and shows only if the 14 fields above exist on the form. Note that the submit button is responsible to Popup the decision screen.

   Note that: The Display rules exists in the function "ShowSubmitBtn" in the web resource "ntwapp_onSubmitRequest".

4. Once the button "Submit" is clicked, it performs the follwing tasks.

   4.1. it calls a script in web resource "ntwapp_DialogSubmit"

   4.2. Get the corresponding actions for current record and shows them in a Modal form.

   4.3. Takes the selected decision and set it in the "ntwapp_action" in the form then saves the current record.

5. Once the ntwapp_action is changed and the record is saved, the registered plugin "Netways.Apps.Plugins.OnUpdateRequest" works and start processing. Please check Paragraph OnUpdateRequest

How to Install PACE

PACE is a managed solution [**Netways Apps**] that needs to be installed on the CRM organization.

You can upload it by following the below steps:

1. Go to **Advanced Settings**

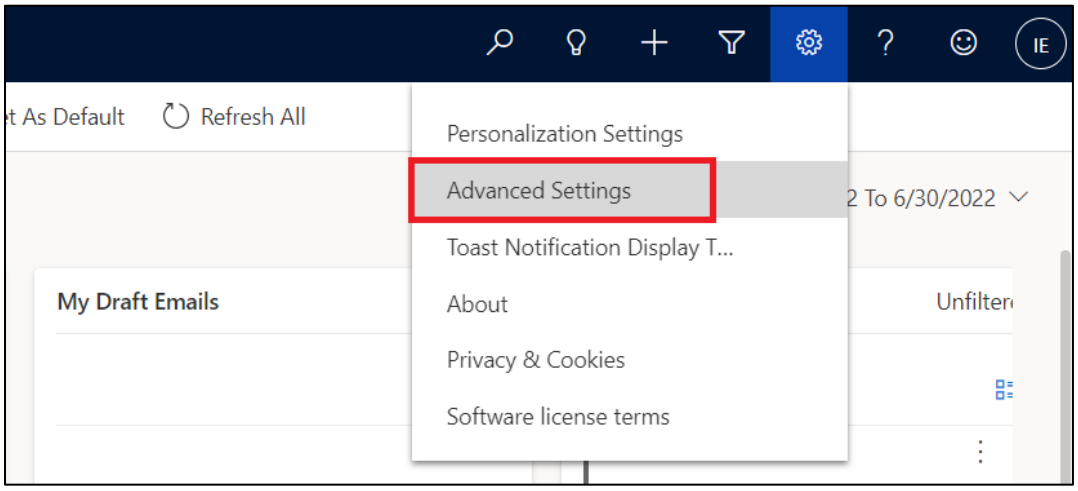


Figure 2 Dynamics 365 Advanced Settings Screenshot

2. Go to Settings then **Solutions**

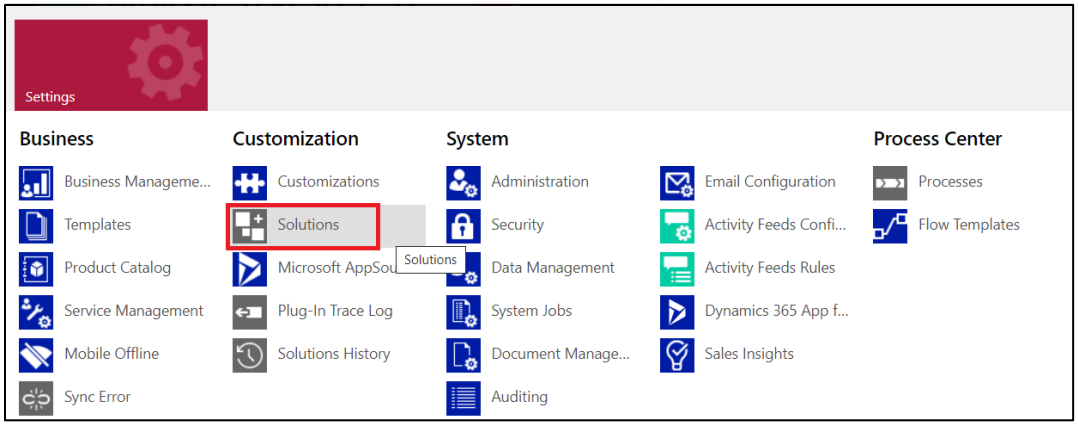


Figure 3 Dynamics 365 Solutions Screenshot

3. Click on **Import** Button

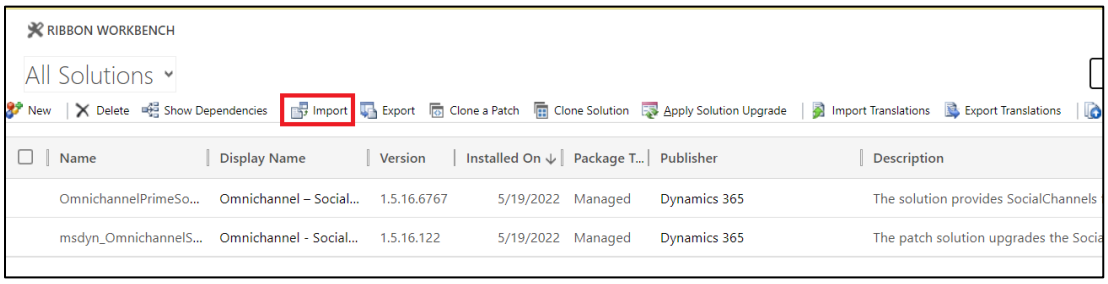


Figure 4 Dynamics 365 Import Solution Button Screenshot

4. Click on **Choose File** then **Next**

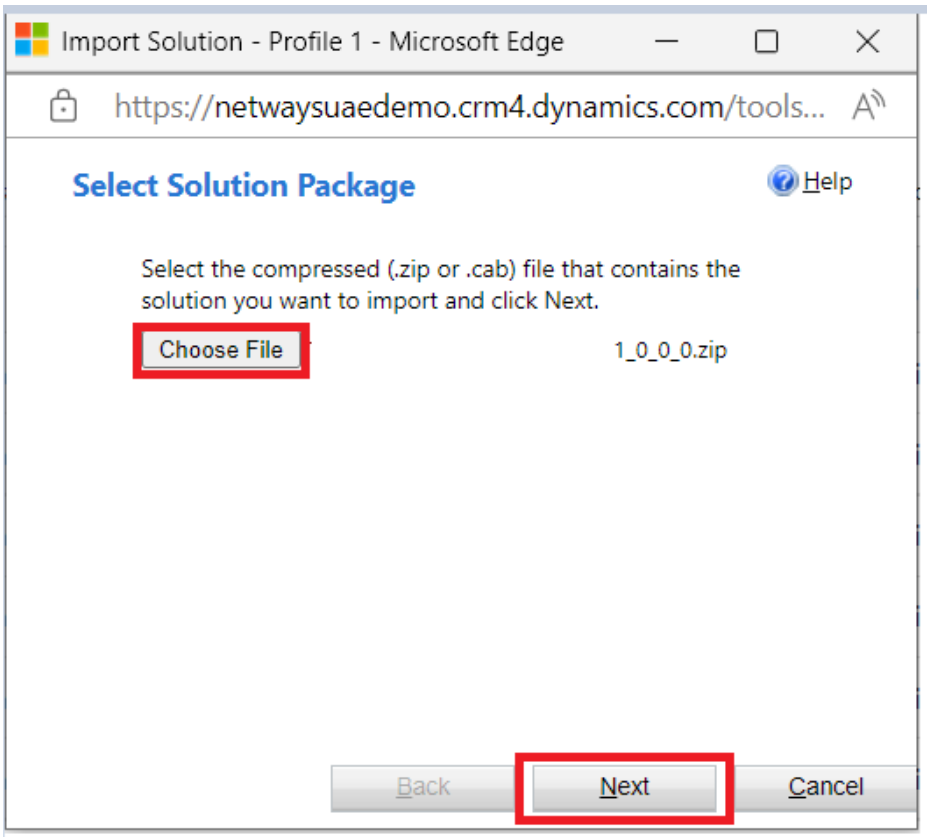


Figure 5 Dynamics 365 Select Solution Package Screenshot

After the Solution is uploaded successfully go to the **configuration part**.

Issues you might face during Installation

Table 1 Issues During Installation

ID	Issue	Solution

How to Uninstall PACE

To uninstall PACE managed solution, you will need to:

1. Remove all dependencies from all entities that have a PACE request Type Configured:
    - Remove the PACE fields from the Form
    - Remove Web Resource added to the form
    - Delete all PACE fields from the entity (schema name starts with **ntwapp_**)
    - Delete SDK messages of assemblies related to PACE
2. Delete the solution.

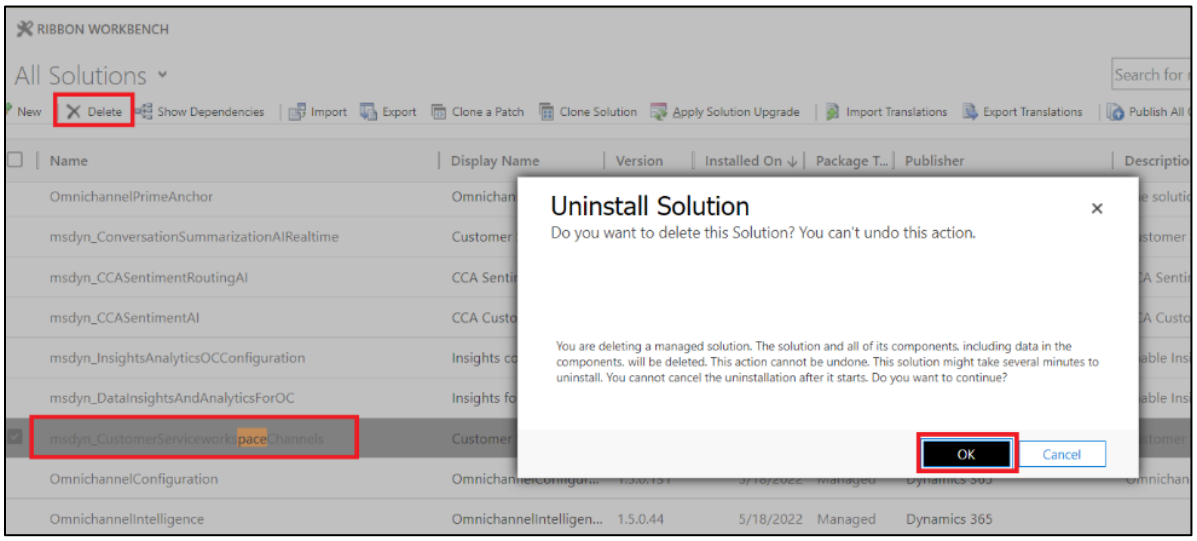


Figure 6 Dynamics 365 Delete Solution Screenshot

Issues you might face during Uninstallation

Table 2 Issues During Uninstallation

#	Issue	Solution

How to Configure PACE Step by Step

The below is one-time configuration steps that need to be done after PACE solution is installed.

