63LABS

Restauranteur Learn & Build

SECTION 03 OF 16

Stock Suppliers and Food Records

Stock movements, purchasing, supplier reviews, ingredient requirements and recipe records.



SEPARATE TOPIC PDF / FULL RESTAURANT EDITION PRESERVED

Read this section

Two page-number systems

The section contents below use PDF page positions in this file. Original lesson footers keep their page numbers from the complete 619-page Restaurant Edition. "Master p/pp" still refers to the original 5,410-page Learn & Build source.

Same lessons, smaller files

Content has been separated, not rewritten. Nine shared-boundary pages were divided between their topics so every lesson keeps its ending. The complete Restaurant Edition and original master remain unchanged.

Start with the shared guide

The original scope guide is included below. Project sections also include the shared build starter and spreadsheet setup. These are learning and build specifications, not finished apps or certified operating instructions.

Moving between sections

Local contents and bookmarks open pages in this PDF. Cross-section links use the companion filenames and may require all files in one folder and a supporting PDF reader. If a link does not open, use the numbered section list and the printed edition-page label. Section 01 retains the original full-edition contents and source index.

The 16-section collection

01 Start Here and Source Index

02 Business Planning and Money

03 Stock Suppliers and Food Records

04 Team Operations and Procedures

05 Facilities Equipment and Resources

06 Catering Guests and Delivery

07 Marketing Publishing and Websites

08 Restaurant Products and Services

09 Careers and Team Learning

10 AI Tools and Task Guides

11 Operations Methods and Data

12 Evidence and Reasoning Cases

13 Integrated Restaurant Projects

14 Guest Clubs and Event Ideas

15 Supplier and Product Ideas

16 Concepts and Quality Checks

Contents for this PDF

Numbers on the right are PDF page positions in this section. Retained lesson footers show their complete-edition page labels.

How to use this edition (shared guide) 4

Shared build starter (shared guide) 5

Spreadsheet and document setup (shared guide) 6

Stock, suppliers and purchasing 7

P011 Ingredient and supply movement tracker 8

P032 Food and supply purchasing control 10

P033 Restaurant supplier review 12

P057 Ingredient lot, purchase checklist, and manual receiving record 14

Menu, recipes and food records 17

P056 Menu or staff-meal planning and ingredient-requirement workbook 17

P058 Restaurant recipe card and portion-template library 20

How to use this edition

This is a restaurant-focused curation of the actual 5,410-page 63Labs Learn & Build combined master. It keeps useful original lessons, projects, worked reasoning and tool guidance, then adds clearly labeled restaurant applications. The original file remains unchanged.

A reference library rather than a course

Start with the problem you are solving. Operators can use the money, stock, supplier, team and guest workflows; restaurant-industry builders can use the service, software, content and commercial project material. You do not need to implement every project or buy every named tool.

Broad restaurant use, not just daily operations

The edition retains all 50 original business projects and 43 adaptable planning, facilities, food-record, event and learning projects. Optional supplier, catering, career, venue-event and creative applications are identified as such. Private health, medication, sleep, fitness, mood and veterinary-care projects are not converted into employee-management tools.

What was changed

Repeated setup instructions and repeated practice layouts are consolidated. Restaurant application paragraphs are new adaptations. Source lessons and project requirements are identified by their original master page numbers. Necessary source corrections are labeled rather than silently presented as original text.

What you receive

A learning and build-specification PDF. It does not include generated apps, connected POS systems, live forms, installed automations, completed spreadsheets, legal policies, or a restaurant-specific food-safety program. The projects explain what to build and how to test it.

Source references

Master p or pp always means the physical page position in 63Labs_Combined_Ready.pdf, not a page in this edition. P001-style labels preserve the original project IDs. The contents and PDF bookmarks navigate this edition; the final index traces its source material.

Keep consequential decisions reviewed

Use fictional data while building. Keep private guest, staff and payment data out of shared examples. Use approved restaurant procedures and current authoritative guidance for food safety, employment, taxes, accessibility, privacy and other regulated matters. This edition does not certify compliance or safe equipment work.

Shared build starter

Use this once per build, followed by either the spreadsheet setup or document setup, then the chosen project specification. These consolidated instructions preserve the common safeguards from the source rather than repeating them in every project.

Copy this starter

Act as a patient build partner. Ask the project intake questions in one message, then wait for my answers. I may answer not sure. Separate my facts, goals, assumptions and research still needed. Never invent business facts, sources, permissions, credentials, customers, results or regulatory requirements. Use fictional sample records. Produce the complete requested artifact, not merely an outline. Explain how to copy, set up, use, test and revise it. Do not claim you created or saved a file unless you actually did. If long, stop at a section boundary and resume when I say continue without omissions. End with a requirement-by-requirement checklist. Do not connect services, publish, send messages, spend money or change live records as a side effect of drafting this build.

