

ALM – Configuração de esteira CI/CD

Este documento tem o objetivo de descrever os passos para a configuração de processo de deploy automático das soluções de Dynamics 365/Power Platform nos ambientes de UAT e Produção.

Todos os projetos deverão ser configurados para a publicação automática para atender os seguintes objetivos:

1. Evitar a criação de camadas de customização nos ambientes de UAT e Produção. Desta forma poderemos garantir que todas as alterações serão aplicadas.
2. Evitar a necessidade de acesso de administrador aos ambientes de Produção melhorando a segurança do ambiente.
3. Menor tempo de publicação.
4. Versionamento das soluções no Azure DevOps. Desta forma poderemos recuperar versões antigas ou recriar ambientes de desenvolvimento.
5. Padronização nos processos de deploy.

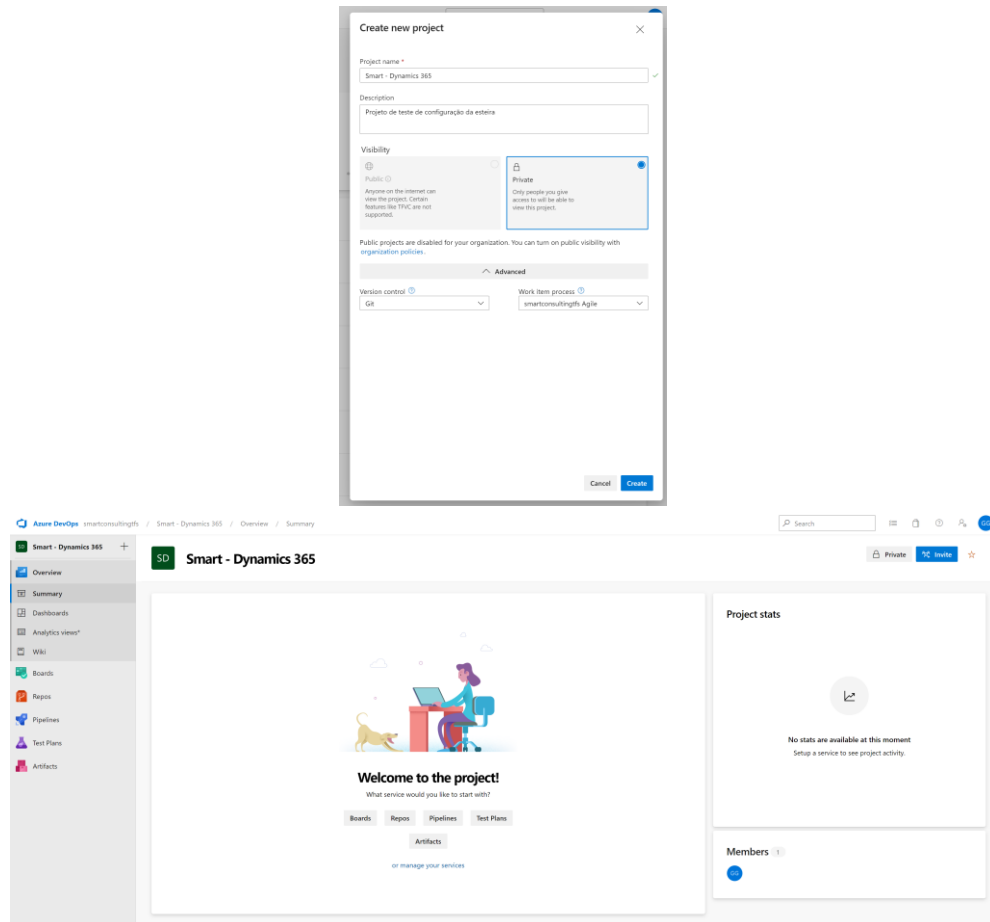
Etapas de configuração

Criação dos ambientes de desenvolvimento, UAT e produção

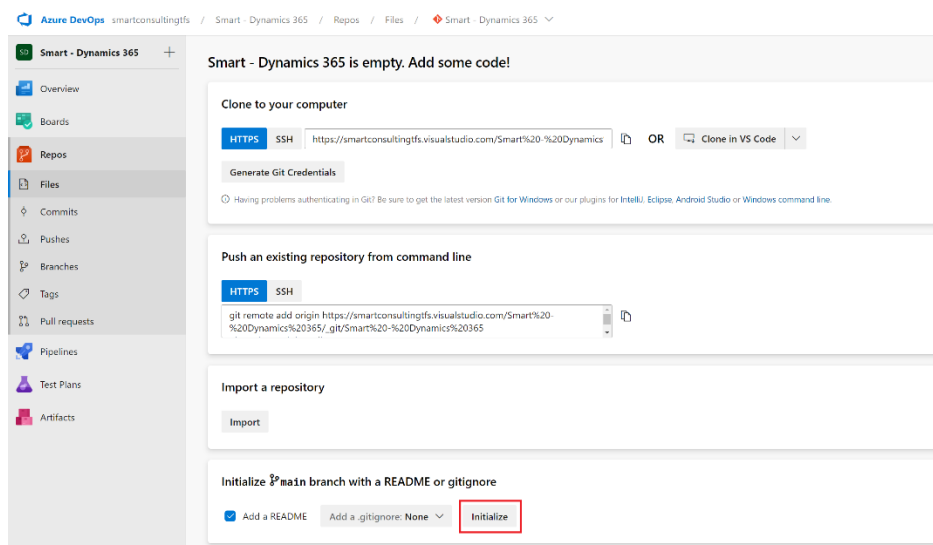
1. Criar os ambientes de Dataverse para Desenvolvimento, UAT e Produção.
2. Criar uma solução não gerenciada no ambiente de desenvolvimento seguindo os padrões de projeto. Esta solução será a base para a esteira.

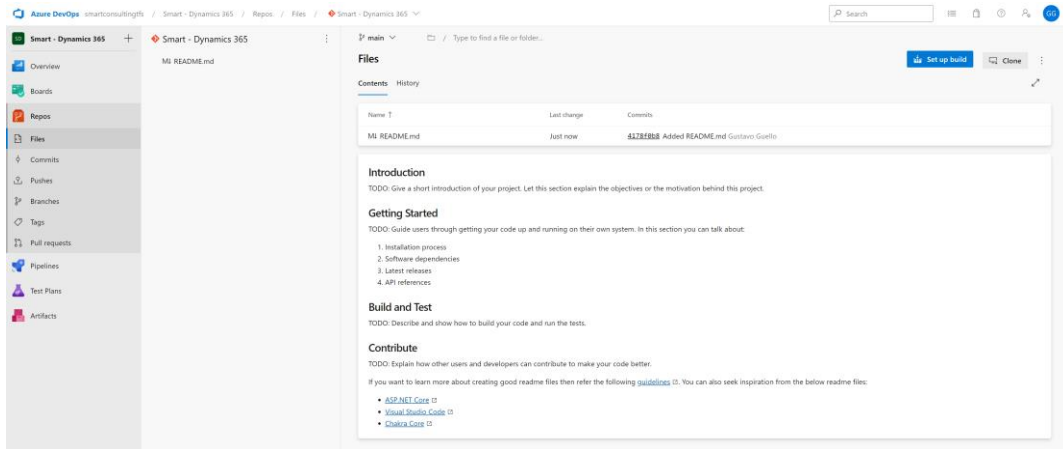
Configuração do ambiente do Azure DevOps

1. Criar um projeto no Azure DevOps que armazenará os códigos-fonte e a esteira. Neste projeto também serão registrados os requisitos e sprints.



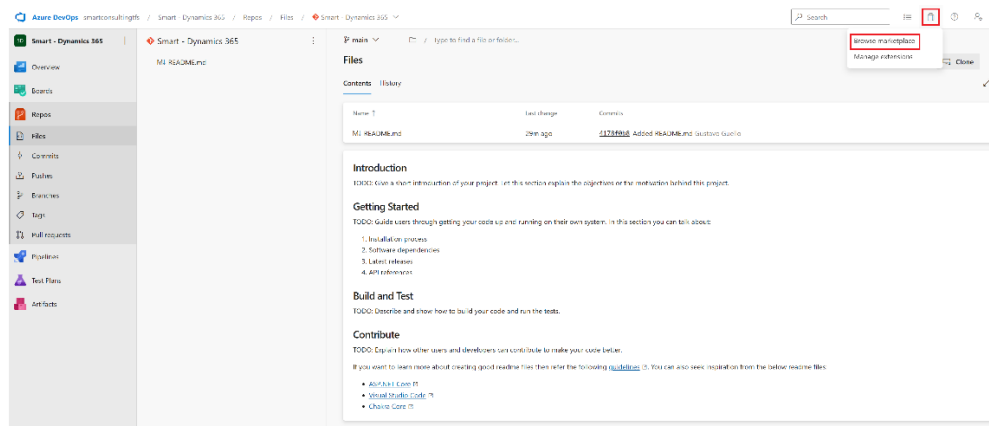
2. Ativar o repositório de códigos-fonte
 - a. Acesse o menu Repos e depois Files
 - b. Na opção Initialize main branch with a README or gitignore, clique em Inicialize



