

Tugas Kelompok ke-1

Week 3

Team 1:

1. Wulan Aprianingsih – 2502126554
2. Muhammad Rizky Fadli – 2502121175
3. Michael Abraham Chan Tulenan - 2502127172
4. Frans Sebastian - 2502121162
- 5.

Soal Teori

1. (LO1; 10%) Jelaskan pendapat anda, mengapa pendekatan DDD ditujukan untuk aplikasi yang kompleks? Apa yang dimaksud kompleksitas pada aplikasi? Berikan contoh!

Jawab:

Pendekatan DDD (Domain-Driven Design) ditujukan untuk aplikasi yang kompleks karena memfokuskan pada pemahaman yang mendalam tentang domain bisnis yang kompleks tersebut. Kompleksitas pada aplikasi dapat merujuk pada berbagai faktor, termasuk:

- a. **Keterlibatan Domain Bisnis yang Tinggi:** Aplikasi kompleks sering kali terlibat dalam domain bisnis yang kompleks pula, yang memerlukan pemahaman yang mendalam tentang proses bisnis, aturan, dan hubungan antar entitas.
- b. **Kebutuhan akan Model yang Kaya:** Aplikasi kompleks memerlukan model yang kaya dan eksplisit untuk merepresentasikan domain bisnis dengan benar. DDD menyediakan alat dan konsep yang kuat untuk memodelkan domain secara efektif.

- c. **Perubahan yang Sering Terjadi:** Aplikasi kompleks sering kali menghadapi perubahan dalam persyaratan bisnis, dan DDD menawarkan pendekatan yang memungkinkan fleksibilitas dan adaptasi terhadap perubahan tersebut.
- d. **Keterlibatan Tim yang Beragam:** Tim pengembangan yang terlibat dalam proyek aplikasi kompleks biasanya terdiri dari berbagai peran dan keahlian, dan DDD memberikan kerangka kerja yang memungkinkan komunikasi dan pemahaman yang lebih baik antar anggota tim.

Contoh dari aplikasi kompleks yang dapat dipecahkan dengan pendekatan DDD adalah sistem manajemen rantai pasokan (supply chain management). Sistem ini melibatkan berbagai entitas seperti produsen, distributor, pengecer, dan pelanggan, dengan berbagai proses, aturan, dan hubungan antar entitas. Dengan menerapkan prinsip-prinsip DDD, pengembang dapat memahami secara mendalam domain bisnis ini dan merancang solusi yang memenuhi kebutuhan kompleksitas yang ada.

2. (LO1; 10%) Jelaskan bagaimana DDD dapat menyelesaikan pengembangan aplikasi yang kompleks.

Jawab:

Pola, praktik strategis dan taktis DDD menyelaraskan desain perangkat lunak dengan domain bisnisnya. DDD dapat mengidentifikasi subdomain organisasi, dimana jenis subdomain yang berbeda memiliki tingkat kerumitan yang berbeda dan penting karena untuk merancang solusi perangkat lunak yang baik.

- DDD menyelaraskan desain perangkat lunak dengan domain bisnisnya, memastikan bahwa struktur dan logika perangkat lunak mencerminkan struktur dan logika dalam domain bisnis.
- DDD membantu menyaring masalah domain yang kompleks dan membentuk arsitektur aplikasi secara holistik sebelum menyelam ke dalam detail implementasi. Ini mirip dengan menggunakan sapuan kuas untuk merencanakan dan menata fondasi sebelum menambahkan detail yang lebih halus.

- Salah satu pola penting dalam DDD adalah pola Agregat, yang membantu dalam menciptakan model yang efektif untuk Bounded Context yang kompleks. Agregat memungkinkan untuk mengelompokkan entitas terkait dan mengatur batasan transaksi serta konsistensi data di dalam konteks yang terbatas.
- DDD mengadopsi pendekatan yang sangat eksplisit dalam pemodelan domain dengan menggunakan peristiwa domain. Dengan menerapkan pola ini, DDD memungkinkan pemodelan yang lebih ketat dan komprehensif, yang memfasilitasi komunikasi antar anggota tim dan pemahaman yang lebih baik tentang domain bisnis.

