

# Silk Test 21.0

Silk4NET クイックスタート チュー  
トリアル

Micro Focus  
The Lawn  
22-30 Old Bath Road  
Newbury, Berkshire RG14 1QN  
UK  
<http://www.microfocus.com>

© Copyright 1992-2022 Micro Focus またはその関連会社。

MICRO FOCUS、Micro Focus のロゴおよび Silk Test は、Micro Focus またはその関連会社の商標または登録商標です。

その他、記載の各名称は、各所有社の知的所有財産です。

2022-10-25

# 目次

Silk4NET .....	4
Silk4NET 入門 .....	5
Silk4NET プロジェクトの作成 .....	6
プロジェクトへの Silk4NET テストの追加 .....	8
Silk4NET テストの記録 .....	10
Silk4NET テストの実行 .....	12
TrueLog の有効化 .....	13
テスト結果の分析 .....	14

# Silk4NET

Silk4NET は、Microsoft Visual Studio 用の Silk Test プラグインです。Silk4NET を使用すると、機能テスト、回帰テスト、およびローカリゼーション テストの作成と管理を Visual Studio で直接、効率的に行うことができます。Silk4NET で実行できる Visual Studio 内の作業は、以下のとおりです。

- テストを記録します。
- Visual Basic .NET を使用してテストを開発します。
- C# を使用してテストを開発します。
- Microsoft テスト環境内のテスト計画の一環としてテストを実行できます。
- ビルドプロセスの一環としてテストを実行できます。
- テスト結果を表示します。

テスト済みの Visual Studio のバージョンについての情報は、『[リリースノート](#)』を参照してください。

Silk4NET は、アプリケーション テクノロジーのさまざまなセットのテストをサポートしています。

新しい機能、サポートするプラットフォーム、テスト済みのバージョンについての情報は、『[リリースノート](#)』を参照してください。

Silk4NET は、複雑なテストに適用した場合にも自動化の利点を実現されるように設計されており、開発者が使い慣れた環境に直接、テストの自動化機能を追加するため、テスト アプリケーションに加えられた変更が簡単に対処できます。また、Silk4NET の強力なテスト フレームワークを使用すると、複数のテスト プロジェクトに対してテストの再利用が促進されるため、より高い投資収益率 (ROI) を達成できます。テストスイートの構築と保守に要する時間が短縮されるため、QA 担当者はテスト範囲を拡張し、アプリケーション品質を最適化することができます。



**注:** Silk4NET の起動時に開始画面を表示しないように選択している場合、**ヘルプ > 製品更新の確認** をクリックして利用可能な更新を確認できます。

# Silk4NET 入門

Silk4NET では、自動機能テストや回帰テストの作成、実行を行うことができます。Silk4NET を使用するには、次の操作を実行します。

1. Silk4NET プロジェクトの作成
2. 操作やキーワードを記録するか、手動でテストを記述して、自動 Silk4NET テストを作成します。
3. Silk4NET テストをプロジェクトに追加します。プロジェクトには記録されたテストおよび、手動でスクリプトを作成したテストを含めることができます。
4. テストを実行します。
5. テスト結果を分析します。