1. Add a calendar in the customer service schedule

    1.1. Go to **Settings**

    1.2. Click on **Service Management**

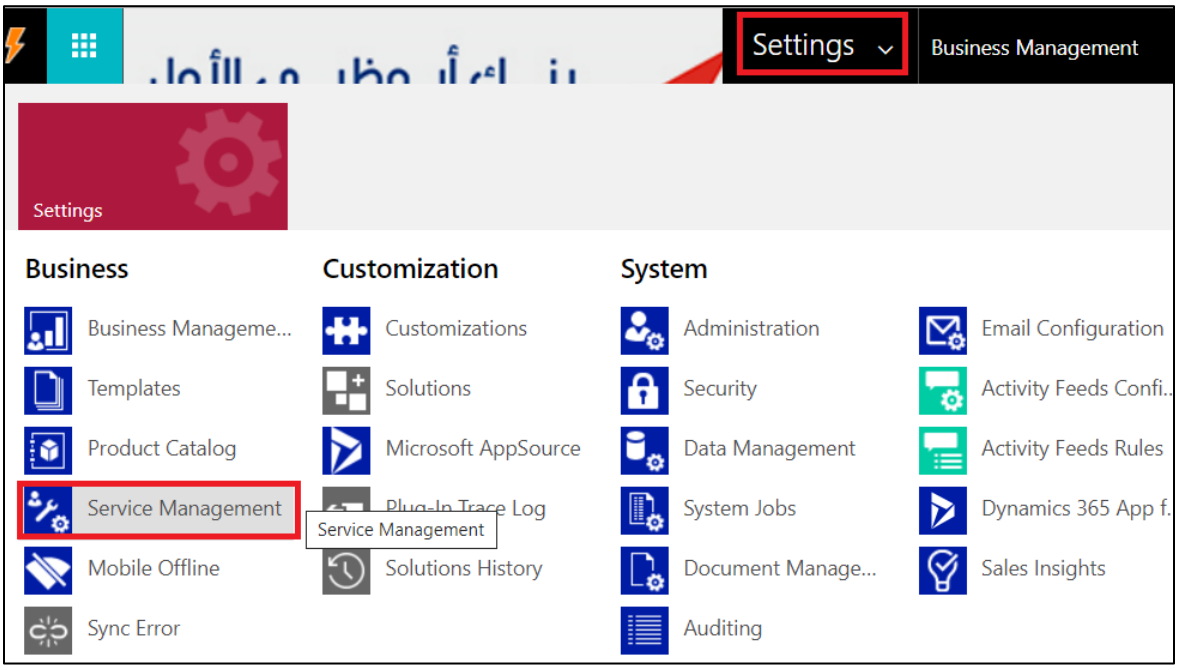


Figure 7 Dynamics 365 Service Management Sceenshot

1.3. Choose **Customer Service Schedule**

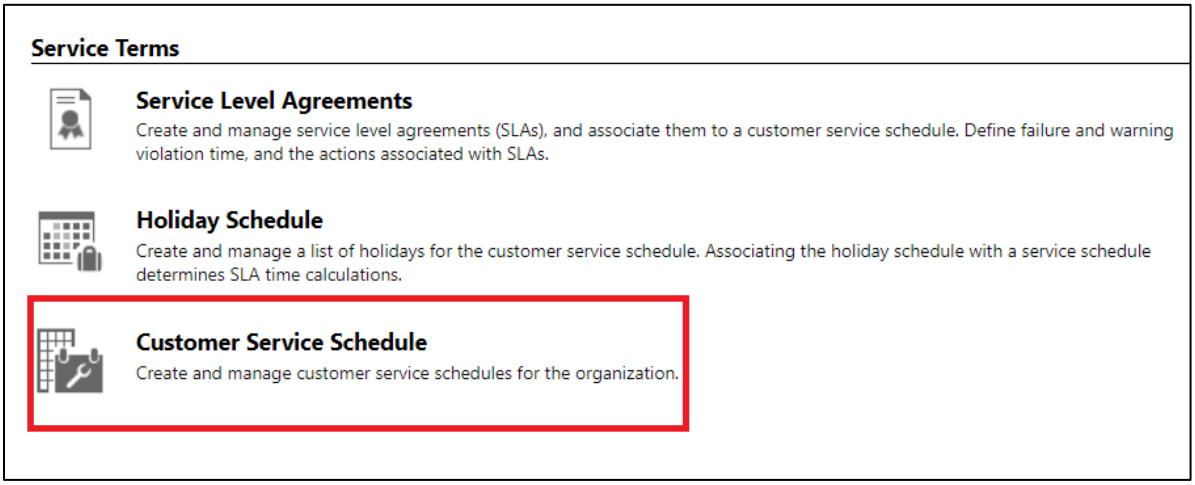


Figure 8 Dynamics 365 Customer Service Schedule Screenshot

1.4. Create new record

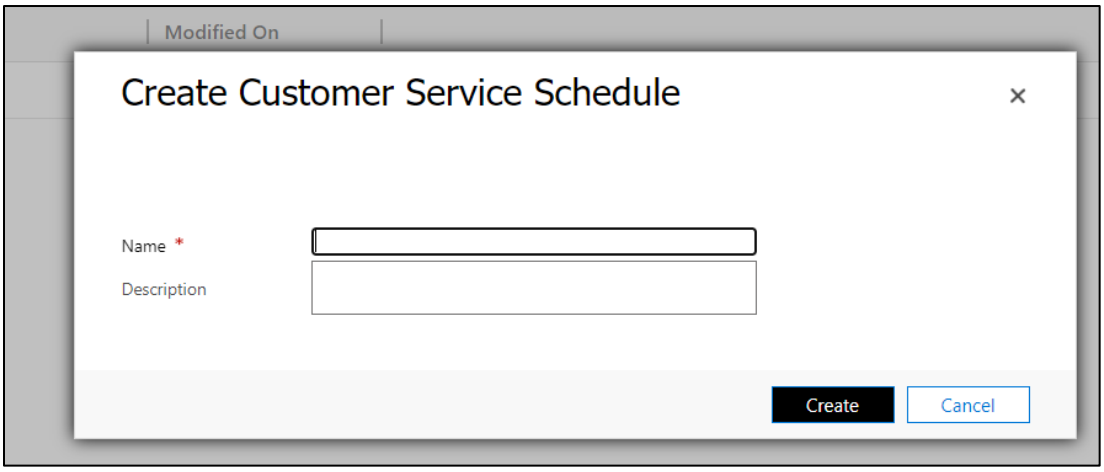


Figure 9 Dynamics 365 Create Customer Service Schedule Screenshot

1.5. Define workdays and hours

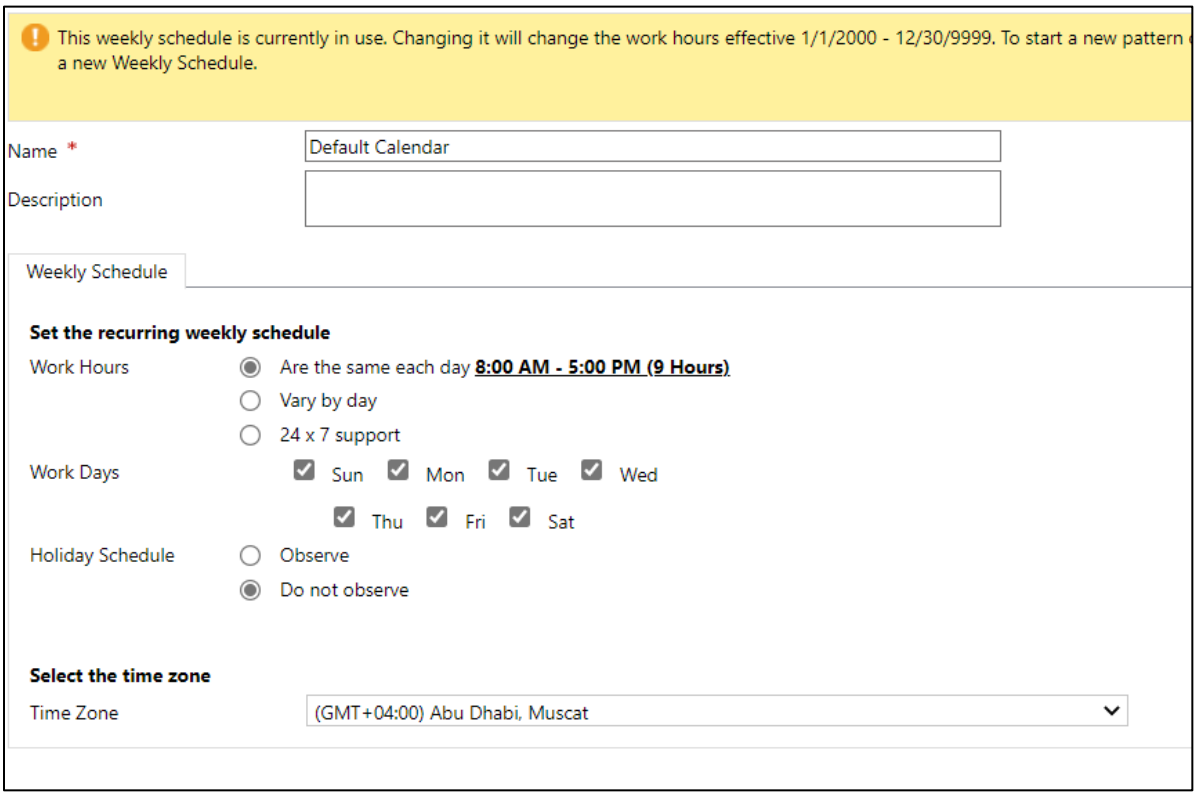


Figure 10 Dynamics 365 Define Work Days and Hours in Customer Service Schedule Screenshot

1.6. Finally, get the ID(GUID) of the created record after saving.

2. In the **APP CONFIGURATION** entity add Key Calendar ID and Value GUID of schedule

2.1. Go to Settings

2.2. Click on **App Configuration** entity

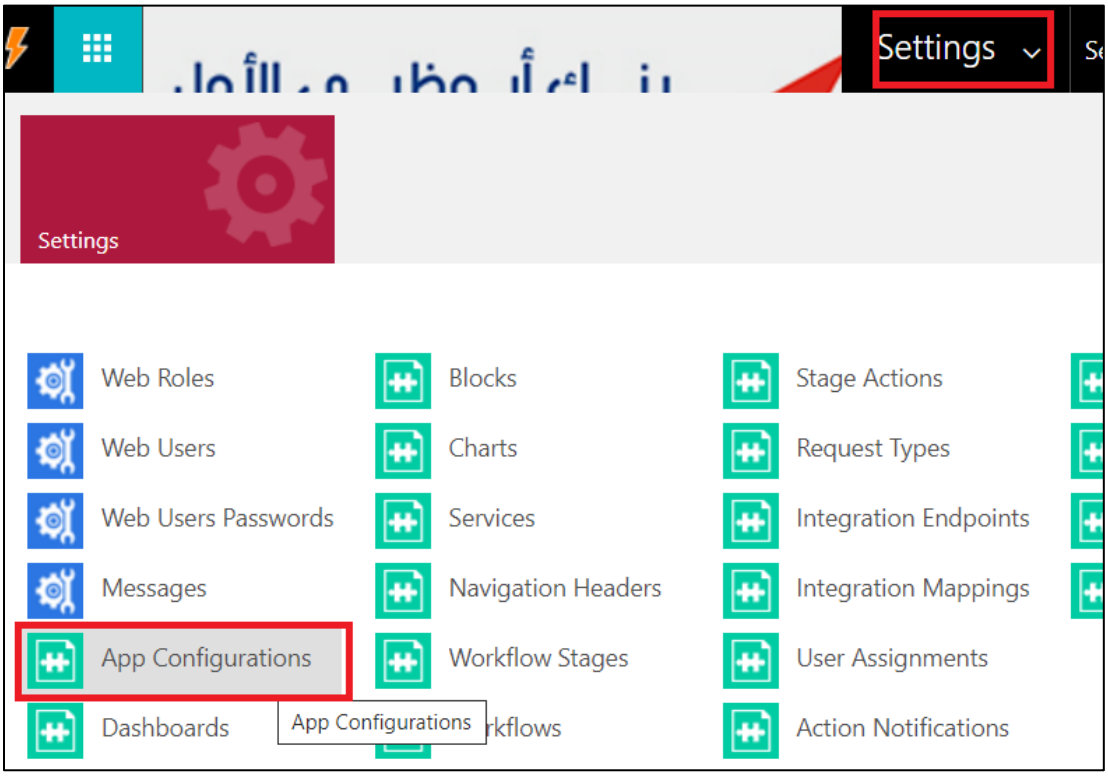


Figure 11 Dynamics 365 App Configuration Screenshot

2.3. Create a new record and fill the below:

- **Key**: Calendarid (type it in the text field)
- **Value**: The GUID of the calendar you created in the previous step

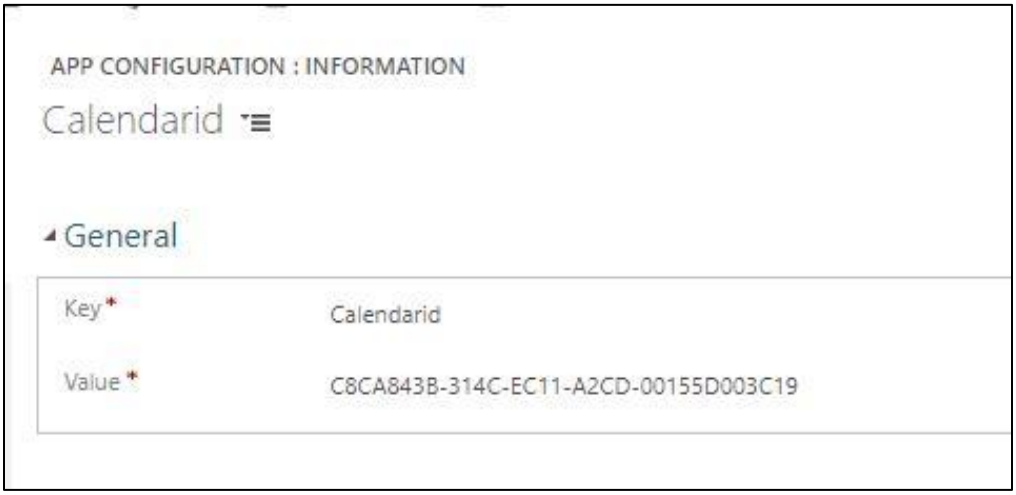


Figure 12 Dynamics 365 App Config Form Screenshot

Issues you might face during Configuration

Table 3 Issues During Configuration

#	Issue	Solution

How to Create a Workflow using PACE

To add PACE to any entity, follow the below steps:

Go to the **Advanced Settings**

1. Create a Request Type Record

This is the beginning point for you to start configuring PACE. It is the most important entity where you define all your services, workflows, and business processes here.

