Ranorex

簡易チュートリアル

モバイルアプリケーションテスト

Last updated: 2023.7
Ranorex version: 10.7

目次

目次 **2**

はじめに **3**

本書について 3

動作環境 3

前提条件・注意事項 4

演習について 5

Lesson1 事前準備 **6**

Ranorex の起動 6

プロジェクトの作成とデバイスの設定 7

Lesson2 テストの作成と実行 **25**

モバイルアプリケーションのテスト作成 25

テストの作成 26

テストの実行 37

テスト結果レポートの確認 38

Appendix アクションのご紹介 **40**

アクションの待機 40

スワイプ操作 41

タッチ操作 43

はじめに

本書について

本書では、モバイルのアプリケーションを対象とした UI テストの自動化方法について説明します。

動作環境

Ranorex を使用したモバイルテストでは、モバイル端末上で動作する、以下のアプリケーションをサポートしています。

- ネイティブアプリケーション（iOS, Android）
- Web アプリケーション
  - Android は、専用のブラウザー（RxBrowser）を使用します。
  - iOS15、16 では、Safari ブラウザーをサポートしています。
    ※iOS 14 以前のバージョンは、Safari ブラウザーをサポートしていないため、RxBrowser を使用します。
- ハイブリッドアプリケーション

テスト環境においては、以下のプラットフォームをサポートしています。

- モバイル端末（iOS/Android）
- エミュレーター（Android）

前提条件・注意事項

Ranorex のモバイルアプリケーションテストにおける、前提条件・注意事項を以下に記載いたします。

Android/iOS 共通

- 対象のモバイル端末を USB または Wi-Fi で、Ranorex をインストールしている PC と接続する必要があります。
- 対象のモバイル端末への Ranorex サービスアプリケーション（Ranorex Service App）のインストールが必要です。
- プリインストールアプリケーションへのテストはサポートしていません。
- マルチタッチ操作（ダブルスワイプ、ピンチイン/アウトなど）はサポートしていません。

Android

- モバイルの Web アプリケーションのテストには、RxBrowser のインストールが必要です。
  ※他のブラウザーでのテストはサポートしていません。

iOS

- ロングタッチ操作をサポートしていません。
- 各アクションの Action Spot は、 Center のみの指定となります。
- Safari ブラウザーでのテストには、Ranorex Service App のバージョン 3.2.7 のインストールが必要となります。
  iOS 14 以前のバージョンのデバイスにおいては、RxBrowser のインストールが必要です。

動作対象の OS バージョン

- Android 12 までサポートします。
  ※*RxServiceApp 2.3.3* は *Android v8* 以上でないとインストールができないため、*Android v8* より下位のバージョンでは動作しません。
- iOS 16 までサポートします。

- 以下のフレームワークを公式サポートしていません。
  ※アプリケーションによっては、一部のコントロールを認識できる場合もございます。
  - iOS：Swift、SwiftUI、UIKit、SFSafariViewController、Flutter
  - Android：Jetpack、Flutter

演習について

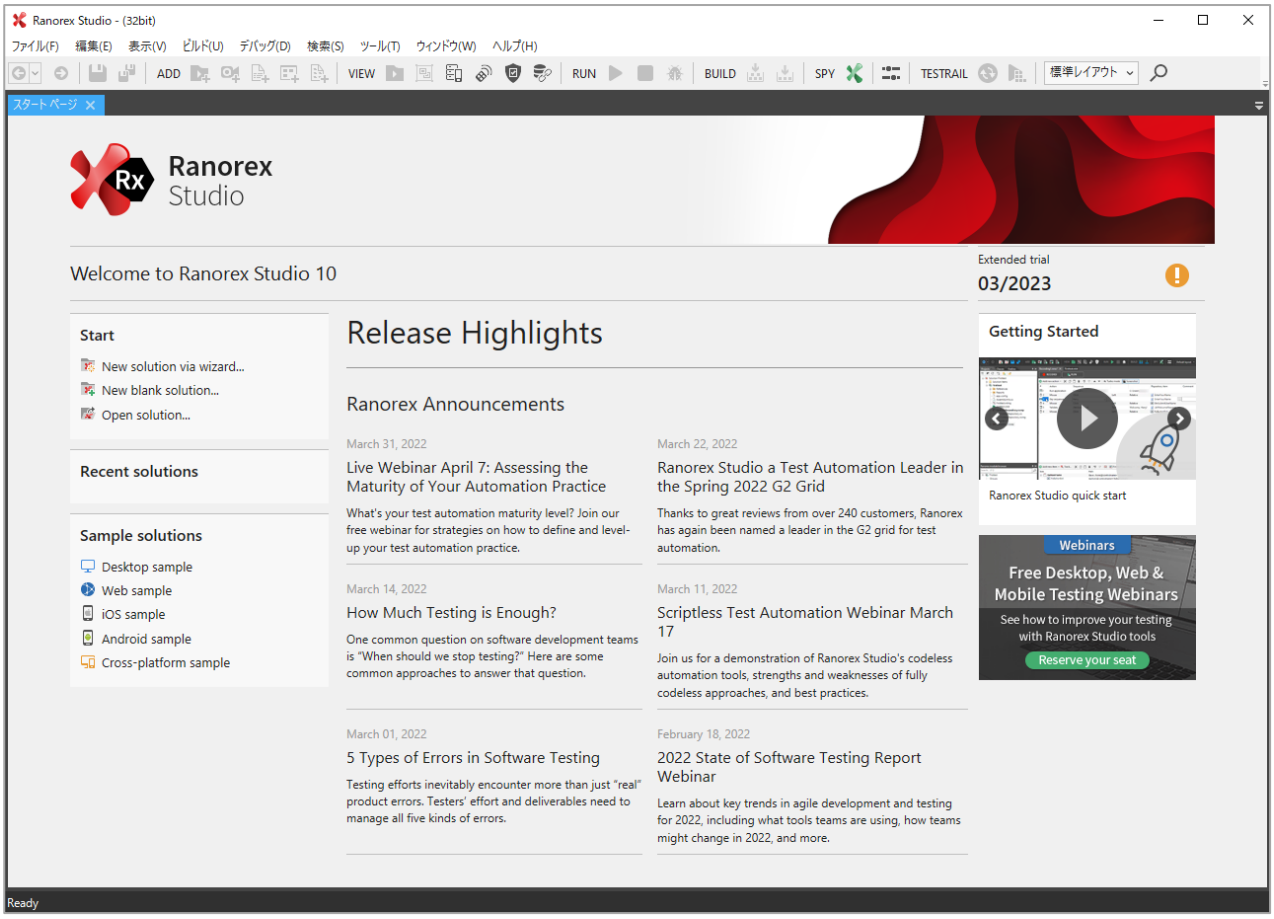
本書の演習では、モバイル端末 Android のネイティブアプリケーション（KeePassDroid）を対象としたテストの作成方法を学習します。

Lesson1 事前準備

Ranorex の起動

はじめに Ranorex（Ranorex Studio）を起動します。

Windows のスタートメニューを選択し、 **Ranorex Studio – Ranorex Studio 10** を選択すると、Ranorex が起動し、下図の画面が表示されます。



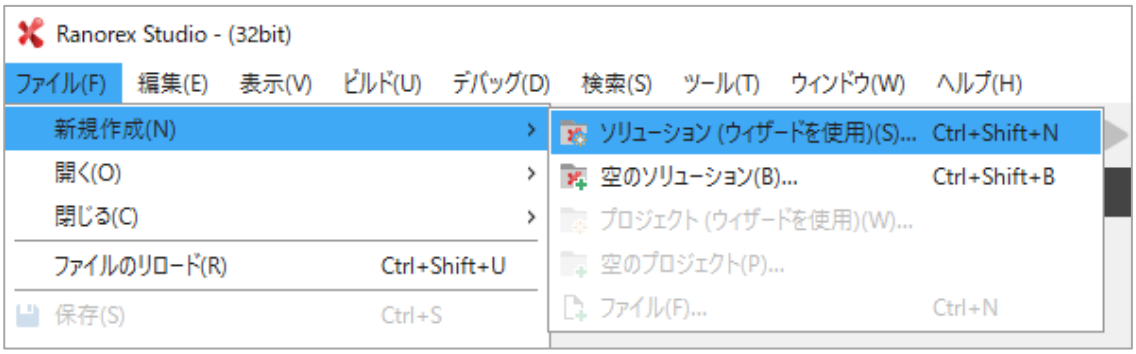
プロジェクトの作成とデバイスの設定

Ranorex にてアプリケーションへのテストをおこなうには、はじめにプロジェクトを作成します。

新規プロジェクトの作成

本演習では、Ranorex でのテスト作成時に、 **ロケットスタートウィザード** という機能を使用します。こちらを使用することで、各設定が自動でおこなわれ、テスト作成方法のガイドがステップバイステップで表示されるため、より簡単にプロジェクトを作成することができます。

1. メニューバーから **ファイル** – **新規作成** – **ソリューション（ウィザードを使用）**を選択します。



2. 表示された画面にて、テスト対象アプリケーションの種類を選択します。

   今回は、 **Mobile** を選択の上、次に進みます。

3. 以下の通りに各項目を設定し、 **続行** ボタンをクリックします。

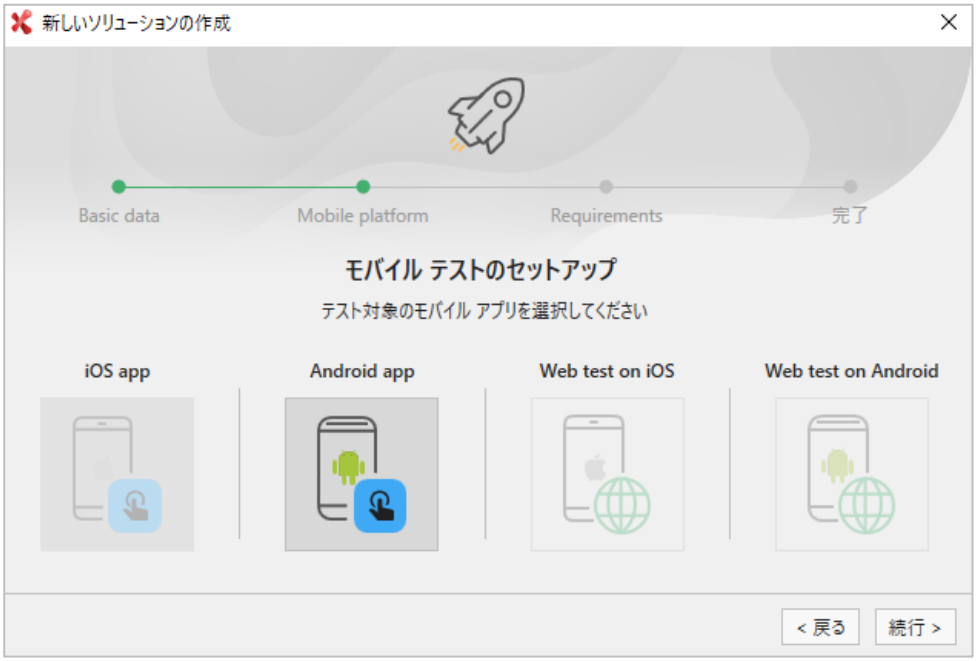
   ソリューション名：`Mobile_Test`

   プロジェクト名　：`Android_Test`

   言語　　：**C#（もしくは VB.NET）**

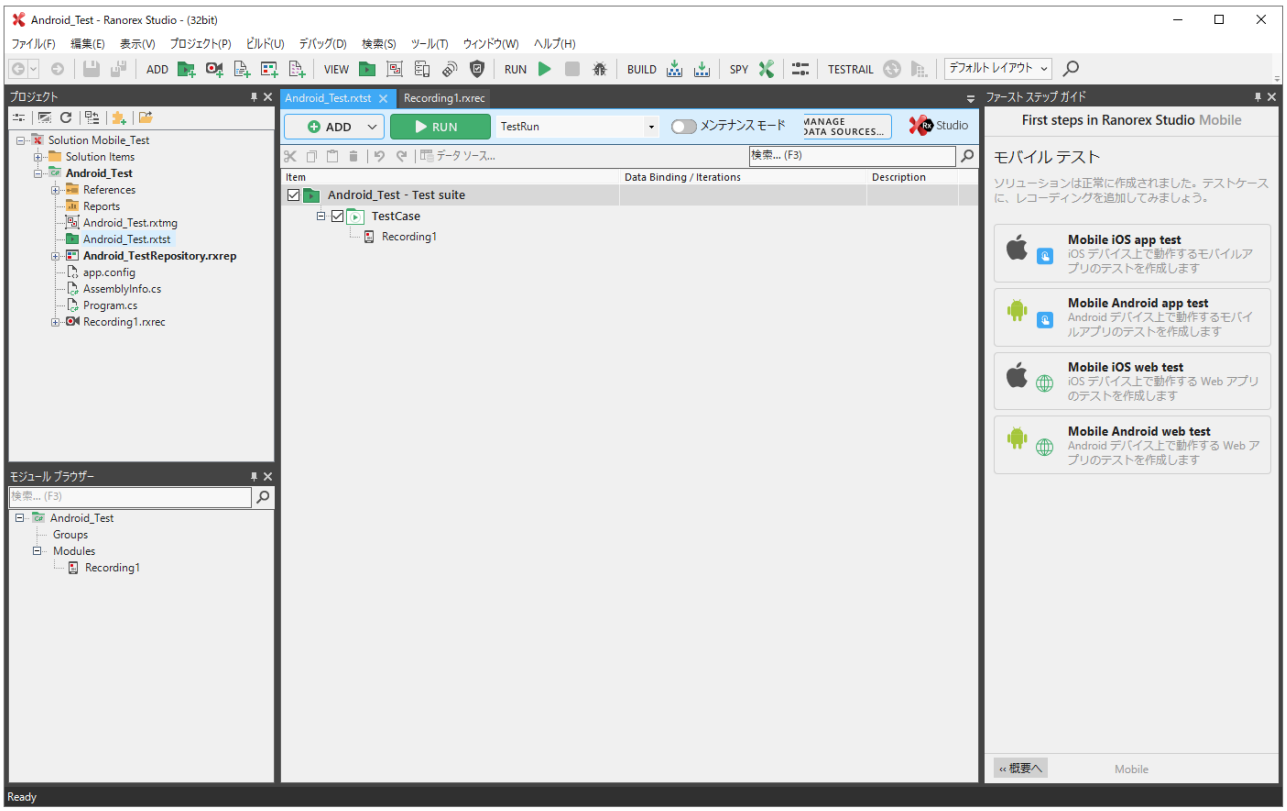
4. テスト対象となるモバイルアプリケーションの種類を選択します。

   本演習では、 **Android app** を選択し、 **続行** ボタンをクリックします。



5. モバイルテストに必要な事前準備が表示されるため、要件を満たしていることを確認した上で、 **続行** ボタンをクリックします。

6. 完了画面にて、右下の **完了** ボタンをクリックすると、プロジェクトが生成されます。



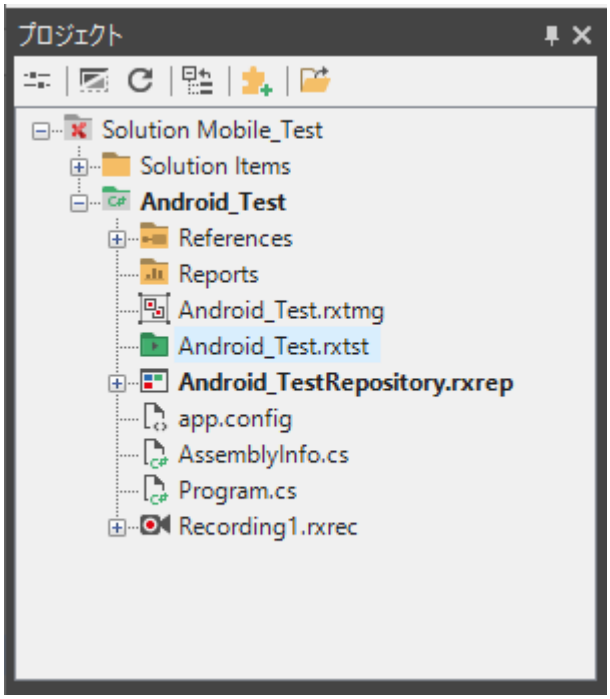
ファーストステップガイド

Ranorex でプロジェクトを作成すると、右側に **ファーストステップガイド** が表示されます。**ファーストステップガイド** では、テスト作成時に必要な手順を表示しています。