Add my restaurant context

My restaurant or industry role: [role]. Intended user: [person or role]. Exact task: [task]. Scope and exclusions: [limits]. Approved sources: [documents and dates]. Tools and access available: [tools]. Currency, locale and units: [values]. Owner and reviewer: [roles]. Deadline or work budget: [limit]. Ask about missing information rather than assuming it.

If a response or formula fails

Continue from the exact point where you stopped; keep the same structure, tab names, column order and references. For a formula issue, inspect the actual formula, cell, expected result, source values, locale, dates, units and copied ranges. Explain the proposed correction and rerun the original test. A polished answer is not proof of correctness.

Spreadsheet and document setup

Spreadsheet setup

For Google Sheets, confirm locale and units. The source examples use English functions, comma separators, real dates displayed YYYY-MM-DD, headers in row 1 and data rows 2:1000. Preserve named tabs and column order. Supply actual tab-separated headers, fictional sample rows and exact paste cells. Leave computed sample cells empty. Give each formula with its exact cell, explanation, fill range and expected result. Keep unused rows blank and distinguish missing inputs from zero. Specify validation, dropdowns, formats, filters, frozen headers and formula protection. Keep raw data tables unmerged. Do not add scripts, paid add-ons or external connections by default. Explain how every related range extends beyond row 1000.

Document setup

Create copy-ready pages with clear headings, useful tables only where appropriate, fill-in fields and a completed fictional example. Keep the reusable blank version separate from the example. Supply the actual wording and templates requested, not instructions to write them later. Name owners, actions, dates and review points. Use approved policy content; for changing rules, cite current primary sources only when verified, otherwise list a precise verification task.

First use

Create a safe copy. Enter the source example and compare every expected value. Test blank inputs, explicit zero, duplicate IDs, refunds/reversals, period boundaries and a changed value. For dates and percentages, inspect the underlying values as well as the display. Save evidence before replacing fictional records with authorized live information.

Limits of the supplied specifications

The arithmetic examples have been reviewed, but this PDF is not an executed application test. Generated formulas, data validation, access controls, links, integrations and regulatory treatment still need testing in the actual implementation. Editorial clarifications identify known source gaps, not a claim that all possible edge cases are covered.

Stock, suppliers and purchasing

Use the projects below independently. P numbers are the original master project IDs, retained for traceability. The short source lessons and worked reasoning are followed by project-specific build specifications.

P011 Ingredient and supply movement tracker

P032 Food and supply purchasing control

P033 Restaurant supplier review

P057 Ingredient lot, purchase checklist, and manual receiving record

P011 Ingredient and supply movement tracker

Stock, suppliers and purchasing

Source project: Inventory Project. Master pp 1022, 1159-1160, 2960. Restaurant application below is an editorial adaptation.

Restaurant application

Apply SKU/unit/location records to countable ingredients and consumables, with manager-reviewed issues, write-offs and count adjustments.

Source based lesson

Track stock movements, available quantities, reorder levels, and stock-count adjustments for a small product catalog.

Gather and build

SKU, opening quantity, receipts, shipments, returns, adjustments. Use a movement log; calculate stock from movements; document physical-count differences.

Check before relying on it

Reservations reduce availability without changing physical stock. Count five products and investigate differences before adjusting records.

Source based practice with fictional inputs

Opening stock 18, receipts 7, shipments 6, reservations 4.

Worked result: Physical stock is 19 and available stock is 15. Calculate physical quantity from movements and availability from reservations separately.

Try a changed condition: Two reservations are cancelled.

Check your answer: Physical stock remains 19; available stock becomes 17.

Restaurant boundary

Original has no batch/expiry/temperature/allergen controls, recipe depletion, yield conversion or valuation. These require separately specified and verified extensions, not silent assumptions.

P011 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Intake

1. What products, stock units, and locations do you use?
2. What date will opening counts represent?
3. Do you reserve stock before dispatch?
4. What reorder points and suppliers are confirmed?

5. Do you need a simple quantity tracker or a separately verified valuation method?

Structure

Products A:H = SKU, Product, Unit, OpeningQty, ReorderPoint, OnHand, Reserved, Available. Movements A:F = MovementID, Date, SKU, SignedQty, Reason, Reference. Reservations A:E = ReservationID, SKU, Qty, Status, OrderRef. Add Counts and Summary tabs.

Build Requirements

Use positive SignedQty for receipts/returns into stock and negative for issues/write-offs. OpeningQty is the count at the chosen start date; do not also enter that opening count as a movement. Movements are permanent records, corrected with referenced reversals. Reservations use Active/Released/Fulfilled; only Active reserves quantity. On fulfilment release the reservation and record the issue once.