# Silk4NET プロジェクトの作成

1. **Silk4NET > 新しいプロジェクト** または **ファイル > 新しいプロジェクト** をクリックします。**新しいプロジェクト** ダイアログ ボックスが表示されます。
2. **インストール済み > テンプレート** で、**Visual Basic** または **Visual C#** をクリックし、**テスト** を選択してから **Silk4NET プロジェクト** を選択します。
3. プロジェクトの名前を **名前** フィールドに入力します。  
チュートリアルでは、「Tutorial」と入力します。
4. 省略可能：ソリューションの名前を **ソリューション** フィールドに入力します。
5. **OK** をクリックします。**プロジェクトが作成されました** ダイアログ ボックスが開きます。
6. **はい** をクリックすると新しいテストの記録が開始され、**いいえ** をクリックすると空のプロジェクトが作成されます。  
チュートリアルでは、**いいえ** をクリックします。  
記憶の開始を選択した場合、**新しい項目の追加** ダイアログ ボックスが開きます。
7. **新しい項目の追加** ダイアログ ボックスで、記録するテストの種類を選択します。
  - 記録した操作をキーワードにまとめる場合は、**Silk4NET キーワード駆動テスト** を選択します。
  - キーワードを作成せずにテストを記録する場合は、**Silk4NET テスト** を選択します。
8. 記録の開始を選択した場合、**新しい項目の追加** ダイアログ ボックスの **名前** テキスト ボックスに新しいテストの名前を入力します。
9. **追加** をクリックします。
- 10 新しい Silk4NET テストを記録するように選択した場合は、**アプリケーションの選択** ダイアログ ボックスが開きます。テストするアプリケーションの種類に対応するタブを選択します。
  - ブラウザで実行しない標準アプリケーションをテストする場合は、**Windows** タブを選択します。
  - Web アプリケーションまたはモバイル Web アプリケーションをテストする場合は、**Web** タブを選択します。
  - ネイティブ モバイル アプリケーションをテストする場合は、**モバイル** タブを選択します。
- 11 標準アプリケーションをテストするには、現在のプロジェクトに対してアプリケーション構成が設定されていない場合は、リストからアプリケーションを選択します。
- 12 Web アプリケーションまたはモバイル Web アプリケーションをテストするには、現在のプロジェクトに対してアプリケーション構成が設定されていない場合は、リストからインストール済みのブラウザまたはモバイル ブラウザのうちの 1 つを選択します。
  - a) **移動する URL の入力** テキスト ボックスに、開く Web ページを指定します。選択したブラウザのインスタンスが既に実行されている場合、**実行中のブラウザの URL を使用する** をクリックして、実行中のブラウザ インスタンスに現在表示されている URL の記録を行うことができます。チュートリアルの場合、**Internet Explorer** を選択し、**移動する URL の入力** テキスト ボックスに <http://demo.borland.com/InsuranceWebExtJS/> を指定します。
  - b) 省略可能：あらかじめ定義されたブラウザ サイズを使用してデスクトップ ブラウザー上の Web アプリケーションをテストする場合は、**ブラウザ サイズ** リストからブラウザ サイズを選択します。  
たとえば、Apple Safari 上の Web アプリケーションを Apple iPhone 7 の画面と同じ大きさのブラウザ ウィンドウでテストするには、リストから **Apple iPhone 7** を選択します。
  - c) 省略可能：ブラウザ ウィンドウの **向き** を選択します。
  - d) 省略可能：**ブラウザ サイズの編集** をクリックすると、新しいブラウザ サイズを指定したり、**ブラウザ サイズ** リストに表示するブラウザ サイズを選択することができます。
- 13 現在のプロジェクトに対してアプリケーション構成が設定されていない場合に、ネイティブ モバイル アプリケーション (アプリ) をテストするには：

a) アプリをテストするモバイル デバイスをリストから選択します。


b) ネイティブ モバイル アプリケーションを選択します。

- モバイル デバイスまたはエミュレータ上にアプリをインストールする場合は、**参照** をクリックしてアプリ ファイルを選択するか、**アプリ ファイル** テキスト フィールドにアプリ ファイルへの完全パスを入力します。このパスでは、Silk4NET は HTTP および UNC 形式をサポートします。
- Android デバイス上に既にインストールされているアプリを使用する場合は、**パッケージ/アクティビティ** リストからアプリを選択するか、**パッケージ/アクティビティ** フィールドにパッケージとアクティビティを指定します。
- iOS デバイス上に既にインストールされているアプリを使用する場合は、**バンドル ID** を指定します。
- UFT Mobile 上で利用可能なアプリを使用する場合は、**アプリ識別子** を指定します。

**140K** をクリックします。テスト メソッドの再生で Google Chrome の既存のインスタンスを選択した場合は、Silk4NET がオートメーション サポートが含まれているかどうかをチェックします。オートメーション サポートが含まれていない場合は、Silk4NET が Google Chrome を再起動する必要があることを通知します。アプリケーションと**記録中** ウィンドウが開きます。

