



Document	Software Design	Release	17/08/26 V1.0
Project	BRN WMS	Project code	200-WMS-26-001-00
Project period	05/01/26–24/08/26		

Basis

This design was from the current repository structure and class list rather than authored before implementation. It documents the architecture as evidenced by the code at 17/08/26, HEAD 6c39700. Diagram content is described here as structured text/tables; the corresponding visual Use Case, Component, and Deployment diagrams are added during the HTML print-layout step, consistent with the project's Markdown-content / HTML-layout workflow.

HIGH LEVEL DESIGN

Use case actors and actions

Actor	Description	Representative actions
Owner	Company owner; highest-privilege role within a company	Full access to configuration, users, and all operational modules within the company
Admin	Administrative user within a company	Manage company settings, users, application access, and all operational modules
Staff	Operational user	Perform warehouse, sales, purchasing, and finance transactions within permitted scope
Viewer	Read-only user	View authorized screens and reports without creating/editing transactions
System (Node.js scheduler)	Automated actor	Executes scheduled stock/GL aggregate maintenance and low-stock/overdue-invoice alerts
Email/SMTP service	External actor	Delivers onboarding, password-reset, and notification email sent by the application

Component layers

Layer	Composition	SR reference
Presentation	PHP page views under each <code>app/<module>/</code> directory; responsive UI; <code>app/assets/js/custom.js</code> and supporting JS/CSS	SR02:002, SR06:002, SR06:003
Business logic	31 manager/service classes in <code>app/assets/utills/classes/</code> (Section "Software Unit" below), invoked from page controllers	SR02:003, SR03:001–SR03:011
Data access	Manager classes query two MariaDB databases directly (no separate ORM layer); <code>StockTablesTrait</code> centralizes shared stock-table access patterns	SR02:004, SR06:001, SR08:001–SR08:005
Real-time/scheduled services	Node.js <code>server.js</code> (Socket.IO notification relay) and <code>scheduler.js</code> (cron-style aggregate/alert jobs), managed by pm2 (<code>ecosystem.config.js</code>)	SR03:011, SR06:004, SR06:005
External integration	SMTP email delivery (<code>SmtplibManager</code>); browser Socket.IO client for real-time notifications	SR06:004, SR06:005

Deployment tiers

Tier	Composition	SR reference
Client	Browser (Chrome/Edge/Firefox), responsive UI, Socket.IO client connection	SR06:002, SR06:003, SR06:005
Web tier	PHP 8+ application; manual LAMP install (<code>setup.php</code>) or <code>docker/php</code> container (php-apache, config generated from <code>.env</code> at entrypoint)	SR01:003, SR02:005
Data tier	MariaDB, two databases: <code>wms</code> (identity/company) and <code>wms2</code> (WMS/accounting); manual install or <code>docker/mariadb</code> container	SR06:001, SR08:001–SR08:004
Real-time/scheduler tier	Node.js <code>server.js</code> + <code>scheduler.js</code> under pm2; manual install or <code>docker/node</code> container	SR03:011, SR06:004, SR06:005

Tier	Composition	SR reference
External	SMTP server (configured per company via <code>company_smtp</code>)	SR06:004

Software Unit

One unit per business-logic manager class, plus the two Node.js services. Each unit's SR reference is its owning module in the SRS (Section SR03).

ID	Description	Functional interfaces detail	SR reference
UN01	UserManager	User CRUD, role assignment, application-access flags	SR03:001
UN02	PasswordManager	Password hashing (bcrypt), validation, change	SR03:001, SR01:004
UN03	PasswordResetManager	Forgot-password token issuance and reset flow	SR03:001
UN04	CompanyProfileManager	Company profile CRUD	SR03:002
UN05	CompanySettingManager	Company-level system settings	SR03:002
UN06	SmtManager	Per-company SMTP configuration and mail dispatch	SR03:002, SR06:004
UN07	WarehouseManager	Warehouse/storage/bin master data and capacity/occupancy	SR03:003, SR03:004
UN08	ProductManager	Product and product-category master data	SR03:003
UN09	ContactManager	Contact type and contact master data	SR03:003
UN10	StockManager	Stock-in/out/transfer, lot/serial/expiry, balances	SR03:004
UN11	StockSourceManager	Stock source/traceability resolution	SR03:004
UN12	StockTablesTrait	Shared stock-table query/aggregation logic reused by stock-facing managers	SR03:004

ID	Description	Functional interfaces detail	SR reference
UN13	BarcodeManager	SKU/location barcode label generation and scan handling	SR03:004
UN14	EtlStockManager	Stock aggregate (ETL) table maintenance	SR03:009, SR05:002
UN15	QuotationManager	Quotation lifecycle	SR03:005
UN16	OrderManager	Sales-order lifecycle	SR03:005
UN17	InvoiceManager	Sales-invoice lifecycle	SR03:005
UN18	ReturnManager	Sales-return / credit-note lifecycle	SR03:005
UN19	PurchaseRequestManager	Purchase-request lifecycle	SR03:006
UN20	PurchaseOrderManager	Purchase-order lifecycle	SR03:006
UN21	SupplierReturnManager	Supplier-return lifecycle	SR03:006
UN22	ReceiptBillingManager	Receipt billing lifecycle	SR03:007
UN23	ReceiptManager	Receipt lifecycle	SR03:007
UN24	PaymentBillingManager	Payment billing lifecycle	SR03:007
UN25	PaymentManager	Payment lifecycle	SR03:007
UN26	PostingManager	Chart of accounts, account formulas, journal, GL posting	SR03:008
UN27	ReportManager	Operational and financial report generation	SR03:009
UN28	DocumentNumberManager	Controlled document-number sequence generation	SR03:010
UN29	BatchActionManager	Bulk/batch record operations	SR02:003
UN30	OperationLockManager	Posting-window / operation-lock enforcement	SR07:002, SR09:003
UN31	UsageGuard	Company usage-package/transaction-quota enforcement	SR07:002

ID	Description	Functional interfaces detail	SR reference
UN32	FileUploader	Supported file-attachment upload handling	SR03:004
UN33	nodejs/server.js	Socket.IO real-time notification relay	SR03:011, SR06:005
UN34	nodejs/scheduler.js	Scheduled stock/GL aggregate maintenance and low-stock/overdue-invoice alerts	SR03:011, SR05:002

User Interface Design

The delivered UI is an implemented, responsive PHP application (login, dashboard, warehouse/inventory, sales, purchasing, finance, accounting, reports, settings). Because the UI already exists as running screens rather than pre-implementation mockups, wireframe capture is not reproduced here; representative screenshots are added during the HTML print-layout step where useful, consistent with the Software User Documentation (work product 18), which documents actual screens.

Approval

Prepared by

Name: Thanakorn Sathitwitayakul

Role: Developer

Signature: _____

Date: _____

Reviewed by

Name: Apirach Supattaratpateep

Role: Project Manager

Signature: _____

Date: _____

Reviewed and authorized by

Name: Seri Viriyasakultorn

Project roles: Project Sponsor / Customer Representative / Authorized Approver

Position: Managing Director

Company: B.R.N. Enterprise Co., Ltd.

Signature: _____

Date: _____