

# StoryBox ESP32 Development

**Project Timeline:** Delivery by January 31st

**Project Cost:** USD 800

## 1. FUNCTIONAL SCOPE

### 1.1 Hardware Platform

**Status:** YES, CONFIRMED

ESP32-S3-WROOM-1-N16 is fully supported.

### 1.2 Core Audio Playback

**Status:** YES, CONFIRMED

WAV format playback is straightforward using the ESP32-audio I2S library.

### 1.3 User Interface

**Status:** YES, CONFIRMED

Rotary encoder with interrupt-based reading and better response time. Circular navigation with infinite scrolling is easily implementable.

### 1.4 Audio Outputs & Headphone Detection

**Status:** YES, CONFIRMED

Dual output routing (speaker + headphone) with automatic switching.

### 1.5 Storage & File Management

**Status:** YES, CONFIRMED

MicroSD card with Structured folder hierarchy (Themes/Subthemes/Stories) enables clear organization.

**Note:** Client will provide labeled audio files with complete structure indicating playback sequence (story order, questions, navigation audio, control files, etc.) as reference samples.

## 1.6 No Audio Recording

**Status:** YES, CONFIRMED

No microphone or on-device recording simplifies firmware significantly. Parents record on smartphones and transfer via USB.

# 2. AUDIO STORAGE & FILE MANAGEMENT

## 2.1 USB Mass Storage

**Status:** TESTING REQUIRED

ESP32-S3 native USB with TinyUSB library enables MSC mode for drag-and-drop file transfer. The device will appear as a removable drive on all platforms.

**Important Note:** While the ESP32-S3 has the required TinyUSB library, there are known limitations. This feature will be tested thoroughly and only finalized after successful validation of all required functionality.

# 3. EDUCATIONAL QUIZ FEATURE (Q&A MODE)

## 3.1 Quiz Mechanics

**Status:** YES, CONFIRMED

Binary choice questions with infinite circular scrolling and audio feedback on each option selection. State machine implementation is well-defined and straightforward.

## 3.2 Validation & Feedback

**Status:** YES, CONFIRMED

Incorrect answers trigger encouragement audio and question repeat; correct answers play success audio and continue. JSON-scripted quiz flow provides flexibility for content creators.

## 3.3 Technical Implementation

**Status:** YES, CONFIRMED

State machine (NARRATION → QUESTION → ANSWER\_SELECT → VALIDATION → RESULT → NARRATION) using ArduinoJson library. Seamless audio transitions with minimal gaps

---

## 4. PERSONALIZED STARTUP AUDIO MESSAGE

### 4.1 Implementation

**Status:** YES, CONFIRMED

Simple file check at `/system/welcome.wav` with conditional playback. Parent-recorded audio transferred via USB MSC requires no firmware recording complexity.

## 5. BATTERY MONITORING & AUDIO WARNING

### 5.1 Implementation

**Status:** YES, CONFIRMED

ADC-based battery voltage monitoring with 3.3V default threshold. Low battery triggers `/system/low_battery.wav` playback automatically.

## 6. OTA FIRMWARE UPDATE

### 6.1 Implementation via USB

**Status:** YES, CONFIRMED

Over-The-Air (OTA) firmware updates will be implemented via USB connection for easy firmware upgrades without specialized equipment.

## RISK ASSESSMENT SUMMARY

- **Low Risk:** Audio playback, UI controls, battery monitoring, startup message
- **Medium Risk:** Quiz feature state machine, file management rescanning
- **Testing Required:** USB Mass Storage (TinyUSB limitations need validation)

Please make sure the number of audios and stories are not in too many numbers. Like just a few stories and a few questions etc. So that it'll be easier for me to implement them.