MASTER PRD — AI-Driven Building Design, Analysis, Estimation & Construction Management Platform
(Architecture → Structural Design → Estimation → Execution → Closure)

---

1. Executive Summary

This application is an **end-to-end AI-powered engineering ecosystem** that:

1. **Collects high-level building requirements** from the user (e.g., "Construct a G+2 Building in Hyderabad").
2. **Converses intelligently** to gather missing details.
3. **Generates preliminary building layouts**, or
   imports user sketches, hand drawings, or CAD-like plans.
4. **Automatically understands structural components** needed:
   - Columns
   - Beams
   - Slabs
   - Foundations
   - Staircases
   - Plinth beams
   - Lintels
   - Water tanks
   - Compound walls / retaining walls
   - Boundary drainage
   - Roof structures
   - PEB/steel members (if industrial)
5. Runs **engineering design analysis** using:
   - **Kratos Multiphysics (preferred)**
   - **FEniCSx (optional)**
   - **Custom solvers**
   - Embedded Indian Standards (IS 456, 800, 875, 1893, 2911...)
6. Produces:
   - Full **structural design outputs**
   - Member sizing
   - Detailed calculations (hand-calc style for approval)
   - Drawings (architectural + structural)
   - Reinforcement detailing
   - BOQs, schedules, bar bending schedules
7. Performs **estimation** using local **SOR/SSR** rates.
8. Creates a **complete project execution plan** with:
   - Gantt planning
   - Material planning
   - Cost control
   - Procurement
   - QA/QC
   - Safety
9. Allows customers to pick **only the modules they need**, each separately billed.

This PRD defines a **unified platform** capable of handling:

- PEB buildings

- RCC buildings
- Steel/composite buildings
- RE walls, retaining walls
- Landfills
- Industrial layouts
- Architectural planning
- Estimation-only mode
- Execution-only mode

---

2. High-Level Problem This Solves

Most customers face fragmented workflows:

Stage	Problem
Requirement	They don't know what drawings/data engineers need.
Planning	They struggle to visualise the building early.
Structural Design	They rely on external firms or complex tools like STAAD.
Detailing	Long turnaround, high cost.
Estimation	SSR/SOR complexity, inflation factors, regional differences.
Execution	No unified workflow, no cost/schedule control.

Your app unifies all stages in **one intelligent platform**.

---

3. Target Users

1. **Residential Customers** (home construction, G+1, G+2, etc.)
2. **Industrial Clients** (PEB factories, warehouses, logistics hubs)
3. **Architects**
4. **Structural Engineers**
5. **Contractors**
6. **PMC/Developers**
7. **Government/municipal projects**
8. **Students & trainees**

---

4. System Modules (Top-Level)

The system consists of 8 super-modules:

4.1. AI Requirement Engine

- Conversational agent powered by LLM (ChatGPT / Gemini / Custom).
- Asks progressively detailed questions:
  - Plot size
  - FSI/Byelaws
  - Setbacks
  - Rooms required
  - Occupancy
  - Road width
  - Soil type
  - Budget
  - Industrial function (if any)
- Auto-detects missing data and requests clarification.

Outputs:

- Requirements sheet
- Auto-generated design brief
- Constraints list (local bye-laws, IS code triggers)

4.2. Layout / Plan Generator

Two modes:

1. **Pre-stored building templates**
2. **Custom layout from:**
    - Hand sketch (image upload)
    - PDF
    - AutoCAD / DXF
    - AI-generated layout (LLM + geometry engine)

Outputs:

- Preliminary 2D architectural plan
- 3D concept model (optional)
- Room/space schedule
- Staircase & circulation

4.3. Structural Understanding Engine

After layout → platform maps structural elements:

Automatically identifies:

- Beam lines
- Column grids
- Slab spans
- Staircase geometry
- Load-bearing walls
- Foundation zones
- Soil-structure interaction requirement
- Roof truss or PEB frames (if industrial)
- Retaining walls or compound walls

This engine uses:

- Geometric rules
- AI-assisted inference
- Template patterns
- Graph-based connectivity extraction

Outputs:

- Structural model skeleton
- Analytical model input representation (JSON)

4.4. Multi-Solver Structural Analysis Engine

The heart of the app.

Supported solvers:

- **Kratos Multiphysics (preferred primary engine)**
- FEniCSx (secondary optional)
- Linear solver fallback (NumPy/SciPy)
- FEM Beam solver for simple cases

- AI-assisted approximate solver for instant previews

Supported structure types:

- RCC framed buildings
- Steel structures
- PEB structures
- Composite buildings
- Retaining walls
- RE walls
- Landfills (stability modelling)
- Industrial steel sheds
- Platform/pipe racks

Loads:

- IS 875 Dead/Live/Wind
- IS 1893 seismic
- IS 456/RCC
- IS 800 Steel
- IS 2911 foundations
- Soil parameters for retaining walls
- Vehicle/UDL loads
- Crane loads
- Temperature loads
- Settlement / differential loads

Outputs:

- Reactions
- BM/SF/AF diagrams
- Storey drift
- Base shear
- Displacements
- Stress contours (Kratos/FEniCSx)

4.5. Design & Code Check Engine

Performs *full compliance* with:

- IS 456: RCC
- IS 875: Loads
- IS 1893: Earthquake
- IS 800: Steel design
- IS 801: Light gauge
- IS 11384: Composite
- IS 2911: Foundations
- MoRTH retaining wall
- CPHEEO/CPCB landfill guidelines
- ACI/AISC/EC support for special cases

Produces:

- Full member design (beam, column, slab, footing)
- Foundation sizing
- Link/stirrup design
- PEB rafter/column/connection design

- Shear wall design
- Water tank design

Includes:

- Hand-calculation style narrative
- Agency-approvable output
- Auto-generated SOR/SSR-based BOQ

4.6. Drawing & Detailing Engine

Generates:

- Architectural drawings
- Structural GA drawings
- Beam reinforcement
- Column schedules
- Slab detailing
- Footing detailing
- Staircase drawings
- Sectional elevations
- PEB shop drawings
- Connection detailing
- BBS (bar bending schedule)
- Landfill bund/bench profile drawings

Exports:

- PDF
- DWG/DXF
- SVG
- Revit-compatible IFC (Phase 2)

4.7. Estimation & BOQ Engine

Based on:

- SSR / SOR
- Custom client rates
- WPI/CPI escalation
- Productivity norms

Produces:

- BOQ
- Abstract estimate
- Material quantity schedule
- Labour costing
- Equipment costing
- Item rate analysis
- Cost comparison (options A/B/C)

User can:

- Jump directly to estimation
- Skip design modules

4.8. Project Execution & Management Engine

Includes:

- WBS generation
- Scheduling (MSP-like)
- Cash flow forecast
- Material procurement
- QA/QC checklist
- Safety plan
- Site diary
- Inspection & approvals
- Progress dashboard
- Risk register
- Cost control (earned value)
- Project closure documentation

---

5. Workflow Summary (User Journey)

Step 1: User describes requirement

Example:

"I want to construct a G+2 building in Hyderabad."

Step 2: AI asks necessary questions

- Plot size
- Rooms
- Soil
- Loads
- Budget
- Industrial/residential
- Aesthetic preferences

Step 3: User chooses

- Pre-defined layouts **OR**
- Uploads sketch **OR**
- Auto-generate plan

Step 4: Structural intelligence engine maps components

- Generate structural grid
- Identify beams, columns, walls, slabs
- Identify foundation type
- Prepare analytical model

Step 5: Run structural analysis

- Use Kratos
- Multiple load cases
- Code compliance checks
- Preliminary sizing

Step 6: Drawing generation

- Architectural layout
- Structural drawings
- Reinforcement details
- PEB drawings (if industrial)

Step 7: Cost estimation

- SSR/SOR
- BOQ

- Abstract estimate
- Option comparison

Step 8: Project execution

- Scheduling
- Material planning
- Procurement
- QA/QC
- Safety
- Cash flow
- Monitoring until closure

---

6. Product Architecture

Frontend:

- React or Next.js
- OR Streamlit for early MVP

Backend:

- Python
- Kratos solver container
- FastAPI
- Postgres/Firestore DB
- Message queue for heavy jobs

AI Integration:

- ChatGPT API
- Gemini API
- Custom RAG using vector DB
- Prompt templates per module

Compute:

- Dockerized Kratos & Python
- HPC-ready workload
- Auto-scaling

---

7. Monetization Model

Per-module pay-per-use

- Layout generation
- Structural design
- Detailed drawings
- Estimation
- Project execution suite
- QA/QC
- Specialized modules (PEB, water tank, RE wall, etc.)

Subscription tiers

- Basic (students)
- Professional (engineers)
- Enterprise (construction companies)
- API access (SaaS)

---

8. Deliverables

For each module, system generates:

- PDFs
- CAD drawings
- Excel estimation sheets
- Schedules
- Structural calculation reports
- Approvals-ready documents
- 3D model (future)

---

9. Phase Plan

Phase 1 – MVP

- AI requirement engine
- Layout generator
- Basic structural skeleton
- Simple Kratos integration
- Estimation engine
- PDF output

Phase 2

- Full structural design (IS codes)
- Detailed drawings
- Multi-solver capability
- Project execution module

Phase 3

- BIM/IFC integration
- Revit plugin
- Contractor marketplace
- Full mobile app

---

10. Conclusion

This platform becomes:

India's first complete AI-driven construction solution

– from requirements → planning → design → estimation → execution → closure.

Your app becomes:

- An architect
- A structural engineer
- A PMC
- A cost estimator
- A QA/QC team
- A safety officer
- A schedule controller

All in one unified platform.

📘 **PRD — PART 1**
MASTER CONCEPT, VISION, AND HIGH-LEVEL PRODUCT BLUEPRINT
(For Unified Building Design + Estimation + Construction App)**

---

1. PRODUCT NAME (Working Title)

"Astra Construct AI" — Unified Architecture, Structural Design, Estimation & Construction Management Platform

(You may finalize the name later. All modules are generic.)

---

2. EXECUTIVE SUMMARY

Astra Construct AI is an end-to-end **building design → structural engineering → estimation → project execution** platform powered by:

- AI-guided requirement discovery
- AI-driven architectural layout generation
- Automated structural modeling
- Code-compliant design using multiple engines (Kratos Multiphysics, FEniCSx, or custom FE)
- Auto-drafting (2D working drawings + 3D BIM)
- Rate analysis, quantities & estimation using SSR/state schedules
- Project execution workflows
- Module-wise pricing for SaaS deployment

The system acts as a **virtual design office**, **engineering consultant**, and **project manager** for residential, industrial, and commercial buildings.

---

3. USER JOURNEY — HIGH-LEVEL WORKFLOW

STEP 1 — User describes project

Example:
"Construction of G+2 building in Hyderabad."

STEP 2 — AI questionnaires

AI interacts with the user to collect complete design intent:

- Site location
- DD/DP rules, setbacks, permissible FAR
- Building usage (residential, commercial, warehouse, industrial, mixed-use)
- Span preferences, parking requirements, ventilation needs
- Architectural preferences (modern, regular, traditional)
- Structural considerations (seismic zone, soil type, loads)
- Internal spaces: #Bedrooms, hall, kitchen, toilets, services
- Staircase selection
- Water tanks, sumps, compound walls, retaining walls
- Basement preference, ramp
- Electrical/plumbing preferences
- Budget range (optional)

STEP 3 — App proposes standard layout templates

System displays **pre-stored standard layouts**:

- G+1 / G+2 typical plans
- Industrial sheds (PEB/Steel/RCC)

- Duplex/villa plans
- Small apartment layouts
- Warehouse layouts
- Commercial building templates
- Mixed-use building templates

User can:

- **Select a template**, or
- **Upload their own sketch**, or
- **Upload a plan from anywhere**
  (JPEG, PNG, PDF, CAD, scanned drawings)

AI will auto-understand the plan and recreate it.

STEP 4 — Architectural Plan Generation

AI generates:

- Floor plans
- Setback-compliant site plan
- Elevations
- Sections
- Staircase details
- Room schedules
- Sanitary schematics
- Electrical outlet plan
- Plumbing plan
- Parking layout
- Schedule of finishes

Each step is priced independently.

STEP 5 — Structural System Understanding

AI identifies all building components required:

- Foundations (isolated, combined, raft, pile)
- Plinth beams
- Columns
- RCC beams
- Slabs (one-way, two-way, waffle, PT optional)
- Staircases
- Lintels & sunshades
- Shear walls (if required)
- Retaining walls
- Compound walls
- Water tank (UG + OH)
- Roof slabs
- Ramps
- PEB/Steel/RCC frames for industrial buildings

User can override AI decisions.

STEP 6 — Structural Modelling (Kratos Multiphysics preferred)

App will:

- Build the complete parametric model
- Assign materials
- Generate FE mesh
- Apply gravity, live loads, wind loads (IS 875), seismic loads (IS 1893)
- Solve for stresses, deflections, bending, shear, axial forces
- Perform design checks:
  - IS 456 (RCC)
  - IS 800 (Steel)
  - IS 801 (light gauge)
  - IS 16700 (tall buildings)
  - IS 3370 (water retaining structures)
  - AISC/EC/ACI parts when required (anchor bolts, composite construction)

STEP 7 — Automated Drawings & BIM

The system outputs:

- Structural GA drawings
- Footing layout
- Column layout
- Beam framing plans
- Slab reinforcement
- Staircase drawings
- Retaining wall design & drawings
- Water tank drawings
- Industrial shed drawings
- Connection design sheets
- Bar bending schedules
- 3D BIM model (IFC / GLTF)

All drawings autogenerated from the FE-derived forces.

STEP 8 — Estimation & Costing

Using SSR or local SOR:

- Material quantities (concrete, steel, shuttering, paint, etc.)
- BOQ
- Rate analysis
- Labour cost
- Machinery cost
- Overheads
- GST
- Comparative estimates (multiple design options)

User can skip design and go directly to estimation.

STEP 9 — Project Execution Workflows

Once user finalizes design & estimate:

- Planning & scheduling (Gantt)
- Material procurement planning
- Labour planning
- Daily progress reporting

- RA bills & measurements
- Quality checklists
- HSE checklists
- Photo logs
- Cashflow forecasting
- Earned value management

STEP 10 — Project Closure

- Final completion report
- Snag lists
- As-built drawings
- Asset handover
- O&M manuals

---

4. PRODUCT MODULES — HIGH LEVEL

MODULE 1 — AI Intake & Requirements Engine

Understands natural language, asks relevant queries, and converts to structured project definition.

MODULE 2 — Template Library

100+ building templates: residential, commercial, industrial, warehouses, sheds, retaining wall types, compound walls, etc.

MODULE 3 — Architectural Engine

Generates editable architectural layouts.

MODULE 4 — Structural Modeling Engine

Powered by:

- **Kratos Multiphysics (Preferred)**
- Optional FEniCSx integration
- Future plugin for OpenSeesPy if needed

Supports:

- RCC buildings
- Steel buildings
- PEB frames
- RE walls
- Retaining walls
- Water tanks
- Landfills (slope FE)
- Composite structures

MODULE 5 — Design Engine

IS 456
IS 875 (all parts)
IS 1893
IS 800
IS 3370
AISC / EC / ACI
Composite design

MODULE 6 — Auto-Drawing Generator

2D drawings + BIM.

MODULE 7 — Estimation & Costing Engine
SSR rates (all states) + CPWD
State SOR libraries
Quantity libraries
Machine & labour libraries
MODULE 8 — Project Execution Module
Gantt, DSR, DPR, RA bills, HSE, QA/QC.
MODULE 9 — Project Management Module
Risk, change management, communication, document control, approvals.
MODULE 10 — Marketplace & Pricing
Users pay module-wise depending on what they use.

---

5. BUSINESS MODEL

- Freemium (basic planning + rough estimate)
- Pro (architectural + structural design)
- Premium (complete project mgmt + BIM)
- Pay-per-module:
    - Architectural plan: ₹499–₹999
    - Structural design: ₹1,999–₹9,999
    - Working drawings: ₹1,500–₹7,500
    - Estimation: ₹1,500–₹4,500
    - Project execution suite: subscription ₹499/month
    - Marketplace add-ons: ₹199–₹999

---

6. CORE VISION

To create the **India-specific Staad + Revit + Primavera + Costing + QS + Architect hybrid**, but:

- AI-driven
- Customer friendly
- Fast
- Affordable
- Easy for architects, engineers, contractors, owners

Where a user can say **“Design a G+2 house in Hyderabad”**
and get:

- Plan
- Structural design
- Drawings
- Estimate
- Construction schedule
- Documentation

Automatically.

PRD — PART 1: EXECUTIVE SUMMARY & VISION
(≈ 25–40 pages worth of depth condensed into narrative format. Full expansion available section-by-section.)

1.0 INTRODUCTION

This PRD defines a **Unified Structural Design, Estimation & Project Execution Platform**—a next-generation AI-driven engineering application that handles:

- **Architecture + Structural Design**
- **Civil/Structural Analysis & Code Compliance**
- **Bill of Quantities (BOQ) & Estimation**
- **Drawing Production**
- **Project Workflow, Planning & Execution**
- **Costing, Procurement & Variations**
- **Construction Monitoring & Closure**

The system supports **any construction type**, including:

- **G+0 to G+30 RCC buildings**
- **PEB/Steel buildings & warehouses**
- **Industrial plants**
- **Retaining walls, RE walls, compound walls**
- **Landfills & waste infrastructure**
- **Water tanks**
- **Foundations, pile caps, pedestals**
- **Composite structures**
- **General civil works**
- **Infrastructure facilities (roads, drains, culverts)**

The platform is powered by:

- **AI for requirements capture & architectural synthesis**
- **Kratos Multiphysics** as the **core analysis engine**
- **FEniCSx compatibility layer** (optional future module)
- **Dockerized architecture**, deployable locally or on Antigravity / AI Studio
- **Automated drafting & detailing engine**
- **SSR-based estimation engine** with region-wise rates

2.0 THE HIGH-LEVEL VISION

2.1 The User Journey in One Line

"User describes the building → AI collects specifications → System generates architecture → System designs → System estimates → System delivers drawings → System helps manage execution."

The app becomes a **complete structural design office**, offering:

✓ Architectural planning

From sketches, hand-drawn plans, uploaded images, Google Maps pins, or AI-generated templates.

✓ Structural modelling

Auto-generated structural grids, column layouts, beam mapping, slab systems, staircase logic.

✓ Kratos-based analysis & code checks

Automatic load generation per IS 875/IS 1893/IS 456/IS 800/IRC/ISM.

✓ Detailed design & reinforcement generation
Footings, beams, slabs, columns, walls, tanks.

✓ Drawing production
Auto CAD/DXF + PDF + BOQ sheets.

✓ Cost estimation
SSR rates (TS/AP/KA/TN/MH/NCR/Delhi etc.), CPWD DSR, MoRTH, custom libraries.

✓ Execution & workflow module
Schedules, quality checklists, material trackers, progress dashboards.

✓ Pay-per-module service model
User pays only for the services they use.

3.0 PRODUCT POSITIONING

3.1 What Today's Market Lacks

Existing tools are:

- **Fragmented** (STAAD for analysis, AutoCAD for drawings, Excel for BOQ...)
- **Not beginner-friendly**
- **No AI-based input collection**
- **No end-to-end flow from concept → architectural → structural → costing → execution**
- **No pay-as-you-use models**
- **High learning curve**
- **Difficult for MSMEs, small architects, and contractors**

3.2 What This Platform Provides

Capability	Our System	STAAD / ETABS / Tekla / Revit
AI requirement capture	Yes	No
AI architectural layouts	Yes	No
Multi-material design	Yes	Limited
Automated SSR estimation	Yes	No
Integrated execution module	Yes	No
Pay-per-feature	Yes	No
Cloud-native + Docker	Yes	Limited
Full code compliance	Yes	Yes
Drawing generation	Yes	Partial

4.0 CORE APP WORKFLOW

4.1 Natural Language Input

Example:
"I want to build a G+2 industrial building in Hyderabad, 30m x 20m, with 6m height and a mezzanine. Need retaining wall and water tank."
The AI engine:

1. Interprets location → applies **IS 875 wind zone, seismic zone, SBC ranges**.
2. Interprets occupancy → **industrial load norms**.
3. Asks questions:
    - Soil test available?

   - Basement?
   - Staircase type?
   - Vehicle movement?
   - Roof type?
   - PEB vs RCC?
   - Mezzanine load?
   - Boundary wall?
4. Generates:
   - Building footprint
   - Floor plans
   - Structural grid
   - Column layout

4.2 Template Selection

User can:

- Pick from pre-stored templates
- Upload sketch/photo/PDF
- Upload CAD plan
- Generate layout via AI

5.0 STRUCTURAL SYSTEM GENERATION

After architecture:

- **Column positioning**
- **Beam mapping**
- **Slab system**
- **Shear walls if needed**
- **Retaining wall placement**
- **Foundation type selection**

AI checks:

- Span/depth ratios
- Load paths
- Preliminary beam sizing
- Punching checks (slabs)
- Effective heights (columns)

6.0 ANALYSIS ENGINE (KRATOS)

Why Kratos?

- Multi-domain multiphysics
- Python scripts + C++ core
- Docker compatible
- Perfect for cloud execution
- Supports:
   - Structural mechanics
   - Contact/fracture
   - Geomechanics
   - PFEM (for landfills, slope stability)
   - Thermal (ETPS)
   - Fluid (for water tanks)

Platform Uses Kratos For:
RCC buildings

- 3D frames
- Slabs as plates
- Staircases
- Water tanks
- Shear walls

PEB/Steel

- 3D truss/frame
- Moment connections
- Baseplate checks
- Portal frames
- Crane loads

Geotechnical

- RE walls
- Retaining walls
- Slope stability
- Landfill stability & settlement

Hydraulic structures

- UGT/OGT tanks
- STPs, ETPs

7.0 AI-DRIVEN DESIGN LOGIC

The system automatically:

- Generates load combinations
- Applies load cases
- Runs analysis via Kratos
- Performs code checks:
    - IS 456/13920 (RCC)
    - IS 800 (Steel)
    - IS 875 (Loads)
    - IS 1893 (Earthquake)
    - ACI/AISC/EC alternative codes where IS lacks provisions
- Designs members
- Produces reinforcement schedules
- Creates detailed drawings.

8.0 DRAWING GENERATION ENGINE

Outputs:

- **Architectural drawings**
- **Structural GA**
- **Column, beam, slab schedules**
- **Footing drawings**
- **Retaining wall sections**
- **RE wall drawings**
- **PEB GA + connection details**
- **Water tank drawings**

Formats:

- DXF
- CAD (via libraries)
- PDF
- SVG

Each drawing auto-labels:

- Levels
- Coordinates
- Reinforcement
- Sizes
- Notes
- Sections

9.0 COSTING ENGINE

- Uses **State SSRs**, CPWD DSR, MoRTH
- Auto-maps quantities from drawing + design output
- Generates:
  - BOQ
  - Rate analysis
  - Cost summary
  - Material/Manpower requirement
- User can override rates
- Supports GST slabs & regional taxes

10.0 EXECUTION ENGINE

Once design is done:

- Generates schedule (Gantt)
- Generates checklist (QHSE)
- Procurement plan
- Material tracker
- Daily progress reporting
- RA bills
- Variation orders
- Completion certification
- Project closure documents

11.0 MODULE-WISE PRICING

Each module is a **pay-per-use** micro-service:

- Concept plan
- Architectural layout
- Structural design
- Estimation
- Drawings
- Project management
- QA/QC
- BOQ
- Retaining walls

- Landfills
- PEB buildings
- Water tanks
- Composite structures

User can buy:

- Single module,
- Multiple modules,
- Monthly subscription.

12.0 TARGET USERS

- Civil engineers
- Architects
- Builders & contractors
- Government engineers (PWD/CPWD/Municipal)
- Students
- Structural consultancies
- Real estate developers
- MSME industrial clients

13.0 DEPLOYMENT MODEL

- **Docker-based**
- Works on:
    - Antigravity
    - AI Studio
    - Local desktop
    - On-prem servers
    - Cloud VPS
- All modules containerized:
    - AI engine
    - KRATOS engine
    - Drawing engine
    - Estimation engine
    - UI server

14.0 SECURITY & COMPLIANCE

- All job files encrypted
- No retention unless user enables
- Indian code compliance
- GDPR-ready
- Audit logs for design steps

15.0 SUMMARY

This platform is:

✓ A full replacement to classic STAAD + AutoCAD + Excel

✓ AI-powered, modular, scalable

✓ Suitable for professionals, students, and enterprises

✓ Pay-per-use + microservices

✓ Capable of handling every structure type
✓ Built on Kratos (analysis), Python (logic), Streamlit/Web (UI)

END OF PART 1

PART 1 — EXECUTIVE SUMMARY & VISION

Comprehensive PRD for the AI-Driven Structural Design, Estimation & Construction-Management Platform

---

1.1. Purpose of the Platform

The objective of this platform is to build a **fully automated, AI-driven end-to-end engineering ecosystem** that can:

1. **Capture user requirements intelligently**
   (e.g., "Construction of G+2 building in Hyderabad", "PEB warehouse", "Factory shed", "Landfill cell", etc.)
2. **Interactively gather all necessary inputs** using an AI engine:
   - Site location
   - Purpose and occupancy
   - Dimensions, constraints, height, setbacks
   - Soil conditions
   - Required structural system
   - Levels, storeys, roofing system
   - Local bylaws (GHMC/DTCP/CDMA/etc.)
   - Relevant codes (IS/ACI/AISC/Eurocodes)
3. **Generate concept layouts & architectural schemes**
   - Auto-layout generation
   - Auto-floor plans
   - Auto-column layouts
   - Staircase & lift core positioning
   - Parking layout
   - Water tank / septic tank positioning
   - Boundary, compound wall, retaining wall logic
4. **Structural modeling**
   - Auto identification of members: slabs, beams, columns, footings
   - Auto generation of load cases & combinations
   - Analysis using **Kratos Multiphysics** (preferably via Docker)
   - Multiple structural templates: RCC / Steel / Composite / Hybrid
   - Multi-system support:
     - PEB
     - Steel buildings
     - RCC buildings
     - Retaining walls
     - RE walls
     - Landfills
     - Tanks
     - Special structures
5. **Structural design & code compliance**
   - IS 456
   - IS 800
   - IS 875
   - IS 16700
   - IS 1893

- IRC codes / ACI / AISC / EC3 wherever needed
- Footing design
- Staircase design
- Water tank design
- Boundary wall design
- PEB member & connection design
- Plate/shell design
- Anchor bolt design (AISC/ACI/EC provisions)

6. **Production drawings generation**
    - Architectural drawings (PDF/DWG)
    - Structural framing plans
    - Slab/beam/column schedule
    - Reinforcement drawings
    - Connection drawings
    - PEB fabrication drawings
    - General arrangement (GA) drawings
7. **Estimation & costing**
    - State SSR / CPWD DSR / Local market rates
    - Auto BOQ generation
    - Item-wise cost breakdown
    - Labour, material, equipment
    - Abstract cost sheet
    - GST & taxes
    - Escalation, optional items
8. **Project planning & management**
    - Auto Gantt chart
    - Materials schedule
    - Procurement schedule
    - Bar bending schedule
    - Site progress monitoring
    - Quality checklists
    - Billing management
    - RA bills & measurements
9. **Execution workflows**
    - Pre-construction workflow
    - Approval workflow
    - Site execution workflow
    - Safety (QHSE) workflows
    - Inspection & test plans
    - Material approval chain
10. **Document Control System (DMS)**
    - Upload drawings
    - Version history
    - Approval tracking
    - Distribution matrix
11. **Commercial model (Pay-per-module)**
    - Architectural module

- Structural design module
- Estimation module
- Project planning module
- Drawing production
- Site management module
- Tendering & procurement
- Full package discount

1.2. Grand Vision – One Platform for Any Structure

The goal is to create a **modular, scalable, AI-powered structural engineering ecosystem** capable of handling:

Structure Type	Supported?	Notes
RCC Buildings	✓	G+0 to G+30, apartments, commercial
Industrial Sheds	✓	PEB, steel, composite
Warehouses	✓	PEB or RCC framed
Retaining Walls	✓	Cantilever, counterfort
RE Walls	✓	MSE / geogrid reinforced
Water Tanks	✓	Circular, rectangular
Landfills	✓	Cells, berms, benches
Tanks / Pits	✓	RCC or steel
Footings	✓	Isolated/pad/raft/pile-cap
Staircases	✓	Dog-legged, spiral, flight

Everything runs on a **unified AI + Kratos Multiphysics computation engine**, providing:

- Fast modeling
- Deterministic, code-compliant results
- Automatic detailing
- Continuous design iterations

1.3. Core Concept Flow (User Experience)

Step 1: User Query

Example:

"I want to construct a G+2 RCC building in Hyderabad."

Step 2: AI Input Extraction

AI asks necessary questions:

- Plot size
- Setbacks
- Soil type
- Occupancy
- Floor heights
- Parking needs
- Staircase / lift requirements
- Architectural preferences
- Budget
- Wind/seismic zone auto-detected

Step 3: Concept Layout Generation

- Auto-generated 2D layout
- Optional: upload sketch image to auto-vectorize
- AI-assisted corrections
- Alternate plan suggestions

Step 4: Structural Modeling

- Auto framing
- Auto slab panels
- Auto beam lines
- Auto column grid
- Footings
- Auto detection of openings and load paths
- Boundary / retaining wall logic
- Wind & seismic loads auto-generated
- Kratos FE model created via backend Docker solver

Step 5: Structural Analysis & Design

- FE solution
- Member forces
- Size optimization
- Code checks
- Reinforcement design
- PEB member optimization (if applicable)

Step 6: Drawing Production

- Auto architectural drawings
- Auto structural drawings
- Bar bending schedule
- Connection drawings (PEB)
- Export to DWG/PDF

Step 7: Estimation & BOQ

- Auto BOM
- SSR/DSR rates applied
- Item-wise abstract
- Material consumption
- Labour & equipment cost
- Optional user-uploaded rate library

Step 8: Project Planning & Execution

- Auto Gantt chart
- Daily progress updates
- QA/QC checklists
- Measurement book
- RA bills

Step 9: Lifecycle Management

- Document control
- Approvals
- Change requests
- Site photos
- Completion certificate

1.4. Target Users

User Type	Needs	Platform Offering
Architects	Fast conceptual layouts	AI assisted architectural engine
Structural Engineers	Analysis & design tools	Kratos + Auto code checks
Contractors	BOQ & drawings	Estimation & Site workflows
Developers	Budgets & cost	End-to-end project dashboard
Students	Learning	Free tier with examples
Govt Agencies	Checks	Standardized design methodology

1.5. What Makes This App Unique?

✓ A complete "Conception to Commissioning" tool

No existing platform covers from concept → architecture → structural design → BOQ → project execution.

✓ AI + Kratos Multiphysics back-end

Industry-grade FE solver but low barrier to use.

✓ Modular pricing model

Pay only for the modules used.

✓ Extremely wide structure support

From buildings to landfills to retaining walls.

✓ Multicode support

IS, ACI, AISC, Eurocodes, CPWD/SSR market rates.

✓ Interoperability

Import & export:

- STAAD models
- CAD files
- Sketch images
- DXF / IFC

✓ Production-ready deliverables

Not just FE results—**actual drawings, schedules, BOQ, and site documents**.

1.6. Deployment Vision

The platform will run on:

- **Docker** (backend solver)
- **Kratos Multiphysics docker images**
- **AI Studio or Antigravity** for orchestration
- **PostgreSQL + S3 storage**
- **Frontend: React or Streamlit or Next.js**
- **Backend: FastAPI + Python solver gateway**
- **Separate worker queue for FE simulations**

All components horizontally scalable.

1.7. High-Level Architecture Summary

1. AI Layer

- Input conversation model

- Layout generator
- Structural configuration reasoning
- Code-based logic prompting
- Document drafting

2. Structural Engine

- Kratos Multiphysics solver
- Mesh generator
- Geometry engine
- Load application
- Code checks
- Optimization engine

3. Drawing Engine

- Vector layout generator
- DWG/PDF exporter
- Rebar detailing engine
- Fabrication drawings

4. Estimation Engine

- SSR/DSR parser
- Material library
- Quantity take-off engine
- Abstract cost generation

5. Project Execution Engine

- Planning
- Billing
- Measurement book
- QHSE
- Daily progress

6. Data Management

- DMS
- Version control
- Cloud storage integration

7. Interoperability Layer

- STAAD import/export
- DXF, DWG, IFC conversion

1.8. Commercial Strategy

Freemium

- Limited number of designs
- No drawings/export

Pay-per-Design

- One structure fully designed
- One set of drawings
- One estimate

Subscription

- Unlimited designs
- Project management
- DMS

- Team access

Enterprise

- Unlimited users
- Custom modules
- Integration with ERP

1.9. Summary of Vision

This platform becomes an **AI-powered structural design companion**, capable of:

- Understanding user needs
- Growing with user input
- Handling any structure type
- Producing complete, agency-approvable designs
- Integrating with market rates
- Managing execution till completion

It aims to be the **STAAD + ETABS + AutoCAD + CPWD DSR + MS Project + Site App combined into one intelligent ecosystem**.

✔ Part 1 Complete

PRD – PART 2
SCOPE & SYSTEM CAPABILITIES (DETAILED)**

2.1 Purpose of This Application

The platform is intended to be an **AI-native, full-stack engineering system** that supports:

1. **Form-finding & requirement discovery**
2. **Architectural concept generation**
3. **Structural design & analysis** (Kratos/FEniCSx backend)
4. **Quantity estimation** based on regional SSR/local rates
5. **Construction planning, monitoring & closure**
6. **Commercial workflow** (pay-per-module / usage-based pricing)

It replaces fragmented tools (AutoCAD, STAAD, Excel, manual BOQ, MS Project) with **one integrated ecosystem** that:

- Understands the user's objective ("G+2 building in Hyderabad")
- Asks only the necessary questions
- Generates models, drawings, calculations
- Estimates cost
- And provides project workflows end-to-end.

2.2 High-Level System Summary

The system includes **6 major capability groups**:

A. AI-driven Requirement Extraction

- User enters:
  "G+2 residential building in Hyderabad with 3 bedrooms per floor."
- AI asks targeted questions:
  - Plot size?
  - Setback requirements?
  - Soil data?
  - Parking?
  - Water tanks?
  - Staircase type?
  - Roof usage?
- AI organizes answers into a **structured input matrix** → used for architecture + analysis.

B. Architecture & Layout Engine

Generates:

- Floor plans
- Column grids
- Typical bay spacing
- Circulation cores
- Furniture layout
- Parking layout
- Plumbing trail
- Electrical DB positions
- Staircase geometry
- Water tank plan

- Elevation massing (AI generated)

Users can:

- Choose from **preloaded template layouts**
- Upload hand sketches
- Upload reference PDFs
- AI will digitize → convert into usable CAD for design stage.

C. Structural Design & Analysis Engine (Kratos Multiphysics)

Supports **all structure categories**:

✓ RCC Buildings (G+0 to G+20)

Beams, columns, slabs, shear walls, foundations, staircases.

✓ Steel/PEB Buildings

Portal frames, mezzanines, crane loads, wind/seismic.

✓ Composite Buildings

Steel beams + RCC slabs, metal decks.

✓ Retaining Walls

Cantilever, counterfort, RE walls, boundary walls.

✓ Landfills & Earth Structures

Slope stability, settlement, bund design, leachate loads.

✓ Industrial Structures

Pipe racks, trestles, equipment foundations.

Backend capabilities using Kratos:

- 1D beam elements
- 2D shell/plate elements
- 3D solid elements
- Multi-step load cases (DL/LL/WL/EQ/Water pressure/Soil pressure)
- Eigenvalue analysis
- Non-linear material/geometry handling for special cases
- Meshing + remeshing
- Model-based postprocessing

Code Compliance:

- IS 456, IS 875, IS 800, IS 1893, IS 16700
- ACI 318
- AISC 360
- Eurocodes (EC2, EC3, EC8)
- IRC for retaining walls & bridges (optional)

D. Drawing & Documentation Engine

AI-generated + rule-based creation of:

- Architectural drawings (plans, sections, elevations)
- Structural drawings:
    - Beam/column/footing detail sheets
    - Framing plans
    - Rebar detailing
    - Bar bending schedules
- Steel/PEB drawings:

    - GA
    - Member drawings
    - Connection drawings
    - Anchor bolt plans

Export formats:

- PDF
- DXF
- SVG
- IFC (optional)

Every drawing generated is **traceable** to underlying calculations.

E. Estimation, BOQ & Rate-Based Costing

Once design is frozen:

- Auto-generated BOQ
- SSR-based estimation (user selects state → rates preloaded)
- Vendor/labour local rates support
- Scenario analysis (budget vs premium vs optimized design)

Outputs:

- Detailed cost sheet
- Material + labour heads
- Manpower loading
- Construction equipment loading
- Timeline-linked cash flow

F. Project Execution & Management

Post-estimation:

- Auto-generated project schedule (Gantt)
- Work breakdown structure
- Daily/weekly reporting dashboards
- Quality & safety checklists
- Material tracking
- Contractor bill preparation support
- Document management
- Site photo AI tagging
- Automated correspondence drafting (letters, reports)

Users may use:

- Only design
- Only estimation
- Only project management
- Or the full workflow.

2.3 System Modules (Master List)

2.3.1 Front-End Modules

1. Requirement Discovery Assistant
2. Plan/Sketch Capture Module
3. Layout Generator
4. 2D/3D Visualizer

5. Design Dashboard
6. Estimation Dashboard
7. Project Manager Dashboard
8. Document Report Centre
9. Payment & Subscription Centre

2.3.2 Backend Engines

1. AI Intent Understanding Engine
2. Kratos FEA Engine
3. Code-Check Engine (per IS/ACI/AISC/EC)
4. Layout AI Generator
5. Drawing Automation Engine
6. Quantity Extractor
7. BOQ/Rate Integration Engine
8. Project Scheduling Engine
9. Workflow Automation Engine
10. NLP-driven Document Writer (Letters, Reports, MOMs)

2.3.3 Interoperability

- Import: CAD (DXF), PDFs, images/sketches, IFC, STAAD models (optional)
- Export: PDF, DXF, SVG, IFC
- API integration for:
    - Google Maps
    - Geotechnical data repositories
    - Material cost APIs
    - AI Studio/Antigravity cloud inference

2.4 Modular Billing Structure

Each step is billable separately:

Module	Description	Billing Model
Requirement Assistant	AI chat + data gathering	Free / low cost
Architecture Layout	Suggestions + plans	Per layout generation
Structural Design	Full analysis & code checks	Per building / per 1000 m^2
Detailing Drawings	Rebar + GA + connection details	Per sheet
Estimation	SSR-based cost sheet	Per project
Project Management	Dashboard access	Monthly subscription
Extras	Geotech AI reports, drone analysis	Add-on pricing

This gives **clear business monetization** for every capability.

2.5 Out-of-Scope (for Now)

- Construction tendering automation
- Real-time sensor integration
- BIM level 300–500 automation
- LCA/Carbon footprint assessment (future module)
- MEP design (planned for v2.0)

2.6 Key Principles

- **AI-first**, not manual-input-first
- **User-driven**, not engineer-driven
- **Modular**, not monolithic
- **Scalable**, not localized
- **Code-compliant**, not approximate
- **Explainable outputs** (calculations, assumptions)
- **Flexible interoperability** (import/export)
- **Professional documentation** for approvals

2.7 Constraints

- Kratos must run via Docker
- GPU/cloud compute for large models
- Regional SSR data must be stored locally
- Some AI services depend on paid inference (AI Studio, Antigravity)

PART 3 — SUPPORTED STRUCTURE TYPES & DESIGN CODES
A Unified Multi-Disciplinary Structural Engineering Platform

3.1 Overview

This application is conceived as a **universal structural design ecosystem** capable of handling:

- Residential buildings
- Commercial & industrial buildings
- PEB structures
- RCC framed buildings
- Masonry structures
- Retaining structures
- RE walls, soil structures
- Landfills & waste containment systems
- Steel buildings
- Composite steel–concrete buildings
- Water tanks, STPs, sumps, pits
- Foundations of all types
- Staircases & architectural components
- Bridges (future module)
- Pavements & roads (future module)

The structural **core engine** will be solver-agnostic and support:

- **Kratos Multiphysics** (primary FE engine)
- **FEniCSx** (optional for continuum mechanics)
- **Custom IS code analytical modules**
- **AI-driven geometry extraction and code selection**

The app will tailor the workflow dynamically based on user requirements such as:
"Construction of G+2 building in Hyderabad"
User inputs → AI interprets → App builds structural definition → App proposes templates / asks additional questions → Finalizes model → Runs design → Produces drawings → Produces BOQ/cost → Produces execution plan.

3.2 Supported Structure Categories (Master List)

This list is **complete**, covering all items required for practical Indian engineering consulting workflows and all items we discussed earlier.

A. RCC Structures

A1. RCC Framed Buildings

- G+0 to G+50+ storeys
- Architectural extraction from uploaded plans (PDF/JPEG/hand sketch)
- AI-driven room detection
- Column grid deduction
- Beam connectivity
- Slab type recognition (one-way, two-way, flat-slab, waffle)
- Staircase auto-detection
- Lift core, shear walls
- Foundation selection

A2. RCC Elements

- Columns
- Beams
- Slabs
- Footings (isolated, combined, strip)
- Raft foundation
- Pile foundation (bored/driven)
- Retaining walls
- Water tanks (UG/OG)
- Sumps, pits, STP tanks

A3. Staircases

- Dog-legged
- Open-well
- Spiral
- Monumental
- AI extraction from architectural sketches

B. Steel Structures

B1. PEB Buildings

- Portal frame buildings
- Multi-span frames
- Saw-tooth structures
- Cranes (EOT, gantry)
- Catwalks & mezzanines

B2. Steel Industrial Structures

- Pipe racks
- Conveyors
- Technological structures
- Equipment supporting frames
- Boiler structures (future)

B3. Steel Towers

- Lattice towers
- Telecom towers
- Solar module mounting structures

C. Composite Structures

- Composite beams
- Composite decks
- Composite columns
- Steel-concrete hybrid systems
- Composite floors (deck sheet + concrete)

D. Soil–Structure Systems

D1. Retaining Structures

- Cantilever retaining walls
- Counterfort walls
- Gravity walls

- Sheet pile walls

D2. Reinforced Earth (RE) Walls

- MSE walls with reinforcement strips
- Wrap-around and panel-type facings

D3. Landfills & Waste Containment

- Bunds
- Berms
- Benching
- Slope stability
- Leachate head loading
- Settlement estimates
- Capacity estimation

E. Water & Liquid Containment Structures

- RCC rectangular/sump tanks
- RCC circular tanks
- Wastewater tanks
- Septic tanks
- Treatment modules (future)

F. Foundations (All Types)

- Isolated footings
- Combined footings
- Strap footings
- Rafts
- Pile foundations
- Pile caps (1–20 piles)
- Machine foundations

G. Masonry Structures

- Non-engineered load-bearing buildings
- Compound walls
- Boundary walls
- Masonry piers/returns

H. Architectural Components (AI-driven extraction)

- Room layouts
- Doors/windows
- Levels & steps
- Orientation
- Site plan
- Column grid
- Slab outline
- Staircase geometry
- Parking layout

3.3 Supported Codes & Standards (Global & Indian)

The application will support **multi-code design**, dynamically selectable per project.

A. Indian Standards (IS Codes)
Primary codes (India):
Structural Design

- **IS 456** – RCC design
- **IS 800** – Steel design
- **IS 801 / 811 / 875 (Part 3)** – Light gauge steel
- **IS 875 (Part 1–5)** – Loads
- **IS 1893 (Parts 1–5)** – Seismic
- **IS 13920** – Ductile detailing
- **IS 2911 (Part 1–4)** – Pile design
- **IS 4091** – Precast concrete
- **IS 3370** – Liquid retaining structures

Soil & Foundations

- **IS 6403** – Bearing capacity
- **IS 2950** – Raft
- **IS 2911** – Piles
- **IS 8009** – Gravity & cantilever retaining walls

Wind Loads

- **IS 875 Part 3** – Wind loading (latest edition)

Fire & Stability

- **NBC 2016** – Architectural & fire norms

B. International Codes (Secondary/Optional)
Steel Design

- **AISC 360**
- **EN 1993 (Eurocode 3)**
- **AS4100**
- **AISI (cold formed)**

Concrete Design

- **ACI 318**
- **EN 1992 (Eurocode 2)**

Seismic

- **ASCE 7**
- **EN 1998**

Composite

- **EN 1994**

Foundations

- **FHWA (RE walls)**
- **Eurocode 7**

Tanks & Hydraulic Structures

- **ACI 350**
- **EN 1992-3**

3.4 Solver Engines Supported
The application will support a **hybrid solver pipeline**:

Primary FE Solver

- **Kratos Multiphysics**
  - StructuralMechanicsApplication
  - GeoMechanics
  - FluidStructureInteraction
  - Contact Mechanics

Secondary Solver (optional modules)

- **FEniCSx**
  - For high-fidelity continuum FE
  - Plate, shell, 2D/3D elasticity
  - Slope stability & soil settlement

Analytical Modules

Written in Python for fast checks:

- Footing capacity
- Retaining wall stability
- Shear/torsion checks
- Wind load calculation
- Seismic load distribution
- Simplified frame analysis
- Load combinations

3.5 Deliverables Supported

For **every structure type**, the app will generate:

A. Drawings (Auto-generated)

- Architectural drawings
- Structural framing plan
- Beam layout
- Column layout
- Foundation plan
- Rebar detailing
- Steel detailing
- PEB fabrication drawings
- Misc. stair, sump, tank drawings

B. Analysis Outputs

- Deformed shapes
- Reaction forces
- Bending/shear envelopes
- Drift checks
- Slab stresses
- Soil pressures
- Foundation settlement

C. Design Reports

- Code references
- Hand-calculation style detailing
- Clause-by-clause verification

D. BOM/BOQ Outputs

- Concrete

- Steel
- Masonry
- Formwork
- Earthwork

E. Costing

- SSR-based rates
- Market-rate override
- Contractor margin
- GST
- Scenario-based cost

F. Execution Plan

- Schedule
- QHSE checklist
- Resource plan
- Method statements

3.6 Interoperability & Data I/O

The engine supports:

A. Import

- Hand sketches (AI extraction)
- PDF plans
- AutoCAD (DXF/DWG via API)
- STAAD input files
- ETABS XML
- IFC/BIM

B. Export

- DXF structural drawings
- Kratos input model
- IFC model
- STAAD-compatible .std
- ETABS .edb equivalent
- Full PDF report
- XLSX BOQ

3.7 Dynamic Module License ("Pay-per-Use")

Each design module is a **separate paid feature**, delivered as microservices:

- RCC building design
- PEB design
- Tank design
- Retaining wall design
- Landfill capacity design
- Staircase design
- Foundation design
- Estimation module
- Project management module

User pays only for modules used.

✔ **Part 3 Completed.**

PART 4 — UNIFIED DATA MODEL (UDM): CORE REPRESENTATION (Detailed PRD)
Master Structural + Architectural + Estimation + Workflow + Project Management Data Model

This part defines the **single source of truth** for the entire application—
from **concept → architectural layout → structural model → analysis → design → estimation → execution → closure**.

It is the backbone that makes the app behave like a fusion of:

- **STAAD / ETABS / Robot Structural Analysis / Tekla** (structural modeling + analysis)
- **Revit / AutoCAD Architecture** (architectural layouts + parametric modeling)
- **CPWD/State SSR Estimation Engines**
- **Primavera/MSP-like Workflows**
- **BIM-enabled lifecycle management**

Everything flows through this **UDM**.

