1. **Case (100%)**

PT Hwaa adalah perusahaan multinasional di bidang entertainment. PT Hwaa memiliki banyak divisi yang berfokus pada bidang-bidang hiburan yang berbeda. Berkat rekam jejak anda yang telah sukses menyelesaikan proyek sebelumnya dari PT Hwaa, anda sekarang diberikan kepercayaan untuk men-develop sistem baru untuk divisi produksi musik. Sistem ini akan digunakan oleh manager untuk mengelola konten musik dan artis milik perusahaan. Adapun fitur-fitur dari sistem tersebut secara detail adalah sebagai berikut.

1. **Music Resource Management**

Fitur ini berguna untuk mengelola koleksi musik yang dimiliki oleh perusahaan. Data musik yang telah disimpan dapat mengalami update apabila terjadi perubahan copyright ataupun evaluasi nilai jual. Sistem anda perlu mengantisipasi keperluan tersebut.

1. **Artist Management**

Fitur ini berguna untuk mengelola daftar artis yang memiliki kontrak dengan perusahaan. Setiap artis memiliki manager yang bertugas mengurus kegiatan sehari-hari mereka, karena itu proses assignment manager kepada artis juga perlu diatur dalam fitur ini. Sesuai aturan perusahaan, pastikan seorang manager tidak boleh mengurus lebih dari 3 artis dalam waktu bersamaan.

Sama seperti data musik, data artis juga dapat mengalami update, misalnya jika artis memutus kontrak dengan perusahaan. Siapkan sistem anda untuk menangani keadaan tersebut.

1. **Music Purchasing System**

Fitur ini digunakan untuk memproses pembelian musik oleh pihak ketiga, misalnya untuk digunakan dalam produksi film atau serial televisi. Pembelian musik dapat dilakukan dalam bentuk kontrak, dimana hak penggunaan musik diberikan dalam waktu sementara, ataupun kepada konsumen, dimana musik dijual kepada individu dan hak penggunaannya hanya untuk non-komersil.

1. **History Tracing**

Fitur ini digunakan untuk keperluan security. Sistem anda diharapkan dapat menyimpan history transaksi-transaksi yang terjadi dalam keseharian perusahaan secara lengkap sehingga perusahaan dapat memiliki bukti kuat apabila terjadi sengketa copyright ataupun kontrak. Pilih konsep Domain Driven Design yang tepat untuk menyiapkan fitur ini.

**Anda bebas menambahkan detail tambahan diluar soal** apabila diperlukan dalam jawaban anda, **tetapi anda tidak boleh menghilangkan detail yang sudah dijabarkan di soal di atas**.

**Jadi, Detail Tambahan saya:**

1. **User Types**

Aplikasi akan membagi tipe pengguna menjadi 4:

* Penonton, yaitu user yang mengomsumsi materi dengan mendengarkan musik secara langsung di website atau membeli lisensi.
* Artis, yaitu user yang memberikan materi musik untuk didengarkan pribadi atau untuk keperluan bisnis.
* Manager Artis, yaitu user yang mengelola kegiatan Artis sehari-hari seperti jadwal pembuatan music, tanda tangan kontrak bisnis, dan lainnya.
* Perusahaan/korporat, yaitu pihak ketiga yang ingin membeli music misalnya untuk digunakan dalam produksi film atau serial televisi dalam bentuk kontrak digital.

Penonton dan Manager Artis bergabung ke aplikasi melalui proses registrasi, sedangkan Artis dan Perusahaan adalah account yang dipegang oleh pihak client dan tidak dibuka ke publik.

1. **Registrasi**

* Saat registrasi, pengguna memasukkan data diri, username, dan password yang diinginkan.
* Selain itu, pengguna perlu memilih tipe account antara Penonton atau Manager Artis. Pengguna boleh memiliki lebih dari satu account, tapi setiap account hanya bisa memiliki satu tipe.
* Pengguna juga memiliki pilihan untuk menghubungkan account mereka dengan authentication pihak ketiga seperti Google dan Facebook agar dapat login menggunakan account pihak ketiga tersebut.

1. **Home Page**

* Login yang sukses akan mengirim pengguna ke home page dan menampilkan nama artis, judul musik, negara asal artis, Berbayar atau tidak, dan layout musik terbaru.
* Selain itu, home page juga menampilkan harga lisensi musik, genre musik, dan berapa lama lisensi bertahan.
* Terdapat Dokumentasi musik yang sudah berlalu seperti musik spesial untuk hari tertentu atau event tertentu saja.

1. **Profile Page**

* Profile page milik Penonton akan menampilkan data diri publik pengguna dan daftar musik yang dibeli, sisa berapa lama lisensi setiap music berlaku dan pernah didengarkan oleh pengguna tersebut.
* Profile page milik Artis akan menampilkan data diri publik pengguna, link ke sosial media yang aktif, dan juga daftar musik dan archived musik yang dimiliki oleh pengguna.
* Melalui halaman ini, Penonton dapat subscribe ke Artis tersebut untuk mendapatkan notifikasi saat Artis upload music terbaru ataupun mengupload suatu berita baru.
* Selain itu, Penonton juga dapat memberikan “Donasi” melalui halaman ini untuk mendukung Artis mereka dan dana sepenuhnya masuk ke rekening Artis bersangkutan.

