PSVITAMAN

Alpha 3 · User's Guide

Install, listen, and keep your library.

Created by ObsoleteSony. Powered by PocketJS.

PS Vita · Local FLAC, MP3, WAV and AAC

For the complete supported source profiles, visit obsoletesony.com/psvitaman/audio-compatibility.

Install and update

Use a homebrew-enabled PS Vita with VitaShell. Install PSVITAMAN-v0.1.0-alpha.3.vpk from the official release ZIP. The application uses title ID PSVTM0001.

1. Fully close PSVITAMAN from LiveArea before updating.

2. Extract the ZIP on your computer. Copy the VPK to ux0:/downloads/ on the Vita.

3. Open the VPK in VitaShell and install over the existing application.

4. Keep ux0:/data/PSVITAMAN/ intact. It holds your settings, favorites, history, playlists, library cache and resume state. An update does not require clearing it.

5. Launch PSVITAMAN and let the startup stages finish. Larger or changed libraries take longer than an unchanged warm launch.

The included SHA256SUMS.txt verifies the package contents. Download releases from https://www.obsoletesony.com/psvitaman/releases.

Add songs and choose sources

Put supported FLAC, MP3, WAV and AAC files in ux0:/data/PSVITAMAN/MUSIC/, including subfolders. For AAC use supported M4A or MP4 containers.

Settings offers PSVITAMAN Folder, Vita Music and Both. Both is the current default. PSVITAMAN Folder reads ux0:/data/PSVITAMAN/MUSIC/. Vita Music reads ux0:/music/ recursively. Both combines them while keeping songs with identical relative paths distinct.

Vita Music is read-only. PSVITAMAN does not rename, move, retag or delete native music, and does not use or change Sony's music database. All PSVITAMAN-owned saved data stays under ux0:/data/PSVITAMAN/. If native music is temporarily unavailable, its absence is not treated as permission to erase your saved state.

After adding, removing or replacing music, use Settings > Rebuild Library. Progress identifies the current step and shows available counters. The visible library stays usable during a background update; a failed update retains the last good library. Allow the operation to finish before closing. The product limit remains 10,000 songs.

Browse and return

Browse Songs, Albums, Artists, Folders, Favorites, History and Playlists. Album and artist grouping uses the files' metadata. Folder views distinguish the two local sources. Selecting a song starts playback from that point in its queue.

The contextual Back action (B) returns from Now Playing to the current browsing context, then back through its parent views. SELECT goes directly Home. Artist, album, song selection and scroll context are remembered within the current session. This browsing context is separate from saved song/resume state and is not promised across app restarts.

Confirm and Back use the Vita system's Cross/Circle assignment, or the swapped behavior selected in Settings. Hold Up or Down to move through a long list. The footer shows the current song and provides transport controls; open the player area to return to Now Playing.

Now Playing and information

Now Playing shows artwork, song title, artist, album, elapsed time and transport. Use Play/Pause, Previous, Next, shuffle and repeat. Drag the front-touchscreen seek control to move within a song. The Favorite heart adds or removes the song from Favorites. The optional EQ icon indicates EQ is enabled.

Titles use one fixed size. Longer titles may be shortened with an ellipsis; Track Information keeps complete details accessible. Artist and album text remain subordinate to the title.

Source captions follow one format everywhere: FLAC 44.1 kHz / 16 bit, WAV 48 kHz / 24 bit, MP3 320 kbps / 44.1 kHz, or AAC-LC 134 kbps / 44.1 kHz. Unknown values are omitted. Lossy files show bitrate, not the decoder's PCM bit depth. Estimated bitrate is identified honestly when no reliable declared bitrate exists.

Album bitrate uses the most common known bitrate among its songs; a tie selects the higher value. Unknown songs do not erase a useful known value. This album summary does not change any song's individual source facts. Mixed source formats are not presented as one uniform encoding.

Open Track Information for source codec/profile, sample rate, channels and available bit depth or bitrate details. Actual playback output is labeled separately. A high-resolution source is not a claim of native high-resolution Vita output.