たとえば、 **Mobile Android app test** を選択すると、Android のネイティブアプリケーションのテストをおこなう場合に必要な手順を確認することができます。

プロジェクトビュー

プロジェクトビューでは、プロジェクトに関連付けられているすべてのファイルが表示され、参照できます。



プロジェクトには、以下のファイルが含まれます。

各ファイルをクリックすると、Ranorex Studio の画面上に各ファイルの内容が表示されます。

ファイル名	内容	ファイル形式
テストスイート	テストケース、テストスクリプトをまとめたテスト構成画面	.rxtst
モジュールグループ	任意のモジュールを 1 つにまとめた際のモジュール	.rxtmg
UI 要素の管理	テストスクリプトで使用される UI 要素	.rxrep
Record ベースのテストスクリプト （レコーディングモジュール）	記録された個々の操作手順の情報	.rxrec
Code ベースのテストスクリプト （コードモジュール）	テストスクリプトのソースコード （C# / VB.NET）	.cs / .vb

モバイルデバイスの接続

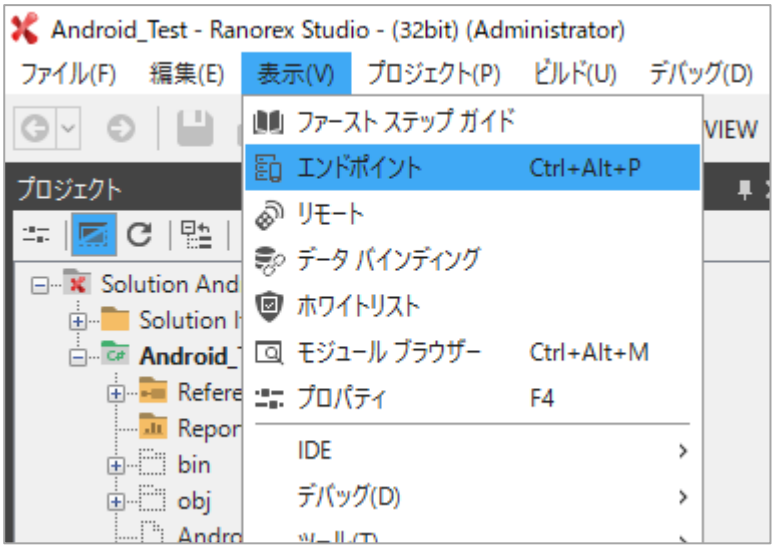
モバイルテストをおこなう場合、Ranorex Studio 上で対象のデバイス（モバイル端末）を認識させ、エンドポイントとして追加します。エンドポイントとして追加したデバイスは、テストの実行環境として使用できます。

> メモ
>
> **エンドポイント** とは、テストを実行する環境です。
>
> Ranorex Studio を起動しているローカルマシンの他、Android / iOS / WebDriver などを **エンドポイント** として指定し、テストを実行することができます。

> メモ
>
> エンドポイントにデバイスを追加する際には、Ranorex をインストールしているマシンとデバイスを、USB もしくは Wi-Fi で接続する必要があります。
>
> 本演習では、USB 接続をおこないます。
>
> なお、モバイルテストでは、Android および iOS の端末に対し、事前に以下の設定をおこなってください。
>
> **Android 端末**
>
> - **開発者モード** を有効にする
> - **USB デバッグ** と **スリープモードにしない** を **有効** にする
>
> **iOS 端末**
>
> - **自動ロック** を **無効** にする

エンドポイントの追加は、以下の手順にておこないます。

1. Ranorex Studio をインストールしているマシンと、対象のデバイスを USB で接続します。
2. メニューバーから **表示** – **エンドポイント** をクリックします。



3. **エンドポイント** 画面から、 **エンドポイントの追加** を選択します。

4. **エンドポイントの追加** 画面 から、以下 3 つの設定をおこないます。

①エンドポイント タイプの選択 を **Android** に設定します。

②Android デバイスの準備 の指示に従い、対象のデバイスに **Ranorex Service App** をインストールします。

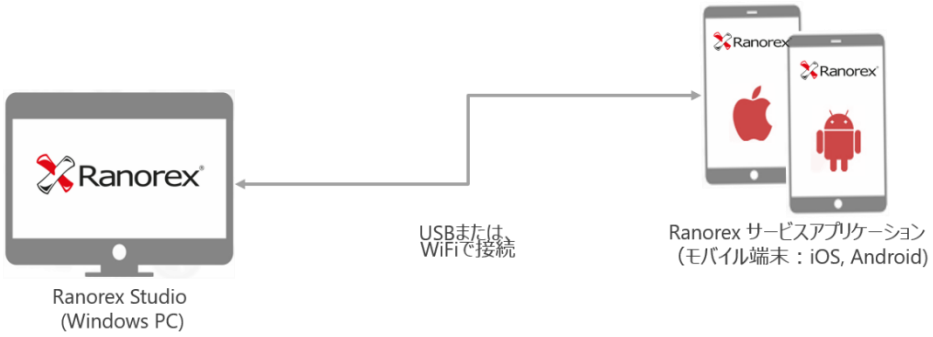
※デバイスで QR コードを読み取り、ダウンロード/インストールをおこなうことができます。

③Android デバイスの選択 に、ホストマシンと接続しているデバイスが表示されるため、該当のデバイスを選択した上で、**エンドポイントの追加** を選択します。

これにより、エンドポイントに対象のデバイスが追加されます。

Ranorex サービスアプリケーション

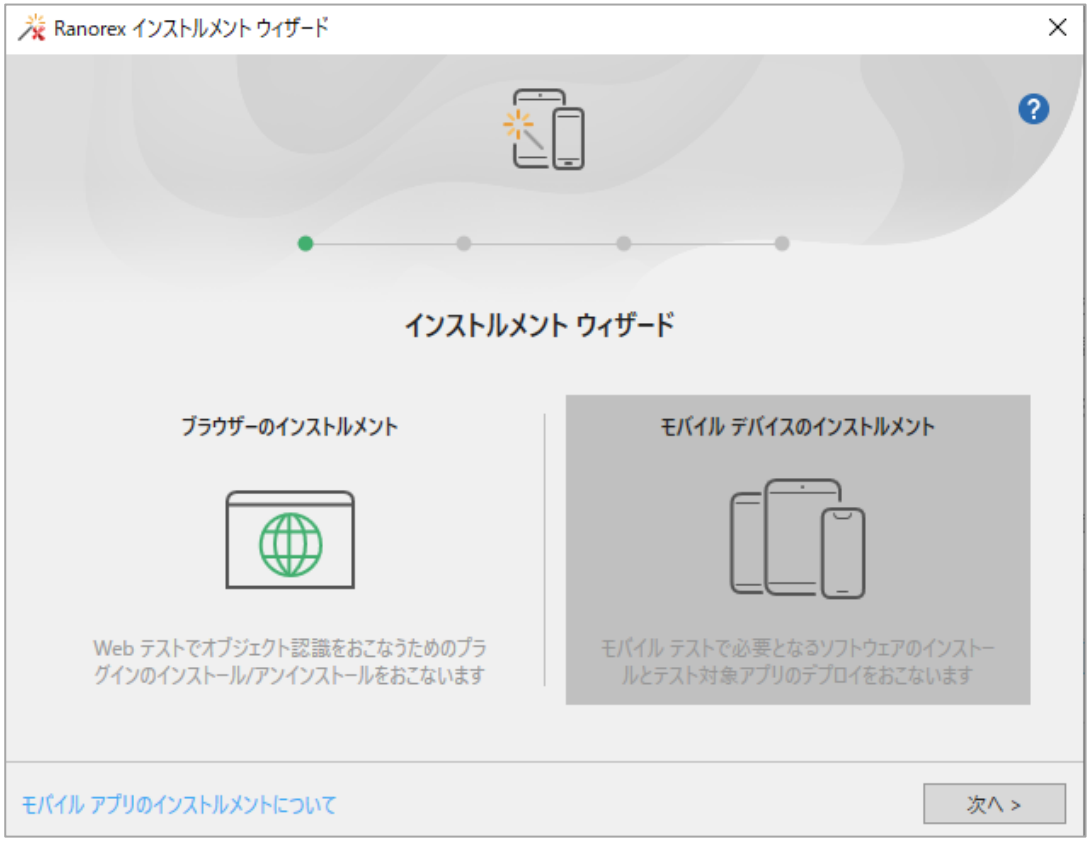
Ranorex のモバイルテストをおこなう場合、対象のデバイスに Ranorex サービスアプリケーションをインストールする必要があります。 **Ranorex サービスアプリケーション** は、テスト対象のアプリケーションを制御するために必要となります。



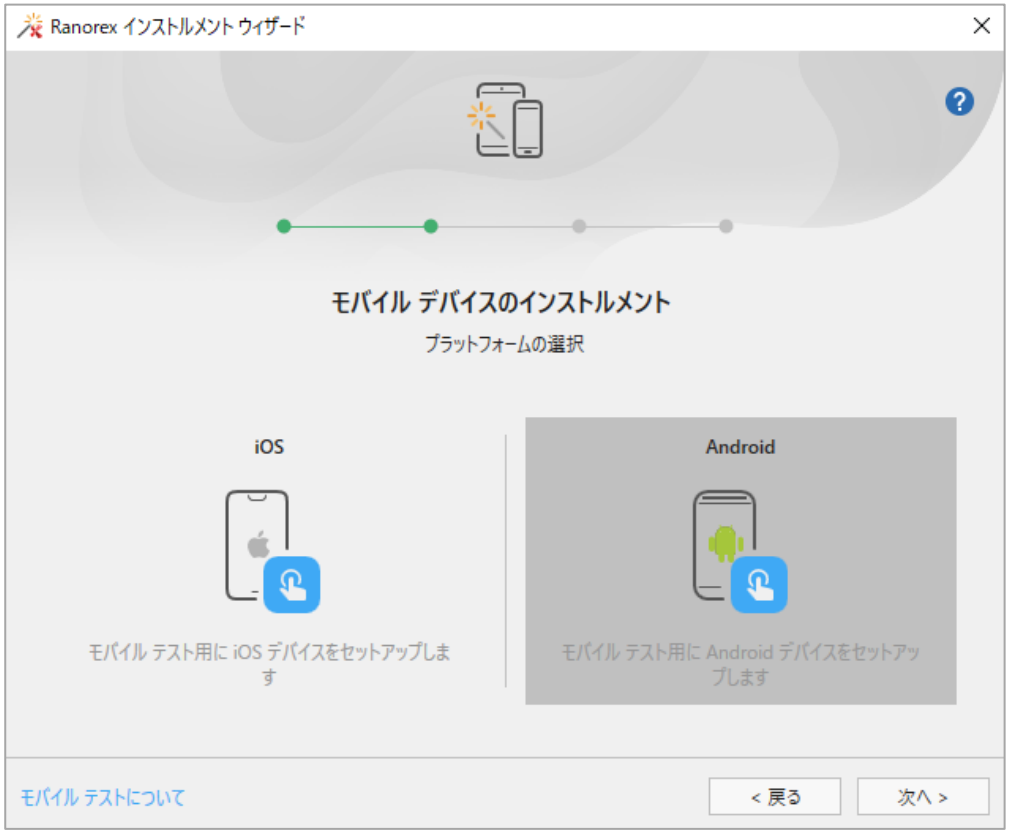
Ranorex サービスアプリケーション は、エンドポイント画面からインストールをおこなえますが **インストルメント ウィザード** を使用し、インストールすることも可能です。

設定手順は、以下となります。

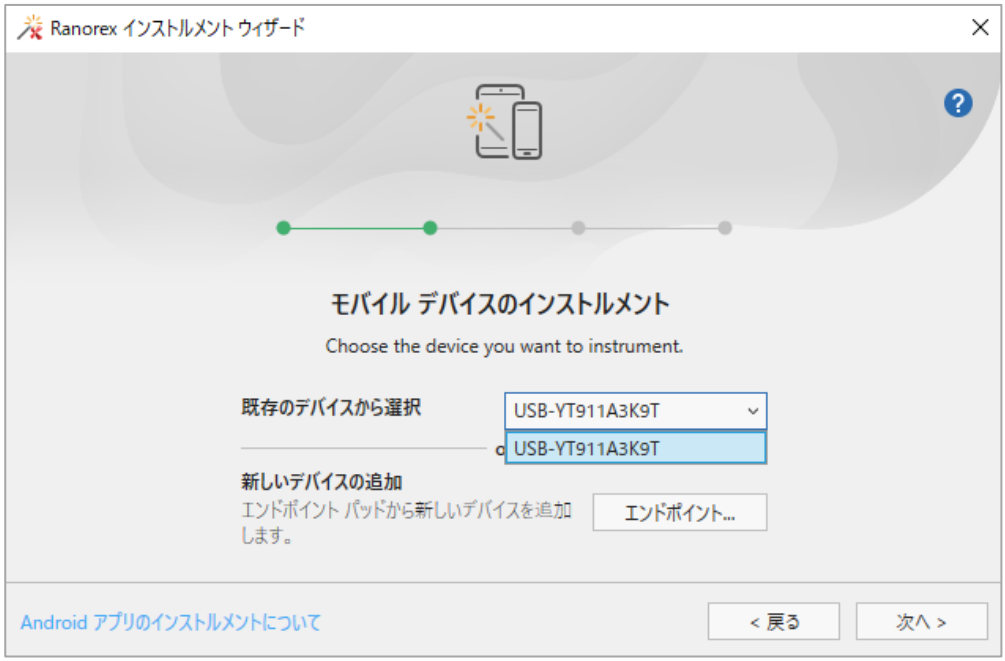
1. Ranorex Studio をインストールしたマシンと、対象のデバイスを USB（もしくは Wi-Fi）で接続します。
2. **ツール** – **インストルメント ウィザード** を選択し、インストルメント ウィザード を起動します。
3. **モバイルデバイスのインストルメント** を選択し、 **次へ** をクリックします。



