

## チャットサービス連携サンプルシナリオ(Teams)

WinActorから、チャットサービス「Teams」に対して通知やファイル送信を行い、Teamsからの応答を受信するサンプルシナリオです。

WinActorとチャットサービス「Teams」間でやり取りをしたいユーザ向けのサンプルシナリオです。  
本サンプルシナリオを利用することで、Teamsに対してメッセージの通知やファイル送信を行い、Teams上のユーザの応答をWinActorに返却することが可能です。

### サンプルシナリオ概要

#### 目次

はじめに

本サンプルシナリオの概要

使用するWinActorモジュール

本サンプルシナリオの想定レベル

動作確認環境

制約事項

本サンプルシナリオの使い方

Step1.資材の準備

Step2.連携アプリの登録

Step3.チャット接続設定

Step4.チャット内容設定

Step5.実行シナリオの実行

トラブルシューティング

サンプルシナリオ解説

1.Teamsからの応答結果の受け取り

2.画像ファイルの送信

3. メッセージ通知の追加

変更履歴

### はじめに

チャットサービス「Teams」に対してメッセージの通知やファイル送信を行い、Teamsからの応答を受信するサンプルシナリオについて説明します。(以降、本サンプルシナリオと呼びます)

## 本サンプルシナリオの概要

本サンプルシナリオは、以下の処理を行います。

チャットサービス連携シナリオ(Teams通知のみ).ums7

WinActorからTeamsへメッセージ通知を行います。

チャットサービス連携シナリオ(Teams通知&ユーザ応答).ums7

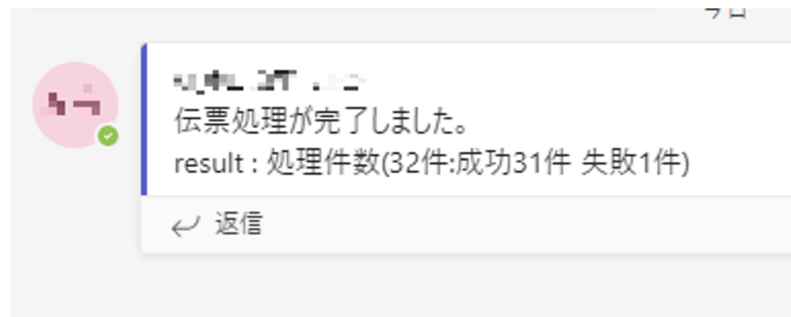
WinActorからTeamsへメッセージ通知を行います。

TeamsからWinActorへ応答結果を返却します。

チャットサービス連携シナリオ(Teams通知&ファイル送信).ums7

WinActorからTeamsへメッセージ通知を行います。

WinActorからTeamsへファイル送信を行います。



### Teamsへのメッセージ通知の例

## 使用するWinActorモジュール

本サンプルシナリオでは、以下のWinActorモジュールを使用します。

チャットサービス連携/チャット接続設定(Teams)

チャットサービス連携/チャット送信(通知のみ)

チャットサービス連携/チャット送信(通知&応答)

チャットサービス連携/チャット送信(通知&ファイル送信)

## 本サンプルシナリオの想定レベル

本ドキュメントは、以下の技術レベルのユーザを対象としています。

Teamsの使い方について理解している。

## 動作確認環境

本サンプルシナリオは以下の環境で動作確認しています。

Windows 10、WinActor 7.4.0、Teams

## 制約事項

本サンプルシナリオを実行するためには、WinActorからTeamsにアクセスするためのインターネット環境が必要です。

本サンプルシナリオは、WinActor 7.3.0 より古いバージョンでは動作しません。

本サンプルシナリオを実行するためには、Chromeのインストールが必要です。(WebDriverの更新も必要です。)

