EPORDO

EPORDO | Cloud Attendance | Payroll

Payroll Complete User Guide

Full operational reference - menus, fields, calculations, FAQ, and go-live checklist

EPORDO Payroll - Complete User Guide

Audience: HR staff, payroll clerks, implementation consultants

Style: Written against the **current product UI**-what to click, what to fill, what happens next.

UI language: Set the interface to **English** (or your deployment language) before following this guide.

Version: September 2026 | Aligned with current product release

Related docs: Documentation Index | Overview & Framework | Quick Start

---

Contents

1 What you can do with it

2 Concepts you must know first

3 Menus and recommended setup order

4 Payroll Settings - the six tabs

5 Salary Assignment

6 Payroll Processing - period to lock

7 Payslip Inquiry

8 How amounts are calculated

9 Typical setup examples

10 FAQ and common mistakes

11 Go-live checklist

---

1. What you can do with it

First principle: There is no hidden "default country tax". What you configure is what the engine uses; what you omit does not exist for the engine.

Start from a blank tenant and configure in order. The product **does not** offer a one-click "Import demo template" button-you define rules that match your jurisdiction.

---

2. Concepts you must know first

```
 Period      = Date range for this pay cycle (e.g. 2026-07-01 ~ 07-31)
Structure    = How to calculate: components, formulas, required inputs
Assignment   = Who uses which structure, base salary, and tax slab
Run          = One calculation job under a period (with a config snapshot)
Policy       = Engine-level parameters (rounding, pay frequency for tax annualization)
```

One run may include many structures (each employee uses their own assignment).

Mid-period structure changes do not auto-split one employee into two segments-align changes to period boundaries.

---

3. Menus and recommended setup order

Under **Payroll**:

Recommended setup order

The Settings page onboarding alert matches this sequence:

Policy -> Input defs -> Provider catalog -> Income Tax Slabs -> Components -> Structure, then Assignment and Processing.

```
 (1) Payroll Policies
(2) Input Definitions
(3) Provider Catalog (browse sources; usually read-only)
(4) Income Tax Slabs (skip if no automatic income tax)
(5) Salary Components
(6) Salary Structures
(7) Salary Assignment
(8) Payroll Processing (every cycle)
(9) Payslip Inquiry / export
```

> **New to EPORDO Payroll?** Follow the hands-on path in Quick Start Guide first.

---

4. Payroll Settings - the six tabs

Open **Payroll Settings**. Tab order matches the product.

4.1 Payroll Policies

Engine-level parameters-not "how basic salary is defined".

What affects calculation today: rounding fields, default flag, and pay frequency (mainly for tax annualization). If the UI notes that LWP/OT JSON or working-days mode are not wired into the engine yet, implement attendance logic via **Input Definitions + component formulas + Provider collect**-do not rely on unused policy JSON.

Practical note: For monthly pay with join/leave proration by calendar days, policy join/leave mode often uses "by payment days". For weekly/daily pay where `payment_days` already reflects actual days worked, avoid double proration.

When creating a policy, enter code and country yourself (no prefilled demo country template).

4.2 Input Definitions

Answer: **What quantities does this cycle need, and where do they come from?**

System payment-day inputs (platform-maintained; UI locks delete/deactivate):

`working_days`, `payment_days`, `employed_payment_days`, `period_payment_days`

Keep them if components use "depends on payment days" or join/leave proration. Types must remain numeric.

Principle: If it is not defined, the engine will not use it-e.g. if you do not define holiday days, no holiday deduction will occur.

4.3 Provider Catalog

Read-only catalog of built-in sources (attendance reports, employee fields, ...). Use **Test collect** with Employee ID + date range to verify outputs.

You do not create providers here; binding happens on **Input Definitions** when Source = Provider.

Collection behavior (overview):

- 12 Full calendar months prefer monthly reports
- 13 Partial months aggregate daily data
- 14 Cross-month ranges split then merge
- 15 Missing reports do not invent data

4.4 Income Tax Slabs

Maintain progressive rows: **From amount / To amount (empty = no upper limit) / Rate %**.