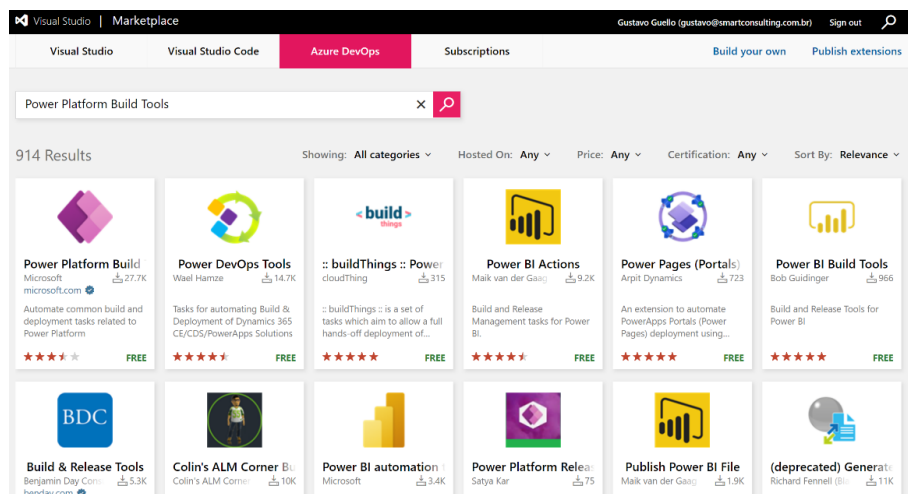


3. Somente para novos ambientes de Azure DevOps é necessário instalar a solução do Power Platform Build Tools.

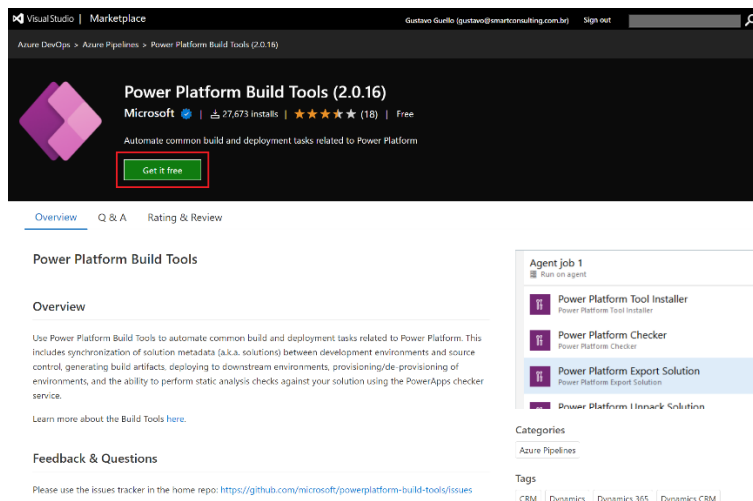
a. Acessar a Loja de aplicativos do Azure DevOps



b. Pesquisar Power Platform Build Tools

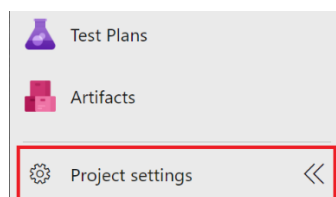


c. Instalar a solução

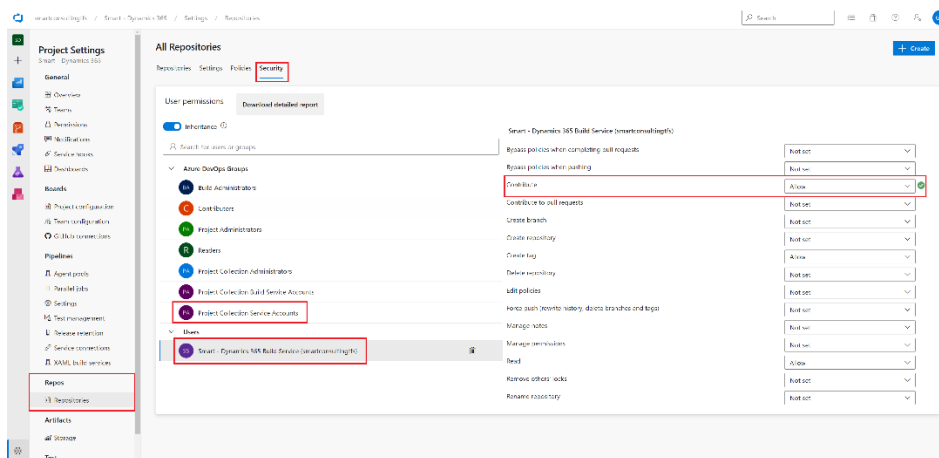


4. Configurar as permissões

a. Acesse o menu Project settings



b. Acesse o menu Repos > Repositories e depois clique na aba Security. Ajuste as permissões dos usuários Project Collection Service Accounts e <ProjectName> Build Service <OrgName> para permitir (Allow) em Contribute.

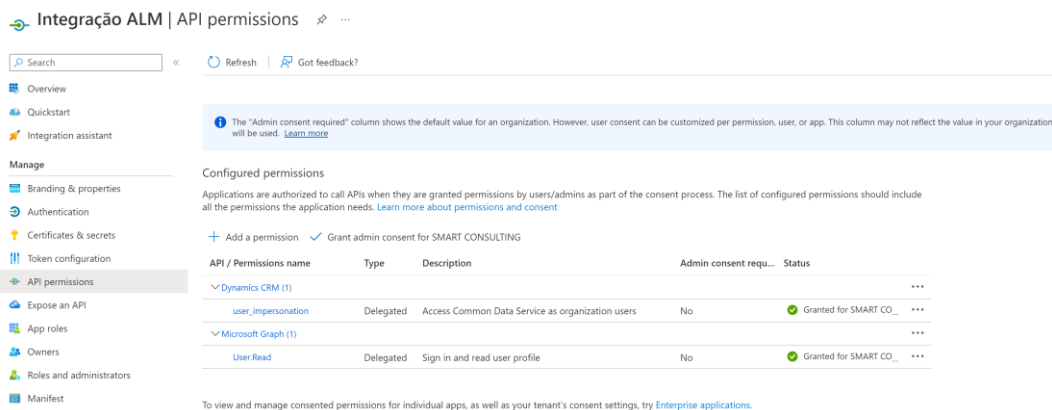


Criar um usuário de serviço

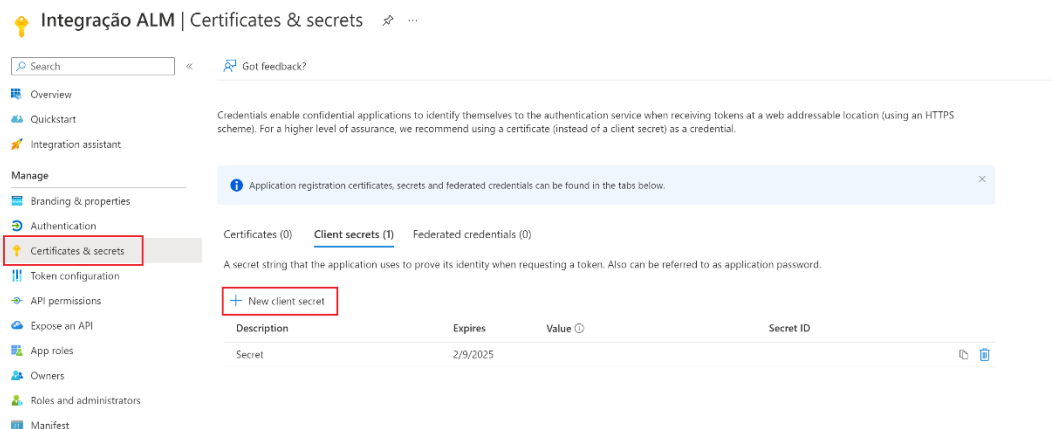
Deverá ser criado um usuário de serviço que será usado na conexão do Azure DevOps com os ambientes para realizar o deploy.

OBS: No ambiente da Smart não é necessário criar um novo usuário de aplicativo. Sempre usaremos o usuário Integração ALM.

1. Acesse o portal do Azure <https://portal.azure.com>
2. Procure por App Registrations e clique em New Registration
3. Preencha o nome e deixe os outros campos com o valor padrão
4. Depois de criado, acesse o menu APO permissions, clique em Add Permission, selecione Dynamics CRM e adicione a permissão.
5. Depois clique em Grant admin consent for <Organization> na lista de permissões.



6. Vá em Certificates & secrets e clique em New Client Secret. Copie o conteúdo do campo Value e anote para utilizar depois. Este valor não será exibido novamente.

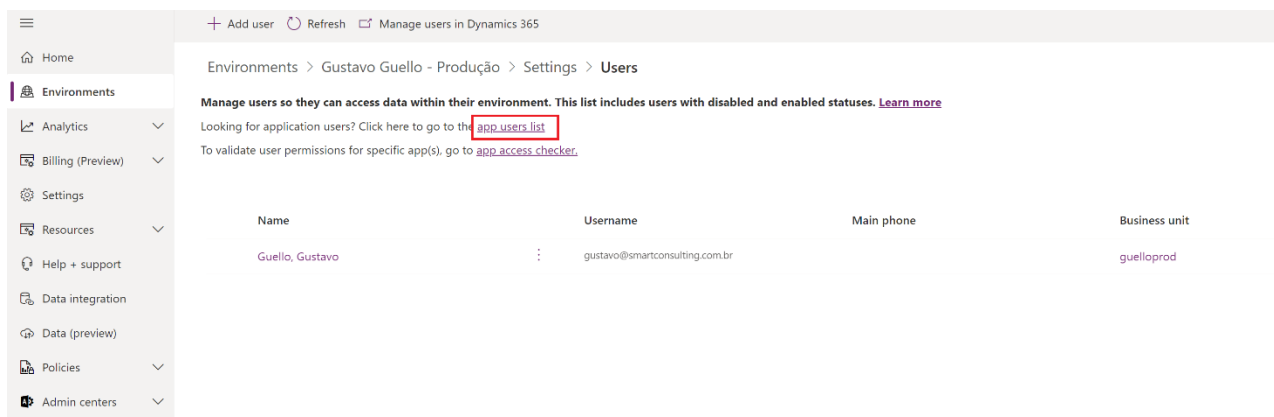


7. Vá em Overview e copie os valores de Application (client) ID e Directory (tenant) ID.

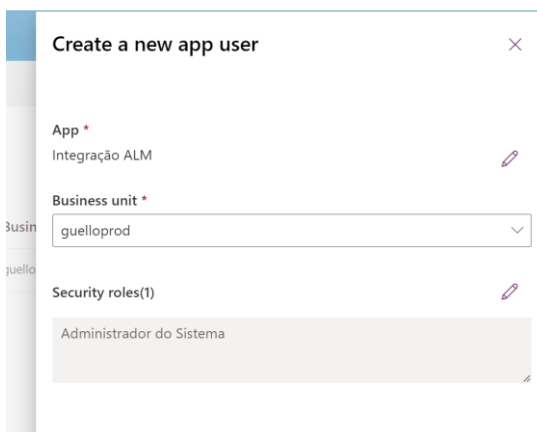
Adicionar permissão no ambiente do Dataverse

Agora devemos adicionar o usuário de aplicativo criado nos ambientes de origem e destino. No caso de Tenants separados (desenvolvimento na Smart em um Trial e destino no cliente), deverão ser criados um usuário de aplicativo em cada Tenant e as permissões serão dadas nos ambientes específicos.

1. Acesse a Central Administrativa da Power Platform e clique em ambientes.
<https://admin.powerplatform.microsoft.com/environments>
2. Selecione o ambiente de origem
3. No menu da direita, acesse usuários > Ver Todos > App user list



4. Clique em New App User, busque o usuário de aplicativo criado (Integração ALM) e adicione o perfil de System Administrator.