1.1. Click on **Request Types**

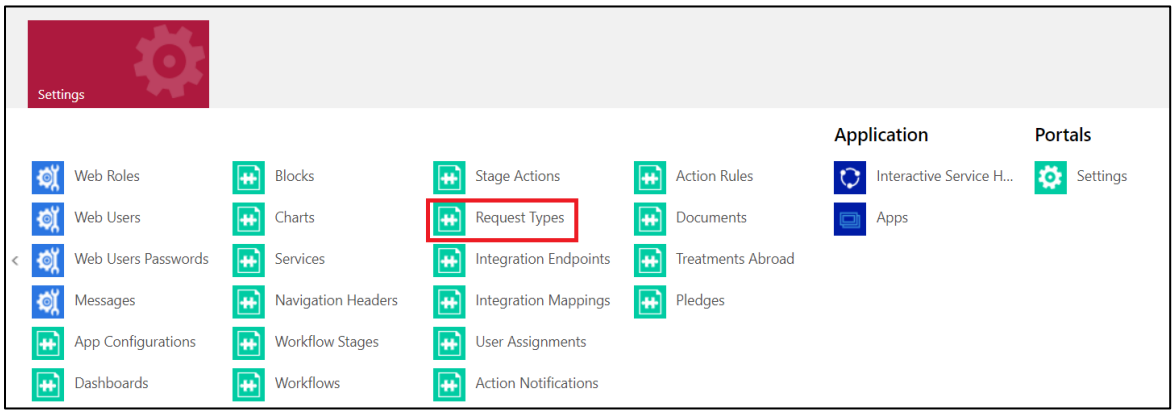


Figure 13 Dynamics 365 Request Type Screenshot

1.2. Click on **New**

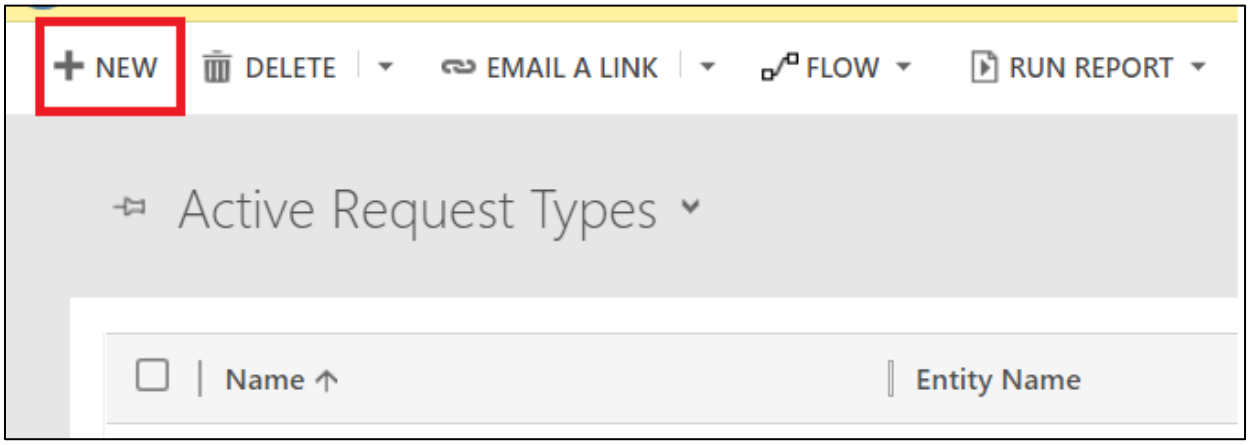


Figure 14 Dynamics 365 Create New Request Type Screenshot

1.3. Fill the below fields on the form:

- **Name**: English Name
- **Request Type Name**: Arabic Name
- **Select Entity Name**: Choose from the dropdown entity you want to add PACE in.
- **Entity Name:** write the schema name of the entity *(In the future this will be hidden)*
- **Primary Field**: auto-filled *(In the future this will be read-only)*
- **Source**: choose from the dropdown (Either **Internal** or **External**)
- **Internal**: Employee (user)
- **External**: Web User outside the company (not a user but a contact with web roles)
    - *ex. A customer submits a request through Portal, so the source of this request is external.*
- **SLA**: it is an integer field used for reporting (Insights, Dashboards). Its range is between 1 and 1000 *(In the future this will support a larger number)*
- **Select Business Process flow**: it is a way to bind the workflow with OOB (Out of the box) BPF *(optional)*
- **Business Process flow**: auto-filled from the Select Business Process flow field *(in the Future this should be read-only/hidden)*

- **Condition field**: It is an optional field if you have only one workflow in your entity. However, if you have two flows, they differ according to a certain condition this should be filled (Create a new field in your entity type dropdown and fill its options
- **Condition Value**: the value of the "**condition field**" you created.

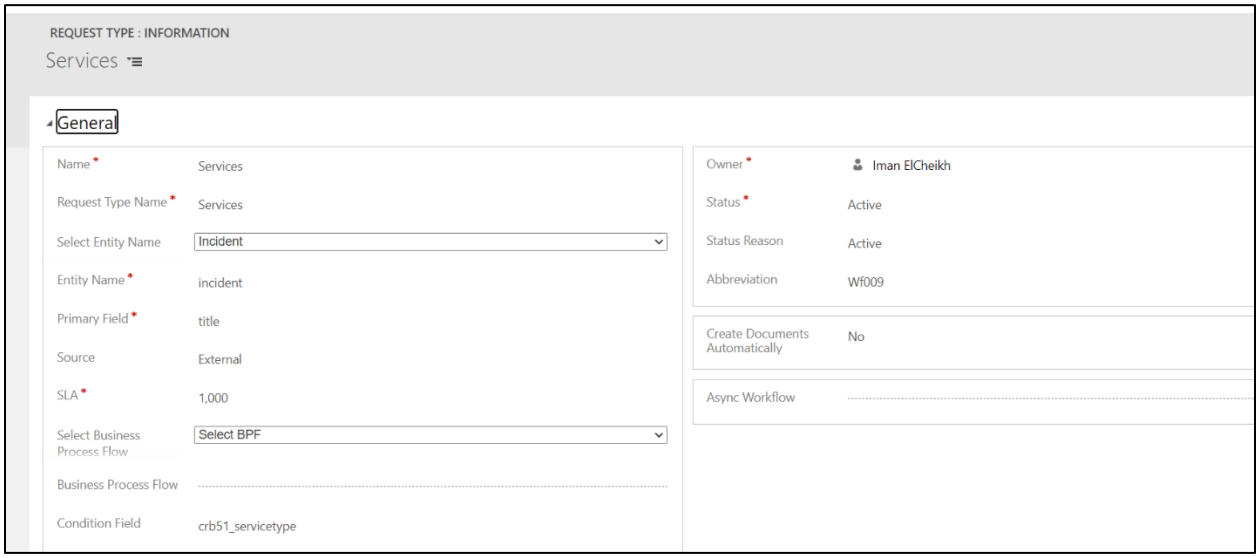


Figure 15 Dynamics 365 Request Type Form Screenshot

To Create the **Condition Field** in your entity if needed, follow the **example** below:

a. Go to **Customization**

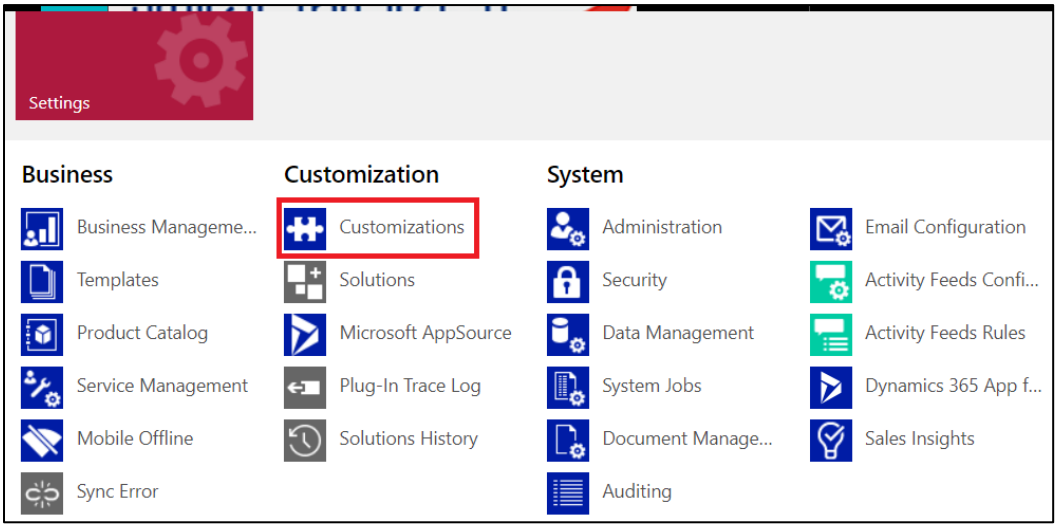


Figure 16 Dynamics 365 Customization Screenshot

b. Click on **Customize the System**

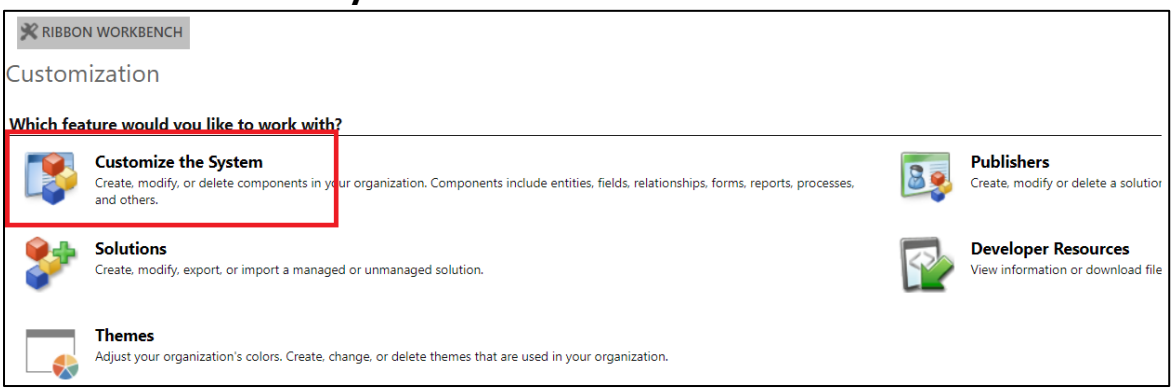


Figure 17 Dynamics 365 Customize the System Screenshot

c. Choose the **Entity** you want to add PACE to it

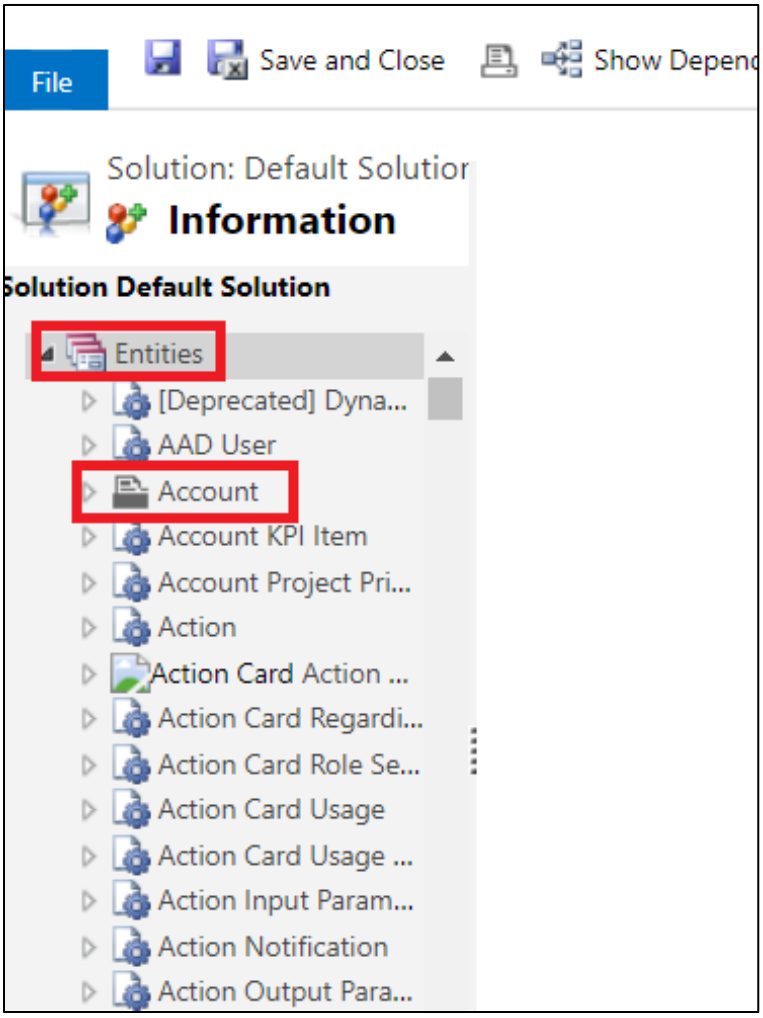


Figure 18 Dynamics 365 Entities Screenshot

d. Open **Fields** Section

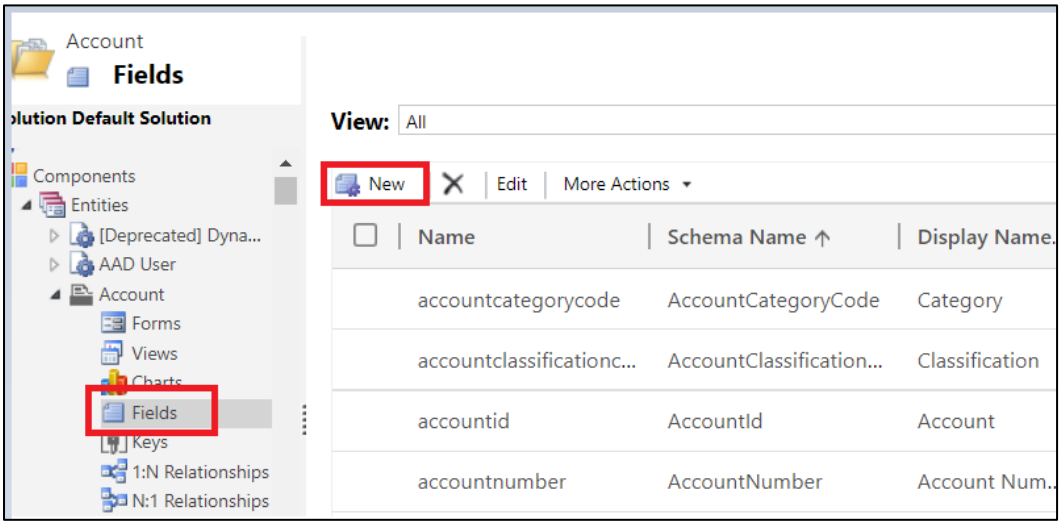


Figure 19 Dynamics 365 Fields View Screenshot

e. Create a new field "Service Type" of type dropdown using the Existing Global Option set: **Service type + Add your Option to it**.