4.1 Objectives of the Unified Data Model

The UDM must:

1. **Represent ANY structure type**
    - RCC buildings (G+1 to G+40)
    - Steel buildings / PEB
    - Industrial sheds
    - Retaining walls
    - RE walls
    - Landfills
    - Water tanks
    - Compound walls
    - Foundations
    - Staircases
    - Bridges (future)
    - Roads (future)
2. **Be fully parametric**
    - Editable like STAAD
    - Re-generatable layouts
    - Auto-update drawings & quantities on every change
3. **Enable AI-driven refinement**
    - The user says: *"G+2 building in Hyderabad"*
    - AI expands → UDM → model builder → design → drawings → quantities
4. **Unify structural + architectural + MEP placeholders**
    (Even if MEP is a placeholder for Phase-1)
5. **Allow modular monetization**
    - Architectural only
    - Structural design only
    - Estimation only
    - Execution management only
    - Full package (discounted)
    - Add-on services (peer review, expert assistance, permit set)

4.2 High-Level Structure of the UDM
The model is divided into 10 major layers:

1. **Project Layer**
2. **Site / Location Layer**
3. **Building / Structure Layer**
4. **Architectural Layout Layer**
5. **Structural System Layer**
6. **Materials + Sections Library**
7. **Load Layer**
8. **Analysis + Solver Layer Hook**
9. **Design Layer Hooks**
10. **Estimation + Rate Analysis Layer**

Below is a detailed breakdown.

4.3 PROJECT LAYER (Root Node)

```
Project {
 id: UUID,
 title: string,
 description: string,
 project_type: "RCC Building" | "Steel Building" | "Industrial" | "Landfill" | "Retaining
Wall" | ...,
 location: Location,
 created_by: UserID,
 created_at: timestamp,
 updated_at: timestamp,
 version: number,
 workflow_state: "Initiation" | "Requirements" | "Layout" | "Design" | "Estimation" |
"Execution" | "Closure"
}
```

Behavior

- Everything starts with a **Project**.
- Every module reads/writes inside this root.

4.4 LOCATION LAYER

Captures parameters essential for codes and rates.

```
Location {
 city: string,
 state: string,
 country: string,
 wind_zone: string,
 seismic_zone: string,
 temperature_range: [min, max],
 soil_type: string,
 SBC: number,
 local_SSR_year: string,
 GST: number,
 labor_rates_multiplier: number
```

}

Auto-derived

- **IS 875 wind zone**
- **IS 1893 seismic zone**
- **Local SSR** & inflation factor
- **Soil parameters** (default + editable)

4.5 BUILDING / STRUCTURE LAYER

This is the high-level representation:

(*For G+2 example: floors = 3 including ground*)

```
Structure {
 id: UUID,
 structure_type: "RCC Building" | "Steel Building" | "Landfill" | "Retaining Wall" | "PEB",
 storeys: number,
 storey_heights: number[],
 bays_x: number,
 bays_y: number,
 bay_spacing_x: number[],
 bay_spacing_y: number[],
 architectural_layout: ArchitecturalLayout,
 structural_system: StructuralSystem,
 loads: LoadCase[],
 materials: MaterialLibrary,
 analysis_model: AnalysisModel,
 design_output: DesignResult,
 drawings: DrawingSet,
 quantities: QuantitySet,
 cost_estimate: Estimate,
 workflow: ProjectWorkflow
}
```

4.6 ARCHITECTURAL LAYOUT LAYER

This layer supports **upload**, **AI generation**, or **template-selection**.

4.6.1 Modes of Input

1. **User uploads hand sketch**
2. **User uploads PDF/PNG plan**
3. **User selects from templates**
4. **AI auto-generates layout**
5. **User manually edits using interactive canvas**

4.6.2 Core Representation

```
ArchitecturalLayout {
 footprint_polygon: Polygon,
 rooms: Room[],
 doors: Door[],
 windows: Window[],
 stairs: Stair[],
 parking: Parking[],
```

 utilities: UtilitySpaces[],
 elevation_hints: ElevationHints
}

4.6.3 Rooms Example

Room {
 id: UUID,
 name: "Bedroom" | "Kitchen" | "Hall" | "Office" | "IndustrialBay",
 polygon: Polygon,
 level: number,
 height: number
}

4.6.4 AI Interpretation Layer

The AI module converts hand sketches to:

- Walls
- Rooms
- Doors
- Windows
- Staircase rules
- Beam grid suggestions
- Column spacing suggestions

4.7 STRUCTURAL SYSTEM LAYER

This is where our app becomes **STAAD-like**.

4.7.1 Representation

StructuralSystem {
 grid_x: number[],
 grid_y: number[],
 grid_z: number[],
 elements: Element[],
 nodes: Node[],
 supports: Support[],
 connections: Connection[],
 foundations: Foundation[],
 walls: StructuralWall[],
 slabs: Slab[],
 load_paths: LoadPath[]
}

4.7.2 Nodes

Node {
 id: number,
 x: number, y: number, z: number,
 constraints: DOF[]
}

4.7.3 Elements

Supports multiple types:

Element {
 id: number,

```
 type: "Beam" | "Column" | "Slab" | "Wall" | "Shell" | "Solid",
 node_start: NodeID,
 node_end: NodeID,
 material_id: string,
 section_id: string,
 design_forces: Forces,
 design_results: DesignResults
}
```

4.8 MATERIALS + SECTION LIBRARY

Country-specific + code-specific (IS/AISC/EC).

```
Material {
 id: string,
 name: string,
 type: "Concrete" | "Steel" | "Soil" | "Masonry",
 properties: {
  E: number,
  fy: number,
  fu: number,
  density: number,
  poisson: number,
  damping: number
 }
}
Section {
 id: string,
 type: "ISMB" | "ISHB" | "Tubular" | "Rectangular" | "Plate" | "Custom",
 dimensions: object,
 area: number,
 Ix: number,
 Iy: number,
 Iz: number,
 weight_per_m: number
}
```

4.9 LOAD LAYER

Supports:

- IS 875 Part 1 – Dead
- IS 875 Part 2 – Imposed
- IS 875 Part 3 – Wind
- IS 875 Part 4 – Snow
- IS 1893 – EQ
- Machine loads
- Point/line/pressure
- Load combinations (customizable)

Representation

```
LoadCase {
```

id: string,
type: "DL" | "LL" | "Wind" | "EQ" | "SoilPressure" | "Traffic" | "Thermal",
magnitude: number | object,
direction: string,
application: "Global" | "Member" | "Area"
}

Load combinations:

LoadCombination {
id: string,
factors: { load_case_id: factor }
}

4.10 ANALYSIS MODEL LAYER (Solver Interface)

This layer is solver-agnostic and supports plugging:

- Kratos Multiphysics
- FEniCSx
- OpenSees
- Calculix
- Custom solvers

Representation

AnalysisModel {
solver: "Kratos" | "FEniCSx" | "Opensees" | "Custom",
mesh: Mesh,
boundary_conditions: BC[],
loads: LoadCase[],
results: AnalysisResults
}

4.11 DESIGN LAYER HOOKS

Supports multiple codes:

- **IS 456** (RCC)
- **IS 800** (Steel)
- **IS 801 / IS 8011 / industrial guidelines**
- **ACI / Eurocode / AISC** for alternative checks
- **IS 2911** for foundations
- **IS 3370** for water tanks
- **IS 14458** for retaining walls
- **IRC guidelines** for pavement/hill roads

Representation

DesignResult {
member_designs: MemberDesign[],
foundation_designs: FoundationDesign[],
wall_designs: WallDesign[],
slab_designs: SlabDesign[],
staircase_designs: StaircaseDesign[],
water_tank_design: TankDesign,
landfill_checks: LandfillStabilityResult

```
}
```

Each has:

- Input forces
- Design calculations
- Capacity checks
- Code references
- Failure modes
- Final capacity utilization

4.12 DRAWINGS LAYER

Auto-generated via:

- SVG
- DXF
- IFC (future BIM)
- PDF (with dimensioning)

```
DrawingSet {
 plans: Drawing[],
 elevations: Drawing[],
 sections: Drawing[],
 reinforcement: Drawing[],
 steel_details: Drawing[],
 BOQ: Drawing[],
 schedules: Drawing[]
}
```

4.13 QUANTITIES + COSTING LAYER

SSR-based multi-source estimation.

```
Estimate {
 items: EstimateItem[],
 subtotals: { civil: number, steel: number, concrete: number, formwork: number, etc. },
 total_cost: number,
 gst: number,
 contingencies: number
}
```

4.14 PROJECT WORKFLOW LAYER

Represents the **full lifecycle**:

```
Workflow {
 phases: [
  "Preliminary Requirements",
  "AI Layout Generation",
  "Structural Modeling",
  "Architectural Finalization",
  "Analysis",
  "Design",
  "Drawing Package",
  "Estimation",
```

```
  "Execution Planning",
  "Monitoring + Billing",
  "Project Closeout"
 ],
 current_phase: string,
 progress: number,
 critical_paths: Path[]
}
```

4.15 HOW THE ENTIRE UDM INTERACTS

User:

“G+2 residential building in Hyderabad.”

The app does:

1. Creates Project
2. Sets Location = Hyderabad → wind, seismic, SSR
3. AI asks clarifying questions → updates Structure
4. Architectural layout engine generates OR interprets the sketch → fills ArchitecturalLayout
5. Grid + column + beam generator produces StructuralSystem
6. Loads assigned from LoadLayer
7. Analysis model generated → AnalysisModel
8. Design checks → DesignLayer
9. Drawings auto-generated → DrawingSet
10. Estimation triggered → Estimate
11. Workflow advances → Workflow

4.16 BENEFIT OF THIS MODEL

This UDM ensures:

- **Any structure type fits into same digital ecosystem**
- **AI can read and modify ANY layer**
- **Kratos/FEniCSx can plug directly into FEM layer**
- **Drawings auto-update with any parameter change**
- **Costing is always linked to design changes**
- **User can jump to any module and pay only for that module**

PART 4 Completed.

This is now a **precise, enterprise-grade, multi-module UDM** suitable for:

- AI Studio
- Antigravity
- Docker
- Kratos/FEniCSx
- Future BIM integration (IFC)

DETAILED PRD — PART 4

UNIFIED DATA MODEL (UDM) & STRUCTURAL DIGITAL TWIN CORE**

This section defines the **single source of truth** for every building/structure the system handles.

Everything the system does—AI reasoning, layout generation, design, analysis, estimation, drawings, scheduling—depends on this UDM.

This is the **heart** of the entire app.

4.1 Purpose of the UDM

The **Unified Data Model (UDM)** acts as:

- The **digital twin** of the project
- The master container for all inputs, derived data, analysis artefacts, results, and documents
- A **schema-driven layer** that ensures consistency across all modules:
  - AI Conversational Input Engine
  - Architectural Layout Engine
  - Structural Modeling Engine
  - Analysis Engine (Kratos / FEniCSx / OpenSeesPy future support)
  - Design Engine (IS/ACI/AISC/EC)
  - Estimation Engine (SSR/state SOR, CPWD, AP/TG/KA/MH/DELHI SOR)
  - Drawings Engine (AutoCAD/DXF/SVG)
  - Project Execution Engine (Schedules, BOQ integration)

The UDM ensures **any structure type** can be defined using one backbone.

4.2 Core Concepts

The UDM is composed of **three layers**:

Layer A — User Intent & AI Normalization Layer

This layer captures **natural language inputs** and converts them into structured context.

Example input:

“Construction of G+2 building in Hyderabad”

AI normalizes into:


```
{
 "project_type": "RCC Building",
 "usage": "Residential",
 "location": "Hyderabad",
 "height": "G+2",
 "estimated_floors": 3,
 "climatic_zone": "Seismic Zone II, Wind Zone 39 m/s",
 "required_drawings": ["Architectural", "Structural", "Estimation"],
 "user_preferences": {
   "foundation_type": "As per soil report or pad footing default",
   "floor_height": 3.0,
   "parking_requirement": "Ground floor"
 }
}
```


This becomes the **seed** for layout generation and modeling.

Layer B — Structural Data Schema (Universal Across RCC/PEB/Landfill/RE Wall)

The UDM defines the structure as a hierarchy:

UDM Level 0: Project Metadata

project_id
project_name
location
topography
usage_type
building_category
importance_factor
client_details

UDM Level 1: Geometry Objects

1. Building Envelope

footprint_polygon
storey_heights
number_of_storeys
plinth_level
terrace_level
basement_info
setbacks

2. Functional Spaces

```
rooms: [
 { name, area, function, floor, adjacency }
]
circulation: {
 stairs, lifts, corridors
}
parking: { type, capacity }
```

3. Load Paths

```
load_transfer_paths: {
 slabs → beams → columns → footings
}
```

UDM Level 2: Structural Components

All structural members are stored as **standardized objects**.

Columns

```
{
 id, floor, size, axis, material, grade,
 loads: { axial, moment, shear },
 design_results: {},
 analysis_results: {}
}
```

Beams

```
{
 id,
```

span_coordinates,
section_properties,
support_conditions,
load_cases,
design_results
}
Slabs
{
id,
slab_type: one_way/two_way/flat_slab,
thickness,
reinforcement,
loading
}
Foundations
{
id,
type: isolated/combined/raft/pile,
soil_properties,
bearing_capacity,
uplift_check,
settlement_check
}
PEB Frames
{
frame_id,
column_section,
rafter_section,
haunch_details,
purlin_type,
wind_bracing_scheme
}
Retaining Structures
{
height,
soil,
surcharge,
wall_geometry,
stability_results
}
Landfill Cells
{
polygon,
benches,
slopes,
bunds,
stability_factors,

```
 leachate_details
}
```

UDM Level 3: Analysis Layer

Analysis engine (Kratos/FEniCSx) writes results back in:

```
analysis_cases: [
  {
    "case_name": "DL+LL",
    "nodal_displacements": [...],
    "member_forces": [...],
    "stress_contours": { path_to_vtk },
    "reactions": [...],
    "mode_shapes": [...]
  }
]
```

UDM Level 4: Design Layer

For each code:

```
design_results: {
  IS800: { beam_checks, column_checks, bolt_design },
  IS456: { slabs, beams, columns, footing },
  IS1893: { base_shear, drift, torsion },
  ACI318: { optional },
  AISC360: { optional }
}
```

UDM Level 5: Drawings Layer

```
drawings: {
 architectural: ["floor_plan.svg", "elevation1.svg", "sections.svg"],
 structural: ["beam_layout.dxf", "column_layout.dxf"],
 footing_detail: ["footing_f1.dxf"],
 rebar_schedules: ["bar_bending_schedule.xlsx"]
}
```

Layer 6: Estimation + BOQ

Each object computes its own quantity:

```
boq = [
 { "item": "Concrete M25", "qty": 78.5, "unit": "m3", "rate": 7800, "amount": ... },
 { "item": "Steel Fe500", "qty": 4.2, "unit": "t", "rate": 65000 },
 ...
]
```

SSR rates come from:

- Telangana SSR
- Andhra SSR
- Karnataka SSR
- Maharashtra SSR
- Delhi DSR 2023

- CPWD DSR

4.3 Why This UDM Works for All Structure Types
Because it is:

🔹 **Extensible**

- Add new object types (tanks, chimneys, warehouses) easily.

🔹 **AI-friendly**

- Every field is introspectable by a LLM:
  - "Find all beams failing in bending"
  - "Optimize this structural system using steel vs RCC comparison"

🔹 **Analysis-Engine Agnostic**
The same UDM can feed:

- Kratos
- FEniCSx
- OpenSeesPy (optional)
- STAAD import/export
- ETABS (later)

🔹 **Drawings + Estimation + Schedules All Feed From the Same Data**
No duplication, no mismatches.

4.4 System Flow Using UDM

```
User Input → AI → UDM (Level 0–2)
         ↓
     Layout Generator (Arch)
         ↓
   Structural Model Generator
         ↓
   Kratos/FEniCSx Analysis
         ↓
    Design Engine (IS/ACI/AISC)
         ↓
  Drawings Engine (AutoCAD/SVG/DXF)
         ↓
 Estimation Engine (SSR / CPWD Rates)
         ↓
 Project Execution (Gantt + Cashflow)
```

4.5 API Contracts (for internal modules)
POST /api/udm/create
Creates new digital twin instance.
POST /api/udm/update_geometry
Updates footprint, floors, grids.
POST /api/udm/add_component
Adds beams/columns/slabs/frames.
POST /api/analysis/run
Runs Kratos/FEniCSx analysis and writes results.
POST /api/design/check

Runs code checks and updates design metadata.
GET /api/drawings/export
Exports DXF/SVG.
GET /api/estimation/generate
SSR + BOQ.

4.6 Storage Representation
Primary Storage
- PostgreSQL (highly relational UDM)
- Geometry stored as WKT/geoJSON
- Analysis results stored in S3/GCS (VTK, HDF5)

Local + Cache
- SQLite for small devices
- Redis for session storage

4.7 Multi-Structure Extensions
Because UDM is hierarchical, you can attach any structure type as a subtype:
type: "PEB"
type: "RCC Building"
type: "Industrial Shed"
type: "Landfill"
type: "RE Wall"
type: "Retaining Wall"
type: "Composite Structure"
type: "Bridge"

SUMMARY OF PART 4
This part establishes the **data backbone** for the entire platform.
Every design, drawing, estimation, and analysis step uses this UDM.

PART 5 – Functional Requirements: AI Intake & Requirements Capture
This part defines how the **AI layer** will understand a vague user statement like:
"G+2 residential building in Hyderabad"
and convert it into fully structured requirements that drive **design**, **estimation**, and **project management**, and how it fits with all structural modules (PEB, RCC, landfills, etc.) we discussed earlier.

5.1 Objectives of AI Intake Layer

1. Convert **natural language intent** into **structured project data**.
2. Automatically drive the user through **only the questions that matter** for their chosen journey:
   - Full design (architecture + structural + detail drawings + BoQ + project plan), or
   - Estimation-only, or
   - Execution/project management only.
3. Support **multi-structure** requirements:
   - G+2 RCC building + compound wall + septic tank
   - Industrial shed/PEB + office block + retaining wall + landfill platform, etc.
4. Keep the platform **modular for billing**:
   - Each step (concept plan, architectural, structural, detailed, estimation, execution) is a separate billable module.

5.2 High-Level AI Workflow

1. **User entry message**
   - Example: "I want to construct a G+2 residential building in Hyderabad on a 40x60 site."
2. **Intent classification** by AI:
   - Project type: Residential / Commercial / Industrial / Mixed-use.
   - Structure family: RCC Building / PEB / Composite / Landfill / RE Wall / etc.
   - Desired path:
     - (a) Design → Estimate → Execution
     - (b) Design only
     - (c) Estimate only (user already has drawings)
     - (d) Execution + monitoring only (user already has DPR & BoQ)
3. **Guided question tree** (AI-driven):
   - AI progressively asks only the relevant questions:
     - Location, dimensions, floors, occupancy, soil info, budget, codes, etc.
4. **Input enrichment**:
   - AI converts free text, uploaded sketches, or plans into:
     - Site boundary,
     - Floor-wise usage,
     - Structural system assumptions,
     - Modules required (retaining wall? water tank? overhead tank? PEB shed?).
5. **Requirements freeze & package selection**:
   - AI proposes a "solution pack":

        - e.g., “Architectural floor plans + 3D views + structural design (RCC) + BoQ + SSR-based estimate + Gantt schedule.”
    - User can accept/edit modules → **this drives billing**.
6. **Trigger downstream engines**:
    - Architectural generator
    - Structural generator (Kratos/other engine under the hood)
    - Estimation engine
    - Project execution engine

5.3 AI Intake – Detailed Functional Requirements
5.3.1 Input Types Supported

- **Text prompts**:
    - “G+2 residential in Hyderabad.”
    - “Industrial warehouse 60x120 m PEB shed in Pune.”
- **Parametric forms**:
    - Predefined dropdowns for users who don’t want chat:
        - Building type, floors, area, location, soil type, etc.
- **Uploads**:
    - Image/PDF of hand sketch.
    - DWG/DXF or simple plan.
    - Excel/CSV with room schedule or functional requirements.

AI responsibilities:

- Parse all inputs.
- Extract structured data.
- Detect missing critical info and ask focused follow-ups.

5.3.2 Requirement Graph (Conceptual)
The AI will maintain an internal **Project Requirement Graph**:

- **Project node**:
    - Project ID, Title, Location, Client type, Stage.
- **Site node**:
    - Plot boundary, setbacks, road(s), zoning constraints.
- **Built space nodes**:
    - Blocks (A/B/C), each with:
        - Usage (residential, office, industrial, etc.)
        - Floors & floor heights
        - Occupancy loads
- **Structure nodes**:
    - RCC building, PEB, composite frame, retaining walls, RE walls, tanks, landfills, etc.
- **Functionality nodes**:
    - Staircases, lifts, water tanks, OHT, compound walls, UG tanks, ramps, basements, podiums.
- **Service nodes** (optional future phase):
    - MEP, fire-fighting, HVAC, plumbing.

AI converts free text into this graph and keeps it updated as user answers more questions.

5.3.3 Question Flow Examples

Example 1 – G+2 Residential in Hyderabad

1. AI: "You want a **G+2 residential building in Hyderabad**.
   Is it for **single-family**, **multi-family (apartments)**, or **mixed-use** (shops + residence)?"
2. User: "Single-family."
3. AI follows the **single-family template**:
   - Site size (length x width)?
   - Corner plot or single road?
   - Approximate budget?
   - Soil type known? (Rock/medium/soft)
   - Parking requirements (ground floor parking or stilt)?
   - Number of bedrooms per floor, special spaces (home office, pooja, etc.)?
4. AI builds a structured requirement set and offers:
   - "I can generate concept floor plans, structural concept, and a preliminary cost estimate.
     Do you want all three, or just any specific ones?"

User selection drives which modules to activate.

Example 2 – Industrial PEB + Retaining Wall + Landfill

1. User: "Need a 60x120m PEB warehouse with G+1 office and a landfill platform with retaining walls."
2. AI:
   - Splits into **sub-projects under one parent**:
     - Sub-project 1: PEB warehouse (60 x 120 m).
     - Sub-project 2: RCC office block (G+1).
     - Sub-project 3: Retaining walls.
     - Sub-project 4: Landfill platform (CPHEEO/CPCB).
   - Asks specific questions per structure type.
3. User chooses:
   - "Design + estimate for PEB and office. Only design (no estimate) for landfill."

System activates only those design + estimation modules.

5.3.4 Module Selection & Billing Logic

- Each logical block is a **module**:
  1. Requirement capture & conceptual proposal (free or minimal flat fee).
  2. Architectural layout generation.
  3. Structural design:
     - RCC module
     - Steel/PEB module
     - Composite module
     - Special modules (retaining wall, RE wall, landfill, tank, chimney).
  4. Detailed drawings (GA, reinforcement, connection details, etc.).
  5. Estimation/BoQ:
     - SSR-based or custom rate library.

6. Execution management:
    - Schedule, site logs, quality, billing, etc.

- AI suggests a default pack based on use case; user can add/remove modules.
- Pricing model:
    - Per module per project OR subscription + per-project credits.
- App must:
    - Track active modules per project.
    - Enforce feature access based on purchased modules.

5.3.5 "Jump Directly to Estimation" Path

Requirement:
User may say, "I already have drawings; I just want an estimate."
AI behavior:

1. Detects **estimation-only intent**.
2. Asks:
    - Do you have **BoQ**?
        - If yes, import and map to SSR.
        - If no, ask for drawings (PDF/DWG) or room-wise data.
3. Required inputs:
    - Location, SSR/state, year.
    - Drawing set or basic quantities.
4. Skip all design modules; directly enter **Estimation Engine**.
5. Option: AI offers optional upsell:
    - "Based on your drawings, we can also perform independent **design check** for an additional fee."

5.4 AI Orchestration Within the System
5.4.1 Integration Points

- **AI Layer** sits above:
    - Design engine(s) (which may internally use Kratos/FEniCSx etc.).
    - Estimation engine.
    - Execution engine.
- It should:
    - Maintain **conversation state** (project context).
    - Update the **Project Requirement Graph**.
    - Call backend APIs to:
        - Create/update Project.
        - Create/update Sub-structures.
        - Trigger design runs.
        - Trigger estimation runs.

5.4.2 Memory & Multi-session Handling

- Project ID is the anchor.
- AI should remember:
    - User preferences (state SSR, typical floor plans, preferred structural system).
    - Previously uploaded drawings.
- For returning users:

    - “Do you want to continue with **Project P-2025-0012** (G+2 Hyderabad) or start a new one?”

5.5 Non-Functional Requirements for AI Intake

- **Explainability**:
    - AI must show a **summary of captured requirements** before triggering design:
        - “Here is how I understood your project…” with an editable checklist.
- **Validation**:
    - Hard constraints:
        - No negative dimensions.
        - Check FSI/zoning issues where simple rules are known (optional advanced feature).
- **Performance**:
    - First response < 3 seconds where possible.
    - Full requirement capture usually 3–10 exchanges.
- **Language Support** (future but design now):
    - English + regional languages (Telugu, Hindi, etc.).
- **Logging**:
    - Store all critical prompts & outputs for QA and audit (important for engineering validation).

5.6 Deliverables for This Part

- AI requirement capture flows for:
    - G+2 RCC residential in Indian city.
    - Industrial PEB + office.
    - Building + retaining walls + landfill.
- JSON schema for **Project Requirement Graph**.
- API specs:
    - POST /ai/intake/session/start
    - POST /ai/intake/step
    - GET /project/{id}/requirements
    - POST /project/{id}/modules/select

PART 6 — MODEL ENTITIES & RELATIONSHIPS (MER)
Master PRD for the AI-Driven Structural Design, Estimation & Construction Management Platform

---

6.1. PURPOSE OF THIS SECTION

This section defines:
a) ALL fundamental engineering entities
(beams, columns, slabs, foundations, walls, frames, nodes, loads, materials, etc.)
b) ALL architectural/interior entities
(rooms, walls, openings, furniture zones, shafts, circulation, etc.)
c) Non-structural but essential project elements
(floors, zones, grids, levels, utility networks, workflows, tasks, cost objects)
d) Their relationships
(how beams attach to nodes, how slabs sit on beams, how walls attach to slabs, how spaces relate to floors, how drawings map to entities)
e) Shared data definitions & rules
(ID formats, units, metadata, revisioning system, inheritance, validation rules)
This MER becomes the **Universal Data Model (UDM)** that every module can rely on.

6.2. UNIVERSAL OBJECT MODEL (UOM)

The UOM is a **multi-hierarchical, graph-based object model** designed to support:

- AI-driven design
- Parametric modeling
- FEM simulation
- Architectural + structural collaboration
- Quantity takeoff
- Estimation
- Scheduling
- Approvals
- Revisions
- Exports (DWG, IFC, DXF, STAAD, Kratos, etc.)

6.3. TOP-LEVEL CATEGORIES OF ENTITIES

The entire platform has **5 primary classes** of entities:

6.3.1. Project Domain Entities

Entity	Purpose
Project	Top-level container for all operations
User / Roles / Permissions	Organizations, teams, designers, contractors
Site	GIS location, climate, wind, seismic zone, soil data
Building / Facility	A project may have multiple buildings
Phase/Package	Excavation, structure, MEP, finishing

6.3.2. Architectural Domain Entities

These support planning, layouts, space programming.

Entity	Description
Floor / Level	Each storey with elevation
Grid System	A-B-C / 1-2-3 grid lines
Space / Room	Bedroom, kitchen, toilets, corridors
Architectural Wall	Partition, load-bearing, non-load bearing
Opening	Doors, windows, ventilators
Staircase / Ramps	Geometric parameters
Shafts	Plumbing, electrical, AC shaft
Roof / Terrace	Sloped or flat roof
Circulation Zones	Lift lobby, corridors

6.3.3. Structural Domain Entities

These are **core for analysis, design, and detailing**.

Primary Structural Entities

Entity	Meaning
Node	FEM node with coordinates
Element	Frame/beam/column/plate/solid
Beam element	1D element
Column element	1D vertical element
Slab element	2D element
Wall element	2D vertical element
Footing / Foundation	Isolated, combined, strip, raft
Pile / Pile Cap	Deep foundation entities
Frame	A collection of connected beams/columns
Truss/Roof Frame	PEB rafters, purlins, bracings

Secondary Structural Entities

Entity	Description
Bracing	X-bracing, K-bracing, portal brace
Tie Beams	Plinth beams, grade beams
Lintels	Over openings
Retaining Wall	Cantilever / Counterfort
RE Wall	Reinforced earth wall systems
Landfill Bund / Berm	Geometric entities
Slope / Embankment	For retaining or landfill

6.3.4. Material and Section Entities

Materials

- Concrete (grades M10–M80)
- Steel (Fe415/500/550/600)
- Structural Sections (ISMB, ISMC, ISWB, PEB tapered)
- Soil (various unit weights, friction angles)

- Masonry (brick, block)

Section Properties

- Standard Rolled Sections
- Custom PEB Sections (tapered)
- RC Sections (rectangular, circular, T-beam)
- Plate thicknesses, shell properties

6.3.5. Loading Entities

Load Type	Description
Dead Load	Self weight, wall loads, floor loads
Live Load	Imposed loads (IS 875 Part 2)
Wind Load	IS 875 Part 3
Seismic Load	IS 1893
Snow Load	IS 875 Part 4
Temperature Load	Thermal effects
Settlement Load	Foundation settlement
Construction Load	Temporary loads
Earth Pressure	Retaining wall, RE wall
Water Pressure	Tanks, underground structures

Each load has:

- **Load Case**
- **Load Combination**
- **Design Scenario**
- **Applicable entity IDs**

6.4. RELATIONSHIP MODEL (GRAPH + HIERARCHICAL)

A modern structural system is neither purely hierarchical nor purely relational. We use a **hybrid graph-relational MER**.

6.4.1. Hierarchical relationships (Tree)

```
Project
 └── Sites
   └── Buildings
     └── Floors
       └── Spaces
         └── Architectural walls
```

6.4.2. Graph relationships (Network)

Node <--> Element

Beam <--> Column <--> Slab <--> Wall <--> Footing

Frames <--> Load Cases <--> Materials

Every element is linked by:

- **Spatial link** = which nodes it connects to
- **Support link** = which support/foundation it rests on
- **Load link** = which load/combination is applied

- **Material link** = properties
- **Design link** = the design module used

6.5. ENTITY DEFINITIONS (DETAILED)

Below are the **canonical definitions**.

6.5.1. Node Entity

Field	Description
NodeID	Unique
Coordinates	(x,y,z)
Boundary Conditions	UX,UY,UZ,RX,RY,RZ
Loads Applied	Node load list
Connected Elements	Beam/column/slab IDs

6.5.2. Beam Entity

Field	Description
BeamID	Unique
NodeStart, NodeEnd	Connects two nodes
SectionID	Cross-section
MaterialID	Material properties
Releases	MY/MZ/FX etc.
Functions	Stiffness, weight, design forces

Design Links

- IS 800 (steel)
- AISC/EC3 (optional)
- IS 456 (RC beams)

6.5.3. Column Entity

Same as beams with additional parameters:

- Slenderness
- Effective length factors
- Foundation link
- Axial force dominant ratios
- Column continuity up floors

6.5.4. Slab Entity

Field	Description
SlabID	unique
Type	One-way, two-way, plate
Thickness	mm
MeshType	Quad/Tri
Support Regions	Beam/Wall edges
Load Distribution	Equivalent UDL

6.5.5. Wall Entity

Structural wall (RC) or masonry:

- Thickness
- Height
- Axial load path
- Stiffness matrix
- Interaction with slabs
- Shear wall coupling

6.5.6. Foundation Entity

Type	Data
Isolated Footing	P, Mx, My, SBC, size
Combined Footing	Pressure diagram, eccentricity
Strip Footing	Continuous beam analysis
Raft	FEM mesh, punching checks
Pile	Capacity, group efficiency
Pile Cap	Force distribution

6.5.7. Retaining Wall Entity

- Backfill soil
- Surcharge
- Drainage
- Stability checks (IS 456 / BS / ACI)
- FEM plane-strain model parameters

6.5.8. Landfill/Bund Entity

- Slope geometry
- Berms
- Bench levels
- Waste density
- Stability regions
- Settlement simulation

6.6. RELATIONSHIPS SUMMARY

Example 1: Beam–Column–Slab Interaction

Beam → Node → Column → Node → Slab → Beam → Load → Material

Example 2: Building–Floor–Space Relationship

Building → Floor → Space → Walls → Slab → Beam → Column

Example 3: Load Combinations

LoadCase -> LoadCombination -> ElementGroup -> DesignCheck

Example 4: Estimation Link

Element -> Material -> Quantity -> Rate -> Cost -> BOQ Item

6.7. WHY THIS MODEL WORKS FOR YOUR APP

This MER is designed to support your entire business model:

🧠 **AI can interpret user inputs**
(e.g., "G+2 building in Hyderabad")
→ Extract building type, floors, seismic zone, wind zone, soil.
📐 **Architecture module**
→ Generates spatial layout using Space, Wall, Opening entities.
🏗 **Structural module**
→ Converts architectural walls/slabs/rooms → nodes/elements.
⚙ **Analysis module**
→ FEM using Kratos/FEniCSx.
🧮 **Design module**
→ Code check using IS/AISC/EC/ACI.
💸 **Estimation module**
→ SSR + quantity extraction from elements.
📄 **Drawing module**
→ Outputs GA, sections, reinforcement, schedules.
📊 **Project execution module**
→ Converts elements → activities → costs → schedule.

PRD – PART 6

MODEL ENTITIES & RELATIONSHIPS (UNIFIED STRUCTURAL DATA MODEL)**

(Foundation for AI-driven design, FEA/kratos workflows, estimation, drawings, and project execution)

6.1 PURPOSE OF THIS SECTION

This part defines the **Unified Structural Data Model (USDM)**:

- A universal digital representation capable of describing **any structure** from a simple retaining wall to a full G+2 building or a 200-acre landfill.
- Supports **AI-driven workflows**, **FEA/kratos**, **drawing engines**, **cost estimation**, and **project management**.
- Ensures all modules speak a **single consistent data language**.

This USDM is the **digital brain** of your entire platform.

6.2 TOP-LEVEL CONCEPT: PROJECT → MODELS → ENTITIES → PARAMETERS → OUTPUT

Every user workflow (AI input, drawings, design, execution) eventually maps to:

```
Project
└── StructuralModel(s)
     ├── GeometryEntities
     ├── Materials
     ├── Sections / Profiles
     ├── Loads & Load Combinations
     ├── AnalysisModel (Kratos-ready)
     ├── DesignResults
     ├── Drawings / BIM
     ├── Quantity & Cost Models
```

6.3 CORE ENTITY HIERARCHY

6.3.1 PROJECT

Represents the entire job.

Attributes:

- ProjectID
- ProjectName
- Description
- SiteLocation (lat, long)
- RegionCode (used for wind, seismic, SSR rates)
- BuildingType (Residential, Industrial Shed, PEB, RCC, Composite)
- WorkflowPath (AI-selected roadmap: Concept → Layout → Design → Estimation → Execution)
- Currency & UnitSystem
- AI Chat History (context)

Relationships:

- Has many → **StructuralModels**
- Has many → **DesignModulesPurchased**
- Has many → **CostingItems**
- Has many → **ExecutionPlans**

6.3.2 STRUCTURAL MODEL

Every structure becomes a model. A project may have multiple models (main building, retaining wall, water tank, etc.)

Attributes:

- ModelID
- ModelType (PEB, RCC Building, Slab, Footing, Staircase, RE Wall, Landfill, etc.)
- DesignStage (Concept, Preliminary, Tender, IFC)
- GlobalOrigin
- StoreyLevels
- StructuralSystemType (Frame, Load-bearing, Portal frame, Shell, Solid continuum)

Relationships:

- Has many → **Geometry Entities** (nodes, members, surfaces, solids)
- Has many → **Materials**
- Has many → **Sections/Profiles**
- Has many → **Loads**
- Has many → **Load Combinations**
- Has one → **AnalysisModel**
- Has one → **DesignResults**
- Has one → **DrawingPackage**
- Has one → **BOQ & Estimation Model**

6.4 GEOMETRY ENTITIES

Kratos & CAD/BIM both depend on clear geometry definitions.

6.4.1 NODE

Attributes:

- NodeID
- X, Y, Z (global coordinates)

- Constraints (UX, UY, UZ, RX, RY, RZ)
- NodeType (Grid node, boundary node, support node, reference point)

Relationships:

- Belongs to → Structural Model
- Used by → Members, Surfaces, Solids

6.4.2 MEMBER (1D ELEMENT)

Used for:

- Beams
- Columns
- Rafters
- Joists
- Purlins
- Bracings
- Truss members
- RE wall strips
- Landfill drains (as line utilities)

Attributes:

- MemberID
- StartNode
- EndNode
- Length
- SectionID
- MaterialID
- MemberType (Beam, Column, Brace, Edge beam, Tie beam, Lintel, Plinth beam)

Analysis Properties (Kratos):

- ElementType (Beam3D, Truss3D)
- Local axis definition
- Release conditions
- Eccentricity
- Imperfection settings

6.4.3 SURFACE (2D ELEMENT)

Used for:

- Slabs
- Wall panels (shear wall, RCC wall)
- Retaining wall stem
- Landfill slope panels
- Steel cladding
- Floor plates

Attributes:

- SurfaceID
- BoundaryNodes
- Thickness
- MaterialID
- SurfaceType (Plate, Shell, Membrane)

Analysis Properties:

- ElementType (ShellMITC4, ShellDKGQ, SolidFace, etc.)

6.4.4 SOLID (3D ELEMENT)

Used for:

- Footings
- Landfill soil mass
- RE wall backfill
- Tanks
- Thick walls
- Pile-soil interaction blocks

Attributes:

- SolidID
- MeshRefinement
- MaterialID
- SolidType (Continuum, Volume mesh)

Kratos Properties:

- Volume mesh
- Constitutive laws
- Boundary conditions

6.5 MATERIAL DEFINITIONS

Supports all building types.

6.5.1 STEEL MATERIAL

- Grade (E250, E350, EN S355, AISC A992, etc.)
- Fy
- Fu
- E
- Density
- CodeSource (IS 800 / EC3 / AISC)

6.5.2 CONCRETE MATERIAL

- Grade (M20–M80)
- fck
- Ec
- Poisson ratio
- Density
- Creep / Shrinkage model
- CodeSource (IS 456 / ACI 318 / EC2)

6.5.3 SOIL MATERIAL

- γ (Bulk density)
- ϕ, c values
- Modulus Es
- Permeability
- Drained/Undrained
- Used for footings, retaining walls, RE walls, landfills.

6.5.4 LANDFILL MATERIAL

- Waste density
- Cohesion
- Angle of internal friction
- Settlement parameters
- Leachate unit weight

6.6 SECTION / PROFILE DEFINITIONS

Supports:

6.6.1 Steel Sections

- ISMB, ISHB
- Built-up tapered profiles
- Cold-formed C/Z
- Tubes, angles, channels

6.6.2 RCC Sections

- Rectangular beam/column
- T-beam
- Slab sections
- Footing blocks

6.6.3 Composite Sections

- Deck sheet + slab
- Composite beam profiles

6.7 LOAD DEFINITIONS

6.7.1 LOAD CASE

- LoadCaseID
- LoadType (Dead, Live, Wind, Seismic, Snow, Temperature, Water Pressure, Earth pressure)
- Standard (IS 875, IS 1893, IS 456, IRC)
- AppliedOn (Nodes / Members / Surfaces / Solids / Global)

6.7.2 LOAD TYPES

- Point loads
- UDL
- UVL
- Area loads
- Pressure loads
- Temperature loads
- Soil pressure distribution
- Surcharge
- Hydrostatic/hydrodynamic loads

6.8 LOAD COMBINATIONS

Automatically generates load combinations per:

- IS 875
- IS 456
- IS 800
- IS 1893

- AISC/EC/AASHTO (if user selects)

Each combination stores:

- CombinationID
- Equation
- Factors
- Applicable Limit State (ULS/SLS)

6.9 ANALYSIS MODEL (KRATOS-READY)

This is the **FE preprocessing engine**.

Attributes:

- Mesh (Nodes, Elements)
- ElementTypes
- Boundary Conditions
- KratosSolverType
- Time/Step settings
- Convergence criteria
- SolverSettings
- Eigenvalue request (for seismic mode shapes)

6.10 DESIGN RESULTS

Stored after analysis:

- Forces & Moments
- Deflections
- Stresses
- Code checks:
  - Steel (IS 800/AISC/EC3)
  - RCC (IS 456/ACI/EC2)
  - Seismic (IS 1893)
  - Footings (IS 2950)
  - Walls (IS 14458/IS 456)
  - Landfill slopes (Stability factors)

Each check has:

- Governing Load Combination
- Governing Element
- Utilization Ratio
- Pass/Fail

6.11 DRAWING ENTITIES

Used by the Drawing Generator (Part 10).

Includes:

- Grid System
- Floor Plans
- Sections
- Elevations
- Reinforcement shapes
- Steel fabrication drawings
- PEB GA drawings

- Landfill benches & bunds
- Dimension styles
- Title blocks

6.12 COSTING MODEL ENTITIES

Each structural model outputs:

- Material quantities (steel, concrete, formwork, blocks, soil)
- SSR mapping
- Rate analysis
- Labour, machinery, overheads
- Final BOQ sheet

6.13 RELATIONSHIP DIAGRAM SUMMARY

A simplified text diagram:

```
Project
 └── StructuralModels
      ├── GeometryEntities
      │     ├── Nodes
      │     ├── Members
      │     ├── Surfaces
      │     └── Solids
      ├── Materials
      ├── Sections
      ├── Loads & LoadCombinations
      ├── AnalysisModel (Kratos)
      ├── DesignResults
      ├── Drawings
      └── CostModel (BOQ)
```

6.14 WHY THIS MODEL IS POWERFUL

This unified model supports:

- AI-driven conversational modeling
- Kratos Multiphysics simulation
- Auto-generation of architectural/structural drawings
- Multi-code design checks
- BOQ, estimation, SSR cost calculation
- Project execution and scheduling
- Modular service billing (user pays only for what they use)
- Interoperability with STAAD → IFC → Revit → DXF

This is the foundation of your full platform.

PART 7 — LOAD ENGINE ARCHITECTURE
A Unified, Code-Compliant, Extensible, Multi-Structure Loading System

---

7.1 Purpose of the Load Engine

The Load Engine is the system that:

1. **Collects user inputs & derived configuration**
   (Building type, location, geometry, purpose, occupancy, soil data, roof type, layout.)
2. **Generates all necessary load cases & combinations** as per relevant codes:
   - **IS 875 (Parts 1–5)** for Indian projects
   - **IS 1893** for seismic
   - **IRC**, **AASHTO**, **Eurocode**, **UBC**, **IBC**, **ASCE**, etc. for global users
   - Special modules for:
     - Landfills (unit weight, surcharge, leachate, wind over exposed dump)
     - Retaining walls (earth pressure, hydrodynamic, live surcharge)
     - RE walls (pullout forces, connection loads)
     - PEB (wind loads on frames, cladding pressures, suction zones)
     - Industrial buildings (crane loads, hanging loads, mezzanine)
     - Water tanks (hydrostatic + hydrodynamic)
     - Solar structures (panel loads + wind + ballast)
3. **Converts loads into analysis-ready formats**
   For whichever solver is selected:
   - **Kratos Multiphysics** *(preferred per your latest instruction)*
   - FEniCSx (optional future backend)
   - OpenSeesPy (legacy fallback)
4. **Stores loads** in a unified data structure (UDM-Load).

7.2 Architectural Philosophy

The Load Engine must satisfy three fundamental principles:

(1) Universality

Should support **any structure**:

- RCC building (G+2, multi-storey)
- PEB / steel warehouses
- Industrial shed with cranes
- Landfills
- Retaining walls
- RE walls
- Tanks
- Bridges (future module)
- Compound walls
- Telecom towers
- Solar plants

(2) Auto-generation based on location

The User inputs simply:
“G+2 RCC building in Hyderabad”
The system automatically pulls:

- **Wind zone**, **basic wind speed**
- **Seismic zone**, **soil type**, **importance factor**
- **Snow**, if applicable
- **Terrain category**
- **Altitude correction**
- **Cladding pressure zones**
- **Dead load library** (SSR-linked densities)

(3) Solver-agnostic, but fully compatible

Loads generated are **converted** into:

- Kratos Loads.json
- UF formulation inputs
- Streamlit visualization
- Internal UDM

7.3 Inputs to the Load Engine

7.3.1 Mandatory location inputs

Parameter	Description
Country	India, GCC, Europe, USA, etc.
State/City	To fetch wind/seismic zones
Soil Type	Rock, Medium, Soft
Elevation	For wind corrections
Land use	Terrain category

7.3.2 Structural system inputs

Category	Examples
Building type	RCC, PEB, Steel frame
Roof type	Sheeted, concrete, curved, truss
Wall type	Masonry, sheeting, RCC
Storeys	G+2, G+5 etc.
Building usage	Residential, Industrial, Warehouse

7.3.3 Geometry inputs

- Plan dimensions
- Heights
- Bay spacings
- Column layout
- Staircase position
- Load-bearing vs RCC framed

7.3.4 Material-based auto-loads

- RCC density
- Steel density
- Brickwork
- Flooring + finishes
- False ceiling + services

7.4 Load Types Supported

7.4.1 Dead Loads (DL)
DL = Material density × volume
Auto-generation from SSR / CPWD / predefined material library.
Examples:

- RCC slab: 25 kN/m^3
- Brick wall: 20 kN/m^3
- Floor finishes: 1.5 kN/m^2
- Roofing sheet: 0.15–0.20 kN/m^2

7.4.2 Imposed Loads (LL)
Per **IS 875 Part 2** or other country standards

- Residential floors: 2.0–3.0 kN/m^2
- Corridors: 4.0 kN/m^2
- Industrial floors: 5–20 kN/m^2
- Mezzanine dynamic factors
- Crane surge, crane vertical, crane tractive loads

7.4.3 Wind Loads (WL)
Per **IS 875 Part 3 (2015)**
Wind pressure generation:

- Basic wind speed
- k1, k2, k3, k4 factors
- Terrain category
- Height variation
- Cladding pressure map
- PEB wind suction zones (A, B, C, D, E, F)

Outputs include:

- Surface pressures
- Frame reactions
- Equivalent nodal loads
- Windward/leeward pressure
- Internal pressure

7.4.4 Seismic Loads (EL)
Per **IS 1893 (Part 1) 2016**
Inputs:

- Zone factor
- Soil type
- Importance factor
- Response reduction factor
- Damping
- Modal combination (SRSS/CQC)

Outputs:

- Storey shear
- Storey drift
- Modal participation
- Equivalent lateral loads
- Base shear
- Load distribution ratio

7.4.5 Special Loads

PEB

- Crane vertical, surge, tractive
- Collateral loads
- Panel dead load
- Solar panel load
- Mezzanine load

Landfills

- Waste density
- Surcharge
- Leachate forces
- Wind load on dump
- Settlement pressures

Retaining Walls / RE Walls

- Rankine / Coulomb earth pressure
- Surcharge
- Water pressure
- Earthquake pressure (Mononobe-Okabe)

Tanks

- Hydrostatic
- Hydrodynamic
- Sloshing

7.5 Load Combinations

Generated automatically based on:

- IS 875
- IS 800 (steel)
- IS 456 (RCC)
- IS 1893 (seismic)
- ACI/Eurocode where needed

Examples (Indian standard):

- 1.5(DL + LL)
- 1.5(DL + WL)
- 1.2(DL + LL + WL)
- 0.9DL ± 1.5WL
- 1.2DL ± 1.2EL
- 1.5(DL + EQ)

7.6 Output of Load Engine

Output is stored as a **unified JSON payload**:


```
{
 "load_cases": [
  {"id": "DL", "description": "Dead Load", "values": [...]},
  {"id": "LL", "description": "Live Load", "values": [...]},
  {"id": "WL_X", "direction": "X", "pressures": [...]},
  {"id": "EQ_X", "base_shear": 265.4, "storey_forces": [...]}
```

],
"load_combinations": [
{"id": "COMB1", "formula": "1.5*(DL+LL)"},
{"id": "COMB2", "formula": "1.5*(DL+WL_X)"}
]
}
Simultaneously, the engine creates solver-compatible files:

For Kratos (preferred)

- ProjectParameters.json
- Loads.json
- NodalLoads.json
- BoundaryConditions.json

For UI (Streamlit/Antigravity)

- Load summaries
- Pressure plots
- Storey shear charts
- Wind pressure diagrams

7.7 Extensibility Architecture

The load engine is built using a **modular adapter pattern**:

Base class

```
BaseLoadModel
 ├── compute_dead_load()
 ├── compute_live_load()
 ├── compute_wind_load()
 ├── compute_seismic_load()
 ├── compute_special_loads()
 ├── generate_load_combinations()
 └── export_to_solver()
```

Derived classes

RCCBuildingLoadModel(BaseLoadModel)
PEBLoadModel(BaseLoadModel)
LandfillLoadModel(BaseLoadModel)
RetainingWallLoadModel(BaseLoadModel)
REWallLoadModel(BaseLoadModel)
TankLoadModel(BaseLoadModel)

Each derived class overrides only what it needs.

