SharePoint Export via Graph API [ENG]

Mit dieser Schnittstelle kann die offizielle SharePoint Graph API von Microsoft integriert werden, um mit einem SharePoint Verzeichnis zu kommunizieren.
Dafür ist eine registrierte Microsoft Azure Applikation notwendig.

Integrierte Services der Graph API

- Dokumente in ein zuvor definiertes SharePoint Verzeichnis hochladen

Zugangsberechtigungen in Microsoft Azure

Benötigte Applikationen

Erstellen Sie zwei neue Applikationen in Microsoft Azure.
Eine dient zu administrativen Zwecken, die Andere für den Nutzer selbst.

> Zur Bewilligung der folgenden Rechte wird ein User mit Administrationsrechten in Microsoft Azure benötigt.

Die Rechte können im Menü "API Permissions" (im deutschen Client: "API-Berechtigungen") hinzugefügt werden:

- Die Admin-Applikation benötigt die API-Berechtigung `Sites.FullControl.All`
- Die Nutzer-Applikation benötigt die API-Berechtigung `Sites.Selected`

In der Admin-Applikation muss ein User im Menü `Certificates & secrets` erstellt werden.
Dieser User ist nötig, um die Nutzer-Applikation mit den nötigen Rechten auszustatten.

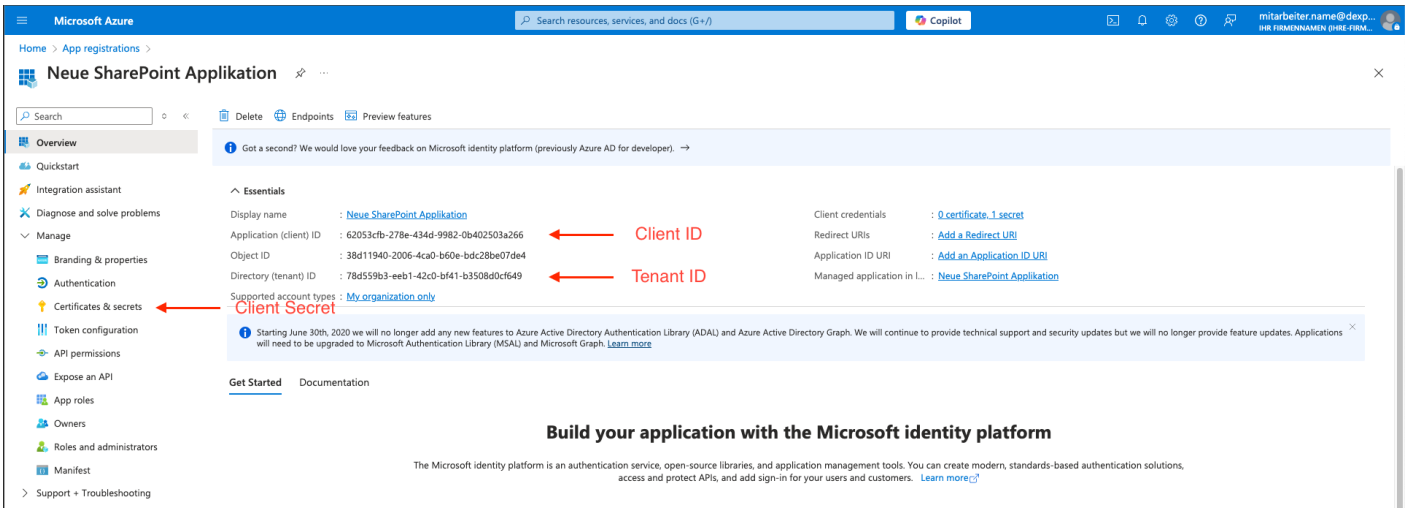


Abbildung - Registrierte Microsoft Applikation "Neue SharePoint Applikation"

Rechte für die Nutzer-Applikation

Für diesen Abschnitt benötigen Sie die Daten der zuvor erstellten Admin-Applikation.

Die Nutzer-Applikation muss zwingend die Berechtigungen auf die von Ihnen gewünschte SharePoint Seite erhalten.
Dafür ist die Id der SharePoint-Seite notwendig.

> Bedauerlicherweise ist es zum aktuellen Zeitpunkt nicht möglich, die SharePoint Seiten Id über die grafische Oberfläche ausfindig zu machen (Stand März 2025).