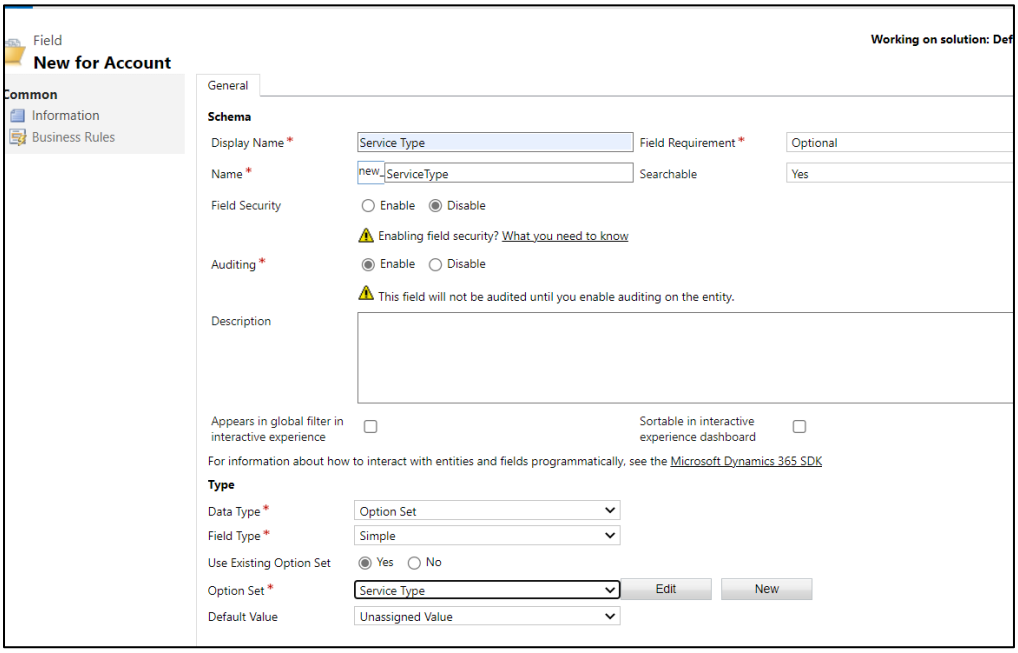


Figure 20 Dynamics 365 Create New Field Screenshot

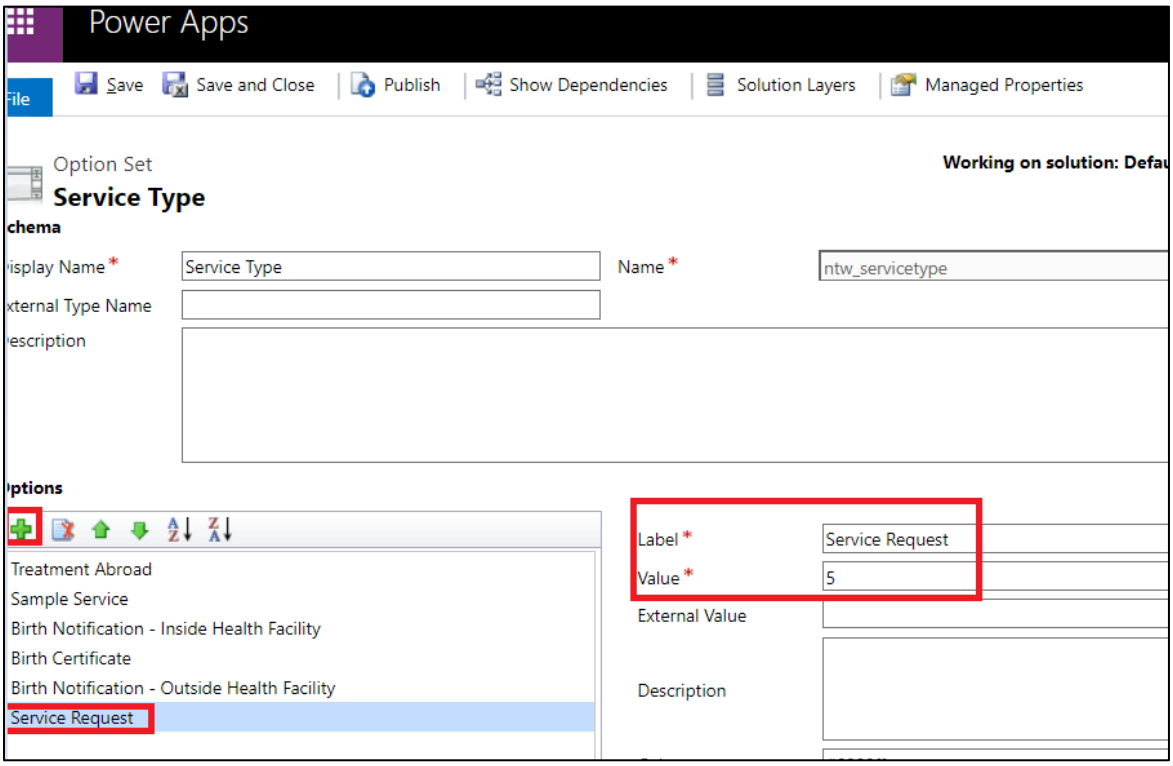


Figure 21 Dynamics 365 Option Set Value Screenshot

Good to Know:

When a **Request Type** Record is created an activity of type **request** is created automatically within the system as well linked regarding the entity we added the activities to it. Request activity view can help you to check all related requests related to this request type.

2. Create a Workflow

Workflow is the version of the request, each workflow has its diagram, and only one published activated workflow will be reflected which is the latest version.

2.1. Go to **Workflows**:

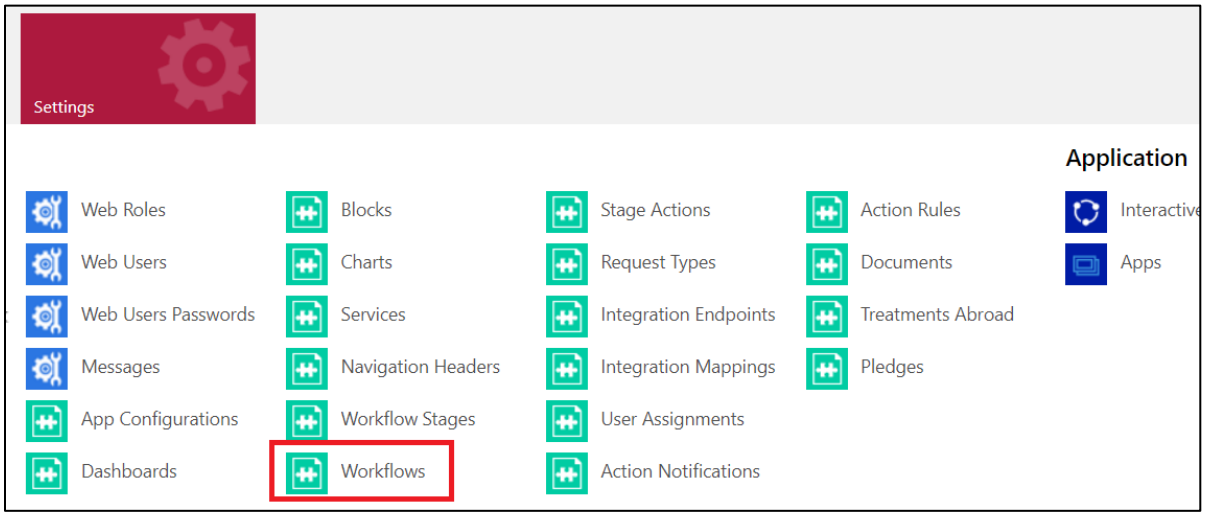


Figure 22 Dynamics 365 Workflow Screenshot

Or from the Request Type Record Sub Grid Workflows

Figure 23 Dynamics 365 Workflow Sub-Grid Screenshot

2.2. Click on new and fill the below fields on the form:

- **Name**: English Name
- **Request Type**: Related Request Type of this workflow
- **Version Number**: it will be filled as the latest version

- **Is Valid**: Yes/No auto-filled (related to the diagram designer. If the logic is valid or not)

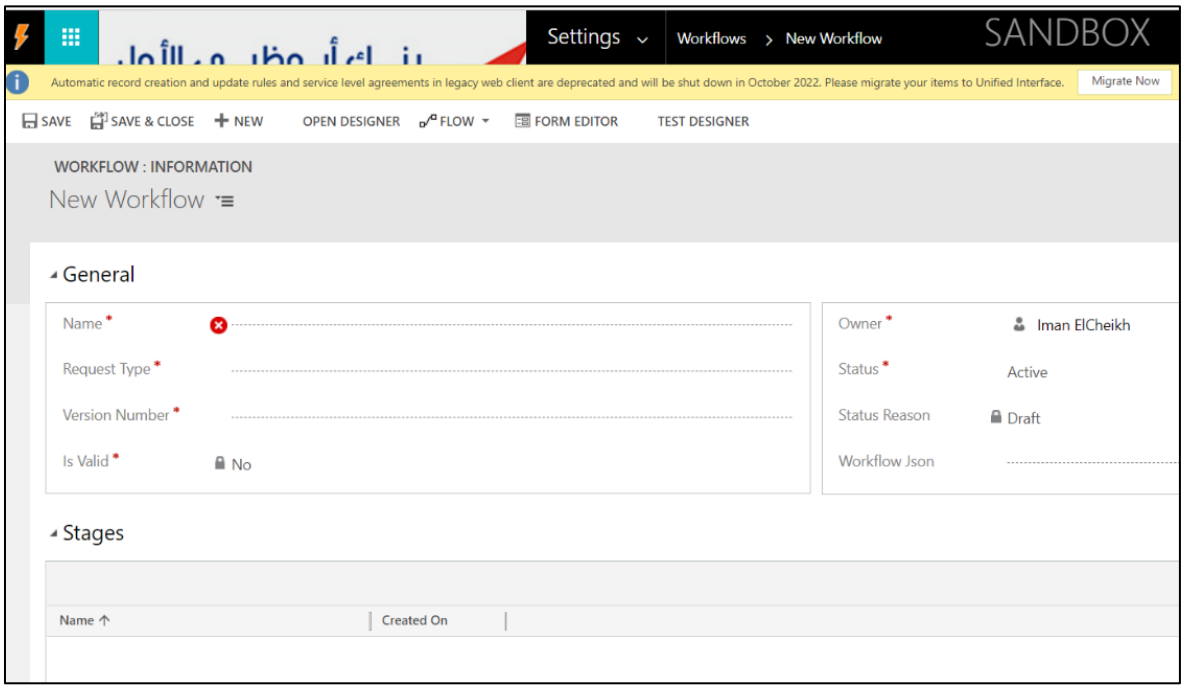


Figure 24 Dynamics 365 Workflow Form Screenshot

2.3. Click on **Open Workflow Designer**

Figure 25 Dynamics 365 Open Desginer Button Screenshot

- **Workflow Designer**

Each designer contains start and end stages.

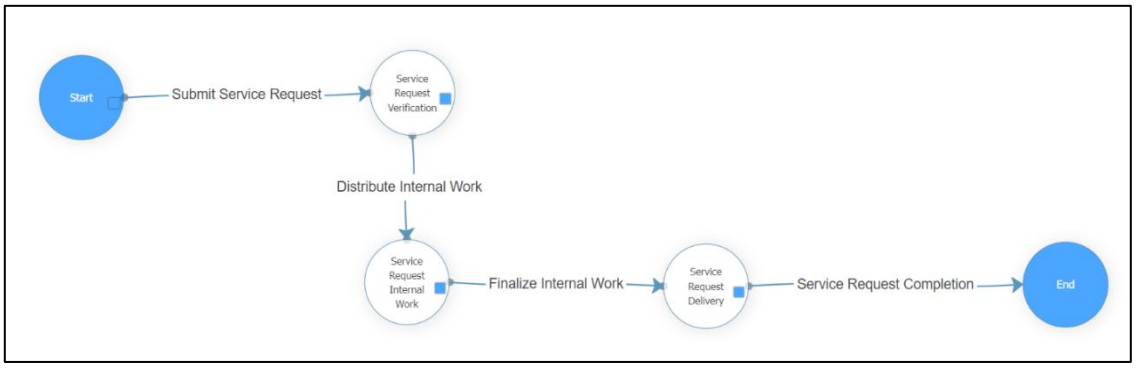


Figure 26 PACE Stages and Actions

Good to Know:

In case you added the stages in the workflow from the sub-grid stages (+) stages will not be reflected in the **Open Designer** button. Open **Test Desginer**

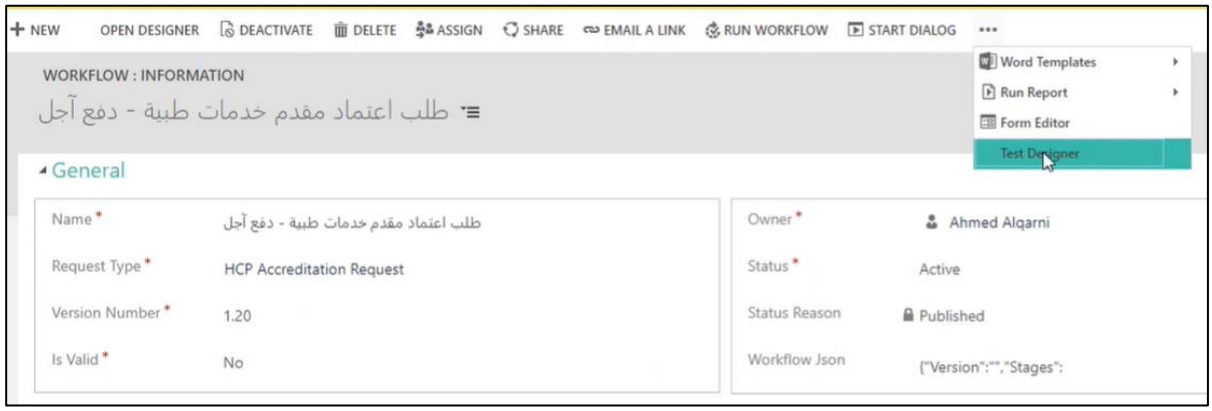


Figure 27 Dynamics 365 Test Desginer Button Screenshot

Good to Know:

You can increase your work effeciencey if you need to do more than workflow by **clonning an existing workflow**.