7.8 Real-Time Intelligence

The engine intelligently adapts loads using AI:

1. **Predicts which loads apply**
   Based on the structure type, occupancy, country, and behavior.
2. **Suggests missing data**
   Example:

"You selected G+2 building. Please specify soil type to compute seismic loads."

3. **Offers load optimization**
   - Suggests lighter sections

    - Suggests mezzanine load reductions
    - Suggests alternative roof profiles

7.9 Integration with the Workflow

Once loads are generated:

1. Geometry →
2. **Load Engine** →
3. Analysis (Kratos) →
4. Design →
5. Drawings →
6. Estimation →
7. Project Management

Users may:

- Use load generator only
- Skip design
- Go directly to estimation

This supports your **modular pricing strategy**.

7.10 Summary of Part 7

The Load Engine is:

- Fully automated
- Code-compliant
- Multi-structure
- AI-assisted
- Solver-agnostic
- Highly extensible

It forms the **core of the engineering brain** that drives all downstream modules.

PART 7 — LOAD ENGINE ARCHITECTURE (DETAILED PRD)

A Unified, Multi-Code, AI-Assisted Structural Loading Framework

---

7.1 Overview & Purpose

The **Load Engine** is the *central logic module* responsible for generating all code-compliant loads required for structural analysis and design.

The engine supports:

- **Indian Codes (Primary)**
    - **IS 875 Parts 1–5** (Dead, Live, Wind, Snow, Special Loads)
    - **IS 1893** (Seismic)
    - **IS 456, IS 800, IS 1343** dependencies
- **Global Codes (Optional/Selective Based on Need)**
    - **ASCE 7 / IBC**
    - **Eurocode EN 1991, EN 1998**
    - **AISC / ACI** provisions
    - **Australian/New Zealand (AS/NZS 1170)**
    - **British Codes (Legacy BS8110/BS5950)**

The Load Engine must always produce:

- **Raw loads**
- **Load maps**

- **Load cases**
- **Load combinations**
- **Engineering diagrams & validation tables**
- **Structured JSON** for the Analysis Engine

7.2 Key Features & Capabilities
Feature Set (Mandatory)

1. **AI-based understanding of building purpose and location**
2. **Automatic load classification**
3. **Automatic code selection (IS 875, IS 1893, etc.)**
4. **Calculation of all load types:**
    - Dead loads
    - Live/Imposed loads
    - Wind loads
    - Seismic loads
    - Temperature loads
    - Snow loads
    - Hydrostatic & earth pressure
    - Machinery & Dynamic loads
5. **Load distribution engine (1D/2D/3D)**
6. **Auto generation of load combinations**
7. **Outputs consumable by multiple solvers (Kratos, FEniCSx, OpenSeesPy, Staad)**
8. **High-fidelity load visualization**

7.3 Load Input Pipeline
The engine accepts *three types* of load inputs:
1. Direct User Inputs
From natural language, forms, sketches, and BIM models.
Example:
"G+2 residential building in Hyderabad with parking on GF."
2. Code-Driven Inputs
Based on:

- City
- Soil type
- Importance factor
- Building use

3. AI-Derived Inputs
AI interprets missing information:

- Material densities
- Live load categories
- Hazard maps
- Occupancy classification

7.4 Dead Load Engine (IS 875 Part 1)
Purpose
Compute self-weight and permanent loads based on materials & thickness.

Key Functions

- Auto-detection of:
    - Wall densities
    - Slab thickness
    - Flooring type
    - Roof type (sheeting / RCC / deck slab)
- Beam/column self-weight calculation
- Soil backfill loads (retaining walls)
- Landfill waste density profiling

Data Sources

- IS 875 Part 1 tables
- Custom density library
- BIM/material library
- User overrides

7.5 Live Load Engine (IS 875 Part 2)

Purpose

Assign imposed loads based on occupancy and usage.

Live Load Logic Includes:

- Residential
- Commercial
- Industrial
- Warehouse
- Assembly buildings
- Corridor/staircase amplifications
- Vehicle loads (garage/parking)
- Crane loads (PEB/industrial)
- Storage racks & mezzanine loads

AI auto-detects LL from building usage descriptions.

7.6 Wind Load Engine (IS 875 Part 3)

Purpose

Generate complete wind load data for:

- PEBs
- RCC buildings
- Steel/concrete towers
- Tanks & chimneys
- Sheds, canopies, boundary walls

Key Components

7.6.1 Wind Speed Retrieval

- City lookup
- Risk coefficient (k1)
- Terrain category (k2)
- Topography multiplier (k3)
- Importance factor (k4)

7.6.2 Pressure Coefficients

- Cp & Cpi from tables

- Roof angle-based coefficients
- Canopy coefficients
- Chimney and silo coefficients
- Internal pressure automation

7.6.3 Load Application Types

- Nodal
- Elemental
- Surface pressure
- Equivalent static forces

7.6.4 Wind Load Maps

Generated per structure:

- Roof wind pressure map
- Wall pressure diagram
- Uplift forces
- Lateral drift checks

7.7 Earthquake Load Engine (IS 1893)

Purpose

Compute seismic forces for all building types including industrial structures.

Key Components

1. **Zone Mapping**
2. **Soil Type Detection**
3. **Response Spectrum Generation**
4. **Base Shear Calculation**
5. **Mode Shape Extraction** (Kratos/FEniCSx/OpenSees depending on solver)
6. **Storey Force Distribution**
7. **Response Combination:**
    - SRSS
    - CQC
8. **Ductility/Importance Adjustment**

Outputs

- Story shear table
- Response spectra graphs
- Mass participation
- Seismic load cases for solver

7.8 Special Load Engine

7.8.1 Temperature Loads

- Expansion/contraction
- Roof sheeting & PEB temperature gradients

7.8.2 Snow Loads

(Only for Himalayan region buildings).

7.8.3 Impact Loads

Cranes, forklift, vehicular.

7.8.4 Water & Pressure Loads

- Retaining walls
- Tanks

- Swimming pools
- Landfills (leachate head pressure)

7.8.5 Blast/Explosion Loads

Optional module.

7.9 Automatic Load Combinations

Indian Standard Combinations

- IS 875 (all parts)
- IS 456
- IS 800
- IS 1893 seismic combinations

Global Standards

Optional:

- ASCE 7
- Eurocode EN 1990
- AISC/IBC

Load Combining Engine Features

- Auto-generate combos for:
    - ULS
    - SLS
    - Seismic
    - Wind critical
- User-defined custom combinations
- AI-based elimination of non-governing combinations

7.10 Load Engine Outputs

Every calculated load must be exportable as:

1. JSON Specification (Primary)

Consumed directly by:

- Kratos Multiphysics
- FEniCSx
- OpenSeesPy
- STAAD file exporter

2. Human-Readable Load Reports

Including:

- Tables
- Diagrams
- Floor load maps
- Pressure diagrams
- Validation with reference codes

3. DXF/IFC Load Drawings

For BIM / documentation.

7.11 Validation Layer (Must-Have)

Before loads go to analysis:

- Check for missing data
- Auto-correct inconsistent inputs

- Verify code compliance
- Sensitivity check (±10% variation)
- Generate warnings & suggestions

Example warnings:

- *“Wind internal pressure undefined — using Cpi = ±0.2 for enclosed building.”*
- *“Seismic mass missing for roof — derived from dead load.”*
- *“Live load reduction applicable for >3 floors.”*

7.12 AI-Assisted Load Extraction (Advanced)

AI should be able to:

- Extract loads from **sketches, PDFs, hand drawings**
- Predict crane loads from plan
- Map load direction from architectural plans
- Predict missing wall loads from plan thickness

7.13 Performance Requirements

- Must compute loads for a 20-storey building < 0.7 seconds
- Must generate 100+ combinations < 0.2 seconds
- Must handle 1M+ elements (Kratos/FEniCSx simulations)

7.14 Extensibility

The Load Engine must be flexible enough to add:

- New standards
- New structure categories
- Custom load modules
- Industry templates (e.g., oil & gas, pharma clean rooms)

7.15 API Specification (Simplified)

Example:

```
{
 "structure_type": "RCC_G+2",
 "location": "Hyderabad",
 "codes": ["IS875", "IS1893", "IS456"],
 "geometry": {...},
 "materials": {...},
 "loads": {
  "dead_load": {...},
  "live_load": {...},
  "wind_load": {...},
  "seismic_load": {...},
  "special_loads": {...},
  "combinations": [...]
 }
}
```

✔ Part 7 Completed

This Load Engine PRD is ready for integration with:

- **Kratos Multiphysics (preferred)**
- **FEniCSx**
- **OpenSeesPy**
- **STAAD import/export module**
- **Your AI reasoning & workflow system**

📘 **PRD — PART 8**

STRUCTURAL VISUALIZATION & INTERACTIVE 3D MODELING ENGINE (STAAD-LIKE MODULE)**

8.1 Purpose

This module enables Engineers to:

- **View** the model in 3D (nodes, members, plates, solids).
- **Interactively Modify** geometry, sections, loads, supports, meshing.
- **Inspect the model** via section planes, exploded views, transparency, displacements.
- **Validate before analysis** (catch modeling mistakes).
- **Edit and regenerate** the analysis model instantly.
- **Export/Import** in STAAD/ETABS/SAP2000/Kratos formats.

It is the visual "workspace" of the entire platform—like STAAD, Tekla Structural Designer, ETABS, Robot.

This is the **core engineering UI**.

8.2 Key Objectives

1. Intuitive, STAAD-like 3D viewer for all structure types.
2. Real-time geometry editing (nodes, beams, plates, solids).
3. Visualization of:
    - Sections
    - Loads
    - Supports
    - Meshing
    - Boundary conditions
4. 3D navigation smooth and modern:
    - Orbit
    - Pan
    - Zoom
    - Fit-to-screen
5. Cross-sectional slicing:
    - Cut XYZ planes
    - Multi-slice mode
    - Adjustable clipping box
6. Preview design results (stress, deflection, utilization) after analysis.
7. Export drawings to DXF/DWG/KML/OBJ/IFC.
8. Integration with Kratos Multiphysics / FEniCSx numerical model.

8.3 Technology Stack

Frontend Visualization

- **Three.js or Babylon.js** (WebGL 3D rendering)
- **React/Next.js** (UI framework)
- **VR/AR optional**: WebXR integrations

Backend Analysis/Model Control

- **Kratos Multiphysics** (preferred FE solver)
- Python-based model builder

- gRPC/REST bridge between UI and solver engine

Data Interchange

- JSON-based Model Definition Language (**SMDL**)
- IFC export via ifcopenshell
- DXF/DWG via ezdxf
- STAAD-like input using custom parser

8.4 Inputs Gathered by This Module

The 3D modeling workspace receives:

- Geometry (grid, nodes, members, plates, solids)
- Sections (steel, concrete, composite)
- Material properties
- Support conditions
- Member releases
- Loads:
    - Point loads
    - UDLs
    - Temperature
    - Prestress
    - Wind & seismic maps
- Meshing parameters
- Storey data
- Design parameters (IS/AISC/EC/ACI)
- Foundation links

All inputs can be changed directly in 3D or in side-panel property grids.

8.5 Functional Requirements

8.5.1 Model Visualization

✓ 3D Wireframe

✓ 3D Solid model

✓ Color-coded elements (steel, RCC, walls, slabs, landfill layers, RE layers)

✓ Visibility layers

✓ Ghost mode (transparent)

✓ Exploded view by storeys or groups

✓ Auto-layout for complex structures

8.5.2 User Interaction

Selection

- Single click
- Multi-select (Ctrl/Shift)
- Lasso/rectangle select
- Select by property (section, material, type)

Editing

- Move nodes
- Add/delete nodes and members
- Change orientation of beams/columns

- Drag-to-extend members
- Change plate thickness
- Edit foundation sizes
- Edit wall heights
- Add doors/windows/openings for architectural flows

Snapping

- Grid snap
- Midpoint, endpoint, intersection
- Orthogonal lock

Undo/Redo

- Unlimited history

8.5.3 Real-Time Validation

After each edit:

- Check connectivity (dangling nodes)
- Check load assignments
- Check member orientation
- Check storey alignment
- Check self-weight directions
- Highlight errors graphically

8.5.4 Section Cuts & Inspection Tools

Cuts

- X-plane, Y-plane, Z-plane
- Dynamic drag-able slice
- Multi-slice mode
- Thickness visualization
- View moments/forces at slice

Section Inspector

Click any element → see:

- Node coordinates
- Length, orientation, releases
- Material
- Section properties
- Local axis orientation

8.5.5 Load Visualization

Types

- Nodal loads
- Beam loads (UDL, UVL, moments)
- Plate pressure
- Surface loads
- Mass participation
- Soil pressure diagrams
- Wind pressure diagram (IS 875)
- Seismic shape modes (IS 1893)

Representation

- Arrows, color maps, contour shading
- Load scale slider
- Toggle per load case / group

8.5.6 Meshing View

- Display mesh (lines/triangles/tetra)
- Mesh quality report (skewness, aspect ratio)
- Auto-mesh generator (Kratos/FEniCSx)
- User-controlled refinement regions

8.5.7 Results Visualization (Post-analysis)

Stress Plots

- Sxx, Syy, Szz
- Von Mises
- Plate bending moments Mx, My, Mxy

Displacements

- Scaled deflection shapes
- Animate mode shapes
- Time-history animations
- Harmonic analysis animations

Member Forces

- Axial, Shear, Moments
- Interaction ratios (IS 800 9.3.2)
- Buckling modes

Color Maps

- Red → critical
- Green → OK
- User-defined thresholds

8.5.8 Model Modification After Results

User can:

- Change geometry
- Change sections
- Change loads
- Change supports

Then **click "Reanalyze"** → backend rebuilds FE model → updates results.

8.5.9 Kratos Integration Requirements

The visualization module interacts with Kratos:

- Sends geometry as JSON SMDL → Kratos model builder converts to:
  - Lines → Beam elements
  - Surfaces → Shell elements
  - Solids → Hex/Tet
- Applies materials
- Applies boundary conditions
- Runs:
  - Static analysis

    - Modal analysis
    - Harmonic analysis
    - Nonlinear analysis
- Returns:
    - Nodal displacements
    - Element stresses
    - Reaction forces
    - Mesh data

The results are mapped back to 3D viewer.

8.6 Export Features

Supported Formats:

- STAAD .std
- ETABS .e2k
- SAP2000 .s2k
- IFC (architectural integration)
- DXF (2D plans)
- OBJ/GLTF (3D model)
- KML (Google Earth overlays)
- Kratos .mdpa

8.7 User Roles

Engineer

- Full 3D editing
- View results
- Generate drawings

Architect

- Modify layout & openings only
- Add architectural blocks

Owner / Client

- View-only mode

Reviewer (Authority)

- View analysis summary
- Comment features

8.8 Performance Requirements

- Should handle up to:
    - **100,000 nodes**
    - **50,000 elements**
    - **100 storeys**
- GPU-accelerated rendering
- Level-of-detail rendering for large models
- Multi-threaded model rebuild
- WebWorkers to keep UI responsive

8.9 UI Layout Structure

Left Panel (Properties Panel)

- Node/Element properties
- Material/Section selection
- Layer visibility

Top Toolbar

- File (new/open/export/import)
- Edit (undo/redo)
- Add (node/member/plate/solids/opening)
- Loads (assign/view)
- Supports
- Meshing
- Slicing
- Results
- Drawing generation

Center Area

- 3D canvas (Three.js/Babylon.js)

Right Panel (Model Explorer)

- Hierarchical tree:
    - Storeys
    - Members
    - Plates
    - Walls
    - Foundations
    - Load cases
    - Groups
    - Materials

8.10 Integration With Downstream Modules

The 3D model is the gateway to:

- Architectural drawings generator (Part 9+)
- Structural design engine (IS/EC/AISC/ACI)
- Detailed drawings & rebar drawings (Part 11)
- Estimation engine (SSR-based)
- BoQ/Rate Analysis
- Project execution planning
- Scheduling (Gantt)
- Site management module

Everything downstream depends on the accuracy and editability of Part 8.

8.11 Summary of Part 8

This module transforms your app from a simple “form-based wizard” into a **full engineering platform similar to STAAD/ETABS/Tekla Structural Designer**, with real-time:

- 3D modeling
- Visualization
- Editing
- Analysis previews
- Interaction with Kratos solver

- Export to common engineering formats
- Scalable UI for buildings, PEBs, RE walls, landfills, foundations, bridges, etc.

This is one of the most critical parts of the entire platform.

📘 PART 8 — GEOMETRY VISUALIZATION & MODEL EDITOR

8.1 Overview

This module forms the **interactive modeling workspace**, replicating and extending the functionalities of tools like **STAAD, Tekla, ETABS, Revit Structure**, but fully integrated inside your AI-driven application.

This module is used by:

- Structural Engineers
- Architects
- Draftspersons
- General Contractors
- Students / Consultants
- AI-based automated design pipelines

It handles:

- **3D geometry generation**
- **2D plan / elevation / section views**
- **Member-level editing**
- **Model validation**
- **Load application visualization**
- **Analysis model preview before solving**
- **Versioning of the model**
- **Interactivity (drag, drop, scale, snap, mirror)**

The user can **view → edit → validate → run analysis → generate drawings** in a seamless workflow.

8.2 Core Objectives

This module must:

1. Create and visualize **any type of structure**:
    - PEB
    - Steel buildings
    - RCC buildings
    - Industrial buildings
    - Retaining walls / RE walls
    - Landfills (bunds, benches, surfaces)
    - Foundations (isolated, combined, raft, pile)
    - Roads / drains
    - Tanks / pits
    - Composite steel–concrete structures
2. Provide a visual interface similar to STAAD:
    - Node / member table
    - Geometry editor
    - 3D orbit/pan/zoom
    - Cross-sections on any axis
    - Member selection by click
    - Property assignment by click
    - Real-time updating of model geometry
3. Allow **real-time editing**:
    - Move nodes

    - Modify spans/heights/levels
    - Adjust cross-sections
    - Add/delete beams, columns, braces
    - Adjust slopes, loads, slab geometry
4. Support **parametric modeling**:
    - Edit a dimension → model updates
    - Change bay spacing → structure regenerates
    - Change floor count → columns replicate automatically
5. Handle **import/export**:
    - Import from STAAD, Tekla, IFC, DXF
    - Export to STAAD (.std), DXF, IFC
    - Export analytical model to Kratos, Fenicsx, OpenSees
6. Provide cross-sectional tools:
    - Section cut at any plane
    - Slice through floors
    - Display bending moment/deflection shape after analysis

8.3 Functional Requirements (Detailed)
8.3.1 Model Creation Tools
Geometric primitives:
- Node
- Beam
- Column
- Slab
- Wall (retaining, shear, RE)
- Footing
- Surface (landfill top, benches)
- Solid (tank, pit, block foundations)

Complex primitives:
- Portal frames (PEB)
- Multi-storey grid frames
- Plate/shell meshes
- Surface models for landfills
- Parametric roofs (gable, curved, saw-tooth)

8.3.2 Model Editing Tools
Users can:
- Drag nodes
- Snap to grid
- Snap to intersections
- Align nodes/members
- Add intermediate nodes
- Modify member release conditions
- Change boundary conditions
- Change local axes
- Change member properties

8.3.3 Parametric Input and AI Guidance

When the user enters a project description like:
“Construct a G+2 building in Hyderabad”
The AI:

1. Extracts requirements
2. Suggests templates
3. Requests missing inputs
4. Generates a baseline model
5. Opens it in **3D Model Editor** for further human refinement

---

8.4 Visualization Engine Requirements

8.4.1 3D View (STAAD-like)

- Real-time WebGL/Three.js visualization
- Orthographic & perspective views
- Pan, zoom, rotate
- Load visualizers (arrows, patches, wind pressures)
- Deflection shape viewer
- Animation for seismic loads

8.4.2 2D Views

Plan View:

- Show slab layout
- Beam/column grid
- Labels, section marks

Elevation View:

- Show storeys
- Member sizes
- Axis labels

Section View:

- Slice at any chosen plane
- Display reinforcement, wall thickness, etc.

8.4.3 Cross-section Inspector

Clicking a beam/column opens:

- Cross-section geometry
- Properties (A, Ix, Iy, Iz, torsion)
- Material
- Releases
- Design forces (after analysis)
- Utilization ratios (IS/AISC/EC codes)

8.5 Model Validation Engine

Before analysis:

- Check unconnected nodes
- Check duplicate elements
- Check unsupported nodes
- Check missing material/section

- Check slab mesh connectivity
- Check load cases completeness

User sees:

- Warnings
- Errors
- Suggestions

8.6 Model Versioning

Every time the user modifies geometry:

- Auto-save version
- Ability to rollback
- Ability to branch designs
- Change logs with diffs

8.7 Integration with Analysis Engine (Kratos/FEniCSx)

When the model is validated:

1. Convert geometry → mesh
2. Apply loads and BCs
3. Solve using selected engine:
    - Kratos Structural
    - Kratos Solid
    - Kratos Geo
    - FEniCSx for continuum
4. Send results back to visualization

8.8 Outputs from this Module

1. **3D Model File**
2. **Analytical Model File**
3. **Mesh File**
4. **Load Visualization**
5. **Design-ready input**
6. **Drawing-ready geometry**

8.9 Non-functional Requirements

- Web-based (WebGL)
- Fast rendering
- Multi-device support (desktop/tablet)
- Zero-lag interaction
- Works with large models (10k–50k elements)

8.10 Deliverables of Part 8

This section completes:

- Full 3D/2D modeling UI
- Editing pipeline
- Visualization engine
- Validation engine
- Integration map to Kratos/FEniCSx

- Import/export support

PART 9 — ANALYSIS ENGINE (SOLVER CORE) — FULL PRD
Unified Multi-Solver Analysis Framework for Buildings, Industrial Structures, Landfills, and Civil Systems

---

9.1 Objective of the Analysis Engine
The Analysis Engine is the **numerical heart** of the system.
Its job is to convert:
- User inputs
- Drawings
- Generated parametric models
- Uploaded sketches
- Selected templates
- Architecture-based auto-models

→ into a **fully solvable 3D model**, capable of:
- Static analysis
- Linear & nonlinear analysis
- Modal/dynamic analysis
- Pushover
- Soil–structure interaction
- Temperature/creep/shrinkage
- Fluid & thermal simulations (where relevant)
- Large deformation analysis (retaining walls, landfills)

The engine must support **multiple solvers**:
Primary Solver (default)
✓ **Kratos Multiphysics** (chosen by you) — ideal for:
- Structural Mechanics
- Geo-mechanics (retaining walls, landfills, slopes)
- Concrete & Steel constitutive models
- Nonlinear FEM
- Coupled multi-physics (optional future upgrade)

Secondary / Optional Backends
The system will allow plug-in solvers:
- **FEniCSx** – academic-grade FEM
- **OpenSees** – structural/dynamics
- **Code_Aster** – high-fidelity EU-grade
- **STAAD import/export** – compatibility layer
- **ETABS/SAP model export** – optional future upgrade

All solvers are wrapped behind a **Unified Analysis API** so the UI and workflows never depend on any single solver.

9.2 Unified Analysis API
Every solver follows one common interface:
9.2.1 Core Methods
1. **build_model(structural_model)**
   Converts UDM → element mesh → loads/boundary conditions → solver-ready format.

2. **run_static(gravity + live + wind + seismic)**
   Produce:
   - Node displacements
   - Member forces (axial, shear, bending)
   - Stress field
   - Reactions
3. **run_modal()**
   - Natural periods
   - Mode shapes
   - Participation factors
4. **run_response_spectrum(IS 1893 / ASCE 7 / EC8)**
5. **run_pushover()**
6. **run_thermal()** (future update)
7. **run_geotechnical()** (retaining wall, RE wall, landfill)
8. **export_results()**
   Standard results for design & drawings.

9.2.2 Output Format

Results shall follow a strict schema:

- Displacements (UX, UY, UZ)
- Rotations (RX, RY, RZ)
- Member Forces (Fx, Fy, Fz, Mx, My, Mz)
- Section Stress/Strain
- Soil pressures / contact reactions
- Plastic hinge formation (if nonlinear)
- Crack width (RCC)
- Buckling factors (steel frames)

Stored in:

- SQLite for session
- JSON for API
- Parquet for fast large-data storage

9.3 Kratos Multiphysics Integration (Primary Engine)

The engine container includes:

Kratos Modules Needed

Module	Purpose
StructuralMechanicsApplication	Steel/RCC frames, slabs, beams
GeoMechanicsApplication	Landfills, slope stability, RE walls
FluidDynamicsApplication	(future) wind/fluid simulation
PFEM2Application	Soil–structure interaction
ConstitutiveModelsApplication	Concrete nonlinearity
ContactStructuralMechanics	Retaining walls, footing-soil contact

Capabilities Enabled

✓ Nonlinear geometry

✓ Nonlinear material

✓ Reinforced concrete constitutive laws

✓ Steel plasticity
✓ Soil consolidation
✓ P-Delta effects
✓ Large deformation (landfill)
✓ Earthquake analysis
✓ Time history
✓ Modal analysis
The final system will allow:
Kratos as the universal solver → Designs generated using IS / AISC / ACI / EC codes.

9.4 Analysis Workflows
9.4.1 For RCC Buildings
- Auto-mesh slabs (quad elements)
- Convert beams/columns to 3D beam elements
- Generate shear-wall meshes
- Assign concrete models:
  - M20 to M70
  - CEB-FIP or Eurocode Concrete Model

Static analysis includes:
- Load combinations per IS 456 + IS 875 + IS 1893
- Gravity load nonlinearities
- Crack modeling (if selected)

Output:
- Storey drifts
- Bending moment envelope
- Shear forces
- Design forces for reinforcement calculation

9.4.2 For Steel Buildings (PEB / Industrial)
- Portal frames meshed as beam elements
- Bracing as truss elements
- Purlins as secondary members
- Connections extracted for design
- Wind loads per IS 875-3 → auto-application
- Seismic loads per IS 1893 → mass modeling

Output:
- Frame-member forces
- Buckling modes
- Base reactions
- Connection forces for design
- Deflection envelopes

9.4.3 Retaining Wall
Two ways:
1. **Quick Mode (Rankine + Stability Checks)**
2. **Advanced Kratos FEM Mode**

    - Soil mesh (plane strain)
    - Wall mesh
    - Soil–wall contact
    - Nonlinear soil (Mohr–Coulomb)
    - Groundwater pressure (optional)

Outputs:

- Soil pressure distribution
- Base shear & moment
- Sliding/overturning FS
- Stress under footing

9.4.4 RE Wall

- Soil + reinforcement zone
- Facing modeled using shell or beam elements
- Pullout resistance
- Tensile stresses in geogrid
- Block stability

9.4.4 Landfill (Civil / Geotech)

- Multi-layer waste profiles
- Compaction density
- Leachate mound pressures
- Slope stability using shear strength reduction
- Settlement using consolidation

Outputs:

- FOS
- Stress paths
- Settlement contours
- Bund stability analysis

9.5 Mesh Generation Architecture

Different structure types → different mesh tactics:

Structure	Preferred Mesh
RCC Frame	1D beams + 2D slabs (shells)
Steel PEB	Beam elements, shell for cladding (optional)
Retaining Wall	2D plane-strain
RE Wall	2D + embedded truss (geogrid)
Landfill	2D/3D soil mesh

Automatic meshing strategies:

- Quad-dominant for slabs
- Tri/quad mix for soil
- Adaptive refinement near stress risers

Mesh quality checks:

- Aspect ratio
- Warping
- Check Jacobian > 0

- Refinement at openings

9.6 Load Application Framework
Automatic load generation:
IS 875 Loads

- Dead load
- Imposed load
- Roof live load
- Collateral loads (purlins, ducts)
- Water tank
- Partition wall loads

Wind Loads

- Based on:
  - Terrain category
  - Structure class
  - Pressure coefficients
  - Frame zones
- Kratos auto-application on surfaces/nodes

Earthquake Loads

- IS 1893:
  - Response spectrum
  - Modal combination
  - Base shear check
  - Vertical EQ (optional)

9.7 Nonlinear Analysis Modes
Supported:

- Material nonlinearity (cracked concrete, steel bilinear)
- Geometric nonlinearity (P-Delta)
- Contact nonlinearity
- Large deformation (landfills)
- Plastic hinge formation
- Load-step convergence control

9.8 Result Visualization and 3D Model Interaction
Your requirement:
"User Engineer should see, visualize, modify the model in 3D with cross-section views like STAAD."
The system will provide:
3D WebGL Viewer

- Full 3D model view
- Node / element visualization
- Cross-section viewer
- Real-time rotation/zoom/pan
- Color-coded results
- Slice view
- Deflected shape display

- Mode shapes animation
- Stress contour maps

Editable Model Interface

- Click-and-drag nodes
- Change section sizes
- Modify member orientation
- Add/delete openings
- Adjust supports
- Regenerate mesh instantly
- Re-run analysis from UI

All edits modify the **Unified Data Model**, not solver files.

9.9 Model Export & Interoperability

Supporting:

Import

- Sketch/image → AI → parametric model
- STAAD .std
- AutoCAD DWG/DXF
- ETABS .edb (optional)
- IFC (BIM Revit)

Export

- STAAD .std
- DXF
- IFC
- Kratos .mdpa
- JSON model for other tools

9.10 Performance and Cloud Compute

- Kratos runs in **Docker microservices**
- Auto-scaling for large models
- Models up to:
    - 1,000,000 DOF (cloud mode)
    - 200,000 DOF (desktop mode)

9.11 Logging, Validation, and QA

- Solver logs exposed to user
- Error messages with fix suggestions
- Engineering validation:
    - Compare base shear vs code
    - Drift limit checks
    - Natural period bounds
    - Stability checks

PART 9 — DESIGN ENGINE ARCHITECTURE (FULL MULTI-CODE SYSTEM)

(This is a major core of the PRD. Extremely detailed. Ready to drop into your final PRD volume.)

PART 9 — DESIGN ENGINE ARCHITECTURE

A Unified, Multi-Code, Multi-Material Structural Design System

---

9.1 PURPOSE AND ROLE OF THE DESIGN ENGINE

The **Design Engine** is the computational heart of the entire application.
It transforms:

- **Geometry**
- **Meshed Finite Element Model** (from Kratos/FEniCSx backend)
- **Load Cases & Combinations**
- **Material Models**
- **Boundary Conditions**
- **Analysis Results** (forces, moments, stresses, displacements)

into:

- **Member Capacity Checks**
- **Section Optimizations**
- **Connection Design**
- **Stability Checks**
- **Foundation Design**
- **Detailing Rules**
- **Design Reports (IS / ACI / AISC / EC format)**

The engine must be flexible enough to design *any* structural system:

- RCC G+2 building
- PEB steel building
- Composite structures
- Retaining walls, RE walls
- Industrial structures
- Landfills (stability + geotechnical modules)
- Customized user-defined buildings imported from plan/sketch

---

9.2 HIGH-LEVEL ARCHITECTURAL OVERVIEW

The Design Engine consists of 7 Sub-Engines:

1. **Design Input Interpreter**
2. **Material & Section Classification Engine**
3. **Code Rulebook Engine (IS / ACI / AISC / EC)**
4. **Member Capacity Engine**
5. **Connection Design Engine**
6. **Foundation Design Engine**
7. **Design Documentation & Hand-Calculation Generator**

These engines run in sequence, but also support re-runs, backtracking, edits, re-calculations and optimizations.

9.3 DESIGN INPUT INTERPRETER

This module converts:

- User manual inputs
- AI-derived building information
- Extracted geometry
- Structural 3D model
- Member layouts

- Loads
- Boundary conditions

Into a *standardized design dictionary*.

Design Input Dictionary Example


```
{
 "structure_type": "RCC_BUILDING",
 "codes": ["IS456", "IS1893", "IS875"],
 "members": {
  "beams": [{...}],
  "columns": [{...}],
  "slabs": [{...}],
  "walls": [{...}]
 },
 "materials": {
  "concrete": "M30",
  "steel": "Fe500"
 },
 "load_combinations": [{...}],
 "analysis_results": {
  "Fx": [...],
  "Fy": [...],
  "Mz": [...]
 }
}
```


This normalizes all model variations into a universal format.

9.4 MATERIAL & SECTION CLASSIFICATION ENGINE

Purpose

Correct classification is required to apply code rules.

Supports:

- **Steel Sections**
  - I-sections
  - Channels
  - Tapered (PEB) profiles
  - Pipes/CHS/RHS
  - Custom welded sections
- **Concrete Sections**
  - Rectangular
  - L-shaped
  - T-beams
  - Flanged beams
  - Shear walls
  - Rafts
- **Composite Sections**
  - SRC columns
  - Composite beams
  - Steel deck slabs

- **Geotechnical Elements**
    - Soil layers
    - RE reinforcement
    - Backfill parameters
    - Landfill waste density

Outputs

- Geometric properties (A, Ix, Iy, Iz, r, Zp, Ze)
- Effective properties (considering local buckling)
- Class (compact / semi-compact / slender)

9.5 CODE RULEBOOKS ENGINE

This is one of the most critical sub-systems.
The app must support **dynamic multi-code compliance**.

Supported Codes (Phase 1)

India (Primary)

- **IS 456:2000** (RCC)
- **IS 800:2007** (Steel – LSD)
- **IS 801** (Thin-walled)
- **IS 875 Part 1–5** (Loads)
- **IS 1893** (Earthquake)
- **IS 13920** (Ductile detailing)
- **IS 2911** (Piles)

International (Secondary)

- **ACI 318** (Concrete design)
- **AISC 360** (Steel design – LRFD/ASD)
- **AISC 358** (Connections)
- **Eurocode 2/3/4/7**
- **IBC/ASCE** (Loading where required)

The rulebook engine benefits:

1. Any structure can be designed using IS codes as default.
2. International clients get AISC/EC options.
3. Connection design automatically picks from AISC/AISC358.
4. Dot-by-dot clause referencing is stored for report generation.

9.6 MEMBER CAPACITY DESIGN ENGINE

Roles

For each member:

- Get design forces:
    - N, Vx, Vy, Mx, My, Mz
- Extract section properties
- Apply code clauses
- Implement interaction formulas
- Determine: **PASS / FAIL / NEED OPTIMIZATION**

For Steel (PEB, Industrial steel, frames)

Checks include:

- Axial compression / tension
- Flexure (major/minor)

- Shear
- Combined axial + bending (Clause 9.3.2 IS 800)
- Lateral torsional buckling
- Web crippling
- Buckling curves (IS 800 Annexure E)
- Local buckling
- Serviceability (deflection, frequency)

For RCC (Beams, Columns, Slabs, Footings)
Checks include:

- Flexure
- Shear
- Torsion
- Combined axial + biaxial bending
- Slab reinforcement (one-way & two-way)
- Shear wall design
- Ductile detailing (IS 13920)

For Retaining Walls / RE Walls
Checks include:

- Overturning
- Sliding
- Bearing pressure
- Structural bending/shear
- RE reinforcement pullout

For Landfills

- Embankment slope stability
- Settlement
- Waste mechanical properties
- Liner stress check
- Leachate load effects

9.7 CONNECTION DESIGN ENGINE
For steel buildings (especially PEB):

- Beam–column moment connections
- Base plates
- End plates
- Splices
- Gusset plates
- Bracing connections
- Welds
- Bolts (HSFG/BO)
- Anchor bolts (**ACI 318 preferred**)
- Hanger connections
- Crane beam connections

Each connection:

- Reads member forces
- Applies selected code (AISC/EC/IS800)
- Generates:

- Plate dimensions
- Bolt layout
- Weld sizes
- Punching checks
- Bearing & block shear checks

9.8 FOUNDATION DESIGN ENGINE

Supports:

- Isolated footings
- Combined footings
- Strap beams
- Rafts (rigid or FEM)
- Pile caps
- Short piles / long piles (IS 2911)

Design includes:

- Soil bearing capacity checks
- Punching
- One-way shear
- Bending
- Combined load envelopes
- Uplift/suction (wind uplift for PEB)
- Sliding

For anchor bolts:

→ **ACI 318 Chapter 17** governs tension, shear, breakout, pryout.

9.9 OPTIMIZATION & AUTO-SIZING

For each member:

- Retrieve failure modes
- Select next higher/lightest feasible section
- Maintain practical depth limits
- Maintain architecture constraints

Optimization strategies:

- Greedy section optimization
- Genetic algorithm for large buildings
- Cost-based optimization (₹/kg + fabrication complexity)
- Constraint-based optimization (deflection limit × cost)

9.10 DESIGN DOCUMENTATION ENGINE

Generates output automatically:

1. Hand-calculation style PDF

- Clause references
- Equations shown
- Step-by-step derivation
- Pass/Fail summary

2. Detailed STAAD-like Report

- Member tables
- Load combinations

- Analysis results
- Design results
- Connection details

3. DXF / PDF Drawings

- Beam reinforcement
- Column reinforcement
- Foundation drawings
- PEB connection sketches
- Weld/Bolt details
- Anchor bolt templates

4. Excel Export

- BOQ
- Steel tonnage
- Concrete volume
- Rebar weights

PART 9 Output Summary

The Design Engine:

- Unifies all codes and practices
- Designs every structural component
- Works seamlessly with Kratos/FEniCSx analysis results
- Produces drawings + reports
- Enables optimization
- Supports AI-driven decision-making
- Scales from residential G+2 to large PEB industrial buildings
- Can be modularly priced per-feature

PART 10 — DESIGN ENGINE (MULTI-CODE, MULTI-MATERIAL) — DETAILED SPECIFICATION

This section defines the **core design engine** that converts the analyzed structural model into **code-compliant member design, connection design, foundation design, and reinforcement detailing**.

The engine must be fully modular, extendable, deterministic, transparent, and auditable—capable of producing outputs acceptable to **Indian Authorities**, including **GHMC, HMDA, Panchayat Raj, PWD, Industrial Local Bodies, Fire Dept, PMC/Consultants, Banks/Financiers**, etc.

The design engine is the backbone of the system, responsible for transforming geometry + loads + user intent into actionable engineering designs.

10.1 PURPOSE OF THE DESIGN ENGINE

The design engine shall:

1. **Interpret the solved FEM model** (from Kratos/FEniCSx).
2. **Extract forces, stresses, reactions, modal results**, etc.
3. **Classify members** (beams, columns, rafters, joists, plates, walls, footings).
4. **Apply design code logic** based on:
    - IS 456, IS 800, IS 801
    - IS 875 (all parts)
    - IS 1893
    - IRC, IS 3370
    - ACI 318, AISC 360, Eurocode 2/3/8 (optional)
5. **Perform section checks** (strength + serviceability).
6. **Perform connection design** using appropriate code rules.
7. **Generate reinforcement / steel detailing**.
8. **Generate auto-layout and drawings** (Part 12 & 13).
9. **Create a structured design report**:
    - Machine readable (JSON)
    - Human readable (PDF)
    - “Handwritten Calcs” style formatted for authorities

10.2 ARCHITECTURAL PRINCIPLES

10.2.1 Modular

Each design module must be isolated:

```
design/
 steel/
   is800/
   aisc/
   eurocode3/
 rcc/
   is456/
   is13920/
   is3370/
 foundations/
 connections/
 stairs/
```

retaining_walls/
re_walls/
slabs/
footings/
raft/

10.2.2 Code Polymorphism

Same input → Different code rules depending on country/standard:

```
beam.design(code="IS800")
beam.design(code="AISC")
beam.design(code="EC3")
```

10.2.3 Deterministic + Auditable

- Every calculation must be reversible to step-by-step formula view.
- Auto-generated "black box" formulas **not allowed**.
- The user can see:
  - governing clause
  - formula
  - substituted values
  - result
  - pass/fail

This is mandatory for **approving authorities**.

10.3 INPUTS TO DESIGN ENGINE

A. From FE Solver

- Nx, Vy, Vz, Mx, My, Mz for each 1D element
- Plate forces: Nx, Ny, Nxy, Mx, My, Mxy
- Stress tensors
- Reaction forces
- Deflections
- Mode shapes (optional)

B. From User

- Preferred materials & grades
- Design codes
- Serviceability limits
- Fire requirements
- Crack width/deflection limits
- Maximum rebar diameters / steel section libraries
- Foundation system type preference

C. From Internal Library

- Steel shapes database (Indian + International)
- Concrete rebar database
- Material properties
- Soil SBC library
- Bolt, weld, anchor bolt databases

10.4 DESIGN WORKFLOW

1. Member Recognition

The system must automatically recognize:

- Primary beams
- Secondary beams / joists / purlins
- Columns
- Rafters
- Truss members
- Bracings
- Slabs (2D)
- Walls (2D or 3D)
- Foundations (auto-placed after reactions)
- Staircases
- Water tanks
- Retaining walls / RE walls

2. Load Combination Generation

Automated LC creation as per:

- IS 875 combinations
- IS 1893 seismic combinations
- Fire combinations (if applicable)

3. Demand Extraction

For each element:

```
D = {
  axial,
  shear-major,
  shear-minor,
  moment-major,
  moment-minor,
  torsion,
  buckling length,
  effective lengths
}
```

4. Section Selection (Steel)

- Auto-select from library OR
- Use user-selected section
- Perform preliminary sizing based on demand-capacity ratios

5. Code Checks

Steel – IS 800:2007 (LSM)

- Cross-section classification (plastic, compact, semi-compact, slender)
- Axial strength
- Bending major/minor
- Shear major/minor
- Combined interaction (Clause 9.3.2)
- LTB buckling (Clause 8.2.2)
- Design of tension members
- Design of compression members (Euler buckling curves)
- Torsion (BS 5950 or Eurocode supplement)

Steel – AISC 360

- ASD/LRFD interaction checks
- Flexural, axial, shear

- Stability checks

Steel – Eurocode 3

- Axial, shear, bending
- LTB
- Interaction formula

RCC – IS 456:2000

- Flexure design (limit state)
- Shear design
- Development length
- Torsion
- Slenderness
- Serviceability
- Crack width control
- Deflection

RCC – IS 13920: 2016 (Ductile Design)

- Detailing rules for seismic zones
- Beam-column joint checks
- Shear wall detailing

Foundations

Includes:

1. **Isolated footing** (IS 456 + SBC)
2. **Combined footing**
3. **Strip footing**
4. **Raft foundation**
5. **Pile foundation**:
    - Pile axial + lateral + group efficiency
    - Pile cap design

Retaining Walls

- Overturning
- Sliding
- Bearing pressure
- Toe/heel reinforcement
- Seismic checks per IS 1893 Part 1 & 2

RE Walls

- Reinforcement length
- Pullout
- Connection strength
- Facing panel design

Water Tank (IS 3370)

- Circular/rectangular tank designs
- Wall + slab + dome + staging

10.5 CONNECTION DESIGN ENGINE

Connections must be selectable:

- Bolted

- Welded
- Hybrid

Steel connections:

- Beam-column moment connection
- Beam-column shear connection
- Base plates with anchor bolts
- Splice connections
- Gusset plates for bracing
- Truss nodes

Each designed per:

- IS 800 (Annexures)
- AISC 360
- Eurocode 3 – Part 1-8

Anchor Bolt Design (critical)

Since IS codes are insufficient:

Use **ACI 318-19 Appendix D** OR **AISC DG1** for:

- Tension breakout
- Shear breakout
- Pullout
- Interaction
- Edge distance checks
- Shear friction

10.6 DETAILING ENGINE INTEGRATION

Once member design is complete → trigger detailing rules:

- Auto rebar cutting/mapping
- Auto bar bending schedules
- Auto drawing coordinates & reinforcement blocks
- Auto dimensioning

Export formats:

- DXF
- DWG (if license available)
- SVG
- PDF
- IFC (BIM)

10.7 OPTIMIZATION LOOP (Integrated with Design)

The system must be capable of optimization:

Objective options:

- Minimize steel weight
- Minimize concrete volume
- Minimize cost (SSR-based)
- Maximize stiffness
- Mixed objectives

Constraints:

- Code-compliant (mandatory)
- Deflection limits

- Vibration limits
- Maximum depth limits
- Material availability

Method options:

- Gradient-free (GA)
- Mixed Integer Programming (MILP)
- Kratos/FE nonlinear loops

10.8 USER CONTROLS & OVERRIDES

User can override:

- Material grades
- Section choices
- Reinforcement preferences
- Ductility requirements
- Connection type
- Foundation type
- Optimization objectives

System must show:

- Original recommendation
- Overridden choice
- Impact (cost/strength ratio)

10.9 OUTPUTS

The Design Engine outputs:

A. Machine Outputs

- JSON summaries
- Member-by-member code check results
- Connection data
- Rebar schedules
- Foundation loads + designs
- QC flags

B. Human-Readable Outputs

- Detailed design report
- Structured “hand-calculation style” computation sheets
- Approval-ready PDF

C. Drawing/Data Outputs

- DXF/DWG
- 3D model (GLB/IFC)
- Bill of Materials

10.10 LOGGING, TRACEABILITY & Q/A

Every designed element:

- Logs all intermediate states
- Includes code clause references
- Is version-controlled
- Has a reproducible calculation path

This is essential for **audit**, **approval**, and **peer review**.

