

OAL Dashboard Builder Brief v2

Complete Implementation Specification

Generated: 2025-12-04

For: Dashboard Builder / Coding Agent

Status: Ready for implementation

Executive Summary

Deliverable	Complexity	Priority
Manual Control Countdown Chip	Medium	★ HIGH
Sonos Modernization	Medium	HIGH
Room Detail Cards (from bedroom template)	Low (templated)	MEDIUM
Whole-Home Scenes Row	Low	MEDIUM
Light Group Controls	Low	LOW

Design Language:

- Orange = Active/On/Adaptive control
- Purple = Manual override / Non-baseline config
- Blue = Alarms / Scheduling
- Coral dot = Manual control indicator on room tiles

Part 1: Manual Control Countdown Chip ★ PRIORITY

1.1 Purpose

Show countdown timer when any lights are in manual override mode. Tap to cancel (reset) or extend the timer, similar to the notification flow.

1.2 Data Source

```
yaml
```

```
# Each AL switch exposes these attributes when lights are manually controlled:
```

```
switch.adaptive_lighting_*:
```

```
    attributes:
```

```
        manual_control: ["light.entity1", "light.entity2"] # List of overridden lights
```

```
        autoreset_time_remaining:
```

```
            light.entity1: 14234.5 # Seconds until autoreset
```

```
            light.entity2: 14234.5
```

1.3 AL Zone Inventory (6 zones)

Zone Switch	Description
switch.adaptive_lighting_main_living	Living room lights
switch.adaptive_lighting_kitchen_island	Kitchen lights
switch.adaptive_lighting_bedroom_primary	Bedroom lights
switch.adaptive_lighting_accent_spots	Accent/spot lights
switch.adaptive_lighting_recessed_ceiling	Ceiling recessed
switch.adaptive_lighting_column_lights	Column accents

1.4 Complete Chip Implementation

```
yaml
```

```
type: custom:bubble-card
card_type: button
button_type: state
entity: sensor.oal_real_time_monitor
name: Manual
icon: mdi:hand-back-left
show_state: false
show_name: true
visibility:
- condition: template
  value_template: >-
  {%
    set zones = [
      'switch.adaptive_lighting_main_living',
      'switch.adaptive_lighting_kitchen_island',
      'switch.adaptive_lighting_bedroom_primary',
      'switch.adaptive_lighting_accent_spots',
      'switch.adaptive_lighting_recessed_ceiling',
      'switch.adaptive_lighting_column_lights'
    ] %}
  {%
    set ns = namespace(count=0) %}
  {% for z in zones %}
    {% set ns.count = ns.count + (state_attr(z, 'manual_control') | default([]) | length) %}
  {% endfor %}
  {{ ns.count > 0 }}
}

tap_action:
action: perform-action
perform_action: script.oal_reset_soft
# Alternative: fire-dom-event for popup with Cancel/Extend options

hold_action:
action: more-info
entity: sensor.oal_real_time_monitor
styles: |
/* Purple chip styling */
.bubble-button-card-container {
  background: rgba(138, 100, 200, 0.15) !important;
  border: 1px solid rgba(138, 100, 200, 0.35) !important;
  border-radius: 20px !important;
  padding: 4px 12px !important;
  min-width: auto !important;
}
.bubble-icon {
  color: rgba(120, 80, 180, 0.9) !important;
  --mdc-icon-size: 18px !important;
}
```

```

}

.bubble-name {
  color: rgba(90, 60, 140, 0.9) !important;
  font-weight: 600 !important;
  font-size: 12px !important;
}

/* =====
JS TEMPLATE - Dynamic countdown display (at END of styles)
===== */

${(() => {
try {
  const zones = [
    'switch.adaptive_lighting_main_living',
    'switch.adaptive_lighting_kitchen_island',
    'switch.adaptive_lighting_bedroom_primary',
    'switch.adaptive_lighting_accent_spots',
    'switch.adaptive_lighting_recessed_ceiling',
    'switch.adaptive_lighting_column_lights'
  ];

  let soonest = Infinity;
  let count = 0;

  zones.forEach(z => {
    const manual = hass.states[z]?.attributes?.manual_control || [];
    const times = hass.states[z]?.attributes?.autoreset_time_remaining || {};
    count += manual.length;
    Object.values(times).forEach(t => {
      if (t < soonest) soonest = t;
    });
  });

  // Format time
  let timeStr = "";
  if (soonest < Infinity && soonest > 0) {
    const h = Math.floor(soonest / 3600);
    const m = Math.floor((soonest % 3600) / 60);
    timeStr = h > 0 ? `${h}h ${m}m` : `${m}m`;
  }

  // Update name element
  const nameEl = card.querySelector('.bubble-name');
  if (nameEl && timeStr && count > 0) {

