# Waste Volume Estimator – Change Log v2

## Dashboard Sync Reliability

### Problem:

* Dashboard used a fixed 3-second reload timer, so the view lagged behind updates.

### Changes Applied:

1. Introduced a dashboard event broker and Server-Sent Events endpoint to push real-time updates.
2. Replaced the polling script in dashboard.html with an EventSource listener that reloads immediately after persistence.

## Firestore Persistence Failures

### Problem:

* Firestore add() returned a timestamp tuple instead of a document reference, breaking event notifications.

### Changes Applied:

1. Switched to pre-creating a document reference with document().set(...) so we always keep the generated ID.
2. Published session\_saved events using the explicit doc\_ref.id to keep the dashboard feed in sync.

## Webcam Start/Stop Stability

### Problem:

* Start failed silently when a specific camera profile was unavailable and streams were not cleaned up on stop.

### Changes Applied:

1. Tried multiple getUserMedia constraint fallbacks with clearer error messaging.
2. Added cleanupStream() so tracks, video playback, and srcObject reset cleanly before each restart.
3. Toggled Start/Stop button disabled states to mirror capture state and prevent double-starts.

## Interface Modernisation

### Problem:

* The capture and dashboard pages relied on basic inline styles and were not responsive.

### Changes Applied:

1. Rebuilt index.html and dashboard.html with Tailwind CSS, adding a cohesive dark theme and responsive grids.
2. Organised controls, metrics, and tables into card layouts with improved typography and status cues.

## Food Classification Integration

### Problem:

* The system lacked food-level insights and could not distinguish sessions by meal type.

### Changes Applied:

1. Implemented a TensorFlow-backed classifier wrapper around keras\_model.h5/labels.txt with lazy loading and caching.
2. Ran classification during /process, surfaced the label & confidence in the HUD/JSON, and stored them with each session.
3. Extended dashboard metrics to count detections and display a Food Detection Breakdown panel.

## Empty Plate Session Finalisation

### Problem:

* Sessions previously ended when the utensil left the frame, not when the plate was actually empty.

### Changes Applied:

1. Defined canonical empty labels (nothing/none/empty) and track an empty streak before closing a session.
2. Kept aggregation running until three consecutive empty frames arrive, ensuring only completed meals persist.

## Classification Diagnostics & Tooling

### Problem:

* There was no quick way to validate keras\_model.h5 predictions outside the app.

### Changes Applied:

1. Added testapp.py CLI to classify arbitrary images with optional model/label overrides.
2. Improved classifier debugging with FOOD\_CLASSIFIER\_DEBUG to surface TensorFlow load errors on demand.