Products F2: =IF(A2="","",IF(D2="","",D2+SUMIF(Movements!$C$2:$C$1000,A2,Movements!$D$2:$D$1000))) G2 sums Reservations Qty for matching SKU and Active status. H2=F2-G2, guarded for missing F. Give exact formulas. Reorder if Available<=ReorderPoint; a missing point should say Set point. Flag negative stock, negative availability, unknown SKUs, duplicate IDs, missing units, and movements dated before the opening date.

Counts columns: CountID, Date, SKU, SystemQtyAtCount, ActualQty, Difference, AdjustmentID. Freeze the system quantity as a value at the count time; Difference=Actual-System. Create exactly one linked adjustment movement after review. Never use a live on-hand formula in a historical count snapshot. Build a filtered low-stock view and per-SKU movement history. Do not add unlike units or imply this is an accounting valuation system.

Test SKU A opening 20, receipt +5, issue -8 gives 17 on hand; active reservation 3 gives 14 available. Reorder point 15 flags a reorder. A physical count of 16 creates adjustment -1, then on hand 16 and available 13. Show how to avoid posting the adjustment twice.

Acceptance Check

The sample ends at 16 on hand and 13 available after one adjustment; unknown SKUs and duplicate movements are visible. Give expected results for the worked example and one missing-input case. Finish with a short first-use checklist and a repeat-use routine.

Editorial implementation clarifications

Apply these explicit additions or scope limits to the source brief before building. They resolve missing fields or boundaries; the original master has not been edited.

D05: Declare a named Settings OpeningDate for the one-date model, or per-SKU OpeningDate if counts differ. Constrain the model to a single documented stock location unless location fields and all formulas are explicitly extended.

Source extensions and troubleshooting

Master p1161. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Locations: Add [locations] as a second stock key and recalculate balances per SKU and location; transfers must produce paired movements.

Reorder planning: Add supplier lead time and a user-entered target stock level; show suggested order quantity without inventing demand.

Variants: Add [size/color] variants with separate permanent SKUs and preserve their movement histories.

Stock doubled. Remove the duplicate opening movement after backup; opening quantity belongs in one place.

Units conflict. Split pack and each units or define a verified conversion before aggregation.

Historical count changes. Replace its live balance with the reviewed count-time snapshot.

Available is negative. Check stale reservations and unrecorded receipts before adjusting inventory.

P032 Food and supply purchasing control

Stock, suppliers and purchasing

Source project: Purchasing Project. Master pp 1043, 1243-1244, 3212. Restaurant application below is an editorial adaptation.

Restaurant application

Use purchase requests, approved orders, partial receiving, delivery discrepancies and invoice matching.

Source based lesson

Control purchasing requests, approved orders, receipts, and supplier invoice matching.

Gather and build

Request ID, supplier, approved quantity, unit cost, receipt, invoice. Separate requested, approved, ordered, received, and billed quantities; investigate differences.

Check before relying on it

Do not approve payment solely because an invoice exists. Reconcile a sample order against its receipt and supplier invoice.

Source based practice with fictional inputs

Order 12 units, receive 9, invoice 12.

Worked result: Three units remain unreceived; the invoice/receipt mismatch needs review. Preserve the actual receipt and route the discrepancy to the responsible reviewer.

Try a changed condition: Three remaining units arrive with a dated receipt.

Check your answer: Update received quantity to 12 and retain both receipt events.

Restaurant boundary

Receiving quantity is not food-safety acceptance and does not authorize payment. Keep quality/temperature checks grounded in approved procedures; preserve over-receipt and reversal logic.

P032 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Intake

What do you buy? Who requests and approves purchases? What currency and tax treatment apply? Are partial receipts common? What information is needed to match invoices?

Structure

Orders: POID, SupplierID, OrderDate, Status, Approver, Currency. Lines: LineID, POID, Item, QtyOrdered, UnitCost, QtyReceived, OpenQty, OrderedValue. Receipts: ReceiptID, Date, LineID, QtyReceived, Note. Add InvoiceMatch and PrintPO.

Build Requirements

Use order states Draft/Approved/Sent/Part received/Closed/Cancelled. Require approval evidence before Sent. Lines F sums receipt quantities by LineID; G=QtyOrdered-QtyReceived; H=QtyOrdered*UnitCost. Guard all formulas for missing required inputs and retain explicit zeros. Provide exact cell formulas and dropdowns. Receipt reversals use signed quantities and a reason; do not overwrite receipt history.