3. (LO1; 10%) Untuk memudahkan pengembangan dalam DDD kita memecah Domain ke dalam Subdomain. Prioritas pengerjaan proyek dapat dilihat berdasar jenis Subdomain tersebut. Jelaskan jenis – jenis Subdomain dan berikan contoh.

Jawab:

○ **Core Subdomain**

Merupakan bagian inti dari sistem yang mewakili keunggulan kompetiti mendasar yang dapat diperoleh perusahaan melalui pengiriman perangkat lunak.

Contoh:

Penjual baju di toko online. Toko online itu penting, tetapi bukan core subdomain melainkan kualitas bajunya. Kualitas baju atau bahan yang digunakan adalah alasan pelanggan membeli produknya dan mengingat mereknya walau banyak penjual baju dengan model yang sama.

○ **Supporting Subdomain**

Jenis subdomain yang mendukung bisnis perusahaan tapi tidak memberikan keunggulan kompetitif apapun.

Contoh:

Pengguna Zalora yang dapat memungkinkan pelanggan menelusuri katalog produk beserta detail produk. Fungsionalitas penelusuran produk Zalora tidak mendefinisikannya sebagai perusahaan, tetapi mendukung penggunaan.

○ **Generic Subdomain**

Subdomain yang umumnya rumit dan sulit diterapkan tapi tidak memberikan keunggulan kompetitif bagi perusahaan.

Contoh:

Website yang memerlukan autentikasi pengguna untuk meningkatkan keamanan.

4. (LO1; 10%) Jelaskan perbedaan antara Subdomain, Bounded Context dan Ubiquitous Language. Berikan contoh!

Jawab:

- **Subdomain** adalah subset konsep yang tidak saling eksklusif dan sewenang-wenang di alam semesta.
 - **Ubiquitous Language** adalah bahasa yang berasal dari model dan shared team language. Setiap subdomain memiliki *Ubiquitous Language* yang relevan dan spesifik dengan konteksnya, tetapi secara khusus menargetkan Bounded Context.
 - Sedangkan, **Bounded Context** adalah batasan suatu model yang merepresentasikan konsep-konsep tersebut, hubungannya, dan aturan-aturannya. Subdomain yang sama dapat diwakili oleh pilihan pemodelan yang jumlahnya tidak terbatas.
5. (LO1; 10%) Bounded Context dapat saling berkomunikasi dan berintegrasi yang dikenal dengan istilah Context Mapping. Jelaskan tiga pendekatan integrasi Bounded Context beserta kelebihan dan kekurangannya.

Berikut beberapa pendekatan integrasi Bounded Context :

- **Partnership**

Pendekatan ini melibatkan kerja sama erat antara dua Bounded Context, di mana kedua tim bekerja bersama untuk menyelaraskan model dan integrasi mereka.

Kelebihan: Memungkinkan kerjasama yang erat dan pemahaman mendalam tentang kedua konteks.

Kekurangan: Memerlukan komunikasi yang intensif dan sering kali memerlukan kompromi dari kedua belah pihak.

- **Shared Kernel**

Pendekatan ini menggunakan model domain bersama yang dibagikan oleh dua atau lebih Bounded Context.

Kelebihan: Mengurangi duplikasi dan memastikan keseragaman antara Bounded Context yang berbeda.

Kekurangan: Dapat menciptakan ketergantungan yang tinggi dan mengurangi fleksibilitas dalam pengembangan.

- **Customer/Supplier**

Salah satu Bounded Context bertindak sebagai 'Customer' yang menentukan persyaratan, dan Bounded Context lainnya sebagai 'Supplier' yang memenuhi persyaratan tersebut.

Kelebihan: Memungkinkan spesialisasi dan fokus pada kebutuhan pelanggan.