To Clone a workflow follow the below steps:

1. Go to **Advanced Settings**

2. Go to **Settings** then choose **workflows**

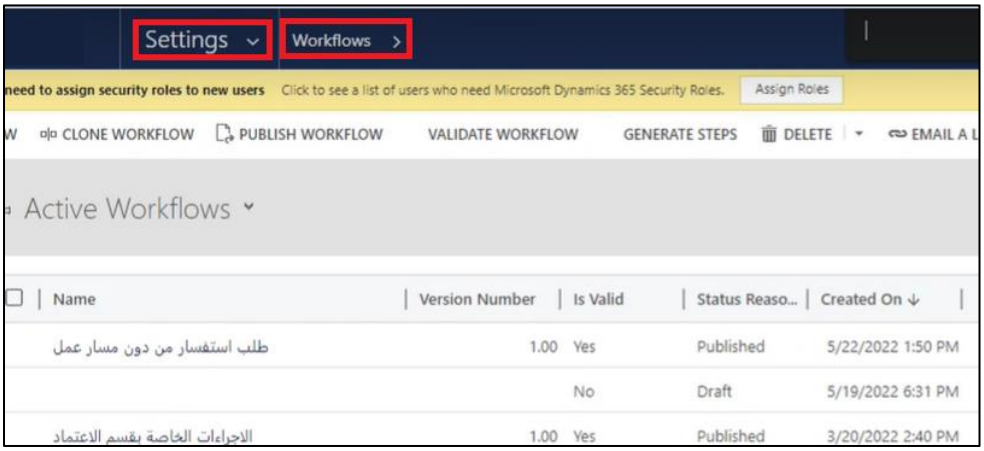


Figure 28 Dynamics 365 Workflow Views Screenshot

3. Click on "**Clone Workflow**"

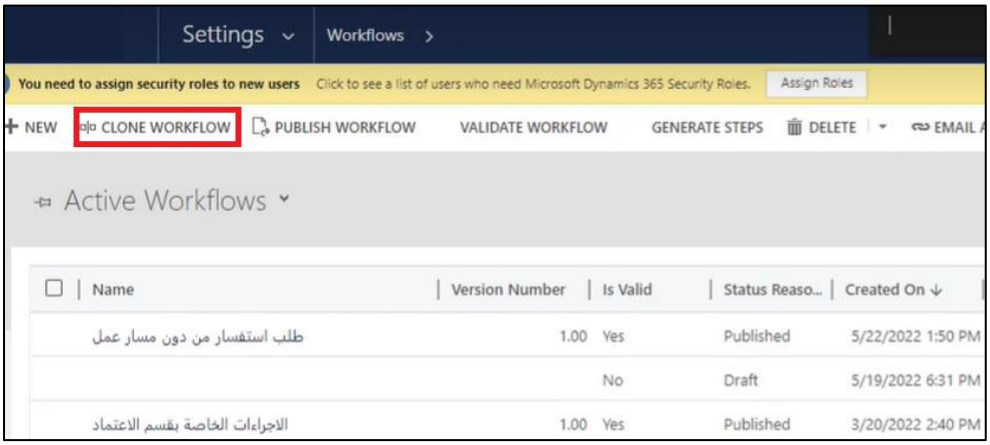


Figure 29 Dynamics 365 Clone Workflow Button Screenshot

3. Create Workflow Stages

3.1 To add new **stages**:

3.1.1. Double click on the designer a bubble will appear red and without name

3.1.2. Right click on the stage and click **Edit.**

Figure 30 Dynamics 365 Stage Form Screenshot

3.1.3. Fill the below fields for each stage:

- **Name**: English Name of the stage
- **Stage Name**: Arabic Name of the stage
- **Workflow ID**: Automatically filled with the related workflow
- **Escalation type:**
  a. **No Escalation**: If the stage is not completed in a period nothing will happen.
  b. **Escalate to Top:** If the stage is not completed by the assigned user, it will be assigned to the manager of the assignee, and it will keep escalating using selected SLA till it reaches the top of the organization chart.

c. **Escalation Matrix:** If the stage is not completed by the assigned user, it will be assigned to the next user or team based on the stage escalation matrix that is defined.

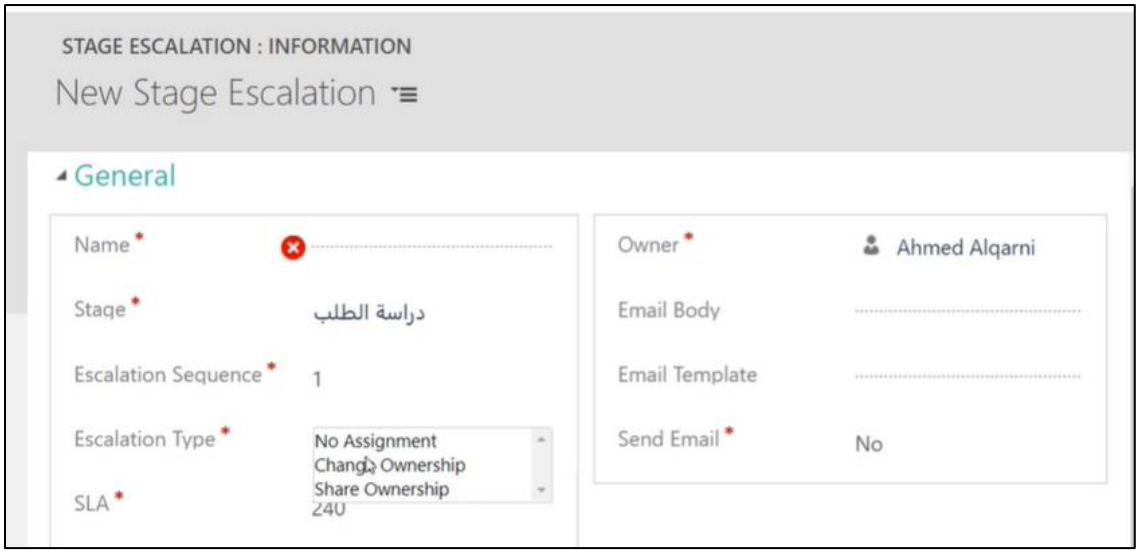


Figure 31 Dynamics 365 Stage Escalation Screenshot

To Create a Stage Escalation, follow the below steps:

a. For each Stage we can add its own SLA and add its specific escalation:

b. Fill Escalation **Name**

c. Fill for which **Stage** you are adding an escalation

d. **Escalation Sequence**: Set the Order of escalations

e. **Escalation type**: choose between (No assignment: not to assign it to anyone, change owner: shift the owner, share ownership: Add more people to work on this request)

f. **SLA**: needed time for this stage to be done and after how many minutes/hours/days to be escalated.

g. **Assignment Type:** Internal /External

h. **Assign it to External:** (Appears if Assignment Type = External)

i. **Assign it to Internal:** Choose one option from the dropdown list.

j. **Stage sequence**: The order in which the stages should move.

k. **Stage Type**: Read-only field representing the type of stage.

- The First stage is of type *Start*
- The Last stage is of type *End*
- All the stages between the start and end are of the type *Intermediate*

l. **Reset Stage Validation**: Boolean field to define whether validation of this stage should be done based on specific fields and business rules on the form

m. **Reset Token**: Boolean field to define if the document-related token is to be reset on this stage or not.

n. **Code**: a read-only field auto-filled with a code for the stage

o. **SLA**: represents the period for the stage to be completed.

p. **Calendar ID**: Use the Template you created while configuring PACE

q. **Workflow Step**: The workflow step is used to define how the stage is to be displayed in the stepper. A workflow step record can be created and linked to each stage and will require the info below:

1. **Arabic Name:** Name in Arabic
2. **English Name**: Name in English
3. **Sequence**: display sequence in the stepper

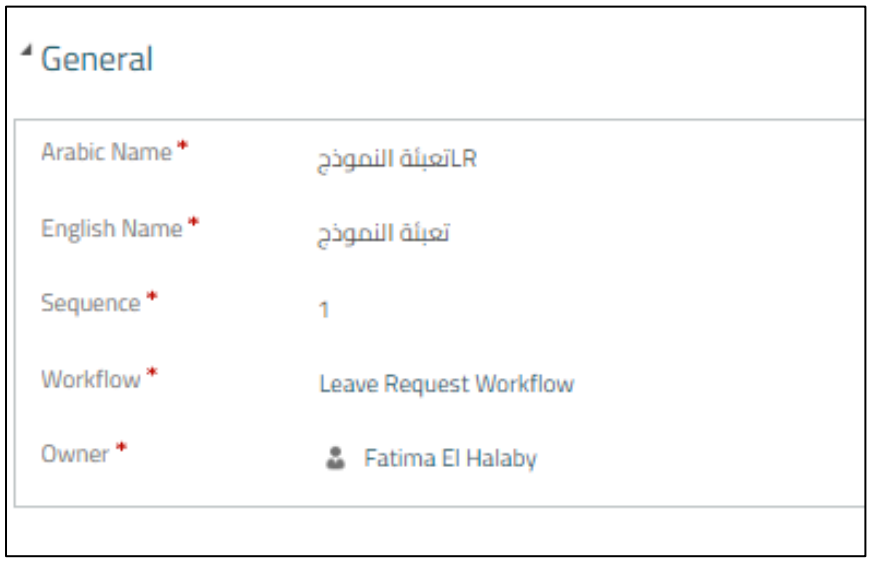


Figure 32 Dynamics 365 PACE Action Screenshot

4. **BPF Stage**: this will be used to link the workflow designer stage to the OOB BPF stage.
5. **Allow Upload**: You can use it to lock any update on the record or any attachment upload from outside during the current stage.
6. **Top**: Position of the Stage node (Read Only)
7. **Left**: Position of the Stage node (Read Only)

4. Create Workflow Stage Actions

4.1. After adding all the needed stages, actions on each stage must be defined

- Each stage could have one or more action
- To add a new action user must click on the anker of the stage and link it to the next stage
- The action will appear red and without the name
- Right-click on the action and click edit

- Fill in the required information for each action:

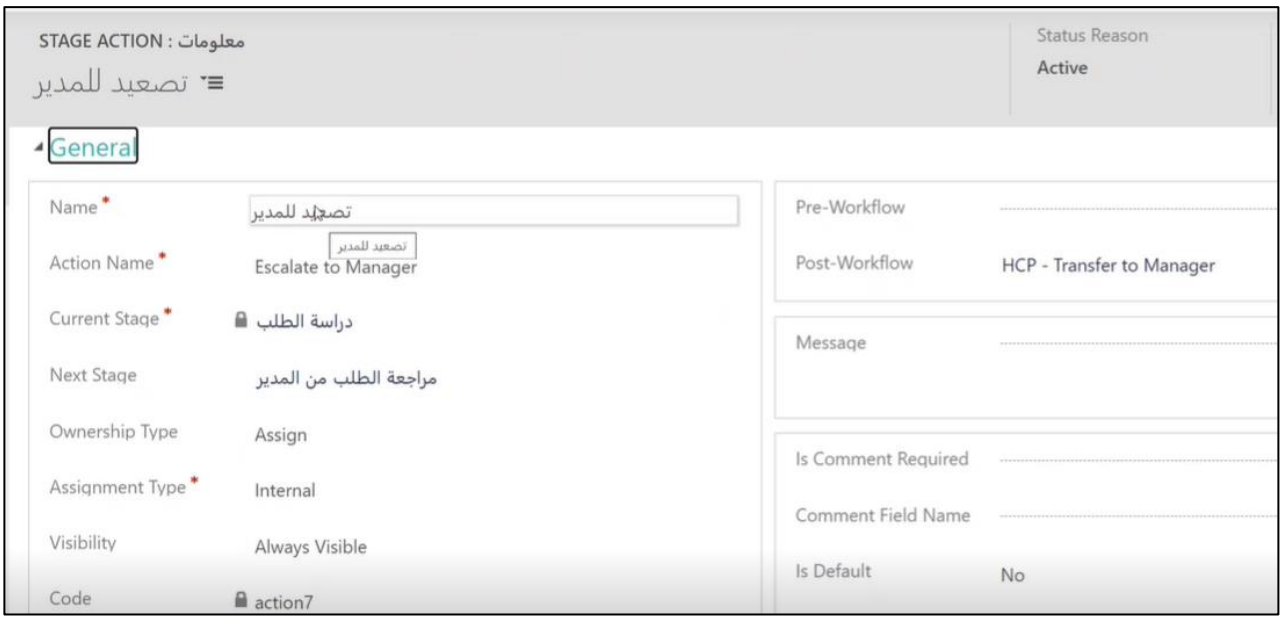


Figure 33 Dynamics 365 PACE Stage Action Form Screenshot

a. **Name**: Arabic Name of the action

b. **Action Name**: English Name of the action

c. **Current Stage** and **Next Stage** will come by default when linking the 2 stages.

d. **Ownership Type:**

1. **Assign**: when assigning the action, the record will be assigned to the assignee, and there is a previous owner, he will lose the all control over this record.
2. **Assign and keep records with previous owner:** when assigning the record and keep with the previous owner, the record will be assigned to the new owner, but both the new and previous owner will have the privilege to read and update it.

e. **Assignment Type:** (Internal, External, Events)

1. **Internal**: Inside the company a system user and have a security role.
2. **External**: Outside the Company, not a system user and doesn't have a security role only a web user.
3. **Event**: It is hidden, and will not be visible in the actions it's more like a decision.

If the assignment type is **Internal**: the record can be assigned to

1. **Queue**: the record will be assigned to a queue, and one user in the queue will pick the record and take the needed actions.
2. **Team-All Users**: the record will be assigned to all the users in the team and any user can take the action
3. **Team-Random User:** the record will be assigned to a random user in the team
4. **Team-Selected User (Least Utilized):** the record will be assigned to the user who has the minimum number of tasks (Available User).
5. **Single Team Voting**: the record will be assigned to all users in the team

   ***Approval percentage should be between 51-100, this Action should be after the stage we will take its decision in this action*.

6. **Multiple Team**: the record will be assigned to multiple teams and the users in each team should take the actions on the record in parallel.

   ** Add the teams you wish to work with in parallel with 100 percent.*

7. **Specific User**: the record will be assigned to a specific user in the system
8. **Field in Current Record:** the record will be assigned to a field on the record it could be an account or a contact
9. **Custom Assignment**: the record will be assigned based on a custom activity.
10. **Direct Manager**: the record will be assigned to the manager of the owner of the record.