Flag unknown LineID, duplicate receipt IDs, negative open quantity, and cancelled orders receiving stock. Keep each line's unit consistent. PrintPO selects one POID and shows supplier, delivery instructions, quantities, costs, totals, currency, and approval reference. Do not invent banking or tax information.

InvoiceMatch fields: MatchID, SupplierInvoiceRef, POID, InvoicedAmount, ReceivedValue, Difference, ReviewStatus. Compare ordered, received, and invoiced evidence; do not automatically authorize payment. Explain that goods received does not prove the invoice is correct. A partial receipt leaves the order open.

Test: order 10 items at 8 each, receive 6 then 4. Ordered value is 80; after the first receipt open quantity is 4 and received value 48; after the second open quantity is zero. A third receipt of 1 flags over-receipt. Provide a purchasing review queue and archive instructions.

Acceptance Check

The partial-receipt example reconciles to 80; over-receipt and unknown lines are flagged. Give expected results for the worked example and one missing-input case. Finish with a short first-use checklist and a repeat-use routine.

Editorial implementation clarifications

Apply these explicit additions or scope limits to the source brief before building. They resolve missing fields or boundaries; the original master has not been edited.

D08: Append Unit to Lines without shifting F:G:H formulas. Append DeliveryInstructions and ApprovalRef to Orders or define a referenced order detail record. Ensure Sent cannot pass without the actual approval evidence.

D09: Bound the starter exercise to one invoice per PO, or add explicit per-invoice line/receipt allocations with non-duplication checks. Do not claim robust partial/multiple-invoice reconciliation without that design.

Source extensions and troubleshooting

Master p1245. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Adapt the scope: Add approval levels for [spending thresholds].

Change the workflow: Adapt partial deliveries to [locations] with separate receipt references.

Add a useful view: Add an overdue-delivery view using a promised date and open quantities.

Check the inputs. Match receipts to LineID, not just product names.

Check the workflow. Do not close an order while unreceived quantity remains without an approved cancellation.

Check the result. Keep invoice approval separate from receipt recording.

P033 Restaurant supplier review

Stock, suppliers and purchasing

Source project: Supplier Management Project. Master pp 1044, 1247-1248, 3224. Restaurant application below is an editorial adaptation.

Restaurant application

Track food/service suppliers using dated quality, delivery and service evidence and verified renewal terms.

Source based lesson

Maintain supplier contacts, service expectations, review evidence, and renewal decisions.

Gather and build

Supplier ID, service, terms, contact, review criteria, renewal. Keep a single supplier record; log performance evidence; schedule renewal decisions before deadlines.

Check before relying on it

A favorable price does not erase missing quality or capacity evidence. Compare two suppliers using the same criteria and documented information.

Source based practice with fictional inputs

Supplier A: $6/unit, minimum 60. B: $8/unit, minimum 20. Need is 20 units.

Worked result: A commits $360; B commits $160. A's lower unit price requires more stock and cash. Compare total commitment, unused stock, quality, and timing.

Try a changed condition: Verified demand rises to 60 units.

Check your answer: A costs $360 and B $480 before other terms; reassess with the new demand and remaining criteria.

Restaurant boundary

Internal scores are not objective public ratings. Missing ratings stay incomplete; never infer notice or termination rights.

P033 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Intake

Which supplier types matter? Who owns each relationship? What terms and renewal dates are confirmed? What performance evidence exists? Which review criteria and weights should apply?

Structure

Vendors: VendorID, Name, Category, Owner, ContactRef, AgreementLink, RenewalDate, Status. Reviews: ReviewID, VendorID, Period, QualityScore, TimelinessScore, ServiceScore, Evidence, Action. Add Issues and ReviewQueue.

Build Requirements

Use Active/Trial/Paused/Retired status. Store agreement links and approved contact references, not payment credentials. Separate verified terms from notes awaiting confirmation. Define review scores from 1 to 5 with observable anchors supplied by the user.

Create a weighted score using user-entered weights that sum to 100%. A missing required score produces Incomplete, never zero. Show exact formulas, input validation, and a missing-evidence flag. Example: quality 4 at 50%, timeliness 3 at 30%, service 5 at 20% gives 3.9 out of 5. Scores are internal evaluations, not objective public rankings.

Issues fields: IssueID, VendorID, OpenedDate, Description, Impact, Owner, DueDate, Status, ResolutionEvidence. Provide a review queue for open issues and renewals within a selected number of days. Archived vendors remain linked to historical records.

Supply a vendor intake questionnaire, performance-review agenda, and renewal decision template covering continue, renegotiate, trial alternative, or end subject to agreement terms. Do not infer termination rights or auto-renewal rules; use verified contract language. Test missing scores, expired dates, and an issue linked to an unknown vendor.