10.11 FUTURE-PROOFING

The engine must support:

- New codes easily
- New materials (GFRG, AAC, Bamboo, FRP)
- New modules (bridges, culverts, towers)
- Batch processing
- API integrations

✅ **PART 10 COMPLETE**

PART 11 — LOAD ENGINE ARCHITECTURE

Unified Multi-Code Load Generation & Load Combination System

(IS + ACI + AISC + EC + Local Byelaws + User-Defined Loads)

11.1 Objective

The Load Engine is responsible for generating **all structural loads**, load combinations, and analysis-ready load cases across all structure categories supported by the platform:

- RCC Buildings (G+2, high-rise, industrial, commercial)
- Steel Buildings / PEB / Hybrid structures
- Composite structures
- Retaining walls
- RE walls
- Landfills (slope loads, surcharge, operational loads)
- Water retaining structures
- Industrial buildings (crane loads, mezzanine loads, equipment loads)
- Bridges (future phase)
- Custom user-defined loads

The engine must support:

- **Automated load calculation based on location** (GIS-linked wind/seismic)
- **User-editable load parameters**
- **Visualization of loads on the 3D model**
- **100% compatibility with analysis solver (Kratos Multiphysics)**
- **Code-compliant load combinations**

11.2 Supported Building Codes

The Load Engine must include modules for the following:

India (Primary)

- **IS 875 Part 1 – Dead Loads**
- **IS 875 Part 2 – Imposed Loads**
- **IS 875 Part 3 – Wind Loads**
- **IS 875 Part 4 – Snow Loads**
- **IS 875 Part 5 – Special Loads**
- **IS 1893 – Earthquake Loads**
- **IS 456 – RCC**
- **IS 800 – Steel**
- **IS 801 – Light gauge steel**
- Local by-laws (GHMC, HMDA, DTCP, Panchayat norms)

International (Fallback & Special Cases)

- **ASCE 7**
- **IBC**
- **AISC 360**
- **AISC 341 (Seismic)**
- **Eurocode 1**
- **Eurocode 3 (Steel)**
- **Eurocode 8 (Seismic)**
- **ACI 318**

- **AASHTO (future)**

Industry-specific

- Crane/Class-IV loads
- Silo/hooper loads
- Tank loads (API 650/653)
- Warehouse racking loads
- Solar loads

11.3 Load Engine Inputs

11.3.1 Automatic Inputs (System Generated)

- **Location / GPS** (entered by user)
- Wind zone → from IS 875 map
- Seismic zone → from IS 1893 map
- Soil type (site class) → user input + database
- Terrain category
- Importance factor (building usage)
- Exposure category
- Topography factor
- Live load reduction factors
- Diaphragm type (rigid/semi-rigid)

11.3.2 User Inputs

- Structure type selection
- Number of floors / bays / spans
- Member sizes (initial)
- Roof type (gable/flat/curved)
- Slab type (RC/Steel/Decking)
- Staircase type
- Water tank loads
- Partition wall loads
- Equipment loads
- Machinery dynamic loads
- Retaining soil parameters (γ, ϕ, c, surcharge)
- Landfill loads (benching, lifts, compaction)

11.4 Load Types Supported

11.4.1 Dead Loads

- Self-weight (from geometry)
- Wall loads
- Floor finishes
- Roofing system (sheeting/purlins/insulation)
- Mezzanine system
- Façade and glazing
- Parapet walls
- Staircases
- Water tank loads
- Foundations (for stress checks)

11.4.2 Live Loads

- Uniform live load per IS 875 Part 2
- Concentrated loads
- Storage loads
- Office / Residential / Assembly area loads
- Corridor loads
- Staircase live loads
- Warehouse racking loads
- Industrial equipment loads

11.4.3 Wind Loads

- Based on IS 875 Part 3 (auto-calculated)
- Directional wind speeds
- Peak pressure coefficients
- Internal/external permeability
- Roof angle correction
- Pressure zones (A, B, C, D)
- Cladding loads
- PEB pressure coefficients (CP, GCp)

11.4.4 Seismic Loads

From **IS 1893 (latest)**:

- Base shear
- Mode shapes
- Participation factors
- Floor forces
- Accidental eccentricity
- Storey drifts
- Time history support (future)

11.4.5 Special Loads

- Crane vertical load
- Crane surge/longitudinal load
- Monorail load
- Solar panel load
- Conveyer loads
- Vehicular loads
- Impact loads
- Blast loads (future)
- Snow loads
- Temperature loads
- Settlement loads
- Construction stage loads

11.4.6 Geotechnical Loads

For retaining/re walls:

- Active earth pressure
- Passive earth pressure
- Surcharge
- Water table effect
- Seismic increment (Mononobe–Okabe)

For landfill:

- Waste load
- Leachate head load
- Side slope pressures
- Operational load (dozers, compactors)

11.5 Load Combinations

11.5.1 Standards Supported

- IS 875 + IS 1893 load combinations
- IS 456/800 limit state combinations
- Eurocode combinations
- ASCE 7 combinations
- ACI/AISC combinations

11.5.2 Load Combination Generator

The system must:

1. Detect all load cases generated
2. Apply appropriate combination rules per selected code
3. Generate:
    - Ultimate limit state (ULS)
    - Serviceability limit state (SLS)
    - Fatigue combinations (future)
    - Fire load combinations (future)
4. Allow **custom combinations**:
    - Add, delete, override factors
    - Create "design-only" combinations
    - Create "serviceability-only" combinations

11.6 Load Visualization (3D Interface)

The Engineer must be able to:

- Turn ON/OFF individual load cases
- See wind pressure distributions on surfaces
- Visualize seismic mass points
- Show live load intensity distribution
- Show soil pressure diagrams
- Show landfill side pressure slices

Visual Layers:

- 3D arrows
- Surface pressure maps (color contours)
- Vector display for seismic forces
- Animated mode shapes (future)
- Animated load cases (future)

11.7 Integration with the 3D Modeller

The Load Engine must interface with:

- **3D geometry (Kratos-based)**
- **Design Engine** for forces in each member
- **Drawing Engine** for marking load intensity
- **Estimation Engine** (SSR-based) for including special loads

- **AI Assistant** (AISTUDIO) for auto explanation of loads

11.8 Kratos Multiphysics Integration

Loads must be translated into standard formats:

Node-based Loads

- Nodal forces
- Nodal displacements
- Nodal moments

Element-based Loads

- Distributed loads
- Pressure loads
- Temperature loads
- Gravity loads

Surface & Volume Loads (for continuum models)

- Pressure
- Self-weight
- Thermal loads
- Hydrostatic pressure
- Soil side pressure

Kratos JSON Schema

All loads must be exported and imported through a unified JSON schema:

```
{
 "load_cases": [
  {
   "name": "Wind +X",
   "type": "wind",
   "entity": "surface",
   "data": { "...": "..." }
  }
 ]
}
```

11.9 AI-Driven Load Assistance

The system must support:

AI Load Advisor

User writes:

“G+2 house in Hyderabad with 3m storey height”

AI must:

- Identify applicable codes
- Infer typical live load
- Infer wind/seismic automatically
- Ask user only the missing details
- Fill tables automatically

AI Load Explainer

Explain the load:

- Why it was chosen
- Which code clause

- What alternatives exist

AI Load Validator

Red flags:

- Underspecified seismic mass
- Incorrect wind category
- Missing live load reduction
- Incorrect soil pressure application

11.10 Extensibility

Future additions:

- Bridge loads
- Dam loads
- Industrial furnace loads
- Fire load calculations
- Blast-resistant design loads
- Vehicular impact loads
- Dynamic time history (via Kratos StructuralMechanicsApplication)

11.11 Outputs

Mandatory Outputs:

- Load summary table
- Per-floor load distribution
- Pressure maps
- JSON export to solver
- CSV export
- Load diagram (2D & 3D)

Optional:

- Annotated PDF report
- Load comparison scenarios
- Sensitivity analysis (“What if wind = 50 m/s?”)

PART 11 COMPLETED

📘 PART 12 – TEMPLATE SYSTEM FOR ALL STRUCTURE TYPES (ARCHITECTURE & SPECIFICATIONS)

(No confirmations needed. I will move directly to Part 13 after this unless you stop me.)

PART 12 — TEMPLATE SYSTEM FOR ALL STRUCTURE TYPES

A template system is the backbone of a multi-disciplinary structural–architectural–estimation–project-management platform.

This section defines the **template engine**, **template file formats**, **template inheritance rules**, **dynamic user-selection logic**, and **AI-assisted customization**.

The module ensures:

- Any structure type (PEB, RCC, Steel Building, Industrial Shed, Retaining Wall, RE Wall, Landfill, Road, Culvert, Bridge, Compound Wall, Drain, STP, ETP, etc.)
- Can be loaded instantly from pre-defined templates
- And customized through a **STAAD-like 3D model interface**
- With **automatic generation of drawings, design inputs, loads, checks, and costing**

12.1 Purpose of the Template System

The template system shall enable:

1. **Rapid Concept-to-Design Conversion**
   A user can start from:
   - A stored building/industrial layout
   - A hand sketch (user uploads)
   - A referenced plan
   - AI-generated layout
   - Library templates
2. **Consistent modelling standards**
   - All structure templates follow unified naming:
     Node, Member, Element, Surface, Solid, Load, Combination.
3. **Automated expansion into:**
   - Loads (IS 875 / IS 1893 / IRC / CPHEEO etc.)
   - Structural elements
   - Drawing sheets
   - Material take-offs
   - Estimation sheets
   - BOQs
   - Construction workflows & schedules
4. **Selectable subscription modules**
   Each template links to the paid module list:
   - Concept layout → Architectural → Structural → MEP → Estimation → PMC
5. **Template evolution**
   Each template updates over time using:
   - AI-based parameter tuning
   - Real project data
   - Best-practice engineering logic

12.2 Template Architecture (High-Level)

Templates are stored in structured JSON/YAML-like formats:

```
/templates/
  /rcc_building/
    g+1_residential.yaml
    g+2_small_office.yaml
    g+4_apartment_core-wall.yaml
  /peb/
    industrial_shed_small.yaml
    warehouse_large.yaml
  /retaining_wall/
    5m_cantilever_wall.yaml
    10m_counterfort_wall.yaml
  /landfill/
    cell_50x40_basic.yaml
    cell_200x150_bermed.yaml
```

Each template has 7 core blocks:

1. **Metadata**
2. **Geometry Definition**
3. **Load Definition**
4. **Design Parameters**
5. **Drawing Requirements**
6. **Estimation Mapping**
7. **Execution Workflow Mapping**

12.3 Template File Structure (Standard Format)

12.3.1 Metadata

```
meta:
  name: "G+2 RCC Small Office"
  version: 1.1
  code_compliance:
    structural: IS 456, IS 875, IS 1893
    geotech: IS 6403
    fire: NBC 2016
  category: "RCC Building"
  tags: ["RCC", "G+2", "commercial"]
```

12.3.2 Geometry Definition

Defines *default* building dimensions:

```
geometry:
  plan:
    length_m: 18
    width_m: 12
  grid:
    x: [0, 6, 12, 18]
    y: [0, 6, 12]
  storeys:
    count: 3
```

```
    heights_m: [3, 3, 3]
  elements:
    columns:
      size: "300x500"
      material: M30
    beams:
      size: "300x450"
      material: M30
    slabs:
      thickness_mm: 150
```

12.3.3 Load Definition

Mapping of template → load patterns.


```
loads:
  dead_loads:
    slab_DL_kN_m2: 2.5
    finishes_kN_m2: 1.0
  live_loads:
    residential_LL_kN_m2: 2.0
  wind:
    zone: 2
    speed_m_s: 39
  seismic:
    zone: II
    soil_type: "Medium"
```

12.3.4 Design Parameters


```
design:
  concrete_grade: M30
  steel_grade: Fe500
  ductile_design: true
  importance_factor: 1.0
  R_factor: 5
```

12.3.5 Drawing Requirements


```
drawings:
  sheets:
    - name: "Architectural – Plans (All Floors)"
    - name: "Structural – Typical Floor Beam Layout"
    - name: "Structural – Column Schedule"
    - name: "Footing Layout"
    - name: "Staircase Section"
    - name: "3D Model Overview"
  export:
    formats: ["PDF", "DWG", "DXF", "3D IFC"]
```

12.3.6 Estimation Mapping

estimation:
  items:
    cement_m3_factor: 0.14
    steel_kg_m3_factor: 100
    shuttering_m2_factor: 5.5
  ssr_rate_region: "Telangana"
  adjust_factors:
    location_factor: 1.0
    contractor_margin: 5%

12.3.7 Execution Workflow Mapping

workflow:
  stages:
    - "Conceptual Layout"
    - "Architectural Design"
    - "Structural Modelling"
    - "Design & Checks"
    - "Detailing"
    - "Estimation & BOQ"
    - "Project Scheduling"
    - "Construction"
    - "Handover"

12.4 Template Inheritance (Very Important)

The system will support **parent–child** template structures.

Example

parent: rcc_building/base.yaml
child: rcc_building/g+2_small_office.yaml

Inheritance rule:

- Child overrides only what is needed
- Base file holds defaults

This enables:

- Faster updates
- Consistency
- Uniformity across thousands of templates

12.5 Template-Based Model Generation (like STAAD)

Once a template loads:

System auto-generates:

- Nodes
- Members
- Surfaces
- Supports
- Load patterns
- Load combinations
- Material assignments

User can then modify:

- Grid spacing
- Nodes
- Beams/columns
- Slabs
- Loads

3D Visualization Features

- Orbit
- Pan
- Zoom
- Select element
- Modify geometry
- Cut sections (XZ, YZ, XY)
- Member property view
- Highlight overstressed elements

After modifications
User clicks **"Proceed to Design →"**
and the design engine (Part 9) takes over.

12.6 AI-Assisted Template Recommendation
When user inputs:
"Construction of G+2 building in Hyderabad"
The system triggers:
AI Step 1 — Parse requirement
structure_type = RCC
levels = 3
location = Hyderabad
AI Step 2 — Query template library
Finds all "RCC G+2" models.
AI Step 3 — Auto-suggest top 5 layouts

- Typical residential 3BHK grid
- Typical office layout
- Narrow plot plan
- Wide plot plan
- Parking + 2 floors

AI Step 4 — Convert selected layout → structural template
AI Step 5 — Allow user to upload hand sketch
AI converts sketch to:

- Grid lines
- Rooms
- Wall positions
- Openings
- Column markers

12.7 Template Pricing Model
Every template links to monetization:

- Architectural Plan – ₹ X
- Structural Plan – ₹ Y

- Detailing – ₹ Z
- Estimation – ₹ A
- Project Management – ₹ B
- 3D Visualisation – ₹ C

User pays only for the activated modules.

12.8 Template Versioning & Updates

Each template has:

- Version number
- Created by
- Last revised by
- Changelog

Engine always loads latest stable version unless user selects otherwise.

12.9 Template–Code Compliance Integration

Each template ties into code-specific rule-sets:

- IS 456 / IS 800 / IS 875 / IS 1893
- ACI 318 / AISC / Eurocode 2/3
- CPHEEO / MoRTH / IRC

Template → Loads → Design Checks → Drawings → Estimation.
Zero manual intervention once the model is finalized.

12.10 Summary of Template System

This part ensures:

- **Instant model creation** from any input
- **3D modification like STAAD**
- **Consistent engineering workflows**
- **Design-ready structural models**
- **Seamless integration with Parts 9–21**
- **User monetization modularity**
- **AI-enhanced layout selection**
- **Universal applicability to all structure types**

📘 PART 13 — DRAWING & DETAILING ENGINE (DDE)

(Architectural, Structural, Services, Reinforcement, Steel GA & Fabrication Drawings, and 3D Visual Documentation)

This section defines how your application will automatically generate all drawings required for submission, construction, fabrication, and approvals — based on user inputs, AI-interpreted layouts, and engineering design outputs.

13.1 PURPOSE OF THE DRAWING & DETAILING ENGINE

The DDE is responsible for:

A. Producing machine-readable, editable drawings

- Architectural plans, sections, elevations
- Structural framing plans & schedules
- RCC reinforcement drawings
- Steel GA drawings
- Steel fabrication drawings
- Staircase/Water tank/Retaining wall/Plinth beam/Compound wall drawings
- MEP/Services routing (optional phase)

B. Auto-generating drawings from the 3D model

Includes:

- Automatic views (plan, section, elevation, isometric)
- Annotated dimensions
- Member IDs, marks, bar bending schedules
- Optimization-aware member labeling
- Clash-free detailing based on BIM/3D geometry

C. Multi-format delivery

Your app must export in:

- **DWG**
- **DXF**
- **PDF**
- **SVG**
- **3D glTF viewer**
- **IFC model export (for interoperability)**

D. User-controlled editing

Engineers can:

- View drawings in browser
- Select layers (grids, beams, columns, doors)
- Modify dimensions
- Move nodes/members
- Add/delete members
- Export updated geometry to analysis

Exactly like STAAD + Tekla hybrid functionality.

13.2 INPUTS TO DRAWING ENGINE

1. **User Inputs**
    - Type of building (G+2, industrial shed, RCC framed structure, warehouse, etc.)
    - User's chosen layout or uploaded hand sketch/plan

    - AI-normalized geometry (cleaned by the Layout Understanding Engine)
    - User modifications in the 3D viewer
2. **Analysis Engine Output**
    - Final coordinates of nodes
    - Member cross-sections
    - Reinforcement data
    - Connections & support details
    - Geotechnical & foundation parameters
3. **Design Engine Output**
    - Bar lengths, lap lengths, cover, spacing
    - Plate thicknesses, weld sizes
    - Bolt grades & patterns
    - Reactions for footing drawings
    - Foundation sizes
    - Retaining wall dimensions
    - Staircase geometry
    - Slab reinforcement meshes

13.3 ARCHITECTURAL DRAWINGS MODULE

Generated automatically:

- Floor plan (cleaned, scaled, dimensioned)
- Door-window schedule
- Furniture layout (optional)
- Staircase plans & sections
- Toilet layout
- Electrical/lighting plan (optional future module)
- Plumbing routing (optional future module)

AI-assisted Auto Cleanup

- Converts user hand sketch → vectorized DWG
- Snaps walls to gridlines
- Detects rooms, openings
- Suggests optimal column grids

13.4 STRUCTURAL GA DRAWINGS

For RCC Buildings

- Structural framing plan for each level
- Column schedule
- Beam layout with sizes
- Slab thickness & reinforcement
- Shear wall plans
- Staircase details
- Ramp details

For Steel Buildings (PEB/Industrial)

- GA plan (roof + floor)
- Elevations & sections
- Member marks (RAFTER, COLUMN, TIE, BRACE)
- Purlin/girt alignment

- Portal frame sections
- Crane beam layout (if applicable)

13.5 RCC REINFORCEMENT DETAILING SYSTEM

Produces:

- Beam reinforcement drawings (L- and T-beams)
- Column reinforcement + lap elevation
- Slab top/bottom bars
- Footing reinforcement (isolated, combined, raft)
- Retaining wall reinforcement
- Water tank reinforcement
- Staircase reinforcement

Bar Bending Schedule (BBS)

Generated in:

- Excel
- PDF
- CSV

Includes:

- Bar mark
- Diameter
- Number of bars
- Length
- Shape code
- Mass per bar
- Total weight

13.6 STEEL FABRICATION DRAWINGS MODULE

Produces:

- Individual fabrication drawings for each steel member
- Flange dimensions
- Web plates
- Stiffeners
- Connection plates
- Base plates
- Hole charts
- Welding symbols
- Bolt lists
- Material list per member

Automated numbering

- Prefixes based on type (C-, R-, B-)
- Single-part & assembly drawings
- Tekla-style numbering

13.7 CONNECTION DETAILING MODULE

For Steel

- Moment connections
- Shear connections

- End-plate connections
- Base plates
- Anchor bolt templates
- Bracing connections
- Splice connections

For RCC

- Beam-column joints
- Shear wall boundary zones
- Footing-column joints
- Stair waist/slab junctions

All with:

- IS 800
- IS 456
- ACI / AISC options (if user chooses international modes)

13.8 3D VISUALIZATION AND INTERACTIVE EDITOR

Your application must provide a **Tekla/STAAD-like interface**.

Capabilities:

- **3D view of entire structure**
- Pan/rotate/zoom
- Perspective & orthographic modes
- Section-boxing & clipping planes
- Color-coded members (by type, load, stress, group)
- Member-by-member highlight
- Node coordinates
- Snap editing
- Modify geometry visually
- Insert/delete:
    - Columns
    - Beams
    - Braces
    - Slabs
    - Walls
- Instantly update:
    - Analysis model
    - Drawings
    - Quantities

Technology Options

- **Three.js** (browser)
- **PyVista** (desktop)
- **OpenCascade/OCC** (for 2D drawings + 3D CAD kernel)
- **IFC.js** (for BIM interoperability)

13.9 DRAWING GENERATION WORKFLOW

1. **User chooses building type**
2. **AI collects all design inputs**
3. **Layout Understanding Engine digitalizes plan**

4. **3D model is generated (editable)**
5. **Engineer modifies geometry in 3D viewer**
6. **Analysis Engine solves model**
7. **Design & code checks performed**
8. **Detailed drawings automatically generated**
9. **User exports final package (DWG + PDF + quantities)**

Each step can be charged separately in your SaaS plan.

13.10 EXPORT FORMATS

Format	Purpose
DWG	CAD consultants, architects
DXF	Universal CAD compatibility
PDF	Submission to authorities
SVG	Web-based editing
IFC	BIM interoperability
glTF	3D web visualization
OBJ/STL	Fabrication or CNC evaluation

13.11 PARAMETRIC DRAWING ENGINE

All drawings are generated through a parametric engine:
Example:

- Beam depth changes → All connected sections update
- Column position moved → Reinforcement BBS updates
- Bay spacing changes → Plan + elevations update automatically

Just like **Revit + Tekla**, but lighter and AI-assisted.

13.12 DETAILED DRAWING LIBRARIES

Stored templates for:

- Residential (G+1/G+2/G+3)
- Commercial
- Industrial
- PEB
- Warehouses
- Foundations
- Retaining walls
- Water tanks
- Staircases
- Compound walls
- Landfills (sections & details)

AI selects the right templates based on user's scenario.

END OF PART 13

📘 PART 14 — REPORTING ENGINE (Dashboards, KPIs, MIS)

Deep-dive, enterprise-grade PRD section

This part defines how the system presents results, insights, dashboards, KPIs, MIS reports, downloadable design summaries, and audit-ready documentation for clients, authorities, engineering consultants, and internal teams.

This Reporting Engine is **unified** across:

- Architectural workflows
- Structural analysis & design (PEB, RCC, warehouses, industrial sheds, RE walls, retaining walls, landfills)
- MEP (future phases)
- Estimation & Bill of Quantities
- Project execution, monitoring, and closure

It ensures the software becomes a **full project lifecycle reporting suite** similar to STAAD + ETABS + Primavera + Cost Control dashboards—built into one system.

14.1 Goals of the Reporting Engine

14.1.1 Provide instant, transparent engineering output

- Structural reports automatically generated after each design iteration.
- Clearly formatted summaries of:
  - Loads
  - Reactions
  - Member forces
  - Deflections
  - Foundation loads
  - Stability checks
  - Code compliance
  - Soil & slope safety factors (landfills, RE walls, retaining walls)

14.1.2 Provide dashboards for non-technical users

- Developers
- Business owners
- Architects
- Cost estimators
- Project managers

A simple interface showing:

- Cost summary
- Material quantities
- Area calculations
- Construction sequence
- Risk flags

14.1.3 Provide audit-ready documentation

- For approvals:
  - GHMC
  - HMDA
  - GMR
  - L&T
  - BHEL
  - Credited third-party proof checkers

- For clients requiring:
    - Design basis report
    - Loading assumptions
    - Structural analysis reports
    - Safety factors
    - Construction drawings
    - MTO + BOQ

14.2 Reporting Engine – Key Capabilities

The engine supports **multi-domain reporting**:

Domain	Outputs Generated
Architectural	Floor plans, 3D renders, space utilization, FAR, setbacks
Structural	Analysis results, design checks, member sizes, connection details, foundation design
Geotechnical	Bearing capacity, settlements, slope stability, RE wall reinforcement tables
Landfill	Volume, phases, slopes, berms, bunds, stability
MEP	Loads, basic demand calculations (future)
Estimation	Material quantities, cost summary, SSR-based cost, rate analysis
Project Execution	Progress dashboards, KPIs, earned value, delays, cost overruns

The reporting engine is **modular**, meaning new structure types can be added without redesigning the system.

14.3 Report Types

14.3.1 Engineering Reports

Mandatory standard reports:

- Design Basis Report (DBR)
- Structural Analysis Report
- Load Summary Report
- Foundation & Soil Report
- Steel Connection Report
- RCC Design Summary (columns, beams, slabs, footings)
- Wind Load Report (IS 875)
- Seismic Load Report (IS 1893)
- PEB Frame & Member Report
- Detailed Plate/Shell Results (FEniCSx / Kratos)

Each report category automatically includes:

- Problem definition
- Methodology
- Assumptions
- Input values
- Governing code clauses
- Safety factors
- Stepwise calculations (auto-generated)

- Figures and diagrams (auto-embedded)
- Tables from Dashboard → printable format
- Compliance summary
  (Pass/Fail tagging with explanations)

Reports can be exported as:

- PDF (default)
- DOCX (editable)
- Excel (tabular results)
- JSON (machine readable)
- HTML (interactive visualization for authorities)

14.3.2 Architectural Reports

- Auto-generated floor plans (PDF)
- 3D visualizations (embedded via WebGL)
- Space utilization report
- Room schedule
- Door/window schedule
- Staircase & lift details
- Parking plan summary
- Setback compliance (city-by-city rules)
- Walkthrough link or 3D viewer (internal)

14.3.3 Estimation Reports

Estimation modules output:

- Quantity takeoff sheets
- Activity-wise material list
- Block-wise or floor-wise BOQ
- SSR-based cost summary
- Contractor quotation comparison
- Escalation calculations
- Procurement plan
- Cash flow & payment schedule

Export formats:

- Excel (native)
- PDF (for submission)
- CSV (for SAP/Tally/ERP imports)

14.3.4 Project Execution Dashboards

KPIs included:

- Planned vs actual progress
- Cost incurred vs BOQ
- Delays, risks, alerts
- Resource usage
- Material movement
- Daily progress photos
- Quality checklists
- Compliance status (HSE, QAQC)

- RA bill evaluation
- Cash flow projections
- S-curve graphs (baseline vs current)

Dashboards support:

- Drill-down to activities
- Filters (contractor/block/floor)
- RFI tracking
- NCR (Non-conformance) alerts
- Snag list & closure status

14.4 3D Visualization Reporting

Your requirement:

"Engineer should see/visualize/modify in 3D like STAAD."

The system supports:

14.4.1 3D Structural Model Viewing

- Beam/column frames
- Slabs, walls, footings
- Soil and terrain
- Landfill benches/slope
- RE wall reinforcement views
- Animated deflected shape
- Stress contours (min/max)
- Mode shapes (from Kratos/FEniCSx FEM)

14.4.2 Model Editing

- Move/modify nodes
- Add/remove members
- Change section sizes
- Add openings in slabs/walls
- Modify loads interactively
- Override generated dimensions

14.4.3 Cross Sections

- Member-by-member section view (like STAAD Physical Model)
- Slice plane (XY, YZ, ZX)
- Cut-through building (for debugging geometry)
- Exploded view of building layers

14.4.4 Export

- DXF
- IFC
- STL/OBJ for 3D
- STAAD .std (for interoperability)
- ETABS .e2k (planned)
- Tekla Structures .tse (planned)

14.5 MIS (Management Information System)

MIS dashboards for management include:

14.5.1 Executive Summary Dashboard

- Overall project status

- Cost at completion
- Time at completion
- Risks & issues
- Profitability
- Variation orders
- Approvals pending
- Material consumption trends

14.5.2 Engineering Status Dashboard

- Drawings completed vs pending
- Designs completed vs pending
- Revisions issued
- QC comments pending
- Vendor/consultant review status

14.5.3 Procurement MIS

- Material requirement plan
- Vendor allocation
- Purchase order summary
- Delivery tracking
- Cost deviation
- Lead time risk alerts

14.5.4 HSE & QA/QC MIS

- Inspection checklists
- NCR status
- Audit history
- Safety observations
- Compliance score

14.6 Automated Report Generation Pipeline

14.6.1 Trigger Points

Reports are auto-generated when:

- User completes architectural layout
- Structural model is validated
- FEM analysis completes (Kratos/FEniCSx)
- Design checks finish
- Estimation is approved
- Project status changes
- QC workflow demands revision
- Client requests updated drawings

14.6.2 Version Control

Every report version is archived with:

- Timestamp
- Hash ID
- Differences from previous revision
- Engineer name
- AI/Manual/Hybrid origin tag

14.6.3 Authority-Ready Packaging

A compressed submission folder is generated:

/Authority-Package/
DBR.pdf
Structural Analysis Report.pdf
Architectural Drawings.pdf
GFC Drawings.pdf
BOQ.xlsx
Stability Reports.pdf
Soil Report.pdf
Environmental Clearances
Calculation Sheets.pdf

14.7 Data Model for Reports

Core entities:

Report
id
type
related_structure
files
created_by
created_at
version
is_official
authority_format

The report system works independently of:

- Geometry engine
- Design engine
- Estimation engine
- Project engine

This allows adding new report types without breaking old ones.

14.8 Integration With Other Modules

Architecture → Reporting

- Floor plans, 3D blocks, area reports

Structural → Reporting

- FEM results
- Design checks
- Member tables
- Connection design

Estimation → Reporting

- BOQ
- SSR cost
- Time/cost curves

Project Execution → Reporting

- Daily logs
- Actual progress
- Material tracking

AI Engine → Reports

- Natural language explanation of:
    - Why a member failed
    - What design changes to make
    - What to submit to GHMC
    - How to reduce cost

14.9 Future Extensions

- BIM 360 integration
- Auto generation of tender documents
- Digital twins
- Real-time IoT-based monitoring dashboards
- AI-generated GFC drawings
- Voice-driven MIS access

📘 PART 15 — PROJECT EXECUTION & RESOURCE PLANNING MODULE

(Fully aligned to your overall vision: design → drawings → estimation → execution → closure)

PART 15 — PROJECT EXECUTION & RESOURCE PLANNING

15.1 Purpose & Position in the Workflow

After the user finalizes:

- Building type (e.g., G+2 RCC in Hyderabad)
- AI-guided requirement gathering
- Layout selection or uploaded sketch
- Auto-generated architectural + structural model
- Structural analysis + design
- All drawings + BOQs + Estimates

... the system transitions into **Project Execution Mode**, enabling the engineer or client to plan, track, modify, and manage the entire project lifecycle.

This module supports:

- Planning
- Scheduling
- Resource allocation
- Material management
- Contractor/vendor engagement
- Financial control
- Progress monitoring
- Quality + Safety
- Documentation + compliance
- Closure

It effectively replaces multiple existing industry tools with a unified platform.

15.2 Key Objectives

1. Convert the **designed model + BOQ + rate analysis** into an actionable execution plan.
2. Auto-derive **task lists, dependencies, crews, materials, timelines** from the BIM/structural model.
3. Provide a **visual 3D interface** synchronised with Gantt, cost, resources.
4. Support **manual edits**: durations, crew sizes, subcontract mode, procurement mode.
5. Provide near real-time **progress tracking** and **delays prediction** using AI.
6. Provide **mobile-friendly field modules** (site engineer app).
7. Automatically update **cashflows, cost-to-complete, material forecasts**, etc.
8. Allow export to standard formats:
    - MS Project (XML)
    - Primavera P6 (XER)
    - Excel
    - PDF
9. Provide templates for:
    - Residential building
    - Commercial building

- Industrial building
- Landfill
- Roads, retaining walls, RE walls
- PEB sheds
- Steel buildings

10. Enable **multi-project dashboards**.

15.3 Core Features

15.3.1 Auto-Generated Work Breakdown Structure (WBS)

Pulled directly from the model (Kratos-based) + drawings:

- Site clearing
- Excavation
- Footings
- PCC / Reinforcement / Shuttering / Concrete
- Columns, beams, slabs
- Retaining walls / compound walls
- Staircases
- Masonry, plastering
- Flooring, doors/windows
- Waterproofing
- Services (Plumbing, Electrical, HVAC)
- Painting
- Final finishing
- Testing & commissioning
- Handover

The WBS is **editable** so the engineer can:

- Add/Delete tasks
- Re-sequence
- Create custom activities
- Group activities
- Assign codes

15.3.2 Scheduling (Gantt + 3D Integration)

The system auto-assigns durations based on:

- BOQ quantities
- Productivity of crews
- Local labour availability
- Weather profile (optional)
- Equipment availability

Engineer can:

- Override durations
- Set dependencies (FS/SS/FF/SF)
- Add milestones
- Create multiple schedule baselines
- Track actuals vs planned

3D Gantt (4D BIM):

The structural model animates construction sequence visually.

15.3.3 Resource Planning
Automatically detects resources:
- Labour categories (Mason, Carpenter, Bar bender, Helpers)
- Machinery (excavators, cranes, concrete mixers)
- Materials (cement, sand, aggregate, steel, bricks)
- Shuttering materials, scaffolding
- Concrete pumps, curing equipment

System calculates:
- Daily labour requirement
- Machinery hours
- Material delivery windows
- Stock levels and reorder needs

Resource conflicts are highlighted.

15.3.4 Procurement Planning
Based on WBS, BOQ, and vendor lists:
- Auto-generate purchase packages
- RFQ creation
- Comparative statements (auto)
- Vendor onboarding workflow
- Delivery tracking
- QC checks at gate
- Material testing logs

Procurement delays automatically update schedule impact.

15.3.5 Cost Control
Supports:
- Budget vs actual
- CV, SV, CPI, SPI
- Cost-to-complete forecasting
- AI-driven delay impact cost estimation
- Contract-wise financial tracking
- Bill checking engine
- Subcontractor running bills
- Material reconciliation

Each structural element has:
- Material cost
- Labour cost
- Equipment cost
- Overheads
- Margin

Cashflow automatically updates in Gantt.

15.3.6 Site Management (Daily Operations)
15.3.6.1 Daily Reports
- Workforce deployed

- Work done
- Concrete cast
- Inspections
- Photos
- Hindrances
- Safety observations

15.3.6.2 Material Tracking

- Issue slips
- GRN
- Consumption
- Reconciliation

15.3.6.3 Machinery Operations

- Fuel logs
- Maintenance
- Utilization reports

15.3.7 Quality Management (QA/QC)

Checklists auto-generated and linked to tasks:

- Excavation checks
- Foundation reinforcement checks
- Concrete cube results
- Weld tests (for PEB/Steel)
- Waterproofing tests
- Plumbing tests

Site engineers can upload images, videos, documents.
All QA logs are linked to structural elements.

15.3.8 Safety (HSE)

- JSA (Job Safety Analysis)
- Permit-to-work
- Safety induction
- PPE tracking
- Incident reporting
- Daily toolbox talk
- Risk assessments

Dashboard of:

- Lost Time Injury
- Near misses
- Safety compliance score

15.3.9 Drawings & Document Management

Supports versioning of:

- Models
- Architectural drawings
- Structural drawings
- Fabrication drawings
- Reports

- Estimation sheets
- QA/QC documents
- HSE documents

User can:

- Compare versions
- Automatic revision clouds
- Download PDFs
- Archive old versions

15.3.10 AI Engine (Project Execution)

AI performs:

- Delay prediction
- Cost escalation prediction
- Workflow optimization
- Suggesting alternative materials/vendors
- Productivity analytics
- Identifying bottlenecks
- Generating daily report drafts
- Auto-detecting risks based on history

15.3.11 Multi-Project Dashboard

Across regions/business units:

- Project start/end
- Cost vs budget
- SPI/CPI
- Labour strength
- Safety performance
- Material consumption
- Monthly reports auto-generated

15.4 User Roles

- **Client / Owner**
- **Project Manager**
- **Planning Engineer**
- **Site Engineer**
- **QA/QC Engineer**
- **Storekeeper**
- **Procurement Manager**
- **Finance Controller**
- **Management / HO**

Each role has restricted access to features.

15.5 Outputs

- Full Gantt schedule
- 4D BIM animations
- Labour/machinery histograms
- Material requirement plan

- Procurement plan
- Cashflow
- DPRs
- QA logs
- HSE logs
- Billing summaries
- Final project completion report

15.6 Integration Points

- **Part 8 – 3D Modelling Interface** (Bi-directional)
- **Part 9 – Structural Design Engine**
- **Part 10 – Drawing Generator**
- **Part 11 – Estimation**
- **Part 12 – Procurement Engine**
- **Part 13 – QA/QC & HSE Engine**
- **Part 14 – Reporting Engine**

📘 **PART 16 — PROCUREMENT MODULE (Deep-Dive PRD)**
(Fully integrated with Architecture, Design, Estimation, Execution & Closure)
(No confirmations needed. I will automatically proceed to the next part after this.)

---

16. PROCUREMENT MODULE (DETAILED PRD)

16.1 Purpose of the Module

The Procurement Module is responsible for converting **design outputs, BOQs, specifications, and project schedules** into a structured process for **material, labour, equipment, subcontracting, services, and vendor management**.

It serves as the **central nervous system** that connects:

- Architectural plans
- Structural designs
- MEP designs
- Bill of Quantities
- Estimation & Costing
- Project schedule (Gantt & WBS)
- Vendors & suppliers
- Approvals workflows
- Inventory & site stores
- Accounts Payable & Billing modules

This module must work for:

✓ Housing (G+1 to G+20)

✓ Industrial Sheds, Warehouses

✓ RCC/Steel/Composite buildings

✓ Retaining Walls, RE Walls, Landfills, Roads

✓ Infrastructure Projects

✓ EPC + Item-Rate + Turnkey contracts

It supports **micro & macro-level procurement**, from a single bag of cement to a full subcontract package.

16.2 Core Capabilities & Objectives

16.2.1 Functional Goals

The system must:

1. **Auto-read BOQs** from design outputs.
2. **Break BOQs → Procurement Packages** (materials/services).
3. **Auto-generate PRs (Purchase Requisitions)** based on WBS, BOQ, schedule.
4. **Vendor Management** (prequalification, rating, performance).
5. **Tendering engine** (RFQ, technical & commercial comparisons).
6. **Purchase Order (PO) creation** with tax mappings.
7. **Delivery tracking**, GRN, QC, short receipt, rejections.
8. **Rate contracts**, Framework Agreements & Multi-year contracts.
9. **Analytics**: consumption, wastage, variances, cost curves.
10. **Integration** with Inventory, Estimation, Billing, Finance.

16.3 Pre-Procurement Data Inputs

Mandatory Inputs:

- Finalized Architectural/Structural/Services drawings
- Structural BOQs (Auto-generated from design)
- Material specs from design module
- Item-wise technical specifications
- SSR/DSR/Client Schedule
- Schedule Baseline (WBS → Program → Quantity Phasing)
- Vendor list (if available)
- Budget (from estimation module)

Optional Inputs:

- Contractor's master material library
- Preferred brands list
- Client-approved vendor list
- Historical purchase data

16.4 Procurement Process Workflow

16.4.1 Design → BOQ → Procurement linkage

1. **Design module outputs**:
   Columns, beams, slabs, footings, stairs, walls, tanks, retaining walls, RE walls, steel sheds, trusses, purlins, rafters, etc.
2. **BOQ module summarises**:
   - Concrete
   - Steel
   - Shuttering
   - Formwork
   - Backfilling
   - Structural steel
   - Flooring
   - Doors/windows
   - Waterproofing
   - Electrical/Plumbing, etc.
3. **Procurement automatically groups**:
   - Material procurement packages
   - Labour packages
   - Subcontracting packages
   - MEP packages
   - Heavy equipment & tools

16.4.2 Purchase Request (PR) Creation

PR may be auto-triggered based on:

- WBS-driven schedule
- Consumption forecasts
- Safety stock limits
- Design revision changes
- Material run-out alerts at site

Each PR includes:

- Detailed line items

- Specs
- Approved brands
- Quantity thresholds
- Delivery milestones
- Priority flags
- Required delivery date
- Quality requirements

16.4.3 Vendor Management

Vendor Data Includes:

- PAN, GST, CIN, Bank details
- Service categories (material, labour, etc.)
- Regions of operation
- Ratings (past performance, quality, delivery)
- Price competitiveness score
- QA/QC compliance records
- Safety compliance

Vendor Actions Supported:

- Vendor onboarding (forms, document upload)
- Vendor approval workflow
- Vendor blacklisting
- Performance reports
- Contract closure evaluation

16.4.4 Tender / RFQ Management

System auto-generates:

- RFQ PDFs
- Technical specifications
- Drawings attachment bundle
- Delivery schedule
- EMD/Bid security terms
- Payment terms

Tender Modes:

- Material tender
- Labour tender
- Subcontract tender
- Rate contract tender
- Framework agreement

Bidder Portal (Optional):

Vendors can log in to view RFQs, upload offers, and sign digitally.

16.4.5 Technical & Commercial Comparison (TCC)

Automated TCC Engine

System compares:

✓ Rates

✓ Taxes

✓ Delivery

✓ Payment terms
✓ Compliance with specs
✓ Warranty
✓ Brand approvals
✓ Freight & loading/unloading
✓ Inspection & testing charges
Outcome:
Recommended vendor with weighted scoring.

16.4.6 Purchase Order (PO) Management
PO creation includes:

- Line item pricing
- Quantity & tolerance
- Delivery conditions
- Freight applicability
- Tax structure (CGST/SGST/IGST)
- QC conditions
- Payment milestones
- LD (Liquidated damages) clauses
- Warranty terms
- Safety obligations
- Arbitration clauses

PO Approvals follow DOA (Delegation of Authority).

16.4.7 Delivery, GRN & QC
At delivery:

- Material entry at gate
- QC verification (tests, certificates)
- Short/Excess receipt
- Rejection slip
- Stock updation
- Auto-notification to Accounts Payable

Field Tests Integrated:

- Cement Fineness
- Rebar Yield Strength
- Aggregates grading
- Cube testing
- Soil density tests
- Bitumen tests
- Structural steel UT/MPT/NDT tests

Each test stored as PDF + values.

16.4.8 Rate Contract & Annual Maintenance
System supports long-term contracts for:

- Cement
- Steel

- Aggregates
- Ready-mix
- Labour gangs
- Mechanical equipment
- Tools & PPE

Features:

- Auto-refresh rates
- Indexation (WPI/CPA)
- Quantity tracking
- Vendor compliance monitoring

16.4.9 Inventory Integration

Procurement must connect to:

- Material In → Material Out
- Issues to contractors
- Consumption
- Wastage
- Stock levels
- ABC classification
- Reorder point alerts

This data also feeds into cost control module.

16.4.10 Project Cost Control Integration

Each PO is mapped to:

- BOQ item
- Cost center
- Activity code
- Project phase
- Budget head
- Earned value metrics

Outputs:

- Cost variance
- Procurement variance
- Material cost per sq.ft / per MT
- Budget vs Actual curves
- S-curve cost integration

16.4.11 Estimation Module Integration

Procurement pulls:

- Item rate analysis
- Approved vendor rates
- SSR mappings
- Contractor rates
- Market rate intelligence

Estimation → Procurement → Billing must be **fully reversible**.

16.4.12 Billing & Accounts Payable Integration

Procurement feeds:

- PO copy
- GRNs
- QC results
- LD calculations
- Advance deductions
- Retentions
- Tax data
- Invoice matching (3-way: PO–GRN–Invoice)

AP then releases payments based on DOA.

16.4.13 Procurement Analytics & Dashboards
Dashboards include:

- PR → PO cycle time
- Vendor performance index
- Cost trends (cement/steel/masonry/steel structure)
- Price escalation trends
- Budget vs Actual
- Material variance (consumption vs theoretical)
- Wastage % (by contractor, by BOQ item)
- PO value breakdown
- Pending deliveries
- Stock aging
- Payment due matrix

16.4.14 Workflow Configuration
System should support **drag-and-drop workflows** for:

- Approvals
- PR → RFQ → TCC → PO
- QC
- Billing
- Vendor onboarding

Admins can rearrange DOA without coding.

16.4.15 AI Assistance in Procurement
AI will:

- Predict material requirements
- Flag abnormal rates
- Suggest vendors based on past performance
- Detect anomalies in BOQ vs consumption
- Generate tender documents
- Interpret specifications
- Auto-fill comparison sheets
- Suggest negotiation strategies
- Predict delivery delays
- Help in vendor risk scoring
- Create contract drafts

16.4.16 Integration with 3D Model

The engineer must be able to:

- Select an element in 3D view
- View required materials for that element
- See procurement status (PR/PO/Delivered)
- Visualize pending vs delivered materials
- See timelines for specific elements
- Trigger PR from 3D model ("Order this beam/column")

This brings a **BIM + Procurement + Cost** fusion.

16.4.17 Security & Auditability

- Immutable audit trail for all actions
- DOA compliance
- Vendor access logs
- Tender logs
- Approval logs
- Procurement fraud detection alerts
- Document version control
- Digital signatures

16.4.18 APIs & Interoperability

Procurement module integrates with:

- BIM models (IFC)
- ERP (SAP, Oracle, Tally)
- GST portal
- E-invoicing APIs
- Barcodes & RFID
- IoT sensors for real-time stock

16.4.19 Mobile App Support

- Site material requests
- Photo-based GRN
- QC forms
- Reject/accept materials
- Vendor communication
- Delivery tracking (GPS)
- AI chat for procurement updates

16.4.20 Monetization Model

Procurement features can be sold as:

Basic Plan:

Purchase requests + vendor directory + single-user tendering

Pro Plan:

Full tender engine + TCC + PO + GRN + QC + Inventory integration

Enterprise Plan:

ERP-level integration + BIM integration + AI predictions + multi-project + multi-company

PART 16 COMPLETE

📘 PART 17 — PLUGIN SYSTEM & EXTENSIBILITY ARCHITECTURE

A core subsystem enabling modular expansion, custom code modules, third-party engineering engines, AI micro-features, and organization-specific workflows.

17.1 Overview & Purpose

The Plugin & Extension System (PES) transforms the platform from a fixed, monolithic engineering tool into a **modular, extensible, configurable, and integrable structural engineering ecosystem**.

The purpose of PES is to allow:

- Structural engineers to **add new member types**, materials, design codes, and checks.
- Software teams to integrate **external solvers** (Kratos, FEniCSx, OpenSeesPy, Abaqus, SAP2000, ETABS, STAAD).
- AI teams to add **AI-powered modular workflows** such as:
    - Image → Floor plan extraction
    - Hand sketch → Editable model
    - Voice → Structural input
    - Document ingestion → Requirements parser
- Companies to implement:
    - Internal QA rules
    - SOP-based design workflows
    - Custom report formats
    - Custom bill of quantities templates
- Universities / training programs to create:
    - Educational plug-ins
    - Validation modules
    - Workflows for learning concepts

This architecture future-proofs the platform by enabling **rapid addition of features without changing the core application**.

17.2 Architectural Principles

1. **Isolation**
   Each plugin runs in a sandboxed environment (VM or isolated container) to ensure security.
2. **Loose Coupling**
   Plugins interact with the core engine via well-defined APIs.
3. **Zero-Core Modification**
   No plugins are allowed to modify core files. All extension is via:
    - Events
    - Hooks
    - Plugin API
    - Safe scripting sandbox
4. **Version-Safe**
   Plugin compatibility is tracked:
    - App version
    - API version
    - Data schema version

5. **Safe Execution**
   Heavy plugins (solvers, AI models) run in separate containers to prevent UI slowdown.

17.3 Plugin Types Supported

17.3.1 Calculation Plugins

For engineering calculations such as:

- New structural member types
- Custom footings
- Regional design codes (e.g., Saudi building code, Eurocode)
- Load generation logic
- Custom seismic spectra

Example: A company can add **IS 2911 Pile Design** plug-in or **Staad-like steel truss design**.

17.3.2 Solver Plugins

Integration with:

- **Kratos** (preferred core engine)
- **FEniCSx**
- **OpenSeesPy**
- **Abaqus**
- **SAP2000**
- **ETABS**
- **STAAD**
- **Midas Gen/Civil**

Use cases:

- User-generated models → solver plugin → run FE → return results.
- Multiple solver plugins let users choose between:
  - Speed (Kratos)
  - Simplicity (OpenSeesPy)
  - Advanced nonlinear (Abaqus)
  - Industrial acceptance (STAAD)

17.3.3 AI Plugins

AI modules can be added as plugins:

- Architectural plan generator
- Structural system recommender
- Material optimization agent
- Document parser (BOQ, drawings, site reports)
- QA/QC assistant to check designs
- BIM → Analysis extractor
- Auto drawing generation

AI plugins run on isolated GPU-enabled containers (Antigravity/GCP/Azure).

