# **Spacebuzz Dashboard Documentation**

## **Overview**

This document describes the HTML structure, functionalities, and styles used in the VR Sync Dashboard. The dashboard serves as a control panel for managing VR devices, including video playback, device status, and other interactions through a WebSocket connection.

## **File Description**

* **File Name**: dashboard.html
* **Purpose**: To provide a user interface for controlling and monitoring VR headsets connected to the system.

## **HTML Structure**

### **Head Section**

* **Meta Tags**: Set character encoding to UTF-8 and viewport settings for responsive design.
* **Title**: "VR Sync Dashboard" to identify the purpose of the page.
* **Stylesheets**: External stylesheet output.css for general styles and an internal style block specifically for styling progress bars and color themes.

### **Body Section**

#### **Navigation Bar**

* **Logo**: Spacebuzz logo as a fixed element in the navbar for branding.
* **Mobile Menu Button**: A hamburger menu for mobile viewports that toggles visibility of a hidden navigation menu.
* **Step Indicator**: A horizontal list displaying steps (e.g., Restart, Intro, Start) with progress indicators.

#### **Main Content Area**

* **Device Table**: A dynamically populated table showing connected VR devices with details such as battery status, connection status, and controls for language and video commands.

## **Styles and Interactivity**

### **CSS Styles**

* **Progress Bar**: Enhanced visual feedback through color transitions.
* **Theme Colors**: Usage of Tailwind CSS classes for background colors and button states.

### **JavaScript Functionality**

* **WebSocket Integration**: Establishes a WebSocket connection to receive and send data in real-time.
* **Dynamic Device Table**: Updates the device information dynamically based on WebSocket messages.
* **Interactive Steps**: Allows users to manually advance through predefined steps and send commands to VR devices.
* **Responsive Menus**: Implements interactive menus for mobile and desktop views, including a collapsible mobile menu.

## **WebSocket Communication**

* **URL**: ws://192.168.20:5500/
* **Functions**:
  + ws.onopen: Confirms the establishment of a WebSocket connection.
  + ws.onmessage: Handles incoming messages and updates the device table accordingly.
  + ws.onerror: Logs errors related to the WebSocket connection.

## **Event Listeners**

* **Button Interactions**: Listeners on control buttons (e.g., play, pause, stop) to send commands through WebSocket.
* **Navigation Toggle**: JavaScript logic to show and hide the mobile navigation menu.

## **External Resources**

* **Tailwind CSS**: Utilized for rapid UI development and responsive design.

There are 3 steps in the navigation bar

Step 1 pressed: this resets the ride  
Step 2 pressed: this activates the introduction video

Step 3 pressed: this starts the video on all the connected headsets

You can play/ pause/ stop the video per headset if you want to