WinActorがインストールされた端末1台につき、連携アプリは1つのみ登録してご利用ください。

複数の連携アプリを用いて本サンプルシナリオを実行すると正常に動作しなくなります。

WinActorとTeams間で一度に送受信できるメッセージは、最大1024文字です。

Teamsに対して送ることができるファイルの種類は、.jpg、.jpeg、.pngの3種類の画像ファイルです。

Teamsからのユーザ応答をWinActorに返却した際に、返却された値に不要な文字(HTMLのタグ文字など)が含まれる場合があります。

## 本サンプルシナリオの使い方

本サンプルシナリオの実施方法は以下の通りです。

Teamsへのメッセージ通知のみを行うシナリオ「チャットサービス連携シナリオ(Teams通知のみ).ums7」を例に説明します。

### Step 1. 資材の準備

ダウンロードしたZIPファイルを展開し、資材を以下に配置します。

#### 「チャットサービス連携」フォルダ

{ドキュメント}¥WinActor¥libraries へ配置してください。

#### 「サンプルシナリオ」、「マニュアル」フォルダ

任意のフォルダへ配置してください。

### Step 2. 連携アプリの登録

お手元の環境が、動作確認環境に記述されている環境と同等の環境であるかご確認ください。

連携アプリを登録します。

・登録方法はStep1で展開した「マニュアル」フォルダ内の「**連携アプリ設定手順書.pdf**」の「3. 設定手順(Teams)」に従ってください。

・連携アプリの登録時に生成された以下の情報をメモしてください。

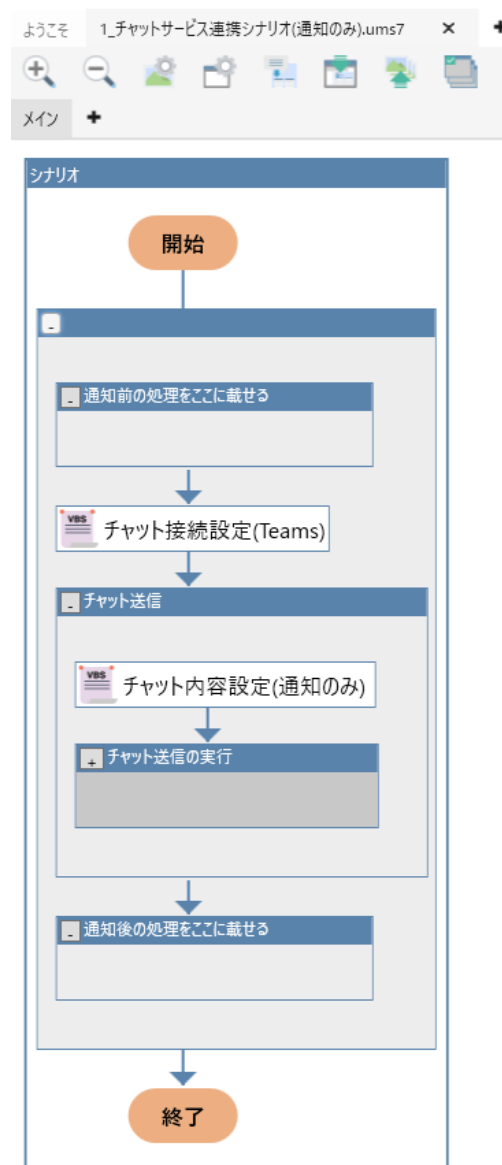
「チームID」「チャンネルID」「アプリケーション(クライアント)ID」「クライアントシークレット」「ディレクトリ(テナント)ID」

### Step 3.チャット接続設定

Teamsに対して、「チャットサービス連携シナリオ(Teams通知のみ).ums7」を実行するために必要となるチャット接続設定を行います。

1.WinActorを起動します。

2.「サンプルシナリオ」フォルダ内の「実行シナリオ>チャットサービス連携シナリオ(Teams通知のみ).ums7」を開きます。



**チャットサービス連携シナリオ画面**

3.「チャット接続設定(Teams)」ノードをダブルクリックし、プロパティ画面を立ち上げます。

プロパティ

スクリプト実行

名前 チャット接続設定(Teams)

コメント

設定 スクリプト 注釈

Teamsにユーザ通知するためのチャット接続情報を設定します。

- ・アプリケーション(クライアント)ID : Teamsアプリのアプリケーション(クライアント)IDを入力してください。
- ・ディレクトリ(テナント)ID : Teamsアプリのディレクトリ(テナント)IDを入力してください。
- ・クライアントシークレット : Teamsアプリのクライアントシークレットを入力してください
- ・チームID : 通知先チャネルのチームIDを入力してください。
- ・チャネルID : 通知先チャネルのチャネルIDを入力してください。
- ・WinActorオプションのプロキシ設定を使用する :  
インターネット接続にWinActor オプションのプロキシ設定を利用する場合、  
「使用する」を選択してください。

---

※以下の項目の設定は任意となります。  
未設定の場合、シナリオの初回実行時に以下の項目の入力を"手動"で求められます。

- ・MS365アカウントのメールアドレス :  
メッセージの投稿を行う、Microsoft 365のアカウントのメールアドレスを入力してください。
- ・MS365アカウントのパスワード :  
メッセージの投稿を行う、Microsoft 365のアカウントのパスワードを入力してください。
- ・プロキシ認証のユーザ名 :  
プロキシを使ったインターネット接続でユーザ認証が必要な場合、ユーザ名を入力してください。
- ・プロキシ認証のパスワード :  
プロキシを使ったインターネット接続でユーザ認証が必要な場合、パスワードを入力してください。

