

Sonos Labs Self-Test Plan

Overview

Welcome to the Sonos Labs self-test plan. This document explains the areas of the Sonos system that each new music partner must test in house.

First, automated tests are designed to check the back-end functionality of your implementation without using the Sonos app. Running the automated tests is required and a clean run must accompany your submission. Go to the bottom of the page [API DOCUMENTS & TOOLS](#) and download the Python Self-Test Suite. For details about the automated tests, see the README from the download.

Second, manual tests using the Sonos app verify portions of your implementation. For these test you [Add your service with customSD](#) to allow you to browse your service with the Sonos app and you also use the [SoapUI](#) program to view the metadata exchange between your service and the Sonos app. Sonos recommends you run the manual tests before submission.

Minimum Required Equipment

If you will be running manual tests, below is the minimum required equipment.

Hardware

At least one of the following:

- PLAY:1
- PLAY:3
- PLAY:5 (formerly the S5)
- CONNECT:AMP (formerly the ZonePlayer 120) with attached speakers
- CONNECT (formerly the ZonePlayer 90) with external amplifier

Sonos Software

At least one of the following installed:

- SONOS Controller for Mac or PC
- SONOS app for iPhone
- Sonos app for iPhone can also be installed in the iPad
- SONOS app for Android

Manual Test Cases

Test cases are listed below by feature area. Sonos recommends that you run these manual test cases if you implement any of the functionality listed. Run only those tests that are applicable for your service.

Account Management

- Add a trial type account.
- Add supported account types.
- Add an unsupported account type.
- Add a deleted account type.
- Add an expired supported account type.

Search

- Verify Search functionality in all applicable containers.
- Verify multi-library search if applicable.
- Verify Incremental Search functionality in all applicable containers (each new character entered should process a new search).
- Verify Search functionality using special characters and accented characters in all applicable containers.
- Verify error messages returned in Search results.
- Verify artwork returned in Search results.
- Verify Metadata returned in Search results.

Browse Music

Browse through all available containers:

- Click through to the leaf node of each container successfully (successful completion requires no browse errors, timeouts or corrupted data).
- Verify that browsing happens within a reasonable timeframe.
- Verify that the metadata returned within the browse containers is correct and includes album art URIs where applicable.
- Verify that all content including artist, album, track, playlist names and ordering match your web site.
- Verify the artwork renders correctly in browse containers.
- Verify that all appropriate nodes are generated in a container (i.e. the All Tracks node).
- Verify that pagination works in large containers.

Playlist Editing Using the Sonos App

- Create a playlist from an album, track, or playlist
- Edit a playlist
- Remove a playlist
- Rename a playlist
- Reorder a playlist

Favorites

- Add a favorite track
- Remove a favorite track
- Add a favorite album
- Remove a favorite album
- Check that favorites added or removed via the service website are represented
- Ensure that updates happen in a reasonable timeframe.

Status Ratings

- Verify selecting the positive rating button rates the content successfully
- Verify selecting the negative rating button rates the content successfully
- Verify the correct success message is shown when selecting the positive ratings button
- Verify the correct success message is shown when selecting the negative ratings button

Dynamic Ratings

- Verify rating a track causes the ratings icon to change to match the rating.
- Verify that re-rating a track causes the rating icon to update to match the new rating state.
- Verify a previously rated track shows the current ratings state on the Now Playing screen when replayed.
- Verify the correct messaging is shown when a track receives a positive rating.
- Verify the correct messaging is shown when a track receives a negative rating.

Custom Error Handling

- Verify the service is returning a custom error code when a specific error condition is met (you can use SOAP UI to do this verification)
- Verify the controller shows the correct custom error message when a specific error condition is met.