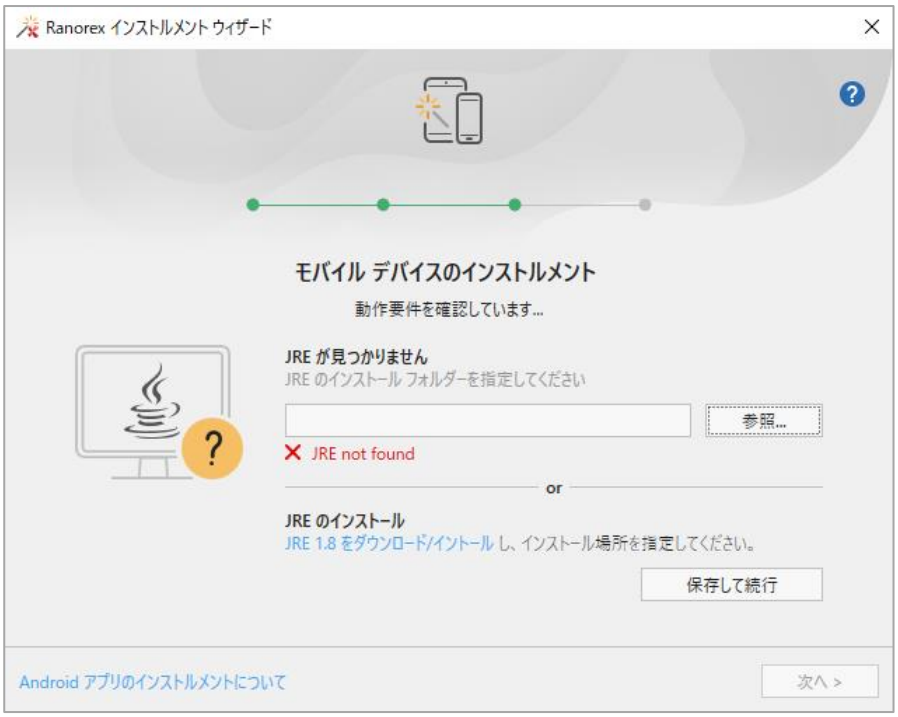
4. **Android** を選択し、 **次へ** をクリックします。



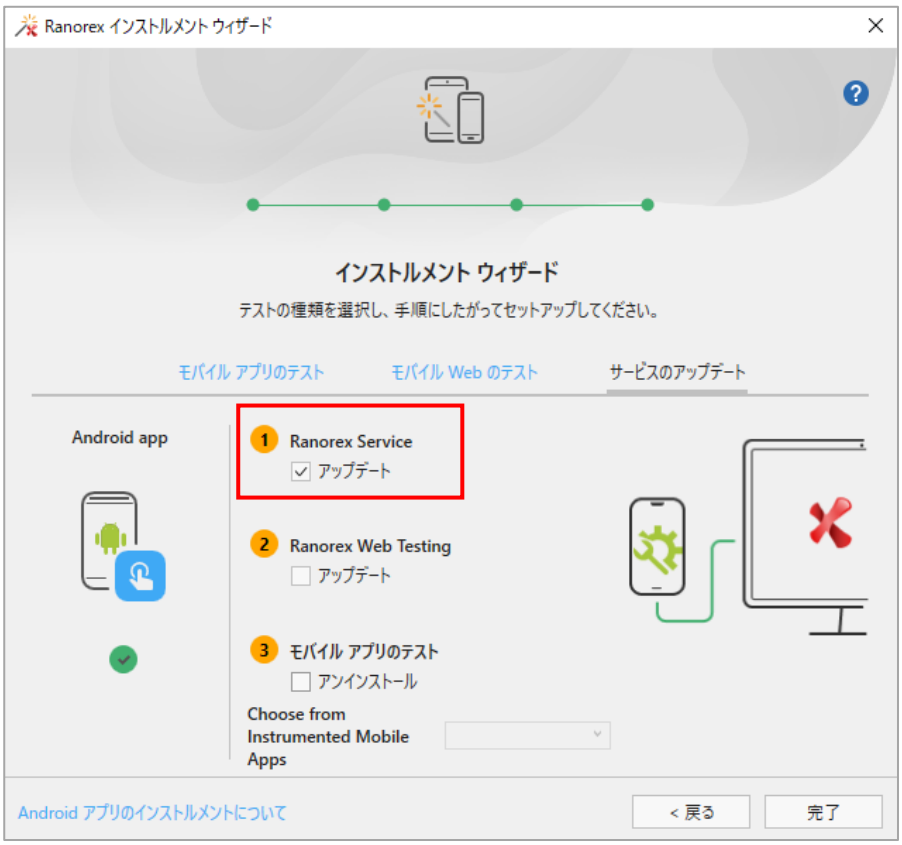
5. **既存のデバイスから選択** から対象のデバイスを選択し、 **次へ** をクリックします。



※初回のみ JRE の設定をおこなう必要があります。**参照** ボタンをクリックし JRE（または JDK）のインストールディレクトリを指定してください。JRE を参照後、**保存して続行** をクリックしてください。また、JRE のバージョンは 1.8 を使用してください。



6. **サービスのアップデート** を開きます。
   Ranorex Service – アップデート を選択後、**完了** ボタンをクリックします。

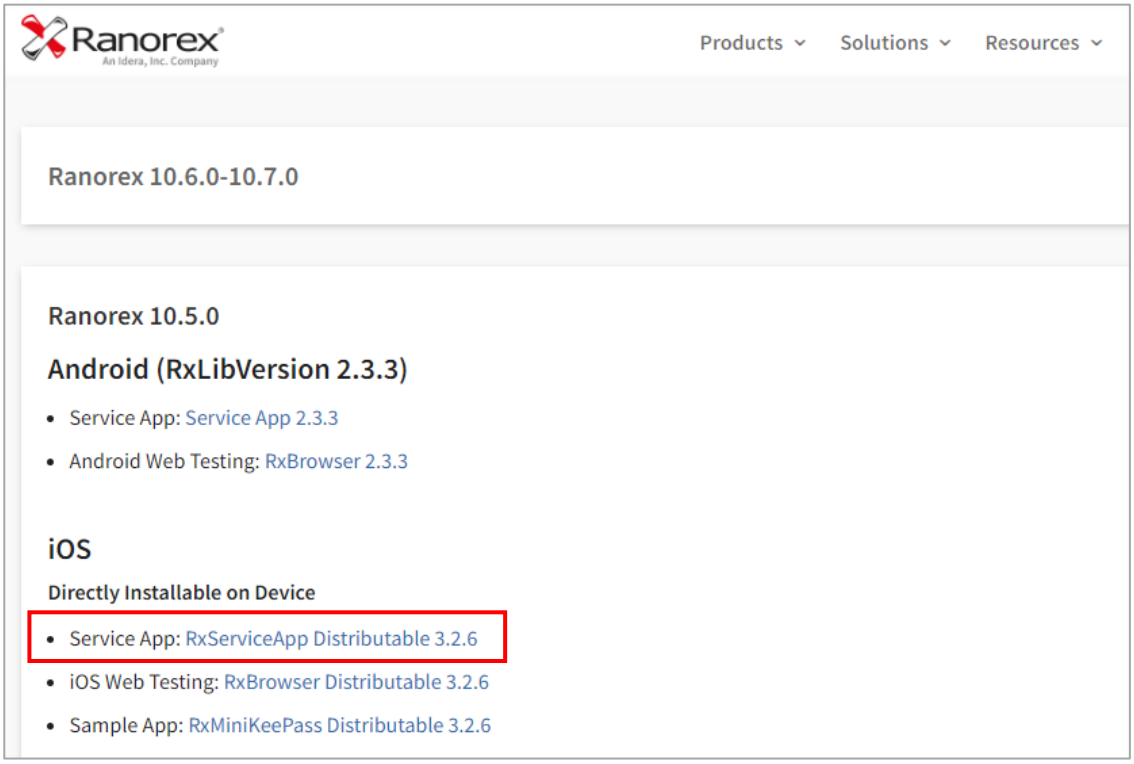


上記手順により、対象のデバイス上に **Ranorex サービスアプリケーション** をインストールできます。

注意

iOS 14 以前のバージョンのデバイスを使用する場合、対象の iOS デバイスから以下の開発元サイトのページにアクセスし、Ranorex サービスアプリケーションのバージョン 3.2.6 をインストールしてください。

開発元サイト URL：https://www.ranorex.com/mobile-download-archive/

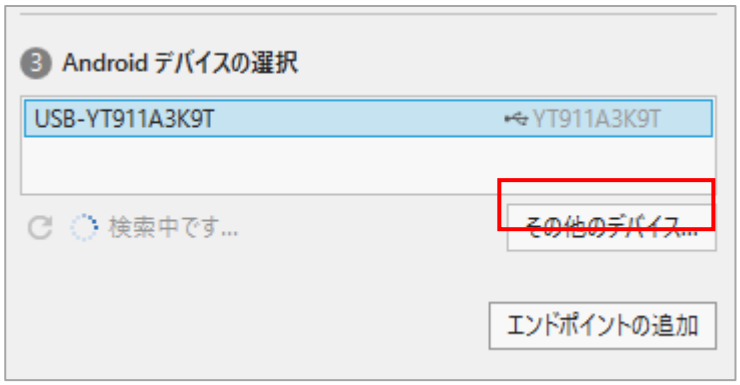


デバイスの登録方法

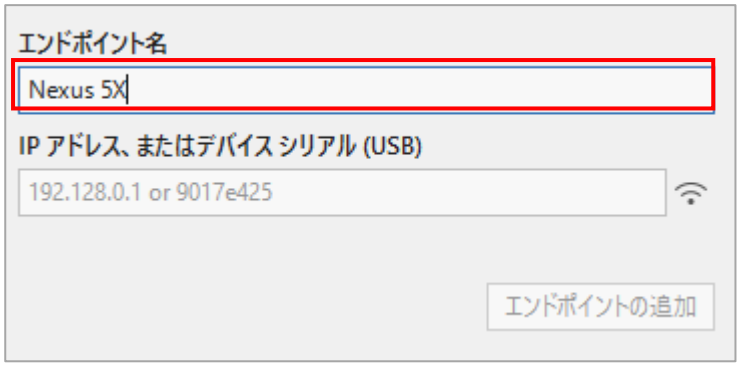
エンドポイント名は、自由に変更することができます。エンドポイント名の設定画面は、 **エンドポイントの追加** 画面右下にある **その他のデバイス** ボタンをクリックすると表示されます。

エンドポイント名をデバイス名などに設定することで、よりエンドポイントの管理がしやすくなります。

画像：**③Android デバイスの選択** 画面



画像：**その他のデバイス...** 選択後の画面

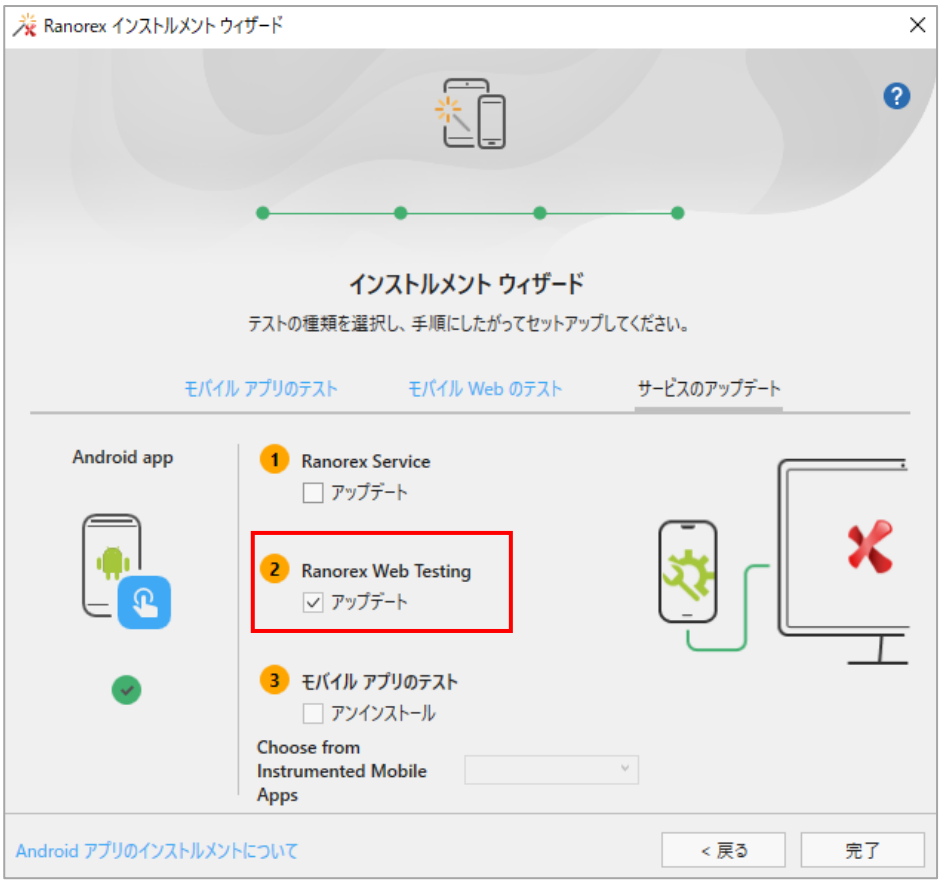


モバイルの Web アプリケーションをテストする場合

モバイルの Web アプリケーションをテストする場合、RxBrowser という Ranorex 独自のブラウザーを使用します。

RxBrowser をデバイスにインストールする場合、Ranorex Studio のメニューバーから **ツール** – **インストルメント ウィザード** を起動し、 **Ranorex Web Testing – アップデート** を選択することで、インストールできます。

※iOS 15、および 16 で Safari ブラウザーを使用した Web アプリケーションテストを実施する場合、この作業は不要です。



注意

RxBrowser をインストールする際には、対象のデバイス上に Ranorex サービスアプリケーション がインストールされている必要があります。

注意

iOS 15、および 16 で Safari ブラウザーを使用した Web アプリケーションのテストを実施する場合、Ranorex サービスアプリケーション（バージョン 3.2.7）がインストール済みの iOS デバイスにて、以下の設定をおこなってください。

1) **設定** をタップします。

2) **Safari** をタップします。

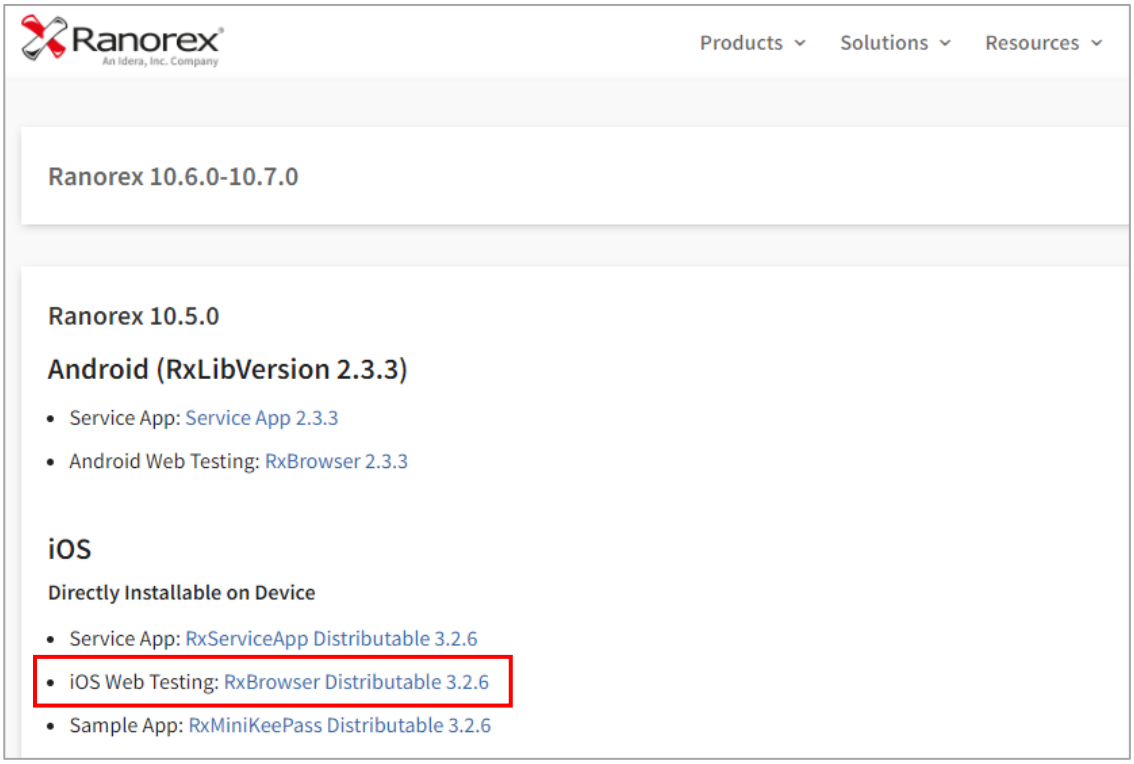
3) **拡張機能** をタップします。

4) **Ranorex Safari Extension** を有効にします。

注意

iOS 14 以前のバージョンのデバイスを使用する場合、対象の iOS デバイスから以下の開発元サイトのページにアクセスし、RxBrowser のバージョン 3.2.6 をインストールしてください。