17.3.4 UI/Visualization Plugins

Plugins that modify or extend the UI:

- Custom 3D visualizers

- Steel detailing libraries
- RC detailing styles for different countries
- Clash detection visual layer
- Template dashboards
- Advanced 3D cross-section viewer (Staad-like interface)

17.3.5 Workflow Plugins

Workflow orchestrators for:

- G+2 residential workflow
- Industrial building workflow
- Warehouse workflow
- PEB workflow
- Landfill workflow
- RE wall workflow
- Water tank workflow
- Pavement design workflow

Each workflow plugin defines:

- Steps
- Required inputs
- Dependencies
- Auto-generated outputs

17.3.6 Reporting Plugins

Custom report formats:

- Company-specific templates
- Submission-ready reports for:
  - Structural approval authorities
  - Industrial clients
  - QA/QC submissions
- BOQ formats
- Rate analysis formats
- Model comparison summaries

17.4 Plugin Execution Model

Plugins are executed via:

17.4.1 Event-driven architecture

Core emits events:

- MODEL_CREATED
- MODEL_UPDATED
- GEOMETRY_CHANGED
- LOADS_APPLIED
- SOLVER_RUN
- DESIGN_READY
- DRAWINGS_READY
- ESTIMATION_READY

Plugins subscribe and process.

17.4.2 Hook-based Architecture

Hooks available:

- **before_model_build()**
- **after_model_build()**
- **before_solver_run()**
- **after_solver_run()**
- **before_design_check()**
- **after_design_check()**
- **before_drawing_generate()**
- **after_drawing_generate()**

17.4.3 Command-based Plugins

Plugins can inject commands:

- /generate_dxf
- /optimize
- /check_code
- /export_ifc
- /apply_regional_loads
- /auto_arrange_frames

17.5 Plugin Packaging Format

A plugin is packaged as:

```
plugin.zip
│ plugin.json     (manifest)
│ main.py         (logic)
│ ui.py           (optional UI)
│ resources/
│ models/         (ML models or numerical DB)
│ examples/       (sample inputs)
```

17.5.1 plugin.json

```
{
 "name": "IS_2911_Pile_Design",
 "version": "1.0.0",
 "api_version": "3.1",
 "author": "ABC Consultants",
 "type": "calculation",
 "permissions": {
  "engine_access": true,
  "filesystem": false
 }
}
```

17.6 Plugin Security

- Python sandbox
- Restricted imports
- File system access denied unless permissioned
- Maximum memory & CPU quotas

- Timeouts
- Logging and audit layer

All plugin executions are recorded for:

- Reproducibility
- QA
- Dispute resolution
- Regulatory compliance

17.7 In-App Plugin Store

A built-in plugin marketplace where:

- Developers can upload plugins
- Users (engineers) can buy or activate specific features
- Organization admins can:
    - Approve plugins
    - Disable plugins
    - Set budgets
    - Define allowed plugin types

Free, paid, enterprise-only plugins supported.

17.8 Plugin Monetization Model

Options:

- Subscription per plugin
- One-time purchase
- Pay-per-use (solver runs, drawing generation)
- Company-wide license
- Enterprise custom integrations

17.9 Plugin Development Kit (PDK)

PDK provides:

- Code samples
- Template plugins
- CLI for packaging & publishing
- Testing sandbox
- FEniCSx/Kratos-ready examples
- Design code examples:
    - IS 456
    - IS 800
    - IS 875
    - IS 1893
- Drawing template samples
- API client libraries
    - Python
    - JavaScript
    - cURL

17.10 Plugin QA & Validation

Before publishing, plugins must pass:

- Automated unit tests
- Integration tests
- Output validation vs benchmark examples
- Performance tests
- Security tests
- Code safety checks

17.11 Plugin Lifecycle

1. **Development**
2. **Testing**
3. **Packaging**
4. **Publishing**
5. **Approval (admin)**
6. **Installation**
7. **Activation**
8. **Execution**
9. **Updating**
10. **Deprecation**

17.12 Examples of Plugin Scenarios

Example 1 — PEB Optimization Plugin

- User selects "Optimize PEB".
- Plugin modifies geometry → runs Kratos → checks IS 800 → updates sections.

Example 2 — Staircase Generator Plugin

Generates:

- geometry
- loads
- reinforcement
- drawings

Example 3 — G+2 Building Auto-Design Plugin

Workflow plugin that handles:

- Column positions
- Slab thickness
- Beam sizes
- Foundation design
- Drawings

Example 4 — Landfill Slope FEniCSx Plugin

Plane strain analysis for landfill benches.

17.13 Summary of Part 17

The Plugin System enables:

- Unlimited expansion
- Third-party integration
- Rapid creation of new engineering features
- Enterprise customization
- Long-term scalability
- Zero modification to core

It transforms the platform into a **structural engineering OS** rather than just a tool.

PART 18 – RESOURCE OPTIMIZATION ENGINE

A Core Intelligence Layer for Labour, Material, Equipment, Takt Planning & AI-Driven Construction Efficiency

---

18.1 Purpose of the Module

The **Resource Optimization Engine (ROE)** is the system's intelligence layer responsible for:

- Optimal selection and scheduling of **labour**, **materials**, **equipment**, and **construction resources**.
- Minimizing cost, duration, rework, downtime, and idle time.
- Automatically generating optimized project sequences and takt plans.
- Ensuring resource alignment with design outputs, BoQs, rates, and availability.
- Providing alternative solutions in case of shortages, delays, price changes, or design modifications.

This module acts as the "brain" behind overall construction execution, connecting:

- Design → Quantities → Costing → Procurement → Execution Planning → Site Workflow → Monitoring → Revision Cycle.

It must operate across **all structure types** supported by the platform:
PEB, RCC, steel structures, industrial buildings, retaining walls, RE walls, landfills, warehouses, G+2 buildings, and future expansions.

18.2 Core Objectives

1. **Predict Optimal Resource Mix**
   Using engineering quantities + productivity norms + SSR/DSR + market rates.
2. **Minimize Project Duration**
   Through dynamic scheduling, constraint satisfaction, and AI-driven sequencing.
3. **Optimize Cost**
   Achieve lowest cost without compromising safety, quality, or compliance.
4. **Remove Human Scheduling Biases**
   Auto-generate realistic and efficient plans based on target completion timeline.
5. **Simulation Before Execution**
   What-if scenarios for:
   - manpower variation,
   - equipment availability,
   - cash-flow constraints,
   - multiple subcontractor performance profiles.
6. **Auto-Adjust Plans**
   Based on:
   - real-time site progress,
   - supply chain delays,
   - weather events,
   - design revisions.

18.3 Inputs to the Module

18.3.1 Engineering & Design Inputs

- Finalized architecture + structural layout

- Member sizes, concrete grades, steel weights
- Foundation type, excavation quantities
- Staircase, plinth beam, lintels, tanks
- Retaining walls, site grading, landfills (if applicable)

18.3.2 Quantity Inputs (BoQ / Abstract of Cost)

- Cubic meters of concrete
- kg/MT of reinforcement
- sqm of formwork
- MT of structural steel
- sqm of blockwork, plastering, flooring, painting, roofing
- Electrical + plumbing quantities
- Site development quantities

18.3.3 Productivity Libraries

Pre-defined productivity tables (editable):

- Labour productivity (per trade)
- Equipment productivity (excavator, batching plant, cranes, rammers, rollers)
- Material handling productivity

18.3.4 Site Constraints

- Site access
- Working hours
- Labour availability (skilled/unskilled)
- Equipment availability
- Area-wise sequencing constraints
- Safety requirements

18.3.5 Time & Budget Objectives

- Target completion date
- Maximum budget allowed
- Milestones / owner constraints

18.3.6 Costing & Procurement Inputs

- SSR rates
- Vendor schedules
- Delivery lead-times
- Approved materials + substitutes

18.4 Outputs of the Module

1. **Resource Plan**
    - Manpower requirement per activity per day
    - Equipment plan (crane hours, excavator cycles, shuttering turnover)
    - Material availability plan
2. **Optimized Schedule**
    - Gantt chart
    - Takt-time plan
    - Floor-wise parallel sequencing
    - Automated critical path (CPM) chart
    - Rolling 3-week look-ahead plan
    - Weekly resource leveling
3. **Cost Optimization Reports**

- Best-cost strategy by adjusting resource mixes
- "Fast-track cost premium" estimates
- Idle time elimination recommendations
- Material substitution proposals

4. **Scenario Comparison**
    - Baseline vs Optimized
    - Labour-heavy vs Equipment-heavy schedules
    - Standard productivity vs Accelerated productivity
    - Cash-flow constrained scenario
5. **Live Site Execution Integration**
    - Auto-update timeline if tasks are delayed
    - Recompute resource plan in seconds
    - Suggested mitigations (add labour? add shuttering? change sequence?)
6. **AI-Powered Recommendation Engine**
    - "Reduce shuttering by adopting early stripping schedule"
    - "Add 1 more crane to reduce steel erection duration by 4 days"
    - "Shift excavation to night shift to improve productivity by 25%"
    - "Change column shutter from plywood to aluminium for faster cycle time"

18.5 Internal Sub-Modules

18.5.1 Quantification Engine Interaction Layer

This micro-module absorbs BoQ and design quantities and converts them into:

- **Work packages**
- **Task units** with uniform measurement units
- **Activity durations** based on productivity norms

18.5.2 Productivity & Simulation Engine

Handles:

- Labour vs Equipment simulation
- Monte Carlo simulation for risk
- Weather-based productivity reduction
- Multi-scenario comparison

18.5.3 Takt Planner

- Breaks building into zones (Grid A–B, B–C, etc.)
- Defines "Takt time" per zone (e.g., 4 days per zone)
- Generates Takt wagons
- Ensures continuous workflow minimizing idle time

18.5.4 Resource Leveling Engine

Ensures:

- No overallocation of equipment
- No conflicts in labour
- Balanced workflow
- Crane clashes avoided
- Formwork reuse maximized

18.5.5 Cost-Benefit Optimizer

Analyzes:

- Changing shuttering type

- Switching crane capacity
- Adopting temp Precast elements
- Steel vs RCC trade-offs
- Using aluminium formwork
- Outsourcing vs in-house labour

18.5.6 Execution Simulation Engine (3D/4D)

Creates:

- 3D visualization of sequencing
- 4D animation (3D + time)
- View by floor, activity, trade, zone
- Clash detection between trades
- Path analysis for cranes, workers

18.6 User Experience Requirements

The optimization engine must be:

- **Fully automatic** for beginners
- **Fully customizable** for engineers
- Tweaked via sliders for:
    - Resource levels
    - Shift timings
    - Target acceleration
    - Cost trade-off sensitivity

Users can visually:

- See the building in 3D
- Switch to 4D (installation timeline)
- View resource distribution per day
- Modify constraints interactively
- Apply filters: by zone, trade, activity, date

18.7 AI Interaction and Prompts

The AI will:

1. Ask user project goals
2. Ask for constraints (budget, time, site conditions)
3. Ask for preferred construction methodology
4. Build multiple scenarios
5. Provide clear recommendation summary:

"Scenario B reduces duration by 16 days with ₹8.5 lakhs extra cost.
Scenario C reduces cost by ₹5.2 lakhs but delays project by 12 days."

6. Provide work packages auto-generated:
    - Earthwork
    - PCC
    - Raft/Foundation
    - Column, Beam, Slab cycles
    - MEP coordination
    - Finishing sequence

18.8 Data Sources & Libraries

- CPWD & State SSR performance outputs
- IS codes:
  - IS 456
  - IS 800
  - IS 875
  - IS 2911
  - IS 2502
  - IS 1893
- Productivity standards (NBCC, CIDC)
- Internal datasets (your organization specific productivity norms)
- Vendor equipment specs (crane cycles, excavator capacities)
- Historical project data for ML training

18.9 Technology Components

- **Backend:** Python, FEniCSx/Kratos (if integrated for heavy simulation), FastAPI
- **Frontend:** Streamlit / React / Three.js for 3D
- **Optimization Engine:**
  - OR-Tools
  - PuLP
  - Pyomo
  - MILP solvers
- **AI Layer:** Gemini/OpenAI
- **Database:** PostgreSQL
- **Versioned Resource Libraries:** Stored configurations
- **3D/4D Engine:** Blender API, Three.js, IFC.js

18.10 Deliverables of Part 18

- Full conceptual flow
- Internal architecture
- Data pipelines
- 3D/4D simulation outline
- Optimization logic
- User journeys
- Technical specifications
- AI workflow integration
- Design requirements
- UI wireframe descriptions
- Phase-wise implementation roadmap

Part 18 Completed.

📘 PART 19 — SECURITY, DATA INTEGRITY & COMPLIANCE FRAMEWORK

(For the AI-Driven Structural Design, Estimation, and Project Management Platform)

19.1 Purpose

This section establishes the **security, integrity, compliance, governance, and operational safety** rules that must be followed across the entire platform — from architectural planning to design, estimation, project execution, document management, and closure. The intent is:

- To protect user data, drawings, models, design files, SSR rates, project estimates, and proprietary engineering logic.
- To ensure compliance with **IS codes, IT standards, data-privacy protocols, and audit requirements**.
- To guarantee that all generated designs, calculations, and outputs are **tamper-proof**, **verifiable**, **traceable**, and **audit-ready**.

This applies to all modules — **Architecture, Structural Design, Kratos/FEniCSx Simulation Layer, Drawing Engine, Estimation, Billing, QA/QC, PMC, Procurement, and Execution workflows.**

19.2 Core Security Principles

19.2.1 Zero-Trust Architecture

- Every user request is authenticated.
- Every design operation is verified.
- No module trusts another module without explicit token-based permission.

19.2.2 Principle of Least Privilege (PoLP)

Each engineer or contractor only sees:

- Their **projects**
- Their **permitted modules**
- Their **allowed design tools**
- Their **billing-activated services**

19.2.3 Defense-in-Depth

Security is layered at:

- UI → API → Microservices → Solver Engine → Database → Storage → Deployment → Backup

19.2.4 Tamper-Proof Engineering Output

All engineering outputs must include:

- Auto-generated **calculation IDs**
- SHA-256 **hash signatures**
- Version stamps
- Author + AI model + date + code references

19.3 Authentication & Authorization

19.3.1 Multi-Layer Auth

Supported modes:

- Password + Email OTP
- OAuth: Google, Microsoft
- SSO: Enterprise (SAML2.0)
- Optional MFA: TOTP / SMS

19.3.2 RBAC (Role Based Access Control)

Roles include:

- **Owner / Admin**
- **Structural Engineer**
- **Architect**
- **Estimator**
- **Site Engineer**
- **Project Manager**
- **Procurement**
- **Client / Viewer**
- **Reviewer / Approver**

Each role has access rules predefined in the ACL dictionary.

19.3.3 Project-Level Permissions

- View Model
- Modify Model
- Run Structural Solver (Kratos/FEniCSx)
- Generate Drawings
- Issue Documents
- Download Reports
- Approve / Reject
- Modify Cost Data
- Upload Sketch / Plan
- Access AI drafting tools

19.4 Data Integrity Requirements

19.4.1 Versioning

Every file (drawing, model, calculation sheet) must maintain:

- Version number
- Change description
- Editor identity
- Date/time
- Comparison diff view

19.4.2 Immutable Design Vault

Once a design is "Approved":

- It becomes **read-only**
- Only new revisions can be created
- All revisions must maintain **history lineage tracking**

19.4.3 Design Chain of Custody

Each model action is recorded:

- Node added/removed
- Member edited
- Loads updated
- Meshing updated
- Solver iterations
- Code compliance checks
- Drawing regenerated

All logs stored in a secure audit database.

19.5 Secure Handling of Engineering Models

19.5.1 3D Model Protection

Models (BIM, CAD, STAAD-like model files, Kratos/FEniCSx meshes) must:

- Be encrypted at rest
- Avoid exposing solver internals directly
- Allow role-based granular editing

19.5.2 Secure Upload Pipeline

User-uploaded plans/sketches must be:

- Virus-scanned
- File-type validated
- Sanitized (SVG/CAD cleansing)

19.5.3 Sandbox Execution

All engineering solvers (Kratos/FEniCSx) run in isolated sandboxes:

- No internet access
- No external calls
- Memory/cpu limits
- Safe Python mode

19.6 Secure APIs

19.6.1 API Gateway Rules

- Token-based authentication (JWT)
- Rate limiting
- IP throttling
- WAF-style rule enforcement

19.6.2 Signed Download URLs

All exported reports (PDF/Excel/DWG) require:

- Time-limited signed URLs
- Mandatory access logs

19.6.3 Secure Interoperability

Import/export with:

- STAAD
- ETABS
- Revit
- Tekla
- DXF/DWG
- IFC

must use:

- Secure conversion microservices
- No direct file system access

19.7 Data Privacy

19.7.1 PII Rules

User identity data stored separately from project data.

19.7.2 GDPR-like Compliance

Even if India-specific:

- Right to access

- Right to delete
- Right to revoke API access
- Right to download all project data

19.7.3 Customer Isolation

For multi-tenant SaaS:

- Complete tenant isolation
- Tenant-specific encryption keys

19.8 Security for AI + Engineering Logic

19.8.1 AI Prompt Firewall

Prevent malicious prompts such as:

- Code injection
- Solver bypass
- Rate file manipulation
- Unauthorized data access

19.8.2 Engineering Safety Layer

AI cannot:

- Override IS/ACI/AISC/EC compliance
- Approve unsafe designs
- Skip validation steps
- Generate drawings without mandatory checks
- Modify SSR / cost data without permissions

19.8.3 Calc Verification

Every AI-generated design is checked via:

- Deterministic baseline rules
- Numerical simulation checks
- Structural safety envelope validation
- Peer-review workflow

19.9 Compliance Requirements

19.9.1 Engineering Code Compliance

Platform must comply with:

- **IS 456, IS 800, IS 875, IS 1893, IS 3370**
- **ACI 318, AISC 360**
- **EC2, EC3, EC7**
- **IRC & MORTH**
- **CPHEEO, SWM Rules**
- **Local municipality bye-laws**

Every design includes:

- Code reference
- Clauses used
- Partial safety factors
- Load combinations
- Material grades
- Solver method

19.9.2 IT Compliance

- ISO 27001

- SOC 2 readiness
- Data retention rules
- Digital signature readiness

19.10 Backup & Disaster Recovery

19.10.1 Auto Backups

Automatic backup every 6 hours:

- Project files
- Inputs
- Drawings
- Calculations
- AI transcripts
- Metadata

19.10.2 Disaster Recovery Objectives

- RPO ≤ 15 minutes
- RTO ≤ 30 minutes

19.10.3 Multi-Region Backup

Data stored in:

- Primary region
- Hot standby region
- Cold storage archival

19.11 Monitoring & Alerts

- Authentication failures
- Abnormal usage
- Large model uploads
- Excess solver time
- Multiple download attempts
- Unauthorized data changes

Notifications sent to:

- Admin
- Project owner
- Security system

19.12 Security in Deployments (Antigravity / Docker)

19.12.1 Docker Security

- Non-root containers
- Read-only filesystem for solvers
- Seccomp profile
- Resource limits: CPU/Memory

19.12.2 Antigravity Deployment

- Environment secrets encrypted
- Locked-down networking
- Isolated microservices
- Logging & auditing enabled

19.13 Summary of Security Framework

Your platform will operate with:
✓ A secured **AI-driven engineering workflow**
✓ Strict access controls across all modules
✓ Tamper-proof **engineering design history**
✓ Safe execution of Kratos/FEniCSx solvers
✓ Immutable drawing + model versioning
✓ Secure import/export (STAAD, Revit, Tekla, CAD)
✓ Full compliance with Indian + global codes
✓ Multi-tenant SaaS readiness
✓ Enterprise-grade monitoring & DR

📘 PART 20 — DEPLOYMENT, CI/CD, VERSIONING & ENVIRONMENTS (FULL DETAILED PRD SECTION)

(For the Unified Structural AI-Driven Design, Analysis, and Execution Platform)

20.1 Objective

This part defines **how the entire platform is deployed, updated, tested, monitored, versioned, and moved from development → staging → production**, ensuring:

- High reliability
- Zero-downtime releases
- Safe rollbacks
- Multi-engine support (Kratos, FEniCSx, OpenSees optional)
- Multi-tenant SaaS support (per-project & per-user isolation)
- Cost-efficient operation with capability to scale

This deployment strategy supports **Antigravity**, **AI Studio**, **Docker-based cloud**, **on-premise setups**, and **enterprise installations**.

20.2 Deployment Architecture Overview

The system uses a **containerized, multi-service architecture**:

20.2.1 Containers

Each container includes:

Component	Description
AI Orchestrator	Manages conversations, workflows, task pipelines
Frontend	React/NextJS or SvelteKit UI with 3D Viewer (three.js)
Backend API Gateway	Python FastAPI/NodeJS orchestrates all modules
Engineering Engines	Kratos + FEniCSx in separate containers
Database	PostgreSQL + Vector DB (Milvus/Weaviate)
Storage	MinIO/S3 for drawings, models, BIM files
Worker Queue	Celery/RQ/Kafka for long-running simulations
Billing Module	Manages credits per module and per step
Logging/Monitoring	Grafana + Prometheus + Loki

20.3 Environments (Dev → Stage → Prod)

20.3.1 Development

- Local Docker Compose
- Hot reload enabled
- Mock engines instead of full Kratos
- Debug logs ON
- Fewer security restrictions
- Local SQLite/mini-postgres

20.3.2 Staging

- Mirrors production setup
- Uses test billing
- Real Kratos & FEniCSx engines
- Synthetic load tests
- Automated drawing generation sandbox

20.3.3 Production
- Auto-scaling containers
- Vertical & horizontal scaling
- API rate limiting
- Full security suite:
    - JWT + OAuth
    - IP throttling
    - WAF
- Multi-region deployments

20.4 CI/CD Pipeline (Full Automated Pipeline)
Pipeline runs on **GitHub Actions**, **GitLab CI**, or **Antigravity CI Runner**.
20.4.1 Triggers
- PR merge
- Commit on main
- Manual deployment
- Version tag creation
- Scheduled nightly build

20.4.2 Steps
1. Pre-Build Checks
- Code linting (flake8, pylint, ESLint)
- Type checks (mypy, pyright)
- Dependency scans
- Secret scanning (Gitleaks)

2. Build Phase
- Build frontend container (Node)
- Build backend container (Python)
- Build Kratos engine compute container
- Build FEniCSx custom solver container
- Build drawing engine container

3. Test Phase
- Unit tests (pytest, jest)
- Integration tests
- Structural engine regression tests (benchmark reference files)
- Drawing generation tests (compare SVG/PNG outputs)
- Workflow simulation tests (AI → design → drawings → cost)

4. Security Phase
- Dependency vulnerability scan
- Docker image scan
- Static code analysis
- SBOM (Software Bill of Materials) generation

5. Deployment Phase
Staging:
- Blue environment deployment
- Smoke tests
- Approvals (if enterprise)

- Auto promote to production if approved

Production:

- **Blue-Green deployment**
- Auto-rollback if health checks fail
- Canary rollout (5% → 20% → 100%)

20.5 Versioning Strategy

Semantic Versioning

MAJOR.MINOR.PATCH

Examples

- 4.0.0 — New engines added (e.g., Kratos + FEniCSx combined release)
- 4.3.0 — New drawing module added
- 4.3.7 — Bug fixes & performance improvements

Version Synchronization

The following components maintain synchronized versions:

1. **Core API**
2. **AI Orchestrator**
3. **Engineering Engines**
4. **Drawing Engine**
5. **Workflow Modules**
6. **Costing/SSR Engine**

Versioning ensures:

- Compatibility across modules
- Guaranteed reproducible design history (for approvals, compliance)
- Backward compatibility

20.6 Deployment Targets

20.6.1 Antigravity

- Native container deployment
- Best for scalable SaaS
- Supports GPU compute (if required)
- One-click version promotion

20.6.2 Google AI Studio

- Handles conversational agent only
- Engineering backend stays in Antigravity or VPS

20.6.3 On-Prem Enterprise

For large companies like L&T, Tata Projects, BHEL:

- Kubernetes cluster
- GPU-enabled engineering nodes
- Private MinIO/S3 storage
- Encrypted internal communication

20.6.4 Standalone Portable Installer

For consultants:

- Single EXE or AppImage
- Local Kratos/FEniCSx via Docker Desktop
- Local SQLite database
- License-locked

20.7 Monitoring & Observability

Monitoring includes:

20.7.1 Performance

- CPU/GPU usage of engine workers
- Queue depth
- Simulation durations

20.7.2 Errors

- Engineering model failures
- AI pipeline errors
- Drawing engine errors
- User input validation failures

20.7.3 Business Metrics

- Module usage count
- Billing consumption
- Conversion from design → costing → execution

Tools:

- Grafana dashboards
- Loki logs
- Prometheus metrics
- Sentry for errors

20.8 Rollback & Recovery

Automatic Rollback Conditions

- 5xx error rate exceeds threshold
- Simulation failures > threshold
- Drawing generation fails
- Database migrations fail

Recovery Procedure

- Switch back to previous Docker tag
- Restore DB from snapshot
- Re-sync all engines
- Trigger regression test suite

20.9 Backup Strategy

Frequency

- DB backups: hourly
- S3/MinIO storage: daily
- Engine outputs: incremental
- AI Knowledge store: weekly

Retention

- 7 days → hot
- 30 days → warm
- 1 year → cold archive

20.10 Multi-Tenancy & User Isolation

The platform supports:

- Project-level isolation
- User-level document separation
- Company-wise data segregation
- Encryption at rest (AES-256)
- Role-based access (Engineer, QA, Client, Admin)

20.11 Deployment Cost Strategy

To keep costs low

- Spot instances for simulation
- Auto-scale down when idle
- Serverless orchestrator (if used)
- Tiered compute (Basic / GPU / HPC)
- Design credits model for monetization

20.12 Deliverables from Part 20

This part produces:

- Complete CI/CD blueprint
- Deployment architecture diagrams
- Version management plan
- Multi-environment setup
- Monitoring & rollback procedures
- Scalability and security blueprint

📘 PART 21 — COMMERCIAL MODEL, PRICING ENGINE & BILLING SYSTEM

This part defines how the platform will **monetize**, **price**, **bill**, **track usage**, **apply metered design credits**, and manage **per-module revenue streams**.
This ensures the app becomes a **scalable SaaS product** that charges users *only for the engineering services they use*.

21.1 PURPOSE OF THIS MODULE

The Commercial Model & Billing Engine must:

- Monetize every design action (architecture, analysis, design, drawings, estimates).
- Allow free/basic access but require credits/payment for premium structural tasks.
- Charge based on **usage**, not subscription alone.
- Track usage per user/project.
- Apply enterprise licensing for consultants, companies, and institutions.
- Integrate with Razorpay/Stripe/PayPal.
- Maintain transparent cost sheets for users.

This makes the platform function like a **digital engineering marketplace + design automation SaaS**.

21.2 KEY CONCEPT — "PAY FOR WHAT YOU USE" ENGINEERING

Each structural or architectural step has a cost:
User flow example
User says:
"G+2 building in Hyderabad"
App asks questions → user provides drawings / sketch →
APP CREATES:

1. Concept Plan
2. Architectural 2D Plan
3. 3D Building model
4. Structural model (columns/footings/beams/slabs)
5. Foundation design
6. Detail drawings
7. Estimation
8. Execution schedule
9. Billing
10. Project tracking

Each step has a **price**.
The user pays only for the steps they use.

21.3 COMMERCIAL MODEL OVERVIEW

Category	Example Features	Pricing Type
Basic UI + Model Viewer	3D viewer, edit model	FREE
Architectural Layout Generator	AI plans, room sizes	Paid per layout
Structure Generator	AI column-beam-slab generator	Paid per model

Category	Example Features	Pricing Type
Structural Analysis & Design	RCC/Steel/Composite	Paid per run
Detailed Drawings	Footing/Beam/Column/Slab drawings	Paid per drawing
Bar Bending Schedule	Auto-generated	Paid per table
Estimation (SSR Rates)	Civil, steel, MEP	Paid per estimate
Project Execution Engine	Schedules, cash flow	Subscription
Quality/Safety/Checklists	Templates	Subscription
Reports & DPRs	Auto-generated	Paid per report

21.4 DESIGN CREDITS SYSTEM (ENGINEERING TOKENS)

User buys *Engineering Tokens*.

Examples:

- 100 tokens → ₹199
- 500 tokens → ₹799
- 2000 tokens → ₹2,499
- 10,000 tokens → custom enterprise pricing

Usage example:

Action	Tokens
Auto Architectural Plan Generation	10
3D Model Build	20
Structural Model Creation	15
RCC Design Run	50
Steel Frame Design Run	40
Code Compliance Check	15
Footing Detailing	20
One Sheet Drawing PDF	10
Estimation	25

Tokens ensure predictable scaling.

21.5 MODULE-WISE PRICING SYSTEM

21.5.1 Architectural Module

- AI Plan: 10 tokens
- Elevations: 15 tokens
- Render: 30 tokens

21.5.2 Structural Module

- Load generation (IS 875/1893): 5 tokens
- Structural model build: 15 tokens
- Analysis run (Kratos/FEniCSx): 20 tokens
- Design run (IS/AISC/ACI/EC): 50 tokens

21.5.3 Detailing Module

(per drawing sheet)

- Footing: 20
- Column: 10
- Beam: 15
- Slab: 15
- Staircase: 20
- Retaining Wall: 25
- Steel connections: 5 per connection
- GA Drawing: 30

21.5.4 Estimation Module

- SSR-based estimate: 25 tokens
- Material-wise breakdown: 5 tokens
- Labour estimate: 10 tokens

21.5.5 Project Management Module

Subscriptions:

- ₹499/month → Individual Engineer
- ₹1,999/month → Contractor
- ₹9,999/month → Company Pack
- Custom → Enterprise

21.6 BILLING ENGINE

The billing engine must support:

21.6.1 Real-time usage tracking

- Each module emits a "usage event"
- Billing system listens & records
- Tokens deducted instantly

21.6.2 Full Invoice Generation

Invoices must include:

- Job ID
- Modules used
- Token consumption
- Currency value
- GST breakdown
- Payment status
- Downloadable PDF

21.6.3 Supported payment gateways

- Stripe
- Razorpay (India)
- PayPal
- UPI (PhonePe, GPay)

21.7 MULTI-PROJECT LICENSES

Engineers/companies can buy project packs:

- 1 Project – ₹799
- 5 Projects – ₹2,999
- 20 Projects – ₹9,999
- Unlimited – Enterprise

Each project unlocks **all AI, modeling, estimation, and reporting** automatically.

21.8 ENTERPRISE FEATURES

Corporates & Government Bodies get:

- Unlimited seats
- Dedicated compute cluster
- API integration with ERP
- SSO & IAM
- Audit logs
- Private AI datasets
- Bulk DPR generation
- Integration with BIM workflows

Pricing: ₹10 lakh – ₹1.5 crore annually depending on scale.

21.9 PAY-AS-YOU-GO MODEL FOR HEAVY COMPUTATION

For Kratos/FEniCSx finite element runs that are very heavy:

- System meters CPU/GPU time
- User pays tokens accordingly

Example:

- Kratos 3D nonlinear run → 200 tokens
- Soil-structure interaction → 150 tokens
- Steel connection FE → 25 tokens

21.10 COMMERCIAL RISKS & MITIGATIONS

Risk	Mitigation
High compute cost	Use GPU server pooling, caching, reuse FE models
Payment failures	Auto-retry, local credits, escrow
Users misusing free tier	Rate limits + token ceilings
Refund disputes	Transparent logs + downloadable reports

21.11 DATA COLLECTION FOR COMMERCIAL ANALYTICS

Track:

- Feature usage frequency
- Average tokens per project
- Most profitable modules
- Drop-off points in user workflow

This guides future upgrades & pricing adjustments.

21.12 REQUIRED BACKEND STRUCTURE

Tables:

- users
- projects
- tokens
- token_transactions
- invoices
- module_usage_log
- pricing_rules

- subscription_plans

Services:

- pricing-engine
- billing-engine
- invoice-generator
- payment-gateway-service
- token-wallet-service

21.13 FRONTEND UX REQUIREMENTS

- Always show **remaining tokens** at top-right
- Show price of each action before running
- Bill summary popup after analysis/design
- Transparent invoices
- One-click token top-up
- Discounts visible during checkout

21.14 COMPLIANCE & TAX

- GST 18% for Indian users
- Export of services → No GST for foreign users
- Invoice numbering as per Indian rules
- Optional TAN/PAN capture for enterprises

21.15 OUTCOME OF PART 21

You now have a **complete commercial architecture** that allows:

✓ Token-based billing
✓ Module-wise monetization
✓ SaaS subscription model
✓ Enterprise licensing
✓ Pay-as-you-go FE computation
✓ SSR-based estimation monetization
✓ GST-compliant invoicing

📘 PART 22 — UNIVERSAL INTEROPERABILITY & DATA EXCHANGE LAYER

(STAAD ↔ Kratos ↔ FEniCSx ↔ ETABS ↔ Tekla ↔ IFC ↔ CAD ↔ PDFs ↔ BIM Engines)

(Deep Technical PRD Specification)

22.1 Purpose of This Module

This layer creates a **unified, loss-minimal, bi-directional translation and communication system** that allows your entire app ecosystem to **read, write, import, export, translate, convert, regenerate, re-simulate, and re-design** any model from:

- STAAD .std
- ETABS .edb (via API extraction)
- SAP2000 .s2k
- Tekla .db1, .ifc
- Revit .rvt (via IFC or Forge)
- CAD .dxf, .dwg
- Kratos input files
- FEniCSx meshes (.xdmf, .xml, .h5)
- IFC 2x3 / 4 / 4.3
- Custom JSON model schema (App's own UDM)
- PDF drawings (via OCR + layout extraction)
- Hand-sketch uploads (AI CAD Reconstruction)

This is the **glue** that makes your app usable by any structural engineer, architect, or contractor regardless of what tools they currently use.

22.2 Core Design Philosophy

22.2.1 "Input-Agnostic, Code-Agnostic, Solver-Agnostic"

No matter what a user uploads or selects, the system converts everything into the **Unified Data Model (UDM)**.

22.2.2 "Round Trip Fidelity"

You can:

- Import STAAD → Modify → Export back to STAAD
- Import Revit → Add beams → Export to IFC
- Upload hand sketch → AI converts → UDM → full model

22.2.3 "Minimal Info Loss Guarantee"

All original metadata (load cases, parameters, materials, offsets, releases, section definitions) is preserved even if unsupported by one solver.

22.3 Supported Import Types

22.3.1 STAAD .std Import

- Parse nodes, beams, plates, solids.
- Extract:
  - Member releases
  - Properties
  - Beta angles
  - Offsets
  - Master-slave links
  - Load combinations
  - Wind & seismic definitions

- Convert STAAD coordinate system to UDM.
- Map sections to IS profiles or custom tapered sections.

22.3.2 ETABS/SAP .e2k/.s2k/.xml Import

- Via native text API.
- Extract:
    - Geometry
    - Storey definitions
    - Load patterns
    - Load cases
    - Groups/diaphragms
    - Wall/slab properties
    - Pier/spandrel labels
- Convert to UDM.

22.3.3 Tekla → IFC → UDM

Extract:

- Columns, beams, plates
- Connection metadata
- Weld sizes
- Fabrication IDs
- Bolt groups
- Assemblies

22.3.4 CAD .dxf/.dwg Import (Architectural Plans)

AI-assisted extraction:

- Walls
- Gridlines
- Columns
- Beams (if annotated)
- Slab boundaries
- Stairs
- Sections
- Dimensions
- Door/windows
- Levels

22.3.5 Hand Sketch → AI → UDM

Workflow:

1. User draws a rough plan on paper
2. Upload photo
3. AI auto-detects:
    - Rooms
    - Walls
    - Openings
    - Grids
4. AI converts to vector plan
5. App builds geometry
6. User edits in 2D/3D UI

22.3.6 PDF → AI OCR → Structural Model

Extract:

- Gridlines
- Column sizes
- Beam sizes
- Slab thickness
- Notes
- Load specs
- Level data

Outputs directly mapped into UDM.

22.4 Supported Export Types
Export Options:

Export Type	What is Preserved
STAAD .std	Nodes, beams, plates, loads, combos, properties
Tekla .ifc	steel members, plates, welds, bolts
Revit .ifc	BIM objects with metadata
ETABS .xml	geometry + loads
DXF/DWG	2D architectural + structural drawings
PDF	GA drawings, schedules, details
Kratos input	mesh, BCs, material blocks
FEniCSx	mesh (.xdmf/.h5), BCs, load definitions

22.5 Internal Architecture
This module contains **five components**:
22.5.1 Model Loader

- Reads file
- Converts into temporary AST
- Validates syntax
- Handles units

22.5.2 Feature Extractor

- Detects:
    - Walls
    - Members
    - Levels
    - Materials
- Converts coordinate system
- Generates element lists

22.5.3 UDM Mapper
Guarantees that:

- Nodes → UDM.Nodes
- Elements → UDM.Elements
- Sections → UDM.Sections
- Groups → UDM.Groups
- Load cases → UDM.LoadCases

22.5.4 Solver Translator
Converts UDM → solver-specific files:

- UDM → Kratos
- UDM → FEniCSx
- UDM → ETC

22.5.5 Round Trip Manager

Tracks:

- What came from source file
- What was modified in UI
- What should be preserved when exporting

22.6 AI-Assisted Plan Interpretation

22.6.1 Objective

Allow user to upload:

- Sketch
- Old PDF plan
- Image
- Cad drawing

AI then:

- Vectorizes (like AutoCAD)
- Generates grids
- Detects rooms
- Guess structural members
- Suggests column layouts
- Suggests beam spans
- Suggests stair configurations

22.7 2D/3D Design Environment Integration

Once imported, the model is editable in:

- 2D plan view
- Elevation/section views
- Full interactive 3D model
- Component inspector
- Member property editor
- Load editor
- Connection editor
- Foundation editor

Exactly the same experience as:

- STAAD Editor
- ETABS Model Explorer
- Tekla 3D Model Tree

22.8 Error Handling & Recovery

Detect:

- Unsupported section profiles
- Missing levels
- Zero-geometry walls
- Duplicate nodes
- Non-matching mesh faces

- Dangling beams

Auto-Fix:

- Snap nodes to nearest grid
- Rebuild topology
- Suggest corrections to user

22.9 Performance Requirements

- Import 50,000+ finite elements
- Load IFC ≤ 2 seconds
- Load DWG ≤ 1 second
- Model translation ≤ 200 ms
- UI should display < 3M polygons in 3D viewer without lag

22.10 Deployment Architecture

This module runs in:

- Backend (Python, Kratos, FEniCSx)
- Microservice (Node/Python)
- GPU-accelerated AI service (for OCR + image detection)

📘 PART 23 — SYSTEM-WIDE INTELLIGENCE LAYER (SWIL)

AI-Powered Reasoning, Learning, and Cross-Module Orchestration

This part defines the *brain* of the entire platform — the layer that understands user intent, orchestrates workflows, extracts engineering meaning from sketches and drawings, recommends optimizations, validates compliance, anticipates missing inputs, learns from previous projects, and assists engineers throughout design, estimation, planning, execution, and closure.

This is a mandatory component enabling the platform to behave like a **virtual structural engineer + architect + estimator + project manager**.

23.1 Purpose of the System-Wide Intelligence Layer (SWIL)

SWIL ensures that the platform moves beyond "manual model-building" and becomes a **guided, semi-autonomous engineering system**, capable of:

1. Understanding *user intent*

- "I want a G+2 building in Hyderabad"
- "I need a 20m clear-span industrial shed"
- "I want to check stability of an RE Wall"
- "I only want estimation; skip design"

2. Asking follow-up questions

Automatically generates structured dialogs:

- Soil profile?
- Site location (to get wind/seismic)?
- Loads (live, dead, machinery, storage)?
- Usage type (residential/industrial/commercial)?

3. Mapping user intent → Engineering modules

- Architectural → Space planning engine
- Structural → Framing system identification
- Geotechnical → Soil & foundation module
- Estimation → SSR-driven rate analysis
- Drawings → 2D + 3D Generator
- Workflow → Project lifecycle generator

4. Reducing manual modelling through automation

- Suggests column grids
- Suggests best-fit framing layouts
- Suggests PEB or RCC depending on span/economy
- Auto-detects missing inputs and requests them

5. Real-time engineering validation

- Detects under/over-designed members
- Warns about IS 875/IS 456/IS 1893 non-compliance
- Validates architectural → structural consistency
- QA/QC rule engine for every module

6. End-to-end process orchestration

- From input → layout → model → design → drawings → BOQ → execution plan
- Allows skipping stages (e.g., estimation only)

23.2 SWIL Architecture Overview

USER INPUT + DRAWINGS + SKETCHES

▼

23.A INTENT INTERPRETATION ENGINE
(NLP-driven Understanding of Requirements)

▼

23.B PARAMETER EXTRACTION & KNOWLEDGE FUSION LAYER
(Extracts spans, floors, loads, spaces, constraints)

▼

23.C ENGINEERING REASONER & RULE ENGINE
(IS/IRC/ACI/AISC/EC code logic + expert heuristics)

▼

23.D WORKFLOW ORCHESTRATION ENGINE
(Architectural → Structural → Estimation → PMIS)

▼

23.E LEARNING & ADAPTATION LAYER
(Improves suggestions based on project history)

23.3 Component 23.A — Intent Interpretation Engine

Inputs:

- Free text (“I want a 50x80 warehouse”)
- Images (sketch, screenshot, hand drawing)
- Uploads (PDF plan, DWG, DXF)
- Menu selections

Responsibilities:

- Identify structure type (PEB / RCC / RE Wall / Landfill / etc.)
- Determine whether the user wants:
    - Planning only
    - Design only
    - Estimation only
    - Full lifecycle
- Detect location → auto-load wind/seismic zones

- Detect missing essential information → generate prompts automatically

Core AI Skills:

- NLP classification
- OCR + vector extraction
- Plan/section detection
- Room/beam/column/wall recognition
- Auto-gridding suggestion

23.4 Component 23.B — Parameter Extraction & Knowledge Fusion

This layer turns unstructured input into engineering-ready parameters.

Extracted Parameters:

- Geometry (spans, grids, storeys)
- Site data (SBC, wind, seismic)
- Material preferences (steel/RCC/composite)
- Load information (live, machinery, floor load)
- Functional requirements (residential/industrial/commercial)
- Architectural constraints (setbacks, staircase, lift core)

Knowledge Sources:

- Code maps (IS 456, IS 875, IS 800, IS 1893, IRC, CPHEEO)
- SSR databases
- Previous project libraries
- Manufacturer catalogues
- GIS & climate data

Outputs:

- Normalized, validated parameter dictionary
- Ready for feeding into geometry engine/model builder

23.5 Component 23.C — Engineering Reasoner & Code Compliance Engine

This is the *core engineering brain*.

Responsibilities:

1. Identify structural system
2. Recommend structural framing
3. Suggest foundation type
4. Detect inconsistencies or missing data
5. Validate inputs against:
    - IS codes
    - ACI/AISC/EC (as selected)
    - Internal engineering rules

Examples of rule checks:

- PEB → span > 18m, bay 6m optimum
- RCC → column spacing ≤ 8m for residential
- RE Wall → reinforcement length ≥ 0.7H
- Retaining Wall → FS sliding > 1.5
- Landfill → slope stability + benching logic

Inputs:

- Normalized data from 23.B

Outputs:

- Structural system proposal:
    - RCC frame, PEB frame, composite, truss, moment frame, shear wall, etc.
- Grid recommendations
- Member sizing starting points
- Foundation recommendations

23.6 Component 23.D — Workflow Orchestration Engine

Automatically creates a stage-by-stage workflow after interpreting user intent.

Supported flows:

- **Full Architectural → Structural → MEP → Estimation → Execution**
- **Structural Only**
- **Estimation Only**
- **Drawing Only**
- **Design Validation Only**
- **Landfill Only**
- **PEB Only**

Responsibilities:

- Direct the user between screens based on chosen workflow
- Trigger appropriate downstream modules:
    1. Space planning
    2. Grid generation
    3. 3D modelling (Kratos)
    4. Analysis
    5. Design
    6. Drawing generation
    7. BOQ/Estimation
    8. Execution plan
    9. Approvals package

Smart Behaviors:

- Auto-skips non-applicable modules
- Auto-fills defaults based on region and building type
- Supports “jump” modes (e.g., go directly to BOQ)

23.7 Component 23.E — Learning & Adaptation Layer

This system becomes smarter over time.

Learns from:

- User preferences
- Region-specific standarization
- Most economical design choices
- Company-specific SOPs
- Government approval formats
- Past completed designs
- Frequent drawing templates

Output Improvements:

- Better initial framing recommendations
- Better questions during input
- Faster workflow creation

- Optimized grid layouts
- Improved estimation accuracy

Storage:

- Structural pattern library
- Local optimization dataset
- Historical costs and SSR updates
- Approved vs rejected designs for tuning

23.8 API Interfaces for SWIL

SWIL interacts with every major subsystem:

Interfaces:

- **Architectural Engine**
- **3D/Kratos Model Builder**
- **Analysis/Design Engine**
- **Drawing Engine**
- **Estimation Engine**
- **SSR Rate Database**
- **Project Management Engine**
- **User Profile & Billing Engine**

Data Formats:

- JSON-based parameter sets
- Geometry objects
- Rule violation objects
- Cost breakdown objects
- Learning datasets

23.9 Constraints & Requirements

- Must operate in **real time (<1 sec)** for parameter checks
- Must support:
  - English + Indian regional languages
  - Image → engineering object pipelines
  - Parametric modelling
- Must maintain compliance logs for:
  - IS codes
  - BIS approval packages
  - Engineering audit trails

23.10 Deliverables of Part 23

This PRD section produces:

1. Detailed architecture diagrams
2. API specification outline
3. Unified intent → workflow mapping template
4. Rule engine framework (IS, ACI, AISC, EC, CPHEEO)
5. Learning data schema
6. Interaction protocols for every subsystem

📘 PART 24 — COMPLIANCE, CERTIFICATION & REGULATORY APPROVAL FRAMEWORK

This section ensures your unified structural-design–to–execution app is **fully acceptable to Indian approving authorities**, while remaining compatible with global codes and digital-checking workflows (BIM, IFC, automated compliance engines, AI-assisted checking).

This is essential because your app must generate output that is **submission-ready**, **digitally stamped**, and **verifiable** — whether for a **G+2 building in Hyderabad**, an **industrial shed in Gujarat**, an **RE Wall for NHAI**, or **landfill embankments** for CPCB.

24.1 OBJECTIVES

The compliance framework ensures:

1. Legal & Statutory Compliance

- IS 456, 875, 1893, 800, 801, 802, 2911, 13920
- IRC codes (for roads, bridges, culverts)
- NHAI/CPWD/MORTH/Indian Railways
- Local municipal bye-laws (GHMC, HMDA, DDA, BDA, BMC, etc.)

2. Engineering Compliance

- Structural safety
- Architectural codes (NBC 2016)
- Fire safety
- Accessibility, ramps, emergency exits
- Material certification (steel grade, concrete mix design, geotextiles, geomembranes, etc.)

3. Approver-Ready Output

- Layouts
- Detailed drawings
- BOQs
- Hand-calculations
- FEM reports (Kratos/FEniCSx)
- Connection design sheets
- Soil testing reports
- Water tank / phasing / construction sequence
- Signage, OHS, site logistics

4. Digital Stamping & Versioning

- Auto-generated “Engineer In Charge” digital seal (if configured)
- Change-tracking
- Version comparison
- Submission logs

24.2 APPLICABLE CODES & STANDARDS (MASTER LIST)

This list is dynamically activated depending on the user’s project type.

