Deliverable 3
DESAIN TAKTIS
II3160 - Integrated Systems Technology

Disusun oleh:
Theresia Ivana Marella Siswahyudi - 18223126

Program Studi Sistem Dan Teknologi Informasi
Sekolah Teknik Elektro Dan Informatika
Institut Teknologi Bandung
2025

DAFTAR ISI

LATAR BELAKANG

Dalam sistem manajemen gudang (*Warehouse Management System* / WMS), pengelolaan data dan proses bisnis yang kompleks menuntut adanya pemodelan domain yang terstruktur dan konsisten. Setiap aktivitas, mulai dari penerimaan barang (*inbound*), penyimpanan dan pengendalian stok (*inventory*), hingga pengeluaran barang (*outbound*), saling berkaitan dan bergantung pada data yang berubah secara dinamis. Kondisi ini menimbulkan tantangan dalam menjaga konsistensi informasi antarproses, terutama ketika berbagai bagian sistem menggunakan istilah, aturan bisnis, dan format data yang berbeda. Tanpa adanya keseragaman bahasa dan model domain yang terdefinisi dengan baik, sistem berpotensi mengalami duplikasi logika, kesalahan komunikasi antar tim, serta kesulitan dalam pengembangan dan pemeliharaan jangka panjang.

Untuk menjawab permasalahan tersebut, pendekatan Domain-Driven Design (DDD) digunakan sebagai kerangka kerja konseptual yang menekankan pentingnya keselarasan antara pemahaman bisnis dan implementasi teknis. Setelah domain WMS diuraikan ke dalam *bounded context* dan hubungan antarkonteksnya dipetakan, tahap berikutnya adalah membangun *Ubiquitous Language* (bahasa bersama yang dipahami oleh pengembang dan pemangku kepentingan) serta merancang model domain yang mewujudkan bahasa tersebut ke dalam bentuk struktur logis dan terukur. Melalui pendekatan ini, setiap istilah bisnis yang digunakan dalam komunikasi sehari-hari diterjemahkan langsung ke dalam model konseptual sistem, seperti *Entity, Value Object,* dan *Aggregate,* yang mencerminkan realitas proses gudang secara konsisten dan terhindar dari tumpang tindih antar domain.

Laporan ini berfokus pada perancangan *Ubiquitous Language* dan model domain untuk konteks *Inventory Control*, yang sebelumnya telah ditetapkan sebagai *core context* dalam domain WMS. Dengan mendefinisikan istilah, entitas, serta hubungan yang jelas di dalam konteks tersebut, sistem dapat memastikan integritas data stok, keakuratan pergerakan barang, dan komunikasi yang efisien antar modul lain seperti *Inbound, Outbound, Reporting,* dan *Notification*. Tahapan ini menjadi landasan penting dalam membangun sistem yang modular, terintegrasi, serta mudah dikembangkan, sekaligus menjembatani pemahaman antara model bisnis dan implementasi teknis pada tahap selanjutnya.

UBIQUITOUS LANGUAGE

A. Gambaran *Ubiquitous Language*

Ubiquitous Language dalam sistem manajemen pergudangan (*Warehouse Management System*/ WMS) berfungsi sebagai bahasa bersama yang menyatukan pemahaman antara pihak bisnis dan pengembang sistem terhadap konsep, proses, serta entitas yang ada di dalam domain pergudangan. Bahasa ini dikembangkan secara kolaboratif agar setiap istilah yang digunakan di dalam sistem memiliki makna yang konsisten, baik saat digunakan dalam percakapan bisnis, perancangan model konseptual, maupun implementasi kode program. Dengan adanya keseragaman istilah, risiko kesalahpahaman antartim dapat diminimalkan, sementara proses komunikasi, desain, dan integrasi sistem dapat berlangsung lebih efektif dan terarah.

Bahasa yang digunakan dibangun dari konsep dan istilah yang muncul secara alami dalam proses bisnis gudang. Contohnya, istilah seperti *Inventory Item, Stock Move, Reservation,* dan *Adjustment* merepresentasikan elemen utama dalam proses pengelolaan stok, sementara istilah *Location, Batch,* dan *Quantity* menggambarkan atribut pendukung yang memperkaya konteks data persediaan. Semua istilah ini disepakati melalui proses analisis dan validasi bersama antara tim pengembang dan pemangku kepentingan bisnis, sehingga dapat digunakan secara konsisten di seluruh konteks sistem, baik di dokumentasi, antarmuka pengguna, maupun struktur data internal.