開発元サイト URL：https://www.ranorex.com/mobile-download-archive/



テスト対象アプリケーションのインストルメントとデプロイ

Ranorex にて、モバイルのネイティブアプリケーションをテストする場合には、テスト対象のアプリケーション（APK/IPA ファイル）に対して Ranorex のライブラリの追加をおこない、リビルドすることで、Ranorex から操作できるようにします。

この処理を **インストルメント** と呼び、Ranorex の機能である **インストルメント ウィザード** を使用しておこないます。

テスト対象アプリケーションのインストルメントとデプロイに必要なファイル

テスト対象のアプリケーションをインストルメントする際に必要となるファイルは、以下となります。

Android 端末

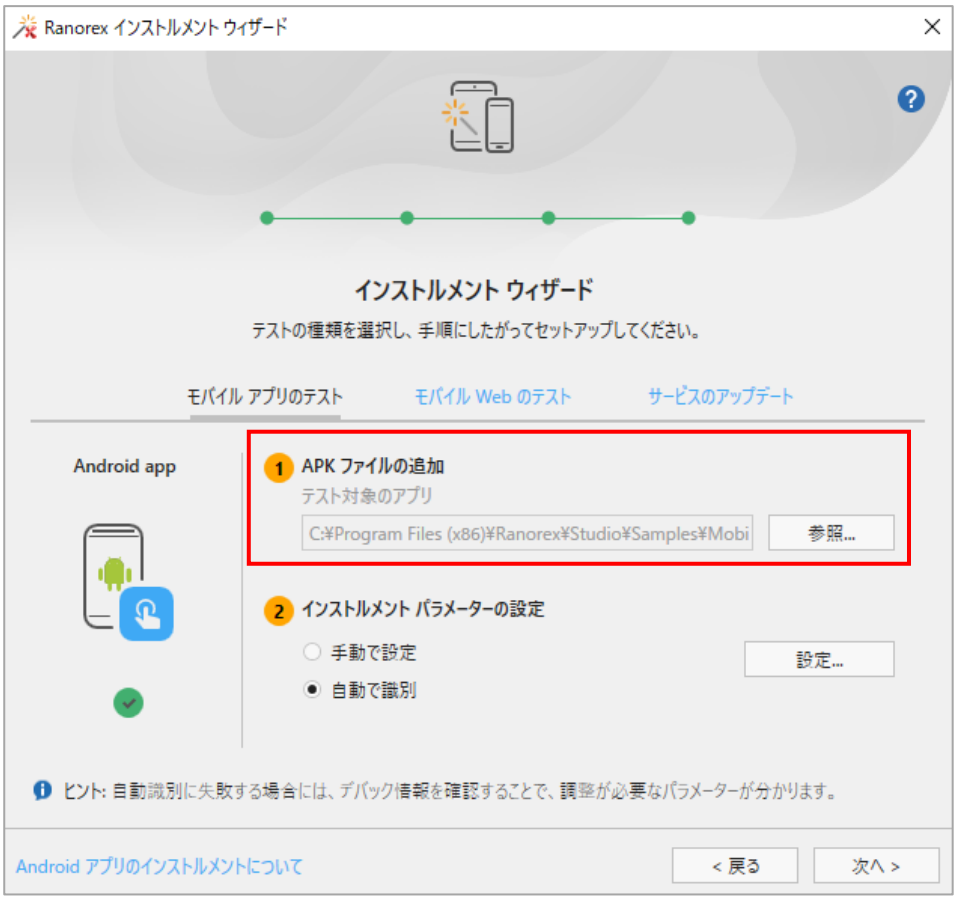
- APK ファイル（難読化されていないもの）

iOS 端末

- IPA ファイル（難読化されていないもの）
- P12 証明書
- プロビジョニングファイル

本演習では、Ranorex のモバイルサンプルアプリケーションとなる、 **KeePassDroid** を使用します。
KeePassDroid は、Ranorex をインストールした際に、一緒にインストールされています。

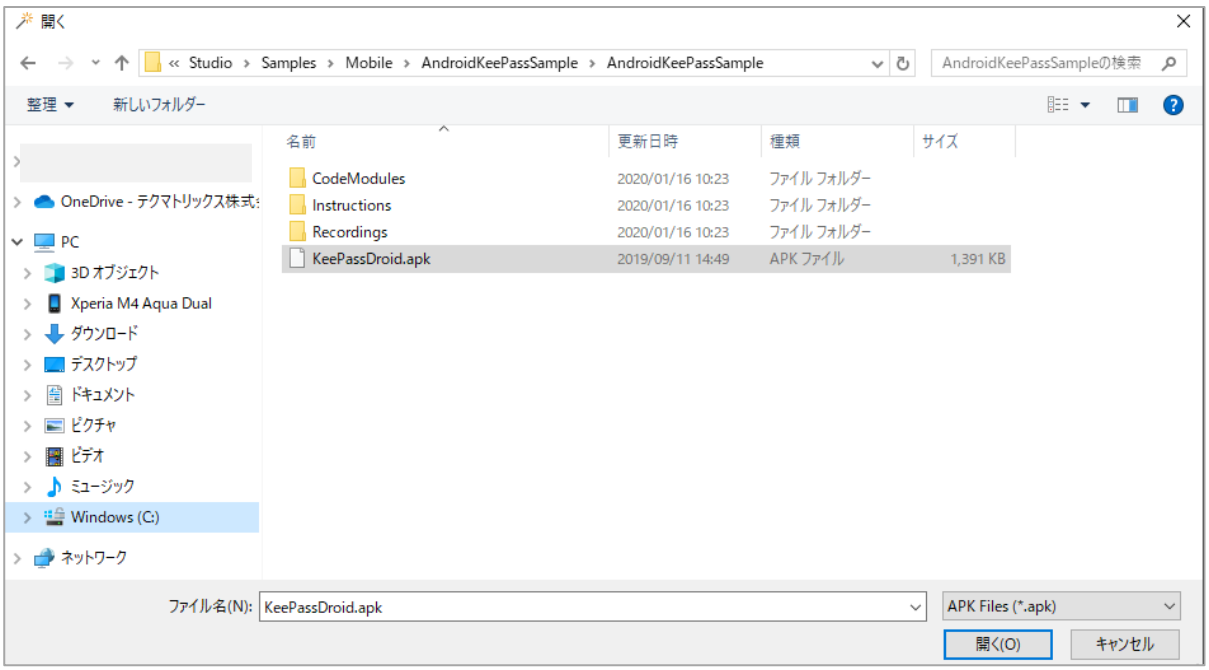
1. Ranorex Studio をインストールしたマシンと、対象のデバイスを USB（もしくは Wi-Fi）で接続します。
2. **ツール** – **インストルメント ウィザード** から、 **モバイルデバイスのインストルメント** を選択し、 **次へ** をクリックします。
3. **Android** を選択し、 **次へ** をクリックします。
4. **既存のデバイスから選択** から対象のデバイスを選択し、 **次へ** をクリックします。
5. **モバイルアプリのテスト** を開き、**APK ファイルの追加** を選択後、 **参照** ボタンをクリックします。



6. 起動されたエクスプローラーから、テスト対象のアプリケーション（APK ファイル）を選択します。

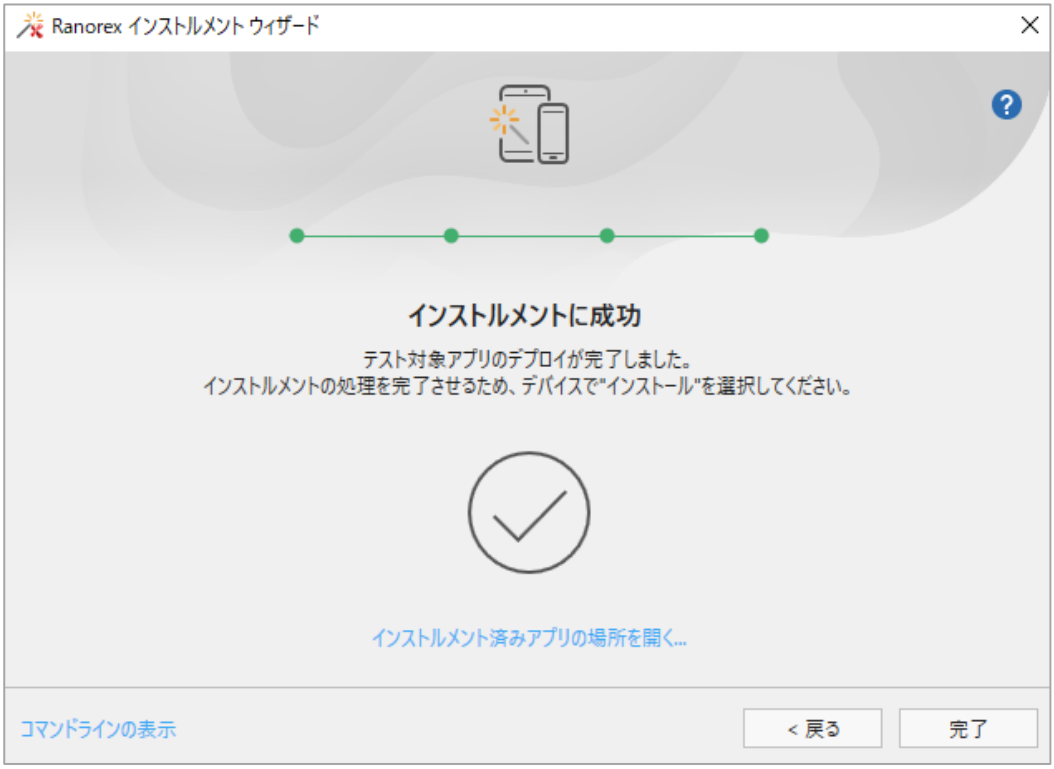
   以下のフォルダ配下にインストールされた **KeePassDroid.apk** を選択し、**開く** をクリックします。

   C:¥Program Files (x86)¥Ranorex¥Studio¥Samples¥Mobile¥AndroidKeePassSample¥AndroidKeePassSample



7. **次へ** をクリックします。

これにより、 **KeePassDroid** アプリケーションのインストルメントとデプロイがおこなわれ、対象のデバイス上には対象のアプリケーションがインストールされます。



> **注意**
>
> Ranorex でインストルメントされていないテスト対象アプリケーション（APK/IPA ファイル）が、既に対象のデバイス上にインストールされている場合には、デバイス上からアプリケーションを削除した上で、Ranorex インストルメントウィザードを使用し、対象アプリケーションのインストルメントとデプロイをおこなってください。

Lesson2 テストの作成と実行

この章では、Ranorex にてモバイルアプリケーションのテストを作成する方法についてご紹介します。

モバイルアプリケーションのテスト作成

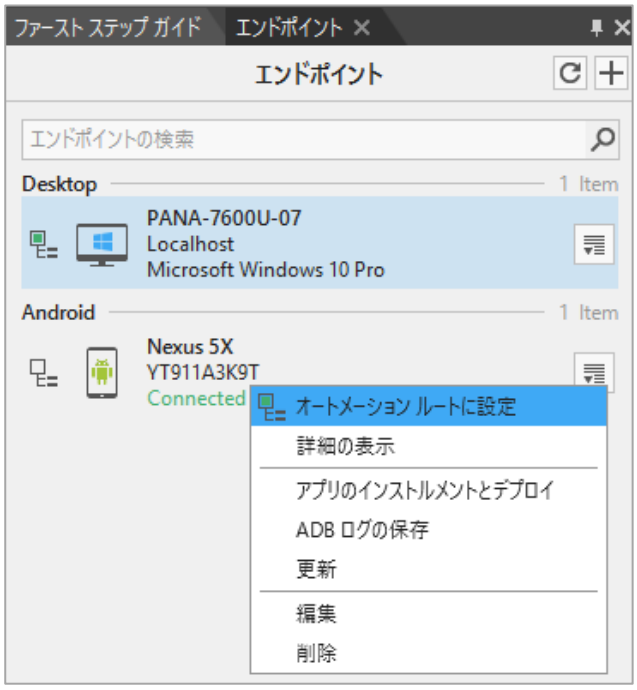
Ranorex にて、モバイルアプリケーションのテストを作成する場合、Ranorex Spy という機能を使用することでテストの作成をおこないます。Ranorex Spy とは、起動しているアプリケーション画面に表示された要素を読み込み、内部情報の解析をおこなうツールとなっているため、操作対象のオブジェクトを Ranorex Spy で取得しながら、テストシナリオの作成をおこないます。

> 参照
>
> Ranorex Spy については、以下のユーザガイドをご参照ください。
>
> https://www.ranorex.com/ja/help/latest/ranorex-studio-advanced/ranorex-spy/introduction/

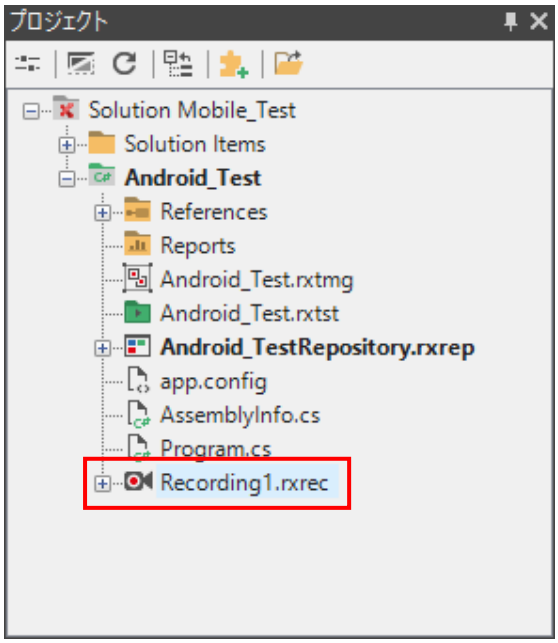
テストの作成

アプリケーションの起動

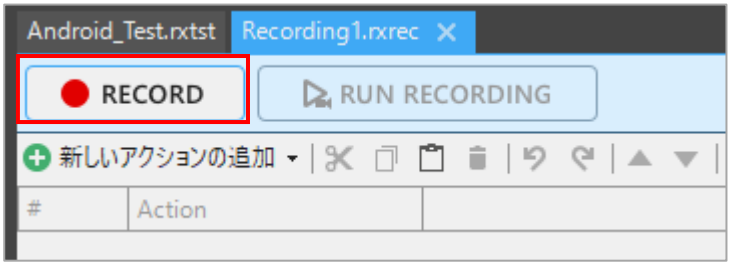
1. エンドポイント画面から、登録したデバイス（Nexus 5X）横の **設定** ボタンをクリックし、 **オートメーションルートに設定** を選択します。