Acceptance Check

The sample weighted score is 3.9; missing scores, unknown vendors, and approaching notice dates are visible. Give expected results for the worked example and one missing-input case. Finish with a short first-use checklist and a repeat-use routine.

Editorial implementation clarifications

Apply these explicit additions or scope limits to the source brief before building. They resolve missing fields or boundaries; the original master has not been edited.

D04: Either explicitly limit that acceptance to renewal dates, or append verified notice rule/basis, NoticeDeadline, and terms/evidence reference. Do not infer deadline from RenewalDate alone. If integrating Project039, keep notice-rule logic consistent.

Source extensions and troubleshooting

Master p1249. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Adapt the scope: Adapt score criteria to [supplier category].

Change the workflow: Add notice deadlines copied from our verified agreements.

Add a useful view: Create a supplier review dashboard separating performance from contract status.

Check the inputs. Weights must total 100% before a score is shown.

Check the workflow. Use agreement evidence for renewal and notice dates.

Check the result. Missing ratings must remain incomplete rather than lowering the score silently.

P057 Ingredient lot, purchase checklist, and manual receiving record

Stock, suppliers and purchasing

Source project: Food Organization Project. Master pp 1068, 1343-1344, 3512. Restaurant application below is an editorial adaptation.

Restaurant application

Map pantry to verified usable restaurant stock; retain canonical item, unit, lot, location, package-label dates, and manual receiving/consumption procedure.

Source based lesson (restaurant wording adapted)

Compare restaurant ingredient requirements with checked available stock in matching units, retaining lot/location and transcribed package-date information.

Gather and build

Item, unit, current stock, planned need, storage location. Confirm product identity and units; subtract only checked available stock; calculate the remaining requirement and show purchasable package rounding separately.

Check before relying on it

Do not subtract grams from packages without a conversion. Record confirmed receipt and actual usage through the chosen stock workflow. Do not overwrite stock from a shopping plan alone.

Source based practice with fictional inputs

Restaurant ingredient-requirement rows request 0.5 kg and 250 g of the same confirmed product.

Worked result: After unit normalization, combined need is 750 g. Convert to a common unit before combining confirmed matches.

Try a changed condition: Two similar names refer to different products.

Check your answer: Keep them separate until product identity is confirmed.

Restaurant boundary

A date formula does not establish safety. Exclude reserved/unavailable stock manually or explicitly extend available-quantity schema; do not subtract unlike units or double-deduct pantry.

P057 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Adaptation: restaurant-only context or a non-restaurant example has been revised. See the source and editorial index.

Intake

Ask for the restaurant purchasing and receiving routine; locale; suppliers or stores; stock locations; exact item names and units; verified available quantities; and package-label fields to transcribe. Do not determine food safety from date formulas.

Base Tabs And Columns

Pantry: A Lot ID, B Item, C Quantity Available, D Unit, E Location, F Label Type, G Label Date, H Opened Date, I Notes. Grocery: A Item ID, B Item, C Required Quantity, D Unit, E Pantry Available, F Suggested Buy, G Manual Buy, H Final Buy, I Store, J Purchased, K Notes. Lists: canonical item names, units, locations, stores, and label types. Summary: shopping totals by item count, pending lines by store, and pantry items with package dates requiring review.

Use one pantry row per lot or package group with the same item, unit, location, and relevant dates. Purchased is a checkbox. Lot IDs and grocery Item IDs must be unique within their tabs.

INVENTORY RULES Use consistent item names such as "Black beans" and consistent units such as can, g, kg, or each. Keep separately flavored or sized products distinct where quantities are not interchangeable. Require nonnegative numeric quantities. Pantry Available means stock the user has checked and considers available; do not determine food safety from a date formula.

Label Type records what the package actually says, such as Best before, Use by, or Other. Label Date is copied from the package, not calculated by the AI. Date reminders are prompts to review the package and appropriate guidance, not declarations that food is safe or unsafe. Leave unknown dates blank and show them as unknown.

Formulas

Grocery E2, filled through E1000: =IF(A2="","",IF(OR(B2="",D2=""),"",SUMIFS(Pantry!$C$2:$C$1000,Pantry!$B$2:$B$1000,B2,Pantry!$D$2:$D$1000,D2))) Grocery F2, filled through F1000: =IF(A2="","",IF(OR(C2="",E2=""),"",MAX(0,C2-E2))) Grocery H2, filled through H1000: =IF(A2="","",IF(G2="",F2,G2)) Manual Buy overrides Suggested Buy when supplied; zero is a valid override. Flag missing item, unit, required quantity, negative quantities, duplicate IDs, and duplicate Item plus Unit shopping lines that might double-count the same need.