Aktor-aktor utama yang berinteraksi dengan bahasa ini antara lain:

1) Staf Gudang, yang menjalankan kegiatan operasional harian seperti penerimaan, penataan, dan pengiriman barang.
2) Manajer Logistik, yang bertanggung jawab terhadap pengawasan kinerja gudang, pengelolaan kapasitas, serta analisis laporan stok.
3) Pemasok (Supplier), yang menjadi sumber pasokan barang masuk ke dalam sistem gudang.
4) Pelanggan atau Unit Pemesan, yang menjadi pihak penerima barang hasil distribusi.

Dengan diterapkannya *Ubiquitous Language*, seluruh aktivitas dan interaksi antaraktor tersebut dapat diwakilkan secara konsisten di dalam sistem, sehingga setiap data dan proses yang terjadi di gudang memiliki arti yang tunggal dan dapat ditelusuri dengan jelas. Pendekatan ini menjadikan WMS bukan hanya alat operasional, tetapi juga platform terintegrasi yang menyatukan komunikasi bisnis, arsitektur sistem, dan pengambilan keputusan berbasis data dalam satu bahasa yang sama.

B. Identifikasi Glosarium *Ubiquitous Language*

Glosarium berikut disusun untuk menyatukan pemahaman istilah yang digunakan di seluruh konteks dalam sistem *Warehouse Management System* (WMS). Setiap istilah merepresentasikan konsep bisnis atau teknis yang muncul dalam proses operasional gudang, mulai dari penerimaan barang (*inbound*), pengendalian stok (*inventory control*), pengeluaran barang (*outbound*), hingga pelaporan dan integrasi sistem eksternal. Dengan adanya glosarium ini, diharapkan komunikasi antara tim pengembang, analis bisnis, dan pemangku kepentingan dapat berjalan konsisten menggunakan bahasa domain yang seragam.

Tabel 1. Glosarium *Ubiquitous Language*

Istilah	Deskripsi / Arti dalam Konteks Sistem WMS
Purchase Order (PO)	Dokumen resmi pemesanan barang dari perusahaan kepada pemasok yang menjadi dasar penerimaan barang di gudang.
Advance Shipping Notice (ASN)	Pemberitahuan dari pemasok sebelum barang tiba, berisi rincian kuantitas, jenis, dan waktu kedatangan barang.
Receiving Schedule	Jadwal penerimaan barang di gudang yang mengatur slot waktu kedatangan truk atau kontainer dari pemasok.
Receiving Dock	Area fisik di gudang tempat barang diterima sebelum diverifikasi.
Goods Receipt	Dokumen yang mencatat penerimaan fisik barang dari pemasok beserta kondisi dan jumlahnya.
Inbound Task	Instruksi kerja untuk staf gudang dalam menangani aktivitas penerimaan barang di area inbound.
Quality Check (QC)	Proses pemeriksaan kualitas barang untuk memastikan kesesuaian dengan standar sebelum disimpan.
Put-Away	Proses pemindahan barang yang telah diterima ke lokasi penyimpanan sesuai konfigurasi gudang.
Put-Away Task	Instruksi lokasi penyimpanan spesifik untuk setiap barang hasil proses inbound.
Inbound Status	Status dari proses penerimaan barang (misalnya *Pending*, *QC*, *Completed*).
Supplier	Pihak eksternal yang memasok barang ke gudang.
Warehouse	Fasilitas penyimpanan utama yang memiliki beberapa zona dan rak penyimpanan barang.
Location	Tempat atau slot penyimpanan barang di dalam gudang (zona, rak, atau bin).
Inventory Item	Representasi stok barang berdasarkan SKU, batch, dan lokasi penyimpanan di gudang.
SKU (Stock Keeping Unit)	Kode unik untuk mengidentifikasi setiap jenis barang di sistem.