A. RCC Structures

- IS 456
- IS 875 (Parts 1–5)
- IS 1893 (Seismic)
- IS 13920 (Ductile design)

- IS 2911 (Foundations)
- IS 16700 (High-rise buildings)

B. Steel / PEB / Industrial Structures

- IS 800:2007
- IS 801 (thin-walled)
- IS 802 (transmission towers)
- IS 806 (steel tubes)
- IS 875 (wind loads)
- IS 1893 (industrial seismic)
- AISC / EN / Eurocode / CSA (if user chooses global mode)
- ACI 318 for anchor bolts
- AWS D1.1 for welding

C. Roads, Bridges, Culverts

- IRC SP- codes
- IRC 6, IRC 22, IRC 112
- MoRTH Specifications

D. Landfills / SWM

- CPHEEO Manual
- CPCB Guidelines
- EPA (secondary reference)
- IS 2720 (Soil tests)
- Geosynthetics standards (GCL, HDPE liners)

E. Architectural & Sanitary Codes

- NBC 2016
- Fire Norms (Part 4)
- Plumbing Codes
- Accessibility norms

F. Environmental Compliance

- EIA norms
- Pollution control board permissions (air/water)
- Stormwater drainage standards
- Harvesting norms

24.3 CODE CHECK AUTOMATION ENGINE

Each design module (PEB, RCC, RE Wall, Retaining Wall, Landfill, Composite Structures, etc.) plugs into a unified **Code Checker Engine (CCE)**:

Core components:

1. **Limit State Check Engine**
   - Bending, shear, axial, torsion
   - Deflection
   - Crack width
   - Ductility
   - Buckling
   - Fatigue (if required)
2. **Connection/Detail Check Engine**
   - Base plate design
   - Anchor bolts (ACI/AISC)

    - Welds, bolts
    - Fin plates, moment connections
    - Bracings
3. **Serviceability Engine**
    - Floor vibrations
    - Deflection
    - Soil settlement
    - Tank uplift/sliding
    - Drainage slopes
    - Landfill stability
4. **Special Seismic Engine**
    - Response spectrum
    - Drift checks
    - Strong column–weak beam
    - Foundation rocking checks
    - Retaining wall + surcharge + dynamic pressure
    - Ductile detailing rules (IS 13920)
5. **Soil Interaction Engine**
    - SBC
    - Settlement
    - Liquefaction
    - Passive resistance
    - Sliding checks
    - Ground improvement options
6. **Automated Failure Reporting**
    - Failure reason
    - Governing load case
    - Governing clause in code
    - Suggested fixing (increase column thickness, change section, add bracing, etc.)

24.4 DIGITAL APPROVAL PACKAGE GENERATION
Automatically generates a **submission-ready pack**:
Documents generated:

- General notes
- Design brief
- Design philosophy with selected codes
- FEM analysis report (Kratos/FEniCSx)
- Load calculations (wind, seismic, dead, imposed, crane, silo, vehicular, soil, hydrostatic)
- Foundation design + soil report
- Structural drawings (PDF + DWG)
- Connection drawings
- BOQ
- Rate analysis (SSR/local/market)
- Safety drawings
- Architectural layouts

- MEP utility paths
- Execution methodology
- Construction sequence
- QA/QC plan
- OHS plan
- Checklists (formwork, steel, concrete, welds, bolting)
- Material test certificates (auto templates)
- NDT/DT templates
- Inspection request formats
- Final completion dossier templates

All can be digitally stamped with:

- Digital signature
- QR code linking model version
- Approval ID
- Audit trail

24.5 USER WORKFLOW FOR COMPLIANCE

1. **User describes project**
   "G+2 building in Hyderabad with parking and water tank."
2. **AI asks structured questions**
   - Plot dimensions
   - Number of floors
   - Soil reports
   - Preferences (RCC/steel)
   - Fire & lift requirements
   - Parking demand
   - Special services (solar, STP, rainwater harvesting)
3. **App selects applicable codes**
   (E.g., IS 456, IS 875, IS 1893 Zone II for Hyderabad)
4. **App builds conceptual layout**
   - Column grids
   - Beam framing
   - Slab system
   - Staircase options
   - Foundation scheme
5. **User modifies model in 3D**
   Like STAAD:
   - Rotate
   - Zoom
   - Edit nodes/sections
   - View cross sections
   - Add openings, cutouts
6. **AI performs full code checks**
   - Capacity vs demand
   - Clause-by-clause compliance
   - Connection checks
   - Failures + solutions

7. **User selects module-wise paid outputs**
    - Architectural plan
    - Full structural design
    - BOQ
    - Execution plan
    - QA/QC
    - Project management / schedule
    - Completion pack

24.6 INTERNAL AUDIT & VERSION CONTROL

Every design version is saved with:

- Design ID
- Time stamp
- Engineer notes
- Code selections
- Material selections
- Structural system
- Load assumptions
- FEM model version
- Drawing version
- Changes from previous version
- Reviewer comments

User can:

- Roll back to any version
- Clone a version for “Option B”
- Generate diff reports

24.7 CROSS-CHECKING, VALIDATION & RED FLAG SYSTEM

The system automatically captures:

Red Flags

- Load mismatch
- Irregular geometry
- Soft storey
- Weak storey
- Torsion irregularity
- Overstressed members
- Boundary violations
- Fire escape issues
- Design vs drawing mismatches
- Soil–structure inconsistencies
- Safety violations

Yellow Flags

- Heavier foundations
- Unbalanced loads
- Economical alternative possible
- Material overconsumption
- Poor ductility elements

Green Flags

- Fully compliant
- Optimized
- No manual intervention required

24.8 INTERFACING WITH AUTHORITIES AND TYPICAL SUBMISSION FORMATS

Supported approval formats:

Municipal & HMDA/GHMC

- Standard submission drawings
- Parking statements
- FAR calculations
- Fire NOC requirements
- Rainwater harvesting notes
- Structural stability certificate draft

Industrial States (TSIIC, MIDC, SIPCOT, KIADB)

- Factory layout
- Utility corridors
- Crane paths
- Storage tank clearances
- EHS compliance

National Agencies

- NHAI
- CPWD
- MoRTH
- Railways
- Irrigation dept
- Pollution control boards

24.9 COMPLIANCE DASHBOARD

A visual panel showing:

- ✓ Compliant items
- ⚠ Needs review
- ✗ Non-compliant
- ☐ Pending inputs

Each element links to:

- Clause reference
- Calculation sheet
- Drawing sheet
- FEM result
- Remediation suggestions

24.10 SCALABILITY OF COMPLIANCE ENGINE

Supports ALL structure types:

- RCC Buildings (G+1 to high-rise)
- PEB / Steel Buildings
- Industrial Sheds
- RE Walls

- Retaining Walls
- Tanks
- Bridges / Culverts
- Landfills (benched, sloped)
- Warehouses
- Composite buildings
- Metro/rail structures (future)
- Water retaining structures
- Tunnels (future module)

24.11 ROLE OF AI IN COMPLIANCE

AI assists in:

- Auto-selecting applicable codes
- Clause-based validation
- Auto-mapping user sketches to structural grid
- Explaining failures with code references
- Suggesting alternate configurations
- Auto-optimizing sections
- Generating architectural corrections
- Creating approval notes and narratives
- Drafting checklists and QC documentation
- Auto-completing the final submission packet

📘 PART 25 — FULLY AUTOMATED BILLING, INVOICING & PAYMENT WORKFLOWS (ENGINEERING → EXECUTION → PROJECT CLOSEOUT)

(This part connects design, estimation, procurement, site execution, QS, client billing, statutory taxes, and project closure into a unified digital workflow.)

25.1 Purpose of this Module

This module transforms the entire **design-to-execution lifecycle** into a **financially accountable**, **auditable**, and **automated billing engine**, ensuring that:

- All design outputs (drawings, quantities, schedules) automatically flow into BOQ and billing packages.
- All estimation outputs flow into project cost baselines and financial dashboards.
- All site progress updates are converted into billable quantities (RA Bills / Progress Bills).
- All statutory tax components (GST / TDS / Cess) are computed automatically.
- All documents, measurements, audits, and approvals are auto-generated.

This makes the system as powerful as **SAP + Primavera + STAAD + CANDY + Tekla QS**, but with AI-driven automation and workflow intelligence.

25.2 Key Business Objectives

1. **Reduce Billing Cycle Delays**
   Auto-generate RA Bills, Material Reconciliation, and Client Invoices from site progress + BOQ.
2. **Ensure 100% Auditability**
   With auto-linked drawings, measurement sheets, photos, DPRs, and approvals.
3. **Prevent Revenue Leakage**
   Ensure every executed quantity is billed as per design, estimate, and contract conditions.
4. **Forecast Cashflow**
   Predict future billing, receivables, working capital, and cost overruns.
5. **Enable Multi-Role Approvals**
   Engineer → QS → PM → Cluster Head → Finance → Client → Internal Audit.
6. **Support EPC, Lump Sum, Item Rate, and Hybrid Contracts**
   Each with different billing rules and workflows.

25.3 Functional Scope

A. Billing Inputs (Auto-Collected)

- Design outputs (drawings, schedules, structural designs)
- Model-based quantities (from PEB/RCC models or FEA models)
- AI digitized BOQs (from PDFs or Excel)
- Site progress updates (DPR)
- Materials receipt / issue logs
- Field measurement books (AI-digitized)
- Photos, videos, drone scans (optional)

B. Billing Categories Handled

1. **Preliminary Works / Mobilization Advance**
2. **Item Rate BOQ Billing**

3. **Lump Sum Billing**
4. **Milestone Billing**
5. **Reimbursable Items Billing**
6. **Variation Orders (AI-assisted generation)**
7. **Price Adjustment / Escalation**
8. **Retention Deductions / Release**
9. **Final Bill & Project Closure**

25.4 Core Features

25.4.1 BOQ Engine Integration

- Auto-generate BOQ from:
  - Structural model (Kratos, FEniCSx, OpenSees-compatible)
  - Uploaded drawings/PDFs (AI extraction)
  - Manual entries
- Auto map BOQ items to:
  - SSR rates
  - Company rate library
  - Project-specific negotiated rates

25.4.2 Measurement & Progress Tracking Engine

Inputs:

- AI-extracted dimensions from drawings
- Site engineer measurements (mobile UI)
- Photos/videos with depth inference (optional)
- Drone scans (future addition)
- LIDAR/point cloud (future)

Outputs:

- Auto-computed measured quantities
- Variance vs BOQ
- Variance vs estimate
- Wastage factors
- AI anomaly detection

25.4.3 RA Bill Generator (Running Bills)

Each RA bill automatically generates:

- Abstract sheet (Item-wise totals)
- Detailed Measurement Sheet (DMS)
- Check Measurement Sheet (CMS)
- Material reconciliation sheet
- Indent reconciliations
- Tax sheet (GST breakup)
- Deduction sheet:
  - TDS
  - Retention
  - Advance recovery
  - GST-ITC adjustments

- Certification workflow

Outputs can be generated in:

- PDF
- Excel
- Word
- XML (for ERP)
- JSON API

25.4.4 Client Billing Package (Single Click)

Auto-composed package:

1. Cover letter
2. RA Bill
3. Measurement Books
4. Drawings used for measurements
5. Foundation/steel/concrete design approval sheets
6. Test reports
7. Material receipts
8. QC & HSE compliance sheets
9. Photos
10. Drone map (if any)

25.4.5 Cashflow Prediction Engine

Uses:

- PPS (planned progress schedule)
- Actual contracted value
- Site progress updates

Predicts:

- Monthly billing
- Monthly receipts (based on payment terms)
- Working capital requirement
- Cost-to-complete
- Variance and risk alerts

25.4.6 Variation Order (VO) Engine

Automatically detects need for VO:

- Additional structural members
- Change in soil conditions
- Changes in elevations, loads, wind/seismic data
- Revised drawings
- Client instructions
- Delays attributable to client

Generates:

- Justification note
- BOQ change
- Rate analysis
- Revised drawings
- Cost impact

- Delay impact (EOT)

25.4.7 Tax & Compliance Engine
Supports:

- GST (CGST/SGST/IGST)
- TDS
- Labour Cess
- Royalty/Seigniorage
- PF/ESI for labour billing
- Reverse charge mechanism
- GST input credit reconciliation

25.4.8 Integration with Other Modules

- **Design Engine** → structural outputs → BOQ
- **Estimation Engine** → baseline cost
- **Procurement Engine** → material rates
- **Inventory Engine** → material consumption
- **Workflow Engine** → approvals
- **Project Control Engine** → time & cost S-curves
- **Drawing Engine** → PDFs linked to DMS

25.5 System Architecture

- All billing elements stored in unified **Project Data Model (PDM)**
- Billing engine runs as microservice with:
  - Billing engine API
  - Measurement engine API
  - Tax engine API
  - VO analytics engine
- Model supports:
  - Multi-project
  - Multi-client
  - Multi-contract
  - Multi-currency
- Drawings + photos stored in:
  - Firebase / S3 / GCP storage

25.6 AI/ML Capabilities

- AI detects missing measurements.
- AI reads hand sketches and converts to measurement sheets.
- AI flags unusually high/low consumption.
- AI predicts escalation impacts.
- AI generates client-ready billing narratives.

25.7 UI/UX Requirements

- Bill creation wizard
- Measurement viewer (side-by-side drawing + quantities)
- 3D model overlay (if design engine used)

- Billing dashboard:
    - RA bill status
    - Payment status
    - VO status
    - Tax calendars
- Printable formats (Excel/PDF)

25.8 Data Model (Simplified)

Entities:

- Project
- Contract
- BOQ Item
- Measurement
- RA Bill
- Material Receipt
- Deductions
- Approvals
- Attachments
- TaxRecords
- PaymentStatus

Relations:

- Project → Many Contracts
- Contract → Many BOQ items
- BOQ → Many Measurements
- RA Bill → Many BOQ quantities
- RA Bill → Many Attachments
- Payment → RA Bill

25.9 Workflows

Site → QS → PM → Finance → Client → Audit

- Every step recorded.
- Audit trail immutable.
- Versioning for RA bills (v1, v2, v3).

25.10 Deliverables

System will generate:

- RA Bills
- Abstract Sheets
- Measurement Books
- Final Bill
- Project Closure Report
- As-built quantities
- Tax computation schedules
- Material reconciliation
- VO documents

25.11 KPIs

- Billing lag (days)
- Bill acceptance ratio
- Reconciliation accuracy
- Revenue leakage
- Cashflow predictability
- Material reconciliation variance
- VO realization %

25.12 Success Outcomes

- RA bills generated in **minutes**, not days.
- Zero missed measurements.
- Automated compliance.
- Real-time cost insights.
- Minimized disputes.
- Faster cashflow.

✔ **PART 25 Completed.**

📘 PART 27 — PROJECT EXECUTION MANAGEMENT MODULE

(A Unified End-to-End System for Planning, Scheduling, Tracking & Closing Construction Projects)

27.1 PURPOSE OF THIS MODULE

This module manages the entire **project execution lifecycle** after design, approval, and estimation.

It ensures that the user (Engineer / Contractor / Project Manager) can:

- Create a complete execution plan
- Develop resource-linked schedules
- Track progress onsite
- Manage issues, risks, & changes
- Monitor cost vs time
- Manage documentation, QA/QC, safety, labour, and materials
- Drive the project to commissioning & handover

This module is central to converting **design** → **execution** → **handover**, especially for the G+2 building example or industrial/PEB/RCC/landfill/retaining wall projects.

27.2 WORKFLOW OVERVIEW

1. **Project Initialization**
    - Import design → BOQ → Costs → Drawings
    - Set project type (Building/PEB/Industrial/Landfill/etc.)
2. **Work Breakdown Structure (WBS) Generation**
    - Auto-generated using AI based on project type
    - User can modify any activity or add their own
    - Linked with codes of account (COA)
3. **Planning & Scheduling**
    - Auto-generate baseline schedule
    - AI-recommended durations and dependencies
    - User edits timeline (drag-drop Gantt)
4. **Resource Allocation**
    - Labour, materials, equipment
    - Auto-pull from BOQ and SSR analysis
    - Alerts for shortages, conflicts, resource overuse
5. **Execution Tracking**
    - Daily logs
    - Material updates
    - Labour attendance
    - Equipment utilization
    - Actual progress vs planned (S-curve)
6. **Quality, Safety, and Checklists**
    - QA/QC forms
    - HSE checklists
    - Photo-tagging
    - NCR management
7. **Issue & Risk Management**
    - Issue log

    - Root-cause mapping
    - Risk scoring
    - Mitigation workflow
8. **Change & Variation Management**
    - Variation item builder
    - AI detects scope changes from drawings vs site
    - Auto-priced variations
9. **Billing & Cost Control**
    - RA bills / IPC construction progress bills
    - Joint measurements
    - Certified vs uncertified quantities
    - Cost-at-completion forecasts
10. **Documentation & Handover**
    - Indexing of all documents
    - Auto-generated completion report
    - QA documents
    - Drawings and as-built records
    - O&M manuals
    - Final handover submission pack

27.3 KEY FEATURES (FULL LIST)

27.3.1 Automatic WBS Generator

- Based on project type
- Auto-expand into:
    - Earthwork
    - Foundations
    - Columns & beams
    - Slabs
    - Walls
    - Staircase
    - Plinth beam
    - Lintel
    - Finishing
    - MEP
    - Testing & commissioning
- User can add/delete/modify WBS nodes
- Stored as template for future projects

27.3.2 AI-Generated Baseline Schedule

- Reads WBS
- Applies default durations using project library
- Auto-creates:
    - Dependencies (FS, SS, FF)
    - Overlaps (lag/lead)
    - Critical path
- Visual Gantt chart
- Export to PDF, Excel, Primavera XER, or MS Project XML

27.3.3 3D-Linked Execution Model (FROM PART 8 VISUALIZATION ENGINE)

- User sees 3D model
- Selecting an element highlights related WBS items
- Progress marking:
  - "Started"
  - "25%"
  - "50%"
  - "Complete"
- Photos attach to elements
- Daily progress updates appear on the 3D timeline

27.3.4 Resource Planning Engine

Auto-generated based on:

- BOQ
- SSR analysis
- Equipment requirement logic
- Labour productivity data
- Material consumption patterns

Outputs:

- Resource histogram
- Labour utilisation curves
- Equipment calendar
- Material procurement schedule

27.3.5 Daily Progress Reporting (DPR) Module

- Weather
- Work done
- Labour returns
- Equipment log
- Material receipts
- Delays & reasons
- Photos/video upload
- Auto-summary generated by AI

27.3.6 Cost & Billing Module

- Joint measurement sheets
- Deductions & extra items
- RA Bills / IPC billing
- Material reconciliation
- Cash flow forecasting
- Cost-at-completion (CAC) algorithm
- Budget vs actual dashboard

27.3.7 QA/QC Management

- AI-generated checklists
- For RCC, masonry, PEB, finishing, MEP

- Cube test tracking
- Material test logs
- Photo evidence
- NCR system:
  - Non-conformance logging
  - Corrective-action workflow
  - Closure digital signatures

27.3.8 HSE & Site Safety

- PPE monitoring
- Toolbox talks
- Incident & accident logging
- Near-miss capture
- HIRA (Hazard Identification & Risk Assessment) register
- Safety audit reports
- Safety score dashboard

27.3.9 Issue & Risk Management

- Issues tagged to WBS, drawings, or BOQ
- Priority scoring
- AI suggests mitigation plans
- Risk heat map
- Escalation matrix

27.3.10 Change Order & Variation Management

- Compare:
  - Latest drawing vs previous revision
  - Site photos vs design
  - BOQ vs actual consumption
- AI identifies:
  - Possible variation items
  - Increase/decrease in scope
- Auto-pricing based on SSR + contractor margins

27.3.11 Collaboration & Communication

- Comment threads per element
- Task assignments
- Notifications
- Chatbot assistance
- Approvals workflow:
  - Site engineer → PM → Client → Head office

27.3.12 Documentation & Handover

- Stores:
  - Drawings
  - Reports
  - QA/QC records

    - Approvals
    - Photos
    - Bills
    - Variation orders
- Auto-compiled handover file:
    - PDF
    - ZIP
    - Structured folders

27.3.13 Analytics & Dashboards

- S-curve (Planned vs Actual)
- Earned Value Management (EVM)
- Cost vs time
- Labour productivity
- Safety score
- QA score
- Delay reasons categorised by AI

27.4 TECHNICAL ARCHITECTURE

Frontend

- React or similar
- Gantt chart library (like Syncfusion or DHTMLX)
- 3D viewer (from Part 8 — same engine, reused here)

Backend

- FastAPI / Python
- Or Node.js option (depending on your final stack)
- AI workflows (OpenAI/Gemini/Azure)
- Postgres database
- Redis for task queue
- Worker system for long processes (Celery / RQ)

Data Models

- Project
- WBS
- Schedule
- Resource table
- Material log
- Labour log
- Equipment log
- QA/QC records
- HSE records
- Issue log
- Risk log
- Variations
- Bills
- Documentation

27.5 INTEGRATION POINTS

With Design Engine

- Extract members → generate WBS
- Auto-prepare foundation & structural works

With Estimation Module

- BOQ → schedule → resource planning
- Rates → cost baseline

With 3D Visualization Engine

- Progress tracking
- As-built documentation
- Clash detection

With Drawing Engine

- Revisions trigger variation alerts
- Site vs design comparisons

27.6 AI FEATURES UNIQUE TO THIS MODULE

- Auto-generate WBS
- Auto-schedule with durations & dependencies
- Auto-cost forecast
- Auto-variation identification
- AI-generated daily/weekly/monthly reports
- AI-driven risk heat map
- AI summarises:
    - Delay causes
    - NCRs
    - Billing deviations
    - Cost overruns

27.7 OUTPUTS OF THIS MODULE

1. Complete WBS
2. Baseline schedule + revised schedules
3. Resource schedules
4. DPRs
5. RA bills
6. Material reconciliation reports
7. QA/QC & HSE reports
8. Variation orders
9. EVM dashboards
10. Final handover packet
11. As-built documentation

PART 28 — PROJECT EXECUTION & CONSTRUCTION MANAGEMENT ENGINE
This module activates **after design + drawings + estimation** are finalized OR immediately if the user directly selects “Execution Only”.
The module mirrors real-world engineering workflows and construction-site processes and is fully integrated with the rest of the platform.

28.1 Purpose & Scope
The **Project Execution & Construction Management Engine** transforms all upstream outputs—architectural layouts, structural designs, detailed drawings, BOQs, and project requirements—into an actionable, continuously updating execution workflow.
It serves as an **end-to-end digital site management ecosystem**, covering:

- **Project planning & scheduling (WBS + Gantt)**
- **Task sequencing & auto-dependency creation**
- **Work packages / deliverables**
- **Issue tracking (NCRs, deviations, rectifications)**
- **Site logs & daily progress**
- **Material planning, indents, consumption & reconciliation**
- **Contractor management & measurements**
- **Billing & RA bills**
- **Quality & HSE management**
- **Checklists, inspections & approvals**
- **Document management & revisions**
- **Completion & handover automation**

Designed for engineers, supervisors, PMPs, and quality/HSE officers.

28.2 Inputs to This Module
The engine consumes:
a) From Architecture

- Floor plans, elevations, sections
- Room schedules & dimensions
- Required finishes per space type

b) From Structural

- Column/beam/slab/wall sizes
- Reinforcement drawings
- Foundation layouts
- Staircase drawings
- Retaining wall plans
- Water tank details
- Plinth beam/lintel/roof beam layouts

c) From Services (if included)

- Electrical layouts
- Plumbing layouts
- Fire-fighting layouts
- HVAC layouts

d) From Estimation

- Activity-wise BOQ
- Material requirements

- Labour/equipment requirements
- Rate analysis

e) User Inputs

- Start date
- Project priority level
- Resource availability
- Payment milestones
- Contractor list
- Regional constraints (monsoon, permits, local compliance)

28.3 Core Functional Objectives

1. Automatic Project Breakdown (AI-driven WBS Generation)

- Auto-generate a full WBS depending on building type / industrial layout.
- WBS levels:
  L0 Project → L1 Substructure → L2 Foundations → L3 Footing type → L4 Reinforcement → L5 Concreting
- Complies with IS codes + industry practices.
- Editable by user with drag-and-drop.

2. Gantt & Schedule Generation

- Automatic CPM (Critical Path Method)
- Auto dependencies:
  - Footing → Column → Plinth → Blockwork → Slab → Finishes
- Resource-level schedules:
  - Labour gangs
  - Equipment days
  - Material availability

3. Daily Progress & Site Logs

- Digital DPR (Daily Progress Report)
- Track:
  - % completion per activity
  - Labour attendance
  - Material delivered & consumed
  - Equipment usage
  - Photos, GPS-stamped
  - Weather conditions auto-import

4. Material & Inventory Management

- Material planning per activity
- Auto indents based on schedule
- GRN (Goods Receipt Note) workflow
- Material testing entries
- Site stock ledger
- Reconciliation module:
  - Theoretical vs actual consumption
  - Shrinkage & wastage factors
  - Alerts if >5% deviation

5. Contractor & Measurement Management

- Track individual contractor work packages

- Record measurements with formulas (IS 1200 rules)
- Auto-generate MB (Measurement Book)
- Track running bills (RA bills)
- Payment recommendations

6. Quality & HSE (QHSE)

- Quality checklists for:
  - Excavation, PCC, shuttering, steel, concreting, curing, masonry, plastering, painting, flooring
- NCR (Non-Conformance Report) workflow:
  - Capture → Assign → Rectify → Close
- HSE registers:
  - Tool box talks
  - PPE compliance
  - Safety incidents
  - Near misses

7. Issue Tracking & Escalation

- "Raise an issue" button on any drawing or activity
- Auto-routing to:
  - Structural Engineer
  - Architect
  - MEP consultant
  - Contractor
- SLA timers
- Reminder & escalation hierarchy

8. Document Management

- Revision tracking
  (R0 → R1 → R2... with cloud stamping)
- Linking drawings to WBS items
- Version-based approval workflow:
  - Draft → Review → Approved for Construction (AFC)

9. Cost Control & Budget Management

- S-curve generation
- Earned Value Management (EVM)
  - EV, PV, AC, CV, SV, CPI, SPI
- Cashflow forecasting
- Budget vs actual tracking by cost head
- Variation order workflows

10. Handover & Closure

- Snag list generation
- Final BOQ reconciliation
- Final bills
- As-built drawings
- Handing over documents
- Digital completion certificate

28.4 System Capabilities

A. 3D Interactive Model Linking

The model created in earlier parts is fully integrated:

- Engineers can navigate **3D model**
- Click any element (column/beam/footing/slab/wall)
  → show:
    - Activity status
    - Material consumption
    - Issues/NCRs
    - Drawings linked
    - Schedules
    - Photos taken during execution

B. Cross-sections & Layer Views

Like STAAD / Revit:

- Hide/show layers
- Show reinforcement only
- Show formwork stages
- Show executed vs planned overlays

C. Voice & Chat-driven AI Guidance

- "Create schedule for G+2 building in Miyapur."
- "Generate concrete quantity for footing F1."
- "Upload site photo and classify the work progress."

28.5 Integrations

1. With Design Engine

- Auto-fetch element sizes & details

2. With Estimation Module

- Activity-wise BOQ → schedule generation

3. With SSR & Market Rates

- Auto-update cost deviations

4. With Procurement Module

- Material indents auto-raised
- Procurement-to-site tracking

5. With QHSE & Inspection Modules

- Checklists triggered by activity status

28.6 Data Output

The module produces:

- Full WBS + schedule (PDF / Excel / App)
- Activity-level progress charts
- Material logs
- DPR books
- MB register
- RA bill summaries
- Quality & HSE reports
- Snag list & closure reports
- As-built drawings
- Handover dossier

28.7 User Roles & Permissions

- **Client**
  - View progress, issues, financial status
- **Project Manager**
  - Full control
- **Site Engineer**
  - DPR, measurements, material
- **Quality Engineer**
  - Checklists, NCRs
- **Safety Officer**
  - HSE logs
- **Contracts/Billing**
  - Bills, approvals
- **Procurement**
  - Indents, GRN

28.8 Automation & AI Add-ons

- Auto-detect progress from **site photos**
  (slab formwork, footing stage, reinforcement progress)
- Auto-generate **monthly reports**
- Predict delays with ML models
- Recommender system for resource optimization
- Forecast material shortages
- Weather-adjusted scheduling

28.9 Future Scope

- Drone-based progress scanning
- BIM Level 2 integration
- LIDAR-based measurement validation
- Full IFC bi-directional sync
- Automated clash detection

📘 PART 29 — MODEL VALIDATION, QUALITY CONTROL & APPROVAL WORKFLOW ENGINE

This section defines the complete validation, verification, QA/QC, peer-review, design audit, and approval process for all deliverables generated inside the platform — architectural, structural, MEP, estimates, BOQs, drawings, reports, models, and schedules.

This is essential for:

- Government authority submission (GHMC, HMDA, DTCP, BMC, KMC, TN-DTCP, AP-DTCP, KIADB, etc.)
- Third-party design checks (IIT/NIT consultants, Proof Check Agencies)
- Internal engineering QA/QC cycles
- Industrial client submissions (BHEL, NTPC, ONGC, L&T, TATA, Reliance, etc.)
- Contractual approvals (GFC drawings, AFC drawings, manufacturing drawings)

29.1 OBJECTIVE

The Approval Workflow Engine ensures that **every architectural, structural, and civil engineering output** generated by the app is:

✓ Code compliant

✓ Properly modelled

✓ Peer-reviewed

✓ Validated against design standards

✓ Audit-tracked

✓ Approved or rejected with comments

✓ Version-controlled

It applies to **every discipline**:

- Architecture (layouts, plans, sections, elevations)
- Structural (models, analyses, designs, drawings)
- Estimation (BOQs, SSR-based costing, rate analysis)
- Construction (method statements, schedules, quality plans)

29.2 VALIDATION TYPES

The platform supports **five layers of validation**:

29.2.1 Geometric Validation (GEO-CHECK)

Validates raw geometry coming from:

- User input
- Uploaded CAD/PDF
- AI-generated architectural layout
- Sketch-to-plan conversion
- Imported STAAD/ETABS/REVIT/IFC models

Checks include:

1. Dimensional correctness
2. Logical alignment (beam-to-column connectivity, slab boundary continuity)
3. Storey alignment & stability
4. Wall intersections & load-bearing path
5. Openings validation (doors, windows, shafts)

6. Stair & lift geometry conformity
7. Sanity checks (zero-length members, duplicate nodes, overlapping beams)

29.2.2 Engineering Validation (ENG-CHECK)

Ensures engineering data is correct:

- Materials (concrete grade, steel grade, densities)
- Section properties (IPE, ISMB, HSS, custom welded profiles)
- Boundary conditions (supports, connections)
- Load assignments (dead, live, wind, seismic, temperature)
- Code selection (IS 456, IS 800, IS 875, IS 1893, ACI 318, AISC, Eurocodes)
- Load combinations (auto generation per selected code)

29.2.3 Analysis Validation (ANALYSIS-CHECK)

Verifies that the FEA model is:

- Stable
- Determinate/indeterminate as expected
- Well-conditioned mesh
- Properly constrained
- Free from singularities
- Free from oscillating DOFs
- Free from unrealistic displacements
- Reaction balancing OK
- Modal checks OK
- Load path is continuous

Special tools included:

- **Mesh Quality Checker** (skewness, aspect ratio, warped faces)
- **Model Stability Analyzer** (zero reaction DOF detection)
- **Mode Shape Visualizer**

29.2.4 Design Validation (DESIGN-CHECK)

Performs automated design code conformance:

RCC

- IS 456 / ACI 318 / Eurocode 2 compliance
- Slab & beam reinforcement control
- Column interaction curves
- Punching shear
- Foundation stability & bearing pressure

Steel

- IS 800 / AISC / Eurocode 3 compliance
- Cross-section classification (compact/semi/slender)
- LTB checks
- Axial + bending interaction
- Connection design checks

Composite

- Shear studs
- Decking design
- Composite beam checks

Geotechnical

- Allowable bearing
- Sliding, overturning
- RE wall checks
- Soil-structure interaction

29.2.5 Drawing & Documentation Validation (DOC-CHECK)

Ensures all deliverables:

- Have correct title blocks
- Proper scale
- Dimensions & annotations
- Cross-sections linked to plan
- Drawing list & revision control
- Compliant with drafting standards (IS 962, company CAD standards)
- Detail callouts properly referenced
- Clash detection between Architecture–Structure–MEP

29.3 APPROVAL WORKFLOW

The app supports **multi-level hierarchical approvals**, customizable for:

- Corporates
- Government
- Consultants
- EPC companies
- Individual engineers

29.3.1 Workflow Stages

1. **Draft** → Auto-generated by AI/Modules
2. **Model Validation** → GEO-CHECK
3. **Engineering Validation** → ENG-CHECK
4. **Analysis Validation** → ANALYSIS-CHECK
5. **Design Validation** → DESIGN-CHECK
6. **Documentation Validation** → DOC-CHECK
7. **Peer Review** (internal team)
8. **Lead Engineer Review**
9. **Principal Engineer / Chief Engineer Approval**
10. **Third Party Proof Check (optional)**
11. **GFC/IFC Issue**
12. **AFC revision cycle**
13. **Construction Package Release**
14. **Archive & Version Lock**

Each stage has:

- Approver role
- Timestamp
- Comments
- Mandatory attachments
- Revisions

29.4 AUTOMATED APPROVAL SUPPORT

The system includes tools to simplify approvals:

29.4.1 Auto-Explain Model

AI explains:

- Load path
- Beam-column layout
- Foundation requirements
- Stability mechanisms

Useful for junior engineers.

29.4.2 Auto-Explain Calculations

Generates handwritten-style or examiner-style detailed:

- Code clauses
- Step-by-step calculations
- Derivations
- Shear/moment diagrams
- Interaction ratios

29.4.3 Auto-Flagging

AI flags:

- Overstressed components
- Invalid geometry
- Unsupported slabs
- Wrong support assumptions
- Missing loads

29.4.4 Auto-Optimization Suggestions

Before approval, system recommends:

- Lighter sections
- Revised column grid
- Revised footing sizes
- Value engineering options

29.5 VERSION CONTROL & AUDIT

Every model, drawing, and calculation is versioned:

- V1.0 → V1.1 → V1.2 …
- Revision cloud auto-generated
- Change history
- Change reason
- Person who made change
- Auto-mapped effects (e.g., column moved 300 mm → beams updated)

Support for Git-style versioning via:

- Commit
- Pull request (PR)
- Merge
- Compare

29.6 ROLE-BASED APPROVAL MATRIX

Roles include:

- User / Owner
- Draft Engineer
- Design Engineer
- Checker
- Lead Structural Engineer
- Architect
- QS Engineer
- Construction Manager
- Client Representative
- Government Reviewer

Each role has:

- Rights (view/edit/approve)
- Commenting privileges
- Signature authority

Digital signature support for:

- eSign
- DSC token
- QR-verified approval seal

29.7 AUTHORITY-SUBMISSION PACKAGE GENERATION

Automatically prepares submission-ready documents:

- Architectural drawings as per GHMC/HMDA/DTCP
- Structural safety certificate formats (Form-A, Form-B as per NBC)
- Soil test reports
- Stability certificate
- Fire drawings (if required)
- Sanction drawing sets
- Calculation reports
- Structural design summary
- Foundation design report
- Estimation sheets
- All drawings in A1/A2/A3 format
- DXF/DWG + PDF export

29.8 AI-POWERED REVIEW ASSISTANT

An AI assistant specifically for reviewers:

- Summarizes the entire design
- Highlights risks
- Suggests improvements
- Checks code compliance
- Classifies the design as **Accept / Reject / Revise**
- Performs automated clash detection
- Identifies potential construction issues

29.9 DESIGN FREEZING

Once the design is approved:

- Model becomes “Read-Only”
- Drawings are frozen
- Version is “Archived”
- All linked BOQs are locked
- Any change triggers a **Change Request** workflow

29.10 MOBILE APPROVALS

Tappable interface for:

- Review
- Comments
- Mark-ups
- Zooming 2D/3D drawings
- Viewing calculations and PDFs
- Approve/Reject with signature

29.11 FINAL OUTPUTS

The module generates an all-in-one:

- Approval History
- Comments Trail
- Final Stamped PDFs
- 3D IFC model export
- Data Book Package
- Editable CAD/DXF and Revit-compatible IFC
- Full design report
- Full calculation sheets
- Fabrication/Construction-ready drawing set

📘 PART 30 — MULTI-STRUCTURE ANALYSIS ENGINE (MSAE) — CORE COMPUTATION LAYER

This part defines the **core analytical computation engine** that supports *all* structure types across the app — RCC, Steel, Composite, Retaining Structures, RE Walls, Landfills, Bridges, Industrial Sheds, and future modules.

It is the computational backbone inspired by STAAD / ETABS / SAFE / RAM / PLAXIS but built with:

- **KRATOS Multiphysics** (primary solver)
- **FEniCSx** (secondary solver for continuum + FEM-based research workflows)
- **Custom IS-code design modules** (IS 456 / 800 / 875 / 1893 / 3370 / 7810 etc.)
- **Universal model abstraction layer** for interoperability with STAAD/ETABS/SAP

This part establishes requirements for the **mathematical engine, modeling abstractions, solver pipeline, physics coupling, and result standardization**.

30.1 PURPOSE

The Multi-Structure Analysis Engine (MSAE) shall:

1. Handle the **full lifecycle** of computational engineering:
    - Preprocessing → Meshing → Load Application → Boundary Conditions
    - Solver Execution → Post-processing → Structural Design → Detailing Integration
2. Support **any structural system**, not limited to buildings:
    - RCC Buildings
    - Steel Buildings & PEB
    - Composite Floors
    - Retaining Walls
    - RE Walls
    - Landfills & Geostructures
    - Tanks, Silos, Reservoirs
    - Trusses, Frames, Bridges
    - Industrial Equipment Foundations
    - Soil–Structure Interaction (future)
3. Provide **consistent data structures** so all modules of the app can plug into and retrieve:
    - Forces
    - Moments
    - Stresses
    - Displacements
    - Modal responses
    - Soil pressures
    - Stability factors
    - Crack widths (RCC)
    - Steel stresses
4. Manage **multi-solver** pathways:
    - Kratos Multiphysics (primary)
    - FEniCSx (secondary)
    - Custom Hand-calculation Modules (fallback)
    - GPU-accelerated modules (future)

30.2 HIGH-LEVEL ARCHITECTURE

The engine consists of 8 layers:

1. **Input Translator & Semantic Understanding Layer**
2. **Universal Model Schema (UMS)**
3. **Geometry Kernel**
4. **Meshing & Discretization Layer**
5. **Physics Modules (Materials & Constitutive Models)**
6. **Solver Pipeline (Kratos/FEniCSx)**
7. **Post-Processing & Result Standardization**
8. **Design-Check Engine (IS Codes)**

30.3 LAYER 1 — INPUT TRANSLATOR & SEMANTIC UNDERSTANDING

Inputs may come from:

- Text prompts ("Construction of G+2 building in Hyderabad")
- User-filled forms
- Uploaded plans / sketches (image → vector extraction)
- DXF / DWG / IFC
- Imported STAAD / ETABS files
- Pre-stored templates ("Sample RCC house plan")

This layer performs:

- NLP interpretation → "structure type", "loads", "elements", "dimensions"
- Missing-value queries → "AI clarifies inputs"
- Geo-tagging → fetches wind/seismic values from IS 875/1893
- Converts everything into **UMS-compliant objects**

30.4 LAYER 2 — UNIVERSAL MODEL SCHEMA (UMS)

Defines the standard representation for all structures.

UMS contains:

30.4.1 Core Entities

- Nodes
- Elements
- Materials
- Sections
- Supports
- Load Definitions
- Boundary conditions
- Connections
- Groups (Floors, Frames, Bays)

30.4.2 Element Families

- 0D: Point supports, springs
- 1D: Beams, Columns, Bracings, Trusses
- 2D: Plates, Shells, Shear Walls
- 3D: Solids (for raft, soil blocks, retaining walls, landfill mass)

30.4.3 Structural System Definitions

- Slab systems
- Staircases

- PEB rafters & columns
- Composite deck profile
- Landfill berm profiles
- RE wall reinforcement bands

30.4.4 Load Families

- Gravity (DL, LL)
- Wind (IS 875 Part 3)
- Seismic (IS 1893)
- Soil pressures
- Water pressures
- Thermal
- Pre-stress loads (future)
- Time-history input (future)

30.4.5 UMS Output Guarantees

Every solver returns:

- Node displacements
- Element forces
- Stress tensors
- Reaction forces
- Instability warnings
- Mesh quality indicators
- Natural frequencies
- Mode shapes
- Soil settlement predictions

30.5 LAYER 3 — GEOMETRY KERNEL

Features:

- Parametric modeling
- Extract geometry from:
    - Hand sketches → vectorization
    - Uploaded images → blueprint AI
    - DXF/DWG
    - IFC
- Boolean operations for:
    - Trimming beams at intersections
    - Slab openings
    - Stair shafts
    - PEB knees
- Automatic vertical alignment of:
    - Columns
    - Beam-slab mating
    - Wall–slab intersections
- 3D visualization:
    - Orbit
    - Cross-section slicing
    - Layer visibility toggle
    - STAAD-like geometry interaction

    - The user can **select, drag, edit, delete, duplicate** any element

---

30.6 LAYER 4 — MESHING & DISCRETIZATION
Supported Mesh Types
- 1D line meshes for frame analysis
- 2D tetra/quad meshes for slabs, walls
- 3D solid meshes for soil, landfill, rafts

Automatic Meshing Rules
- Mesh refinement controlled via UI slider
- Error-driven refinement suggestions
- Auto-mesh around openings and stress concentrations
- Auto-sweep meshing for floor systems
- Support for:
    - Structured meshes
    - Unstructured meshes
    - Adaptive mesh refinement (AMR) — future

30.7 LAYER 5 — MATERIAL & PHYSICS MODULES
Material Library
- RCC (IS 456 grades M20–M60)
- Steel (IS 800 + AISC shapes)
- Composite Steel Deck
- Masonry
- Soil (Mohr–Coulomb)
- Rock (Hoek-Brown)
- Landfill waste material

Physics Modules
- Elastic
- Elasto-plastic
- Nonlinear geometry
- Creep & shrinkage
- Soil-structure interaction
- Seepage (for retaining walls/landfills)
- Thermal expansion

30.8 LAYER 6 — SOLVER PIPELINE
Primary Solver: Kratos Multiphysics
Modules used:
- StructuralMechanicsApplication
- SolidMechanicsApplication
- GeoMechanicsApplication
- FluidMechanics (for tank hydrodynamics)
- ContactMechanics (for soil–structure interface)

Secondary Solver: FEniCSx
Used for:
- Research-grade FEM
- Custom element formulations

- PDE-based formulations

Solver Routines

- Static linear
- Static nonlinear
- Modal
- Response spectrum
- Pushover (future)
- Time history (future)
- SSI with soil blocks
- Extreme event loads

30.9 LAYER 7 — POST-PROCESSING & RESULT FORMATS

Outputs

- Node displacements
- Element end forces
- Stress tensors
- Crack width reports (RCC)
- Slab reactions
- Wall stability
- Landfill settlement
- Foundation bearing pressures
- 3D stress-cloud visualization
- Vector diagrams (shear, moment, axial)
- Live 3D interaction like STAAD:
    - Pan
    - Orbit
    - Section cut
    - Deformation shape

Export formats

- STAAD .std
- ETABS .xlsx
- SAFE .f2k
- IFC
- DXF plan, elevation, section
- PDF summary report
- Detailed Excel calculation sheets

30.10 LAYER 8 — CODE-BASED DESIGN ENGINE (IS CODES)

After analysis, the design module auto-runs:

RCC

- IS 456
- IS 13920
- Slab, beam, column, footing

Steel

- IS 800 (Limit State)
- IS 875
- IS 801 (light gauge)

- AISC (optional)

Composite

- IS 11384
- Eurocode 4 fallback

Retaining Walls

- Stability
- Structural design
- IS 456 + IS 1904

Landfill

- CPHEEO
- Stability (FOS)
- Settlement prediction

30.11 PERFORMANCE TARGETS

- Up to **200,000 DOF** per model
- Solve typical RCC building (G+10) in < 10 sec (default mesh)
- Solve PEB frame in < 3 sec
- Handle landfill slope (3D) < 30 sec
- GPU acceleration optional

30.12 FAIL-SAFE CALCULATORS

If full FEM fails, fallback options should run automatically:

- Hand-calculations
- Simplified strut-and-tie models
- Simplified truss behavior for frames
- Rankine/Coulomb for soil
- Approximate load calculators

30.13 LICENSING & EXTENSIBILITY

Allow external developers to:

- Create custom elements
- Create custom design modules
- Export data into KRATOS-ready JSON

30.14 SUMMARY

This engine is the **heart** of the entire app.
It ensures the system can:

- Model anything
- Solve anything
- Design everything
- Export standard drawings
- Present results like STAAD/ETABS/SAP
- Scale massively

It unifies all structure types into **one computational platform**.

📘 PART 31 — MULTI-STRUCTURE FINITE ELEMENT ENGINE (FEA CORE) – KRATOS + DOCKER + AI LAYER

This part defines the **core computational engine** that powers *all* structure types—PEB, RCC, Steel Buildings, Industrial Structures, Retaining Walls, RE Walls, Landfills, Tanks, Silos, Footings, etc.—using:

- **KRATOS Multiphysics** (primary FEA solver)
- **Dockerized execution**
- **AI-assisted model generation**
- **3D interactive visualization layer** (STAAD-like interface)
- **Unified model schema for geometric, material, and load data**

This engine is the "heart" of the entire product.

31.1 PURPOSE OF THE KRATOS FEA CORE

The FEA Core must enable:

✅ **1. Automated Model Creation (AI → Geometry → Mesh → Solver)**

- Converts user intent & sketches → structured model
- AI identifies members, loads, slabs, foundations, retaining walls, tanks, etc.
- Converts architectural plan → structural grid → analysis model

✅ **2. Multi-physics support**

- **Structural mechanics**
- **Geotechnical (soil–structure interaction)**
- **Fluid interaction (tanks, STPs)**
- **Thermal coupling (roofs, chimneys)**

✅ **3. Multiple element technologies**

- 1D frame/beam elements
- 2D shell/plate/wall elements
- 3D brick/solid elements
- Interface/spring elements
- Joint/nonlinear links

✅ **4. Standalone Dockerized Solver**

- No OS conflict
- All dependencies pre-installed
- AI Studio/Antigravity compatible
- Runs on cloud or local with same behavior

31.2 ARCHITECTURE OVERVIEW

```
AI INTERFACE LAYER
(Natural Input → Structured Model Data)
        |
        ▼
MODEL BUILDER (Python API Wrapper Layer)
- Geometry generator
- Material & section library
- Load definitions
```

```
- Boundary conditions
        │
        ▼
KRATOS MULTIPHYSICS SOLVER
- StructuralMechanicsApplication
- GeoMechanicsApplication
- FluidStructureInteraction
- SMOOTHED PARTICLE HYDRODYNAMICS (for landfills if needed)
        │
        ▼
POST PROCESSOR + EXTRACTOR
- Displacements, stresses, reactions
- Soil pressures, bending moments, shear forces
- Drift + capacity checks
        │
        ▼
3D VISUALIZATION ENGINE
(Three.js / Babylon.js + cross-section views)
STAAD-like: modify nodes, members, loads
        │
        ▼
DESIGN ENGINE (ALL CODES)
IS 800, IS 456, IS 875, IS 1893, IS 2911, IS 3370, ACI, EC
Code checks + member design + foundation design
        │
        ▼
DRAWING & ESTIMATION ENGINE
Detailed drawings, BOQs, estimation, SSR-based costing
```

31.3 INPUT TYPES THE ENGINE MUST ACCEPT

A. High-level natural language

"G+2 building in Hyderabad, 5m x 4m grid, soil SBC 200 kPa…"

AI converts → structured model.