2. 左上のプロジェクトビューから、 **Recording1.rxrec** を選択し、モジュールを開きます。



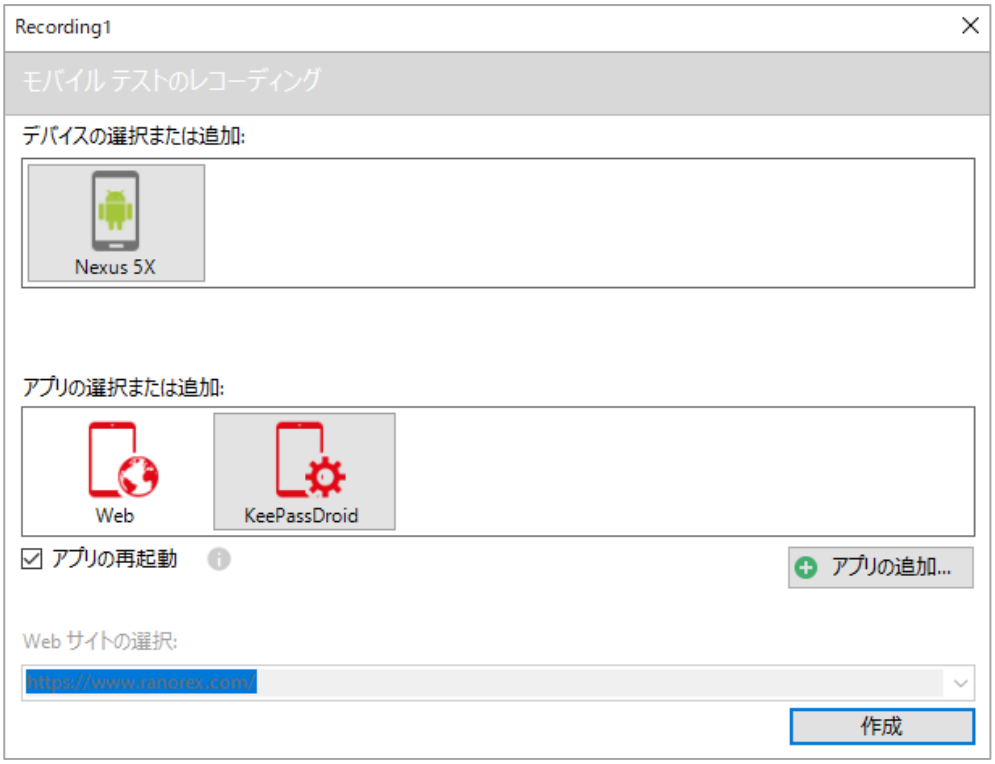
3. 画面の左上にある、 **RECORD** ボタンをクリックします。



4. **モバイル テストのレコーディング** 画面から、 対象のデバイス と テスト対象アプリケーション を選択し、 **作成** ボタンを選択します。

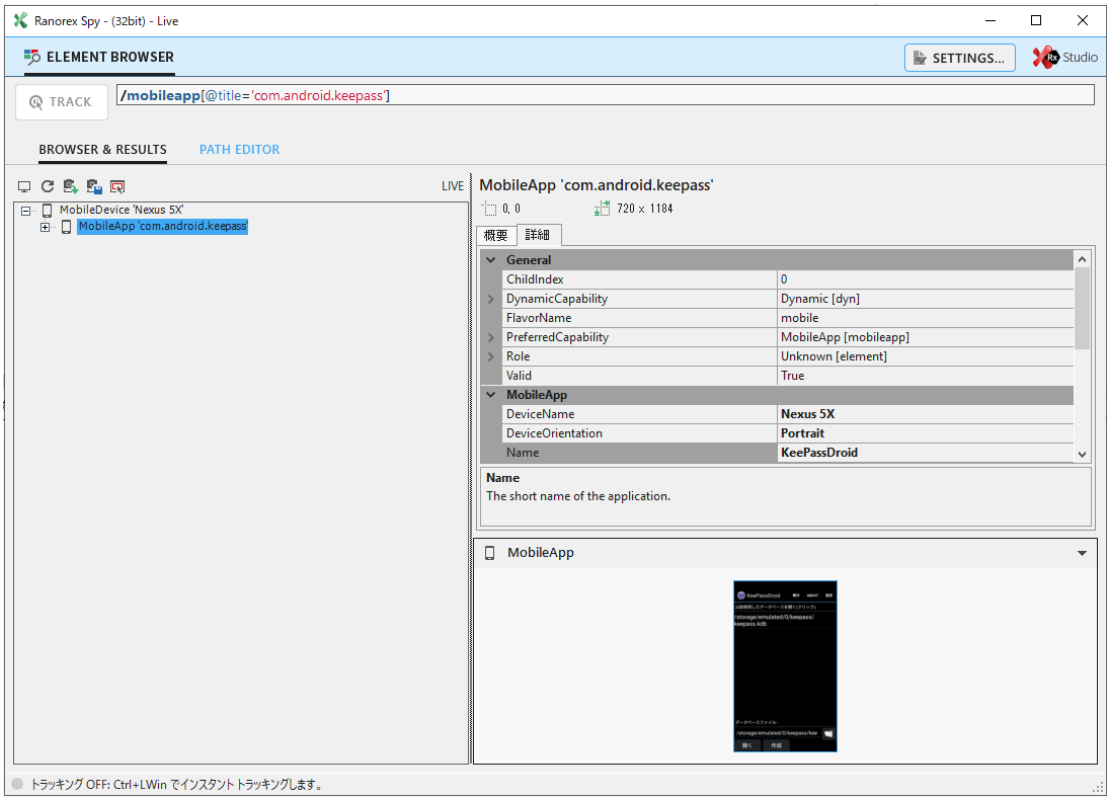
   デバイスの選択または追加：**Nexus 5X**

   アプリの選択または追加　：**KeePassDroid**



5. デバイス上では、選択したテストアプリケーション（KeePassDroid）が起動します。

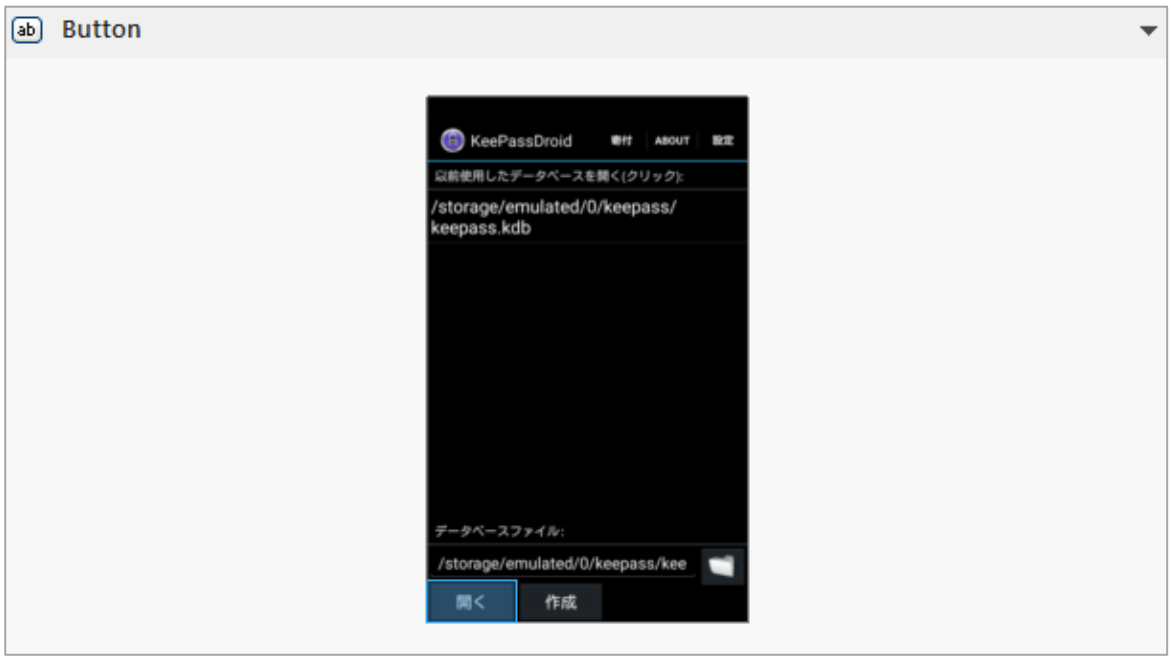
また、ホストマシン上には、Ranorex Spy の画面が表示されます。



テストシナリオの作成

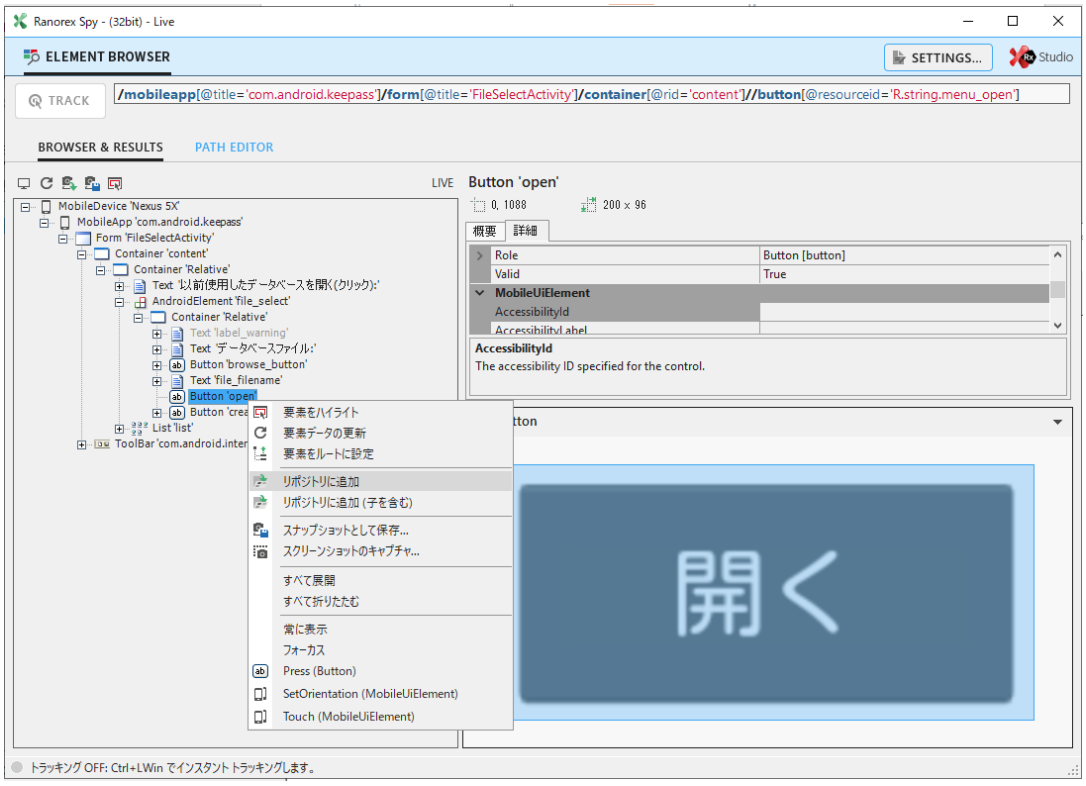
テストの作成は、以下の手順にておこないます。

1. Spy 画面の右下に表示されているイメージ画像から、 **開く** ボタンをクリックします。

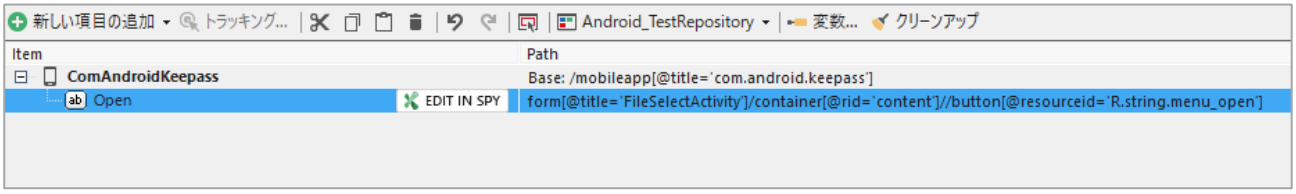


2. **開く** ボタンの要素を選択すると、左側の **BROWSERS & RESULTS** の一覧には、 **開く** ボタンの要素（**Button 'open'**）が青くハイライトされます。

