FaDE — Facility Design Engine

Project Development Report

Author: Shubham Gambhire • IIM Mumbai — MBA (Operations & Supply Chain)

Executive Summary

FaDE (Facility Design Engine) is a browser-based facility layout planner that combines **manufacturing layout design** and **warehouse layout design** into a single tool with shared analytics and export layers. It began as an academic experiment in heuristic factory layout, matured into a warehouse-only design engine, and — over the course of this project — grew into a unified facility planner covering SLP relationships, CRAFT-style refinement, WaDE rack fill with ABC slotting, industry-standards compliance, economics, and one-click exports.

Project Progression

The project evolved as a single continuous line of work rather than isolated versions. Each stage kept what worked, discarded what didn't, and layered on new capability. The stages below describe the progression from the earliest prototype to the current tool.

Stage 1 — Heuristic Factory Layout Prototype

Goal: Prove that a classical SLP + CRAFT pipeline could be run interactively on a small factory grid.

Built as a Streamlit + Python prototype. The user entered a list of departments, a Muther relationship matrix (A/E/I/O/U/X), and floor dimensions. The engine placed departments greedily by total closeness rating, then ran CRAFT pairwise swaps to reduce a distance × weight moment score. A simple matplotlib canvas rendered coloured blocks.

Learnings: SLP + CRAFT works, but factories are only half the problem — warehouses need a different engine because racks, aisles, docks and pallet slotting don't fit CRAFT's swap model.

Stage 2 — Warehouse Design Engine (WaDE)

Goal: Design real warehouses end-to-end: zones, racks, docks, ABC slotting, KPIs.

Reframed the tool around warehouses. SLP was retained for zone-to-zone placement (Receiving, Storage, Picking, Packing, Shipping, Returns, Value-Added), but rack layout inside a zone was solved by a WaDE-style rack fill: rack pairs back-to-back with an aisle between them, ABC slotting based on pick frequency, and dock strips on exterior walls. Industry presets (E-commerce fulfilment, Cold storage, Automotive/3PL, General distribution) seeded typical zone sets and rack parameters.

Also introduced: building shape (Rectangle / L / U) via VOID cells, entry/exit points on perimeter, regional standards compliance (OSHA 1910.36 / 1910.176, NFPA 101, FM Global 7-45, Muther SLP), and per-zone financial roll-ups.

Stage 3 — Rewrite to a Modern Web Stack

Goal: Turn WaDE from a local Python app into a shareable web tool without losing engine fidelity.

Streamlit/Python was replaced with a TanStack Start + React 19 + TypeScript stack, styled with Tailwind v4. All layout engines were rewritten in TypeScript so everything runs in the browser — no server calls for the core solver. This unlocked a proper multi-page UI (Home, designers, Analytics, Export, Help), an in-memory layout cache for tab switches, and 2D canvas rendering plus an isometric 3D viewer for warehouses.

Stage 4 — Manufacturing Module Rebuilt Inside the Same Tool

Goal: Bring the factory design flow back, but as a first-class module sharing UI, analytics and export with the warehouse module.

Implemented the manufacturing designer with departments (machine count, machine size, aisle allowance, time/unit, fixed-position preference), a Muther relationship matrix, entry/exit editor, guided setup dialog, and presets for the four canonical layout types:

- Process Layout — same-function equipment pooled into functional areas; wide aisles because routing varies.
- Cellular Layout — U-shaped group-technology cells producing a part family end-to-end.
- Product Layout — dedicated production line(s); machine count drives parallel lines.
- Fixed-Position Layout — large/heavy product stays put, resources come to it.

The solver runs SLP block-placement first, then CRAFT pairwise swaps that keep the change only when the total flow moment drops. Machine count × machine size + aisle allowance determines each department's footprint so a zone can't be split across the plan.

Stage 5 — Unified Analytics

Goal: Give both modules a shared economics and comparison layer.

Created a single Analytics page with two modes — Manufacturing and Warehouse — that read from a common snapshot format persisted in the browser. Metrics include:

- Manufacturing — bottleneck identification, per-department utilization %, CAPEX / OPEX rollups, CRAFT-based material-handling cost, cost per unit.
- Warehouse — per-zone pallet positions, $/pallet position, cube utilization, amortized cost per pallet-year, payback.
- A/B compare — snapshot two iterations per module, view a delta report, and export any report to CSV.