If the assignment type is **External**: the record can be assigned to the

1. **Requester**: the user who submitted the request from the portal
2. **Specific Role in Requester's Facility**:
3. **Specific User**: the record will be assigned to a specific user in the system
4. **Field in Current Record**: the record will be assigned to a field on the record it could be an account or a contact

f. **Visibility**

1. **Always visible**: the action will be always visible
2. **Custom visible**: the action will be hidden and shown based on the custom action

* *(OOB Actions , fill it as the unique name of the Action).*

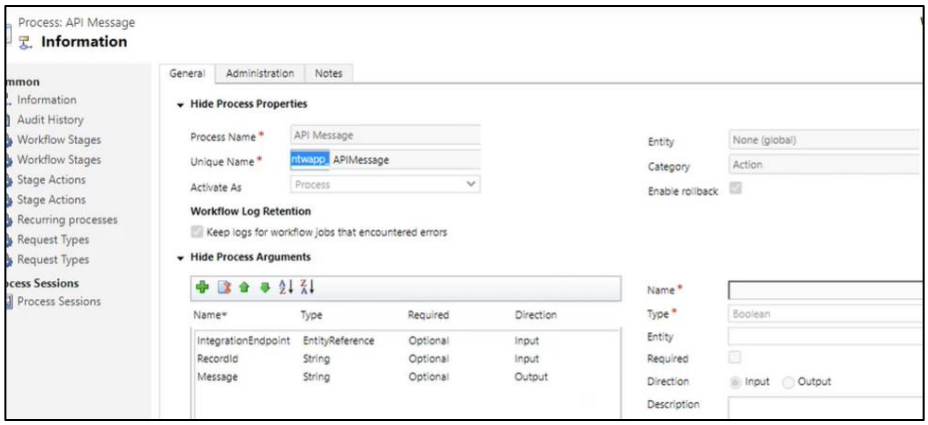
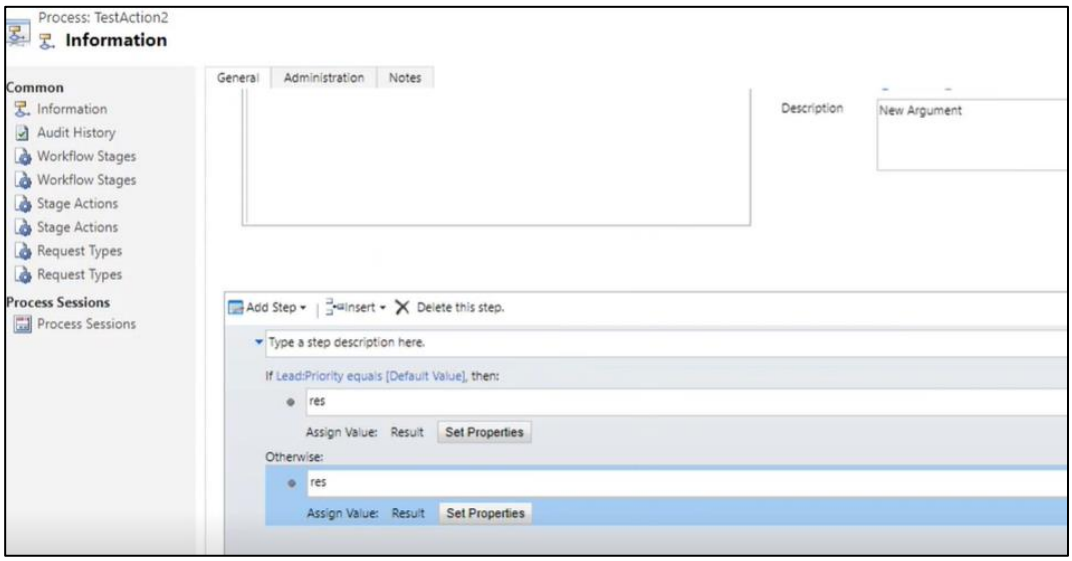


Figure 34 Dynamics 365 OOB Action Screenshot



g. **Assignment custom action** in case the user selected "custom visible" according to OOB Actions

h. **Action Notification:** Automate send emails. SMS or push notifications when the action is about to start/End.

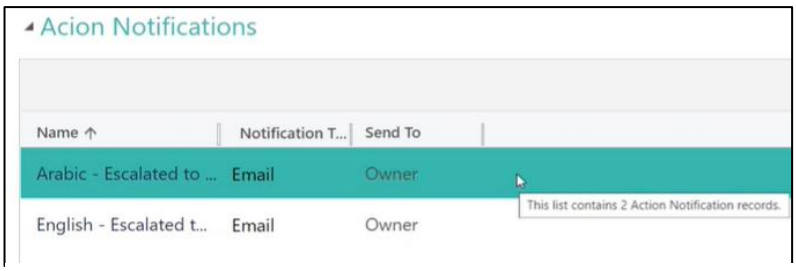


Figure 35 Dynamics 365Action Notification Screenshot

1. **Sequence:** If the stage has multiple actions user can define the sequence of the actions
2. **Pre-Workflow**: An action can have a pre workflow (a CRM process of type workflow) that can be triggered before this action.
3. **Post-Workflow**: An action can have a pre workflow (a CRM process of type workflow) that can be triggered once the action is completed.
4. **Message**: The message that will be displayed on the form in case of validation issue
5. **Is Comment Required:** define if the comment is required while completing the action.
6. **Comment field name:** the schema name of the field where the comment needs to be saved.
7. **Is Default:** define if this action is the default action in the stage
8. **Required field list:** the list of all required fields before completing this action
9. **Action Rules**: inside the sub grid, action rules records can be created to define how the action should be assigned:

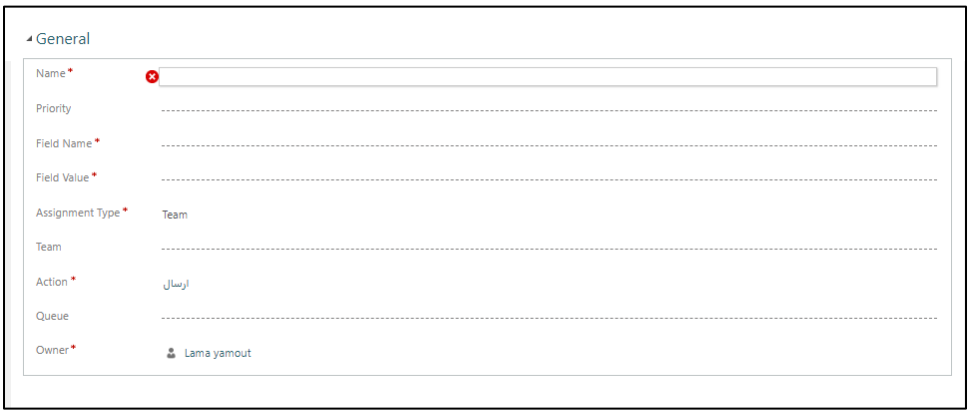


Figure 36 Dynamics 365 Action Notification Form Screenshot

a. **Name**: Name of the rule
b. Priority: critical/ normal
c. **Field Name**: schema name of the field we are defining in the rule
d. **Field value**: the value of the field for this rule to be valid
e. Assignment type: how the record will be assigned (team, queue...)
f. **Team**: in case the assignment type is team, the name of the team

5. After defining all the stages and actions, from the workflow designer window the user must save the flow, then after saving he should validate the workflow so the workflow can be published, and finally the user must publish the workflow to move it from the draft state to published state.

5. Add Pace Stepper to the Form **(Optional)**

Pace steppers show the progress of the workflow it is similar to Dynamics CRM Business Process Flows, but it is read only and it doesn't allow the user the move next and previous or change the predefined diagram.

