Airtable Interface Builder Guide

This document provides complete, step-by-step setup instructions for all Airtable interfaces, automations, and related workflows currently in use for your mushroom cultivation inventory and production management system.

1. Sterilizer IN (Start a Run)

Purpose: Log a new sterilization/pasteurization run and its plan.

Fields (editable):

- process_type (Sterilized / Pasteurized)

- planned_item (link to items)

- planned_recipe (link to recipes)

- planned_unit_size

- planned_count

- start_time

- operator

- ui_error (read-only)

Button:

"Start Run" → Run a script (`Sterilizer IN – Validate & Start`)

2. Sterilizer OUT (Complete Run & Create Lots)

Purpose: Input run outcomes and auto-create bags.

Fields (editable):

- good_count

- destroyed_count

- process_type

- planned_item

- planned_recipe

- planned_unit_size

- start_time

- ui_error (read-only)

Button:

"Complete Run & Create Lots" → Run a script (`Sterilizer OUT – Validate & Create Lots`)

3. LC → Grain Inoculation

Purpose: Use LC source and volume; mark grain as colonizing; decrement LC.

Fields:

- item_id

- recipe_id

- unit_size

- lc_lot_id (filter to lc_flask, lc_syringe)

- lc_volume_ml

- override_inoc_time (optional)

- inoculated_at (read-only)

- ui_error (read-only)

Button:

"Inoculate Grain" → Run script (`LC→Grain Inoculate`)

4. LC – Draw Syringes

Purpose: Convert LC flask into syringes.

Fields:

- item_id

- strain_id

- remaining_volume_ml (read-only)

- syringe_item_id (link to lc_syringe)

- syringe_count

- ui_error

Button:

"Make Syringes" → Run script (`LC Draw Syringes`)

5. LC – Inoculate Flask

Purpose: Use syringe to inoculate flask.

Fields:

- item_id (flask)

- recipe_id

- lc_lot_id (filter to lc_syringe)

- lc_volume_ml

- override_inoc_time

- ui_error

Button:

"Inoculate Flask" → Run script (`LC→Flask Inoculate`)

6. LC – Receive Purchased Syringes

Purpose: Intake vendor syringes.

Form Fields:

- item_id (lc_syringe)

- strain_id

Automation: `LC Receive Syringe`

7. Dark Room

Purpose: Manage colonization, casing, fridge moves, cold shock, fruiting.

Buttons:

- Fully Colonized → `action="FullyColonized"`

- Shake → `action="Shake"`

- Move to Fridge → `action="MoveToFridge"`

- Cold Shock → `action="ColdShock"`

- Start Fruiting → `action="StartFruiting"`

- Apply Casing → `action="ApplyCasing"`

Automation: `Dark Room – Actions`

8. Fruiting

Purpose: Start fruiting and compost spent blocks.

Buttons:

- Start Fruiting → `action="StartFruiting"`

- Compost → `action="Composted"`

Automation: `Fruiting – Actions`

9. Spawn to Bulk

Purpose: Mix grain and substrate → new fruiting blocks.

Fields:

- grain_inputs

- substrate_inputs

- output_count

- unit_size

- override_spawn_time

- ui_error

Button:

"Spawn to Bulk" → Run script (`Spawn→Bulk Create Blocks`)

Automation sets:

- item_id (FB-COCO/MM variants)

- category: fruiting_block

- strain_id from grain

- status: Colonizing

- marks inputs Consumed

- logs SpawnedToBulk event

10. Harvest (Formerly Freezer Intake)

Purpose: Record harvests from fruiting blocks to trays.

Fields:

- item_id (fresh_tray / freezer_tray)

- source_block_id (link to fruiting block)

- net_weight_g

- flush_no

Automation: `Harvest – Create Tray Product`

11. Freeze Dry & Package

Purpose: Process freezer trays and package mushrooms.

Fields:

- item_id (freezer_tray)

- source_block_id

- package_item_id (freezedriedmushrooms)

- net_weight_g

- ui_error

Buttons:

- FreezeDry → `action="FreezeDry"`

- Package Freeze-Dried → `action="PackageFreezeDried"`

Automation: `FreezeDry & Package – Actions`

12. Productize Grain/Substrate

Purpose: Turn sterilized materials into packaged products.

Fields:

- item_id

- source_lot_id

- net_weight_lb / g

- pack_date

- location_id

Automation: `Productize – Create Product`

Universal Interface Tips:

- Always display `ui_error` (read-only).

- Use "Link / unlink records" with filtered pickers (by item.category).

- Keep computed fields read-only.

- Match `action` values exactly.

- Automations trigger when action changes or a button runs a script.