ChatGPT

Ocean Pearl Ops V2 – Production Audit Report (January 26 2026)

1️⃣Executive Verdict

After extensive hands-on testing across the CEO, location manager and unit operator roles, I **do not** consider Ocean Pearl Ops V2 safe for unmonitored daily use in production.
There are severe permission and data-integrity flaws (e.g., *View As* mode allows writes [1], expense requests cannot be submitted [2]), confusing context switching without safeguards, and UI bugs that can silently corrupt data or mislead staff.
Confidence Level: *30 percent* – The core architecture shows promise, but critical controls are missing and numerous bugs undermine trust.

2️⃣Critical Issues (Must Fix)

#	Issue & Evidence	Why It's Dangerous	Exact Fix Recommendation
C1	**View As mode grants write permissions** – When the CEO enters *View As* mode to audit a unit operator, the UI states it is read-only. However, I was able to create and save a "Receive Goods" invoice in this mode – the system responded with a normal "Invoice Saved" alert and generated a batch number [1]. The invoice was actually created, proving that the supposed read-only mode performs full writes.	CEOs or auditors could inadvertently manipulate production data under the assumption that they are only observing. This breaks trust and can lead to fraudulent or duplicate transactions. The risk is amplified because the invoice creation uses the CEO's name in logs, destroying the separation of duties.	Implement a strict server-side permission check for *View As* sessions: any POST/PUT/DELETE request triggered while in *View As* should be rejected. In the UI, disable all inputs and hide action buttons. Add automated tests to verify that no write APIs can be called from this mode.
C2	**Missing first-action confirmation in Operate As mode** – The "Operate As" mode warns in the modal that actions will be recorded, but once activated there is no additional confirmation before a write. I was able to save a purchase invoice immediately without any "Are you sure?" prompt [3].	A CEO experimenting with the system could accidentally commit data to production (e.g., approve a payment) while thinking they are still safe to explore. Without a clear checkpoint, mistakes can easily occur.	After entering Operate As mode, intercept the first write action (e.g., first POST/PUT) and present a modal confirmation: "You are about to make a live change as Unit Operator/ Manager. Continue?" Require the CEO to confirm once per session.

#	Issue & Evidence	Why It's Dangerous	Exact Fix Recommendation
C3	**Expense request submission broken** – Unit operators cannot submit expense requests. In the "Create Request" modal, entering a valid amount (≥100 IDR) and description leaves the modal open and no request is created [2]. Even after repeated attempts with valid data, the list of requests remains empty [4].	This bug completely blocks the expense reporting workflow. Without the ability to raise a request, field staff will resort to informal channels (WhatsApp, paper receipts), bypassing controls and creating untracked liabilities.	Fix the form submission logic on the front-end and back-end. Ensure the `Submit Request` button triggers a create-request API call and displays success/ error feedback. Add automated tests covering valid and invalid amounts.
C4	**No confirmation when switching location or unit** – The dropdowns for location and unit context switch instantly with no warning or visual delay. A distracted user could click the wrong item and start performing actions in the wrong plant [5]. There is no banner change when switching within a role.	Accidental context changes could result in transactions recorded against the wrong location or unit, producing incorrect financial and inventory data. Because there is no server-side guard, the wrong wallet could be debited or wrong report updated.	Add a confirmation dialog whenever a user changes the context (location, unit). The UI should display the current context conspicuously (e.g., colored header). Backend requests should include the context and reject actions if mismatched with the user's session.
C5	**Password resets are insecure and error-prone** – Resetting a user's password displays a tiny alert with a generated password (e.g., "Ops164722") which cannot be copied easily and is easy to misread. After clicking OK the site opens a blank `chrome://newtab`, losing the admin page [6]. Attempts to use the temporary password often failed during login, causing confusion [7].	Admins may repeatedly reset passwords hoping to capture the correct string, resulting in account lockouts. The random password may be captured incorrectly (e.g., zero vs letter O) leading to help-desk requests and downtime.	Provide a proper modal with the temporary password clearly displayed and a copy button that writes to clipboard. Send the temporary password via email or SMS to the user. Stop opening a new tab after confirmation. Allow admins to set a custom password or force the user to set a new password on first login.