Quantity	Jumlah atau volume barang yang dinyatakan dalam satuan tertentu (pcs, box, kg, dst).
Batch / Lot	Kelompok barang dengan karakteristik sama seperti nomor produksi atau tanggal kedaluwarsa.
On-Hand Stock	Jumlah barang yang tersedia secara fisik di gudang.
Available Stock	Stok yang dapat digunakan untuk pesanan setelah dikurangi stok yang telah dipesan.
Reserved Stock	Stok yang sudah dialokasikan untuk pesanan tertentu.
Reservation	Catatan alokasi sementara stok untuk pesanan pelanggan tertentu.
Cycle Count	Penghitungan sebagian stok gudang secara berkala untuk menjaga akurasi inventori tanpa menghentikan operasi.
Stock Transfer	Pemindahan stok antar lokasi, zona, atau antar gudang yang dicatat dalam sistem.
Replenishment	Pemindahan stok dari area penyimpanan besar ke area picking agar selalu tersedia untuk pengambilan pesanan.
Adjustment	Koreksi stok akibat selisih antara data sistem dan kondisi aktual barang di gudang.
Adjustment Reason	Alasan yang mendasari perubahan stok (kerusakan, kehilangan, kesalahan input, dsb).
Stock Move	Catatan mutasi stok yang mencakup transaksi masuk, keluar, pemindahan, dan penyesuaian.
Stock Audit	Proses pencocokan stok fisik dengan catatan sistem secara berkala.
Audit Trail	Riwayat lengkap aktivitas dan perubahan data stok untuk kebutuhan kepatuhan dan pelacakan.
Order	Permintaan barang dari pelanggan atau unit internal yang diproses melalui sistem outbound.
Sales Order (SO)	Dokumen pemesanan barang dari pelanggan yang menjadi acuan pembuatan order pengiriman.
Order Line	Detail setiap jenis barang dalam satu order pelanggan.

Outbound Order	Dokumen atau entitas yang merepresentasikan permintaan pengeluaran barang dari sistem.
Outbound Fulfillment	Proses pengeluaran barang dari gudang untuk memenuhi pesanan, meliputi picking, packing, dan shipping.
Picking Task	Instruksi pengambilan barang dari lokasi penyimpanan sesuai kebutuhan pesanan.
Picking List	Daftar barang yang harus diambil untuk satu atau beberapa pesanan pelanggan.
Staging Area	Area penempatan sementara barang hasil picking sebelum pengepakan dan pengiriman.
Packing List	Daftar isi paket yang dikirim sebagai bukti dan referensi pengiriman.
Dispatch	Tahapan akhir outbound di mana barang diserahkan ke pihak pengangkut.
Carrier / Logistics Partner	Pihak ketiga yang bertanggung jawab mengirim barang ke pelanggan.
Shipment	Proses dan data pengiriman barang dari gudang ke tujuan pelanggan.
Shipping Confirmation	Konfirmasi bahwa barang telah dikirim ke pelanggan dan sedang dalam perjalanan.
Tracking Record	Riwayat pergerakan dan status barang selama perjalanan atau distribusi.
Tracking ID	Identitas unik yang digunakan untuk melacak posisi barang atau batch tertentu.
Delivery Status	Status terkini barang yang dikirim (misal: *In Transit*, *Delivered*, *Returned*).
Trace Event	Peristiwa yang merekam perpindahan atau perubahan status barang di sepanjang rantai pasok.
Goods Received	Peristiwa penerimaan barang setelah diverifikasi dan dimasukkan ke stok.
Stock Changed	Kondisi perubahan stok akibat transaksi inbound, outbound, atau adjustment.