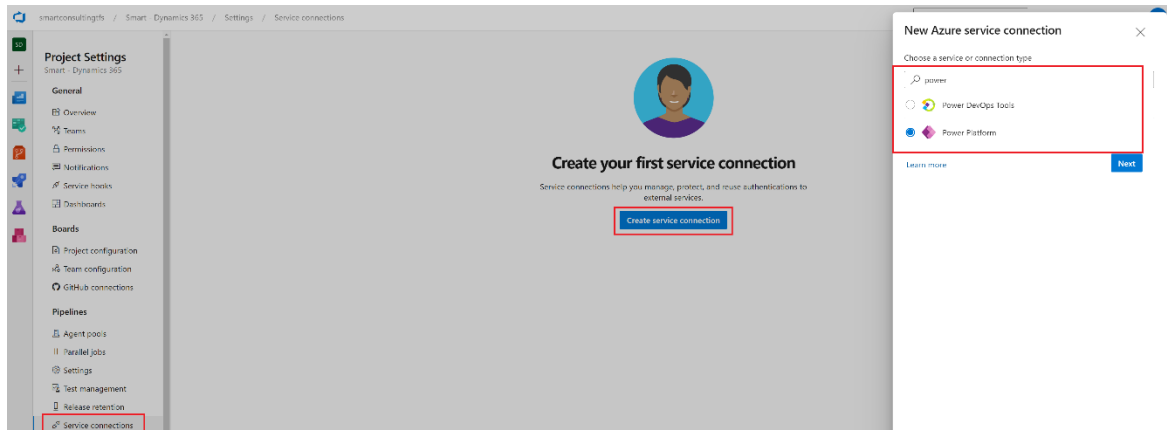


5. Repita os passos acima para o ambiente de destino

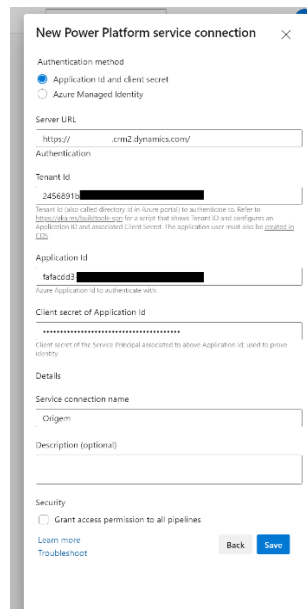
Criar conexão do Azure DevOps

O próximo passo é criar as conexões com os ambientes dentro do Azure DevOps.

1. Acesse o projeto do DevOps criado e então Project Settings
2. Em Pipelines, clique em Service Connections > New Service Connection > Selecione Power Platform



3. Selecione o método de autenticação como sendo Application Id e cliente secret, preencha a URL do ambiente de origem, o Tenant ID, Application ID e Client Secret e dê um nome para a conexão.

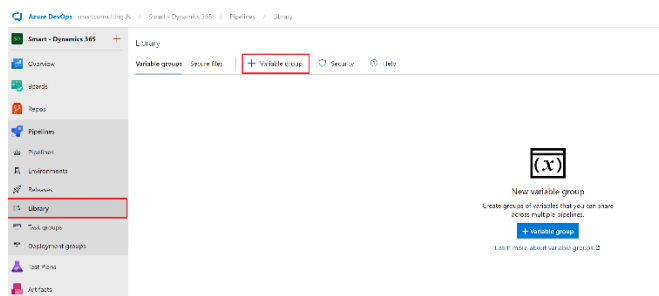


4. Repita o processo para o ambiente de destino

Criar variáveis de ambiente

Serão criadas algumas variáveis de ambiente para a configuração das esteiras de publicação.

1. No Azure DevOps, navegue para pipeline > Library
2. Clique em + Variable Group

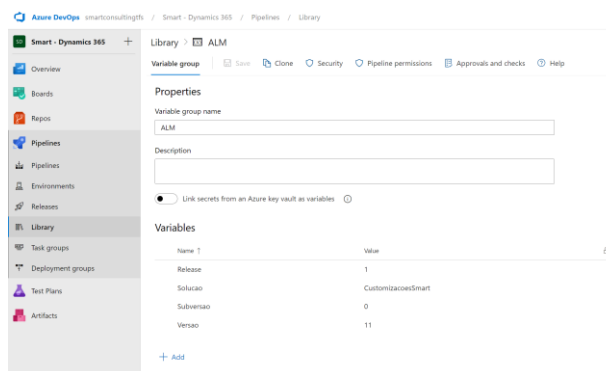


3. Defina o nome como ALM
4. Crie 4 variáveis com os nomes abaixo
 - a. **Release** – Preencha com o valor da Release atual da solução que será extraída. Deverá ser atualizado manualmente sempre que necessário.
 - b. **Versao** - Preencha com o valor da Versão atual da solução que será extraída. Este valor será incrementado automaticamente sempre que rodar o deploy para o ambiente de destino.
 - c. **Subversao** - Preencha com o valor da Subversão atual da solução que será extraída. Deverá ser atualizado manualmente sempre que necessário.
 - d. **Solucao** – Nome lógico da solução a ser exportada

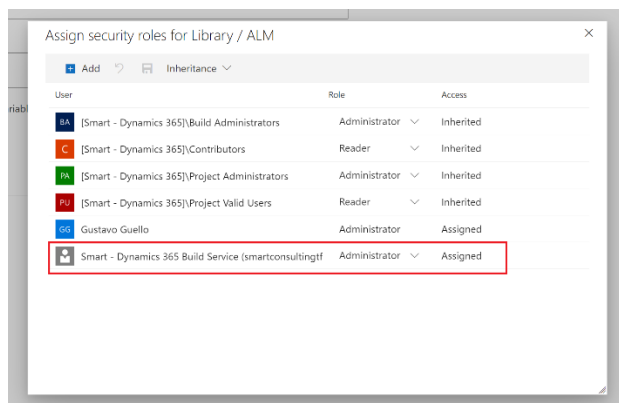
OBS: Usaremos o seguinte padrão de numeração da versão das soluções: [Realese].[Versao].[Subversao].0

OBS2: Se for necessário criar esteiras para mais de uma solução, deve-se criar novos grupos de variáveis com as mesmas variáveis.

5. Clique em Save e anote o ID do grupo de variáveis que é apresentado na url.
(.../_library?itemType=VariableGroups&view=VariableGroupView&variableGroupId=3&path=ALM)



6. Clique em Security, adicione o usuário <Projeto> Build Service (<Organização>) e atribua permissão de Administrador.



Gerar o arquivo de configuração

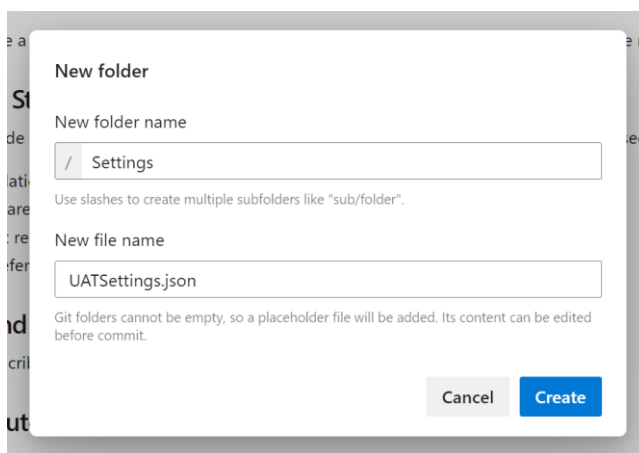
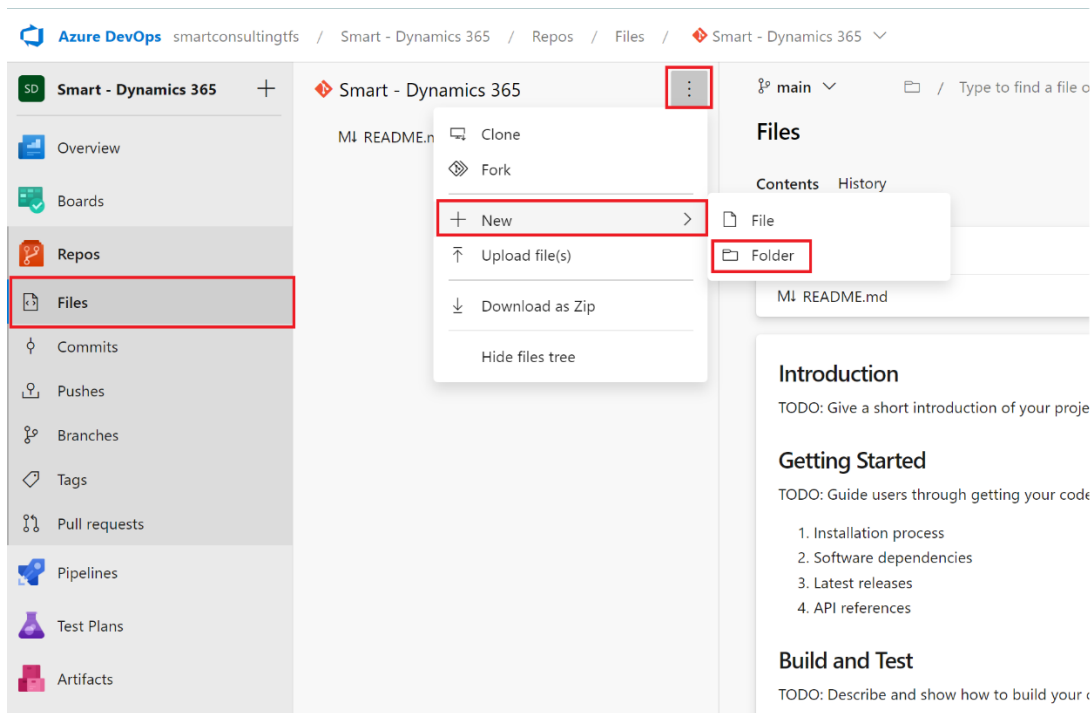
Algumas configurações podem ser diferentes no ambiente de origem e destino como por exemplo:

1. Conexões
2. Variáveis de Ambiente

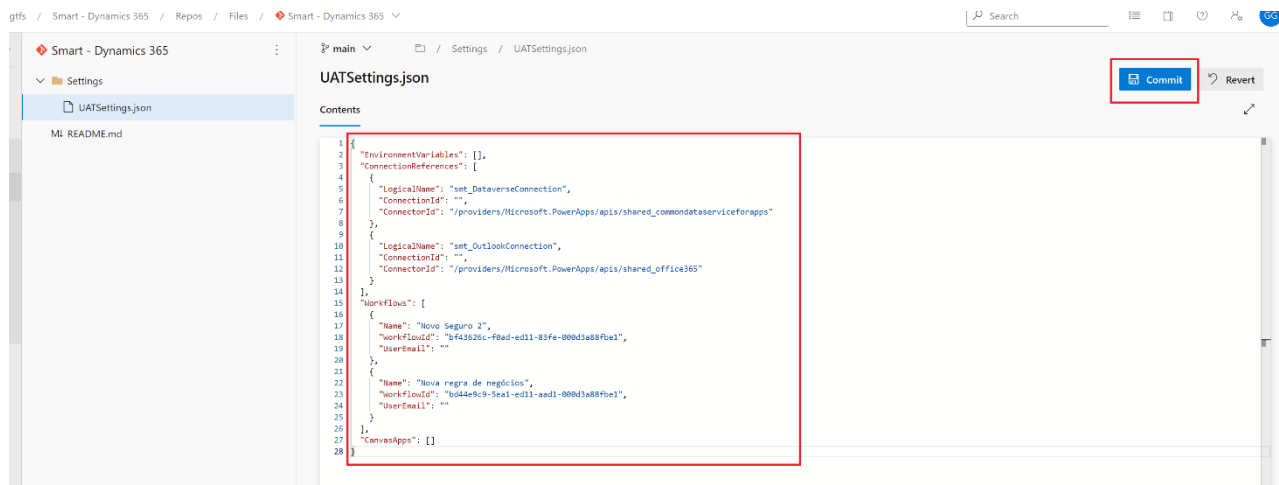
Para automatizar o processo de deploy, é necessário criar um arquivo de configuração para alterar estes parâmetros durante o processo de geração da solução.