   Button 'open' 要素を右クリックし、 **リポジトリに追加** を選択します。

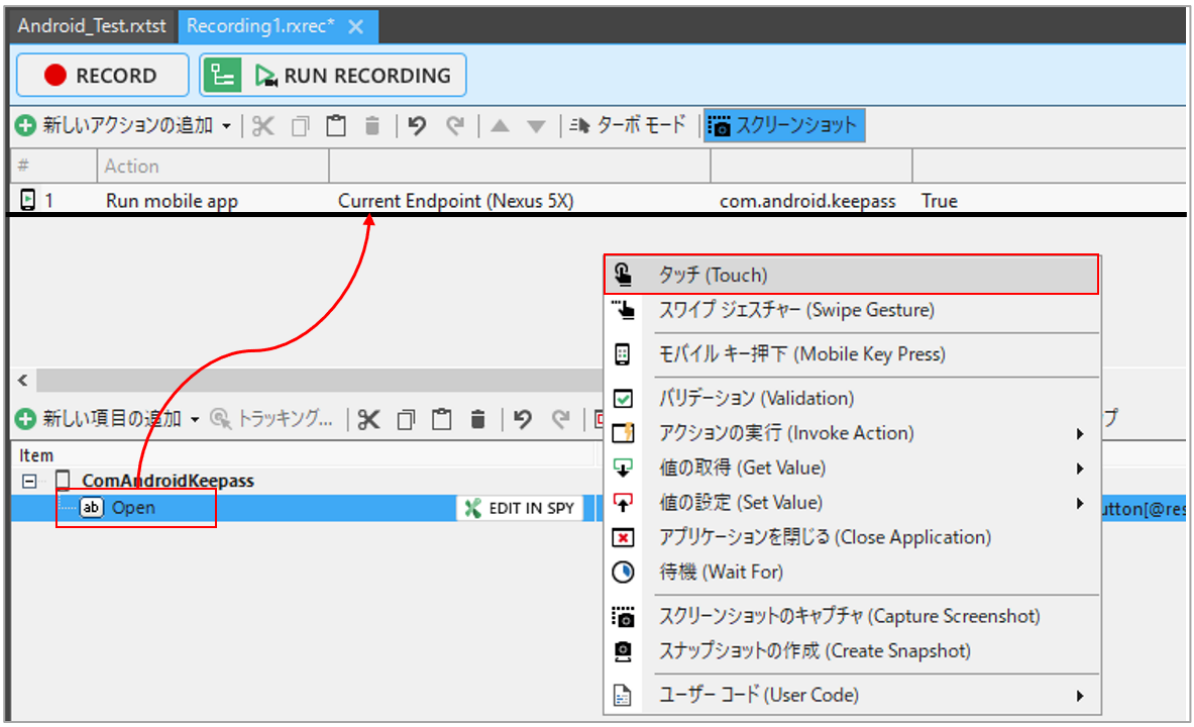


これにより、リポジトリ内には対象のオブジェクト（**Open**）が追加されます。

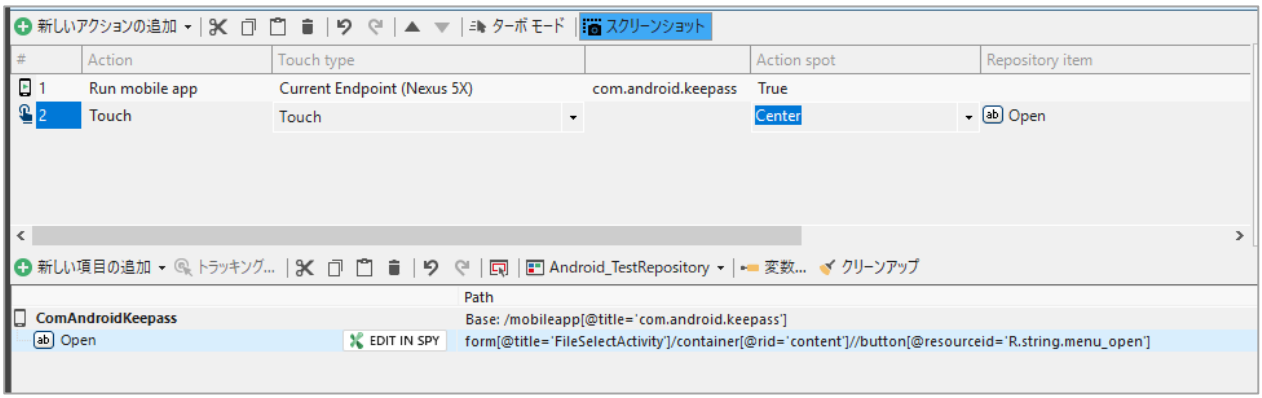


3. リポジトリ内から、追加したオブジェクト（**Open**）を選択し、アクションテーブルへドラッグアンドドロップをおこないます。

表示されたアクションの一覧から、 **タッチ（Touch）** を選択します。

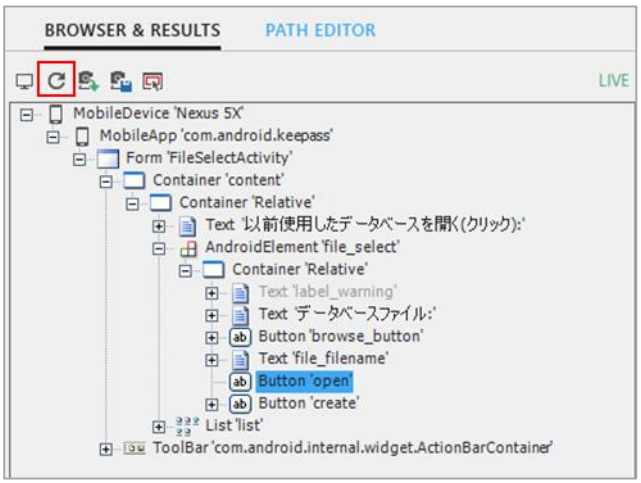


これにより、 **開く** ボタンをタッチするアクションが追加されます。



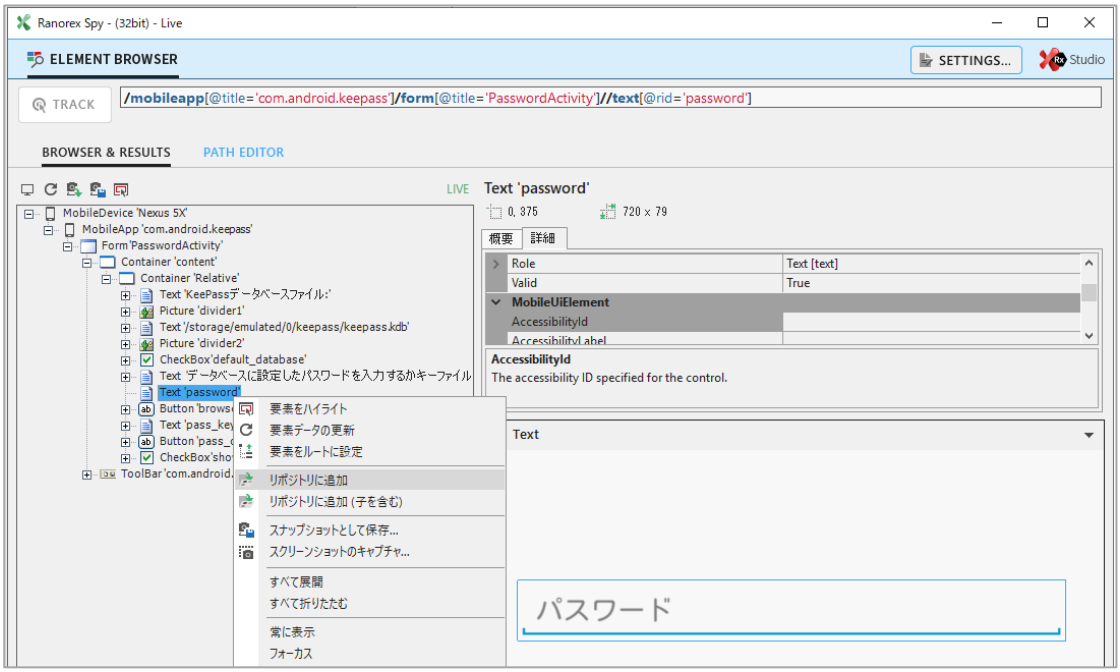
4. デバイス上で **開く** ボタンをクリックし、次の画面を表示します。

   デバイス上の画面遷移をおこなった後、Spy 上でも情報の更新をおこなうため、Spy 画面左上にある **更新** ボタンをクリックします。

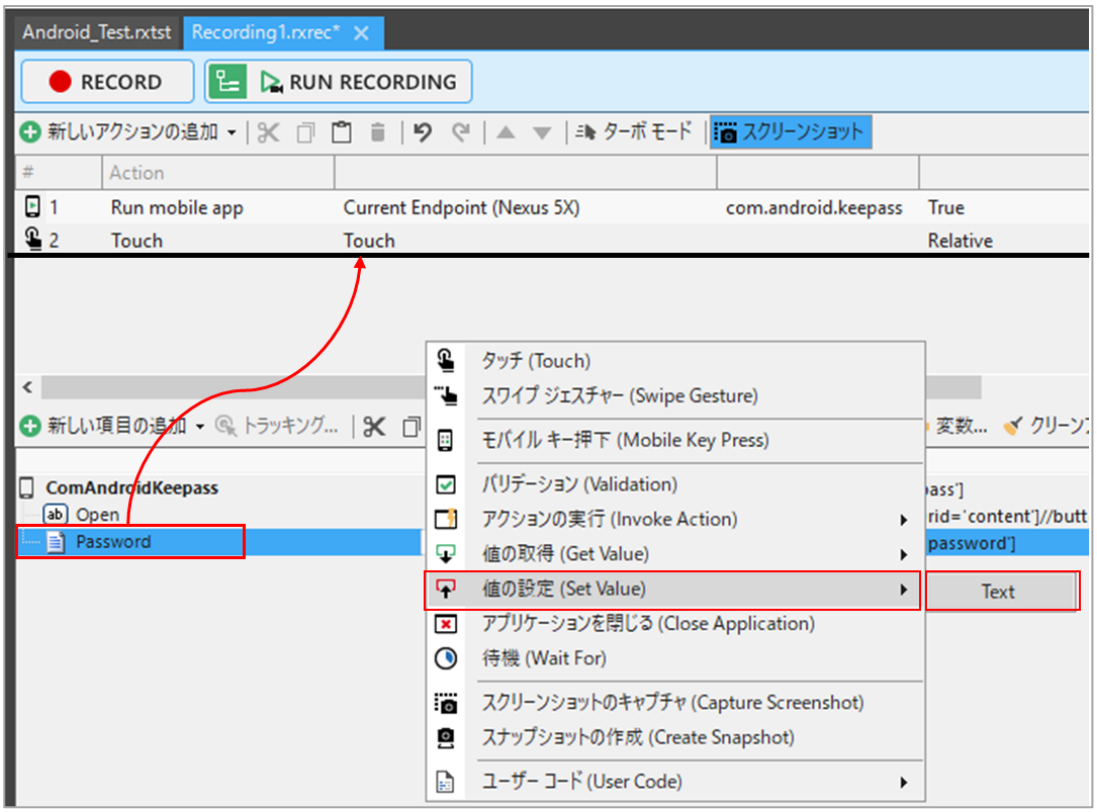


   ここからは、手順 1 ～ 5 の手順を操作対象となるオブジェクトごとに繰り返します。リポジトリ内に対象のオブジェクトを追加する場合、デバイス上に対象のオブジェクトを表示した状態（操作対象となる画面を表示した状態）とし、Spy 画面からオブジェクトの取得/追加をおこないます。

5. Spy 画面の **BROWSERS & RESULTS** の一覧から、パスワードの入力フォーム（**Text 'password'**）を選択し、右クリックをおこない、 **リポジトリに追加** を選択します。

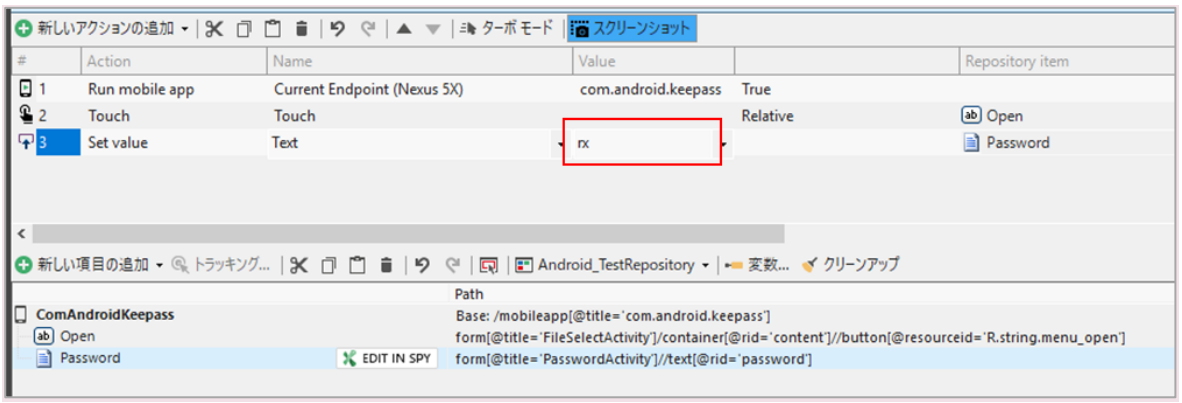


6. 追加したオブジェクト（**password**）を選択の上、アクションテーブルへドラッグアンドドロップをおこない、アクションの一覧から **値の設定（Set Value）** - **Text** を選択します。

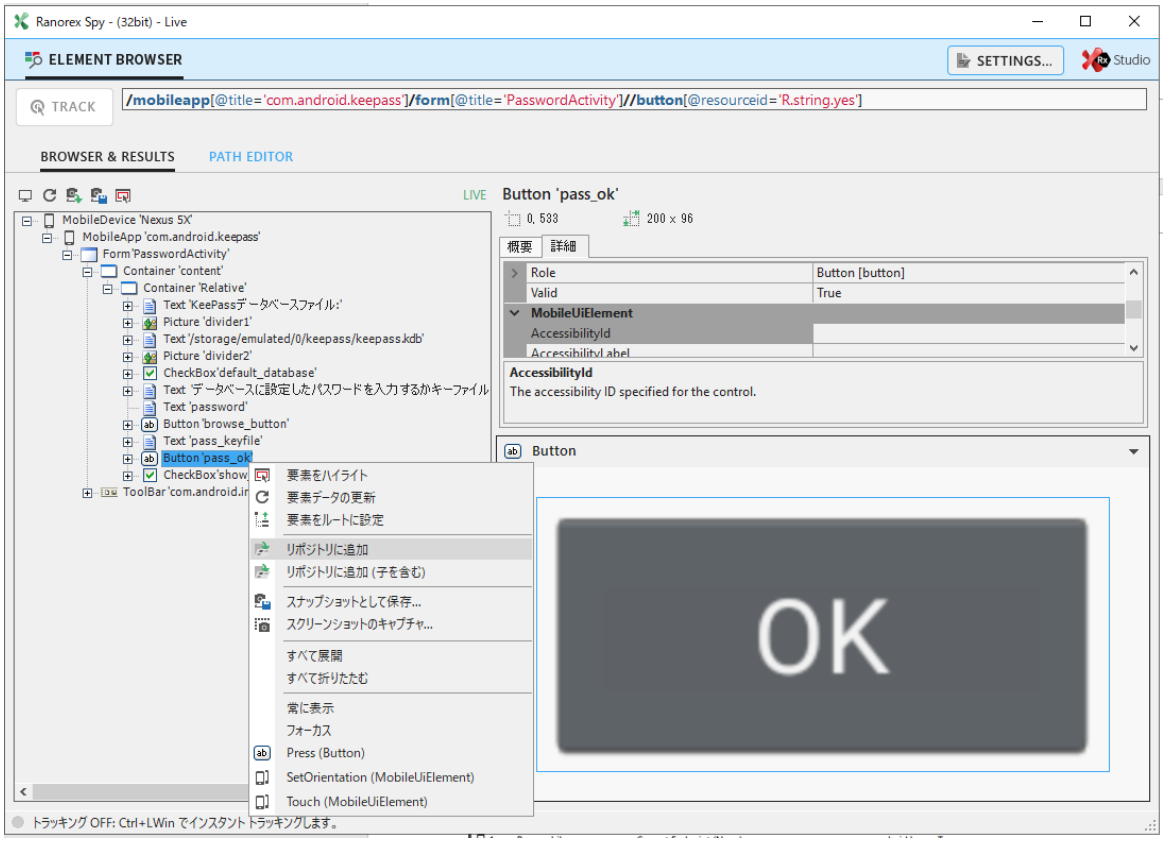


7. 追加した **Set Value** アクションにて、 **Value** 部分に以下の入力値を設定します。

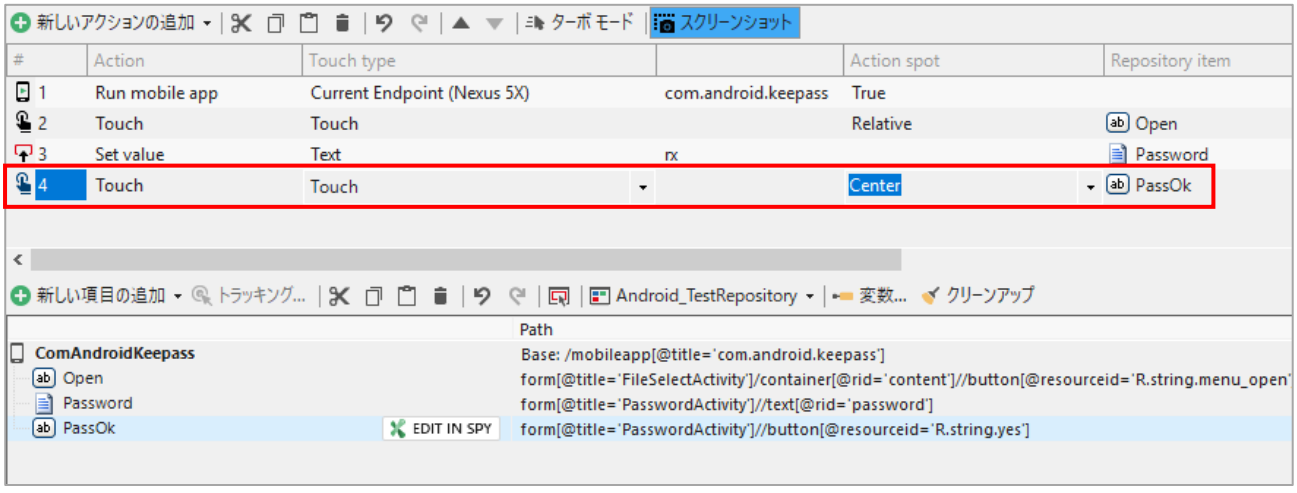
入力値：`rx`



8. Spy 画面の **BROWSERS & RESULTS** の一覧から、パスワードの入力フォーム（**Button 'pass ok'**）を選択し、右クリックをおこない、 **リポジトリに追加** を選択します。



9. 追加したオブジェクト（**PassOk**）を選択の上、アクションテーブルへドラッグアンドドロップをおこない、アクションの一覧から **タッチ（Touch）**を選択します。