空の Silk4NET テストを作成するように選択した場合は、Silk4NET プロジェクトを含む新しいソリューションが作成されます。また、このプロジェクトには、言語固有の以下のファイル名を使用して、Silk4NET テストも作成されます。

- UnitTest1.vb
- UnitTest1.cs

 **注:** **ソリューション エクスプローラ** のコンテキスト メニューを使用して、Silk4NET プロジェクトを既存のソリューションに追加することもできます。

# プロジェクトへの Silk4NET テストの追加

既存の Silk4NET またはテストプロジェクト、Silk4NET テストを追加することができます。Silk4NET またはテストプロジェクトが存在しない場合、Silk4NET テストを作成する前に Silk4NET またはテストプロジェクトを作成してください。

1. **Silk4NET > 新しいテスト** または **プロジェクト > 新しい項目の追加** をクリックします。



**注:** ソリューションに複数の Silk4NET プロジェクトが存在する場合、新しいテストを追加したいプロジェクトを**プロジェクトの選択** のリストから選択します。

**新しい項目の追加** ダイアログ ボックスが開きます。

2. **インストール済み** で次のいずれかをクリックします。

- プロジェクトが Visual Basic プロジェクトの場合は、**共通項目 > Silk4NET テスト** をクリックします。
- プロジェクトが Visual C# プロジェクトの場合は、**Visual C# アイテム > Silk4NET テスト** をクリックします。

3. テストの名前を **名前** フィールドに入力し、**追加** をクリックします。**Silk4NET テストの作成** ダイアログ ボックスが開きます。

4. 次のいずれかのオプション ボタンをクリックして、Silk4NET テストの作成方法を選択します。

**Silk4NET テストの記録** テスト対象アプリケーションに対する操作および検証を記録し、記録されたオートメーション ステートメントを含む新しいテストを生成します。

**空の Silk4NET テストの作成** オートメーション ステートメントを後で入力できる空のテストを作成します。

チュートリアルの場合、**空の Silk4NET テストの作成** をクリックします。

5. **OK** をクリックします。

6. 新しい Silk4NET テストを記録するように選択した場合は、**アプリケーションの選択** ダイアログ ボックスが開きます。テストするアプリケーションの種類に対応するタブを選択します。

- ブラウザで実行しない標準アプリケーションをテストする場合は、**Windows** タブを選択します。
- Web アプリケーションまたはモバイル Web アプリケーションをテストする場合は、**Web** タブを選択します。
- ネイティブ モバイル アプリケーションをテストする場合は、**モバイル** タブを選択します。

7. 標準アプリケーションをテストするには、現在のプロジェクトに対してアプリケーション構成が設定されていない場合は、リストからアプリケーションを選択します。

8. Web アプリケーションまたはモバイル Web アプリケーションをテストするには、現在のプロジェクトに対してアプリケーション構成が設定されていない場合は、リストからインストール済みのブラウザまたはモバイル ブラウザのうちの 1 つを選択します。

a) **移動する URL の入力** テキスト ボックスに、開く Web ページを指定します。選択したブラウザのインスタンスが既に実行されている場合、**実行中のブラウザの URL を使用する** をクリックして、実行中のブラウザ インスタンスに現在表示されている URL の記録を行うことができます。チュートリアルの場合、**Internet Explorer** を選択し、**移動する URL の入力** テキスト ボックスに <http://demo.borland.com/InsuranceWebExtJS/> を指定します。

b) 省略可能：あらかじめ定義されたブラウザ サイズを使用してデスクトップ ブラウザー上の Web アプリケーションをテストする場合は、**ブラウザ サイズ** リストからブラウザ サイズを選択します。

たとえば、Apple Safari 上の Web アプリケーションを Apple iPhone 7 の画面と同じ大きさのブラウザ ウィンドウでテストするには、リストから **Apple iPhone 7** を選択します。



c) 省略可能：ブラウザー ウィンドウの **向き** を選択します。

d) 省略可能：**ブラウザー サイズの編集** をクリックすると、新しいブラウザー サイズを指定したり、**ブラウザー サイズ** リストに表示するブラウザー サイズを選択することができます。