【必須】アプリケーション(クライアント)ID 値⇒

【必須】ディレクトリ(テナント)ID 値⇒

【必須】クライアントシークレット 値⇒

【必須】チームID 値⇒

【必須】チャネルID 値⇒

WinActorオプションのプロキシ設定を使用する 使用しない

MS365アカウントのメールアドレス 値⇒

MS365アカウントのパスワード 値⇒

プロキシ認証のユーザ名 値⇒

プロキシ認証のパスワード 値⇒

更新 元に戻す

### 「チャット接続設定(Teams)」のプロパティ

4. Step 2でメモした項目を入力し「更新ボタン」を押し、シナリオ画面に戻ります。

- ・【必須】アプリケーション(クライアント)ID
- ・【必須】ディレクトリ(テナント)ID
- ・【必須】クライアントシークレット
- ・【必須】チームID
- ・【必須】チャネルID
- ・WinActorオプションのプロキシ設定を使用する場合は、「使用する」を選択してください。

※以下の項目の設定は任意となります。

未設定の場合、シナリオの初回実行時に以下の項目の入力を"手動"で求められます。

- ・MS365アカウントのメールアドレス
- ・MS365アカウントのパスワード
- ・プロキシ認証のユーザ名
- ・プロキシ認証のパスワード

## Step 4.チャット内容設定

「チャットサービス連携シナリオ(Teams通知のみ).ums7」を実行するために必要となるチャット内容設定を行います。

1. 「チャット内容設定(通知のみ)」ノードをダブルクリックし、プロパティ画面を立ち上げます。

プロパティ

スクリプト実行

名前 チャット内容設定(通知のみ)

コメント

設定 スクリプト 注釈

ユーザ通知するメッセージ内容を設定します。

・通知メッセージ : チャットサービスに通知するメッセージを入力してください。  
・メンション対象のアドレス : メンションしたい相手のメールアドレスを入力してください。

【必須】通知メッセージ 値 →

メンション対象のアドレス 値 →

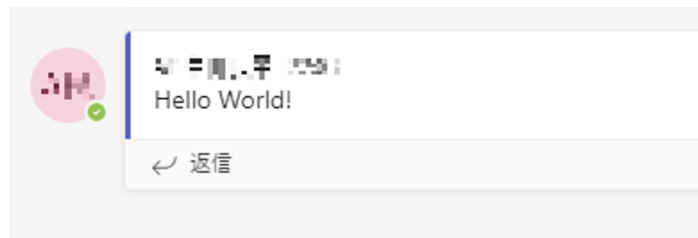
更新 元に戻す

### 「チャット内容設定(通知のみ)」のプロパティ

2. 「通知メッセージ」に「Hello World!」と入力し「更新ボタン」を押し、シナリオ画面に戻ります。  
・メンションをしたい場合は、メンションしたいユーザーに登録されたメールアドレスを入力してください。

## Step 5.実行シナリオの実行

1. 「チャットサービス連携シナリオ(Teams通知のみ).ums7」をシナリオ実行します。
2. Teamsのチャンネルに、「Hello World!」のメッセージが通知されます。



### Teamsに通知されたメッセージ

- ・グループ「通知前の処理をここに載せる」には、通知前に実行したい任意のノードを配置してください。
- ・グループ「通知後の処理をここに載せる」には、通知後に実行したい任意のノードを配置してください。

## トラブルシューティング

本サンプルシナリオでエラーが発生した場合、以下を確認してください。

詳細については、エラーメッセージ一覧 ([https://winactor.biz/samplescenario/errmsg\\_24277.html](https://winactor.biz/samplescenario/errmsg_24277.html))を参照してください。

## サンプルシナリオ解説

本サンプルシナリオで使用している実装テクニックについて説明します。

### 1.Teamsからの応答結果の受け取り

Teamsからユーザの応答を受け取りたい場合、シナリオ「チャットサービス連携シナリオ(Teams通知&ユーザ応答).ums7」を利用します。

- ・Step1～Step3までの手順は「チャットサービス連携シナリオ(Teams通知のみ).ums7」と共通です。
- ・Step4のプロパティ画面は以下の通り異なります。プロパティ画面の説明に従って設定してください。