SHOPPING PROCESS Required Quantity is the total amount needed, before pantry deduction. If it already means amount to purchase, use Manual Buy rather than subtracting stock twice. Do not add incompatible units. Never round to package sizes automatically unless I supply the package size in the same unit.

Purchased only marks the shopping checklist. It must not also increase Pantry automatically, because that can duplicate stock on repeated checkbox edits. Give a deliberate receiving procedure: confirm the purchased quantity, update or add a pantry lot, record dates if known, then archive or clear the shopping line. Consumption and disposal require manual stock updates in this base version.

SUMMARY AND TESTS Show pending lines where Purchased is false and Final Buy is positive, grouped by store. Show missing inputs separately. Need 5 cans and stock 2 gives Suggested Buy 3. Stock 7 gives zero. Manual Buy 0 must remain zero. Stock 500 g cannot reduce a requirement of 2 cans. Two pantry lots of 1 and 2 cans sum to 3. End with shopping, receiving, and stock-review checklists.

Source extensions and troubleshooting

Master p1345. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Add package sizes: Add package size and package unit fields. Round purchases up only when Required Quantity and package size use the same unit. Preserve the original quantity needed.

Organize multiple locations: Use these locations: [list]. Give a pantry view by location while keeping matching item-and-unit totals across locations.

Prepare a restaurant purchasing list: Use these needs: [items, quantities, units]. Compare them with my recorded stock and flag anything requiring a manual check.

Suggested Buy is too small. Required Quantity may already be a purchase amount. Use Manual Buy in that case so pantry stock is not subtracted twice.

Stock is missing. Check exact item names and units, including spaces and singular/plural variations.

A purchased checkbox changed stock twice. Keep Purchased as a checklist only. Update pantry stock through a deliberate receiving step.

A date reminder looks like a safety judgment. Rename it Package date review. The tracker records labels and dates; it does not evaluate food safety.

Menu, recipes and food records

Use the projects below independently. P numbers are the original master project IDs, retained for traceability. The short source lessons and worked reasoning are followed by project-specific build specifications.

P056 Menu or staff-meal planning and ingredient-requirement workbook

P058 Restaurant recipe card and portion-template library

P056 Menu or staff-meal planning and ingredient-requirement workbook

Menu, recipes and food records

Source project: Meal Planning Project. Master pp 1067, 1339-1340, 3500. Restaurant application below is an editorial adaptation.

Restaurant application

Map household servings to verified portions; preserve recipe IDs, base yield, planned portions, scale factor, and gross ingredient expansion.

Source based lesson (restaurant wording adapted)

Organize planned menu or staff-meal portions using approved recipe records, the service schedule, and checked ingredient availability.

Gather and build

Approved recipes and yields, planned portions, verified ingredient exclusions, service schedule, checked available ingredients. Plan portions around confirmed capacity; keep recipe requirements, prepared output, and available stock distinct. Use approved preparation instructions only.

Check before relying on it

Do not count prepared or consumed food as still available. Date and quantity calculations do not establish food safety. Compare planned portions, recorded production, and remaining verified inventory; investigate differences without inventing safe reuse rules.

Source based practice with fictional inputs

Two not-yet-prepared restaurant recipe requirements need 300 g and 200 g of one ingredient; 150 g is checked as available in matching units.

Worked result: The total need is 500 g; the shopping quantity is 350 g. Reconcile future requirement and checked available quantity in matching units. Quantity arithmetic does not establish safe suitability or reuse.

Try a changed condition: The 150 g entry is actually reserved and unavailable to these recipes.

Check your answer: Do not subtract it from available stock; the shopping need returns to 500 g.

Restaurant boundary

Record/quantity planning only. No validated commercial food-safety, allergy, shelf-life, cook-time, pan-size, or nutrition system. Prepared meals remain in source gross weekly need; distinguish remaining-to-buy.

P056 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Adaptation: restaurant-only context or a non-restaurant example has been revised. See the source and editorial index.

Intake

Ask for planning week; menu or staff-meal scope; confirmed planned portions; approved recipe IDs and base yields; available preparation time and equipment; verified ingredient exclusions; and checked available ingredients. Do not prescribe diets, declare allergy safety, or invent commercial cooking/storage instructions.

Base Tabs And Columns

Recipes: A Recipe ID, B Recipe Name, C Base Servings, D Active Minutes, E Total Minutes, F Source or Link, G Notes. Ingredients: A Ingredient Row ID, B Recipe ID, C Ingredient, D Quantity, E Unit, F Notes. Plan: A Meal ID, B Date, C Meal, D Recipe ID, E Recipe Name, F Planned Servings, G Scale Factor, H Preparation Note, I Status. Lists: meal types, units, and statuses. ShoppingDraft: required ingredient, quantity, unit, and contributing meal IDs for the selected week.