```

```
    nameEl.innerText = `👋 ${timeStr} (${count})`;
}
} catch(e) { console.log('Manual chip error:', e); }
return "";
})()
```

1.5 Interaction Options

Option A: Tap = Reset, Hold = Extend

- Tap → script.oal_reset_soft (clear all manual control)
- Hold → Call service to extend autoreset timer by 1 hour

Option B: Tap = Popup with choices

yaml

```
tap_action:  
  action: fire-dom-event  
  browser_mod:  
    service: browser_mod.popup  
    data:  
      title: "Manual Control Active"  
      content:  
        type: vertical-stack  
        cards:  
          - type: markdown  
            content: "{{ count }} lights under manual control!"  
          - type: horizontal-stack  
            cards:  
              - type: custom:bubble-card  
                card_type: button  
                name: Reset All  
                icon: mdi:refresh  
                tap_action:  
                  action: perform-action  
                  perform_action: script.oal_reset_soft  
              - type: custom:bubble-card  
                card_type: button  
                name: Extend 1h  
                icon: mdi:timer-plus  
                tap_action:  
                  action: perform-action  
                  perform_action: script.oal_extend_manual_control
```

Part 2: Sonos Modernization

2.1 Current Sonos Entities (from Document 2)

Media Players:

Entity	Room
media_player.living_room	Living Room
media_player.dining_room	Dining Room
media_player.kitchen	Kitchen
media_player.bath	Bathroom
media_player.bedroom	Bedroom

Group Membership Sensors:

Entity	Purpose
binary_sensor.sonos_living_room_in_playing_group	Living in group?
binary_sensor.sonos_dining_room_in_playing_group	Dining in group?
binary_sensor.sonos_kitchen_in_playing_group	Kitchen in group?
binary_sensor.sonos_bath_in_playing_group	Bath in group?
binary_sensor.sonos_bedroom_in_playing_group	Bedroom in group?

Scripts:

Script	Function
script.sonos_group_all_to_playing	Group all to active speaker
script.sonos_ungroup_all	Ungroup all speakers
script.sonos_toggle_group_membership	Toggle specific room in/out
script.sonos_all_pause	Pause all playback