Kekurangan: Dapat menciptakan hubungan yang tidak seimbang di mana 'Supplier' sangat bergantung pada 'Customer' untuk arahan.

- **Conformist:**

Dalam pendekatan ini, satu Bounded Context (konsumen) secara pasif menyesuaikan diri dengan model yang disediakan oleh Bounded Context lain (pemasok), tanpa mencoba mempengaruhi desainnya.

Kelebihan: Mengurangi kompleksitas integrasi karena konsumen menerima model pemasok apa adanya.

Kekurangan: Konsumen mungkin terpaksa menerima model yang tidak optimal untuk kebutuhan mereka sendiri.

- **Anticorruption Layer (ACL)**

Pendekatan ini melibatkan pembuatan lapisan antarmuka atau fasad yang menerjemahkan panggilan antara dua Bounded Context yang berbeda, sehingga menghindari kontaminasi model satu sama lain.

Kelebihan: Memungkinkan Bounded Context untuk berkomunikasi tanpa harus mengorbankan otonomi atau desain internal mereka.

Kekurangan: Memerlukan usaha tambahan untuk memelihara lapisan tambahan ini dan dapat menambah kompleksitas sistem.

- **Open Host Service (OHS)**

Pendekatan ini menggunakan Bounded Context yang menyediakan API atau layanan yang dirancang untuk digunakan oleh Bounded Context lain.

Kelebihan: Memfasilitasi integrasi dengan menyediakan antarmuka yang jelas dan stabil.

Kekurangan: Pemasok layanan harus memastikan bahwa layanan mereka cukup umum untuk memenuhi kebutuhan berbagai konsumen.

- **Separate Ways**

Dalam pendekatan ini, Bounded Context memilih untuk tidak berintegrasi sama sekali, mengakui bahwa integrasi tidak memberikan nilai yang cukup untuk usaha yang diperlukan.

Kelebihan: Menghindari kompleksitas dan biaya integrasi.

Kekurangan: Mungkin melewatkan peluang untuk sinergi atau efisiensi yang bisa didapat dari integrasi.

Referensi:

- Evans, Eric. (2003). Domain-Driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley.
- Vernon, Vaughn. (2013). Implementing Domain-Driven Design. Addison-Wesley.

Soal Kasus

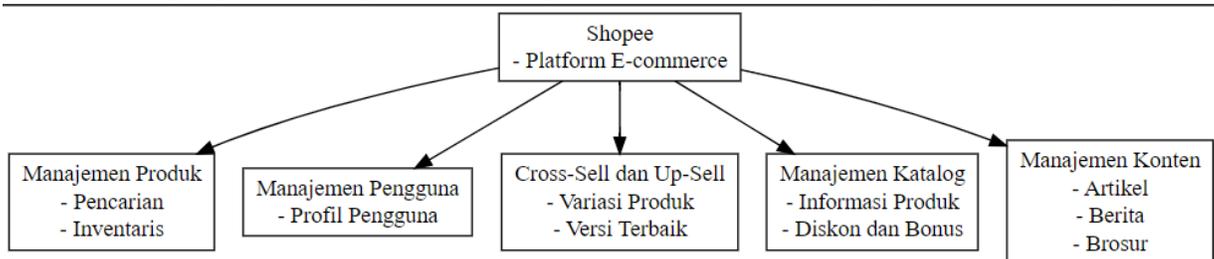
Pilihlah salah satu e-commerce yaitu Tokopedia, Shopee dan Lazada. Diasumsikan anda telah bekerjasama dengan domain expert. Lakukanlah analisa dan jelaskan mengenai :

- a. Apa yang menjadi kompleksitas dari e-commerce yang anda pilih.