Meal types: Breakfast, Lunch, Dinner, Snack. Statuses: Planned, Prepared, Skipped. The selected Week Start is stored in ShoppingDraft B2.

MEAL RULES Use recipes I provide or suggest meal ideas for my review. Mark new recipes as untested. Respect stated ingredient exclusions, but do not promise allergy safety from a recipe name; retain label and cross-contact checks as user responsibilities. Do not invent storage lifetimes or substitute unsupported food-safety instructions.

Recipes C must be a positive number. Ingredients has one ingredient per row linked to Recipe ID. Quantities are numeric and units explicit. Items such as salt to taste may have blank Quantity with a note; keep them visible as manual-check items. Recipe Name in Plan is a lookup; flag unknown Recipe IDs instead of hiding them.

Plan G is Planned Servings divided by the matched recipe's Base Servings. Return blank when input is missing and flag a zero or missing base yield. Give the exact lookup and scaling formulas with their cells. Scaling changes ingredient quantities; it does not guarantee that cook time, pan size, or method scales linearly.

SHOPPING DRAFT Expand every non-skipped meal in the seven-day period into its ingredient requirements, multiplied by its scale factor. Include Meal IDs for traceability. Explain one fully specified helper-tab or formula method, with exact headers, ranges, and paste locations, rather than merely saying "combine the ingredients".

Sum only exact normalized ingredient names with the SAME unit. Never add grams to cups or assume one onion has a universal weight. Keep ambiguous units, missing quantities, and to-taste ingredients in a manual-review list. This draft is gross recipe need; restaurant pantry deductions happen after I inspect stock. Do not subtract restaurant pantry quantities unless I supply a matching item and unit.

PREPARATION VIEW Create a seven-day menu and preparation checklist organized by date. Show active and total recipe time separately; do not assume parallel cooking times can simply be summed into an accurate schedule. Include leftovers only when I identify them and provide their preparation date. Point to the original recipe or authoritative food-safety guidance when timing or storage is uncertain.

Tests

A four-serving recipe using 200 g rice planned for two servings needs 100 g. Two such meals need 200 g. A separate ingredient in cups remains a separate line. A Skipped meal is excluded; Prepared meals stay in the week's gross plan. A zero base serving count is flagged. End with a menu, shopping-review, and preparation checklist.

Editorial implementation clarifications

Apply these explicit additions or scope limits to the source brief before building. They resolve missing fields or boundaries; the original master has not been edited.

E02: Label GrossWeeklyRequirement. For a restaurant remaining-to-buy extension, explicitly define future Planned meals/outstanding prep only and a matching stock snapshot. Preserve original gross-plan exercise separately.

Source extensions and troubleshooting

Master p1341. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Plan a simpler week: Use [number] dinners with at most [minutes] active preparation each, based on these recipes: [list]. Include a flexible night and ask before choosing substitutions.

Scale approved recipe portions: Change planned servings to [number]. Recalculate ingredient requirements and flag quantities, pan sizes, and steps that need manual review.

Use restaurant pantry ingredients: Prioritize these ingredients: [items and quantities]. Match exact units and suggest recipe choices without assuming unknown stock amounts.

Shopping amounts look doubled. Check repeated Meal IDs and whether a leftover meal was also entered as a full new cook. Label intentional batch cooking.

Units do not match. Keep them separate until a recipe-specific conversion is available. Do not use generic cup-to-gram conversions for every ingredient.

Scaling fails. Base Servings must be a positive number and Recipe ID must match an existing recipe.

An ingredient is missing. Review blank quantities and to-taste notes; these belong in the manual-check shopping list.

P058 Restaurant recipe card and portion-template library

Menu, recipes and food records

Source project: Recipes Project. Master pp 1069, 1347-1348, 3524. Restaurant application below is an editorial adaptation.

Restaurant application

Map personal note to approved dish description; retain source, original yield, units, ingredients, method, tested changes, and trial observations.

Source based lesson (restaurant wording adapted)

Create an editable restaurant recipe-card and portion-template library with consistent ingredient quantities, approved methods, observed yields, and sources.

Gather and build

Recipe title, source, base yield, ingredient quantities and units, supplied method, tested observations. Standardize the layout; retain source attribution; record tested changes separately from the original.

Check before relying on it

A scaling change must not silently overwrite the base recipe. Under the restaurant's approved preparation process, record observed timing and yield for one familiar recipe. Keep the base source and the observed trial separate.

Source based practice with fictional inputs

An approved restaurant recipe record has a base yield of four portions using 200 g of an ingredient.

Worked result: A simple ingredient scaling calculation for six servings gives 300 g. Keep the base recipe and label the scaled version; cooking behavior may need separate checking.