5.1. Go to **Customizations**

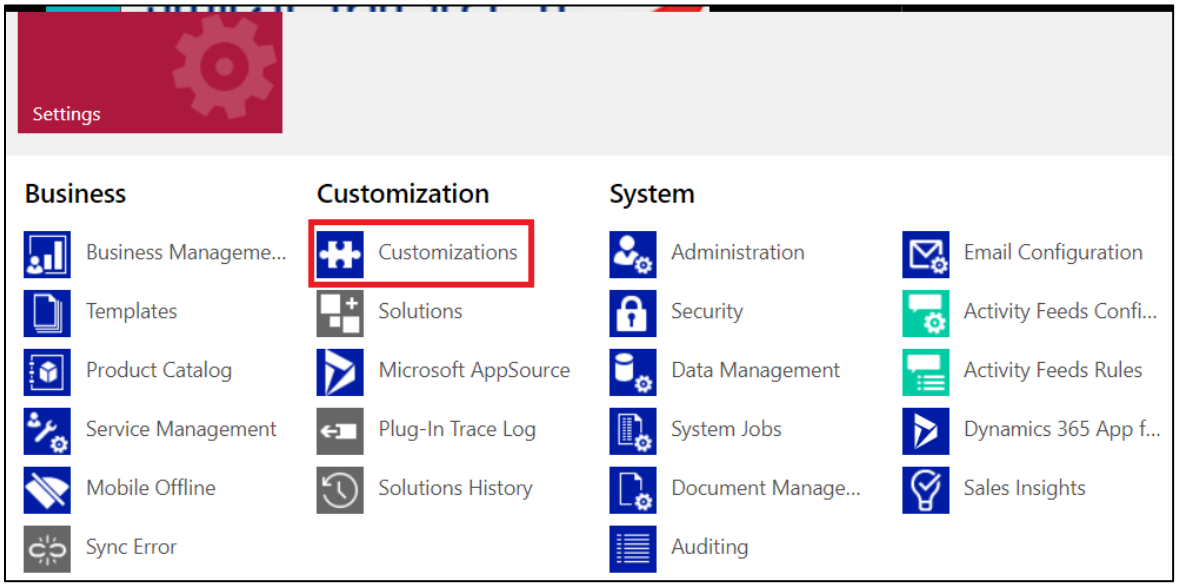


Figure 37 Dynamics 365 Customizations Screenshot

5.2. Click on **Customize the System**

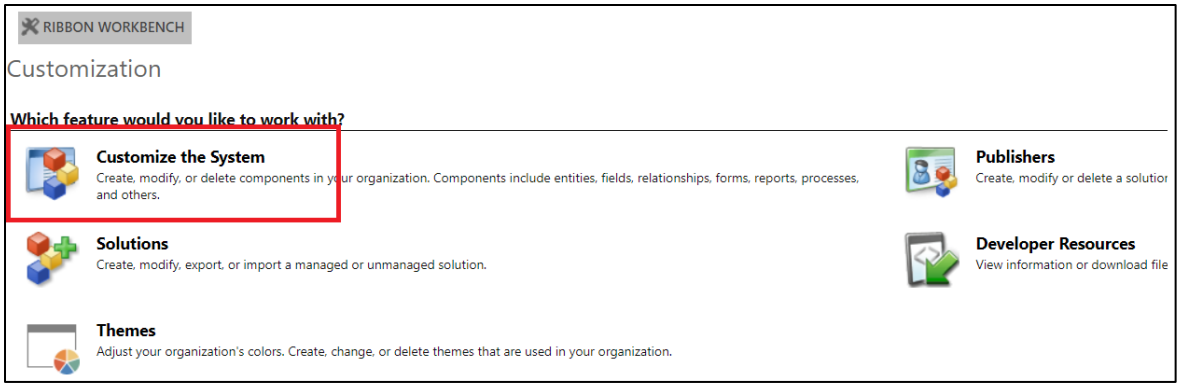


Figure 38 Dynamics 365 Customize the System Screenshot

5.3. Choose **Entity** and enable its activities

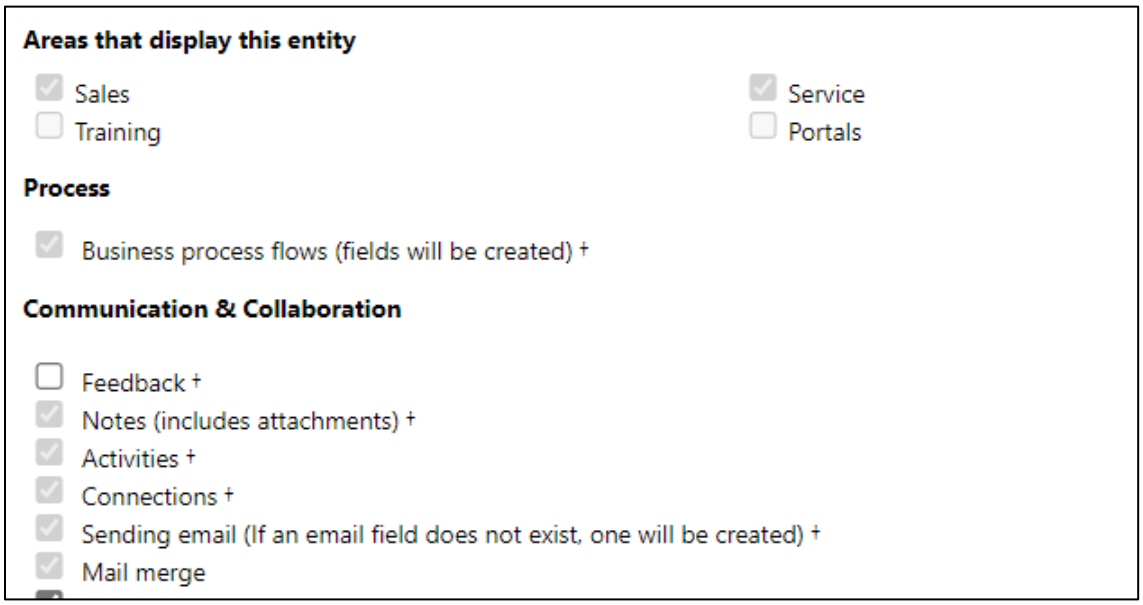


Figure 39 Dynamics 365 Activate Activities Screenshot

5.4. and go to **Forms**: Open the Form

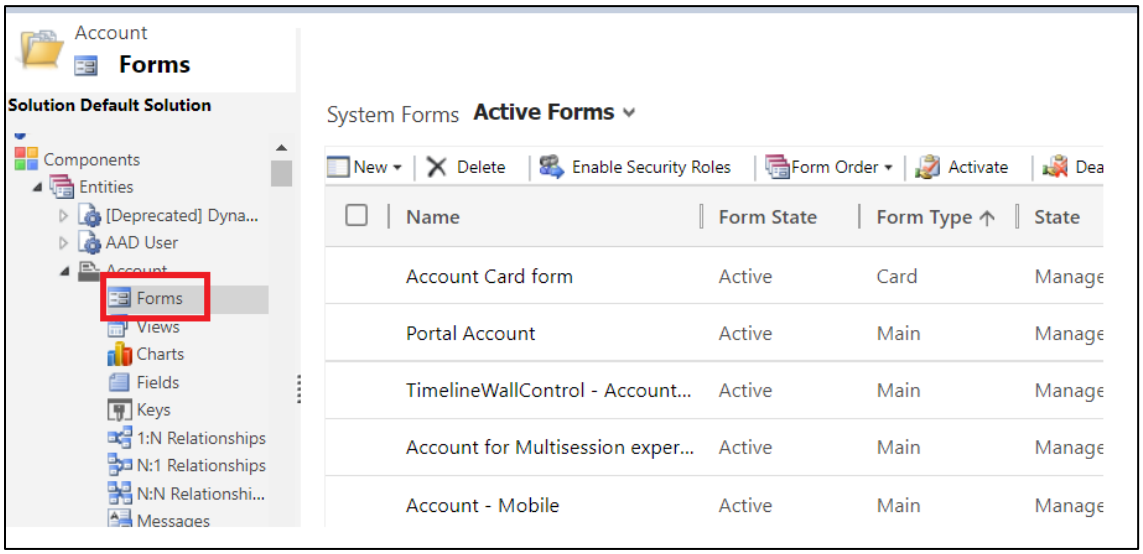


Figure 40 Dynamics 365 Forms Screenshot

5.5. Add a new **Section** in the form

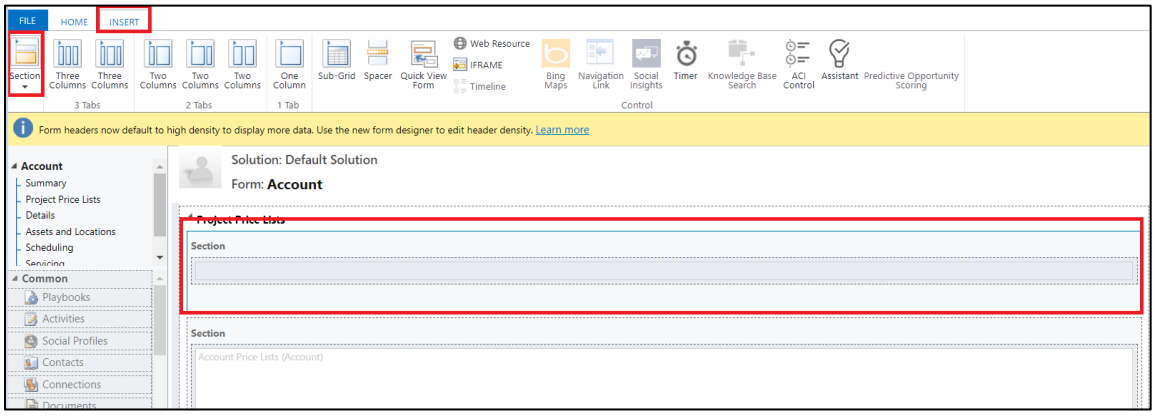


Figure 41 Dynamics 365 Form Designer Screenshot

5.6. Insert **Web Resource** and choose: **ntwapp_stepper_index.html**

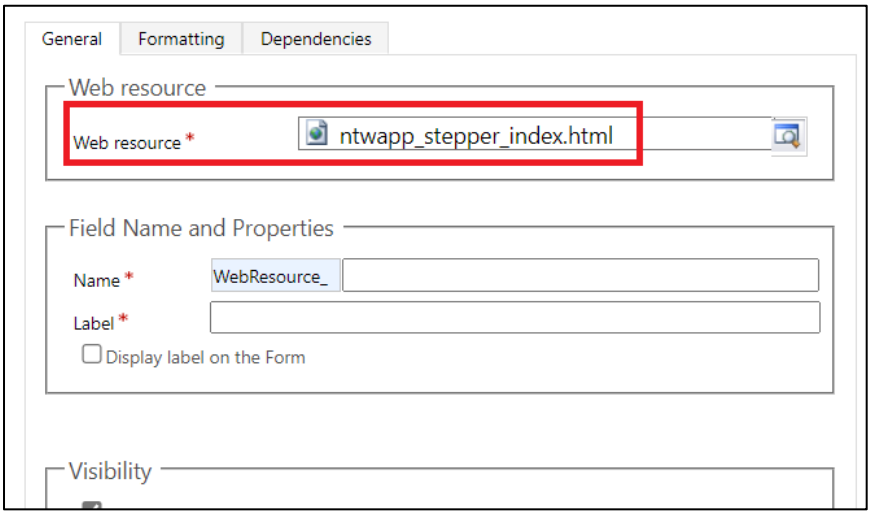


Figure 42 Dynamics 365 Web Resource Screenshot

5.7. **Save** then **Publish**

After Publish, a typical PACE Stepper should like like the below Widget

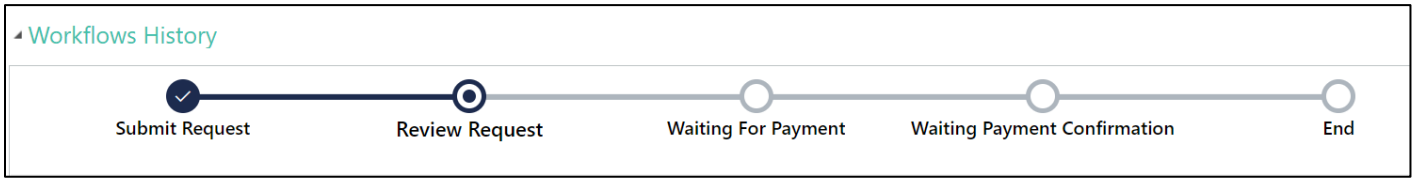


Figure 43 PACE Stepper

6. Add PACE Fields to Form

6.1. When we define a new request type on a specific entity, Pace creates couple of fields on behalf of you in that selected entity, we should add them to the form in a hidden section . the Fields are as follows. Note that the submit Button in the ribbon will not show unless you add the fields below.

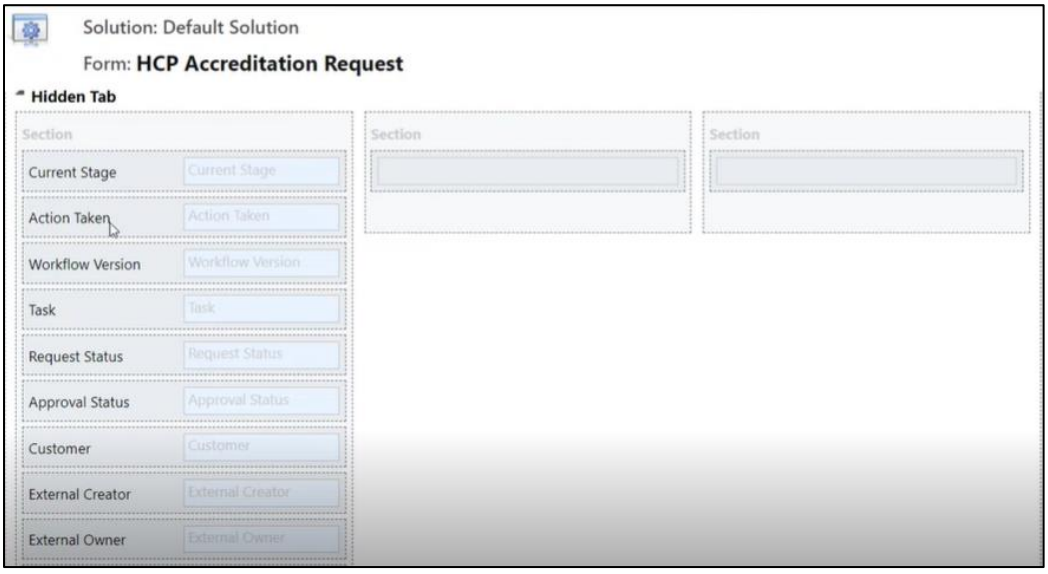


Figure 44: Adding PACE Autocreated fields to the Form

- **Current Stage**: Last Stage the flow reached.

- **Action Taken**: Last decision took in the last stage.
- **Workflow Version**: on create of the record workflow version will be filled automatically with the latest version number.
- **Task**: Last task done for this record.
- **Request Status**: Closed, Pending
- **Approval Status**: Rejected, Approved, Cancelled
- **Customer**: who is the customer who submitted the request (Account, Contact).
- **External Creator**: The one who created the record – Current **Contact** of the record
- **External Owner**: System User
- **Is Stage Valid**: true/false
- **Token**: Authorize the current user who is assigned to the record. It gives the write permission for the record.
- **Web Role**: External Roles to filter which contact outside the company have the access to read and write this record
- **Previous Owner**: Last owner
- **Allow Upload**: Close or open the record according to the conditions.
- **Status**: State-Code
- **Sequence**: Number of stages
- **Last Comment:** Read-only field that stores the last comment added on submit button

Add Tasks History Sub-Grid to the form (Optional)

We need to add Task Sub-grid to track tasks created history related to the Request Type and to check it status.

6.2. Go to Insert

6.3. Click on Sub-Grid

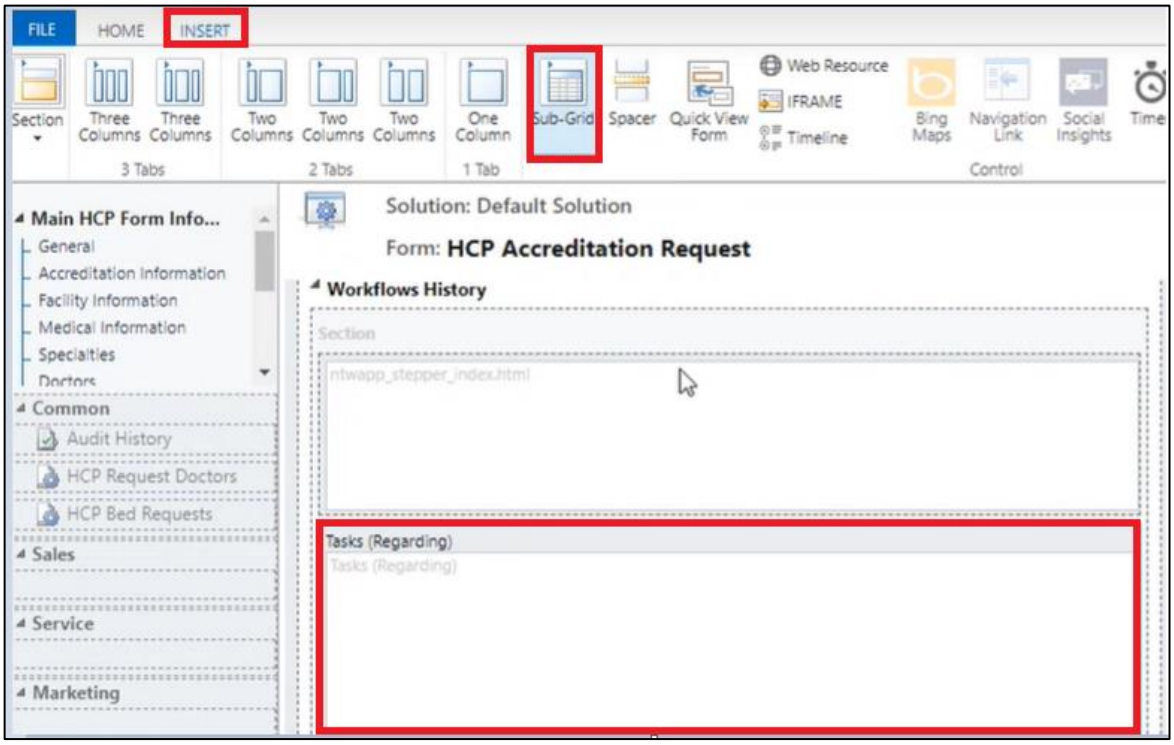


Figure 45 Dynamics 365 Form Screenshot

6.4. Choose the below in the Pop-up up for the sub-grid

6.4.1. **Records**: Only Related Records

6.4.2. **Entity: Tasks**

6.4.3. **Default View**: Workflow History

6.4.4. **Name**: choose a name for this sub-grid

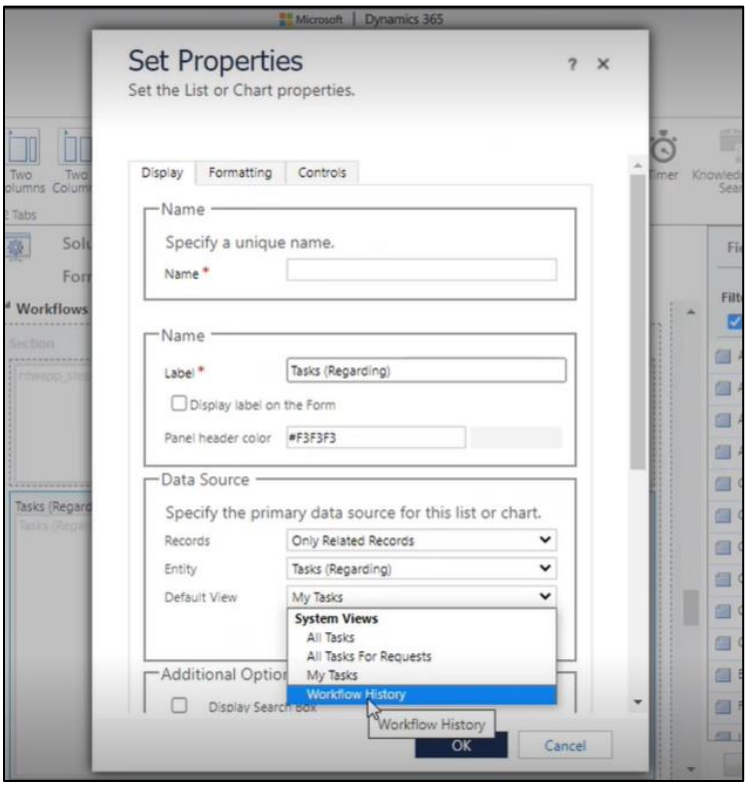


Figure 46 Dynamics 365 Sub-Grid Properties Screenshot

Figure 47 Dynamics 365 Tasks Sub Grid Screenshot

Good to Know:

The visibility of the submit button is based on the owner of the record and security roles granted for the users.

ex. if I was the owner, or any team I'm included in it submit button will show otherwise it will be hidden.