Berikut kompleksitas dari shopee :

- Model Bisnis: Shopee memiliki pendekatan bisnis yang difokuskan pada pengiriman gratis dan diskon menarik, membutuhkan manajemen logistik yang kompleks dan efisien untuk mempertahankan keberlanjutan dan profitabilitasnya.
- Kolaborasi: Shopee melakukan kerjasama dengan berbagai merek terkenal, influencer, dan selebritas, yang membutuhkan strategi pemasaran dan manajemen hubungan yang rumit untuk memperoleh dampak maksimal dari kemitraan tersebut.
- Fitur Interaktif: Fitur seperti Shopee Live meningkatkan interaksi dengan pengguna, memerlukan infrastruktur teknologi yang andal dan tim yang tanggap untuk mengelola interaksi real-time serta memberikan pengalaman belanja yang personal.
- Pesaing: Persaingan ketat dengan platform e-commerce lain seperti Lazada dan Tokopedia mendorong Shopee untuk terus berinovasi dan meningkatkan pengalaman pengguna guna tetap relevan di pasar yang kompetitif.
- Keamanan dan Keaslian Produk: Shopee dihadapkan pada tantangan memastikan keamanan dan keaslian produk yang dijual di platformnya, membutuhkan sistem verifikasi dan pengawasan yang ketat untuk melindungi konsumen dari barang palsu atau tidak sesuai deskripsi.
- Sistem Informasi: Infrastruktur sistem informasi yang kuat mendukung peran Shopee sebagai pemimpin e-commerce dan pasar seluler di Asia Tenggara, termasuk manajemen data besar dan transaksi yang aman.
- Dukungan Pelanggan: E-commerce seperti Shopee membutuhkan dukungan pelanggan yang efektif dan responsif untuk menangani pertanyaan dan masalah yang mungkin timbul dari jutaan pengguna mereka.
- Metode Pembayaran: Penyediaan berbagai opsi pembayaran yang mudah digunakan oleh pelanggan juga merupakan aspek kompleks, memerlukan integrasi dengan berbagai penyedia layanan pembayaran serta kepatuhan terhadap regulasi keuangan yang berlaku.

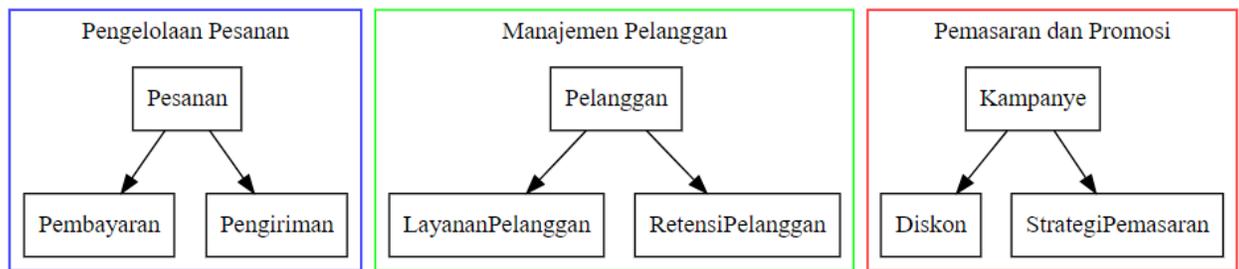
- Logistik Pengiriman: Ketersediaan opsi pengiriman yang beragam dan efisien merupakan faktor penting lainnya, membutuhkan koordinasi yang baik dengan berbagai pihak logistik untuk memenuhi kebutuhan pengguna di berbagai wilayah.
- b. Jelaskan Subdomain dan jenis – jenisnya dari e-commerce yang anda pilih. Beri penjelasan atas pemilihan jenis – jenis Subdomain tersebut. Anda dapat menambahkan gambar desain Subdomain untuk memperkuat jawaban.
 - Manajemen Produk (Product Management): Ini mencakup fitur dan layanan untuk mengelola produk yang dijual secara online. Di Shopee, fitur pencarian membantu pembeli dalam menemukan produk yang mereka cari, sementara penjual dapat dengan mudah mengelola stok mereka.
 - Manajemen Pengguna (User Management): Ini terkait dengan fitur untuk mengelola profil dan aktivitas pengguna di dalam aplikasi. Sebagai contoh, di Shopee terdapat fitur 'Saya' yang menyajikan profil pengguna.
 - Cross-Sell dan Up-Sell: Layanan ini menitikberatkan pada penawaran produk tambahan (cross-sell) dan peningkatan penjualan melalui versi terbaik produk (up-sell), membantu penjual meningkatkan penjualan dan memungkinkan pembeli menemukan produk terbaik.
 - Manajemen Katalog (Catalog Management): Layanan ini memfasilitasi penjual dalam mengatur katalog barang yang dijual secara online, memberikan informasi dan perbandingan produk seperti diskon dan bonus.
 - Manajemen Konten (Content Management): Ini terkait dengan pengaturan isi konten dalam aplikasi e-commerce, termasuk artikel, berita, dan brosur katalog online, yang membantu penjual mempromosikan produk terbaru dan penawaran lainnya.