これにより、KeePassDroid のログイン操作を作成できました。

バリデーションの設定

デバイス上に表示されている、 **OK** ボタンをクリックし、ログイン操作をおこないます。

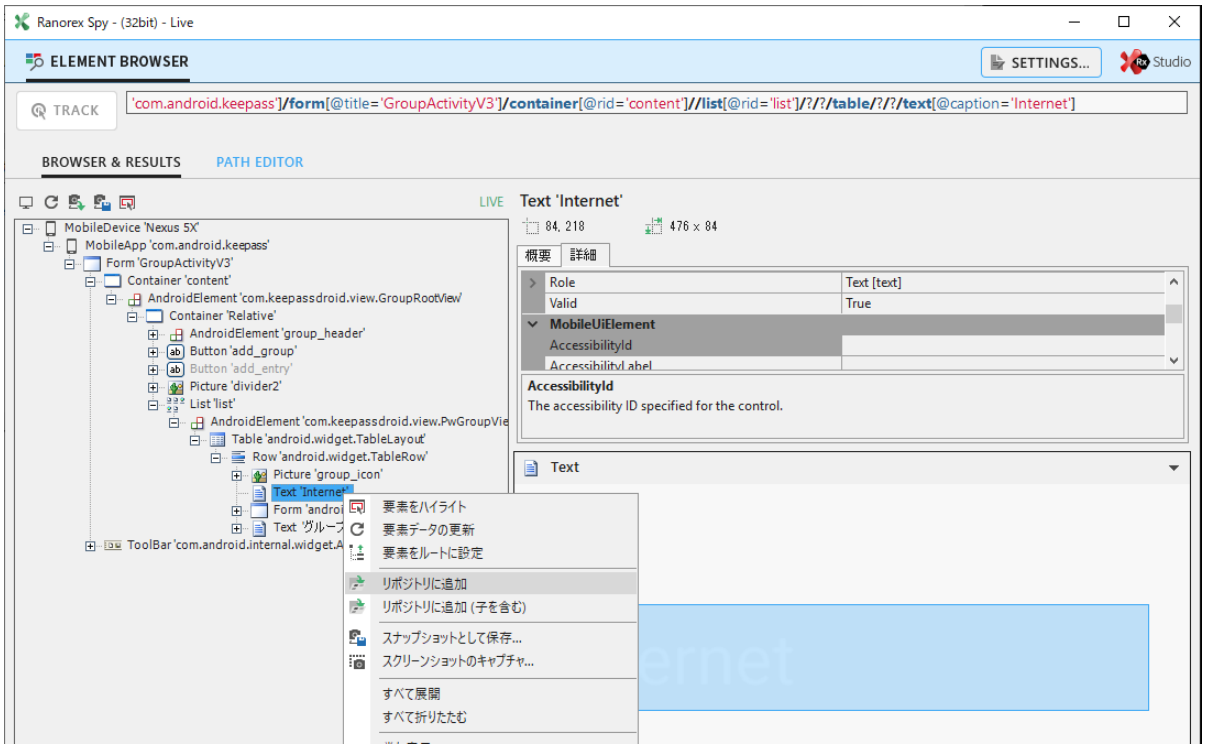
これにより、以下の画面が表示されます。

ここでは、ログイン後の画面に表示されている要素に対し、検証をおこないます。

本演習では、ログイン後の画面に表示された **Internet** という文字列が正しいかどうかの検証をおこないます。

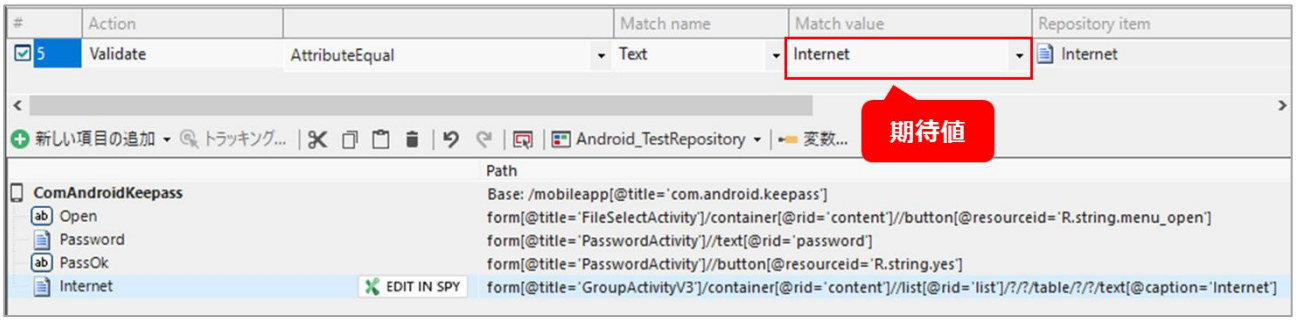
以下の手順にて、検証（Validate）アクションを追加します。

1. Spy 画面の **BROWSERS & RESULTS** の一覧から、 Internet グループ（**Text 'Internet'**）を選択し、右クリックをおこない、 **リポジトリに追加** を選択します。



2. 追加したオブジェクト（**Internet**）を選択の上、アクションテーブルへドラッグアンドドロップをおこない、アクションの一覧から **バリデーション（Validation）** を選択します。

本演習では、表示された文字列（**Internet**）が正しいかどうかの検証をおこなうため、Text 属性を使用します。**Match Value** に設定した値が、Ranorex での期待値となります。

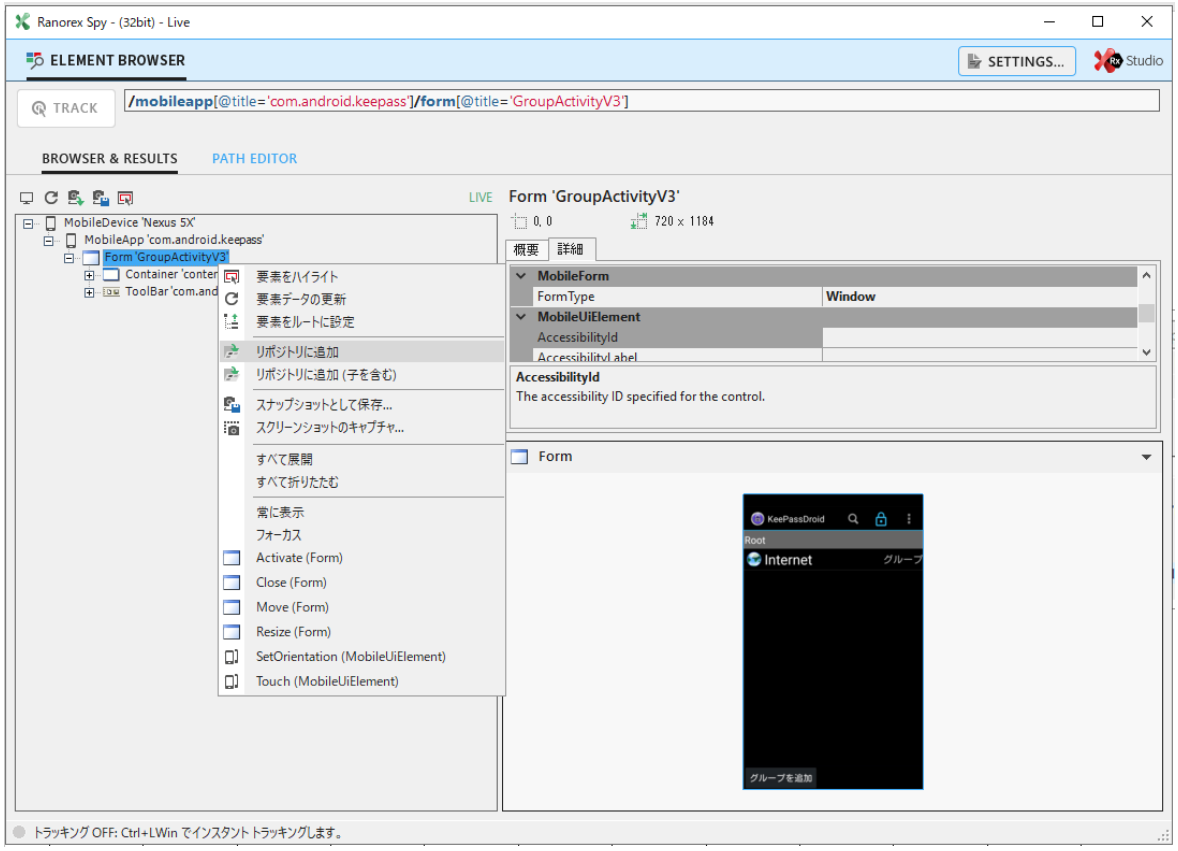


画面ショットの取得

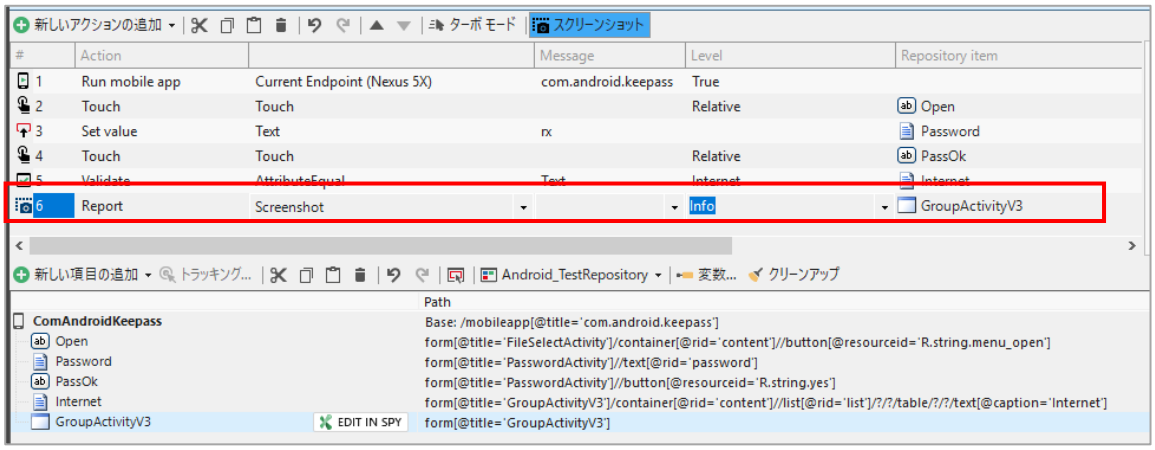
ログイン後の画面ショットを取得する操作を追加します。

今回は、画面全体のスクリーンショットを取得するために、Form 要素を使用したアクションを追加します。

1. Spy 画面の **BROWSERS & RESURTS** の一覧から、 Form 要素（**Form 'GroupActivityV3'**）を選択し、右クリックをおこない、 **リポジトリに追加** を選択します。



2. 選択したオブジェクト（**GroupActivityV3**）を選択の上、アクションテーブルへドラッグアンドドロップをおこない、アクションの一覧から **スクリーンショットのキャプチャ（Capture Screenshot）**を選択します。



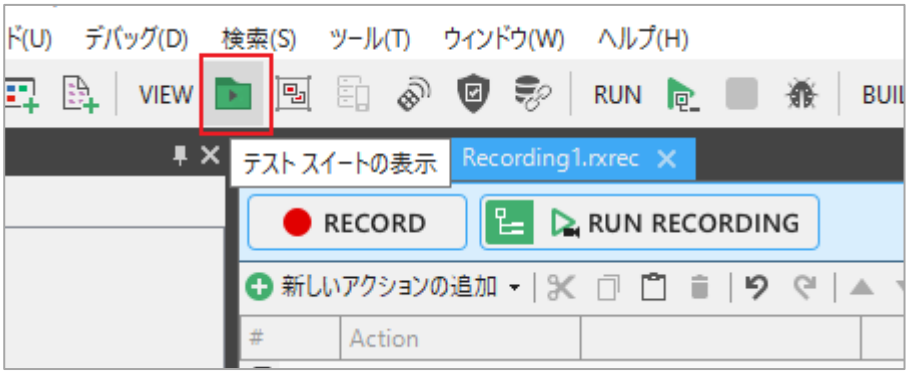
これにより、Form 要素に対するスクリーンショットが取得されます。

テストの実行

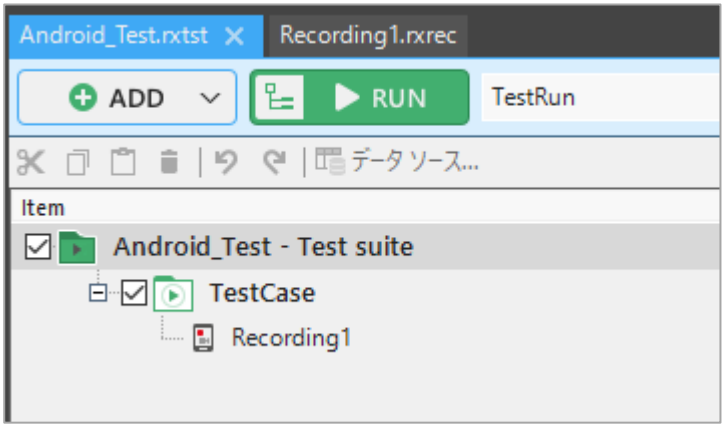
Ranorex Test Suite ビュー（rxtst ファイル）の 実行（**RUN**）ボタンから、テストを実行します。

Ranorex は、ユーザーが操作した操作（マウス移動/キーボードイベント）をシミュレートします。

1. Ranorex Studio のメニューバーから、 **テストスイートの表示** ボタンを選択します。



2. 表示された **Ranorex Test Suite** ビューにて、 **Android_Test – TestSuite** のチェックボックスが有効になっていることを確認後、 **RUN** ボタンをクリックします。