Low Stock Alert	Peringatan otomatis ketika stok barang berada di bawah ambang batas minimum.
Threshold	Nilai ambang batas stok yang memicu peringatan *Low Stock Alert*.
Performance Report	Laporan kinerja gudang meliputi kecepatan pemrosesan, akurasi stok, dan efisiensi penggunaan ruang.
KPI Dashboard	Tampilan visual indikator kinerja utama gudang secara real-time.
Trend Analysis	Analisis pola perubahan data stok atau performa operasional dari waktu ke waktu.
Scenario Simulation	Fitur analitik untuk mensimulasikan dampak skenario tertentu terhadap operasi gudang.
Data Aggregation	Pengumpulan dan penyatuan data dari berbagai konteks untuk analisis performa.
Forecast Engine	Modul yang melakukan prediksi kebutuhan stok dan kapasitas gudang berdasarkan tren historis.
Notification Rule	Aturan yang menentukan kondisi dan penerima pesan notifikasi di dalam sistem.
Notification Message	Pesan yang dikirimkan kepada pengguna atau sistem eksternal sebagai hasil dari suatu peristiwa.
Exception Alert	Peringatan terhadap anomali atau penyimpangan operasi seperti keterlambatan pengiriman.
Event Bus	Infrastruktur komunikasi internal yang digunakan untuk menyebarkan event antar konteks secara terdistribusi.
Webhook	Mekanisme notifikasi otomatis yang mengirim data ke sistem eksternal saat event terjadi.
Retry Policy	Kebijakan pengulangan otomatis saat pengiriman notifikasi atau integrasi gagal.
Escalation Rule	Aturan peningkatan prioritas notifikasi atau pelaporan jika kejadian tidak segera ditangani.
User Account	Identitas pengguna sistem yang berisi informasi login dan hak akses.

Role & Permission	Hak akses pengguna berdasarkan peran dan tanggung jawabnya dalam organisasi.
Authentication	Proses verifikasi identitas pengguna sebelum diberikan akses ke sistem.
Authorization	Proses penentuan hak akses pengguna terhadap fitur atau data tertentu.
Session Log	Catatan aktivitas login, logout, dan penggunaan sistem oleh pengguna.
External System Adapter	Komponen yang menjembatani komunikasi antara WMS dan sistem eksternal (ERP, POS, e-commerce).
Integration Channel	Jalur integrasi dengan sistem eksternal, mencakup konfigurasi endpoint, format data, dan jadwal sinkronisasi.
Mapping Profile	Aturan konversi data internal WMS ke format eksternal selama integrasi berlangsung.
API Contract	Kesepakatan format, struktur, dan protokol data untuk komunikasi antar sistem.
Data Synchronization	Mekanisme sinkronisasi data antar sistem agar tetap konsisten.
Message Queue	Sistem antrian pesan yang menjamin pengiriman data antar konteks atau sistem eksternal secara andal.
Connector	Modul teknis yang mengatur koneksi dan pengiriman data ke sistem pihak ketiga.

PERANCANGAN MODEL *CLASS DIAGRAM*

A. Gambaran Umum *Class Diagram*

Perancangan model domain dilakukan untuk menggambarkan struktur konseptual sistem berdasarkan hasil pemetaan *bounded context* dan glosarium *Ubiquitous Languag*e yang telah disusun sebelumnya. Tahap ini berfokus pada pembentukan model yang merepresentasikan logika inti dari sistem manajemen gudang (*Warehouse Management System*), khususnya pada konteks *Inventory Control* sebagai inti domain.

Model ini tidak secara langsung menerjemahkan istilah ke bentuk teknis, tetapi membangun abstraksi domain yang mencerminkan entitas, nilai, relasi, serta aturan bisnis utama dalam sistem. Pendekatan yang digunakan mengacu pada prinsip *Domain-Driven Design* (DDD), di mana model domain berperan sebagai jembatan antara pemahaman bisnis dan rancangan implementasi perangkat lunak.

Struktur model domain terdiri dari tiga komponen utama:

1) *Entities*, yaitu objek yang memiliki identitas dan siklus hidup sendiri.
2) *Value Objects*, yaitu objek yang tidak memiliki identitas tetapi membawa nilai penting yang digunakan untuk menjaga konsistensi data.
3) *Aggregates*, yaitu unit konsistensi yang mengelompokkan entitas dan nilai dalam satu batas transaksi.

Setiap bagian model dirancang untuk mendukung komunikasi antar konteks melalui *event-driven interaction* atau antarmuka API yang telah didefinisikan dalam *context map* sebelumnya.

B. Perancangan *Class Diagram*

Berikut merupakan perancangan *class diagram* yang diidentifikasi pada *bounded context* Warehouse Operations Management dengan menganalisis arsitektur sistem berbasis *Boundary, Controller, and Entity* (BCE).