- c. Jelaskan Bounded Context dan Ubiquitous Language yang ada. Beri penjelasan atas jawaban anda tersebut. Anda dapat menambahkan gambar desain Bounded Context untuk memperkuat jawaban.

Setiap Bounded Context memiliki model, terminologi, dan aturan bisnis yang spesifik, memungkinkan tim untuk fokus pada area mereka tanpa terbebani oleh kompleksitas keseluruhan domain e-commerce. Berikut beberapa Bounded Context pada aplikasi shopee :

- Manajemen Pesanan (Order Management): Ini mencakup semua proses yang terkait dengan penanganan pesanan, mulai dari penerimaan pesanan hingga pengiriman ke pelanggan, termasuk pemilihan produk, pembayaran, dan logistik pengiriman.
- Manajemen Inventaris (Inventory Management): Fokus pada pengelolaan stok barang, termasuk pembaruan stok, penyimpanan, dan pengawasan atas ketersediaan barang.
- Pengolahan Pembayaran (Payment Processing): Di sini, semua aktivitas yang berkaitan dengan transaksi pembayaran dikelola, termasuk verifikasi pembayaran, penanganan metode pembayaran yang berbeda, dan keamanan transaksi.



Bounded Contexts di Shopee

Lalu Ubiquitous Language membantu memastikan bahwa semua pengguna platform, dari pembeli hingga penjual, memiliki pemahaman yang sama tentang proses dan fitur yang tersedia di Shopee. Berikut beberapa contoh Ubiquitous Language pada shopee :

- Keranjang Belanja (Shopping Cart): Digunakan oleh pengguna ketika mereka menambahkan produk ke daftar belanjaan mereka sebelum melakukan pembayaran.
- Flash Sale: Event penjualan dengan waktu terbatas yang menawarkan diskon besar pada produk tertentu, sering digunakan untuk menarik perhatian dan mendorong pembelian cepat.
- Gratis Ongkir (Free Shipping): Frasa yang menunjukkan bahwa pengiriman produk tidak dikenakan biaya tambahan, sering menjadi faktor penting dalam keputusan pembelian.
- ShopeePay: Metode pembayaran digital milik Shopee yang memudahkan transaksi dan menawarkan berbagai promo.

- Shopee Mall: Subdomain khusus untuk produk-produk yang dijamin 100% asli dan berasal dari brand-brand resmi.
- Shopee Coins: Mata uang virtual yang dapat dikumpulkan dan digunakan untuk mendapatkan diskon pada pembelian selanjutnya.
- Pre-Order: Sistem pemesanan untuk produk yang belum dirilis ke pasar, memungkinkan pengguna untuk memesan lebih awal.
- Resi Otomatis (Automatic Receipt): Sistem yang secara otomatis mengupdate informasi nomor resi pengiriman setelah penjual mengirim barang.
- Shopee Guarantee: Jaminan yang diberikan oleh Shopee untuk melindungi pembeli dari transaksi yang tidak aman.