#	Issue & Evidence	Why It’s Dangerous	Exact Fix Recommendation
C6	**Manager accounts are read-only** – Logging in directly as a location manager shows a purple “READ ONLY” dashboard with no way to approve expenses or view the site wallet. The manager can only see recent activity and cannot execute their core responsibilities. This contradicts the intended workflow where managers review and approve expenses.	Managers will be forced to ask the CEO to act on their behalf or use the manager phone app, destroying autonomy and delaying operations. It also undermines the segregation of duties when only the CEO can operate as a manager.	Elevate manager permissions to include expense-review and wallet-management pages. Provide dedicated navigation for approvals and reporting. Keep read-only dashboards for auditors only.
C7	**Amounts shown as NaN in wallet management** – In the manager’s wallet, the “Recent Transactions” section displays amounts as `Rp NaN` for all rows [8]. This indicates a data parsing error.	Managers cannot see correct cash balances or transaction values, which compromises financial oversight. It also hints at underlying data type issues that could lead to mis-calculated totals in reports.	Identify the source of NaN values (improperly parsed numbers or missing fields) and fix the front-end formatting. Validate numeric data at the API layer to ensure amounts are always numbers.
C8	**Unexpected blank tabs after certain actions** – Several flows (saving invoices, resetting passwords, dismissing alerts, invalid production runs) cause Chromium to open a new blank `chrome://newtab` tab [6]. This disrupts the user’s workflow and can lead to data loss if they close the original tab mistakenly.	Opening extra tabs confuses users and makes them think the system has reset. It can interrupt unsaved forms and break the back/forward history stack.	Audit all `window.open` or redirect calls and remove unintended new-tab triggers. Confirm modals close gracefully without leaving the current context.

3️⃣Medium Improvements

These issues are not catastrophic but significantly degrade usability and should be addressed promptly.

1. **Unclear species dropdown options** – The “Receiving” form shows multiple `undefined (-)` entries in the species list [9]. Rename or remove invalid species options.
2. **Cannot add new vendors or expense types inline** – Unit operators cannot add a new vendor while receiving goods or filing an expense. They must use pre-existing vendors, forcing workarounds. Provide “Add Vendor” and “Add Expense Type” buttons with modals that create new records.

3. **Lack of confirmation when navigating away from unsaved forms** – Starting a request or invoice and clicking elsewhere discards data silently. Implement an “Unsaved changes, are you sure?” prompt when leaving mid-form.
4. **Confusing role names** – The user creation form lists “Loc Admin” but after creation the user is labelled “LOC MANAGER” [10] . Align terminology across the system to avoid confusion.
5. **Readability of generated passwords** – Use fonts that clearly distinguish `0` (zero) and `0` (capital O), and avoid mixing similar characters.
6. **Loading states & feedback** – Many actions show no spinner or toast; users cannot tell if an operation succeeded or failed. Add consistent loaders and success/error messages.
7. **Top-bar context indicator** – The location and unit dropdowns are small and could be overlooked. Add color coding or a persistent label indicating the current context, especially in Operate As mode, to reduce mistakes.
8. **API error handling** – When the server rejects a request (e.g., due to invalid weight in production), the front-end shows only a basic alert and then opens a blank tab. Show a detailed error message and keep the user on the same page.

4️⃣ Optional Enhancements

These are nice-to-have features that can improve efficiency but are not essential for safety.

1. **Bulk approvals for managers** – Allow location managers to approve or reject multiple expenses at once instead of one by one.
2. **Search and filter for transactions** – Provide filters by date, vendor, amount and status in the wallet and reports pages. This becomes critical as data volume grows.
3. **Mobile-responsive improvements** – Although the app is usable on desktop, some modals are difficult to scroll on mobile (e.g., the CEO Control Panel). Optimise for smaller screens.
4. **Audit trail export** – Allow downloading of a full audit log (who acted as whom, what they did) as CSV for external audits.
5. **User self-service password reset** – Permit users to request a password reset via email/SMS, reducing admin workload.

5️⃣ Final Recommendation

Given the severity of the permission errors and broken core workflows, **I do not recommend live operational use without significant fixes.** The platform requires remediation of critical issues listed above before it can be trusted for daily operations. Once the bugs are resolved and guardrails added, another audit should be performed to verify readiness.

---

[1] [3] [9] OPS — VIEW AS: KAIMANA / UNIT OP
https://oceanpearl-ops.web.app/receiving

[2] [4] [8] OPS — OPERATE AS: KAIMANA / UNIT OP
https://oceanpearl-ops.web.app/wallet

[5] OPS — CEO (Global Admin)
https://oceanpearl-ops.web.app/

6 OPS — OPERATE AS: KAIMANA / UNIT OP
https://oceanpearl-ops.web.app/production

7 OPS-Management
https://oceanpearl-ops.web.app/login

10 OPS-Management
https://oceanpearl-ops.web.app/admin