Daher wird wie folgt beschrieben, wie die Id der Seite ermittelt und die Erteilung der Berechtigung per curl Requests erteilt werden kann.

Einloggen

Zu erst benötigen Sie einen Bearer Token, um sich an der API zu authentifizieren. Dies erfolgt durch folgenden Request:

- URL: https://login.microsoftonline.com/<TENANT_ID>/oauth2/v2.0/token
- HTTP Methode: POST
- Request Body Typ: form-data
- Request Body:
  - client_id:<CLIENT_ID>
  - client_secret:<CLIENT_SECRET>
  - grant_type:client_credentials
  - scope:https://graph.microsoft.com/.default

Statt "<TENANT_ID>", "<CLIENT_ID>" und "<CLIENT_SECRET>" müssen Sie Ihre jeweiligen Daten verwenden.

Wo Sie diese Daten finden steht in den Abschnitten "Feld Tenant Id", "Feld Client Id" und "Feld Client Secret" dieses Artikels.

Anbei der cURL Befehl:

```
curl --location 'https://login.microsoftonline.com/<TENANT_ID>/oauth2/v2.0/token' \--form
'client_id="<CLIENT_ID>"' \--form 'client_secret="<CLIENT_SECRET>"' \--form
'grant_type="client_credentials"' \--form 'scope="https://graph.microsoft.com/.default"'
```

Dieser Request gibt den Bearer Token zurück, im Index "access_token" der Response. Dieser Bearer Token wird im folgenden "<BEARER_TOKEN>" genannt, zur Verdeutlichung wo dieser einzusetzen ist.

Site Id erhalten

Die Site Id wird für den Request zu den Rechten benötigt:

- URL: https://graph.microsoft.com/v1.0/sites/<TENANT>.sharepoint.com/sites/<NAME>
- HTTP Methode: GET
- Header: "Authorization: Bearer <BEARER_TOKEN>"

Anbei der cURL Befehl:

```
curl --location 'https://graph.microsoft.com/v1.0/sites/<TENANT>.sharepoint.com:/sites/<NAME>'
\--header 'Authorization: Bearer <BEARER_TOKEN>'
```

Die Response dieses Requests ist etwas umständlicher zu lesen, ein Beispiel:

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#sites/$entity",
  "createdDateTime": "2025-03-13T00:11:15.033Z",
  "description": "TestDescription",
  "id": "<TENANT>.sharepoint.com,808dec17-aa6d-4158-a9fe-8caa8d909dff,81f7ad14-65ae-46c2-b0fc-10
  "lastModifiedDateTime": "2025-03-13T07:07:19Z",
  "name": "TestName",
  "webUrl": "https://<TENANT>.sharepoint.com/sites/<SITE_NAME>",
  "displayName": "TestName",
  "root": {},
  "siteCollection": {
    "hostname": "<TENANT>.sharepoint.com"
  }
}
```

Die Site Id in diesem Fall ist der Wert `808dec17-aa6d-4158-a9fe-8caa8d909dff`.
Er befindet sich im Index "id" und ist der mittlere der Werte (wenn den Wert an seinen Kommata aufteilt).

Zugriffsrechte erteilen

Für die Zugriffsrechte benötigen sie die Client ID der Nutzer-Applikation und den Namen.
Diese setzen Sie an der Stelle von "<NUTZER_APPLIKATION_CLIENT_ID>" bzw. "<NUTZER_APPLIKATION_NAME>" ein.

Die Rechte zum Schreiben beinhalten auch die Leserechte.

- URL: https://graph.microsoft.com/v1.0/sites/<SITE_ID>/permissions
- HTTP Methode: POST
- Request Body Type: JSON
- Request Body:
  - ```
    {
      "roles": [
        "write"
      ],
      "grantedToIdentities": [
        {
          "application": {
            "id": "<NUTZER_APPLIKATION_CLIENT_ID>",
            "displayName": "<NUTZER_APPLIKATION_NAME>"
          }
        }
      ]
    }
    ```

Anbei der cURL Befehl:

```
curl --location 'https://graph.microsoft.com/v1.0/sites/<SITE_ID>/permissions' \
--header 'Authorization: Bearer <BEARER_TOKEN>' \
--header 'Content-Type: application/json' \
--data '{
  "roles": ["write"],
  "grantedToIdentities": [{
    "application": {
      "id": "<NUTZER_APPLIKATION_CLIENT_ID>",
      "displayName": "<NUTZER_APPLIKATION_NAME>"
    }
  }]
}'
```

Der Server gibt im Erfolgsfall eine Response mit dem Statuscode 201 zurück.
Da der Statuscode aussagekräftig genug ist, kann der Inhalt der Response vernachlässigt werden.