Try a changed condition: The scaled batch is observed to yield only five portions of the chosen size.

Check your answer: Record the observed yield and portion basis instead of claiming six were produced.

Restaurant boundary

Preserve supplied meaningful temperatures/timings but do not invent them. Scaling ingredient quantities is not validation of commercial batch cooking, allergens, storage, or nutrition. Attribute sources and label untested adaptations.

P058 Source based build specification

Copy the shared starter, the matching document or spreadsheet setup, your restaurant context, and this specification together. Project-specific requirements are retained and adapted for restaurant use; repeated setup is consolidated. Apply the labeled implementation clarifications as part of the brief. This is not finished software.

Adaptation: restaurant-only context or a non-restaurant example has been revised. See the source and editorial index.

Intake

Ask for recipe-library title; intended internal-training or authorized-publication use; supplied or lawfully usable recipes; categories; measurement system; base yields; source method; tested status; page format; and missing fields requiring confirmation. Do not invent food-safety details.

BUILD APPROACH Start by formatting one recipe for review. If essential quantities, yield, timings, or steps are missing, ask focused questions and mark unknowns. Never silently invent missing cooking temperatures or food-safety instructions. Once I approve the format, use it consistently for all recipes.

Recipe categories may include Breakfast, Lunch, Dinner, Sides, Snacks, Baking, and Drinks. Use only the categories that fit my collection. Assign stable recipe IDs such as R-001 for cross-references.

RECIPE PAGE TEMPLATE For every recipe, include these fields in order: 1. Recipe ID and title. 2. A short approved dish description or contributor note, using supplied facts. 3. Yield with unit, such as 4 servings or 12 muffins. 4. Active preparation time, cooking time, resting time if applicable, and total elapsed time. Do not blindly add overlapping durations; explain the stated total. 5. Equipment and pan dimensions when known. 6. Ingredients in order of use, with separate quantity, unit, ingredient, and preparation-note columns. 7. Numbered method steps with the source's meaningful quantities, temperatures, timings, and cues preserved. 8. Serving suggestions and optional substitutions clearly separated from required ingredients. 9. Ingredient exclusions or dietary notes based only on confirmed ingredients. Do not guarantee allergy safety or medical suitability. 10. Storage and reheating notes only when supplied or supported by a reliable food-safety source. Otherwise mark them To verify. 11. Original source or recipe contributor, adaptation notes, tested status, and last-made date. 12. A notes area for results and future changes. BOOK STRUCTURE Create a title page, a brief how-to-use page, a contents list grouped by category, the recipe pages, a short ingredient index, and a blank recipe template. Use one recipe starting on a new page; allow longer recipes to continue naturally rather than shrinking text. Do not invent final page numbers before layout is complete. Explain how to update the contents list after editing. UNITS AND SCALING Preserve original measurements by default. If I request scaling, calculate desired yield divided by original yield and multiply scalable ingredient quantities by that factor. Keep both original and scaled versions. Flag indivisible items, leavening, seasoning, pan size, and cooking method for review. Never assume a universal conversion between cups and grams for every ingredient. CONTENT AND REVIEW Work from recipes I provide or original ideas developed here. For externally sourced recipes, retain attribution and a link; do not reproduce a full copyrighted page without permission. Mark generated recipes and substantial adaptations as untested until I record a successful trial. Never invent restaurant stories, personal experience, or test results. CHECKS Confirm that every ingredient is used in the method and every ingredient mentioned in the method appears in the list. Check that yield and units are present, source notes are retained, unknown safety details are flagged, and scaled quantities are arithmetically correct. End with a recipe-by-recipe completeness checklist and the blank template.

Source extensions and troubleshooting

Master p1349. These are optional follow-on changes, not included features. Keep the restaurant scope and approval boundaries above.

Match my preferred style: Format my recipe pages with [font, colors, and spacing]. Keep ingredient quantities easy to scan and leave notes space. Start each recipe on a new page.

Scale a recipe: Create a separate version of recipe [ID] for [yield]. Show the scaling factor and changed quantities. Flag method, pan, temperature, and timing details that need review.

Add a supplied restaurant recipe: Format this recipe without inventing missing information: [recipe]. Ask for essential gaps and preserve the contributor and verified notes I supply.

The method uses an unlisted ingredient. Add it only after confirming the intended quantity and unit. Run the two-way ingredient-method check.

Total time is inconsistent. Separate active work, cooking, resting, and overlapping steps. Use the source timing rather than an automatic sum.

Conversions look wrong. Use ingredient-specific conversions when available; otherwise preserve the original unit.

Page numbers are wrong. Update the contents after final layout. Use recipe IDs while the collection is still being edited.