Tabel 2. *Boundary, Controller, Entity per Bounded Context*

Bounded Context	Boundary (UI / API / Gateway)	Controller / Application Service	Entity (Domain Data & Logic)
Inventory Control	InventoryOHS (Public API), InventoryUI (admin), Event consumer (Stock events)	InventoryController, StockAllocatorService, ReorderCalculator, LowStockPolicy	InventoryItem, Reservation, StockMove, Adjustment, Location, Warehouse, Threshold, SKU, Quantity, Batch

Inbound Management	InboundAPI, ReceivingDockUI, QCUI	InboundController, ReceivingService, QCService, PutAwayService	GoodsReceipt, InboundTask, QCResult, PutAwayOrder(ops), PutAwayTask, ReceivingStatus
Outbound Fulfillment	OutboundAPI, OutboundUI (Picking/Packing), StagingAreaUI	OutboundController, PickingService, PackingService, ShippingService	Shipment, PickWave(ops), PickTask, PackingList, DispatchInfo, Carrier
Order Orchestration	OrderAPI (OHS), OrderUI	OrderController, OrderWorkflow, ReservationOrchestrator	Order, OrderLine, OrderStatus
Tracking & Traceability	TrackingAPI, TrackingUI	TrackingController, TraceProjectionService	TrackingRecord, TraceEvent, TrackingId, DeliveryStatus
Reporting & Analytics	ReportingAPI, KPIDashboardUI	AnalyticsService, ETLJob, ForecastEngine	PerformanceReport, KPIView, TrendView, AggregationWindow
Notification & Alert	NotificationAPI, NotificationAdminUI, Webhook endpoint	NotificationController, Notifier, EscalationService, RetryPolicyService	NotificationRule, NotificationMessage, Channel, Recipient
Identity & Access	AuthAPI (OHS), AuthUI	AuthController, RBACService, SessionLogService	UserAccount, Role, Permission, SessionLog
External Integration	IntegrationAdminUI, WebhookGateway, ConnectorEndpoints	SyncService, EventBusGateway, MessageQueueWorker, MappingService	IntegrationChannel, MappingProfile, APIContract, Connector, MessageEnvelope

Setelah melakukan analisis arsitektur sistem berbasis *Boundary, Controller, and Entity* (BCE), disusun pula penghubung dengan model domain yang disusun menggunakan pendekatan *Domain-Driven Design* (DDD). Melalui pemetaan ini, hubungan antara elemen arsitektur aplikasi dan komponen inti domain menjadi lebih jelas serta konsisten secara konseptual. Komponen *Boundary* atau API berperan sebagai titik masuk (*entry point*) yang mengeksekusi use case sesuai kebutuhan pengguna atau sistem eksternal. Selanjutnya, *Controller* atau *Application Service* bertugas mengorkestrasi alur komando (*command orchestration*) yang mengarah pada operasi di dalam *aggregate domain*. Sementara itu, *Entity* pada model BCE direinterpretasikan ke dalam bentuk *Entity, Value Object,* atau *Aggregate Root* dalam konteks DDD, sesuai dengan batas transaksi, identitas, dan aturan konsistensi yang berlaku dalam domain bisnis. Pendekatan ini memungkinkan setiap bounded context, khususnya *Inventory Control* sebagai konteks inti, untuk dipetakan secara utuh, sehingga arsitektur sistem dan model domain dapat berjalan selaras, modular, serta mudah dikembangkan tanpa kehilangan makna bisnis yang mendasarinya.