Pocket Mode and system sleep

Tap the padlock in Now Playing to lock app controls while music continues. Short contacts remain locked. Hold the padlock continuously for about one second to unlock. Lock guidance appears below the Play button.

Settings > Show Lock Icon hides or shows the padlock. Hiding it clears any active lock. Settings > Disable L/R Triggers independently disables their normal app actions. Both settings are saved. System PS, power and volume controls remain system-owned.

Pocket Mode is not screen-off or background playback. System sleep stops playback; waking restores the application and playback after storage recovery. A brief recovery delay is normal. Resume, navigation and a fresh launch recover unlocked so you cannot be stranded behind a hidden lock.

Supported audio

FLAC: qualified stereo 16-, 24- and 32-bit sources at 44.1, 48, 88.2, 96 and 192 kHz. Compatible FLAC transitions support gapless playback.

MP3: MPEG-1 Layer III at 44.1 kHz, mono/stereo, CBR and VBR within the qualified decoder/admission scope. Alpha 3 adds bounded compatibility with the proven stacked-ID3 and zero-padding layouts. MP3 gapless playback is not guaranteed.

WAV: RIFF/WAVE PCM, 16- or 24-bit, mono/stereo, at 44.1 or 48 kHz.

AAC: AAC-LC in unencrypted, non-fragmented M4A/MP4 containers, mono/stereo, at 44.1 or 48 kHz.

See https://www.obsoletesony.com/psvitaman/audio-compatibility for the complete supported matrix. File extensions alone do not guarantee a supported profile. These local profiles do not imply support for every WAV/AAC encoding or every Jellyfin server delivery mode.

Artwork, EQ and visual modes

PSVITAMAN first uses supported embedded artwork. If that is unavailable, it checks the song's immediate folder for cover.jpg, cover.jpeg, cover.png, folder.jpg, folder.jpeg or folder.png, in that order and without case sensitivity. It does not search parent folders or the internet. Unsupported, malformed or oversized images can remain No Cover without preventing playback. Rebuild Library after changing folder covers.

The equalizer offers Flat, Bass, Treble, Vocal, Rock, Pop, Jazz, Classical and Custom. PREAMP is saved independently; CLIP warns of clipping. Reduce boosted bands or preamp when clipping appears. Playback speed is a separate control.

Spectrum visualizes current playback. Cassette View opens manually or automatically and offers eleven bodies with ten colors each. These are playback presentations. Use contextual Back to return from secondary views. Lyrics availability depends on the song's supported lyric data.

Jellyfin and troubleshooting

Network opens the configured Jellyfin library on a trusted private IPv4 LAN over HTTP. Server availability, authentication, delivery format and Wi-Fi conditions affect playback. Local-library compatibility does not guarantee every network delivery path. Very large Jellyfin artist lists remain a known limitation planned for a later update.

If new songs do not appear immediately, allow background reconciliation to finish or request Rebuild Library. The app reports its current work instead of requiring a cache reset. A failed or interrupted scan should keep the last good library. Do not delete your saved state merely to retry.

For a missing cover, check the embedded image or the supported direct-folder filenames. For a rejected song, compare the actual source profile with Audio Compatibility; a valid file outside the supported matrix is not necessarily corrupt.

Close through LiveArea when finished. For a reproducible problem, preserve the diagnostic log before launching repeatedly. Remove private media paths, server details and authentication data before sharing it. Include the app version, device model and reproduction steps at https://www.obsoletesony.com/psvitaman/report-a-bug.

Credits and distribution

PSVITAMAN is created by ObsoleteSony and powered by PocketJS. Audio uses libFLAC, minimp3, dr_wav and the Vita native AAC service. Fonts include Inter and Noto Sans CJK JP. See the included CREDITS.txt, THIRD_PARTY_NOTICES.md and LICENSES directory for upstream attribution and terms.

Official distribution: https://www.obsoletesony.com/psvitaman/. PSVITAMAN-owned code and artwork remain reserved to ObsoleteSony. Third-party licenses retain their own rights. PSVITAMAN is independent homebrew and is not affiliated with or endorsed by Sony, PlayStation or Jellyfin.