B. Sample building layout templates

- Pre-stored plans
- Standard industrial grids

C. User plan upload

- Hand sketch (image)
- PDF
- CAD/DWG
- SVG
- IFC/BIM

D. Manual STAAD-like modeling

- Add nodes
- Add beams
- Add plates
- Add solid elements
- Add loads

E. Import from other software

- STAAD (.STD)
- ETABS (.E2K)
- SAP2000 (.S2K)
- IFC models
- DXF

F. Modify geometry directly in 3D

- Drag nodes
- Edit member properties
- Change orientation
- See cross-sections in 3D

31.4 MODEL GENERATION VIA AI

AI must interpret:

1. Loads

- IS 875 Part 1 – Dead
- Part 2 – Live
- Part 3 – Wind
- Part 5 – Load combinations
- Seismic zone → IS 1893 loads

2. Component identification

Detected from drawing:

- Columns
- Beams
- Slabs
- Shear walls
- Staircase
- Lift wall
- Plinth beam
- Lintel
- Footing types
- Pile vs Isolated
- Water tank vs Sump
- Boundary wall
- Road pavement

3. Structural system selection
AI suggests optimal system:

- RCC frame
- Steel frame
- Composite
- Precast
- Load-bearing

31.5 KRATOS SOLVER CONFIGURATION

A. Applications used

Component	Kratos Application
RCC & Steel	StructuralMechanics
Soil pressure/walls	GeoMechanics
Landfill stability	GeoMechanics + contact
Water tank design	FluidDynamics + FSI
Pavements	StructuralMechanics
Foundations	GeoMechanics

31.6 ELEMENT TECHNOLOGIES

1D Elements

- Frame
- Truss
- Beam-column
- Compression-only
- Tension-only

2D Elements

- Shells (DKM, MITC)
- Plates
- Plane stress/plane strain
- Wall elements

3D Elements

- Hexahedral solids
- Tetrahedral solids

31.7 LOAD TYPES SUPPORTED

- Dead load (component auto-detection)
- Live (room-by-room)
- Staircase load
- Soil pressure
- Wind (IS 875)
- Seismic (IS 1893)
- Temperature
- Hydrostatic
- Vehicle load
- Machine vibration load
- Water load for tanks

- Retaining wall active/passive loads
- Landfill self-weight + leachate load

31.8 ANALYSIS TYPES
Linear static analysis
Nonlinear static
Modal analysis
Response spectrum
Pushover
Soil–structure interaction
Contact analysis (RE walls, footings)
Fluid-structure interaction (tanks)

31.9 VISUALIZATION ENGINE REQUIREMENTS (STAAD-like)
Must support:

- 3D model manipulation
- Node/member highlighting
- Cross-section viewer
- Slab/plate contours
- Soil domain view
- Stress contours
- Deflected shape animation
- Load arrows (wind, live, dead)
- Member capacity ratio (color-coded)
- Selecting members → property panel on right side

User modifications:

- Add/delete/modify nodes
- Add beams/slabs/walls
- Move grid lines
- Modify boundary conditions
- Apply loads
- Change materials
- Insert foundations
- Insert shear walls
- Change meshing density

31.10 AI + USER HYBRID MODELING
User starts with:
"I need a G+2 house in Hyderabad."
AI asks guided questions:

- Room sizes?
- Stair location?
- Architectural preferences?
- Slab type?
- Soil details?
- Seismic zone?
- Parking?

- Water tank?
- Retaining wall?

Then AI generates:

- Architectural plan
- Structural grid
- Suggested column locations
- Slab edges
- Preliminary foundation type

User edits visually → AI syncs model.

31.11 EXPECTED OUTPUTS

A. Architectural

- Floor plans
- Sections
- Elevations

B. Structural

- Analysis model
- BMD/SFD
- Slab stresses
- Soil pressure maps
- Seismic drift report

C. Design

- Member design reports
- Footing design
- Staircase design
- Retaining wall design
- Water tank design
- Composite beam design
- PEB design

D. Drawings

- Foundation plan
- Column layout
- Beam layout
- Slab reinforcement
- Staircase drawings
- Retaining wall
- Water tank
- Plinth beams
- Lintel beams
- Roof details
- Sections & elevations
- PEB GA drawings
- Fabrication drawings

E. BOQ & Costing

- SSR integration
- Material summary
- Manpower estimates

- Equipment estimates

F. Project Execution

- Gantt charts
- Progress monitoring
- Daily logs
- Cost control
- QA/QC
- Safety checklists

31.12 MONETIZATION MODEL

Each module unlocks separately:

1. **Architectural generator**
2. **Structural analysis**
3. **Structural design**
4. **Drawings**
5. **Estimate from SSR**
6. **Project management tools**
7. **Detailing & Fabrication drawings**
8. **PEB design package**
9. **Landfill design & stability**
10. **Retaining wall design**

User pays **only for modules used**.

31.13 DOCKER ARCHITECTURE

Inside solver container:

- Ubuntu base
- Kratos Multiphysics
- Python 3.11
- Preloaded material libraries
- IS codes formulas
- GeoMechanics + Structural apps
- GPU-enabled optional builds

All apps interact via API.

31.14 API DESIGN FOR INTEGRATION

REST endpoints:

- /model/build
- /model/update
- /model/visualize
- /analysis/run
- /design/column
- /design/slab
- /design/footing
- /design/retainingwall
- /design/peb
- /drawings/generate
- /estimate/generate

- /project/setup

31.15 RISKS & MITIGATIONS

Risk	Mitigation
Heavy FEA loads	Split into async jobs, queue, optimize meshes
Kratos complexity	Provide simplified wrappers
AI misinterpretation	Multi-step confirmation flow
High GPU cost	Optional local solver for Pro users

PART 31 COMPLETE

PART 32 — AI-ASSISTED DRAWING INTELLIGENCE ENGINE (ADIE)
Autonomous Drawing Generation, Review, Optimization & Compliance Engine

---

32.1 Purpose
The **AI-Assisted Drawing Intelligence Engine (ADIE)** is a unified subsystem that:

1. **Interprets user requirements**
(spoken, typed, uploaded image/PDF/scanned sketch/CAD)
2. **Generates all architectural, structural, and construction drawings**
(concept → schematic → detailed → IFC-ready)
3. **Validates design intent**
against NBC, IS codes, zoning bye-laws, fire norms, structural design outcomes.
4. **Provides a STAAD-like 3D model view + editable elements**
so that the user engineer can:
    - Visualize the model
    - Orbit, pan, zoom
    - View cross-sections
    - Select members
    - Modify geometry graphically
    - Re-run analysis on the updated model
    - Regenerate drawings
5. **Learns and Improves** based on past drawings & user corrections
(active learning loop).

32.2 Inputs to ADIE
ADIE consumes multi-modal inputs:
32.2.1 User Requirements

- "Construct G+2 building in Hyderabad"
- "PEB + mezzanine + 10-ton crane"
- "Retaining wall around landfill"
- "Industrial shed 50 × 120 m with 12 m height"

32.2.2 Uploaded Artifacts

- Hand sketch (scanned or mobile photo)
- Marked-up A3/A4 drawings
- Existing CAD drawings (DWG/DXF)
- Architectural plans (PDF)
- BIM IFC models
- GIS/KML site layouts
- Existing STAAD/ETABS input files

32.2.3 Design Engine Outputs (Parts 9–15)

- Structural analysis model & forces
- Foundation design
- Column-beam-slab checks
- Rebar schedule
- Steel detailing
- MTO/BOQ
- 3D geometry database
- Load combinations, envelopes

- Optimization reports

32.2.4 Compliance Libraries

- NBC 2016 + Amendments
- Local building bye-laws
- Fire safety norms
- Accessibility guidelines
- Plot area, setbacks, height limits
- IS code provisions affecting drawings:
    - IS 456, IS 875, IS 800, IS 1904, IS 1893, IS 2911
    - Steel/Concrete detailing codes

32.3 Outputs from ADIE

32.3.1 Architectural Drawings

- Floor plans, elevations, sections
- Door/window schedules
- Toilet/kitchen layouts
- Staircase sections
- Parking layout
- Fire exit plans
- Site layout with setbacks
- Massing model in 3D

32.3.2 Structural GA Drawings

- Column centerline plan
- Beam layout
- Slab layout
- Footing layout
- Plinth beam layout
- Shear wall layout
- Staircase reinforcement
- Water tank structure
- Ramp/driveway structure

32.3.3 Reinforcement Detailing

- Column rebar drawings
- Beam rebar drawings with sections
- Slab detailing & bar bending schedule
- Footing details
- Shear wall detailing
- Core wall detailing

32.3.4 Steel Detailing

- PEB general arrangement
- Frame elevations & sections
- Connection plates
- Baseplate & anchor bolt details
- Bracing & sheeting layouts

32.3.5 Retaining Wall / RE Wall Drawings

- Plan, elevation
- Reinforcement

- Key, counterforts (if applicable)

32.3.6 Landfill Drawings

- Plan
- Cell layouts
- Bunds, berms
- Sections with slopes
- Drainage, gas vents

32.3.7 IFC/BIM Exports

- IFC2x3 / IFC4
- BIM-compatible layers
- 3D model exchange

32.3.8 Cost-Integrated Drawings

- Every drawing linked to estimation database
- BOQ auto-updates after any drawing change

32.4 Core Functional Modules

32.4.1 Drawing Interpretation Engine (DIE)

- Parses sketches, PDFs, photos
- Converts to vector geometry
- Recognizes rooms, walls, openings, symbols
- Detects mistakes, missing dimensions
- Converts to parametric model

32.4.2 AI Layout Generator

Automatically generates:

- Room layouts
- Column grids
- Beam/slab patterns
- Staircase geometry
- Plumbing, ventilation shafts
- Fire escape routes
- Structural system selection

32.4.3 Structural–Architectural Coordination Engine

- Ensures beams align with walls
- Special seismic zones:
  - soft storey warnings
  - torsional irregularities
  - heavy cantilever warnings
- Auto-adjusts layout to meet IS 1893 irregularity limits

32.4.4 Construction Detailing Engine

- Auto dimensioning
- Auto annotation
- Auto generation of section cuts
- Dynamic reinforcement templates
- Auto bar bending schedule

32.4.5 3D Visualizer (STAAD-Like Interface)

The engineer can:

- View entire structure in 3D (WebGL/Three.js/Kratos/VTK backend)

- Rotate, zoom, explode assemblies
- View structural members with color-coded forces
- Select beams/columns to see design results
- Modify geometry graphically
- Trigger re-analysis
- Trigger drawing regeneration
- View cross-sections interactively
- Switch between architectural/structural views
- Toggle:
    - loads
    - supports
    - reactions
    - results envelopes

32.4.6 Revision Management & Audit Trail

- Every drawing change is versioned
- User can revert
- PDF comparison diff view
- Drawing change history
- Forensic audit records

32.4.7 Output Engines

- PDF
- DXF/DWG
- BIM IFC
- 3D Web model
- QR code per drawing

32.5 Integration with Upstream & Downstream Systems

Upstream Modules

- User Requirement Engine (Part 1)
- AI Questionnaire Engine (Part 2)
- Site Context Engine (Part 3)
- Design Load Engine (Part 6)
- Analysis Engine (Part 7)
- Design Engine (Part 8–15)

Downstream Modules

- Estimation Engine (Part 16)
- Project Planning & Management (Part 17–21)
- Procurement (Part 22–26)
- Construction Monitoring (Part 27–29)
- Handover & Closure (Part 30–31)

32.6 Data Structures

32.6.1 Unified Drawing Object (UDO)

Contains:

- Geometry
- Layers
- Dimensions

- Styles
- References
- Line weights
- Annotation schema
- Scale
- Export settings

32.6.2 Design-Linked Drawing Features

Each beam/column/slab is linked to:

- Design forces
- Governing load combination
- Governing clause
- Rebar design
- Bill of materials entry

32.7 AI Automation Levels

Level 0 — Manual

User draws manually.

Level 1 — Semi-Automatic

User draws → AI adjusts & validates.

Level 2 — Automatic Generation

User describes project → AI generates complete drawing set.

Level 3 — Iterative Refinement

User changes one parameter → AI updates all related drawings.

Level 4 — Autonomous Architect–Engineer

AI proposes multiple design options:

- Optimal
- Cost-effective
- Fastest construction
- Greenest/IGBC-compliant

32.8 Drawing Quality & Compliance Engine

This engine automatically checks:

- Dimensions completeness
- Out-of-scale drawings
- Missing rebar cover
- Beam-column alignment
- Slab span limits
- Foundation eccentricity
- Staircase headroom
- Parking norms
- Fire exit travel distances
- Lift shaft compliance
- NBC 2016
- Local building rules (using pinned geolocation)

32.9 Monetization & Pay-per-Use Model

Users are charged per drawing package:

- Concept plan
- GA drawings
- Structural drawings
- Rebar drawings
- Steel fabrication drawings
- Landfill cell drawings
- Architectural set
- BIM export
- DXF package
- PDF package
- Revisions

This supports your multi-module business model.

32.10 Roadmap for ADIE Implementation

Phase 1 — MVP

- Architectural plan generator
- Basic structural GA
- 3D visualization
- PDF output

Phase 2 — Engineering Integration

- Steel/RCC detailing
- BIM exports
- DXF output
- Reinforcement drawings

Phase 3 — AI & Automation

- AI sketch-to-drawing
- Layout optimization
- Earthquake irregularity detection
- Fire compliance auto-checks

Phase 4 — Kratos/FEniCSx/Custom Solvers

- Direct analysis–linked drawings
- 3D results visualization
- Auto detailing based on FE stresses

PART 32 completed.

📘 PART 33 – MULTI-STRUCTURE UNIFIED DESIGN ENGINE (MSUDE)

A Central Intelligence Layer that Converts Any User Requirement Into a Coherent Structural System, Analysis Model, and Design Workflow

---

33.1 Purpose of MSUDE

This module acts as the **brain of the entire platform**.
Because your app handles:

- **PEB Steel Buildings**
- **RCC G+1, G+2, G+5... buildings**
- **Industrial sheds**
- **Composite buildings**
- **Retaining walls**
- **RE walls**
- **Water tanks & sumps**
- **Staircases**
- **Foundations of all types (isolated, combined, raft, pile)**
- **Landfills & bunds**
- **Roads, pavements, drains**
- **Any user-uploaded building plan**

...the design engine must unify all of them into a single **core logic**, ensuring:

1. **Consistent modelling logic**
2. **Consistent design workflow**
3. **Consistent checking logic (codes, failure modes, safety)**
4. **Consistent data formats for visualization and drawing generation**
5. **Consistent interoperability formats (STAAD import/export, IFC, DXF)**

MSUDE ensures that no matter what structure the user requests, the **design workflow is predictable, validated, and automatable**.

33.2 High-Level Function

MSUDE performs the following:

1. **Interprets the user requirement**
   Example: *"G+2 residential building in Hyderabad, 40x60 plot, 2 flats per floor"*
2. **Generates a Structural Typology Tree**:

Component	Options Identified	Chosen by Engine
Structural system	RCC framed / shear wall / composite	RCC framed
Slab type	One-way / Two-way / Flat slab	Two-way
Staircase type	Dogleg / Open-well	Dogleg
Foundation	Isolated / Combined / Raft / Pile	Isolated
Lateral system	Moment frames / Shear walls	Moment frames
Special elements	Retaining wall at front	Yes

3. **Creates a Unified FEM Model Descriptor**
   MSUDE converts the typology into a **mesh definition**, **boundary conditions**, and **loads** settings.
4. **Passes the descriptor to the selected solver**
   - **Kratos Multiphysics** (preferred)

- FEniCSx (secondary)
    - STAAD-like model (export)
    - IFC (export)
5. **Receives results**, validates them, re-runs if unstable.
6. **Triggers design checks**
   Based on IS 456, IS 800, IS 875, IS 1893, etc.
7. **Passes clean "Design Decisions" to the Drawing Generator**
    - Beam sizes
    - Column sizes
    - Footing dimensions
    - Reinforcement
    - Steel member sizes
8. **Passes "Final Quantities" to Estimation Engine**
    - Concrete
    - Steel
    - Shuttering
    - Masonry
    - Finishes

33.3 Internal Architecture

USER REQUIREMENT INTERFACE

▼

REQUIREMENT PARSER

▼

STRUCTURE TYPE CLASSIFIER (ML)

▼

STRUCTURAL TYPOLOGY ENGINE

▼

UNIFIED FEM MODEL GENERATOR
(Kratos Core Compatibility)

▼

SOLVER DISPATCHER
Kratos / FEniCSx / etc

▼

RESULTS

```
        ▼
┌──────────────────────────────┐
│ CODE CHECKS + OPTIMIZATION   │
└──────────────┬───────────────┘
        ▼
┌──────────────────────────────────┐
│ DRAWING + DETAILING + REPORT ENGINE │
└──────────────────────────────────┘
```

33.4 Structural Typology Engine (STE)

Input Examples:

- "*G+2 residential in Hyderabad, 40x60 plot*"
- "*Industrial PEB shed 80x140 in Gujarat*"
- "*RE wall 3 m height with 2 m surcharge*"
- "*Landfill with 50k m^2 footprint*"

Output Typology:

RCC Building Example

System Type: RCC Framed
Slab Type: Two-way
Beam Depth Rule: Clear span / 12
Column Layout: Grid-based
Foundation: Isolated for columns < 3000 kN, Combined > 3000 kN
Shear design: Ductile per IS 13920 (if Zone ≥ III)

PEB Example

System Type: Steel Portal Frame
Frame Spacing: 6 m
Roof Slope: 1:10
Eave Height: 9 m
Purlins: Z-200 @ 1.2 m spacing
Bracings: X-bracing between frames 1-6

Landfill Example

Lift Height: 5 m
Bench Width: 4 m
Final Slope: 1V:3H
Leachate Pipe Spacing: 15 m
Top Cap Thickness: 1.5 m
This typology governs model creation.

33.5 Unified FEM Model Generator (UFEMG)

This is the most critical module.
It produces **one universal FEM representation** for all structure types.

Supports:

Type	Mesh	Kratos Application
RCC frame	1D beam, 2D slab	StructuralMechanicsApplication
Steel PEB	Beam elements	StructuralMechanicsApplication
Retaining wall	Plane strain	GeoMechanicsApplication

Type	Mesh	Kratos Application
Landfill	2D/3D continuum	GeoMechanicsApplication
RE wall	Interface + soil	ContactMechanicsApplication
Pavements	2D continuum	StructuralMechanicsApplication

UFEMG produces:

1. **Nodes**
2. **Elements**
3. **Boundary conditions**
4. **Loads**
5. **Materials**
6. **Sections**
7. **Solver settings**

All in **Kratos-compatible JSON/TMD format**.

33.6 Solver Dispatcher

Depending on user license, complexity, and cost:

Solver	Use Cases
Kratos Multiphysics	Primary, most flexible, all building types
FEniCSx	Academic + FEM for custom constraints
STAAD Export	For clients requiring compliance output
ETABS Export	For building consultants
IFC Export	BIM workflows
DXF Export	Architecture drafts

33.7 Code Check & Optimization Layer

Included Standards (India + International)

- **IS 456** (RCC)
- **IS 800** (Steel)
- **IS 875** (Loads)
- **IS 1893** (Seismic)
- **IS 13920** (Ductile detailing)
- **IRC Codes** (roads)
- **ACI, AISC, EC** extensions allowed

Optimization:

- Beam/column auto-sizing
- Steel weight minimization
- Concrete economy factors
- Foundation type selection
- PEB frame optimization
- Topology-based optimization (Kratos SMA)

33.8 Drawing & Detailing Binding

Uses core outputs to produce:

- Architectural layouts
- Structural layouts

- Beam schedules
- Column schedules
- Reinforcement drawings
- Steel GA drawings
- Connection details
- Bill of Quantities
- Construction sequence drawings

All optimized for **DXF, SVG, PDF, IFC** output.

33.9 API Layer for Frontend (3D Visualization)

Provides:

- Node coordinates
- Element meshes
- Material zones
- Section properties
- Load diagrams
- Support conditions
- Internal forces

Used by:

3D Viewer / Model Editor

- Orbit, pan, zoom
- Cross-section cuts
- Node/element editing
- Parameter editing
- Real-time model rebuild
- STAAD-like interface

33.10 Role of Part 33 in Full System

This part is the **heart of automation**:

- Converts vague user intent into structural reality
- Ensures consistency across all building types
- Enables scalable module-by-module pricing
- Eliminates need for STAAD-like manual modelling
- Supports revisions and quick design iterations

ART 34 — PROCUREMENT & CONTRACTS MANAGEMENT MODULE (ADVANCED)
Integrated With Design → Estimation → BOQ → Execution → Billing → Closure
This module governs **how materials, services, contractors, consultants, and vendors are procured**, fully aligned with the AI-driven design + estimation ecosystem of your platform. It includes automated workflows, rate intelligence, vendor qualification, tender preparation, bid evaluation, contract drafting, and lifecycle management.
This module must work seamlessly irrespective of project type:
PEB, RCC Building, G+2 Residential, Industrial Shed, Warehouse, Retaining Wall, Landfill, RE Wall, Roadworks, Utilities, Infrastructure, MEP, etc.

34.1 OBJECTIVE
Create a **Unified AI-Enabled Procurement Engine** that:
1. **Understands the designed model** (structural system, materials, MTO, BOQ, drawings).
2. **Sources procurement packages** automatically (civil, steel, MEP, finishes, site works, PEB, prefab, etc.).
3. **Generates tender documents** automatically.
4. **Evaluates vendor bids** digitally.
5. **Negotiates using AI** (comparative + predictive analytics).
6. **Issues and tracks Work Orders / Purchase Orders**.
7. **Manages risk, variations, claims**, and **contract close-out**.

34.2 CORE PROCUREMENT WORKFLOW
1. Inputs Trigger Procurement
Procurement starts automatically when any of the following completes:
- Conceptual layout
- Structural design
- Architectural drawings
- BOQ auto-generated
- Cost estimation completed

2. Package Definition (Automated)
AI groups items by:
- Specialisation
- Work trade
- Sequencing logic
- Risk allocation
- Market availability

Examples of packages automatically generated:
- Site enabling
- Earthworks
- RCC / Structural
- Formwork
- Reinforcement supply
- Concrete supply
- Masonry
- Waterproofing
- Flooring

- PEB fabrication & erection
- Structural steel supply
- MEP works
- Painting / finishing
- External development
- Landfill bunding, liners, leachate drains
- Retaining wall construction
- Roadworks
- Material supply only (cement, steel, aggregates)

34.3 PROCUREMENT BREAKDOWN STRUCTURE (PBS)

Level 1 – Project Procurement
  Level 2 – Work Category (Civil, Structural, MEP, Infra, Landfill, PEB)
    Level 3 – Package Groups (e.g., Structural Frame)
      Level 4 – Packages (e.g., Formwork, Concrete, Reinforcement)
        Level 5 – Line Items (BOQ-linked)

All levels are **mapped to BOQ → Design → Drawings**.

34.4 PROCUREMENT TEMPLATES

Each project type has **pre-defined templates** (customizable):

Building Templates

- G+1, G+2, G+3 RCC templates
- Steel/Composite buildings
- Industrial buildings
- Warehouses
- Offices / Commercial

Infrastructure Templates

- Retaining walls
- RE walls
- Landfills
- Roadworks
- Culverts and stormwater drains

Specialized Templates

- PEB supply and erection
- Fabrication packages
- HVAC systems
- Electrical panels
- Plumbing packages

34.5 AI-DRIVEN REQUIREMENT EXTRACTION

AI reads:

- Architectural drawings
- Structural drawings
- Sketches uploaded by the user
- Layout plans
- Vendor brochures
- User sketches (hand drawings)

- Google Maps/KML base plan
- BIM models (if uploaded)

AI identifies:

- Material needs
- Package boundaries
- Quantities required
- Quality specifications
- Lead times
- Execution constraints
- Critical path procurement items

34.6 PROCUREMENT MODES SUPPORTED

1. **Open Tender**
2. **Limited Tender**
3. **Single Vendor**
4. **Rate Contract**
5. **Spot Purchase**
6. **Emergency Purchase**
7. **Reverse Auction**
8. **E-Auction**
9. **Framework Contracting**

AI suggests the best mode based on:

- Value
- Risk
- Lead time
- Market behaviour
- Historical prices

34.7 VENDOR ECOSYSTEM MANAGEMENT

Vendor Registration

Collects:

- GST
- PAN
- MSME status
- Labour licence
- ISO certifications
- Equipment list
- Manpower strength
- Work experience

Vendor Ranking (AI)

Scores based on:

- Quality rating
- Timeliness
- Past work orders
- Location advantage
- Cost competitiveness
- Safety performance

- Historical disputes

34.8 TENDER DOCUMENT AUTOMATION

AI generates:

- NIT (Notice Inviting Tender)
- Scope of Works
- Technical Specifications
- Drawings (auto-insert latest revision)
- BOQ
- Payment terms
- Completion schedule
- Warranty clauses
- Insurance requirements
- Safety & compliance clauses
- LD & penalty clauses
- Deviations schedule

Supported formats:

- PDF
- Word
- Excel
- GitBook-style online tender

34.9 BID SUBMISSION PORTAL (DIGITAL)

- Secure vendor login
- Auto-validated quote formats
- Rate break-up enforcement
- Make/brand information capture
- Submission timestamp
- Deviation notes
- Tender Q&A

34.10 BID EVALUATION (AI + RULE ENGINE)

AI compares:

Technical Evaluation

- Compliance matrix
- Material specifications
- Brand quality
- Methodology
- Team strength
- Relevant experience

Financial Evaluation

- Base rates
- Taxes
- Loading for deviations
- Correction for arithmetic errors
- Rate normalization
- L1 identification

- Value-for-money scoring

AI-Powered Insights

- Market trend comparison
- Outlier detection
- Red flags (too low / unusually high bids)
- Supplier performance predictions

34.11 CONTRACT GENERATION

Once the vendor is finalized:

- AI generates **Work Order / Purchase Order**
- Auto-inserts:
  - Payment terms
  - Safety clauses
  - QA/QC requirements
  - Completion schedule
  - LD clauses
  - Contract drawings
  - Revisions log

Legal boilerplate adjustable to:

- Indian Contract Act
- FIDIC
- CPWD/MoRTH templates
- Company-specific templates

34.12 MATERIAL PROCUREMENT INTEGRATION

For material supply:

- Auto-generate **Material Indent**
- AI suggests vendors
- Compares prices with:
  - SSR
  - Historical data
  - Market index
- Predicts lead time and stock-out risks
- Issues purchase orders
- Tracks delivery using QR codes

34.13 SUBCONTRACT PROCUREMENT INTEGRATION

Subcontractor WOs integrate with:

- Schedules
- Labour productivity
- Billing
- RA measurements
- QC approvals
- Site instructions
- Non-conformance reports

34.14 PROCUREMENT–ESTIMATION FEEDBACK LOOP

When a vendor quote differs significantly from SSR:

- AI updates the cost database
- Suggests revising estimation
- Recalculates project cost
- Updates cashflow forecasts

34.15 PROCUREMENT DASHBOARDS

Summary Views

- PO/WO summary
- Savings achieved
- Vendor performance index
- Tendering progress
- Delayed procurements
- Budget vs award value
- Package-wise status

3D Model Integration

- Click a member in the 3D view to see:
    - Package
    - Responsible vendor
    - Cost
    - Delivery status

34.16 RISK MANAGEMENT

AI flags:

- Critical path items
- Single supplier dependencies
- High-cost variance items
- Material price volatility
- Vendor reliability risks

34.17 PROCUREMENT LOGS & AUDIT TRAIL

Everything is recorded:

- Tender creation
- Vendor queries
- Bid submission timestamps
- Evaluation decisions
- Approvals and rejections
- Contract modifications
- Delivery notes
- GRNs & QC records

Compliant with:

- ISO 9001
- ISO 45001
- Internal audit requirements

34.18 PROCUREMENT INTEGRATION WITH PROJECT EXECUTION

Automated triggers:

- Material arrival updates schedule
- Vendor delays update risk register
- Quality rejections notify PM/QC head
- RA bill generation tied to WO BOQ
- Completion triggers retention clause
- Final bill triggers contract close-out

34.19 PROCUREMENT INTEGRATION WITH BILLING

Billing module auto-fetches:

- Contract BOQ
- Tender specifications
- Variations (VOs)
- Measurements
- Material reconciliation
- Retention percentage
- Advances & deductions

34.20 CONTRACT CLOSE-OUT

Final outputs:

- Completion certificate
- Final RA bill
- Retention release
- Vendor performance scoring
- Lessons learned
- Digital archive of entire package

📘 PART 35 — MULTI-PROJECT PROGRAM MANAGEMENT & PORTFOLIO CONTROL MODULE

This module elevates the platform beyond a single-project engineering tool into a **portfolio-level construction management system** comparable to Primavera P6 + Oracle Aconex + MS Project—but deeply integrated with your AI-driven design-to-execution pipeline.

35.1 Purpose of the Module

The objective is to allow:

- Organizations
- Project management offices
- EPC contractors
- Developers
- Government/municipal agencies

...to manage **multiple concurrent projects**—each at different stages of design, estimation, procurement, execution, billing, and closure—inside a **single unified dashboard**.

It bridges:

- **Design stage outputs** (layouts → BoQ → drawings)
- **Estimation** (SSR/DSR-based)
- **Procurement** (vendors, tenders, GRN)
- **Construction execution** (daily logs, materials, labour)
- **Quality & safety**
- **Schedules (CPM/PERT)**
- **Financial monitoring (cashflow)**
- **Risk & delay tracking**
- **Change orders**
- **Handing over & closure**

All under one integrated umbrella.

35.2 Key Users

1. Owner / Client Organization

- Reviews overall portfolio health
- Approves budgets, invoices, changes
- Tracks progress vs. strategic goals

2. Engineering Team / Designers

- Manages multi-project design queues
- Tracks design revisions, drawing logs

3. Project Manager / PMC

- Monitors progress of multiple jobs
- Reviews risks, delays, and escalations

4. Site Engineers

- Logs daily site activity for each project
- Updates quantities and completed items

5. Contractors / Subcontractors

- Inputs daily progress and labour deployment
- Submits RA bills & variations

6. Procurement Team

- Tracks purchases across all projects
- Compares consumption vs. budget

7. Finance

- Monitors cashflow across all sites
- Controls budget deviations

35.3 Core Features

A. Portfolio Dashboard

A high-level interface showing:

- **Total active projects**
- **Design stage status**
- **Execution stage status**
- **Pending approvals**
- **Portfolio budget vs. actual**
- **Critical delayed projects**
- **Risk indicators**
- **Upcoming milestones**
- **Cashflow curves for all projects**
- **Heat map of cost overruns**

AI provides:

- “Projects most likely to miss deadlines”
- “Top 5 cost overrun risks”
- “Vendor performance deviation alerts”
- “Recommended recovery schedule”

B. Multi-Project Scheduling Engine

Equivalent to a simplified P6 engine:

- CPM/PERT for each project
- Cross-project dependencies (optional)
- Resource leveling across portfolio
- Primavera-style Gantt with overlays
- AI auto-generation of schedules from:

"Construct a G+2 building in Hyderabad"

- Schedule templates per project type (PEB, RCC, Landfill, RE Wall, Roads, etc.)
- Auto-updated based on:
    - daily progress logs
    - completed activities
    - delayed material
    - labour shortage
    - climate/weather interruptions

C. Portfolio Resource Optimization

Manages all resources across multiple sites:

Resource Pools

- Labour categories
- Equipments

- Shuttering / scaffolding
- Formworks
- Cranes
- Excavators
- Steel cutters/benders

AI Allocation Suggestions
Example:
"Assign Excavator-02 from Site A to Site C on Tuesday to avoid idle time."
Conflict Resolution
If the same team/equipment is scheduled in 2 projects at same time → alert + suggestions.

D. Multi-Project Cost Monitoring
Central financial control:

- Budgeted cost vs. actual cost for each project
- Portfolio cashflow S-curve
- Monthly burn rate across all projects
- Committed vs. uncommitted cost
- Vendor-wise cost distribution
- Subcontractor liabilities
- Head office allocations

AI detects:

- "Project B is overshooting steel consumption by 12% compared to similar projects."
- "Cashflow shortage in Q3 due to procurement spikes."

E. Inter-Project Document Control
Unified DMS across all projects:

- Drawing logs
- Approvals
- RFI management
- Change orders
- Material samples
- Test reports
- Site photos
- Geo-tagged site evidence

Document interlinking:

- If a drawing revision impacts another project using similar standard details → notify automatically.

F. Multi-Project Risk & Issue Tracker
A comprehensive facility to log:

- Design risks
- Material risks
- Site conditions
- Labour delays
- Contractor issues

- Authority approvals
- Environmental risks

AI auto-detects risks from:

- Daily logs
- Site photos
- Trend of delays
- Quantity deviations
- Safety incidents

Each risk has:

- severity score
- probability score
- mitigation plan
- owner (responsible person)
- deadlines
- escalation path

G. Cross-Project Procurement Consolidation

Procurement optimization across portfolio:

- Club common items (steel, cement, shuttering)
- Bulk purchase recommendations
- Inter-site material transfer suggestions
- Consumption analytics for each BOQ item
- Vendor performance benchmarking cross-project

Example:

"You can save ₹ 24,00,000 by combining steel procurement for Projects C and F."

H. Organizational View

Across all business units:

- Industrial buildings
- Residential
- Commercial
- Landfills
- RE walls
- PEB warehouses
- WTE plants

Each business unit has KPIs:

- Avg. delay
- Cost overrun %
- Rework %
- Safety score
- Productivity vs. benchmark

35.4 System Architecture for Multi-Project Layer

The architecture extends the platform into:

1. Multi-Tenant Capability

Each company manages its own portfolio.

2. Multi-Project Databases

Shared tables for:

- Resources
- Vendors
- Design libraries
- Standard drawings
- SSR/DSR rates

Project-specific tables for:

- BoQ
- Schedules
- Materials
- Logs
- RA bills
- QC records
- Expenses

3. AI Engines

- Schedule generation
- Workflow drafting
- Drawing auto-production
- Resource optimization
- Cost prediction
- Risk analytics

4. 3D Model Integration

Each project hosts:

- 3D wireframe
- Sectional views
- Interactive editing (like STAAD)
- Cross-project comparison of geometry
- Model-based cost estimation

35.5 Workflows

A. When a New Project is Created

1. User enters:
    - "Construction of G+2 building in Hyderabad"
2. AI asks all required questions
3. AI suggests design templates
4. User selects template or uploads own sketch
5. System auto-builds:
    - Layout
    - Structural skeleton
    - Preliminary BOQ
    - Preliminary design
6. Project now enters the portfolio dashboard

B. During Execution

- Daily logs update schedule
- Schedule updates cost and productivity
- Cost updates cashflow

- Cashflow updates risk dashboard

A closed loop.

C. After Completion

- Auto-generate completion report
- Cost variance summary
- Final drawings archive
- Handover documents
- As-built model
- Post-mortem analytics for future benchmarking

35.6 KPIs Tracked

Performance

- Schedule performance index (SPI)
- Cost performance index (CPI)
- Labour productivity index

Quality

- NCR rate
- Rework percentage
- Cube test performance trend

Safety

- LTIFR
- Near misses
- Unsafe act reports

Cost

- Budget deviation %
- Material consumption variance
- Vendor quality index

Progress

- Physical vs. financial progress
- Milestone compliance

35.7 AI Capabilities in This Module

Predictive Intelligence

- Delay forecast
- Cost overrun prediction
- Resource shortage
- Vendor risk

Generative Intelligence

- Schedules
- Method statements
- Risk mitigations
- Contract clauses
- Design assumptions

Diagnostic Intelligence

- Root cause of delays
- NCR pattern analysis

- Repeated vendor issues

35.8 Deliverables of This Module

1. **Portfolio Dashboard**
2. **Portfolio Analytics**
3. **Multi-Project Gantt**
4. **Resource Allocation Engine**
5. **Procurement Clubbing Engine**
6. **Financial Monitoring**
7. **Risk & Issue Management**
8. **Document Control**
9. **Org-Level BI Dashboards**
10. **Cross-Project AI Insights**

📘 PART 36 — FAILURE MODES, SAFETY CHECKS & ENGINEERING VALIDATION ENGINE

This section defines the built-in system responsible for identifying, flagging, and quantifying structural failures, serviceability violations, constructability risks, and code non-compliances across all supported structure types (RCC, Steel, PEB, Composite, Landfills, Retaining Walls, RE Walls, Bridges, Industrial Sheds, Warehouses, etc.).

The Failure Modes & Validation Engine must operate **independently** of the design engine and must be capable of:

1. **Stand-alone post-processing** (for models imported from STAAD, Etabs, SAP2000, Robot, Tekla, or any other tool).
2. **Inline real-time checks** while the user is modelling in the app's 2D/3D editor.
3. **Batch validation** for multiple models in automated workflows (AI-driven).

36.1 Purpose & Objectives

The engine must:

- Detect unsafe design conditions.
- Identify code violations as per IS / IRC / ACI / AISC / Eurocode.
- Evaluate serviceability failures (deflection, crack width, drift, settlement).
- Provide **failure-specific recommendations**.
- Give numerical, graphical, and AI-explanatory feedback.
- Prevent user from proceeding to drawing generation if critical failures exist.
- Log failure history for quality audits and certification.

36.2 Supported Codes

Indian Codes (Primary)

- IS 456
- IS 800
- IS 801
- IS 875 Parts 1–5
- IS 1893
- IS 16700
- IS 2950 (Foundations)
- IS 3370 (Water tanks)
- IS 13920 (Ductile detailing)
- IS 14458 (Retaining/RE walls)
- IRC 6, IRC 21 (Bridge components)
- SWM 2016 (Landfills)
- CPHEEO Manual (Landfills & STP/Water infra)

International Codes (Secondary)

- AISC LRFD
- ACI 318
- EN 1992, 1993, 1997, 1998
- ASCE 7
- FHWA (For RE walls)

36.3 Failure Mode Categories

The engine must group failures into 7 master categories:

Category 1: Strength (Ultimate Limit State) Failures
Applicable to RCC, Steel, Composite & Foundations.
Checks include:

- Flexure capacity exceeded
- Shear/diagonal tension failure
- Bond & anchorage failure
- Bearing stress exceedance
- Buckling (local & global)
- Torsion failure
- Combined axial + bending + shear failures
- Plastic hinge concentration limits
- Punching shear (slabs, footings)
- Plate/shell membrane ruptures
- Steel member limit state failures (IS 800 Cl. 8 & 9)

Outputs:

- Failure ratio
- Governing load combination
- Governing location (element ID, node ID)
- Recommended redesign steps

Category 2: Serviceability Failures (SLS)
Includes:

- Excess vertical deflection
- Excess lateral drift
- Crack width violation
- Vibration frequencies below acceptable thresholds
- Differential settlement
- Opening distortion

Outputs:

- Code-referenced limits
- Measured values
- Recommended sizing changes

Category 3: Stability Failures
Includes:

- Global structural instability
- Local buckling (steel/web/flange)
- Lateral-torsional buckling
- P-Delta effects
- Soil bearing failure
- Slope instability (Landfill)
- RE wall sliding/overturning/pullout/connection failure

Category 4: Constructability Failures
Includes:

- Reinforcement congestion

- Bar spacing below minimum
- Non-constructible beam-column joints
- Steel connection geometry clash
- Bolt/plate access issues
- Equipment installation clearance failure
- Inadequate lap lengths
- Rebar bends violating IS constraints

Category 5: Durability Failures
Includes:

- Cover deficiencies
- Chloride exposure under-designed
- Sulphate class violation
- Corrosion environment mismatch
- Crack width > durability limits

Category 6: Detailing Failures
Includes:

- Rebar anchorage insufficient
- Hooks missing
- Shear reinforcement spacing violation
- Column-beam joint design mismatch
- Steel detailing inconsistencies
- Weld size below minimum
- Missing stiffeners
- End-plate bolt pattern failure

Category 7: Geotechnical Failures
Includes:

- Soil uplift
- Bearing capacity failure
- Sliding
- Overturning
- Liquefaction (IS 1893)
- Slope failure (Landfill/RE Wall)
- Reinforcement pullout in RE walls

36.4 Inputs to the Failure Engine

- Mesh, nodes, elements
- Section properties
- Material properties
- Internal forces (Fx, Fy, Fz, Mx, My, Mz)
- Displacements, rotation
- Soil parameters
- Load combinations
- Boundary conditions
- AI-inferred system constraints

- User inputs from the 3D modeller
- Model imported from STAAD/Etabs/IFC/DXF

36.5 Outputs

1. Failure Summary Panel

- Red/amber/green classification
- Failure percentages
- Governing combinations
- Critical elements

2. Graphical Visualization

- 3D heatmaps on the model
- Element-wise failure ratio
- Drift profile
- Crack width map
- Settlement contours

3. AI Explanations

- Reason for failure
- How the code determines the failure
- What redesign options exist
- Possible risks if not corrected

4. Automated Recommendations

- Section resizing
- Material grade improvements
- Bracing addition
- Foundation enlargement
- Load redistribution options

36.6 Automated Fix Suggestions Engine

The system must automatically propose fixes such as:

For RCC

- Increase depth
- Increase width
- Modify reinforcement
- Increase cover
- Switch to higher grade

For Steel

- Increase section size
- Add stiffeners
- Choose built-up section
- Increase bolt diameter or count
- Change connection type

For Retaining/RE Walls

- Increase base width
- Increase stem thickness
- Add heel/toe reinforcement
- Increase reinforcement length
- Improve drainage

For Landfills

- Modify slope angles
- Add berm/bench
- Reinforce weak zones
- Improve liner system

36.7 Failure Codes & Standardized Messages

Each failure must map to:

- Unique failure ID
- Code reference
- Severity level
- Detailed description
- Suggested fix
- Required engineer approval flag

36.8 Safety Decision System

The system will classify the model as:

- **SAFE — Ready for Design & Drawing**
- **NEEDS REVISION — Fix mandatory**
- **UNSAFE — Abort design**
- **DANGEROUS — Fatal*

This determines whether drawings can be generated.

36.9 Integration with 3D Modeller

Failures must be shown interactively in:

- 3D view
- Cross sections
- Element inspector popups
- Model tree
- Design-by-voice/AI panel

User must be able to click a failed element → zoom → see details.

36.10 Integration with Costing Engine

When a fix increases or decreases material, the costing engine automatically updates rates and quantities.

36.11 Integration with Document Generator

Every failure must be embedded in:

- Structural Notes
- Design Basis Report
- Calculation Sheets
- Audit Trails
- QA/QC logs

📘 **PART 36 — ADVANCED SAFETY ENGINE & CODE COMPLIANCE MODULE**

(Next major section of the PRD)

36.1 Overview

The Advanced Safety & Code Compliance Engine ensures that every structural, architectural, and services-related decision passes through mandatory national and international code checks before approval. This module acts as the authoritative validation engine across the entire lifecycle — conceptual design → detailed engineering → drawings → BOQ → execution.

This is not a "post-processing checker"; it is integrated into **every toolpath** inside the platform.

36.2 Supported Codes (Phase 1 & Phase 2)

36.2.1 Indian Codes (Primary for Production)

- **IS 456:2000** – RCC design
- **IS 800:2007** – Steel design
- **IS 875 (Parts 1–5)** – Dead, live, wind, snow, load combinations
- **IS 1893:2016** – Earthquake
- **IS 13920:2016** – Ductile detailing
- **IS 3370 series** – Water retaining structures
- **IS 16700:** High-rise buildings
- **IS 2911:** Pile foundations
- **IRC Codes:** Applicable for boundary roads, culverts, external works
- **NBC 2016:** Fire, safety distances, corridor widths, exits, stairs

36.2.2 Global Codes (Optional, Chargeable Modules)

- **ACI 318**, **AISC 360**, **AISC 341**, **ASCE 7**
- **Eurocode 2, 3, 4, 7, 8**
- **BS 8110**, **BS 5950**
- **Australian Standards (AS)**
- **Kratos-approved constitutive models** where required by clients requesting FEM-level fidelity.

All code modules are plug-in based and chargeable individually.

36.3 Automated Code Interpretation Engine

The platform includes a rule-based + ML-enhanced interpreter that extracts:

- Applicable code clauses
- Load combinations
- Detailing rules
- Limit state checks
- Ductility requirements
- Minimum reinforcement requirements
- Slenderness limits
- Buckling curves
- Shear–flexure interaction
- Wind pressure zones
- Seismic spectra
- Foundation capacity

Each rule is a **digital twin of the code clause**:

IF structure_type = RCC
 AND ft = <user_input>
 THEN check IS456-26.5.1 flexure: $Mu \le 0.138\ fck\ b\ d^2$

Every rule is versioned so future code updates don't break older projects.

36.4 Safety Rating Score (0–100)

Every structure is automatically scored for:

1. **Design Code Compliance (40%)**
2. **Load Adequacy (20%)**
3. **Detailing Compliance (15%)**
4. **Geotechnical Safety (10%)**
5. **Serviceability (10%)**
6. **Redundancy & Robustness (5%)**

Score displayed as:

- **Green** > 80
- **Yellow** 60–80
- **Red** < 60 (Action Required)

36.5 Automated Design Error Detection

The system flags:

- Under-designed beams/columns/slabs
- Missing supports
- Overstressed members
- Torsional irregularities
- Soft-storey issues
- Excessive lateral drift
- Soil bearing capacity violations
- Inadequate footing thickness
- Improper anchorage length
- Non-compliant ductile detailing

Each warning includes:

- Cause
- Code clause reference
- Suggested fix
- Auto-correct option

36.6 FEM-Level Safety Validation (via Kratos Multiphysics)

For critical structures, the user can trigger a **Kratos-based verification run**:

- Non-linear static analysis
- Eigenvalue buckling
- Modal analysis
- Pushover analysis
- Soil–structure interaction
- Contact mechanics for retaining walls
- Material nonlinearity & damage-plasticity
- Fluid–structure interaction (for tanks)

Kratos results will be included in the final design report.

36.7 3D Safety Visualization

Users can view:

- Stress contours (σx, σy, τxy)
- Strain fields
- Crack maps
- Plastic zones
- Buckling modes
- Safety zones (color-coded)
- Capacity vs. demand maps

Visual format similar to STAAD, ETABS, MIDAS, but in modern WebGL.

36.8 Auto-Generated Safety Reports

Reports include:

- Executive summary
- Code compliance table
- Load combinations used
- Demand-capacity ratios
- Failure mode checks
- Safety score
- 3D visuals
- Recommendation list
- Detailing review

Export formats:

- **PDF**
- **Excel**
- **STAAD-like TXT**
- **Client letter format**

Each report has a QR code linking to a verifiable cloud snapshot.

36.9 Integration With the User Journey

The Safety Engine is invoked:

1. During conceptual modeling
2. During structural modeling
3. During FEM simulation
4. Before design finalization
5. Before drawing generation
6. Before BOQ and estimation
7. During construction (revision checks)

This ensures that every stage is validated before moving to the next.