Tabel 3. Pemetaan *Boundary, Controller, and Entity* ke *Domain-Driven Design* per *Bounded Context*

Bounded Context	**Boundary/API → Use Case**	**Controller/Service → Orkestrasi Command**	**Entity (UML) → DDD Mapping**	**Event Domain (keluar/masuk)**
Inventory Control	InventoryOHS.reserve() / getAvailability()	InventoryController.reserve(orderId, sku, qty) + StockAllocatorService	**Aggregate Root:** InventoryItem **Entity:** Reservation, StockMove, Adjustment **Value Object:** SKU, Quantity, Batch, Threshold, LocationId	**Keluar:** StockReserved, StockReleased, StockChanged, LowStockAlert; **Masuk:** GoodsReceived, ShippingConfirmation

Inbound Management	InboundAPI.postGoodsReceipt() / submitQC()	InboundController + ReceivingService, QCService, PutAwayService	**Aggregate Root:** GoodsReceipt **Entity:** InboundTask, QCResult, PutAwayTask **Value Object:** ReceivingStatus	**Keluar:** GoodsReceived **Masuk:** (ops) ASNReceived
Outbound Fulfillment	OutboundAPI.createShipment() / confirmDispatch()	OutboundController + PickingService, PackingService, ShippingService	**Aggregate Root:** Shipment **Entity:** PickWave(ops), PickTask, PackingList **Value Object:** DispatchInfo	**Keluar:** ShippingConfirmation; **Masuk:** StockReserved
Order Orchestration	OrderAPI.placeOrder() / cancelOrder()	OrderController + OrderWorkflow/Orchestrator	**Aggregate Root:** Order **Entity:** OrderLine **Value Object:** OrderStatus	**Keluar:** OrderPlaced, OrderCancelled **Masuk:** StockReserved, ShippingConfirmation

Tracking & Traceability	TrackingAPI.getTrace(sku/batch)	TrackingController + TraceProjectionService	**Entity:** TrackingRecord, TraceEvent **Value Object:** TrackingId, DeliveryStatus	**Keluar:** TraceCorrected **Masuk:** StockChanged, ShippingConfirmation
Reporting & Analytics	ReportingAPI.getKPI() / getTrends()	AnalyticsService, ETLJob, ForecastEngine	**Entity:** PerformanceReport, KPIView, TrendView **Value Object:** AggregationWindow	**Keluar:** (ops) ReportPublished **Masuk:** StockChanged, GoodsReceived, ShippingConfirmation
Notification & Alert	NotificationAdminAPI.saveRule()	NotificationController, Notifier, EscalationService	**Aggregate Root:** NotificationRule **Entity:** NotificationMessage **Value Object:** Channel, Recipient, RetryPolicy	**Keluar:** ExceptionAlert, LowStockDelivered **Masuk:** LowStockAlert, ExceptionDetected
Identity & Access	AuthAPI.login()/assignRole()	AuthController, RBACService, SessionLogService	**Aggregate Root:** UserAccount	**Keluar:** UserLoggedIn, RoleAssigned

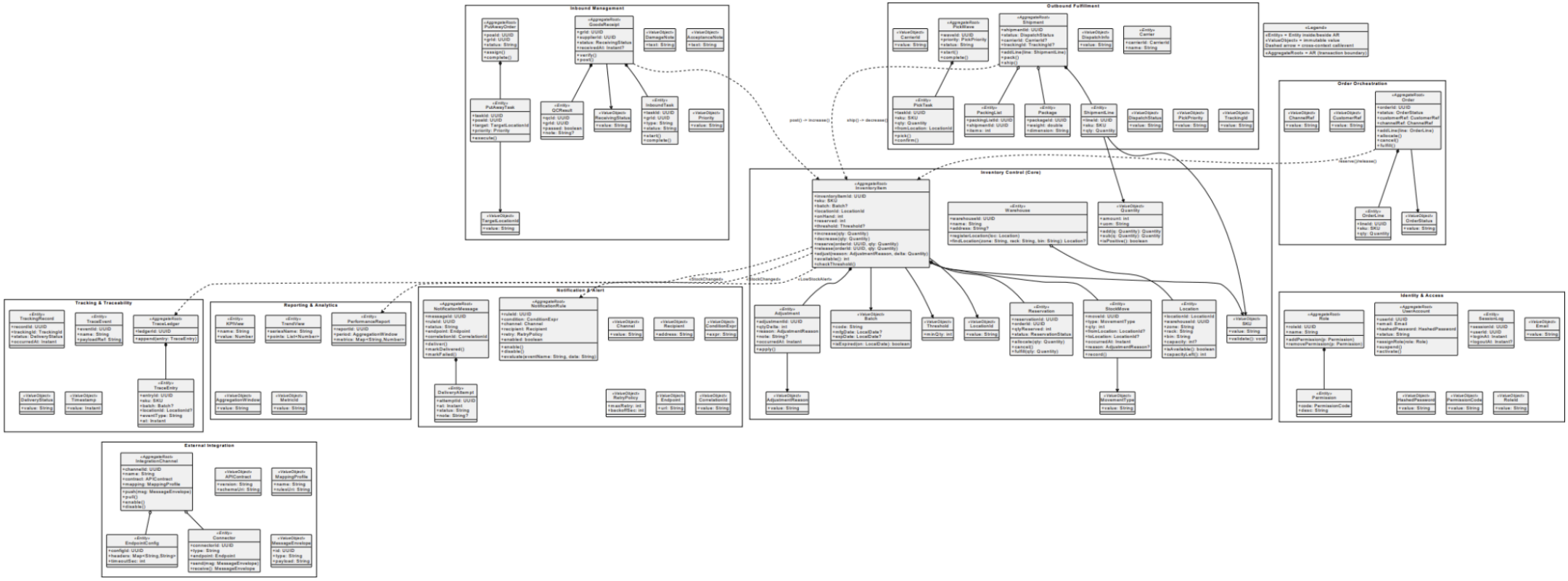
			Entity: Role, Permission, SessionLog	**Masuk:** —
External Integration	IntegrationAdminAPI.createChannel() / webhook endpoints	SyncService, EventBusGateway, MappingService	**Aggregate Root:** IntegrationChannel **Entity:** MappingProfile, APIContract, Connector **Value Object:** MessageEnvelope	**Keluar:** SyncPushed, WebhookCalled **Masuk:** ERPOrderImported, ASNReceived

