Station-2100 Development – Intermediate Report

This document provides a comprehensive intermediate-stage summary of the Station-2100 development project. The Station-2100 system is a modernized, web-based inventory and workshop management platform being rebuilt with Next.js/Vite, Supabase, Node.js, Prisma, and supporting automation scripts. The goal is to replace the legacy Windows Forms application with a scalable, maintainable, and fully cloud-ready web system.

# 1. Current Progress

The following milestones and fixes have been completed during this stage of development:

• PowerShell Automation Scripts  
 - Created Station-2100.ps1, dev-health.ps1, and sync-related scripts.  
 - Added helpers to kill stray node.exe processes, start the Vite dev server, and perform health checks on sync endpoints.  
 - Verified that ALLOW\_SYNC=1 is properly applied in .env.local.

• Sync and Health Checks  
 - Cleaned up dev-sync-plugin.ts to ensure only one export default function.  
 - Confirmed .env.local is UTF-8 encoded with no BOM/newline corruption.  
 - Verified health endpoints:  
 /\_\_sync/ping → {ok:true, pong:true}  
 /\_\_sync/status → {ok:true, allow:true}

• Cursor Integration  
 - Built all-in-one prompts for Cursor to perform full quality and startup passes with zero prompts.  
 - Prompts handle duplicate code, environment issues, and auto-confirm health checks.  
 - This reduces manual intervention during dev startup.

• GitHub Automation  
 - Implemented push-changes.ps1 that stages, commits (with timestamp), rebases, and pushes to GitHub.  
 - This enforces safe, consistent version control practices across machines.

# 2. Current Status

The dev server is currently functional and returns valid responses from health checks. However, occasional instability occurs when Cursor hangs mid-task. To mitigate this, we restart sessions using a context summary, ensuring smooth continuation without redoing all setup steps.

# 3. Issues & Challenges

1. Environment Variable Consistency: Ensuring .env.local remains identical across different machines is crucial. Minor encoding or whitespace issues have previously caused errors.  
2. Cursor Stability: Cursor occasionally freezes during long tasks, requiring restarts.  
3. GitHub Sync: While push-changes.ps1 is in place, a complementary pull-changes.ps1 script is needed for safer collaboration across multiple environments.

# 4. Next Steps

The following priorities have been identified for the next phase of development:

• Environment Reliability  
 - Automate validation of .env.local before dev startup.  
 - Provide a shared canonical template for all developers.

• GitHub Workflow Enhancements  
 - Implement pull-changes.ps1 for automated fetch, rebase, and sync.  
 - Add conflict resolution helpers for smoother collaboration.

• Feature Development  
 - Job Cards: Create forms and workflows to track workshop tasks, resource usage, and labor.  
 - Inventory: Expand tracking with batch numbers, expiry dates, and supplier references.  
 - Customers & Suppliers: Build master-detail records with contact info, transaction logs, and reporting.

• Documentation  
 - Continue maintaining DEVLOG.md and README.md for developer onboarding.  
 - Expand architecture diagrams and schema documentation.

# 5. Development Workflow Restart Guide

To restart development in Cursor or Lovable after a hang or fresh session, use the following approach:

1. Paste the latest development summary into the assistant.  
2. Run Station-2100.ps1 to initialize environment and dev server.  
3. Confirm health endpoints return {ok:true} responses.  
4. If GitHub sync is needed, use push-changes.ps1 or the upcoming pull-changes.ps1.  
5. Continue building features incrementally, always documenting progress.

# 6. Conclusion

The Station-2100 project has reached a stable intermediate stage. Core development workflows, automation scripts, and health checks are functional. GitHub automation and Cursor integration have reduced manual overhead, but improvements in environment consistency and collaborative sync tools remain necessary. The foundation is now solid enough to proceed with feature-rich modules such as Job Cards, Inventory, and Customer/Supplier management. This report serves as both a checkpoint and a guide for resuming or extending development without losing context.