1. Instalar o Cliente da Power Platform na sua máquina (se ainda não tiver instalado). Também pode ser usado para atualizar a versão da sua máquina.
 - a. Abra uma janela de PowerShell
 - b. Digite **pac install latest**
2. Conectando com o ambiente
 - a. No PowerShell digite **pac auth list** para listar os ambientes que estão conectados. Caso o ambiente não esteja configurado, digite **pac auth create --url https://myorg.crm.dynamics.com** para criar uma nova conexão. Será aberta uma tela para conexão.
 - b. Para selecionar a conexão que deseja usar, digite **pac auth select --index [número da conexão]**. Por exemplo: **pac auth select --index 1**
3. Mudando de diretório
 - a. Digite o comando para mudar para uma pasta específica: **cd d:\Temp**
4. Exportando a solução não gerenciada
 - a. No PowerShell digite **pac solution export --name <solutionname> --path .\ --managed false**. Por exemplo: **pac solution export --name CustomizacoesSmart --path .\ --managed false**

5. Criando o arquivo de configuração
 - a. No PowerShell digite **pac solution create-settings --solution-zip .\<SolutionName>.zip -settings-file <SolutionName>.json**. Por exemplo: **pac solution create-settings --solution-zip .\CustomizacoesSmart.zip --settings-file UATSettings.json**
6. No repositório no Azure DevOps, crie uma pasta Settings e crie um arquivo chamado UATSettings.json.



7. Copie o conteúdo do arquivo de configurações gerado para este arquivo e clique em Commit.



OBS: Sempre que forem criadas novas conexões (com provedores diferentes), novos Canvas e variáveis de ambiente, este arquivo deverá ser atualizado.

Atualizar os parâmetros

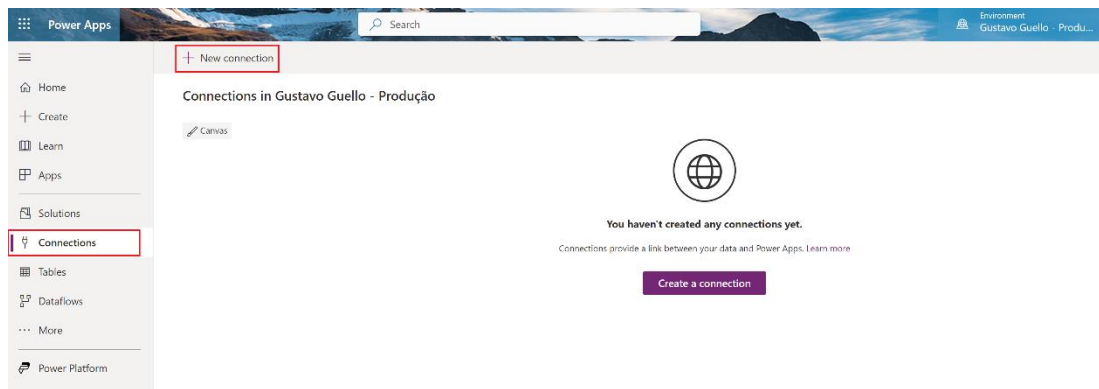
Os parâmetros do arquivo de configuração devem ser atualizados com os dados do ambiente de destino. Pode-se criar arquivos de configuração diferentes para ambientes diferentes.

Variáveis de ambiente

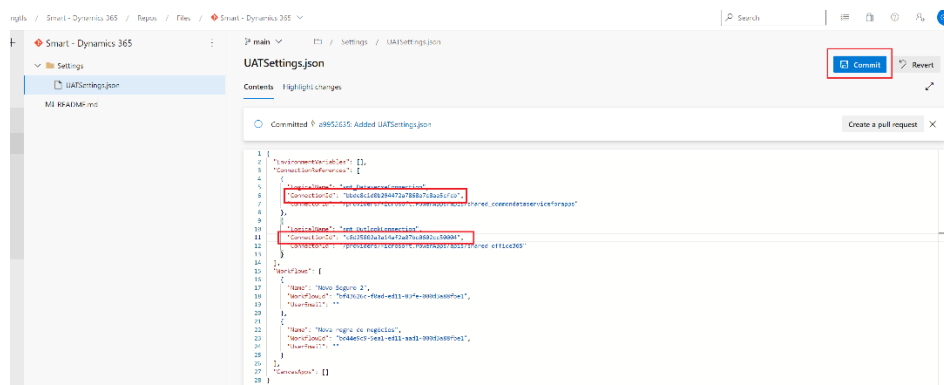
Atualize os valores diretamente no arquivo de configuração

Referências de Conexão

1. Crie uma conexão para cada tipo usado (Dataverse, Outlook, etc.) no ambiente de destino com o usuário a ser utilizado.
2. Acesse o <https://make.powerapps.com/> e altere para o ambiente de destino
3. Clique no menu Connections e New Connection



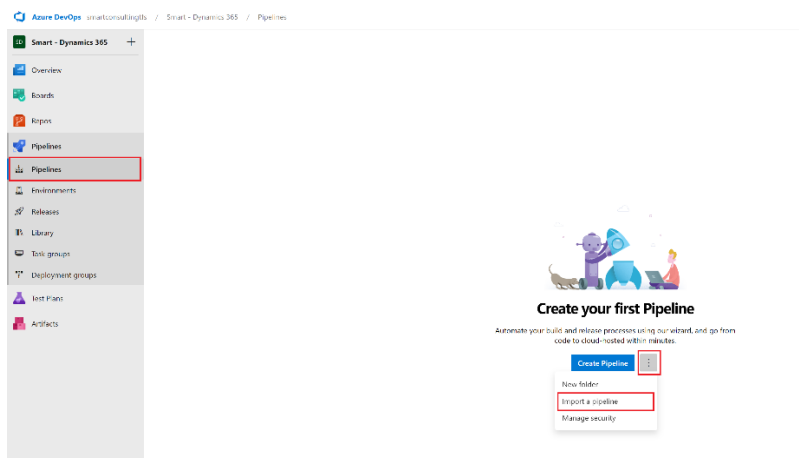
4. Selecione o conector e crie a nova conexão
5. Clique no nome do usuário para abrir os detalhes da conexão
6. Na URL, copie o ID da Conexão
 - a. https://make.powerapps.com/environments/db2271cb-d7bb-ee91-8a97-64bd8cd134e2/connections/shared_commondataserviceforapps/bbde8c1d0b294472a7868e7c8ae5cfcb/details#
7. Repita os passos para cada conexão necessária
8. No arquivo de configuração, preencha os Ids e clique em Commit



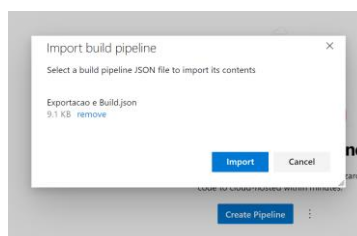
Criar o Pipeline de Exportação e Build

Vamos importar o pipeline que será responsável por exportar a solução do ambiente de origem, versionar e gerar as soluções para importação.

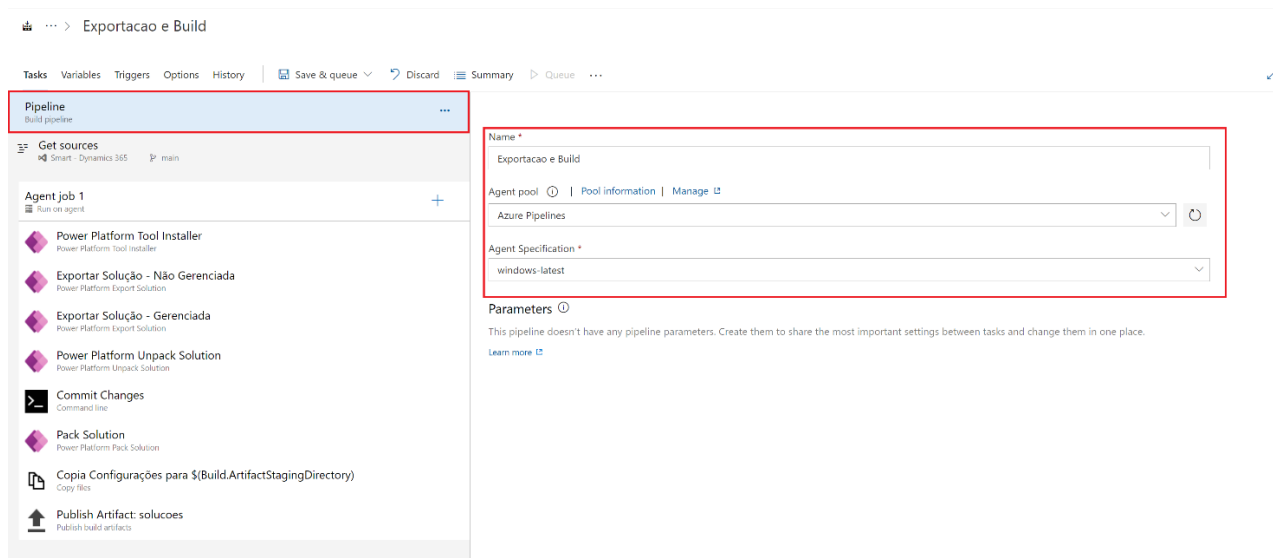
1. Acesse Pipelines > Pipelines
2. Clique nos três pontos verticais e selecione Import a pipeline