Setelah dilakukan pemetaan antara komponen *Boundary–Controller–Entity* (BCE) dengan struktur model domain (DDD) pada masing-masing konteks, langkah berikutnya adalah mengidentifikasi elemen utama pembentuk domain yang terdiri atas *Aggregate Root, Entity,* dan *Value Object.* Tabel berikut merangkum ketiga elemen tersebut pada seluruh bounded context yang telah didefinisikan. Pemetaan ini menunjukkan bagaimana setiap konteks memiliki batas tanggung jawab logika bisnisnya masing-masing, sekaligus memperlihatkan bahwa konteks *Inventory Control* berperan sebagai inti domain (*core context*) yang menjadi sumber kebenaran data dan pusat konsistensi sistem, sementara konteks lainnya berfungsi mendukung proses bisnis operasional, pelaporan, serta integrasi eksternal.

Tabel 4. Model Domain (DDD): *Aggregate Root, Entity,* dan *Value Object* per *Bounded Context*

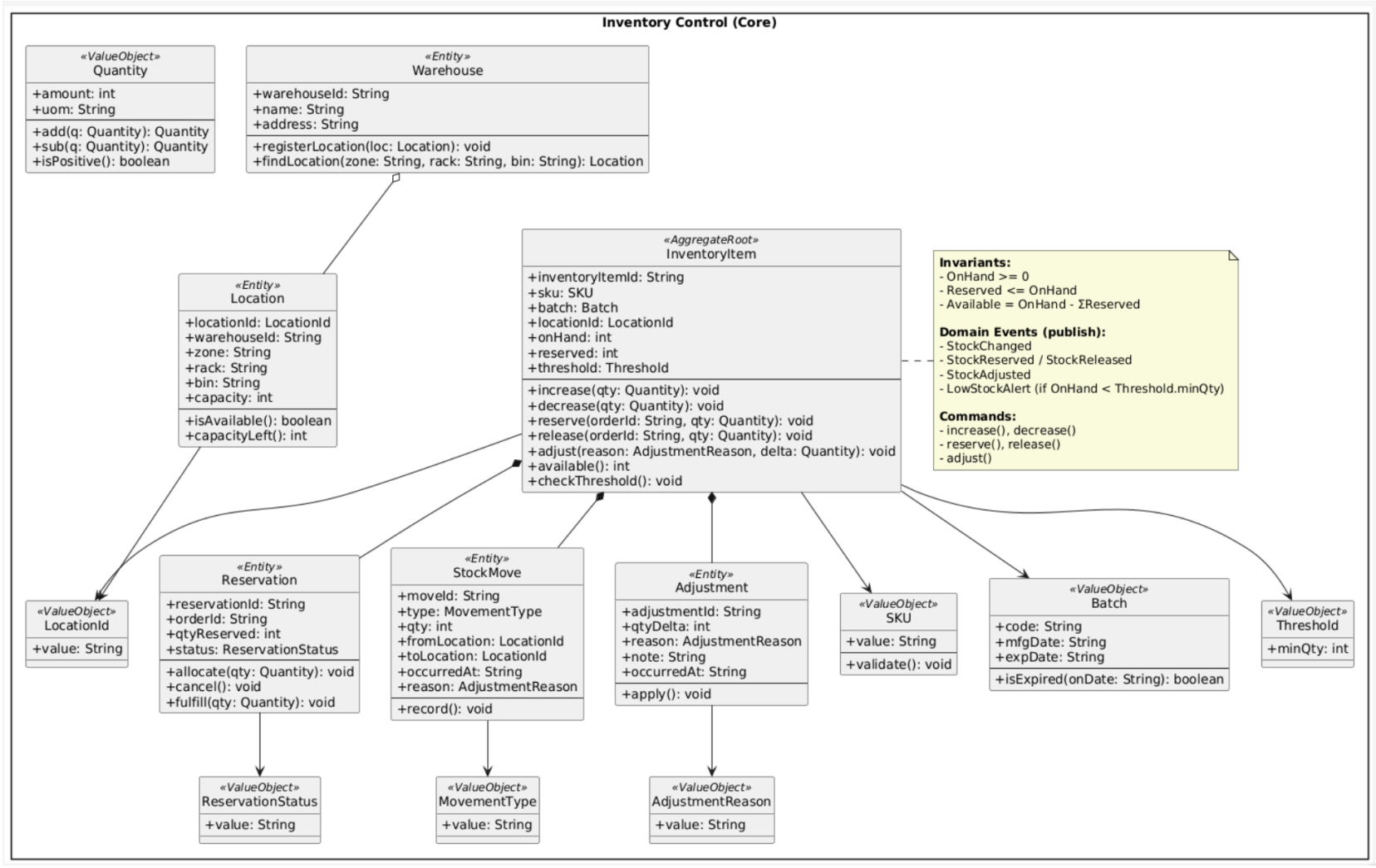
Bounded Context	Aggregate Root	Entity	Value Object
Inventory Control (Core)	InventoryItem	Reservation, StockMove, Adjustment, Location, Warehouse	SKU, Quantity, Batch, Threshold, LocationId, MovementType

