

model_scene.svelte + model_viewer.svelte + model_debug.ts — 3D Model Viewer

These three files together form the 3D model display system used on the dashboard.

model_scene.svelte

The outer container and lifecycle manager. Handles: delayed initialisation (100ms timer to ensure the DOM is ready before the WebGL canvas is created), WebGL context loss detection, error state with a retry button, and a resize nudge that forces the canvas to reflow correctly after mount.

Uses `model_debug.ts` to persist error state across the component lifecycle using `localStorage` — if the model failed to load in the previous render, the error state is restored immediately on mount to avoid a flash of broken content.

model_viewer.svelte

The inner Three.js / Threlte scene. Loads the GLB model file by calling `invoke("load_model")`, which returns the raw bytes of the file. The bytes are parsed directly in the browser using `GLTFLoader.parse()` — no HTTP request is made.

After loading, all mesh materials are replaced with a uniform `MeshStandardMaterial` in the Roboteam's purple brand colour (`#5A1C74`). The camera is automatically fitted to the model's bounding box so any model file will be centred and fill the view regardless of its original scale.

The model auto-rotates slowly using `OrbitControls` with `autoRotate`. User rotation, zoom, and pan are disabled — the view is fixed. A scale animation eases the model from 0 to 1 on load using Threlte's `useTask`.

model_debug.ts

A small utility module with two functions: `setLoadFailed(bool)` and `wasLoadFailed(): bool`. These read and write a `localStorage` key to persist the model error state across component re-renders.

Revision #1

Created 2026-04-15 08:34:52 UTC by Candela Cimadevilla Gonzalez

Updated 2026-04-15 08:35:59 UTC by Candela Cimadevilla Gonzalez