9. 現在のプロジェクトに対してアプリケーション構成が設定されていない場合に、ネイティブ モバイル アプリケーション (アプリ) をテストするには：

a) アプリをテストするモバイル デバイスをリストから選択します。

b) ネイティブ モバイル アプリケーションを選択します。

- モバイル デバイスまたはエミュレータ上にアプリをインストールする場合は、**参照** をクリックしてアプリ ファイルを選択するか、**アプリ ファイル** テキスト フィールドにアプリ ファイルへの完全パスを入力します。このパスでは、Silk4NET は HTTP および UNC 形式をサポートします。
- Android デバイス上に既にインストールされているアプリを使用する場合は、**パッケージ/アクティビティ** リストからアプリを選択するか、**パッケージ/アクティビティ** フィールドにパッケージとアクティビティを指定します。
- iOS デバイス上に既にインストールされているアプリを使用する場合は、**バンドル ID** を指定します。
- UFT Mobile 上で利用可能なアプリを使用する場合は、**アプリ識別子** を指定します。

**100K** をクリックします。テスト メソッドの再生で Google Chrome の既存のインスタンスを選択した場合は、Silk4NET がオートメーション サポートが含まれているかどうかをチェックします。オートメーション サポートが含まれていない場合は、Silk4NET が Google Chrome を再起動する必要があることを通知します。アプリケーションと**記録中** ウィンドウが開きます。

テストを記録するように選択した場合は、記録したテストがプロジェクトに追加されます。空のテストを追加するように選択した場合は、空の Silk4NET テストがプロジェクトに追加されます。



**注:** **ソリューション エクスプローラ** のコンテキスト メニューを使用して、Silk4NET を Silk4NET またはテスト プロジェクトに追加します。

# Silk4NET テストの記録

1. Silk4NET > 新しいテスト または プロジェクト > 新しい項目の追加 をクリックします。



**注:** ソリューションに複数の Silk4NET プロジェクトが存在する場合、新しいテストを追加したいプロジェクトを**プロジェクトの選択** のリストから選択します。

**新しい項目の追加** ダイアログ ボックスが開きます。

2. **インストール済み** で次のいずれかをクリックします。

- プロジェクトが Visual Basic プロジェクトの場合は、**共通項目** > **Silk4NET テスト** をクリックします。
- プロジェクトが Visual C# プロジェクトの場合は、**Visual C# アイテム** > **Silk4NET テスト** をクリックします。

3. テストの名前を **名前** フィールドに入力し、**追加** をクリックします。**Silk4NET テストの作成** ダイアログ ボックスが開きます。

4. **Silk4NET テストの記録** を選択して、**OK** をクリックします。

5. 現在のプロジェクトに対してアプリケーション構成が設定されていない場合、テストするアプリケーションの種類に対応するタブを選択します。

- ブラウザで実行しない標準アプリケーションをテストする場合は、**Windows** タブを選択します。
- Web アプリケーションまたはモバイル Web アプリケーションをテストする場合は、**Web** タブを選択します。
- ネイティブ モバイル アプリケーションをテストする場合は、**モバイル** タブを選択します。

6. 標準アプリケーションをテストするには、リストからアプリケーションを選択します。

7. Web アプリケーションまたはモバイル Web アプリケーションをテストするには、現在のプロジェクトに対してアプリケーション構成が設定されていない場合は、リストからインストール済みのブラウザまたはモバイル ブラウザのうちの 1 つを選択します。

a) **移動する URL の入力** テキスト ボックスに、開く Web ページを指定します。選択したブラウザのインスタンスが既に実行されている場合、**実行中のブラウザの URL を使用する** をクリックして、実行中のブラウザ インスタンスに現在表示されている URL の記録を行うことができます。チュートリアルの場合、**Internet Explorer** を選択し、**移動する URL の入力** テキスト ボックスに <http://demo.borland.com/InsuranceWebExtJS/> を指定します。

b) 省略可能：あらかじめ定義されたブラウザ サイズを使用してデスクトップ ブラウザー上の Web アプリケーションをテストする場合は、**ブラウザ サイズ** リストからブラウザ サイズを選択します。