Inbound Management	GoodsReceipt, PutAwayOrder	InboundTask, QCResult, PutAwayTask	ReceivingStatus, DamageNote, AcceptanceNote, TargetLocationId, Priority
Outbound Fulfillment	Shipment, PickWave / PickList	ShipmentLine, PickTask, PackingList, Package, DispatchInfo, Carrier	TrackingId, DispatchStatus, PickPriority, CarrierId
Order Orchestration	Order	OrderLine	OrderStatus, CustomerRef, ChannelRef
Tracking & Traceability	TraceLedger	TrackingRecord, TraceEntry, TraceEvent	TrackingId, DeliveryStatus, Timestamp
Reporting & Analytics	*(read-model, tanpa AR)*	PerformanceReport, KPIView, TrendView	AggregationWindow, MetricId
Notification & Alert	NotificationRule, NotificationMessage	DeliveryAttempt	Channel, Recipient, ConditionExpr, RetryPolicy, Endpoint, CorrelationId
Identity & Access	UserAccount, Role	SessionLog, Permission	Credential, Email, HashedPassword, RoleId, PermissionCode
External Integration	IntegrationChannel	Connector, EndpointConfig	APIContract, MappingProfile, MessageEnvelope



Gambar 1. *Class Diagram Wirehouse Management System*

Untuk diagram yang lebih jelas dapat dilihat pada link berikut Class Diagram WMS.pdf .



Gambar 2. *Class Diagram Core Context Inventory Control* pada *Wirehouse Management System*