

Technical Disclosure Commons

Defensive Publications Series

July 2022

Obtaining User Consent and Indicating Progress During a Sequence of Automated Actions

Keun Soo Yim

Adam Coimbra

Follow this and additional works at: https://www.tdcommons.org/dpubs_series

Recommended Citation

Yim, Keun Soo and Coimbra, Adam, "Obtaining User Consent and Indicating Progress During a Sequence of Automated Actions", Technical Disclosure Commons, (July 07, 2022)

https://www.tdcommons.org/dpubs_series/5250



This work is licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/).

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

Obtaining User Consent and Indicating Progress During a Sequence of Automated Actions

ABSTRACT

Automation of actions within a software application can improve user experience, e.g., by enabling repetitive complex sequences to be performed with no user effort. Such actions can include high-level operations, such as actions to be performed via the user interface (UI) of the app. Automation is typically achieved by requesting that a trusted automation service perform each of the high-level actions, including UI input in non-privileged apps running in user space, in the desired sequence on behalf of the user. This disclosure describes techniques to display appropriate UI indicators and get user consent when needed when a sequence of higher-level actions within an app is automated via a trusted automation service, such as a virtual assistant. If any action within the automated sequence of actions requires user consent, such consent is obtained via the automation service.

KEYWORDS

- App automation
- Task automation
- Automation service
- App API
- Action sequence
- Virtual assistant
- User consent
- Consent dialog

BACKGROUND

Automation of actions within a software application can improve user experience, e.g., by enabling repetitive complex sequences to be performed with no user effort. Such actions can include high-level operations, such as actions to be performed via the user interface (UI) of the app. Automation is typically achieved by requesting that a trusted automation service perform each of the high-level actions, including UI input in non-privileged apps running in user space, in

the desired sequence on behalf of the user. In essence, the automation service is an interface between the app in which the user seeks to automate actions and the device operating system.

DESCRIPTION

While a sequence of automated actions is ongoing, users are typically informed of the progress via appropriate UI indicators, such as sound alerts, notification bubbles, progress bars, etc. The length of time that the UI progress indicators last depends on the number and length of the individual actions within the automated sequence. In practice, depending on the task, the time can vary from a second to several minutes. In some cases, the sequence of actions being automated can involve intermediate steps that generate pop-up dialogs in which the user's consent is needed prior to proceeding to the next step. In such cases, the user must explicitly consent manually before the rest of the automated sequence can proceed.

This disclosure describes techniques to display appropriate UI indicators and get user consent when needed when a sequence of higher-level actions within an app is automated via a trusted automation service, such as a virtual assistant. When a user requests that a sequence of actions within an app be automated, the request invokes the automation service to perform each action in sequence.

As each individual action within the automation sequence is performed via the automation engine by requesting the corresponding operation from the underlying operating system, suitable UI indicators are generated while the action is being performed. Such indicators can be auditory, e.g., sound alerts, and/or visual, e.g., notification bubbles. Different types of indicators can be employed at different stages within the automation sequence. For instance, the first and last actions within the sequence can be associated with a sound alert while the

intermediate steps result in a corresponding notification bubble shown at the top of the device screen.

If any action within the automated sequence of actions requires user consent (or other input), such consent is obtained via the automation service when the automation is invoked via any of the following ways:

- explicit interaction with a UI button, e.g., tap, long press, corner swipe, etc.;
- Application Programming Interface (API) call, e.g., remote procedure call;
- typed request; or
- tapped request, e.g., tapping on a suggestion, link, etc.