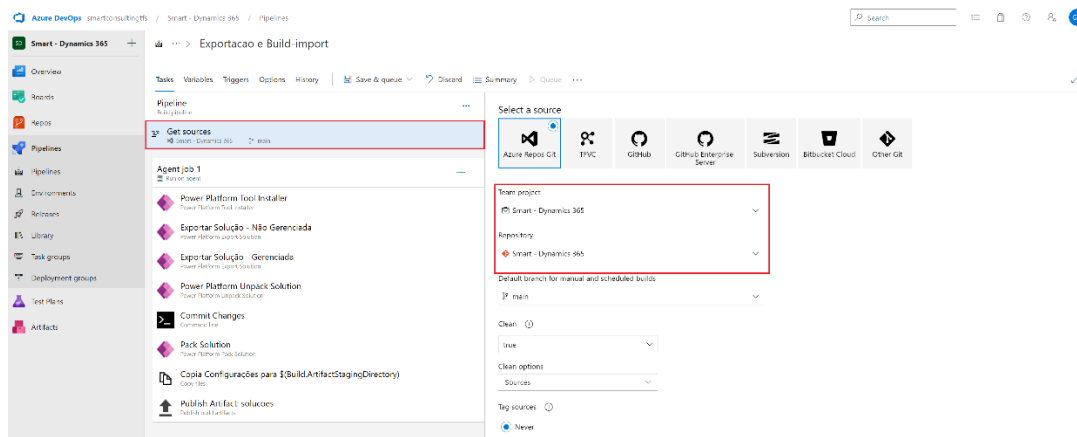
3. Selecione o arquivo Exportacao e Build.json e clique em Import



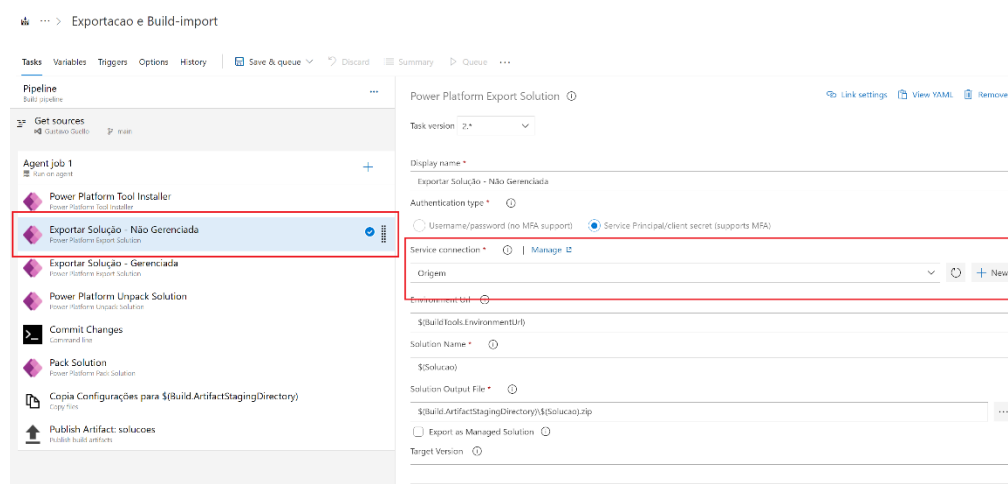
4. Selecione Pipeline e Selecione Azure Pipelines em Agent Pool e depois Windows-latest em Agent Specification. Altere o nome para Exportacao e Build.



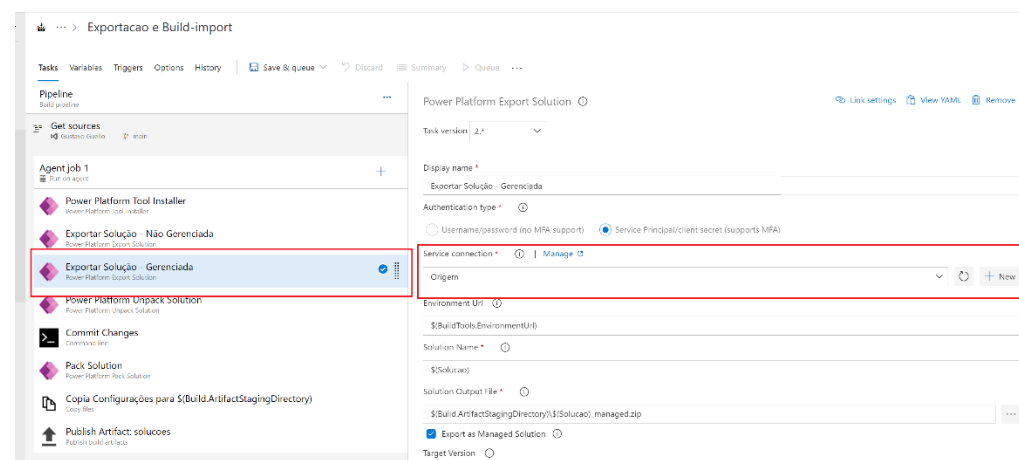
5. Clique em Get Sources e troque o Tem Project para o projeto atual



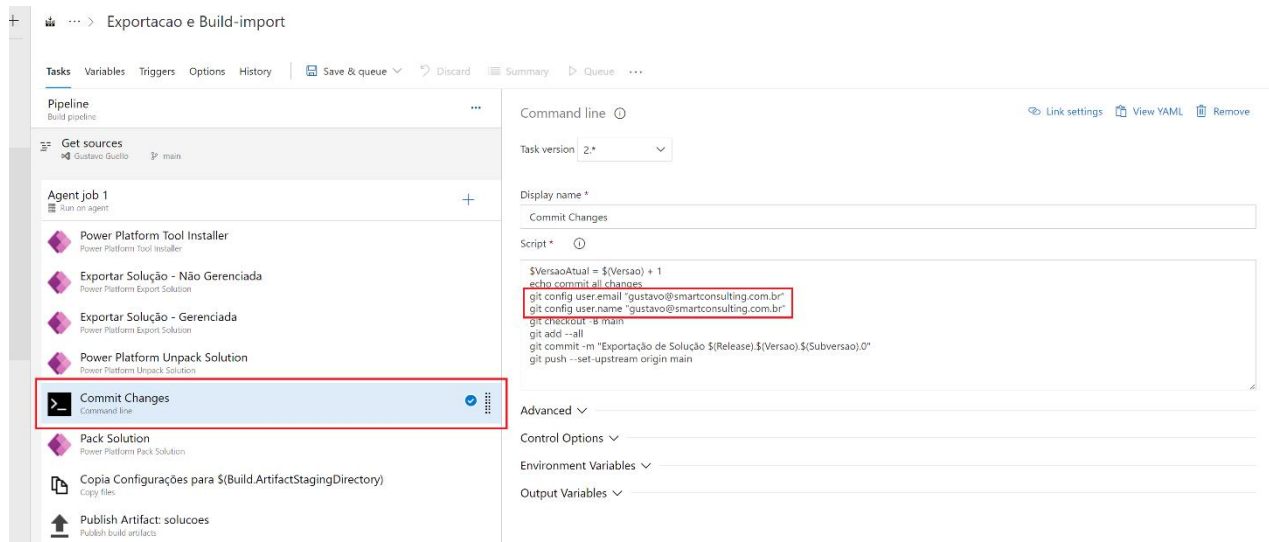
6. Clique em Exportar Solução – Não Gerenciada e selecione o Service Connection de Origem



7. Clique em Exportar Solução – Gerenciada e selecione o Service Connection de Origem

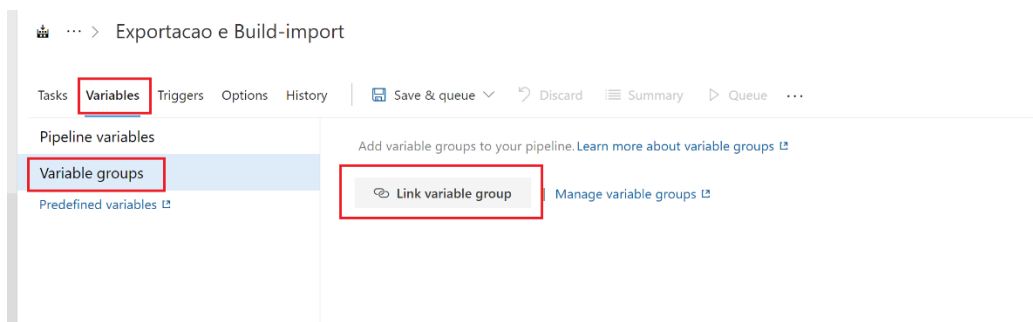


8. Selecione Commit Changes e altere o e-mail para o e-mail do seu usuário

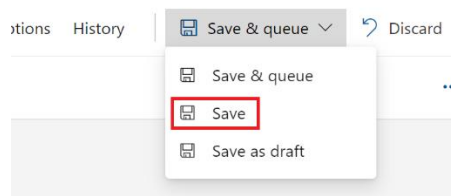


9. Clique na aba Variables > Variable groups e desvincule a variável ALM

10. Adicione o grupo de variáveis ALM do projeto



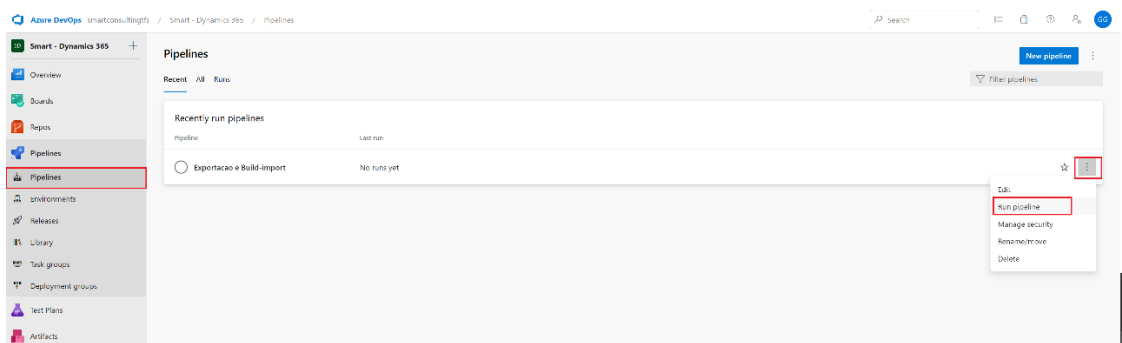
11. Clique em Save



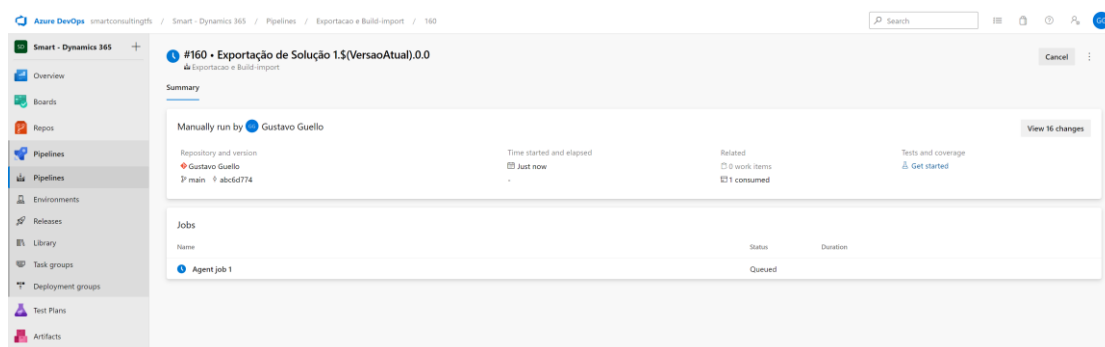
Rodando o pipeline

Vamos rodar o pipeline para exportar a solução, versionar e criar a solução para para instalada no ambiente de destino.