たとえば、Apple Safari 上の Web アプリケーションを Apple iPhone 7 の画面と同じ大きさのブラウザ ウィンドウでテストするには、リストから **Apple iPhone 7** を選択します。

c) 省略可能：ブラウザ ウィンドウの **向き** を選択します。

d) 省略可能：**ブラウザ サイズの編集** をクリックすると、新しいブラウザ サイズを指定したり、**ブラウザ サイズ** リストに表示するブラウザ サイズを選択することができます。

8. 現在のプロジェクトに対してアプリケーション構成が設定されていない場合に、ネイティブ モバイル アプリケーション (アプリ) をテストするには：

a) アプリをテストするモバイル デバイスをリストから選択します。

b) ネイティブ モバイル アプリケーションを選択します。

- モバイル デバイスまたはエミュレータ上にアプリをインストールする場合は、**参照** をクリックしてアプリ ファイルを選択するか、**アプリ ファイル** テキスト フィールドにアプリ ファイルへの完全パスを入力します。このパスでは、Silk4NET は HTTP および UNC 形式をサポートします。

- Android デバイス上に既にインストールされているアプリを使用する場合は、**パッケージ/アクティビティ** リストからアプリを選択するか、**パッケージ/アクティビティ** フィールドにパッケージとアクティビティを指定します。
- iOS デバイス上に既にインストールされているアプリを使用する場合は、**バンドル ID** を指定します。
- UFT Mobile 上で利用可能なアプリを使用する場合は、**アプリ識別子** を指定します。

**9. OK** をクリックします。テストメソッドの再生で Google Chrome の既存のインスタンスを選択した場合は、Silk4NET がオートメーションサポートが含まれているかどうかをチェックします。オートメーションサポートが含まれていない場合は、Silk4NET が Google Chrome を再起動する必要があることを通知します。アプリケーションと**記録中** ウィンドウが開きます。


**10** 記録を行う、テスト対象アプリケーションとの対話を実行します。

モバイルアプリケーションの記録についての詳細は、モバイルアプリケーションの記録を参照してください。記録中に利用可能な操作についての詳細は、「記録中に利用可能な操作」を参照してください。

**11** 記録が終了したら、**停止** をクリックします。**記録完了** ダイアログボックスが開きます。ダイアログボックスにある**再生** をクリックして、記録したテストを再生できます。

- Visual Studio 2010 を使用している場合、Visual Studio の**テストビュー** にアクセスしてテストを再生したり管理することもできます。
- Visual Studio 2012 以降を使用している場合、Visual Studio の**テストエクスプローラー** にアクセスしてテストを再生したり管理することもできます。

記録された対話は、プロジェクトにファイルとして追加されます。生成されたファイルのデフォルトのファイル名は、プロジェクトのデフォルトプログラミング言語に応じて、UnitTest<Index>.cs または UnitTest<Index>.vb になります。たとえば、Visual Basic プロジェクトの初回テストを記録している場合、生成されたファイルの名前は UnitTest1.vb になります。

 **注:** 新しいプロジェクトを作成し、そこに新しいテスト記録することもできます。

# Silk4NET テストの実行

デフォルトでは、Silk4NET 19.5 以降のキーワード駆動テストの実行に 64 ビット JVM が使用されます。Silk4NET でキーワード駆動テストを実行するには、規定のプロセッサアーキテクチャを X64 に設定する必要があります。この設定は、Visual Studio メニューの **テスト > テスト設定 > 規定のプロセッサアーキテクチャ** から選択できます。また、`SILK_KEYWORD_DRIVEN_JAVA_HOME` 環境変数を使用して、Silk4NET が 32 ビット JVM を使用するように設定することもできます。

