

## チャットサービス連携サンプルシナリオ(Slack)

WinActorから、チャットサービス「Slack」に対して通知やファイル送信を行い、Slack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「Slack」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、Slackに対してメッセージの通知やファイル送信を行い、Slack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.Slack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「Slack」に対してメッセージの通知やファイル送信を行い、Slack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(Slack通知のみ).ums7

WinActorからSlack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(Slack通知&ユーザ応答).ums7

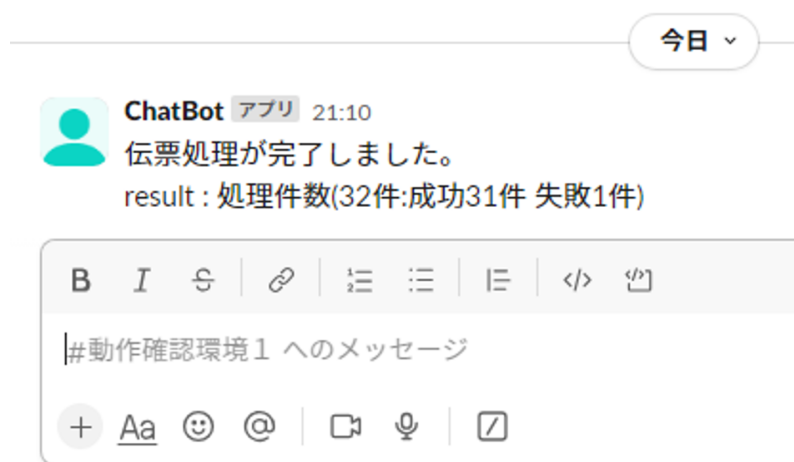
WinActorからSlackへメッセージ通知を行います。

Slack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(Slack通知&ファイル送信).ums7

WinActorからSlackへメッセージ通知を行います。

WinActorからSlackへファイル送信を行います。



### Slack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(Slack)

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。

Slack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、Slack

## 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Slack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を実行すると正常に動作しなくなります。

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

Slackに対して送ることができるファイルの種類は、.jpg、.jpeg、.png、.gifの4種類の画像ファイルです。

## 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。

Slack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(Slack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」、「アプリ設定」、「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「2. 設定手順(Slack)」に従ってください。

・上記の手順書ではStep 1で展開した「アプリ設定」フォルダ内の「manifest.json」を使用します。

・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。

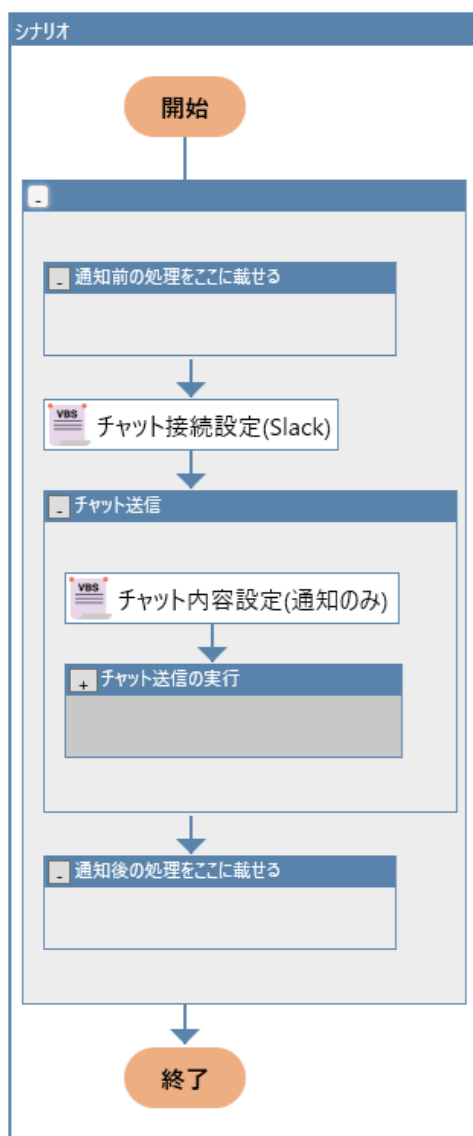
「アクセストークン」「チャンネルID」

### Step 3.チャット接続設定

Slackに対して、「チャットサービス連携シナリオ(Slack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(Slack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(Slack)」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。