2.2 Modernized Sonos Card

yaml

```
type: custom:bubble-card
card_type: media-player
entity: media_player.living_room
name: Now Playing
icon: mdi:speaker-multiple
card_layout: large-2-rows
scrolling_effect: false
show_state: true
show_icon: true
show_name: true
cover_background: true
hide:
  power_button: true
button_action:
  tap_action:
    action: more-info
  double_tap_action:
    action: navigate
    navigation_path: "#sonos-detail"
hold_action:
  action: more-info
sub_button:
  main:
    # Room toggles - orange when in group
    - entity: binary_sensor.sonos_living_room_in_playing_group
      name: Living
      icon: mdi:sofa
      show_state: false
      show_background: true
      tap_action:
        action: call-service
        service: script.sonos_toggle_group_membership
        data:
          target_speaker: media_player.living_room
    - entity: binary_sensor.sonos_dining_room_in_playing_group
      name: Dining
      icon: mdi:table-furniture
      show_state: false
      show_background: true
      tap_action:
        action: call-service
        service: script.sonos_toggle_group_membership
        data:
```

```
    target_speaker: media_player.dining_room
- entity: binary_sensor.sonos_kitchen_in_playing_group
  name: Kitchen
  icon: mdi:countertop
  show_state: false
  show_background: true
  tap_action:
    action: call-service
    service: script.sonos_toggle_group_membership
    data:
      target_speaker: media_player.kitchen
- entity: binary_sensor.sonos_bath_in_playing_group
  name: Bath
  icon: mdi:shower-head
  show_state: false
  show_background: true
  tap_action:
    action: call-service
    service: script.sonos_toggle_group_membership
    data:
      target_speaker: media_player.bath
- entity: binary_sensor.sonos_bedroom_in_playing_group
  name: Bed
  icon: mdi:bed
  show_state: false
  show_background: true
  tap_action:
    action: call-service
    service: script.sonos_toggle_group_membership
    data:
      target_speaker: media_player.bedroom
# Action buttons
- name: Group
  icon: mdi:link-variant
  show_background: true
  tap_action:
    action: call-service
    service: script.sonos_group_all_to_playing
- name: Ungroup
  icon: mdi:link-off
  show_background: true
  tap_action:
    action: call-service
    service: script.sonos_ungroup_all
```

```

styles: |

/* =====
   GLASSMORPHISM CONTAINER
===== */

ha-card {
  background: rgba(255, 255, 255, 0.08) !important;
  backdrop-filter: blur(8px) saturate(120%) !important;
  -webkit-backdrop-filter: blur(8px) saturate(120%) !important;
  border: 1px solid rgba(255, 255, 255, 0.12) !important;
  border-radius: 20px !important;
  box-shadow: 0 4px 20px rgba(0, 0, 0, 0.06) !important;
}

/* Cover art styling */
.bubble-media-cover {
  border-radius: 12px !important;
  box-shadow: 0 4px 16px rgba(0, 149, 0, 0.15) !important;
}

/* Play button - orange accent */
.bubble-play-pause-button {
  background: rgba(255, 149, 0, 0.15) !important;
  border: 1px solid rgba(255, 149, 0, 0.3) !important;
}
.bubble-play-pause-button ha-icon {
  color: rgb(255, 149, 0) !important;
}

/* Volume slider - orange fill */
.bubble-range-fill {
  background: rgba(255, 149, 0, 0.8) !important;
}

/* =====
   ROOM TOGGLE CHIPS - Orange when in group
===== */

/* Living (sub-button 1) */
.bubble-sub-button-1 {
  background: ${hass.states['binary_sensor.sonos_living_room_in_playing_group']?.state === 'on'
    ? 'rgba(255, 149, 0, 0.18)'
    : 'rgba(255, 255, 255, 0.08)'} !important;
  border: ${hass.states['binary_sensor.sonos_living_room_in_playing_group']?.state === 'on'
    ? '1px solid rgba(255, 149, 0, 0.35)'
    : '1px solid transparent'} !important;
}

