POWER PLATFORM

A time clock that catches labor issues before payroll

Built on Power Platform for hourly staff — where the interesting engineering happens between the punches.

Duarte Clemente · Power Platform & Azure consultant · Simple Smooth Safe

CONTEXT

The shift in thinking

The clock-in button is the boring part. The value is in watching shifts as they happen.

1

The old way

A missed meal break surfaces at payroll, two weeks late — when it's expensive to fix.

2

The new way

The system watches active shifts in near-real-time and flags issues as they occur.

3

Why it matters

Compliance is far cheaper to enforce live than to reconstruct after the fact.

MECHANICS

What runs between the punches

1

15-minute sweep

A background flow checks every active shift on a recurring cycle.

2

Thresholds as config

Break-due and overtime limits live in environment variables.

3

Tune per state

Rules change per jurisdiction with no flow edits at all.

4

Duplicate-punch guard

A double tap can't open two overlapping shifts.

5

14-hour safety net

A shift left running is auto-closed before it corrupts payroll.

6

One data model

Shift + child break records, shared across every surface.

ARCHITECTURE

Designed so only one thing changes

The data model is fixed. The part that's allowed to evolve is deliberately isolated.

1

Stable core

The same shift and break tables back the clock whoever punches, wherever.

2

Swappable edge

Only the payroll export is meant to change — manual today, an API tomorrow.

3

Cheap to extend

New surfaces and new export targets don't touch the compliance logic.

TAKEAWAY

Enforce compliance live, not in hindsight.

Catching a problem while the shift is open is worth ten reports that find it after payroll. Where are you catching things late that you could catch live?