The screenshot shows a 'Script Execution' dialog box with the title 'プロパティ' (Property) and subtitle 'プロパティ' (Property). The main title is 'スクリプト実行' (Script Execution). The dialog has a '名前' (Name) field containing 'チャット接続設定(Slack)' (Chat Connection Settings (Slack)) and a 'コメント' (Comment) field. Below these are three tabs: '設定' (Settings), 'スクリプト' (Script), and '注釈' (Notes). The '設定' (Settings) tab is active, showing instructions for setting up Slack user notifications. The instructions include: 'Slack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。' (Set chat connection information for user notifications on Slack.), '・アクセストークン：Slackアプリのアクセストークンを入力してください。' (Token: Enter the Slack app's access token.), '・チャンネルID：通知先チャンネルのチャンネルIDを入力してください。' (Channel ID: Enter the channel ID of the notification destination channel.), and '・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。' (Use WinActor option proxy settings: If you want to use WinActor option proxy settings for internet connection, select 'Use'). There are three dropdown menus: '【必須】アクセストークン' (Required Access Token) with '値' (Value) selected, '【必須】チャンネルID' (Required Channel ID) with '値' (Value) selected, and 'WinActorオプションのプロキシ設定を使用する' (Use WinActor option proxy settings) with '使用しない' (Do not use) selected. At the bottom are '更新' (Update) and '元に戻す' (Reset) buttons.

### 「チャット接続設定(Slack)」のプロパティ

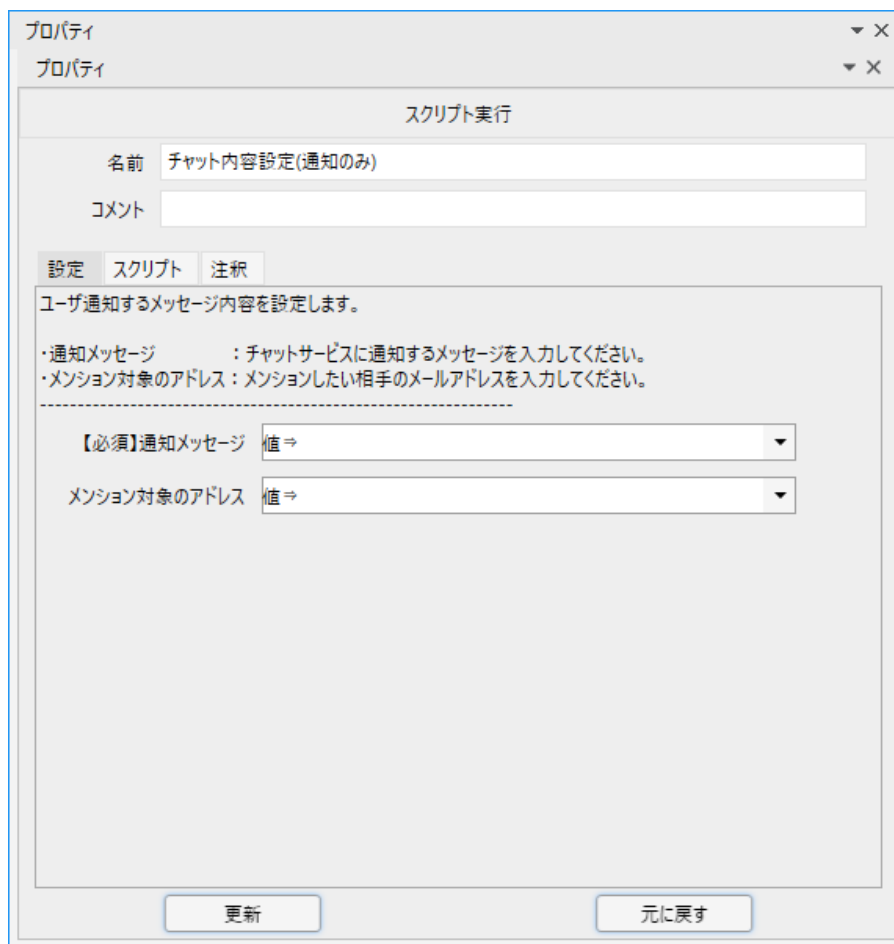
4. Step 2でメモした項目を入力し「更新ボタン」を押し、シナリオ画面に戻ります。

- ・【必須】アクセストークン
- ・【必須】チャンネル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。