```

```
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}

.bubble-sub-button-1 ha-icon {
color: ${hass.states['binary_sensor.sonos_living_room_in_playing_group']?.state === 'on'
? 'rgb(255, 149, 0)'
: 'rgba(0, 0, 0, 0.4)'} !important;
}

/* Dining (sub-button 2) */
.bubble-sub-button-2 {
background: ${hass.states['binary_sensor.sonos_dining_room_in_playing_group']?.state === 'on'
? 'rgba(255, 149, 0, 0.18)'
: 'rgba(255, 255, 255, 0.08)'} !important;
border: ${hass.states['binary_sensor.sonos_dining_room_in_playing_group']?.state === 'on'
? '1px solid rgba(255, 149, 0, 0.35)'
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}

.bubble-sub-button-2 ha-icon {
color: ${hass.states['binary_sensor.sonos_dining_room_in_playing_group']?.state === 'on'
? 'rgb(255, 149, 0)'
: 'rgba(0, 0, 0, 0.4)'} !important;
}

/* Kitchen (sub-button 3) */
.bubble-sub-button-3 {
background: ${hass.states['binary_sensor.sonos_kitchen_in_playing_group']?.state === 'on'
? 'rgba(255, 149, 0, 0.18)'
: 'rgba(255, 255, 255, 0.08)'} !important;
border: ${hass.states['binary_sensor.sonos_kitchen_in_playing_group']?.state === 'on'
? '1px solid rgba(255, 149, 0, 0.35)'
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}

.bubble-sub-button-3 ha-icon {
color: ${hass.states['binary_sensor.sonos_kitchen_in_playing_group']?.state === 'on'
? 'rgb(255, 149, 0)'
: 'rgba(0, 0, 0, 0.4)'} !important;
}

/* Bath (sub-button 4) */
.bubble-sub-button-4 {
background: ${hass.states['binary_sensor.sonos_bath_in_playing_group']?.state === 'on'
? 'rgba(255, 149, 0, 0.18)'
: 'rgba(0, 0, 0, 0.08)'} !important;
border: ${hass.states['binary_sensor.sonos_bath_in_playing_group']?.state === 'on'
? '1px solid rgba(255, 149, 0, 0.35)'
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}
```

```

? 'rgba(255, 149, 0, 0.18)'
: 'rgba(255, 255, 255, 0.08)'} !important;
border: ${hass.states['binary_sensor.sonos_bath_in_playing_group']?.state === 'on'
? '1px solid rgba(255, 149, 0, 0.35)'
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}

.bubble-sub-button-4 ha-icon {
color: ${hass.states['binary_sensor.sonos_bath_in_playing_group']?.state === 'on'
? 'rgb(255, 149, 0)'
: 'rgba(0, 0, 0, 0.4)'} !important;
}

/* Bedroom (sub-button 5) */

.bubble-sub-button-5 {
background: ${hass.states['binary_sensor.sonos_bedroom_in_playing_group']?.state === 'on'
? 'rgba(255, 149, 0, 0.18)'
: 'rgba(255, 255, 255, 0.08)'} !important;
border: ${hass.states['binary_sensor.sonos_bedroom_in_playing_group']?.state === 'on'
? '1px solid rgba(255, 149, 0, 0.35)'
: '1px solid rgba(255, 255, 255, 0.12)'} !important;
border-radius: 14px !important;
}

.bubble-sub-button-5 ha-icon {
color: ${hass.states['binary_sensor.sonos_bedroom_in_playing_group']?.state === 'on'
? 'rgb(255, 149, 0)'
: 'rgba(0, 0, 0, 0.4)'} !important;
}

/* Group/Ungroup action buttons (6-7) - subtle */

.bubble-sub-button-6, .bubble-sub-button-7 {
background: rgba(255, 255, 255, 0.08) !important;
border: 1px solid rgba(255, 255, 255, 0.12) !important;
border-radius: 14px !important;
}

```

2.3 Sonos Detail Popup (custom:sonos-card)

Keep full sonos-card in popup accessed via double-tap or hold:

yaml

```
# In browser_mod popup or navigation target
```

```
type: custom:sonos-card
```

```
sections:
```

- player
- volumes
- groups
- grouping
- media browser
- queue

```
hidePlayerControlPowerButton: true
```

```
heightPercentage: 80
```

```
dynamicVolumeSlider: true
```

```
dynamicVolumeSliderMax: 100
```

```
showAudioInputFormat: true
```

Part 3: Room Detail Cards

3.1 Bedroom Card Template (PRODUCTION REFERENCE)

This is the actual working bedroom card - use as template for all rooms:

Key patterns:

1. Separator header with room icon (orange), alarm chip (blue), reset chip (purple), toggle all
2. 2-column grid of light slider cards
3. Orange/purple theming based on `manual_control` attribute
4. Hand emoji (👉) appended to name when manual
5. JS template for dynamic alarm time display

3.2 Room Configurations

Room	Icon	Zone Switch	Lights	Group Entity	Has Alarm
Living	<code>mdi:sofa</code>	<code>adaptive_lighting_main_living</code>	5	<code>light.main_living_lights</code>	No
Kitchen	<code>mdi:countertop</code>	<code>adaptive_lighting_kitchen_island</code>	2-3	<code>light.all_kitchen_lights</code>	No
Bedroom	<code>mdi:bed-king-outline</code>	<code>adaptive_lighting_bedroom_primary</code>	2	<code>light.bedroom_primary_lights</code>	Yes