Stage 6 — Export Centre and Help / About

Goal: Make the tool publishing-ready with clean downloads and clear documentation.

Added an Export page that captures the last generated layout in each module as a PNG plan, a full Excel workbook (Zones/Departments, KPIs, per-item Economics, Totals, Assumptions), and a raw JSON snapshot. Refreshed the Help & About page to reflect the current engines, the four manufacturing layout types, standards compliance, and creator information.

Core Capabilities Today

Area	What the tool does
Manufacturing designer	SLP + CRAFT solver; departments with machine count/size/aisle/time-per-unit/fixed pref; presets and guided setup for the 4 layout types.
Warehouse designer	SLP zone placement + WaDE rack fill with ABC slotting; building shape (Rect/L/U); dock strips on perimeter; industry presets.
Bulk racking systems	Selective, Double-deep, Drive-in, Push-back, Pallet flow, VNA — each with system-specific aisle/depth specs and cost multipliers.
Optimization goals	Minimum travel time, workforce efficiency, maximum throughput, storage density — drive aisle spacing, cross-aisle frequency, ABC split.
Standards compliance	Built-in rules (entry/exit, X-pair separation, zone coherence, aisle coverage, utility anchoring) plus uploadable regional standards JSON.
Analytics	Utilization, bottlenecks, CAPEX/OPEX, $/pallet-position, payback; A/B scenario compare with delta report.
Export	PNG layout, full XLSX workbook, JSON snapshot — all captured client-side.

Standards & Compliance

Every generated layout is checked against a rules module. Errors trigger an automatic repair pass; anything still failing is reported in red on the compliance panel.

- OSHA 29 CFR 1910.36 — designated ingress in every workplace.
- OSHA 29 CFR 1910.176(a) — permanent aisles marked and kept unobstructed.
- NFPA 101 — marked exits from every occupied space.
- FM Global 7-45 — separation between incompatible processes.
- Muther SLP — X-rated pairs must be non-adjacent, ideally separated by a U/O buffer.

Custom / regional rules can be uploaded as JSON via the Standards Uploader and are evaluated alongside the built-ins.

Issue Log & Fixes

The following issues were identified and fixed during the project. They are listed chronologically.

Issue	Resolution
Warehouse rack fill was ~10% of usable area	Rewrote fillZoneRacks for high-density back-to-back rack pairs with configurable

	aisle width.
Docks and aisles appeared in a checkerboard pattern	Placed docks in continuous strips along exterior walls matching real warehouse practice.
Large grey unallocated areas remained in Max Density mode	Introduced an automatic Bulk Storage zone that absorbs remaining floor space (excluding zones and gates).
Racks appeared in zones with rack-fill disabled	Enforced the per-zone rack-fill checkbox in the solver and added a keep-out buffer.
Inaccessible double/triple racks at bottom of plan	Added accessibility check — a rack pair must have an aisle adjacent on at least one long side.
Manufacturing inputs were clipped (M size, aisle, fixed pref)	Widened the department table cells and fixed number-spinner styling.
Black block appeared in warehouse preview	Replaced VOID rendering with a hatched "UNBUILDABLE" pattern.
Analytics → Warehouse tab crashed on load	Added schema validation on snapshot load and switched to Number.isFinite for payback math.
Layout vanished after switching to Analytics/Export and back	Introduced a module-level layoutCache so the last generated layout survives route changes in the same session.
Exported XLSX file downloaded but wouldn't open	Sanitised NaN/Infinity/undefined cells, switched to Blob download with delayed URL.revokeObjectURL, and validated the output opens cleanly.

Uses of the Tool

- Early-stage facility planning — sketch a manufacturing or warehouse layout in minutes with defensible SLP / rack logic behind it.
- Design comparison — snapshot two iterations of the same facility and compare CAPEX, OPEX, utilization, pallet economics and payback side-by-side.
- Teaching operations & supply-chain concepts — SLP, CRAFT, ABC slotting, rack systems, standards compliance are visible and interactive.
- Bid / proposal support — export a PNG plan and an XLSX workbook of KPIs and economics for client-ready deliverables.
- Regional standards evaluation — upload a JSON rules file for local codes and see which rules a given layout satisfies.

About Us

Shubham Gambhire

MBA — Operations & Supply Chain Management, IIM Mumbai

Creator & Developer of FaDE (Facility Design Engine)
Email: shubhambgambhire@gmail.com
LinkedIn: linkedin.com/in/shubhambg