

Mobile: Action Steps (Aktionsschritte)

Kurzübersicht

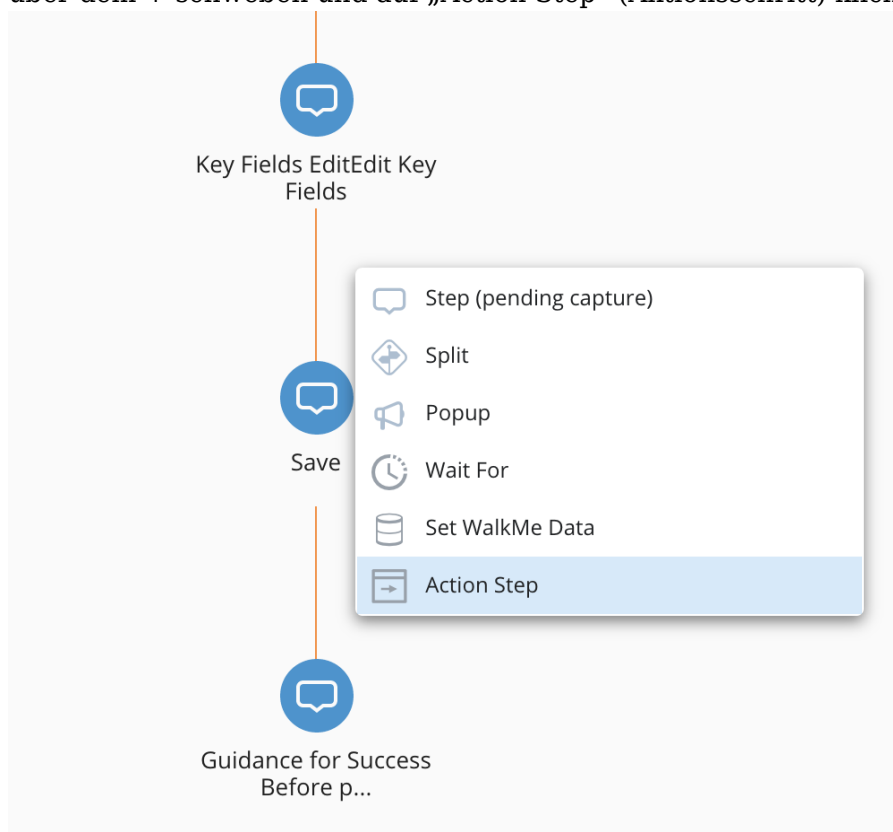
Action Steps sind ein Step-Typ, der beim Einsatz von Automationen sehr nützlich ist. Die Fähigkeit, im Sinne von Endnutzern agieren zu können, verbessert die Produktivität und die Time-to-Value für App-Benutzer. Zuvor unterstützte WalkMe Mobile Sprechblasen- und Banner-Schritte sowie Auto-Tap-Schritte. Der erste hinzugefügte „Action Step“-Typ ist „Scroll“, und in der Zukunft werden mehr Aktionsarten dazu kommen.

Anwendungsfälle

- Auto-scroll zu einem bestimmten Element-„Typ“

Schritte zum Hinzufügen des Action Steps

1. Erstellen Sie oder modifizieren Sie in der Mobile-Konsole ein SWT, indem mit dem Mauszeiger über dem + schweben und auf „Action Step“ (Aktionsschritt) klicken.



2. Wählen Sie „Type“ (Typ), „Sub Type“ und „Value“ (Wert)

Type:

Scroll

▼

Sub Type :

Callback

▼

Value :

CANCEL

OK

Typ	Sub-Typ	Wert
Scrollen	Callback (Rückruffunktion)	Typ des Elements (d. h. UILabel)

So aktivieren Sie Action Steps (Aktionsschritte)

Unten finden Sie eine Beschreibung aller erforderlichen Schritte zur Einbindung von Action Step Callbacks.