Room	Icon	Zone Switch	Lights	Group Entity	Has Alarm
Dining		adaptive_lighting_column_lights	3	light.column_lights	No
Bath		adaptive_lighting_accent_spots	1-2	light.accent_spots_lights	Yes

3.3 Light Slider Card Template

Template Variables:

```

{LIGHT_ENTITY} → light.master_bedroom_table_lamps
{LIGHT_NAME} → Table Lamps
{LIGHT_ICON} → mdi:lamp
{ZONE_SWITCH} → switch.adaptive_lighting_bedroom_primary

```

Core Pattern (from bedroom card):

yaml

```
type: custom:bubble-card
card_type: button
button_type: slider
entity: {LIGHT_ENTITY}
name: {LIGHT_NAME}
icon: {LIGHT_ICON}
show_state: false
show_attribute: true
attribute: brightness
scrolling_effect: false
tap_action:
  action: toggle
hold_action:
  action: more-info
double_tap_action:
  action: call-service
  service: light.turn_on
target:
  entity_id: {LIGHT_ENTITY}
data:
  brightness_pct: 100
styles: |-
/* Container - orange/purple border based on manual */
.bubble-button-card-container {
  background: ${state === 'on' ? 'rgba(255,255,255,1)' : 'linear-gradient(145deg, rgba(0,0,0,0.02), rgba(0,0,0,0.04))'} !important
  border: ${state === 'on'
    ? (hass.states['{ZONE_SWITCH}'].attributes?.manual_control?.includes('{LIGHT_ENTITY}')
      ? '2px solid rgba(156,39,176,0.5)'
      : '2px solid rgba(255,152,0,0.5)')
    : '1.5px solid rgba(0,0,0,0.06)'} !important;
  /* ... rest of container styles ... */
}
/* Icon - orange/purple based on manual */
.bubble-icon {
  color: ${state === 'on'
    ? (hass.states['{ZONE_SWITCH}'].attributes?.manual_control?.includes('{LIGHT_ENTITY}')
      ? 'rgba(156,39,176,0.96)'
      : 'rgba(255,152,0,0.96)')
    : 'rgba(0,0,0,0.45)'} !important;
}
/* Slider fill - orange/purple based on manual */
```

```

.bubble-range-fill {
  background: ${hass.states['{ZONE_SWITCH}']?.attributes?.manual_control?.includes('{LIGHT_ENTITY}')
    ? 'rgba(156,39,176,0.9)'
    : 'rgba(255,152,0,0.9)'} !important;
}

/* JS TEMPLATE - Dynamic name with hand emoji */
${(() => {
  const isManual = hass.states['{ZONE_SWITCH}']?.attributes?.manual_control?.includes('{LIGHT_ENTITY}');
  const nameEl = card.querySelector('.bubble-name');
  if (nameEl) {
    nameEl.innerText = isManual ? '{LIGHT_NAME} 🤝' : '{LIGHT_NAME}';
  }
})()}


```

3.4 Room Card Access Options

Recommended: Popup from Hero Tile

```

yaml

# On hero card room tile
tap_action:
  action: fire-dom-event
  browser_mod:
    service: browser_mod.popup
    data:
      title: "Bedroom"
      content:
        type: vertical-stack
        cards:
          # Include bedroom card content here

```

Alternative: Navigate to view

```

yaml

tap_action:
  action: navigate
  navigation_path: /lovelace/bedroom

```

Part 4: Whole-Home Scenes

4.1 Scene Strategy

Your OAL configs effectively ARE your scenes:

- Config 1: Baseline → "All On" / "Day Mode"
- Config 2: Reduced Overhead → "Evening"
- Config 3: Evening Relax → "Relax"
- Config 4: Accents Only → "Movie Night"
- Global Manual Adjustment → Manual mode

Question: Do you have separate `scene.*` entities, or should scenes just change the OAL config?