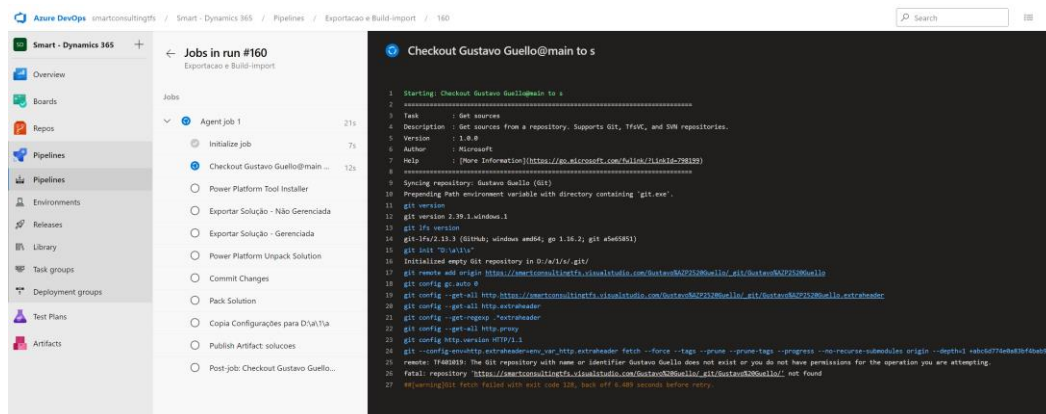
1. Acesse Pipelines > Pipelines
2. Selecione os 3 pontos do pipeline e selecione Run Pipeline



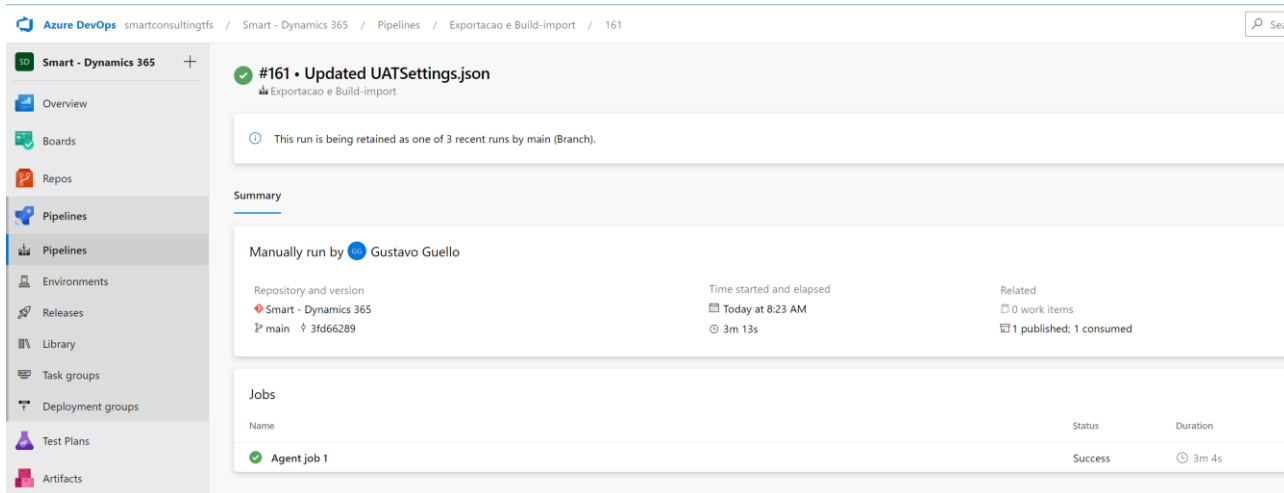
3. Acompanhe a execução



4. Clique em Agent job 1 para os detalhes da execução



5. Aguarde a conclusão do Pipeline



#161 • Updated UATSettings.json
Exportacao e Build-Import

This run is being retained as one of 3 recent runs by main (Branch).

Summary

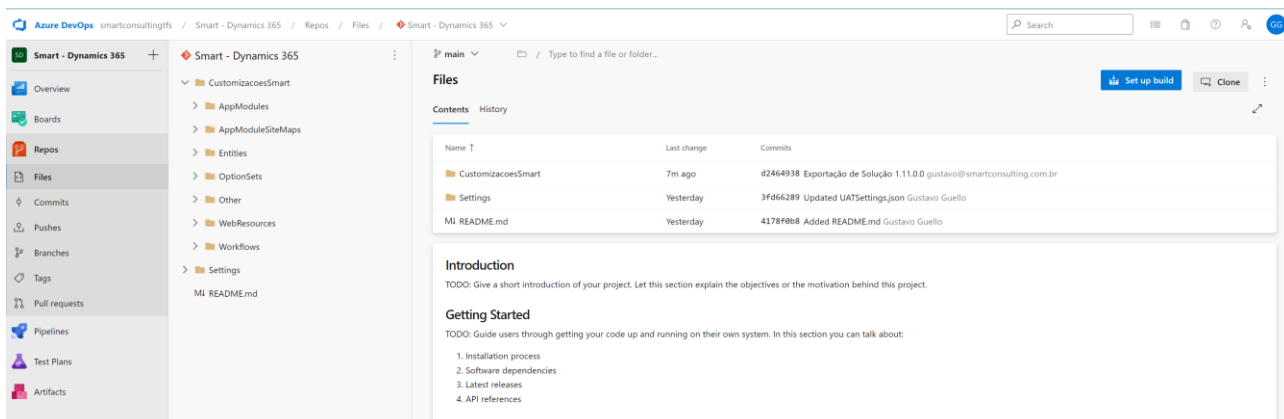
Manually run by Gustavo Guello

Repository and version	Time started and elapsed	Related
Smart - Dynamics 365 main 3fd66289	Today at 8:23 AM 3m 13s	0 work items 1 published; 1 consumed

Jobs

Name	Status	Duration
Agent job 1	Success	3m 4s

6. Será criada uma pasta no repositório de arquivos com os componentes da solução. A partir destes arquivos será possível gerar a solução gerenciada e não gerenciada e também comparar alterações entre as versões.



Files

Name	Last change	Commits
CustomizacoesSmart	7m ago	d2464938 Exportação de Solução 1.11.0.0 gustavo@smartconsulting.com.br
Settings	Yesterday	3fd66289 Updated UATSettings.json Gustavo Guello
MI README.md	Yesterday	4178f9b8 Added README.md Gustavo Guello

Introduction
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

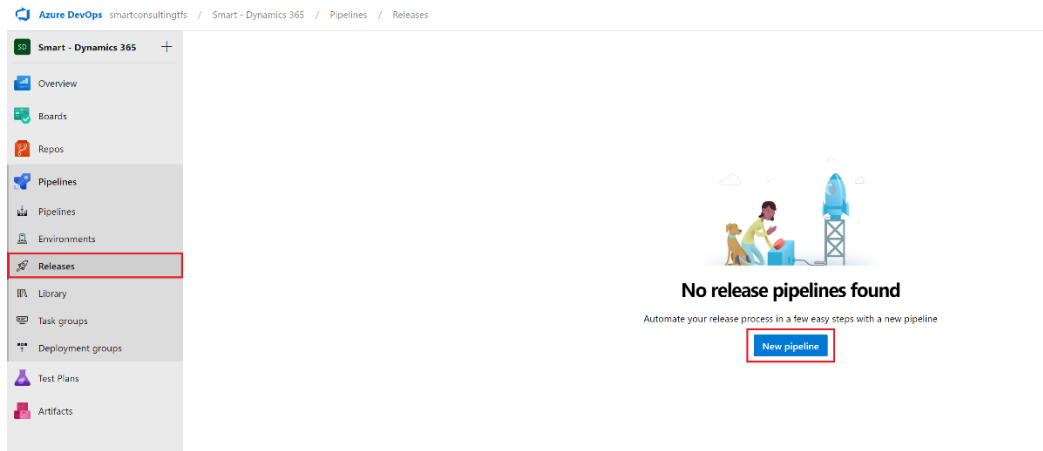
Getting Started
TODO: Guide users through getting your code up and running on their own system. In this section you can talk about:

1. Installation process
2. Software dependencies
3. Latest releases
4. API references

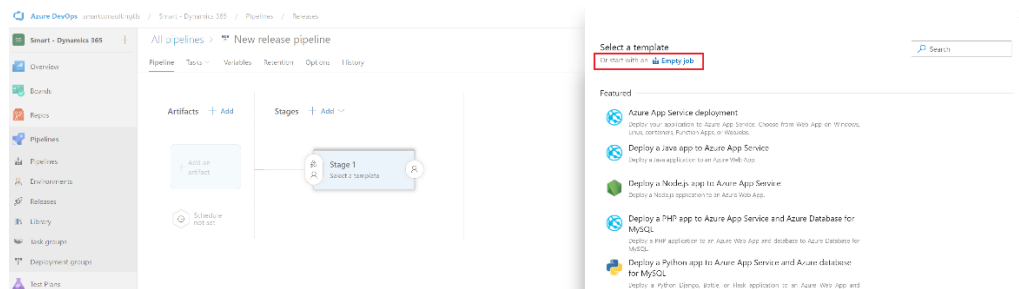
Criar o pipeline para publicação da solução

Demos criar um pipeline para recuperar a solução gerada no pipeline de build e importar no ambiente de destino. Poderão ser criados pipelines diferentes para cada ambiente (UAT e Produção).