このトピックでは、Silk4NET テストを Visual Studio で実行する方法について説明します。

1. 選択したプロジェクトまたはソリューションで利用可能なすべてのテストを表示するには：
  - Visual Studio 2010 の場合、**テスト > ウィンドウ > テスト ビュー** をクリックします。
  - Visual Studio 2012 以降の場合、**テスト > ウィンドウ > テスト エクスプローラー** をクリックします。
2. **テスト ビュー**、または **テスト エクスプローラ** (使用している Visual Studio のバージョンによる) で、実行するテストを選択します。
3. 選択項目を右クリックして、次のいずれかをクリックします。
  - Visual Studio 2010 の場合、**選択範囲の実行** をクリックします。
  - Visual Studio 2012 以降の場合、**選択したテストの実行** をクリックします。

選択したプロジェクトまたはソリューションのすべてのテストを実行するには、**テスト ビュー**、または **テスト エクスプローラー** (使用している Visual Studio のバージョンによる) で、**すべて実行** をクリックします。

4. Web アプリケーションをテストする場合は、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、**実行** をクリックします。
5. 省略可能：必要に応じて、両方の **Shift** キーを同時に押して、テストの実行を停止できます。
6. テストの実行が完了すると、**再生完了** ダイアログ ボックスが開きます。TrueLog を使用してテスト結果を調べるには、**結果の検討** をクリックします。ダイアログ ボックスを閉じるには、**OK** をクリックします。



**注:** テストを実行し、Visual Studio がテストの実行に必要なコンポーネントを開始する場合、Visual Studio はテスト実行が完了したときにすべてをクリーンアップします。つまり、Open Agent とすべての開いているブラウザー ウィンドウが強制終了されます。

# TrueLog の有効化

新しい Silk4NET スクリプトでは、TrueLog がデフォルトで有効になっています。**スクリプト オプション** ダイアログ ボックスの **TrueLog** タブにある **TrueLog の有効化** チェック ボックスのオン/オフを切り替えて、TrueLog を有効、または無効にすることができます。**スクリプト オプション** ダイアログ ボックスを開くには、**Silk4NET > オプションの編集** をクリックします。

Visual Studio Unit Testing Framework を使用している既存の Silk4NET スクリプトで TrueLog を有効にするには、スクリプト内のすべてのテスト クラスの TestClass 属性を SilkTestClass 属性で置き換える必要があります。

既存の Silk4NET スクリプトに対して TrueLog を有効にするには：

1. TrueLog を有効にするテスト クラスが含まれたスクリプトを開きます。
2. テスト クラスに SilkTestClass 属性を追加します。

デフォルトでは、Visual Studio ソリューション ファイルおよび Visual Studio Unit Testing Framework の結果が格納されているディレクトリの TestResults サブディレクトリに TrueLog が作成されます。TrueLog の別の場所を指定するには、**Silk4NET > オプションの編集** をクリックして **スクリプト オプション** ダイアログ ボックスを開き、**TrueLog ファイル** フィールドの右側にある **参照** をクリックします。

Visual Studio ソリューション ファイルには、Silk4NET スクリプトが格納されています。Silk4NET テストの実行が完了したら、ダイアログ ボックスが開きます。**結果の検討** をクリックすると、完了したテストの TrueLog を確認できます。

## 使用例

Visual Basic スクリプト内のクラスに対して TrueLog を有効にするには、以下のコードを使用します。

```
<SilkTestClass()> Public Class MyTestClass
  <TestMethod()> Public Sub MyTest()
    ' my test code
  End Sub
End Sub
```

C# スクリプト内のクラスに対して TrueLog を有効にするには、以下のコードを使用します。

```
[SilkTestClass]
public class MyTestClass {
  [TestMethod]
  public void MyTest() {
    // my test code
  }
}
```

# テスト結果の分析

テストを実行後、テスト結果を確認し、テスト実行の成功または失敗を分析できます。

1. Silk4NET テストを実行します。実行が終了すると、**再生完了** ダイアログ ボックスが開きます。
2. **結果の検討**を実行して、テストの結果を確認します。
3. 結果をクリックします。

テストに失敗すると、Silk4NET によってスクリーンショットがキャプチャされます。