Konfiguration

Um diese Schnittstelle zu verwenden und Aktionen auszuführen, muss an der Squeeze Dokumentenklasse eine Export-Schnittstelle definiert werden.
Die benötigten Zugangsdaten entnehmen Sie aus der Nutzer-Applikation.
```

Authentifizierung

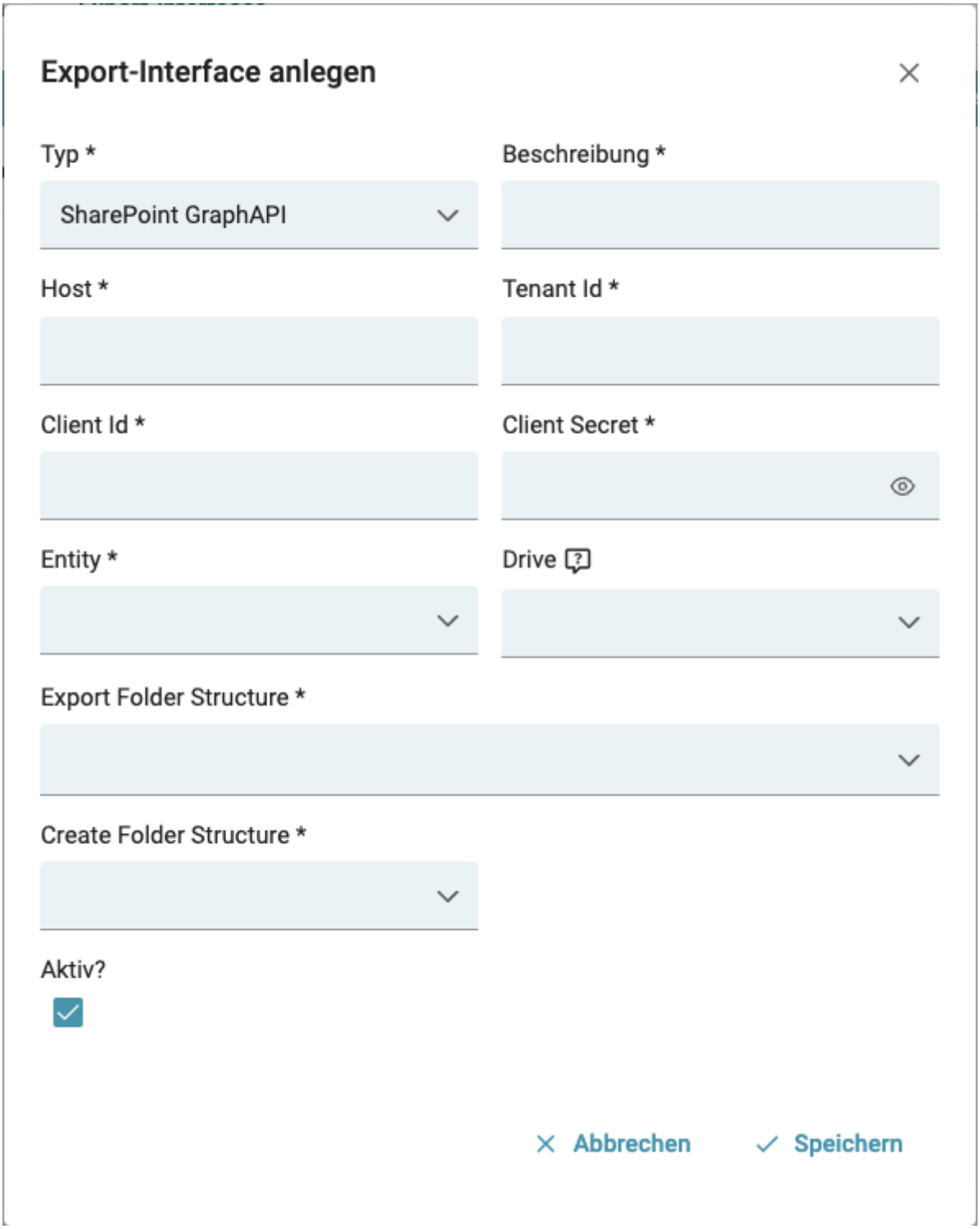


Abbildung - Export Interface SharePoint GraphAPI (WIP)

Feld *Beschreibung*

Für das Export Interface kann ein Name angegeben werden, im Feld Beschreibung.

Feld *Host*

In diesem Feld wird die Adresse zur SharePoint Seite hinterlegt. Diese hat folgendes Muster: `<TENANT>.sharepoint.com/sites/<NAME>`.

Feld *Tenant Id*

Die Tenant Id befindet sich in der Übersicht einer registrierten App. Eine Auflistung dieser Apps finden sie hier.
Um die Liste zu sehen müssen Sie eingeloggt sein.

Die Tenant Id wird auf der Seite auch "Directory (tenant) ID".

Feld *Client Id*

Die Tenant Id befindet sich in der Übersicht einer registrierten App. Eine Auflistung dieser Apps finden sie hier.
Um die Liste zu sehen müssen Sie eingeloggt sein.

Hier wird sie auch unter dem Begriff "Application (client) ID" geführt.

Feld *Client Secret*

Die Tenant Id befindet sich in der Übersicht einer registrierten App. Eine Auflistung dieser Apps finden sie hier.
Um die Liste zu sehen müssen Sie eingeloggt sein.

Das Client Secret wird innerhalb der registrierten App hinterlegt. Das dafür nötige Menü finden Sie in, in der ausgewählten App, unter dem Menü "Certificates & secrets".