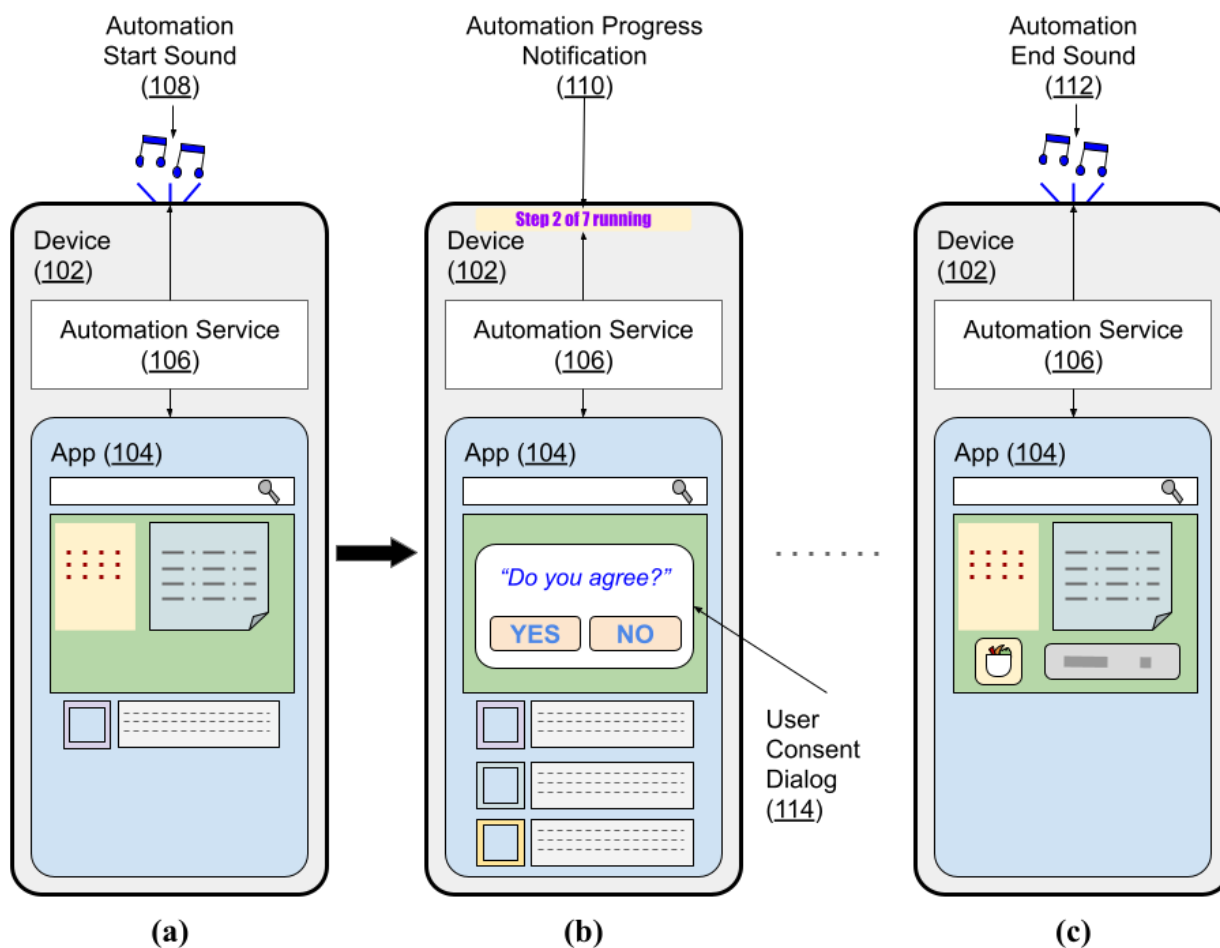


Fig. 1: UI indicators for individual actions within an automated sequence

Fig. 1 shows an example of operational implementation of the techniques described in this disclosure. While using an app (104) on a device (102), a user invokes an automation service (106) to automate a sequence of actions. As shown in Fig. 1(a), an automation start sound (108) is played to indicate that the first action within the sequence is initiated via the automation service. As shown in Fig. 1(b), as subsequent actions are ongoing, corresponding progress notification bubbles (110) are shown at the top of the screen with user permission. During the process, the trusted automation service can be used to automatically respond to dialogs that seek user consent (114), or if the user has configured the automation service to seek user input, to seek such input. As shown in Fig. 1(c), the end of the automation sequence is indicated by an end sound (112).

Each action within a sequence can be associated with one or more suitable indicators. The type of indicator(s) associated with the various actions within a sequence can be set by the developers and/or specified by the users and/or determined dynamically at runtime.

User consent input can be skipped when the user invokes automation mode in any of the following ways:

- voice-based invocation, e.g., via a wake word for a virtual assistant;
- voice query; or
- physical action, e.g., performing a physical gesture detected by device hardware, such as squeezing the device.

Automation mode for performing actions within an app can be enabled or disabled in one any of the following ways:

1. Use of different APIs for actions performed via an automation service compared to those performed normally;

2. Maintaining an operating system flag that indicates automation status individually for each app or globally for all apps;
3. Augmenting API calls for each action by addition of a parameter that indicates whether the action is to be handled by an automation service or processed normally; or
4. Using a default mode (e.g., automation) to perform an action as long as the action is supported in that mode and if not supported, using other modes.

If a user request does not result in invoking the automation service for an action, the relevant app is used to perform the action as usual, without automation. If for any reason the requested action cannot be performed, the user can be shown an error message to that effect.

With user permission, the techniques described above can be implemented within an app, device, operating system, and/or platform that includes automation capabilities. The techniques can be provided via any virtual assistant (or other application) that supports user requests for automating sequences of actions within apps. Implementation of the described techniques can serve to raise user awareness of the progress of various actions within an automated sequence. Moreover, the techniques can make it more seamless for users to provide consent during sequences of automated actions, thus enhancing the user experience (UX).

Further to the descriptions above, a user may be provided with controls allowing the user to make an election as to both if and when systems, programs or features described herein may enable collection of user information (e.g., information about a user's apps, user information requested by an app, a user's preferences, or a user's current location), and if the user is sent content or communications from a server. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be

determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so that a particular location of a user cannot be determined. Thus, the user may have control over what information is collected about the user, how that information is used, and what information is provided to the user.

CONCLUSION

Automation of actions within a software application can improve user experience, e.g., by enabling repetitive complex sequences to be performed with no user effort. This disclosure describes techniques to display appropriate UI indicators and get user consent when needed when a sequence of higher-level actions within an app is automated via a trusted automation service, such as a virtual assistant. If any action within the automated sequence of actions requires user consent, such consent is obtained via the automation service.

REFERENCES

1. Whitney, Lance. "How to Automate Your Life With Apple's iOS Shortcuts App."
<https://www.pcmag.com/how-to/how-to-automate-your-life-with-apples-ios-shortcuts-app>
accessed Jan 12, 2022.
2. LlamaLab. "Automate: Make your phone or tablet smarter with automation."
<https://llamalab.com/automate/doc/security.html> accessed Jan 12, 2022.