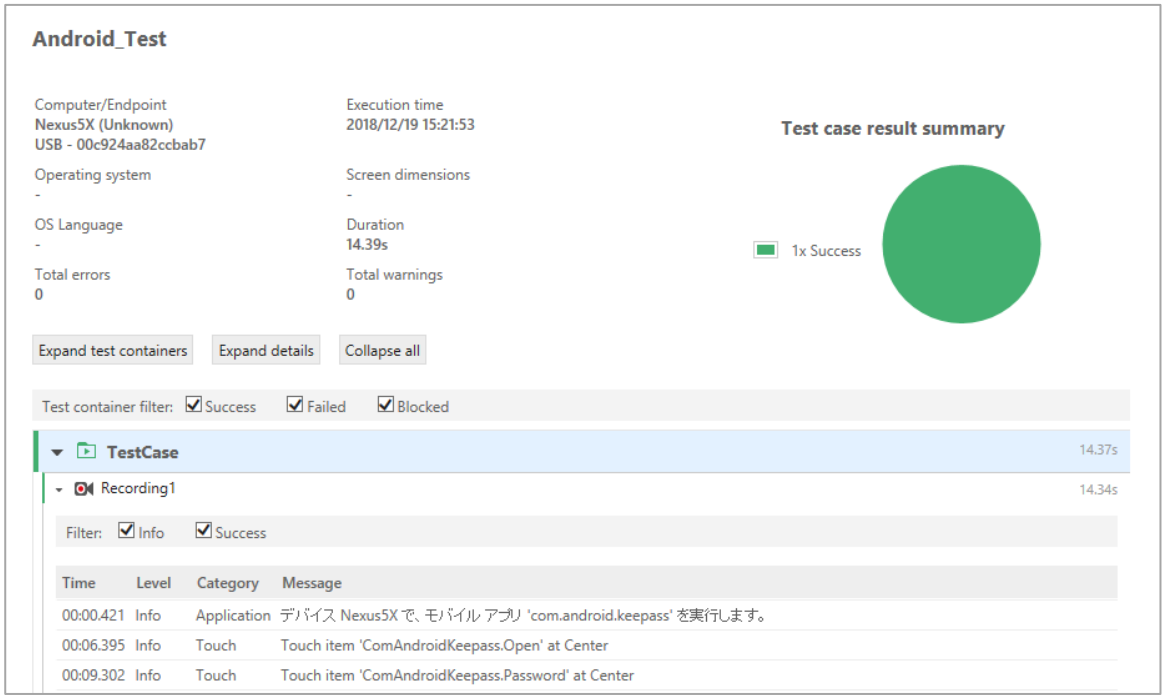


テスト結果レポートの確認

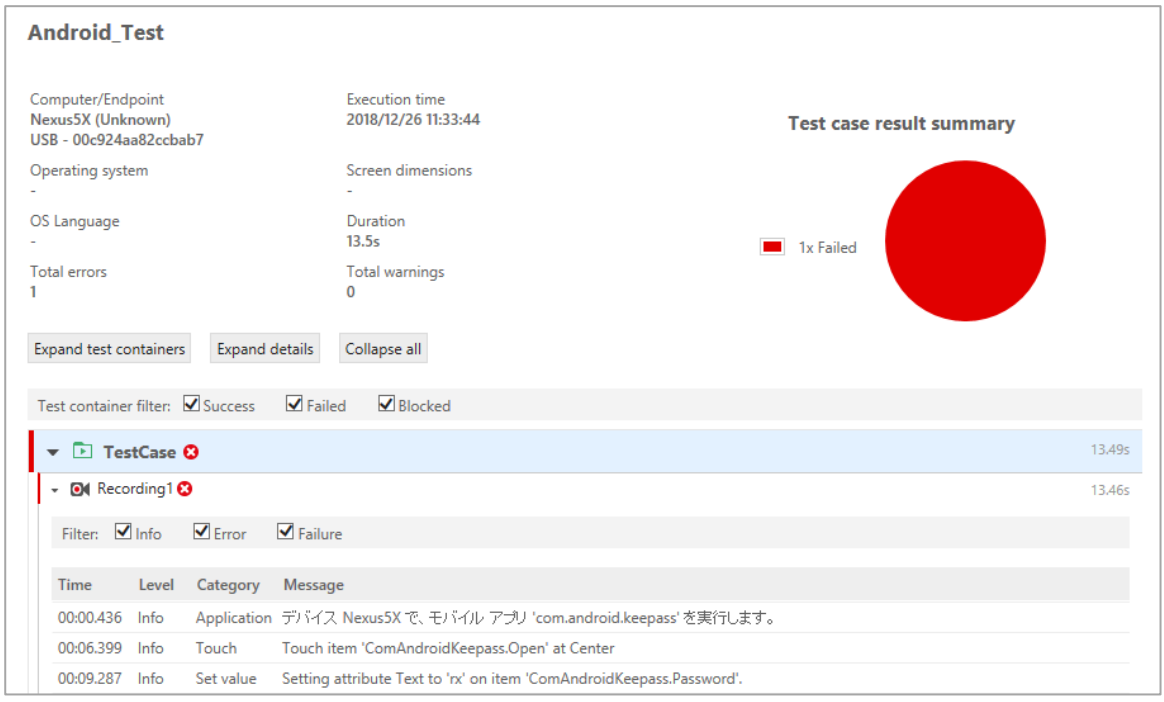
テスト実行後、テスト結果レポートを確認します。

テストが終了すると、Ranorex はテストレポートファイル（*.rxlog）を表示します。

【テスト成功時】

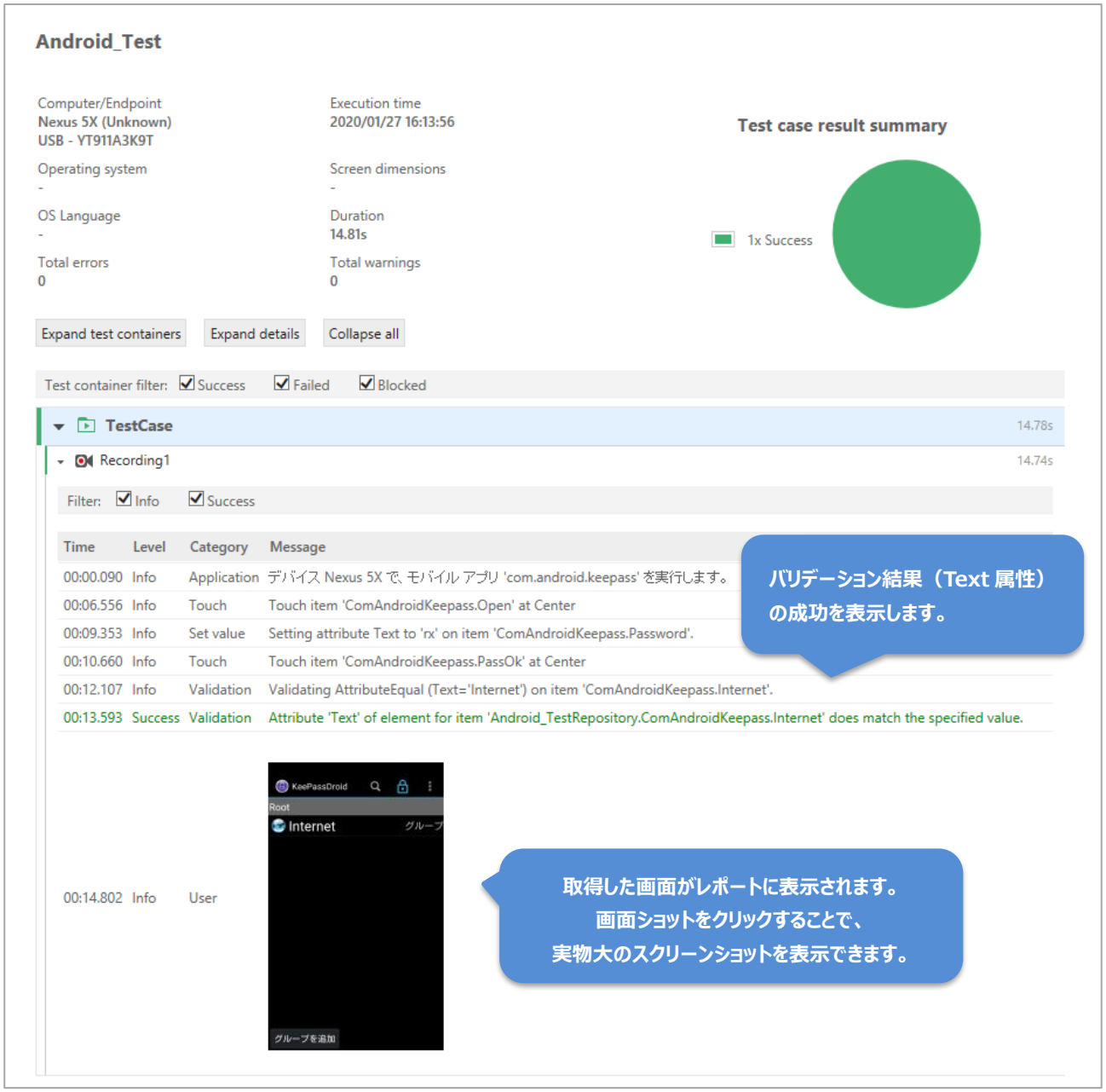


【テスト失敗時】



下図は、出力された成功レポートの内容です。

テストレポートでは、Validation が成功し、かつスクリーンショットも取得されていることが確認できます。



Appendix アクションのご紹介

アクションの待機

テスト実行中、アプリケーションによっては画面遷移に時間がかかることがあります。Ranorex には、画面遷移の完了まで次のアクションの実行を待機するための方法が用意されています。

- **Delay** アクション

Delay アクションで指定した時間だけ、待機します。

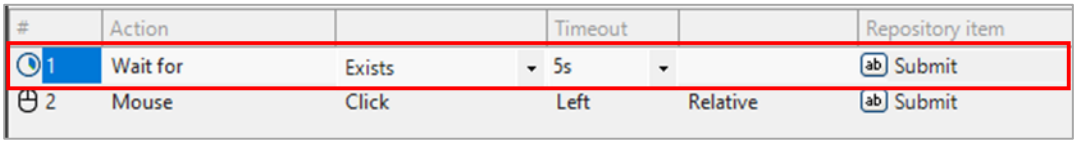
- **Wait For** アクション

指定したオブジェクトの存在/消失状態を待機するアクションです。

期待の画面に遷移すると、次のアクションが実行されます。

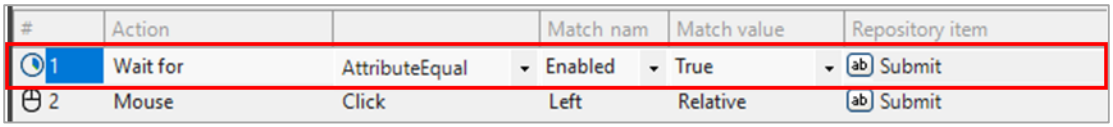
例：**Wait For – Exists** アクション

指定したオブジェクト（Submit）が指定のタイムアウト時間内に表示されるまで待機します。以下の例では、タイムアウト時間を 5 秒間としているため、5 秒以内にオブジェクトが表示されない場合にはエラーとなります。



例：**Wait For – AttributeEquel** アクション

指定のタイムアウト時間内に、割り当てられたリポジトリ アイテムの属性 **Match Name** が指定の値 **Match Value** と等しくなるまで待機します。今回は、ボタンの活性/非活性を表す **Enabled** 属性を使用することで、対象のボタンが有効化される（True になる）まで待機します。

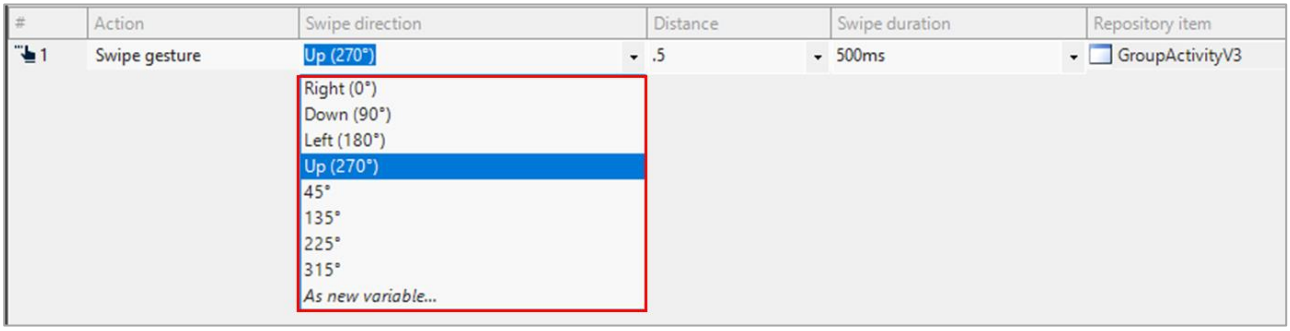


スワイプ操作

Ranorex でスワイプ操作を行いたい場合には、レコーディングモジュールを開き、 **新しいアクションの追加** – **スワイプジェスチャー（Swipe Gesture）**アクションを追加します。

追加した **Swipe gesture** アクションの **Repository Item** が **(No item)** となっているため、スワイプ対象の要素（たとえば、Form 要素や Dom 要素など、画面全体を表す要素）を設定します。

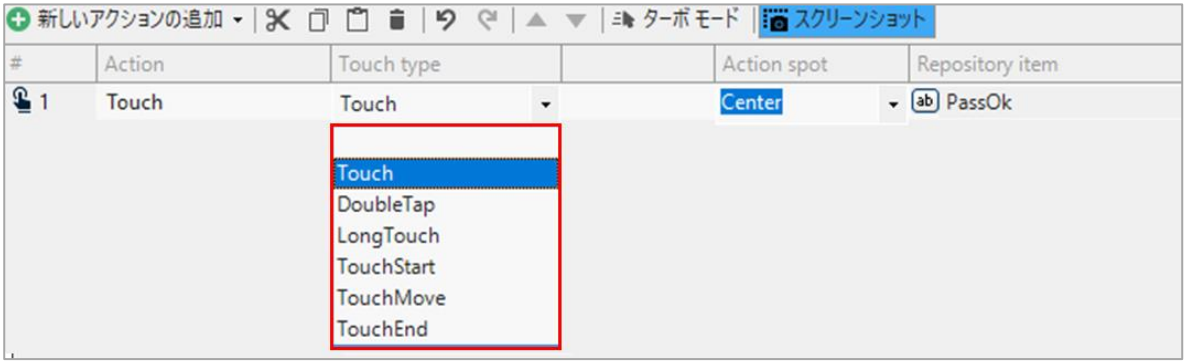
引数となる **Swipe direction** にて、スワイプの方向を指定することができます。



例：**Swipe gesture – Up（270°）**

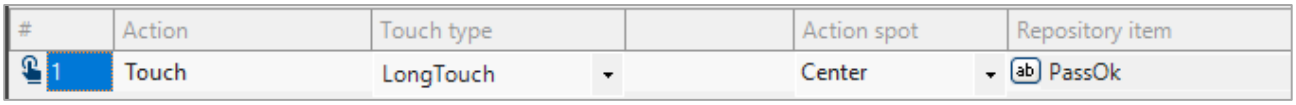
タッチ操作

Ranorex では、Touch アクションの中でもダブルタップや、ロングタッチなど様々な種類のタッチアクションを用意しています。追加した Touch アクションに設定されている Touch type から Touch の種類を選択できます。



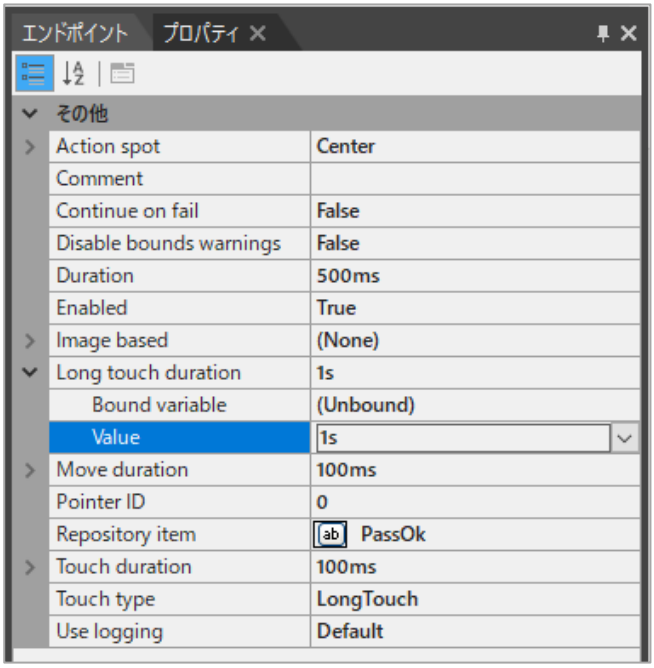
例：**Touch – LongTouch** アクション

対象のオブジェクトを長押ししたい場合には、 **LongTouch** を使用します。



また、各アクションに対する様々な設定は、 **プロパティ** 画面からおこなえます。

たとえば、長押しをおこなう時間を変更したい場合、アクションのプロパティ **Long touch duration** の **Value** を長めの値に変更します。 **プロパティ** 画面は、対象のアクションを右クリックし、表示されたメニューから **プロパティ** を選択することで表示できます。



注意

iOS では、Touch アクションの Touch type は Touch のみに対応しています。

※Touch type の Double Tap / Long Touch / Touch Start / Touch Move / Touch End に対応していません。また、Touch type – Touch を使用し、プロパティ値 Touch Duration の値を変更した場合であっても、ロングタッチはおこなえません。

※Android では、すべての Touch Type に対応しています。

テクマトリックス株式会社
ソフトウェアエンジニアリング事業部

Tel : 03-4405-7853
E-Mail : ranorex-info@techmatrix.co.jp