## Step 4.チャット内容設定

「チャットサービス連携シナリオ(Slack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。



### 「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(Slack通知のみ).ums7」をシナリオ実行します。
2. Slackのチャンネルに、「Hello World!」のメッセージが通知されます。



### Slack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\\_24276.html](https://winactor.biz/samplescenario/errmsg_24276.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.Slackからの応答結果の受け取り

Slack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(Slack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(Slack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  
プロパティ

スクリプト実行

名前: チャット内容設定(通知&応答)

コメント:

設定 スクリプト 注釈

【必須】通知メッセージ: 値 →

メンション対象のアドレス: 値 →

【必須】応答期限(分): 値 → 5

【必須】スリープ間隔(秒): 値 → 1

応答用ボタン1: 変数に固定値を入れて継続

応答用ボタン1: ボタン表示名: 値 →

応答用ボタン1: 値を入れる変数: 【チャット連携】未選択

応答用ボタン1: 入れる値: 値 →

応答用ボタン2: 未設定

応答用ボタン2: ボタン表示名: 値 →

応答用ボタン2: 値を入れる変数: 【チャット連携】未選択

更新 元に戻す

## 「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、ユーザ応答するボタンを設定します。

- ・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、ボタン応答の受け付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にボタン応答の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。
  - ・応答用ボタン:
    - メッセージの投稿後に表示するボタンの動作を下記の中から選んでください。(最大6個まで設定できます)
    - ・変数に固定値を入れて継続: ボタンをクリック後、「値を入れる変数」に「入れる値」を設定します。
    - ・変数に任意のテキストを入れて継続: ボタンをクリック後、チャットサービスに投稿したテキストを「値を入れる変数」に設定します。
    - ・未設定: ボタンとして表示されません。
- ※「ボタン表示名」に設定した値が、ボタン上に表示されます。

## 2.画像ファイルの送信

Slack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(Slack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(Slack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  
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

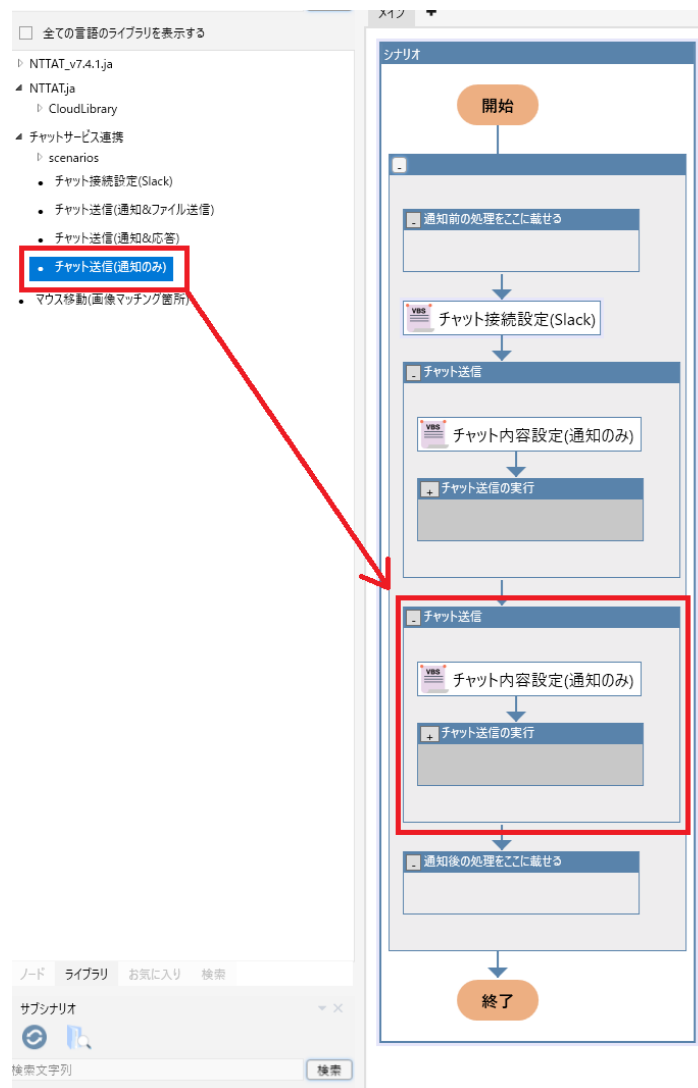
ユーザ通知するメッセージ内容と画像ファイルを設定します。

- ・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