36.10 Pricing Model

The Safety Engine is a core feature with:

- **Free basic compliance checks**
- **Pro-level IS checks (auto-fix + report)**
- **Premium FEM safety verification (Kratos)**
- **International code checks** (separate modules)

36.11 Future Enhancements

- AI-based risk prediction

- Seismic non-linear time-history analysis (chargeable)
- Scenario-based collapse simulation
- Real-time safety alerts during construction via mobile app
- Integration with drone scanning (structural health)

📘 **PART 36 COMPLETED**

📘 PART 37 — AI–POWERED DESIGN HINT ENGINE (GUIDED QUERY ORCHESTRATION)

(A core system that asks the user the right questions, understands intent, clarifies missing inputs, and automatically converts incomplete statements into full engineering-ready data.)

37.1 Purpose & Vision

Most users will begin with vague or incomplete statements:

- “I want to build a G+2 house in Hyderabad.”
- “I need a factory shed about 120x80.”
- “I’m planning a retaining wall for my site.”

This module is the **brain** that turns such natural language statements into:

- A complete **structural design brief**,
- A structured **engineering parameter set**,
- A mapped **design workflow**,
- Selection of correct **codebooks, templates, and modules**,
- And a prioritized **questionnaire** to fill missing details.

This becomes the **first step** before Geometry Builder, Analysis Engine, and Drawing Engine.

37.2 Functional Objectives

✔ 37.2.1 Natural Language → Engineering Intent Extraction

The system automatically extracts:

- Structure Type (RCC/Steel/PEB/Landfill/RE Wall/Retaining Wall/etc.)
- Storeys / spans / bays / functional requirements
- Site location (map → wind/seismic/SBC defaults)
- Usage category (residential / industrial / institutional / warehousing)
- Expected loads (or allow defaults)
- Constraints (plot size, height limits, setbacks, fire norms)
- Local code requirements (IS, NBC, EC, etc.)

If user says:

“G+2 building in Hyderabad for residential use on 200 sq yd plot.”

System automatically infers:

- **Structure** → RCC building
- **Location** → Hyderabad → Wind zone 39 m/s, Seismic Zone II
- **Plot area** → 167 m^2
- **Usage** → Residential → Live load defaults
- **Typical layout** → Auto-suggest 3–4 room layouts
- **Soil type** → Ask or assume medium (ask confirmation)
- **Foundation type** → Isolated footing likely, but ask
- **Staircase type** → ask or suggest
- **Water tank** → provide options: OHT, UG sump, both
- **Boundary wall** → ask
- **Retaining wall need** → ask based on slope

✔ 37.2.2 AI-Guided Question Flow (Dynamic Questionnaire Generator)

The AI dynamically generates questions depending on structure type:

For RCC Building

- Number of storeys
- Floor-to-floor heights
- Grid layout
- Wall thickness preferences
- Staircase preferences
- Beam depth constraints
- Slab type → conventional / flat slab / PT
- Lift core needed?
- Water tank size
- Soil bearing capacity or bore log upload
- Basement? Parking requirements?

For PEB/Industrial

- Clear span
- Eave height
- Crane load?
- Sheeting type
- Bay spacing
- Ventilation
- Skylights
- Mezzanine?
- Fire norms for industrial occupancy

For Landfill

- Capacity target
- Footprint
- Bench heights
- Slope stability requirement
- Bund height & berms
- KML upload option

AI decides question order based on dependencies.

✔ 37.2.3 Pre-Stored Layout Suggestions (Templates Library Integration)

Once enough data is collected, the system automatically recommends:

- **Best-fit RCC residential plans**
- **Best-fit industrial/PEB layouts**
- **Warehouse layouts**
- **Office layouts**
- **Landfill cell arrangements**
- **Retaining wall typical cross-section templates**

User can:

- Accept template
- Modify template
- Upload their own sketch (image/PDF)
- Upload AutoCAD/DXF
- Upload KML/SHP for geospatial works

System then **vectorizes** or **parameterizes** the input.

✓ 37.2.4 Sketch/Plan Understanding Module

Users may upload:

- A hand-drawn sketch (photo)
- A layout downloaded from internet
- A PDF plan
- A DWG/DXF file

AI will:

1. **Detect walls/beams/columns/layout elements**
2. **Reconstruct a parametric plan**
3. **Extract grids**
4. **Identify structural system**
5. **Check for building code anomalies**
6. Propose **optimized column grid** if needed
7. Prepare geometry for 3D model builder

This becomes input to **Part 8 Geometry Builder** and **Part 9 Structural Modeler**.

✓ 37.2.5 Auto-Workflow Builder

After understanding the user's requirement:

- Generates a project workflow:
    1. Concept →
    2. Architectural layouts →
    3. Structural system selection →
    4. Geometry creation →
    5. Analysis →
    6. Design →
    7. Drawings →
    8. Estimation →
    9. Execution →
    10. QA/QC →
    11. Closure.

Each step has:

- Duration
- Cost
- Required inputs
- Execution sequence
- Dependencies

Modules are activated based on subscription.

✓ 37.2.6 Auto-Cost Segregation per Step

Each module has its own fee:

- Basic layout generation
- Structural analysis
- Design
- Detailed drawings
- Estimation
- Quantity take-off
- Compliance checks

- Soil-structure interaction
- Value engineering
- Project planning
- Execution monitoring
- QA/QC
- BIM model export
- Digitized DPR generation

User only pays for activated flow.

37.3 Technical Architecture

37.3.1 Components

1. **Intent Parser**
2. **Missing Data Detector**
3. **Questionnaire Generator**
4. **Template Recommendation Engine**
5. **Sketch Understanding Engine**
6. **Workflow Engine**
7. **Cost Segregation Engine**

37.3.2 Integrations

- Model Builder (Part 8)
- Analysis Engine (Part 9)
- Drawing Engine (Part 10)
- Estimation Engine (Part 11)
- SSR/DSR Rate Libraries (Part 24)
- AI Site Assistant (Part 32)

37.4 Inputs

- Natural language user description
- Location
- Uploaded sketch/plan/CAD
- Preferred materials
- Budget
- Plot dimensions
- Site photos
- Soil report (optional)

37.5 Outputs

- Fully structured engineering dataset
- Auto-generated questionnaire results
- Suggested plan/layout
- Model-ready geometry data
- A design workflow plan
- Estimated fees per module
- Estimated project cost ranges
- Preliminary risk flags

37.6 Performance Requirements

- Must respond to user's vague statements in < 2 seconds
- Must auto-generate missing questions < 0.5 seconds
- Must support > 100 structure types
- Must integrate with all design modules
- Must handle DWG/DXF up to 500 MB
- Must support geospatial files: SHP, KML, GeoJSON

37.7 Security & Compliance

- No storing user's sketches/DWGs unless user agrees
- All project files stored encrypted
- Indian data residency option
- Audit logs

37.8 Future Extensions

- Voice-based structural input
- Walk-through 3D understanding of a site using mobile camera
- Integration with drones for contour generation
- Real-time interactive Structural AI chat companion

📘 PART 38 — MODEL CHECKING & VALIDATION ENGINE

This module ensures that **every model—Architectural, Structural, Services, Site, BIM, 3D, FEM, Kratos-based, or user-uploaded—passes through a multi-layer validation pipeline** before advancing to design, estimation, or execution workflows.
This is mandatory to meet real-world expectations of engineers, architects, contractors, peer-review consultants, statutory authorities, and clients.

38.1 Purpose

The Model Checking & Validation Engine (MCVE) is the "gatekeeper" of the entire platform.
It performs:

- **Geometry validation**
- **FE mesh validation (Kratos/FEniCSx)**
- **Design rule validation (IS / ACI / AISC / EC codes)**
- **Architectural + Structural consistency checks**
- **Execution feasibility checks**
- **Constructability checks**
- **Regulatory compliance checks**
- **Drawing validation**
- **Interdisciplinary clash detection (Structural vs MEP vs Architectural)**

The MCVE ensures that **nothing invalid propagates into design, drawings, or cost estimation**.

38.2 Inputs

MCVE accepts models from:

1. **User-created layout**
2. **AI-proposed layouts**
3. **Uploaded CAD files (DWG/DXF)**
4. **Uploaded hand-sketch (image/PDF) → AI vectorization**
5. **Imported STAAD/ETABS/Revit IFCs**
6. **Kratos/FEniCSx-generated meshes**
7. **Procedural templates (G+2, warehouse, etc.)**

38.3 Automatic Validation Categories

A. Architectural Validation

- Room area standards (NBC 2016)
- Door/Window clearances
- Staircase geometry
- Fire escape routes
- Parking norms
- Setbacks, permeability, open spaces
- Toilet placement + ventilation norms
- Corridor width
- Headroom clearances
- Natural light/ventilation percentages

B. Structural Pre-Check

- Member connectivity
- Column alignment across floors
- Beam continuity
- Slab spans within permissible limits
- Load path continuity
- Foundation load distribution
- Torsional irregularity
- Vertical & horizontal regularity
- Check for floating columns
- Soft/weak storey detection
- Slab punching risk areas

C. FEM Mesh Validation (Kratos/FEniCSx)

- Orphan nodes
- Zero-width or collapsed elements
- Extremely high aspect ratio elements
- Non-manifold edges
- Duplicate nodes
- Sharp angles < 5° without refinement
- Boundary condition gaps
- Material assignment consistency

D. Code Compliance Pre-Check

- Gravity load compliance (IS 875 Part 1–5)
- Seismic configuration compliance (IS 1893)
- Structural system classification
- Ductility requirements (IS 13920)
- Durability checks (IS 456, ACI 318)
- Steel member slenderness (IS 800 / AISC)
- Foundations (IS 1080, IS 2911)
- RE walls (BS 8006)
- Landfill slopes (CPCB, CPHEEO)

E. Constructability Checks

- Crane accessibility
- Scaffolding feasibility
- Transportability of members
- Site constraints assessment
- Beam placement sequence
- Shear wall liftability
- Bar bending feasibility
- Equipment turning radius

38.4 Clash Detection

The system performs a BIM-style clash test:

- Structural vs Architectural
- Structural vs Plumbing

- Structural vs Electrical
- Foundation vs Underground utilities
- Slabs vs services
- Staircase vs beams
- Ramp vs beams
- Retaining wall vs service trenches
- Landfill bund vs drainage lines

Clash severity levels:

- **Critical** — must fix before design
- **Major** — must fix before drawings
- **Moderate** — fix before GFC
- **Minor** — suggestion only

38.5 Model Repair & Auto-Correction

Where possible, the system **offers corrections**, including:

- Auto-aligning columns
- Auto-adjusting beam depths
- Auto-generating staircases meeting code
- Auto-fixing room dimensions
- Auto-resolving clashes with vertical/horizontal adjustments
- Auto-generating mesh refinements
- Auto-assigning materials
- Auto-reconnecting nodes
- Auto-regenerating load paths
- Auto-reassigning boundary conditions

Engineers may:

- **Accept**,
- **Reject**, or
- **Manually revise** corrections.

38.6 Report Generation

MCVE produces:

1. Validation Summary (Executive Level)

- Pass/Fail
- List of key warnings
- Estimated severity of issues
- Recommendation on proceeding to design

2. Engineering Validation Report

- Structural system overview
- Code compliance matrix
- FEM mesh quality statistics
- Member connectivity report

3. Clash Report (BIM-style)

- 3D visualization of clashes
- Screenshots
- Highlighted elements
- Commentary

4. Auto-Correction Log
A list of fixes applied by the AI.

38.7 Integration with 3D Viewer
The MCVE connects directly to the:
- **3D structural model viewer**
- **Cross-section viewer**
- **Plan & elevation viewer**
- **Mesh viewer (Kratos/FEniCSx)**

It overlays:
- Red: errors
- Yellow: warnings
- Blue: suggestions
- Green: resolved

Engineers can interactively:
- Inspect
- Correct
- Approve
- Revalidate

38.8 APIs Provided
POST /validate/model
Full architectural+structural validation.
POST /validate/mesh
FEM validation.
POST /validate/clash
Clash detection.
POST /validate/codes
Code compliance validation.
POST /validate/autofix
Automatic model correction.

38.9 Output Used By Other Modules
Outputs feed into:
- **Part 39 — Design Engine**
- **Part 40 — Detailing Engine**
- **Part 41 — Estimation Engine**
- **Part 42 — Scheduling Engine**
- **Part 43 — Construction Workflow Planner**

No model progresses without a valid MCVE pass.

38.10 Monetization
Each validation type is chargeable:
- Architectural validation
- Structural validation
- FEM validation
- Clash detection

- Auto-correction
- Peer-review style compliance report

Users on basic plans get **limited checks**, premium users get:

- Unlimited checks
- Full clash analysis
- Advanced code checks

38.11 Future Expansion

- Machine-learning based anomaly detection
- Municipality rulebook integration (Hyderabad, Mumbai, Delhi etc.)
- AI-based constructability scoring
- Steel/RCC detailing validation
- DfMA compatibility checks
- Green building compliance (IGBC, GRIHA)

✔ **PART 38 Completed**

📘 PART 39 — ADVANCED AI REASONING ENGINE & KNOWLEDGE-GRAPH BASED DECISION LAYER

Part 39 introduces the **AI Reasoning Core**, which becomes the "brain" of your entire Design → Estimate → Execute → Close-Out workflow.

This module enables the system to *understand, infer, recommend, query, and explain* all engineering, architectural, construction management, and project-control logic used throughout the platform.

This is NOT a basic LLM integration.

This is a **structured, standards-driven, rules-based, multi-modal engineering intelligence layer**.

39.1 PURPOSE OF THE AI REASONING ENGINE

The AI Reasoning Engine (AIRE) is responsible for:

1. Understanding user project intent

Examples:

- "Construction of G+2 building in Hyderabad..."
- "Industrial shed 40m × 120m with mezzanine..."
- "RE wall required along south boundary..."

AIRE interprets the instructions and breaks them into:

- Structural components
- Functional requirements
- Architectural expectations
- Site constraints
- Applicable codes (IS/IRC/MoRTH/NEC/IBC/Eurocode, etc.)
- Execution requirements
- Costing regimes
- Timelines

2. Generating missing questions automatically

AIRE determines what additional information is required:

- Soil conditions
- Functional loads
- Basements
- Clear heights
- Column grid preference
- Staircase locations
- Fire requirements
- Ventilation, water tank, service rooms
- Site access & circulation
- Green norms, EHS norms
- Urban bylaws
- Parking & setbacks

It asks the user until data is complete.

3. Converting user intent → Engineering Objects

AIRE generates:

- Column grids
- Load cases
- Structural systems

- RCC vs Steel vs Composite recommendation
- Preliminary sizing
- Architectural space zoning
- Boundary & contour extraction (if site KML provided)

4. Interpreting uploaded plans / hand sketches

Using:

- Vision Transformer (ViT)
- CNN-based plan extraction
- OCR for dimension reading
- Edge/contour detection
- Line-weight classification
- Symbol classification

AIRE converts the drawing into:

- Rooms
- Levels
- Slabs
- Beams
- Columns
- Openings
- Doors/Windows
- Stairs
- Water tanks
- Compound wall / gate layout

5. Producing engineering reasoning explanations

AIRE **justifies**:

- Why the load path is chosen
- Why column spacing changes
- Why shear walls suggested
- Why pile foundation required
- Why certain code clauses apply
- Why a wall thickness or steel grade is recommended

Everything is traceable.

39.2 KNOWLEDGE GRAPH (KG) ARCHITECTURE

AIRE is backed by a **Construction Knowledge Graph (CKG)** — the "database of relationships".

It stores:

- Structural hierarchies
- Load-flow logic
- Code clauses
- Material properties
- Component dependencies
- Connection rules
- Construction sequences
- Estimation formulas
- Rate analysis methods
- Site constraints

- Regional regulations

Nodes (examples)

- Building
- Level
- Column
- Beam
- Slab
- LoadCase
- Foundation
- SoilType
- MemberDesign
- CostItem
- SSR_Item
- TimelineActivity
- ConstructionRisk
- SafetyRequirement

Edges (examples)

- Column **supports** Beam
- LoadCase **acts_on** Member
- IS800_Cl_8.2 **governs** FlexuralDesign
- M20_Concrete **belongs_to** RCC
- UnitRate **maps_to** SSR_Item

The KG becomes the reasoning backbone.

39.3 AI PIPELINE WORKFLOW

Step 1 — User Intent Capture

Natural language input → LLM parsing → Internal structured JSON.

Step 2 — Missing Data Inference

Engine identifies missing parameters.

Step 3 — Template Selection

Based on:

- RCC / Steel / Composite / PEB / RE Wall / Retaining Wall / Landfill / Road / Industrial etc.
- Locality (Hyderabad → Seismic Zone II, wind zone Vb=39 m/s).
- Type of construction.

Step 4 — Model Construction Workflow

Creates:

- Column grids
- Levels
- Structural systems
- Foundation selection logic
- Load combinations
- Analysis approach (Kratos, FEniCSx, etc.)

Step 5 — Codes & Requirements Binding

Assigns:

- IS 875-1 dead loads
- IS 875-2 live loads

- IS 875-3 wind
- IS 1893 seismic
- IS 456 RCC design
- IS 800 steel design
- ACI / AISC / Eurocode depending on module

Step 6 — Reasoning Output & User Dialogue

AI explains:

- What it is doing
- Why
- What else may be required
- Risks
- Design alternatives

Step 7 — Handoff to Kratos/FEniCSx Processing

AI → Analysis engine:

- Geometry primitives
- Materials
- Meshing rules
- Load definitions
- Boundary conditions
- Solver settings

Step 8 — Post-Processing Interpretation

AI reads results and:

- Highlights failures
- Suggests redesign
- Summarizes critical utilization
- Warns about serviceability violations

39.4 REASONING MODES

(A) Deterministic Code-Based Reasoning

Using codified rules:

- E.g., "Minimum slab thickness per IS 456 Table 2"

(B) Data-Driven Insight

Based on stored examples:

- Typical grid sizes
- Typical reinforcement
- Good design patterns
- Anchor bolt spacing norms

(C) Analytical Reasoning

From Kratos results:

- Stress hotspots
- Excessive drifts
- Punching risks

(D) Risk Reasoning

Recommends:

- Soil investigation
- Waterproofing
- Shuttering supports

- Fire safety provisions
- QA/QC checks

39.5 AI CHECKPOINTS FOR ERROR-PREVENTION

1. Code-compliance watchdog

Detects:

- Missing load combinations
- Incorrect factors
- Wrong units
- Unsupported member definitions
- Missing bracings/shear wall requirements

2. Ambiguity checker

Flags:

- Undefined storey height
- No staircase location
- Wrong room sizes
- Overlapping beams or walls

3. Economic sanity checker

Detects:

- Unnecessarily heavy sections
- Over-reinforcement
- Over-sized footings
- Under-designed floors

4. Constructability checker

Highlights:

- Difficult shuttering
- Bar congestion
- Crane accessibility
- Difficult connections

39.6 INTEGRATION WITH OTHER MODULES

Integrates With:

- **Part 8**: 3D Model Builder
- **Part 10**: Optimization
- **Part 13**: Drawing Automation
- **Part 16**: Foundation Design
- **Part 18**: Estimation & SSR Integration
- **Part 22**: Construction Planning
- **Part 30+**: AI-powered Site Execution Monitoring

AIRE ties all modules together.

39.7 OUTPUTS OF THE AI ENGINE

1. Engineering Decision JSON

Used internally for further design.

2. Human-readable Explanations

Provided to the engineer.

3. Code Clause References

E.g.,

- *“IS 456 Cl. 23.2.1 governs minimum reinforcement…”*

4. Risks & Warnings

E.g.,

- “Shear wall required due to high drift.”
- “Soil SBC insufficient for isolated footings.”

5. Design Alternatives

- RCC vs Steel
- Shallow vs Deep foundation
- Frame vs Shear wall system
- PEB vs Conventional

39.8 DATA STORAGE & VERSIONING

- All decisions stored in version-controlled job folders
- Every reasoning step is logged
- Drawing revisions linked to reasoning logs
- Affordable for auditing, approvals, and dispute management

39.9 FUTURE EXTENSIONS

- Physics-AI hybrid models
- Automated generative architecture
- Automated BIM-to-Execution supervision
- AI-based safety monitoring
- Continuous learning from past designs
- Enterprise Knowledge Graph for large construction orgs (ReSL, L&T, Shapoorji, etc.)

PART 40 — POST-CONSTRUCTION PERFORMANCE MONITORING & DIGITAL TWIN OPERATIONS

40.1 Overview

After the physical construction of the building/warehouse/industrial structure is complete, the platform must transition into **Lifecycle Performance Monitoring & Digital Twin Operations**.

This module ensures:

- Real-time structural health monitoring
- Maintenance planning
- Asset performance analytics
- Digital twin visualizations
- Warranty/O&M compliance tracking
- Integration with IoT devices and periodic inspection inputs

This forms the final phase of the end-to-end workflow:

Concept → Design → Engineering → Drawings → Construction → Monitoring → Maintenance → Decommissioning.

40.2 Objectives

1. Provide a **long-term digital twin** of the built structure.
2. Create a system for **predicting deterioration** (steel corrosion, concrete cracking, foundation settlement, etc.).
3. Enable **smart maintenance planning** based on sensor data and inspection logs.
4. Track **energy, occupancy, environmental performance** for operational optimization.
5. Provide **alerting mechanisms** for structural abnormalities.
6. Maintain a **permanent digital repository** for post-construction documentation.
7. Offer **monetizable lifecycle services** for the platform (subscription-based O&M).

40.3 Inputs

Primary inputs

- As-built drawings and BIM model
- Geotechnical and environmental data
- Material properties as finally used during execution
- Construction logbooks & QC reports
- Sensor specifications (if installed)
- Maintenance contract details

Sensor inputs (optional but recommended)

- Strain gauges
- Tilt meters
- Accelerometers
- Temperature & humidity sensors
- Vibration sensors
- Ground settlement sensors
- Water ingress/leachate sensors (for landfills)

Manual data capture

- Periodic inspection photos

- Drone survey photos
- Crack mappings
- QA/QC logs
- Load and occupancy changes
- Environmental exposure updates

40.4 Core Features

40.4.1 Digital Twin Creation

- Create a **3D digital replica** of the structure based on:
  - As-built CAD/BIM,
  - Kratos/FEniCSx FE model (simplified for runtime),
  - Sensor location mapping,
  - Construction change logs.
- The twin is interactive with:
  - Orbit, pan, zoom
  - Section cut viewing
  - Layer toggles (beams, slabs, walls, utilities)
  - Real-time sensor overlays
  - Highlighting of overstressed members
  - Time-based animation of deformations or temperature gradients

40.4.2 Structural Health Dashboard

- Stress/strain trends
- Crack growth predictions
- Dynamic behavior changes (frequency/acceleration drift)
- Alert-level color coding
- Remaining useful life (RUL) predictions

40.4.3 Maintenance Planner

- Auto-generated maintenance tasks:
  - Painting schedules
  - Waterproofing & sealant inspection
  - Structural repairs
  - Foundation settlement monitoring
  - Drainage and leachate system checks (for landfills)
- AI-based prediction:
  - “Next 90 days likely issues”
  - “Critical attention areas”
  - “Recommended intervention priority”

40.4.4 Sensor Integration APIs

Support for:

- MQTT
- REST
- OPC UA
- Modbus
- CSV uploads
- IoT gateways
- Manual offline entries

All sensor data aggregated into a unified time-series database.

40.4.5 Performance Analytics

Includes:

- Energy consumption comparison
- HVAC performance (if applicable)
- Solar generation (optional)
- Occupancy impact on structural loads
- Floodwater/groundwater impact assessment

40.5 Automated Engineering Re-Check

Every time a new load pattern or event is detected (cyclone, earthquake, heavy rainfall):

1. System generates **derived FE loads**.
2. Uses simplified Kratos/FEniCSx model to run a structural re-analysis (low-resolution model for speed).
3. Identifies stresses/deflections vs. permissible limits.
4. Flags any requirement for human engineer review.

40.6 Landfill-Specific Monitoring (If Applicable)

For landfill designs:

- Slope stability monitoring
- Surface settlement (InSAR + GPS + tilt sensors)
- Leachate levels
- Gas generation performance
- Bund/berm integrity
- Phased landfill closure monitoring
- Vegetation/gas well coverage validation

40.7 Risk Scoring Engine

Each monitored element (beam, slab, footing, retaining wall, etc.) gets a **live score**:

- Structural Condition Index (SCI)
- Operational Safety Index (OSI)
- Environmental Risk Index (ERI)
- Maintenance Priority Score (MPS)

Scores are derived from:

- Sensor anomalies
- Visual inspections
- Degradation models
- Load history
- Environmental exposure

40.8 Integration with Project Execution Module

Connects back to:

- Contractor warranty obligations
- AMC contractor tasks
- Work order & invoice management
- Safety audit trails
- Compliance documentation

40.9 Notifications & Escalations

Alerts sent via:

- App notifications
- Email
- SMS
- WhatsApp API

Escalation chain based on severity:

- Minor → Engineer
- Moderate → Site Manager
- Major → Organization Safety Head
- Critical → Regulatory Authority (if required)

40.10 Deliverables Generated by the Module

- Digital Twin (interactive 3D)
- Monthly structural health reports
- Sensor logs & anomaly reports
- Maintenance schedule PDFs
- AI-generated advisories
- Repair scope documents
- Time-lapse animations of deformation/settlement
- Annual compliance dossier

40.11 Monetization Model

This module is **subscription-based**, with the following tiers:

Basic

- Manual inspection logging
- Basic 3D twin
- Monthly reports

Professional

- IoT integration
- Automated maintenance planning
- AI-based predictions

Enterprise

- Full sensor suite
- Kratos/FEniCSx automated re-analysis
- Regulatory compliance pack
- 24×7 monitoring dashboard

40.12 Technical Architecture Summary

- Backend: **Python + FastAPI**
- FE Engine: **Kratos Multiphysics / FEniCSx**
- DB: **Postgres + TimescaleDB** (for sensor data)
- 3D Visualization:
    - **Three.js**,
    - **PyVista** for FE results,
    - **CesiumJS** for site mapping (optional)
- Storage: MinIO / S3

- Mobile App (optional): Flutter
- Deployment: Docker + Kubernetes + Antigravity/AIS

40.13 Future Enhancements

- Machine vision for crack detection
- Drone-based automated façade inspections
- BIM-based BOM change tracking
- ML-based failure forecasting
- Integration with digital permitting systems
- Integration with insurance risk evaluations

📘 PART 41 — MULTI-DISCIPLINARY COORDINATION ENGINE (MDCE)

Integrated Architectural, Structural, MEP, Civil, and Site-Engineering Coordination Layer

---

41.1 Purpose of the MDCE

The purpose of this engine is to ensure that the system can:

1. **Synchronize all engineering disciplines** (Architectural, Structural, MEP, HVAC, Electrical, Fire, Plumbing, Site Works, Infra).
2. **Resolve inter-discipline conflicts automatically** using rule-based + AI-based clash detection.
3. **Generate coordinated drawings, schedules, and 3D views** showing discipline overlays.
4. **Ensure design constructability**, regulatory compliance, and safety.
5. **Provide a central model that evolves from concept → design → detailed engineering → execution.**

This engine becomes the **coordination brain** of the entire platform.

41.2 Inputs to the MDCE

The MDCE receives inputs from:

A. User Inputs

- Project type (e.g., G+2 residential, industrial shed, warehouse, school, mall, etc.)
- Functional requirements (rooms, halls, machinery, fire norms, occupancy loads)
- Site boundary + levels
- Materials preferences
- Local bye-laws
- Client constraints (budget, height limits, FAR, parking, etc.)

B. Discipline Engines

- **Architectural Layout Engine** (from earlier parts)
    - Floor plans, walls, rooms, openings
    - Staircases, shafts, corridors

- **Structural Design Engine**
    - Column grid, beams, slab systems, shear walls
    - Model geometry + load maps
- **MEP Engine**
    - Ducts, pipes, electrical trays, fire lines
    - Plant rooms, equipment, clearances
- **Site/Civil Engine**
    - Road alignment, drainage, retaining walls, tanks
    - Excavation + backfilling levels
- **BIM Data Layer / IFC module**

C. Libraries

- National Building Code (NBC) constraints
- Fire norms (NFPA, NBC Fire, State Fire Rules)
- Green building norms
- Manufacturer constraints (duct size limits, equipment clearance, etc.)
- Structural code constraints (IS 456, IS 800, IS 875, IS 1893, IRC, etc.)

41.3 Core Responsibilities

41.3.1 Automated Clash Detection

The MDCE performs multi-layer clash detection:

- Architectural ↔ Structural
- Structural ↔ MEP
- MEP ↔ MEP (HVAC vs Fire, Fire vs Plumbing, Electrical vs HVAC)
- Structural ↔ Site
- Tanks/Pipelines ↔ Foundations
- Roads vs Retaining Walls
- Basement vs Drainage Network

Clash Types Handled:

- Hard clash (object intersects)
- Soft clash (clearance violation)
- Rule-based clash (NBC mandated spacing, fire clearance zones)

The engine highlights clashes in:

- 2D plan view
- 3D model
- Section view
- Grid-wise clash tables
- Exportable Excel/PDF reports

41.3.2 Discipline Priority Resolution Matrix

The MDCE uses a **pre-defined hierarchy** to resolve conflicts:

Priority	Discipline	Notes
1	Structural	Grid and load path dominate; cannot be violated
2	Fire System	Life safety cannot be compromised
3	HVAC	Depends on ducts, large equipment
4	Plumbing	Wet-service zones preferred
5	Electrical	Flexible routing
6	Architecture	Adapts if needed except critical rooms

The MDCE automatically proposes resolution suggestions:

- Shift duct
- Increase floor height
- Shift beam to hidden beam
- Change ceiling levels
- Move pantry or toilet core
- Add access panel
- Alter column size slightly

- Move utility shaft location ("AI Shaft Recommender")

41.3.3 Real-Time 3D Coordination Workspace

Your requirement (like STAAD visualization) is integrated here:

- Pan, zoom, orbit 3D view
- Discipline toggle (A/S/M/E/P)
- Section box / cut-plane controls
- Grid-based view
- Color-coded discipline layers
- Clash blink highlighting
- Draw/modify elements directly in 3D

The UI supports:

- Snapping to grid
- Editing extrusions
- Moving ducts, pipes, beams
- Changing sizes
- Updating levels
- Locking elements
- Undo/redo

This becomes like a mini-BIM modeler inside your app.

41.3.4 Rules Engine for Design Coordination

Rule sets include:

- **NBC 2016** building bye-laws
- **Fire norms** (stair width, refuge floor, exits, travel distance)
- **Structural code constraints**
- **Plumbing fixture norms**
- **HVAC clearance rules**

- **Parking norms**
- **Architectural room size & ventilation norms**
- **Site grading rules**
- **Storm-water drainage slopes**
- **Minimum setbacks**

Every element placed by the user or auto-generated is validated automatically.

If constraints violated → engine highlights violations + proposes fixes.

41.4 Outputs of the MDCE

41.4.1 Coordinated 3D Model

- IFC
- GLTF
- JSON
- Kratos/FEniCS-compatible meshes
- 2D/3D visual previews

41.4.2 Discipline-Specific Drawings

- Architectural GA plans
- Structural plans
- MEP composite drawings
- Fire layout
- Plumbing layout
- Electrical routing plan
- Civil site layout
- Section & elevation sheets
- Coordination drawings

41.4.3 Reports

- Clash detection report
- Compliance report (NBC, Fire, Structural)

- Inter-discipline coordination summary
- Clearance violation report
- Constructability review

41.5 AI Features Inside MDCE

41.5.1 AI Layout Intent Interpreter

User describes:

"G+2 house, 3 bedrooms each floor, modern elevation, site 40x60 feet, prefer open kitchen"

AI generates:

- Walls
- Rooms
- Circulation
- Staircase
- Balcony
- Toilets
- Kitchen ventilation
- Parking layout
- Entry orientation
- Structural grid suggestion
- MEP tentative shafts
- Fire escape path
- Energy-efficient envelope

41.5.2 AI Model Optimizer

Across disciplines:

- Minimizes clashes
- Reduces duct/pipe reroutes

- Optimizes beam sizes
- Reduces cost
- Improves serviceability
- Suggests alternate grid spacing
- Proposes best retaining wall type
- Suggests water tank location
- Generates efficient cold/hot plumbing paths

41.5.3 AI Standards Checker

Reads the model:

- Flags rule violations
- Suggests fixes
- Generates compliance report

41.6 Integration With Kratos Multiphysics

The MDCE prepares:

- FEM-ready geometries
- Meshing definitions
- BCs & loads
- Element definitions
- Spatial domain mapping
- Material mapping

and passes them to:

- **Kratos Structural Mechanics**
- **Kratos Fluid**
- **Kratos GeoMechanics**
- **Kratos Contact / DEM**
- **Kratos Optimization**

41.7 Integration With Estimation & Project Management

After coordination:

- Quantities (steel, concrete, brickwork, MEP materials) → Extracted directly
- Work packages auto-generated
- Project schedule based on construction sequence
- Costs auto-mapped to SSR / custom rates
- Execution workflow auto-generated

41.8 Monetization Layer

Each of the following modules becomes a billable service:

- Architectural Layout Module
- Structural Design Module
- Kratos FEM Analysis Module
- MEP Layout Module
- Composite Coordination
- Clash Report Generator
- Compliance Checker
- Detailed Drawings
- Estimates & BOQs
- Construction Schedule Generator

User pays per module used.

41.9 Summary

The MDCE is the **heart of your multi-disciplinary engineering platform**, providing:

- True BIM-level coordination
- STAAD-like modification tools
- Clash detection

- Rule-based corrections
- 3D visualization
- Kratos-ready meshes
- Drawing + report generation
- Single-source-of-truth model

This is the engine that allows your app to scale across:

- Residential
- Commercial
- Industrial
- Warehouses
- Landfills
- Civil infrastructure
- MEP-heavy buildings
- Public buildings
- Mixed-use projects

📘 PART 42 — DEPLOYMENT BLUEPRINT (AI + ENGINEERING STACK)

(Part of the Full 300-Page Master PRD for the Universal Structural-Design-to-Execution Platform)

42.1 Purpose of This Section

This section defines how the entire platform—AI engine, structural solvers, drawing engines, costing engine, project management modules—will be **packaged, deployed, scaled, monitored, secured, and updated** across:

- Functional SAAS
- Engineering Offices
- Individual Consultants
- Cloud (Antigravity / AI Studio / Docker)
- Enterprise Environments (ReSL, L&T, Tata Projects, Shapoorji, etc.)

This guarantees:

- Predictable performance
- Easy onboarding
- Consistent deployments
- Zero-friction updates

42.2 Overall Deployment Strategy

The platform uses a **hybrid modular deployment architecture**:

42.2.1 Frontend

- Streamlit / React (user-friendly)
- Heavy 3D operations handled by:
    - **Kratos Multiphysics + WebGL Renderer**, OR
    - **Three.js + Kratos Output + glTF Viewer**

Fully browser-based:

- No desktop install
- Runs using WebSockets streaming model chunks and solver outputs

---

42.2.2 Backend

A multi-service backend running in separate containers:

Service	Technology	Purpose
AI Brain Service	Gemini API / OpenAI API	Conversational design + workflow generation
Engineering Model Builder	Python + Kratos	Mesh, BCs, Loads, Solver prep
Solver Service	Kratos Multiphysics (Structural+Geo+Fluid)	Real FE analysis
Code Check Engine	Python	IS 456 / IS 800 / IS 875 / IS 1893 / ACI / AISC / EC
Drawing Engine	Python + CADKernel + IFC Builder	2D/3D drawings
Estimator Engine	Python + SSR Database	BOQ + cost sheets
Workflow/PM Engine	Python	Schedules, Gantt, QA/QC, Project management
User & Billing Engine	FastAPI	Login, permissions, payments

42.2.3 Runtime Deployment Options

Option A: AI Studio + Antigravity (Primary Mode)

- AI Studio handles:
  - Prompting
  - Policy enforcement
  - User workflow generation
- Antigravity hosts:
  - Multi-container Docker environments
  - Database

    - SSL
    - Domain + CDN

This allows:

- Entire system to run without infrastructure knowledge
- Push-button deployments from browser
- GitHub integration for CI/CD
- Auto-restart on failure

This is the **default deployment for your company and clients.**

Option B: Local Docker Deployment

For clients without internet access or government jobs:

- Single compose file:

docker compose up --build

- Runs all microservices locally
- Optional GPU mode for Kratos
- Offline SSR costing
- Offline drawing engine
- Offline AI model (Gemma 7B/9B or Phi-4-mini)

Option C: On-Prem Enterprise Deployment

For large companies:

- Docker Swarm / Kubernetes cluster
- Multiple solver nodes
- Load balancing
- Per-project resource allocation
- Compliance logging
- Air-gapped admin panel

42.3 Deployment Pipeline

42.3.1 Developer Flow (Your Team)

GitHub → Unit Tests → Build Docker Images → Push to Registry → Deploy to Antigravity

Continuous Deployment triggered when:

- PR merged
- Version tag released

42.3.2 Customer Flow

User → AI Input → Workflow Generated → Model → Solve → Check → Drawings → Estimation → Execution

All modules deployed as independent microservices, so a user can:

- Do only estimation
- Do only design
- Do only drawings
- Or use the entire workflow

42.4 Scaling the Solver Engine

42.4.1 Kratos Solver Scaling

The solver runs as its own container:

- Auto-scales based on load
- Each analysis → new job
- Results streamed back to UI through WebSockets

Supports:

- Beam/Frame
- Shell/Plate
- Solid (Nonlinear)
- Contact (Footings, RE Walls)
- Soil Model (Mohr-Coulomb, HS Small, Drucker-Prager)

- Fluid (sludge, leachate)
- Thermal (chimneys)
- Multi-physics (if required)

42.4.2 GPU Acceleration

If available:

- CUDA-enabled Kratos
- Parallel solvers
- Faster meshing

If not:

- Standard CPU-only mode

42.5 Security & Access Control

42.5.1 User Levels

- **Owner**
- **Project Manager**
- **Design Engineer**
- **Drafter**
- **Estimator**
- **Client Viewer**
- **Billing User**

42.5.2 Built-in Policies

- Encrypted data at rest
- HTTPS
- Zero-trust identity
- Project data isolation
- Role-based endpoints
- Version-controlled data logs

42.6 Storage & Data Persistence

42.6.1 Databases

Data Type	DB
Users, Roles	PostgreSQL
Projects	PostgreSQL
Geometry Models	S3 or MinIO
Drawings	S3 or MinIO
Cost Libraries	PostgreSQL
AI Workflow	Firestore / PostgreSQL
Solver Results	S3 or MinIO

42.6.2 Versioning

Every change → new version:

- Model v1, v2, v3
- Drawings v1, v2
- Checks v1, v2

Traceable history.

42.7 Continuous Monitoring

- Real-time logs (Elastic/Antigravity)
- Solver health
- Memory usage
- API gateway health
- Error monitoring
- Auto-restart failing services

42.8 CI/CD Policies

- Safe rollbacks
- Staging environment
- Production approval
- Blue/green deployments
- Canary releases

42.9 Disaster Recovery

- Daily snapshot backups
- Cloud region replication
- 30 days retention
- One-click restore

42.10 Offline Mode

For government projects / client sites:

- Full app runs in offline Docker bundle
- Local SSR rates loaded via CSV
- Drawings export locally
- AI model runs locally (Gemma 7B)
- Reconnects to cloud later to sync

42.11 Licensing and Monetization

- Pay-per-service modules:
    - Architecture
    - Structural design
    - RCC design
    - Steel design
    - Landfill
    - PEB

    - Estimation
    - Drawings
    - Project execution
- Subscription plans:
    - Basic
    - Professional
    - Enterprise

42.12 Summary of Part 42

This part establishes the **complete deployment layer** for your universal structural-to-execution platform using:

- **Kratos + Docker**
- **AI Studio + Antigravity**
- **Modular microservices**
- **Scalable solver architecture**
- **Enterprise-grade security**

Fully aligned with everything we've discussed.

PART 43 — ADVANCED AI-ASSISTED MODEL VALIDATION & CORRECTION ENGINE

43.1 Purpose & Scope

This module ensures that every structural model—whether user-created, AI-generated, imported from STAAD/ETABS/Revit, or sketched—is **validated, corrected, optimized, and made design-ready automatically**.

The system becomes the "intelligent checker" that performs:

- Automatic model sanity checks
- AI-driven correction of geometry, loads, supports, connectivity
- Suggestions for improving design safety, cost, constructability
- Compliance checks against IS/ACI/AISC/EuroCodes
- Generation of layer-by-layer warnings and proposed fixes
- Real-time validation while the Engineer edits the 3D model
- A *continuous quality assurance AI loop*
- Prevention of common design failures

This eliminates rework, reduces errors, and ensures that models coming from any source become clean, analysable FEA-ready models.

43.2 Key Responsibilities of This Module

1. Geometry Validation

- Detect:
  - Duplicate nodes
  - Coincident nodes not merged
  - Members with zero length
  - Unconnected beams/columns
  - Orphaned elements
  - Inconsistent boundary conditions
- Provide a **map of geometric issues**:
  - Severity: Critical / Major / Minor
  - Location in 3D model
  - Recommended correction

2. Load Validation

- Check:
  - Load cases completeness (DL, LL, WL, EQ)
  - Missing tributary areas
  - Wrongly assigned loads
  - Illogical combinations (e.g., EQ without mass)
  - Inconsistent wind direction/orientation
- Suggest improvements:
  - Auto-generate IS 875 loads
  - Auto-generate IS 1893 load patterns
  - Auto-generate snow, drift, ponding loads
  - Auto-generate moving loads (if required)

3. Material & Section Checks

- Detect:
  - Missing material definitions
  - Wrong material in elements (e.g., RCC slab as steel)
  - Unsupported section naming

- Non-optimized sections
- Suggest:
  - Standardized ISMB/ISMC choices
  - Tapered optimization improvements
  - Reinforcement suggestions for RCC

4. Connectivity & Boundary Condition Checks

- Identify:
  - Nodes with conflicting constraints
  - Members without support
  - Incomplete foundation data
  - Pinned vs. fixed differences
  - Incorrect diaphragm assignments
- Auto-correction options:
  - Snap-to-grid
  - Auto-join
  - Auto-assign supports based on construction logic

43.3 Compliance Engine (IS / ACI / AISC / EC)

This module performs a **code pre-check** BEFORE actual design.
It validates:

IS Codes

- IS 456: RCC general provisions
- IS 800: Steel design
- IS 875 (1–5): Loadings
- IS 1893: Seismic
- IS 13920: Ductile detailing
- IS 2911: Foundations
- IS 1904: Soil considerations

ACI / AISC / EuroCodes

- ACI 318 – RCC
- AISC 360 – Steel LRFD
- Eurocode 2 – RCC
- Eurocode 3 – Steel
- Eurocode 8 – Seismic

For each standard, system produces:

- Compliance pass/fail
- Warnings
- Missing parameters
- Suggested default values
- Calculation references
- Fix options

43.4 AI-Driven Error Detection Mechanisms

1. LLM-Based Structural Logic Validator

The model is converted into a structured JSON:

```
{
  "nodes": [...],
```

"elements": [...],
"materials": [...],
"sections": [...],
"spring_supports": [...],
"loads": [...],
"load_combinations": [...]
}

The AI detects:

- Illogical geometry
- Redundant elements
- Missing supports
- Missing load paths
- Potential failure points
- Imbalanced tributary distribution

2. Vision + Geometry AI

If user uploads:

- Hand sketch
- PDF drawing
- Plan image

The AI detects:

- Grid layout
- Beams/columns
- Slab edges
- Stairs & walls
- Openings
- Load-bearing walls
- Outline geometry errors

Then it validates the extracted model and corrects inconsistencies.

43.5 Interactive 3D Validation Viewer

Integrated into the 3D interface (Part 8):

- Shows **model errors visually**
- Elements turn **Red** for critical, **Yellow** for warnings
- One-click correction buttons
- Cross-section viewer displays:
  - Slenderness issues
  - Incorrect connectivity
  - Wrong boundary conditions

Example visualization:

- Invalid column → highlighted in red
- Click → “Fix alignment” or “Merge nodes”

43.6 Diagnostic Dashboard

Panels

1. **Geometry Warnings**
2. **Load Issues**
3. **Support/Boundary Errors**

4. **Design Code Conflicts**
5. **FEA Stability Issues**
6. **Meshing Concerns (Kratos)**

Each item comes with:

- Screenshot preview
- 3D camera focus link
- Suggested fix
- Auto-fix button

43.7 Auto-Correction Engine

AI proposes and performs corrections like:

- Merge nodes within tolerance
- Align columns on gridlines
- Add missing beams
- Auto-complete load cases
- Recompute tributary widths
- Replace unstable supports
- Normalize load combinations
- Auto-fix meshing errors in Kratos

Every auto-correction generates:

- Reason for correction
- Before/after geometry snapshot
- Code reference
- Undo option

43.8 Pre-Solver Stability Checks (Kratos)

Before sending to the Kratos solver:

- Check determinacy
- Check singularities
- Evaluate stiffness matrix conditioning
- Detect floating nodes/elements
- Confirm mesh validity
- Confirm material model assignment
- Confirm boundary condition completeness

Outputs:

- Pass
- Pass with warnings
- Fail with diagnostics + fix options

43.9 Continuous Validation Loop

Every time the user modifies the model:

1. Model updated
2. Validator checks consistency
3. AI detects changes
4. Updated warnings displayed
5. Suggestions appear instantly
6. User accepts or ignores

This creates **real-time design correctness enforcement**.

43.10 API Interfaces

Import API

- STAAD .std
- ETABS .e2k
- Revit .ifc
- DXF
- JSON-based model exchange

Validation API

POST /validate-model
Inputs: Model JSON
Outputs: Validation JSON + Auto-fix JSON

Correction API

POST /apply-autofix

43.11 Output Deliverables

- Validation Report (PDF)
- Auto-correction logs
- Model-health score (0–100)
- Recommendations (safety, cost, optimization)
- Pre-design checklist
- Ready-for-design status
- Design risk indicators

43.12 How It Helps Users

- Engineers avoid common modelling mistakes
- Clients get reliable designs
- Drastically reduces failures in analysis stage
- Ensures high-quality models for AI-driven design
- Reduces design iterations
- Prevents disasters due to modeling mistakes
- Improves trust in the system

43.13 Integration With Other PRD Sections

PRD Section	Integration
Part 8 – 3D Modelling	Provides real-time model checking while user modifies the model
Part 9 – Analysis Engine	Ensures clean model before sending to Kratos
Part 10 – Design Engine	Ensures compliance with IS/ACI/AISC/EC
Part 12 – Cost Estimation	Removes errors that cause wrong BOQ
Part 18 – AI Generative Architecture	Ensures AI-generated models are clean
Part 21 – Templates	Validates templates for universal correctness

📘 PART 44 — AI-Aided Construction Sequencing, Scheduling & Automation Module

This module provides **end-to-end AI-powered planning, sequencing, scheduling, and construction automation**, deeply integrated with the design, estimation, and workflow engines.

It ensures that after the user completes **Design → Drawings → BOQ → Estimation**, the system can automatically:

- Generate a **construction sequence**
- Create a **resource-loaded schedule**
- Flag **project risks**
- Auto-identify **critical path**
- Produce **cashflow curves**
- Simulate execution progress
- Provide **AI-based work-front management**
- Monitor delays, re-plan, re-sequence

This is designed to function like a hybrid of **Primavera P6 + MSP + BIM-4D/5D tools**, but fully automated.

44.1 MODULE OBJECTIVES

✓ Automatic sequencing from design models

AI reads the structural/architectural designs, understands every component (columns, beams, slabs, walls, foundations, finishes, MEP items