A slab alone does nothing. Tax appears only when **all** of the following are true:

- 16 A **Deduction** component uses calc type **Tax slab**
- 17 That deduction is sequenced **after** all taxable earnings in the structure
- 18 The employee's assignment selects an **Income tax slab**
- 19 At least one earning is marked **Taxable earning** and has a calculated amount

If no slab is selected on assignment, tax is typically **0** (the engine does not guess).

4.5 Salary Components

A component is one calculated line on the payslip.

Type

Calc type

For **Fixed**, amount source is often assignment base salary / fixed amount / formula.

Optional flags: depends on payment days, join/leave proration, condition expression, taxable earning (earnings only).

Grid status: **Active / Inactive**.

Use codes with letters, digits, and underscore only-avoid spaces and operators so formulas remain reliable.

4.6 Salary Structures

A structure = **ordered component rows** + **required inputs** + optional structure overrides.

Steps:

- 20 Select a **Policy**, set **Currency** (you must enter this-no country default currency).
- 21 **Add component row**; sequence controls calculation order (save often renumbers 10, 20, 30...).
- 22 Mark **required inputs**

that must exist before calculate.

- 23 Optionally override amount/formula/condition per structure;
  Tax slab deductions cannot be overridden
  at structure level.

Ordering rules:

- 24 Referenced components must calculate first
- 25 **Tax slab deduction must follow all taxable earnings**
- 26 Fix sequence errors shown on save

Usually keep **one** "assignment base salary" Fixed component per structure; other fixed lines use fixed amount or formula.

---

5. Salary Assignment

Menu: **Salary Assignment**.

5.1 Single assignment

Use **Add Assignment** (or equivalent):

5.2 Batch assignment

Use **Batch Assignment**:

- 27 **By employees**
  : multi-select people; shared structure/salary/slab
- 28 **By departments**

: same assignment for employees in selected departments (
not
sub-departments, per UI hint)

- 29 Option to
  skip employees with overlapping assignment
- 30 Review success / skipped / total after
  Apply Batch

Department scope follows the user's department permissions.

---

6. Payroll Processing - period to lock

Menu: **Payroll Processing**. Step bar:

Period & Run -> Prepare Inputs -> Calculate -> Approve -> Lock

6.1 Step 1: Period & Run

Payroll Period

Period scope filter: **Active** / **Include closed**. You cannot **Create Run** on a closed period until it is reopened.

Processing Run

Run status: **Draft -> Calculated -> Approved -> Locked**, plus **Void**.

If you see *"No employees with valid assignment"*, finish **Salary Assignment** first (base salary > 0, effective dates covering the period).

6.2 Step 2: Prepare Inputs

Guided order: **Preflight -> Auto collect -> Manual entry -> Saved overview**.

- **31** **Calculation preflight**

  Review employee count, saved inputs, and gaps for Provider vs manual sources (sample missing rows).

- **32** **Auto collect (Provider)**

  - Click **Collect provider inputs**

  - Options: allow partial missing / fail on first error; **Skip existing values**

  - Review Written / Missing result table

- **33** **Manual entry**

  - Shows only Manual inputs referenced by the structure

  - Use **Batch Fill** -> **Apply to Filtered** for bulk entry

  - **Must click Save Inputs**-values are not saved until you do

- **34** **Saved inputs overview**

  Filter by employee code, input item, source (Manual / Provider).

6.3 Step 3: Calculate

Click **Calculate** or **Recalculate**.

Hard rule: **Every** employee in the run must have **all** required inputs saved. Any missing item **blocks the whole batch**-partial success is not supported. The UI lists missing rows so you can return to Step 2.

On success, status becomes **Calculated**; review gross / deduction / net in the preview (full list also in Payslip Inquiry).

6.4 Step 4: Approve

Click **Approve**. Inputs are generally frozen after approval; you can still **Lock** or **Revert to Draft** (follow confirmation dialogs).

6.5 Step 5: Lock

Click **Lock**. Payslip snapshots are saved in the **current UI language** and become immutable.

To correct a locked run: **Void Run**, then create a new run and recalculate.

---

7. Payslip Inquiry

Menu: **Payslip Inquiry**.

Filters

- 35 Payroll Period, Processing Run
- 36 Status: All / Draft / Calculated / Approved / Locked / Void
- 37 Department, employee code / name keyword

The list shows **created** time.

