

☆ 1. SYSTEM USERS (3 ROLES)

1. Admin / Backend

Controls all master data

Controls user, rates, manpower standard, line items, etc.

2. Supervisor / Site User

Enters daily work data

Cannot see rates

Can only see their own entries

3. Validator

Validates, edits, rejects, finalizes data

Can see rates & edit the submission

Sends back drafts to supervisor

☆ 2. FULL SCREEN LIST (FINAL) — 8 SCREENS TOTAL

Common

1. Login Screen

Admin

2. Admin Dashboard (navigation only)
3. Master Data Setup
4. User Management
5. View Final Approved Data

Supervisor

6. Data Entry Screen
7. View/Edit Own Submissions

Validator

8. Validation Panel (List + Detailed Review)

☆ 3. MASTER DATA (ADMIN) — FULL DETAILS

Admin/Backend sets the following **6 items**:

1. User Database

- Name
- Emp ID
- Role (Admin / Supervisor / Validator)

- Username / Password
- Active / Inactive

2. Work Order Number (Dropdown for Supervisor)

Admin creates work orders:

- WO-001
- WO-002
- etc.

3. Line Items Under Each Work Order

(Example)


WO-001 →

- Excavation
- PCC
- RCC

Each one displayed as dropdown for Supervisor.

4. Rate of Each Line Item

Stored in backend.

 **Not visible to Supervisor**

✓ Visible to Validator

✓ Rate × Quantity used for revenue calculation

5. UOM of Each Line Item

Supervisor can see, but:

✗ cannot edit

✓ Validator can edit (if needed)

6. Manpower Deployment — Standard Composition

Supervisor:

✗ cannot edit

Validator:

✓ can edit

Example:

1 Team = 1 Supervisor + 4 Labor + 1 Mason

☆ 4. SUPERVISOR DATA ENTRY — FULL INPUTS

Supervisor selects:

1. Work Order No (Dropdown)

Populated from Admin

2. Line Item (Dropdown)

Filtered based on selected WO

3. UOM (Auto Display – Non editable)

4. Standard Manpower (Auto Display – Non editable)

Fetches from backend

Supervisor Manual Inputs

5. **Work Quantity** (Manual entry)
6. **Manpower Actual Deployed** (Manual entry)
7. **Material Consumed** (Manual entry, optional)

System Auto Calculation (Not separate screen)

- Standard vs Actual manpower
- Revenue = Quantity × Rate (hidden from Supervisor)

On Submit

- Sent to Validator
- Status becomes: **Pending Validation**

☆ 5. VALIDATOR REVIEW — FULL ACTIONS

Validator receives every entry from supervisor.

Validator can see:

- Work Order
- Line Item
- Quantity
- UOM
- Standard Manpower
- Actual Manpower
- Material Consumed
- **Rate (Admin only allowed)**
- **Revenue Calculations**

Validator can perform:

✓ Edit

- Quantity

- UOM
- Standard Manpower
- Rate
- Any field

✓ **Approve**

Entry becomes **FINAL**

✓ **Send Back to Supervisor (Reject)**

Must enter **Remarks**

Supervisor receives rejected entry

Supervisor edits + resubmits

Sent again to validator

☆ **6. ADMIN FINAL DATA VIEW**

Admin dashboard shows:

- Work Order
- Line Item
- Quantity
- Standard vs Actual manpower
- Revenue (based on rates)
- Validator status
- Date
- Supervisor name

Admin only sees **Approved** data.

☆ **7. COMPLETE WORKFLOW (START → END)**

STEP 1 — Admin Creates Data

Admin creates:

- Users
- Work Orders
- Line Items
- UOM
- Rates

- Standard Manpower

Everything stored in backend.

STEP 2 — Supervisor Logs In

Supervisor → Dashboard

STEP 3 — Supervisor Creates Work Entry

Supervisor fills:

- WO
- Line item
- Quantity
- Actual manpower
- Materials

Auto-fetched:

- UOM
- Standard manpower

Submit → Goes to Validator

STEP 4 — Validator Validates the Entry

Validator sees:

- All supervisor-entered values
- Hidden data (Rate, Revenue)
- Calculation summaries

Validator actions:

1. **Approve** → Final submission
 2. **Reject** → Sent back to supervisor
 3. **Edit** → Then approve
-

STEP 5 — Supervisor Resubmits (if rejected)

Supervisor sees remark

Edits

Resubmits
Goes back to validator

STEP 6 — Admin Views Final Approved Dataset

Admin sees only **approved** records
Can export reports
Can perform further backend analysis

☆ 8. FINAL OUTPUT — COMPLETE STEPS (AS REQUESTED)

Here is your **final readable list of complete steps**:

☆ COMPLETE STEPS (END-TO-END)

ADMIN STEPS

1. Create Users + Roles
 2. Create Work Order Numbers
 3. Add Line Items to each WO
 4. Assign UOM, Rates, Standard Manpower
 5. Activate/Deactivate users
 6. View only *approved* data
-

SUPERVISOR STEPS

1. Login
2. Select Work Order
3. Select Line Item
4. Auto-view UOM & Standard Manpower
5. Enter Work Quantity
6. Enter Actual Manpower
7. Enter Material Consumed
8. Submit to validator

If rejected:

9. See validator remarks
 10. Edit & Resubmit
-

VALIDATOR STEPS

1. Login
 2. See list of Pending submissions
 3. Open each entry
 4. View hidden data (Rate & Revenue)
 5. Edit if required
 6. Add remarks (if rejecting)
 7. Approve OR Send Back to Supervisor
-

SYSTEM AUTO LOGIC

1. Standard Vs Actual manpower difference
 2. Revenue target vs actual
 3. Automatic status updates (Pending → Rejected → Resubmitted → Approved)
-

☆ FINAL STATUS FLOW

Supervisor → Validator → Admin (view only)

✓ READY FOR IMPLEMENTATION

If you want next steps, tell me:

“Create wireframes”

or

“Create tables / backend schema”

or

“Create PowerApps screens”

or

“Create React frontend screens”

I will generate the full UI/UX immediately.

You said:

how the supervisor, validator got their userid and password who will create those

ChatGPT said:

Here is the exact answer based on **your system design** and **standard industry workflow**:

☆ WHO CREATES THE USER ID & PASSWORD FOR SUPERVISOR AND VALIDATOR?

✓ **The Admin / Backend Team creates ALL user accounts.**

This matches your document:

SI No 1 — User Database (Admin/Backend)

Meaning:

- Only **Admin** has permission to create **users**.
- Supervisors and Validators **cannot create their own accounts**.

☆ HOW THE USER ACCOUNT IS CREATED (Step-by-Step)

Step 1 — Admin Opens “User Management” Screen

Admin enters:

- Name
- Employee ID
- Role (Supervisor / Validator / Admin)
- Mobile number (optional)
- Email (optional)

Step 2 — Admin Assigns Login Credentials

Admin sets:

- **User ID (Username)**
- **Password**

Or system can auto-generate a password.

Step 3 — Save User

User is added to the system with:

- Role-based access
- Active status

*****remember

Initially store every thing in local .db file ..

In my new git repo which I am already in..

My super admin userid-pradhansayan2@gmail.com password-Sayan@0306

2. And do not in my previous netlify id..as a project already hosted.. I will give you a new one later..currentlt I want to test in my local machich so server and client should run on my local host..

** when a action is taken submit ..reject/etc any thing ..button a loader should shown in realtime until the work done..to have better experience for the user.