### 「チャット内容設定(通知&応答)」のプロパティ

ユーザ通知するメッセージ内容と、ユーザの応答結果を入れる変数を設定します。

- ・通知メッセージ:チャットサービスに通知するメッセージを入力してください。
- ・メンション対象のアドレス:メンションしたい相手のメールアドレスを入力してください。
- ・応答期限:メッセージの投稿後、ユーザの応答受けを終了させる時間を入力してください。5～1440の値が設定できます。
- ・スリープ間隔:入力した値が小さいほど、WinActorにユーザの応答結果を反映させる時間が短くなります。1～300の値が設定できます。
- ・値を入れる変数:通知メッセージに対する応答として、ユーザがチャットサービスに投稿したテキストを入れる変数を設定します。

## 2.画像ファイルの送信

Teamsに画像ファイル送信を行いたい場合、シナリオ「チャットサービス連携シナリオ(Teams通知&ファイル送信).ums7」を利用します。

- ・Step1～Step3までの手順は「チャットサービス連携シナリオ(Teams通知のみ).ums7」と共通です。
- ・Step4のプロパティ画面は以下の通り異なります。プロパティ画面の説明に従って設定してください。



プロパティ

スクリプト実行

名前 チャット内容設定(通知&ファイル送信)

コメント

設定 スクリプト 注釈

ユーザ通知するメッセージ内容と画像ファイルを設定します。

- ・通知メッセージ : チャットサービスに通知するメッセージを入力してください。
- ・メンション対象のアドレス : メンションしたい相手のメールアドレスを入力してください。
- ・送信する画像ファイルのパス : チャットサービスに投稿する画像ファイルのパスを入力してください。

【必須】通知メッセージ 値 →

メンション対象のアドレス 値 →

送信する画像ファイルのパス 値 → ...

更新 元に戻す

### 「チャット内容設定(通知&ファイル送信)」のプロパティ

ユーザ通知するメッセージ内容と画像ファイルを設定します。

- ・通知メッセージ:チャットサービスに通知するメッセージを入力してください。
- ・メンション対象のアドレス:メンションしたい相手のメールアドレスを入力してください。
- ・送信する画像ファイルのパス:チャットサービスに投稿する画像ファイルのパスを入力してください。

### 3. メッセージ通知の追加

本サンプルシナリオで、2回以上のメッセージ通知を行いたい場合は以下を実施して下さい。

1. 「チャット送信(通知のみ)」をシナリオフロー図内にドラッグ&ドロップして配置します。
- ・ユーザからの応答を受け取りたい場合は、代わりに「チャット送信(通知&応答)」を配置して下さい。
  - ・ファイル送信を行いたい場合は、代わりに「チャット送信(通知&ファイル送信)」を配置して下さい。



### 「チャット送信(通知のみ)」の配置

2. 「変数名重複」のメッセージが表示されたら、左のチェックをすべてOFFにし、「OKボタン」を押してメッセージを閉じます。

- ・チェックがONの状態で「OKボタン」を押すと誤った設定がされ、正常にメッセージ通知ができなくなります。
- ・その場合は、配置した「チャット送信(通知のみ)」を一度削除して、再配置してください。

変数名重複

×

以下の変数名が重複しているため、リネームされます。  
重複している変数名を変えずにそのまま使用する場合はチェックを外してください。  
リネーム後の変数名は修正可能です。  
※チェックを外した場合は変数の値が上書きされます。

<input type="checkbox"/>	No	重複した変数名	リネーム後の変数名
<input type="checkbox"/>	1	【チャット連携】連携チャットサービス種別	【チャット連携】連携チャットサービス種別_1
<input type="checkbox"/>	2	【チャット連携】アプリ情報	【チャット連携】アプリ情報_1
<input type="checkbox"/>	3	【チャット連携】チャンネル情報	【チャット連携】チャンネル情報_1
<input type="checkbox"/>	4	【チャット連携】プロキシ利用の有無	【チャット連携】プロキシ利用の有無_1
<input type="checkbox"/>	5	【チャット連携】アカウント情報	【チャット連携】アカウント情報_1
<input type="checkbox"/>	6	【チャット連携】プロキシ認証情報	【チャット連携】プロキシ認証情報_1
<input type="checkbox"/>	7	【チャット連携】トークン自動発行フラグ	【チャット連携】トークン自動発行フラグ_1

OK

キャンセル

「変数名重複」メッセージ

3. 配置した「チャット送信(通知のみ)」の左上にある「+」を押して、展開します。
4. Step 4の手順を実施します。
- ・以上でメッセージ通知の追加が完了しました。

変更履歴

版数
1.0版
日付
2024/2/13
修正内容
初版