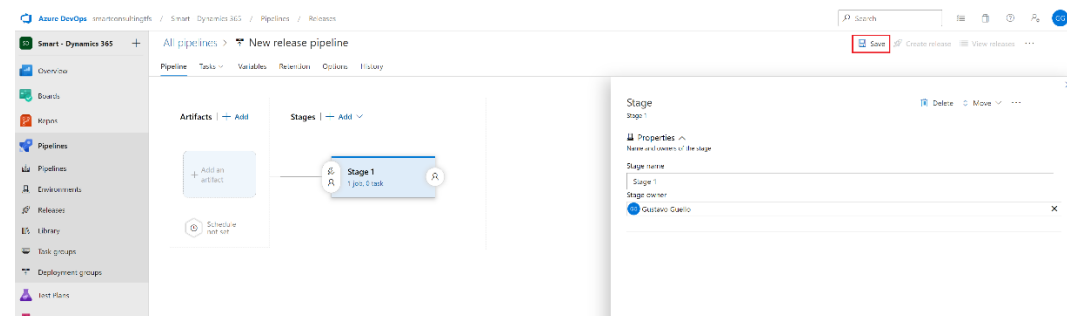
1. Acessar o menu de Releases e clicar em New Pipeline (será necessário criar um pipeline vazio para poder importar o modelo)



2. Selecione a opção start with an Empty job

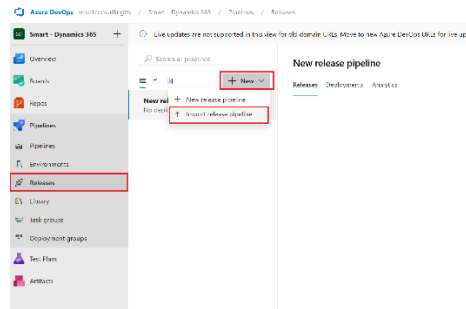


3. Clique em Save e depois OK

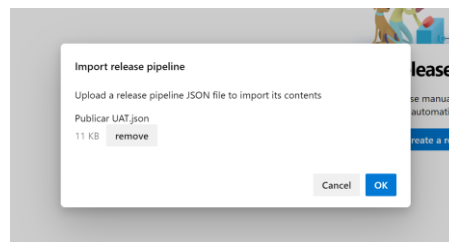


4. Volte para a tela de Releases

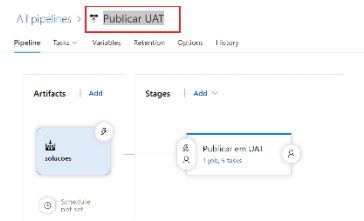
5. Clique em New e selecione import release pipeline



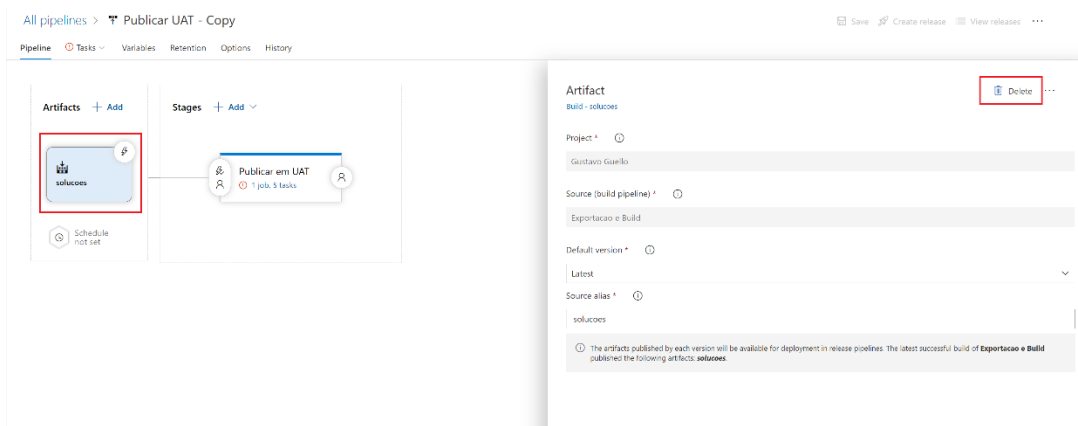
6. Selecione o arquivo Publicar UAT.json e clique OK



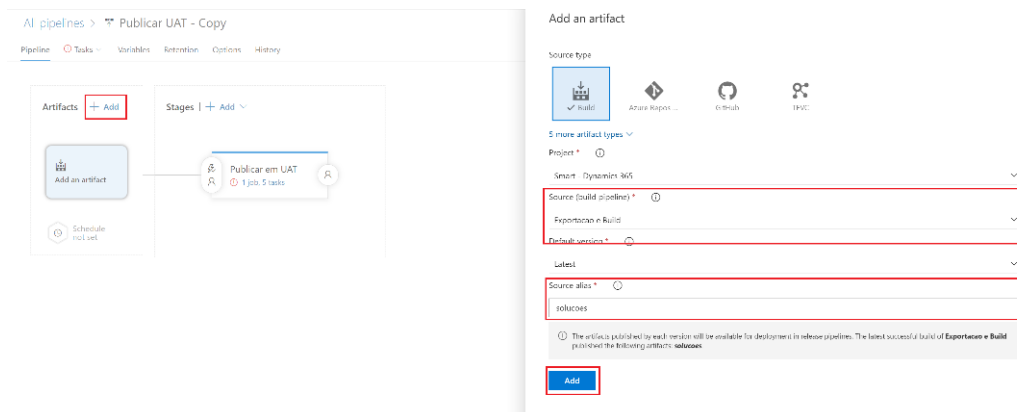
7. Altere o nome da Release para Publicar UAT



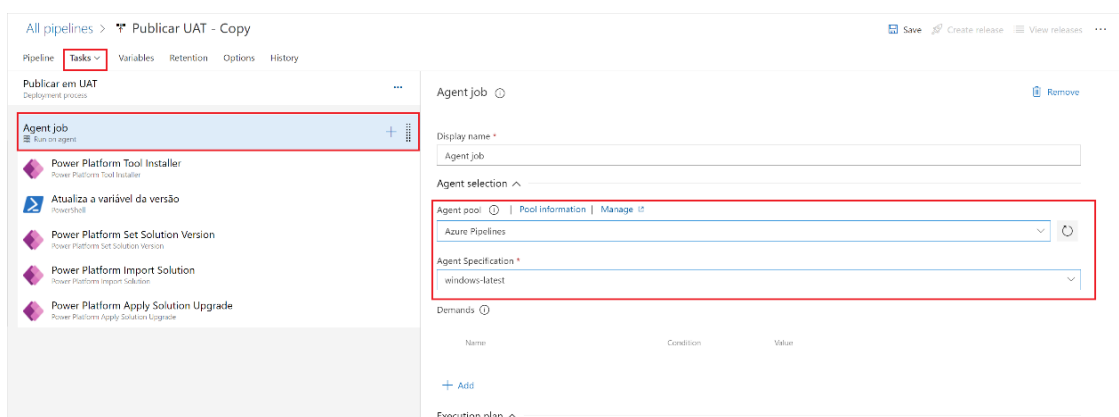
8. Em artifacts, clique em solucoes e exclua



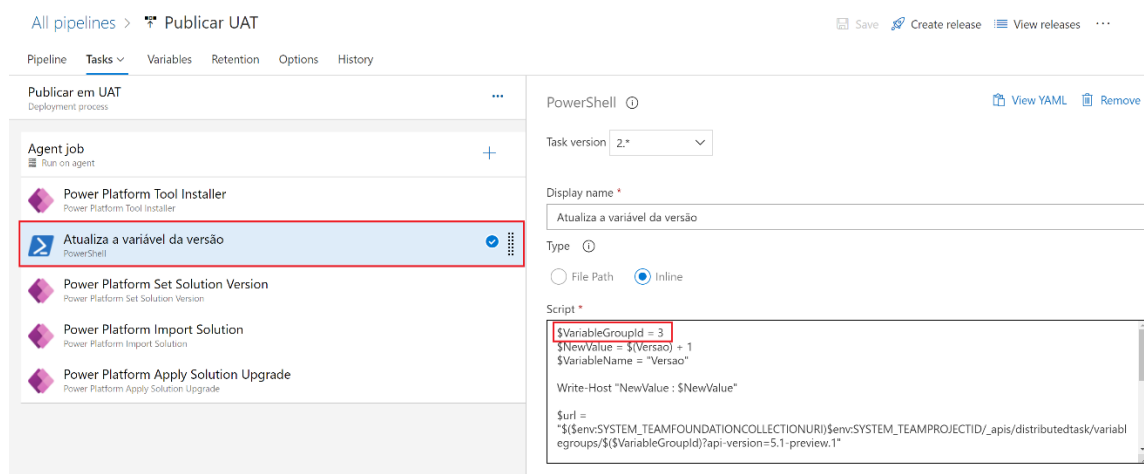
9. Clique em Add na caixa de Artifacts, Selecione o pipeline Exportacao e Build como Source e altere o Source Alias para solucoes. Clique em Add.