Für das Erstellen eines Secrets müssen sie lediglich eine aussagekräftige Beschreibung angeben, und den Zeitraum, in welchem das Secret valide ist.
Das Client Secret wird bei Microsoft unter "Secret Value" geführt. Diese Information wird nur einmalig angezeigt, aus diesem Grund sollte es in einer sicheren Umgebung gespeichert werden. Verwechseln Sie das Client Secret nicht mit "Secret Value".

Feld *Entity*

Über die Entity wird der Service ausgewählt, welchen Sie verwenden möchten:

- SharePoint Document Export
  - Service um ein Dokument hochzuladen

Feld *Export Folder Structure*

Dateien werden alle in das Home Verzeichnis des SharePoint Verzeichnisses hochgeladen.
Um zu spezifizieren, in welche Verzeichnis Struktur ein Dokument abgelegt werden soll, können in dieser Auswahlliste mehrere Felder ausgewählt werden.
Die Felder werden durch ihre in SQUEEZE erkannten Daten ersetzt. So könnte beispielsweise die Auswahl des Feldes "IBAN" dazu führen, dass ein Dokument in das Verzeichnis der erkannten IBAN abgelegt wird (nicht in einem Verzeichnis mit dem Namen "IBAN").

Feld *Drive*

Bei der korrekten Eingabe von Host, Tenant ID, Client ID und dem Client Secret, wird im Feld *Drive* eine Liste von möglichen Verzeichnissen aufgeführt, welches als Home Verzeichnis des SharePoints dienen soll, das Sie auswählen müssen.