Actions

Detail layout: Employee Earnings & Deductions and **Employer Contributions (CTC)** are shown in separate sections. Employer contribution lines are not paid to the employee and are excluded from employee gross/net.

---

8. How amounts are calculated

Calculation pipeline

- 38 Load the run's **configuration snapshot** (structures, components, policies)
- 39 Load each employee's **saved inputs** and assignment (base salary, tax slab)
- 40 Evaluate components in structure sequence (Fixed / Percentage / Formula / Manual input / Tax slab)
- 41 Components with a false **condition** evaluate to 0
- 42 Apply policy rounding to components and/or net
- 43 Sum gross, deductions, net; employer contributions separately

8.1 Payment-day proration (Fixed-style)

When a component depends on payment days, a common pattern is:

`amount × (payment_days / working_days)`

With join/leave proration enabled, multiply by `SYS.proration_factor` (derived from employed payment days vs full period days).

If `working_days = 0`, related proration yields **0** (no divide-by-zero).

8.2 Income tax (Tax slab)

- 44 Sum all **Taxable earning** amounts for the period -> period taxable base
- 45 Annualize using pay frequency (e.g. monthly × 12)
- 46 Apply progressive slab to annual amount -> annual tax
- 47 Divide by periods per year -> period tax
- 48 Write result to the tax-slab deduction component

Pay frequency is taken from **Structure** first, then **Policy**-do not rely on Period settings alone for tax.

8.3 Formula variables (for configuration)

Use **Validate Expression** in the formula editor before save.

---

9. Typical setup examples

Example A: Monthly pay + base + housing % + OT formula + income tax

- 49 **Policy** : Monthly; round component and net to `0.01`; half up; both.
- 50 **Input defs** : four payment-day inputs (Manual or Provider); overtime hours; nationality if needed for conditions.

- 51 **Tax slab**
  : progressive rows per local rules.
- 52 **Components**
  :

  - BASIC - Fixed, assignment base, optional proration

  - HRA - Percentage of BASIC

  - OT - Formula referencing overtime input

  - TAX - Deduction, Tax slab

  - Mark taxable earnings per local law (often BASIC; HRA may or may not be taxable)
- 53 **Structure**
  : BASIC -> HRA -> OT -> ... -> TAX; mark required inputs.
- 54 **Assignment**
  : structure, base salary, tax slab per employee.
- 55 **Processing**
  : Create Period -> Create Run -> collect/manual + Save Inputs -> Calculate -> Approve -> Lock.

Example B: Fixed salary only, no income tax

Skip Income Tax Slabs and tax-slab deduction components. Structure contains earnings and ordinary deductions only. Assignment needs no tax slab.

Example C: Weekly pay with attendance-based OT

- 56 **Policy**

: Weekly pay frequency.

- 57 **Input defs**
  : OT hours (Provider from attendance daily report).
- 58 **Components**
  : BASIC (Fixed), OT (Formula: `IN.ot_hours * hourly_rate`).
- 59 Ensure policy/structure frequency aligns-mismatch with period labels can cause incorrect tax annualization if tax is added later.

---

10. FAQ and common mistakes

---

11. Go-live checklist

Configuration

- 60 Policy frequency and rounding match company rules
- 61 Input definitions cover all required structure inputs; Manual vs Provider sources correct
- 62 Provider outputs match variable mappings (spot-check with Test collect)
- 63 Component types, calc types, taxable flags correct
- 64 Structure order validates on save; tax slab after all taxable earnings
- 65 Assignments non-overlapping; tax slab assigned where needed; base salary > 0

- 66 Currency set correctly on each structure

One pilot cycle

- 67 Active period -> Create Run -> preflight -> collect/manual -> **Save Inputs**
- 68 Calculate succeeds; spot-check gross/deductions/net vs manual calculation
- 69 Approve -> Lock; Payslip Inquiry Excel/PDF export OK
- 70 Switch UI language and confirm labels (status, source, calc type) display correctly

Access control

- 71 Payroll menus and action buttons match role design
- 72 Department data scope correct for batch assignment and inquiry

---

Appendix A - UI term reference

Appendix B - Related documentation

For engine fields, DDL, and Provider ADR, refer to your technical documentation index- not required for day-to-day operation.