Um Kampagnen-Callbacks registrieren zu können, implementieren Sie zuerst dieses Protokoll:

- iOS

```
/**
 * Interface definition for a callback to be invoked in Campaign
 * actions.
 */
@protocol WMCampaignInfoDelegate <NSObject>
/**
 * Called after campaign was dismissed.
 *
 * @param campaignInfo The dismissed campaign info.
 */
@optional
- (void)campaignDidDismiss:(WMCampaignInfo *)campaignInfo;
/**
 * Called right before the campaign is about to be shown (will not be
 * called on Power Mode preview)
 *
 * @param campaignInfo The shown campaign info.
 */
@optional
- (void)campaignWillShow:(WMCampaignInfo *)campaignInfo;
/**
 * Return a value for SWT Action step of type "Callback"
 *
 * @param type the action type (ex. scroll).
 * @param campaign The shown campaign info.
 * @param args array of values from the Value field on the console
 * (comma separated values).
 */
@optional
- (void)campaign:(WMCampaignInfo *)campaign didPerformAction:(NSString
*)actionType withArgs:(NSArray<NSString *> *)args andCompletion:(void
(^)(__nullable id result))completion;
@end
```

- Android

```
public interface WMCampaignInfoListener {
/**
 * Called right before the campaign is about to be shown (will not be
 * called on Power Mode preview)
 *
 * @param campaignInfo The shown campaign info.
 */
void onCampaignPresented(WMCampaignInfo campaignInfo);
/**
 * Called after campaign was dismissed.
 *
 * @param campaignInfo The dismissed campaign info.
 */
void onCampaignDismissed(WMCampaignInfo campaignInfo);
/**
 * Called on action step
 *
 * @param campaignInfo The campaign info.
 * @param actionType The action type.
 * @param args String array of arguments,
 * @param completionStepListener The listener to receive the result.
 */
void onCampaignAction(WMCampaignInfo campaignInfo, String actionType,
String[] args, WMCampaignActionListener completionStepListener);
}
```

Nach der Implementierung des Protokolls registrieren Sie sich bei den Kampagnen-Callback-Ereignissen:

- iOS

```
/**
 * Register a delegate to campaign events
 *
 * @param delegate The delegate
 *
 */
+ (void)setCampaignInfoDelegate:(id<WMCampaignInfoDelegate>)delegate;
```

- Android

```
/**
 * Register a listener to campaign events
 *
 * @param campaignInfoListener The listener
 *
 */
public static void setCampaignInfoListener(WMCampaignInfoListener
campaignInfoListener)
```

Schließlich implementieren Sie die Kampagnenaktionsmethode, um den gewünschten Wert für die Aktion zurückzugeben:

- iOS

```
- (void)campaign:(WMCampaignInfo *)campaign didPerformAction:(NSString *)actionType withArgs:(NSArray<NSString *> *)args andCompletion:(void (^)(id _Nullable))completion {
    if (args.firstObject) {
        NSLog(@"Value for action type: %@ with args: %@", type, args.firstObject); // Value for action type: scroll with args: ProductCardTypeClass
    }
    NSIndexPath *indexPath = [NSIndexPath indexPathForRow:4 inSection:1];
    if (completion) {
        completion(indexPath);
    }
}
```

- Android

```
@Override
public void onCampaignAction(WMCampaignInfo campaignInfo, String actionType, String[] args, ABBI.WMCampaignActionListener completionStepListener) {
    if (args != null && args.length > 0) {
        Integer index = null;
        // Calculate the index based on the provided args
        index = myCardIndexCalculator();
        if (completionStepListener != null) {
            completionStepListener.onActionCompleted(index);
        }
    }
    else {
        if (completionStepListener != null) {
            completionStepListener.onActionCompleted(null);
        }
    }
}
```

Anforderungen/Einschränkungen

- iOS SDK Version 2.9.7 und Android SDK Version 2.10.9
- Ein regulär erfasster Schritt muss direkt auf den „Scroll“ Action Step folgen. Dies liefert den Aktions-Kontext, um zu den passenden Containern zu scrollen, in denen das Element existiert.