Figure 48 Submit Button Screenshot

7. Document Management

Dynamics CRM by default stores all documents as annotations or activities related to the current record. In an E-Service platform, each service often requires a set of predefined documents. In that case using out of the box annotations or SharePoint integration will add bull of document in the current record but the decision maker cannot distinguish which file is related to which required document. For the best user experience a required Tax certificate for instance should show in the same Tax section on the form. Otherwise, the decision maker should keep scrolling up and down to compare each section with its required documents in the attached documents area. On the other hand, a customer might ask to integrate with his own DMS rather than OOB storage.

PACE abstracts the documents storage for you by providing a standard APIs along with it is Source which you can use to integrate with any DMS. It allows you to create a file for each attachment in the form then it renders a download/Upload button on that file that previews or downloads that document or even uploads a new document. PACE does not save the file binary content on the record, it uploads it to the selected DMS, then saves only it reference in the record.

To Configure DMS

7.1. Go to **Advanced Find**

7.2. Search for **Attachment Configuration**

Attachment Configuration is an entity in that I configure the required attachments for the entity. It is an integration with document management system "ex. **SharePoint**"

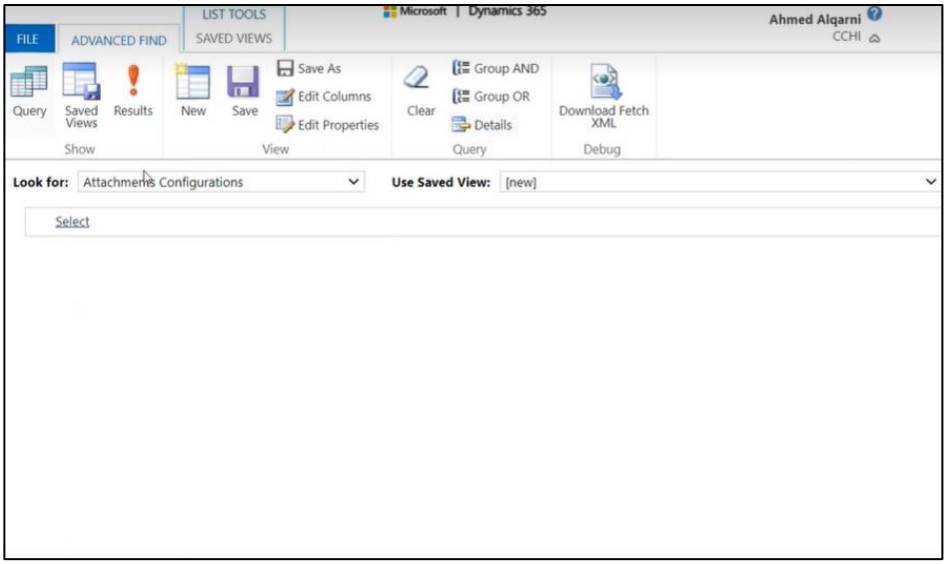


Figure 49 Advanced Find Screenshot

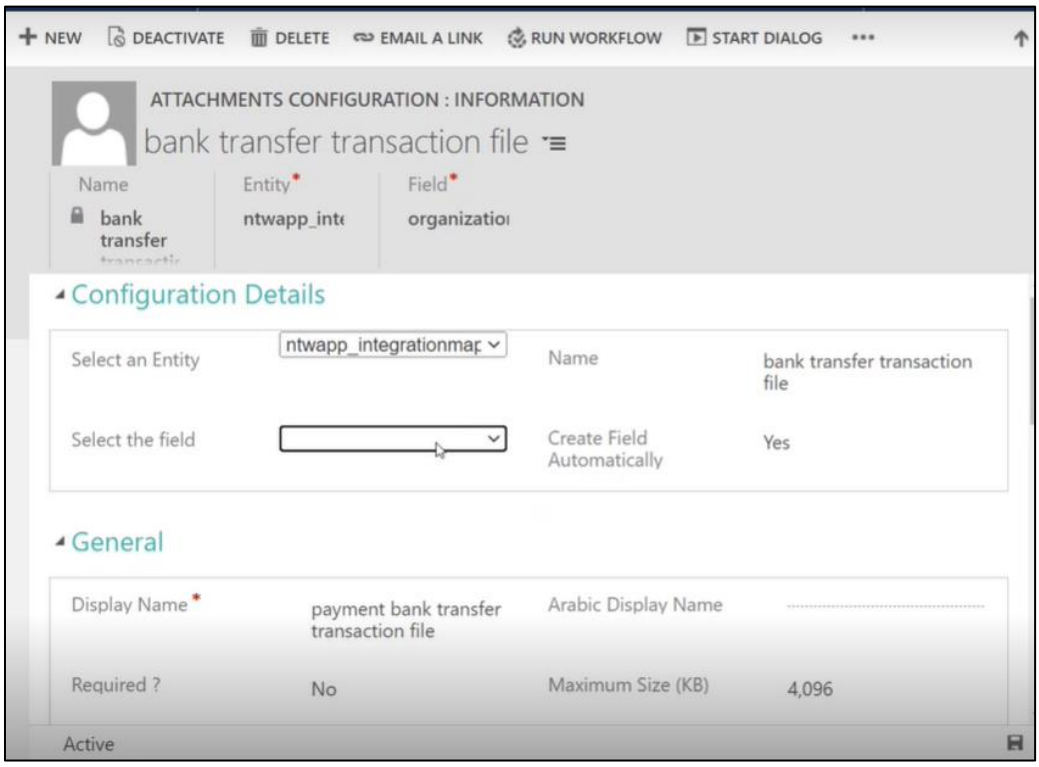


Figure 50 Attachment Configuration Form

Go to the form, add a web resource so we can add the documents related to the record, and add the parameter of this web resource the schema name of the field that we will store the URL/ID in it.

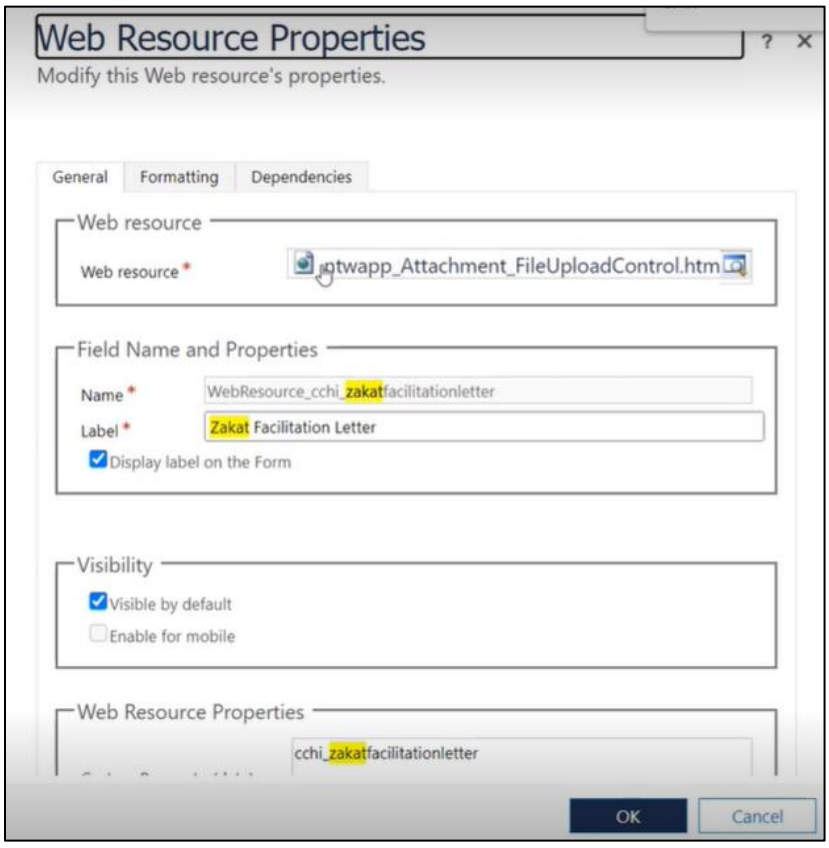


Figure 51 Add Document Preview Web Resource on the form

As a result in the screenshot below, you can click on the License Attachment to preview the file, but for License renew letter it is empty and shows "No Data Found"

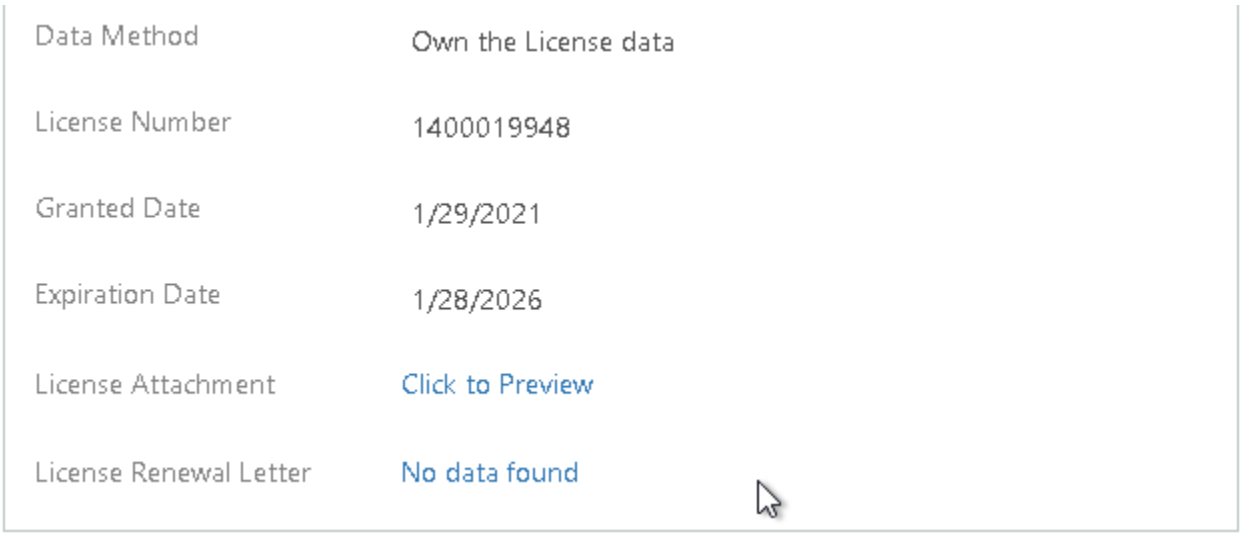


Figure 52 How Required Attachments looks on the form

On the other hand the preview show look as below

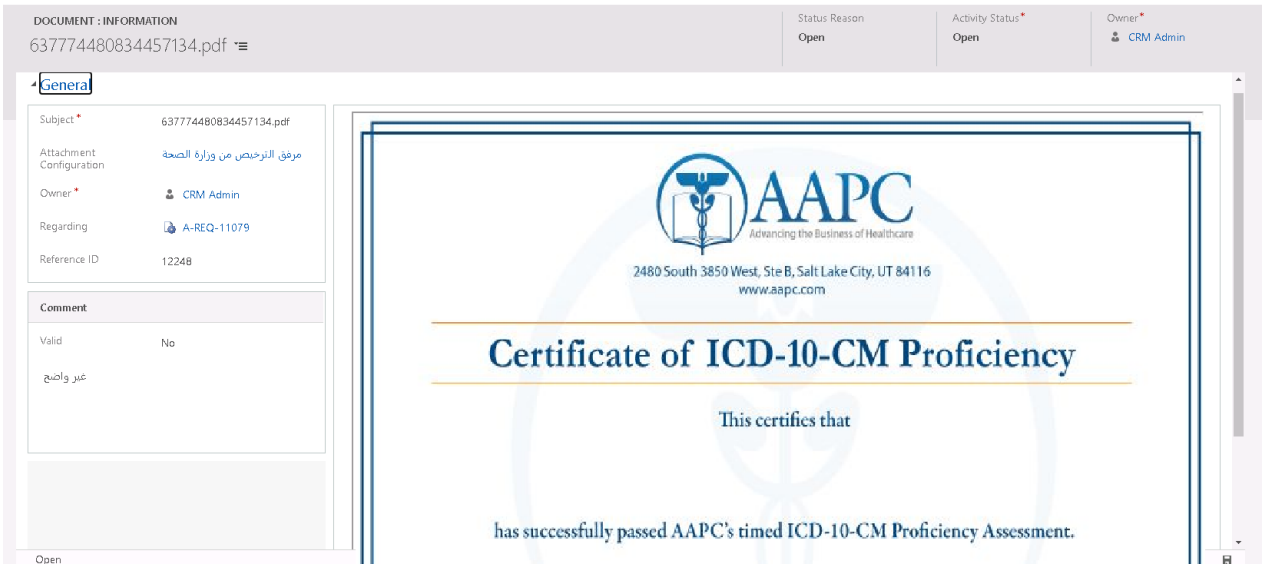


Figure 53 How to Preview a Document