Feld *Create Folder Structure*

Diese Funktion erstellt beim Wert "Ja" ein Pfad Verzeichnis, auch wenn es vorher nicht existiert. Sollte das Verzeichnis bereits existieren wird kein neues Verzeichnis erstellt.
Der Pfad für das Verzeichnis wird durch das Feld *Export Folder Structure* bestimmt.

Beim Wert "Nein" wird beim Export an Sharepoint geprüft, ob der Verzeichnis Pfad existiert. Existiert das Verzeichnis, dann werden die verarbeiteten Dokumente hochgeladen (exportiert). Existiert das Verzeichnis jedoch nicht, dann wird das Dokument nicht hochgeladen (nicht exportiert).
Es wird beim Export eine entsprechende Fehlermeldung ausgegeben.
Dadurch soll vermieden werden, das unerwartete Verzeichnisse erstellt werden und Dokumente ggf. verschwinden.

Beispiel

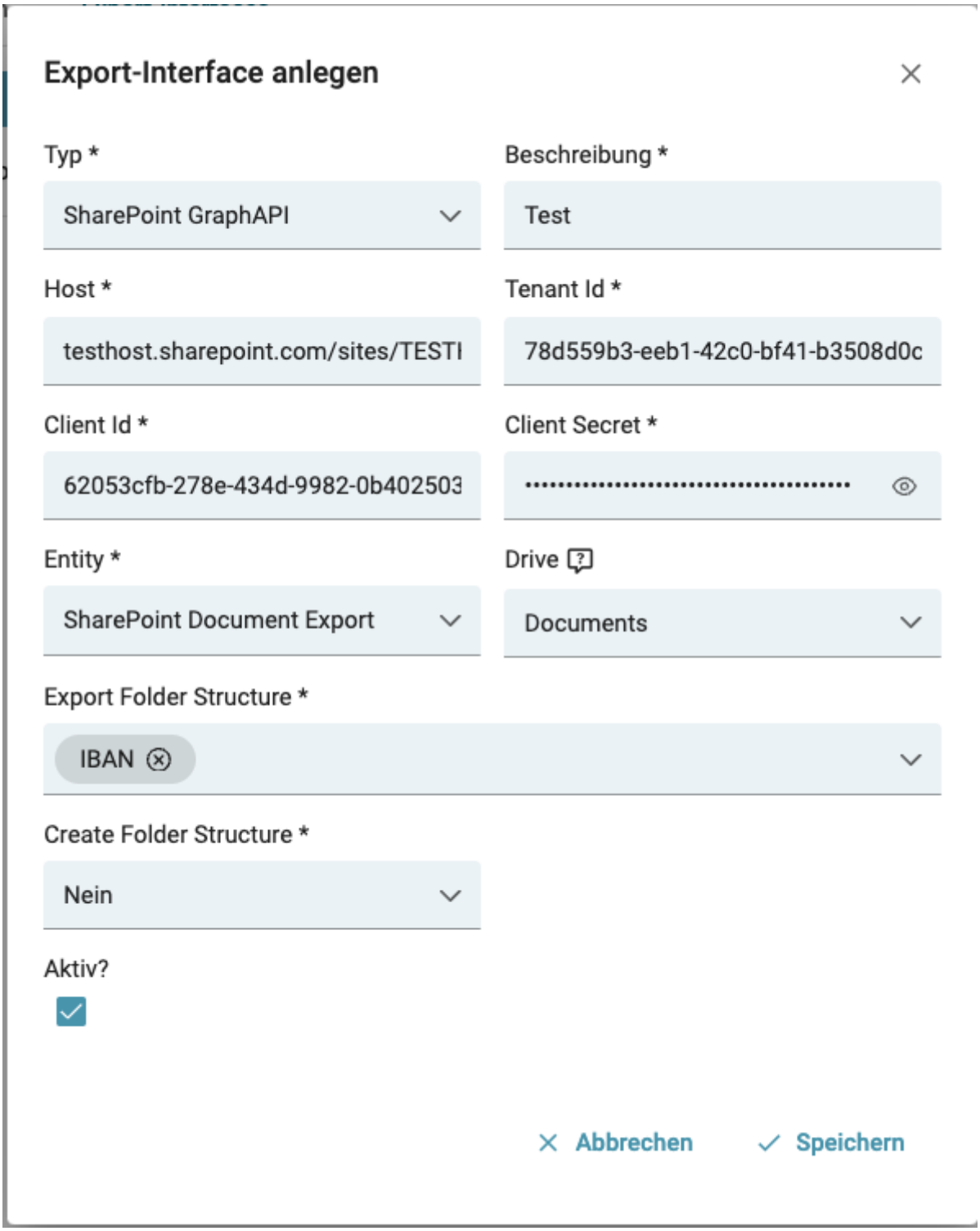


Abbildung - Beispiel ausgefülltes Formular für Export Interface (WIP)

Wenn wir eine neue Applikation in Microsoft Azure erstellt haben, rufen wir diese auf und erhalten folgende Ansicht.

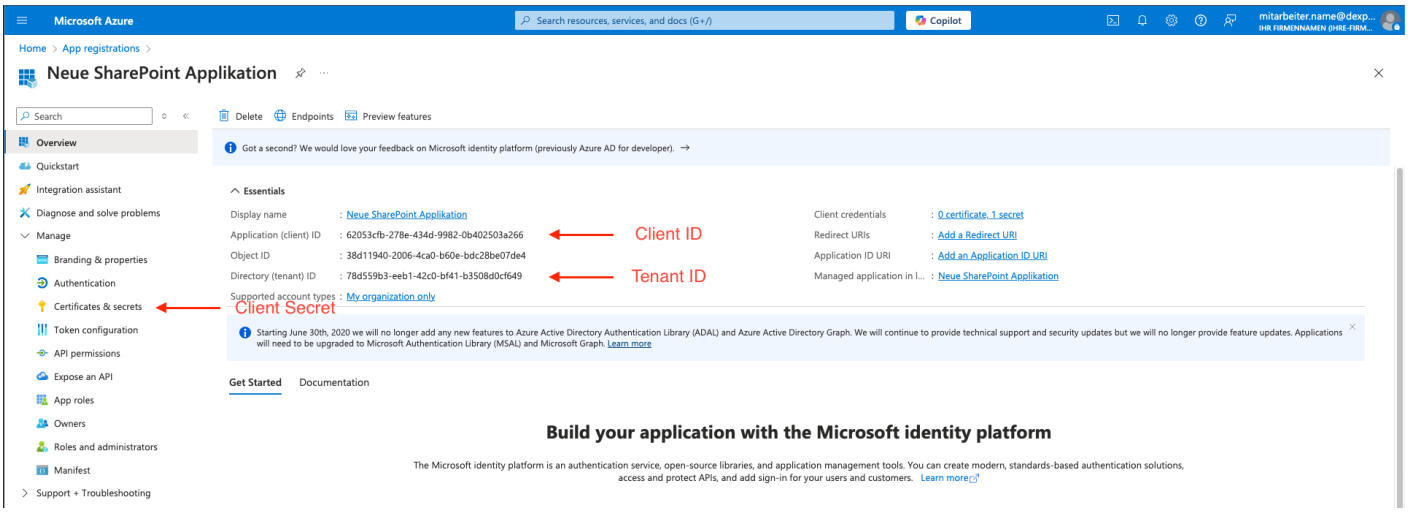


Abbildung - Registrierte Microsoft Applikation "Neue SharePoint Applikation"

Auf dieser Abbildung ist zu sehen was die Client ID ist, sowie wo die Tenant ID zu finden ist.
Bei Auswahl des Menüs "Certificates & secrets" existiert ein kleiner Button mit dem Namen "New client secrets". Nachdem die nötigen Daten angegeben wurden erhalten wir einen neuen Eintrag:

Test	8/3/2025	uN98Q~ygKu2f2EXpbZ~Rl6F3nRzGmfBL...	614ef681-df9a-450a-8083-bdf13b295a9b

Feldwerte / Metadaten exportieren

Um Feldwerte als Metadaten zu exportieren, beim Upload, müssen diese in der Dokumentenklasse angegeben werden.
Dafür muss das nötige Feld ausgewählt werden, und der entsprechende Wert muss im Feld "Name (Zielsystem)" eingetragen werden.

Allgemein | **Felder** | Tabellen | Export-Interfaces

Kopfdaten | Neue Feldgruppe +

ID	Name	Beschreibung	Typ	Standard Wert	Lokator	Lokator-Unterfeld	Name (Zielsystem)	Pf
1	Company	Mandant	Text	1000	Mandant		Principal	
256	CompanyCountry	Mandantenland	Text	DE	Mandantenland		CompanyCountry	
2	Creditor	Lieferanten-Nr.	Text		Lieferant		CreditorID	
3	CreditorName	Lieferanten-Name	Text		Lieferantenname	Name	CreditorName	
4	CreditorCountry	Lieferanten-Land	Text		Lieferantenland		Creditor_Country	
182	DocumentType	Vorgang	Text		Vorgang		DocumentTypeTest	
220	CostCenter	Kostenstelle	Text		Kostenstelle		CostCenter	
180	DocumentSubType	DocumentSubType	Text		Vorgang		InvoiceCreditVoucher	
6	DocumentDate	Belegdatum	Date		Belegdatum		InvoiceDate	
7	DocumentReference	Belegnummer	Text		Belegnummer		InvoiceNumber	
8	OrderNumber	Bestellung	Text		Bestellnummer		OrderNumber	

Abbildung - Kopfdaten-Felder der Dokumentenklasse

Revision #1
Created 2025-07-14 10:07:21 UTC by Vahdettin Balum
Updated 2025-07-14 10:07:21 UTC by Vahdettin Balum