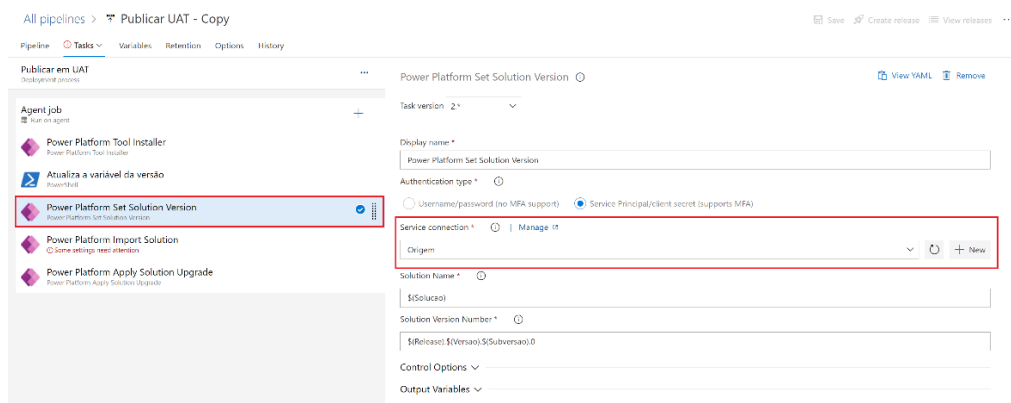
10. Clique na Aba Tasks, Selecione Agent job, Selecione Azure Pipelines e Windows-latest



11. Selecione Atualiza a variável da versão e altere o Id do Grupo de Variáveis



12. Selecione Power Platform Set Solution Version e selecione a conexão de **origem**



Power Platform Set Solution Version

Task version: 2.x

Display name: Power Platform Set Solution Version

Authentication type: ☐ Username/password (no MFA support) ☒ Service Principal/client secret (supports MFA)

Service connection: **Origem**

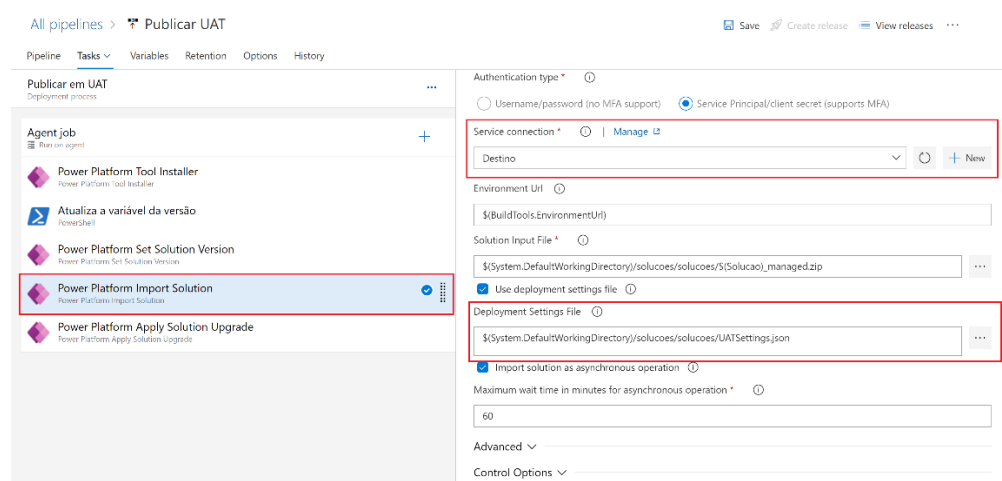
Solution Name: \$(Solucao)

Solution Version Number: \$(Release)\$(Versao)\$(Subversao).0

Control Options: ☐ Use deployment settings file

Output Variables: ☐ Import solution as asynchronous operation

13. Selecione Power Platform Import Solution e selecione a conexão de **destino** e ajuste o caminho do arquivo de configurações



Power Platform Import Solution

Authentication type: ☐ Username/password (no MFA support) ☒ Service Principal/client secret (supports MFA)

Service connection: **Destino**

Environment Url: \$(BuildTools.EnvironmentUrl)

Solution Input File: \$(System.DefaultWorkingDirectory)/solucoes/solucoes/\$(Solucao)_managed.zip

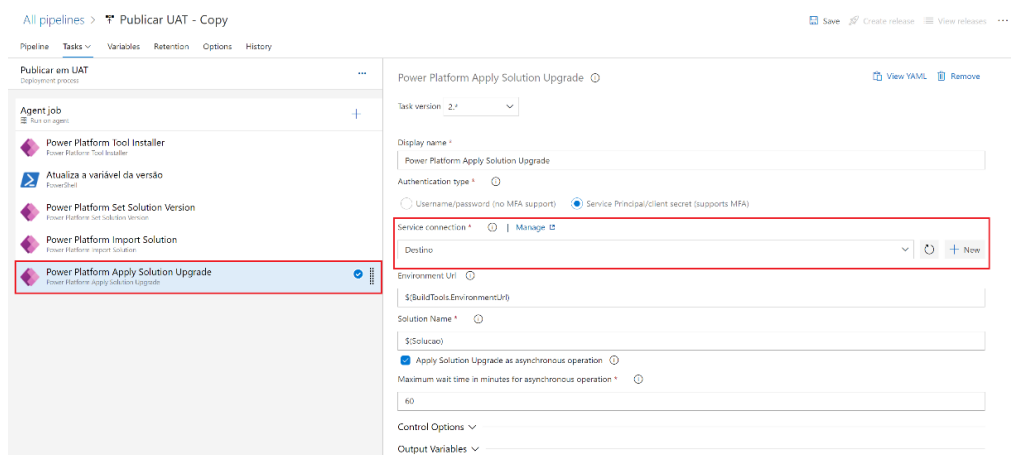
Deployment Settings File: \$(System.DefaultWorkingDirectory)/solucoes/solucoes/UATSettings.json

Maximum wait time in minutes for asynchronous operation: 60

Advanced: ☐ Import solution as asynchronous operation

Control Options: ☐ Use deployment settings file

14. Seleccione Power Platform Apply Solution Upgrade e selecione a conexão de **destino**



Power Platform Apply Solution Upgrade

Task version: 2.x

Display name: Power Platform Apply Solution Upgrade

Authentication type: ☐ Username/password (no MFA support) ☒ Service Principal/client secret (supports MFA)

Service connection: **Destino**

Environment Url: \$(BuildTools.EnvironmentUrl)

Solution Name: \$(Solucao)

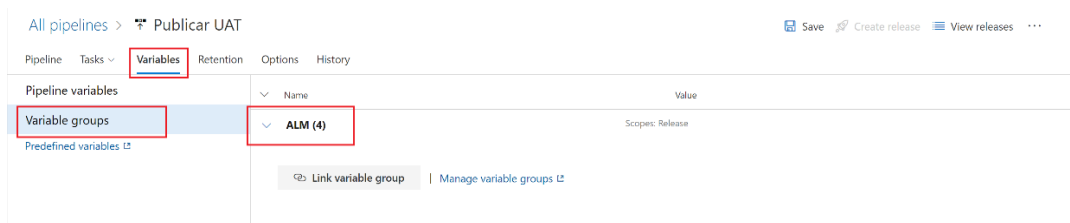
Apply Solution Upgrade as asynchronous operation: ☒

Maximum wait time in minutes for asynchronous operation: 60

Control Options: ☐ Use deployment settings file

Output Variables: ☐ Import solution as asynchronous operation

15. Selecione a Aba Variables > Variable groups e adicione o grupo de variáveis ALM

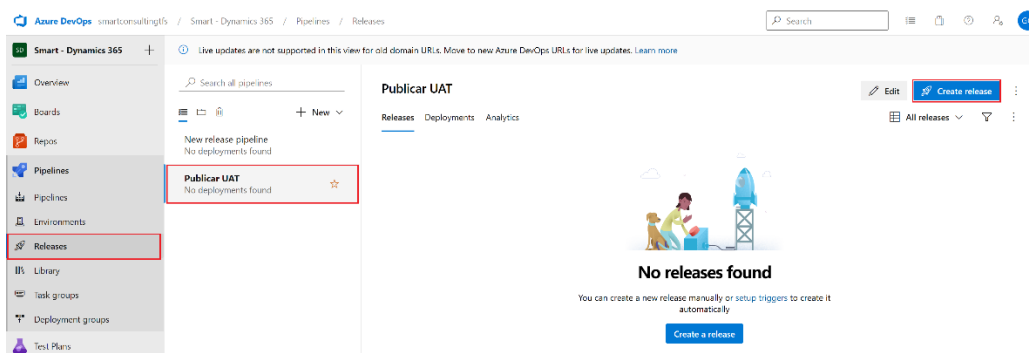


16. Clique em Save e OK

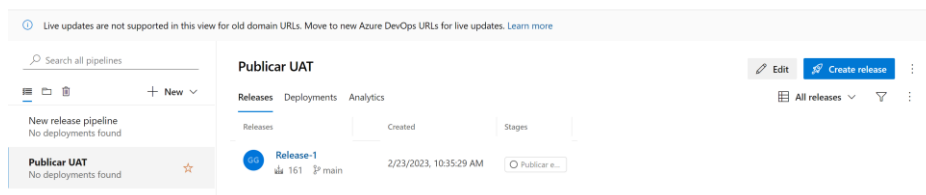
Testar o Deploy

Vamos executar o pipeline para importar a solução no ambiente de destino.

1. No menu Releases, selecione o Release Publicar UAT, clique em Create Release e depois Create



2. Aguarde a conclusão do Release

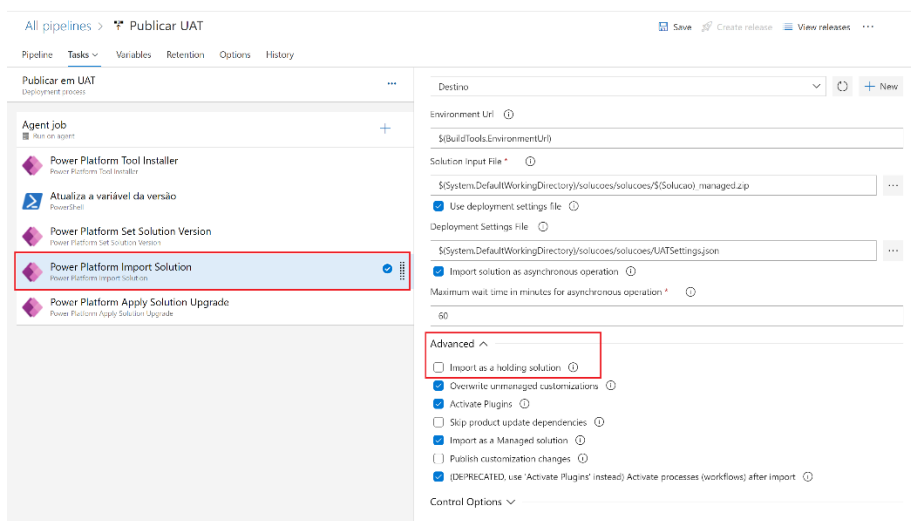


OBS: Os fluxos do Power Automate novos deverão ser ativados manualmente no ambiente de destino. Não é necessário editar o fluxo pois as conexões já estarão corretas. Este processo deverá ser feito sempre que for criado um novo fluxo.

Primeiro deploy da solução

O primeiro deploy da solução exige algumas etapas adicionais.

1. Rode o pipeline de Exportação e Build para gerar a primeira solução
2. Altere o pipeline de release para alterar a função de importação da solução desmarcando a opção de Import as a holding Solution.



3. Rode a Release. Irá apresentar erro na etapa de upgrade mas a solução terá sido publicada.
4. Marque novamente a opção Import as a holding Solution
5. Rode o pipeline de Exportação e Build para gerar uma nova versão
6. Rode novamente o Release