4.2 If Using Separate Scenes

```
yaml
type: horizontal-stack
cards:
  - type: custom:bubble-card
    card_type: button
    entity: scene.morning
    name: Morning
    icon: mdi:weather-sunset-up
    tap_action:
      action: call-service
      service: scene.turn_on
      target:
        entity_id: scene.morning
    styles: |
      /* Scene chip styling */
      .bubble-button-card-container {
        background: rgba(255, 255, 255, 0.08) !important;
        border: 1px solid rgba(255, 255, 255, 0.12) !important;
        border-radius: 16px !important;
        padding: 8px 16px !important;
      }
```

4.3 If Using OAL Configs as Scenes

The Config dropdown on the hero card already provides this. Could add dedicated scene chips that just set the config:

```
yaml
- type: custom:bubble-card
  card_type: button
  entity: input_select.oal_active_configuration
  name: Movie
  icon: mdi:movie-open
  tap_action:
    action: call-service
    service: input_select.select_option
    target:
      entity_id: input_select.oal_active_configuration
    data:
      option: "Config 4 (Accents Only)"
```

Part 5: Entity Inventory (To Verify)

5.1 Known Entities (from templates)

Bedroom:

- light.master_bedroom_table_lamps
- light.master_bedroom_corner_accent
- light.bedroom_primary_lights (group)
- switch.adaptive_lighting_bedroom_primary

Sonos:

- 5 media players (listed above)
- 5 binary sensors for group membership
- 4 scripts for grouping/control

5.2 Need Discovery

```
yaml
```

```
# Run ha_search entities to find:
```

```
# Living Room lights (5 total)
```

```
query: "living"
```

```
domain_filter: "light"
```

```
# Kitchen lights (2-3)
```

```
query: "kitchen"
```

```
domain_filter: "light"
```

```
# Dining lights (3)
```

```
query: "dining OR column"
```

```
domain_filter: "light"
```

```
# Bath lights (1-2)
```

```
query: "bath OR accent"
```

```
domain_filter: "light"
```

```
# All scenes
```

```
domain_filter: "scene"
```

```
# Light groups
```

```
query: "all OR group"
```

```
domain_filter: "light"
```

Part 6: Implementation Phases

Phase 1: Core Chips (This Session)

Task	Status	Notes
Manual control countdown chip	READY	Template in Part 1
Verify AL zones expose autoreset_time_remaining	READY	May need OAL config change

Phase 2: Sonos Modernization

Task	Status	Notes
Apply glassmorphism to Sonos card	READY	Template in Part 2
Wire room toggles with orange highlighting	READY	Binary sensors exist

Task	Status	Notes
Move sonos-card to popup	[]	Use browser_mod

Phase 3: Room Cards

Task	Status	Notes
Discover all light entities per room	[]	Run HA queries
Parameterize bedroom card template	[]	
Build Living room card (5 lights)	[]	
Build Kitchen card	[]	
Build Dining card	[]	
Build Bath card	[]	
Wire popups from hero tiles	[]	

Phase 4: Polish

Task	Status	Notes
Scene row (if separate from configs)	[]	
Mobile responsiveness	[]	
Animation tuning	[]	

Appendix: Quick Reference

Color Palette

css

```

/* Orange - Active/Adaptive */
rgba(255, 149, 0, 0.15) /* Background */
rgba(255, 149, 0, 0.35) /* Border */
rgba(255, 152, 0, 0.96) /* Icon/Text */

/* Purple - Manual Override */
rgba(156, 39, 176, 0.15) /* Background */
rgba(156, 39, 176, 0.35) /* Border */
rgba(156, 39, 176, 0.96) /* Icon/Text */

/* Blue - Alarms */
rgba(66, 165, 245, 0.15) /* Background */
rgba(66, 165, 245, 0.25) /* Border */
rgba(66, 165, 245, 1) /* Icon/Text */

/* Glassmorphism */
rgba(255, 255, 255, 0.08) /* Background */
rgba(255, 255, 255, 0.12) /* Border */
background-filter: blur(8px) saturate(120%)

```

Key Scripts

Script	Function
script.oal_reset_soft	Clear all manual control
script.sonos_group_all_to_playing	Group all Sonos
script.sonos_ungroup_all	Ungroup all Sonos
script.sonos_toggle_group_membership	Toggle room in/out
script.sonos_all_pause	Pause all playback

End of Builder Brief v2