1. **Streaming Musik**

* Streaming adalah fitur aplikasi ini dimana Penonton dapat mendegarkan music secara online di website menggunakan link URL khusus (Token Auth berbeda setiap user dan berubah setiap waktu) yang dibuat system untuk melindungi hak cipta dari pencipta musik.
* Selama streaming musik, Penonton dapat diskusi dan berkomentar di kolom komentar setiap musik dan memberikan Donasi ke Artis menggunakan virtual currency melalui kolom chat.

1. **Previous Music Events Documentation or Next Events Trailer**

* Selain streaming, Artis juga dapat mengupload dokumentasi musik sebelumnya yang pernah dilakukan maupun gambaran untuk musik selanjutnya melalui profile page.
* Penonton yang menonton dapat menaruh comment yang dapat dibaca dan di-reply oleh Penonton lain.

1. **Virtual Shop**

* Virtual shop adalah bagian aplikasi dimana Penonton dapat membeli virtual currency menggunakan mata uang asli.
* Virtual currency ini akan digunakan oleh Penonton untuk membeli lisensi musik yang berbayar atau membeli kontrak penjualan suatu music untuk digunakan secara komersil atau memberikan Donasi ke Artis yang mereka sukai baik saat mengikuti streaming ataupun melalui profile page Artis tersebut.
* Dari pihak Artis, virtual shop digunakan untuk menukarkan virtual currency yang mereka dapatkan menjadi mata uang asli sebagai sumber pendapatan sampingan mereka.

Untuk saat ini, proses pertukaran virtual currency menjadi uang asli dan sebaliknya akan diarahkan ke aplikasi third party yaitu Paypal.

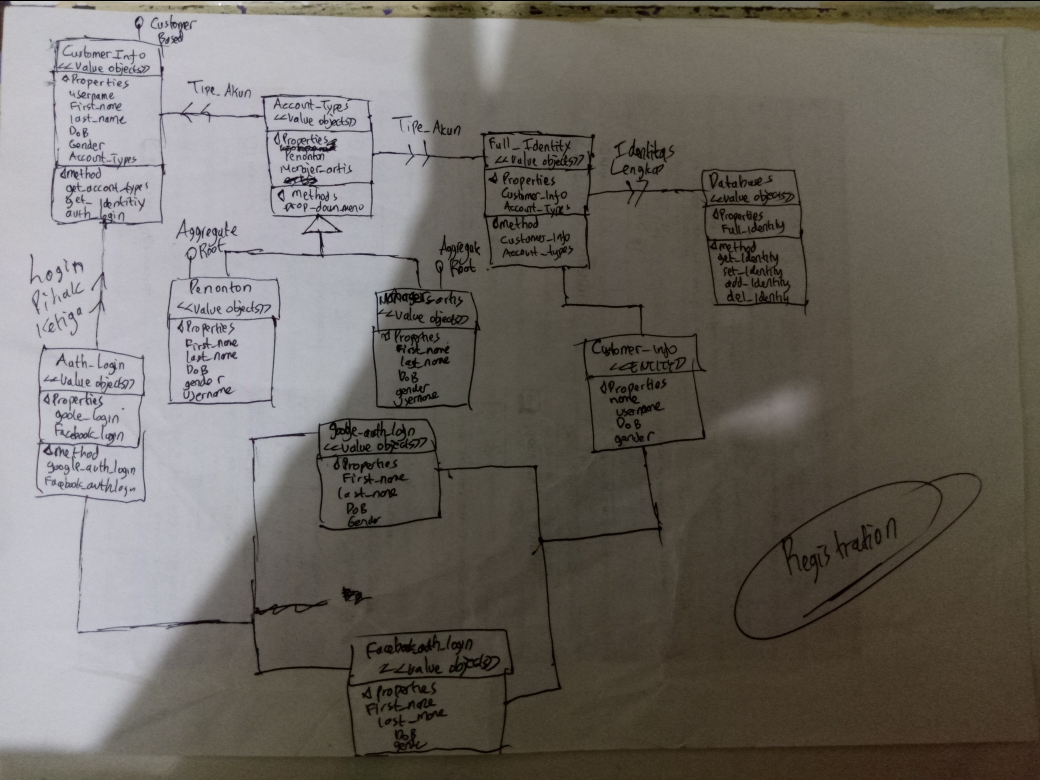
**Soal:**

Anda diminta membuat model (boleh dalam bentuk diagram atau source code) untuk design keseluruhan sistem anda. Adapun model yang anda buat perlu menggunakan konsep-konsep Domain Driven Design untuk menjaga kualitas sistem anda. Model yang anda buat perlu mengandung komponen-komponen berikut:

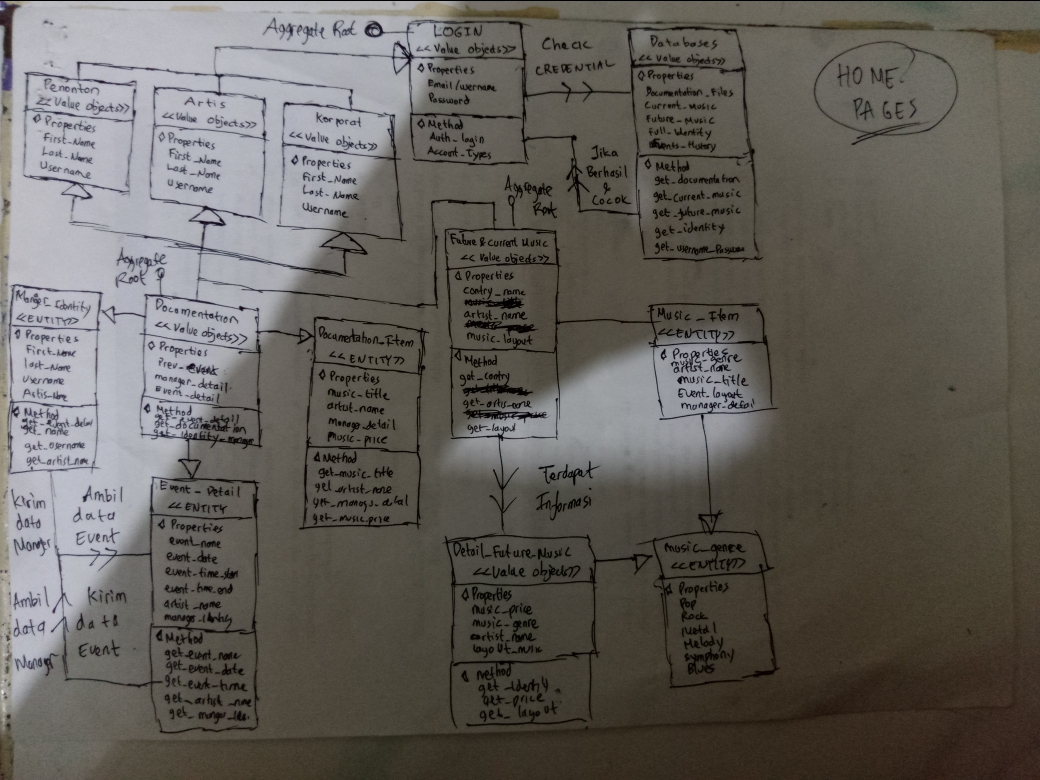
1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 20 points]** **Entities dan Value Object dengan konsep Aggregate**.

**Jawaban:**

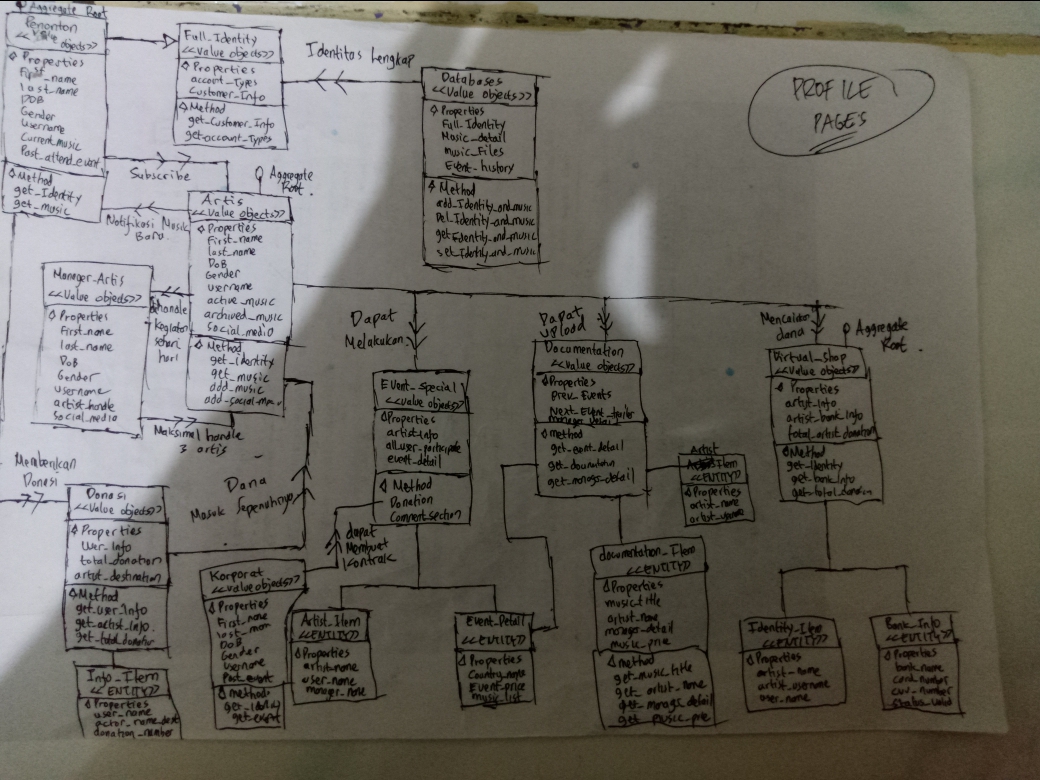
1. **Registration**



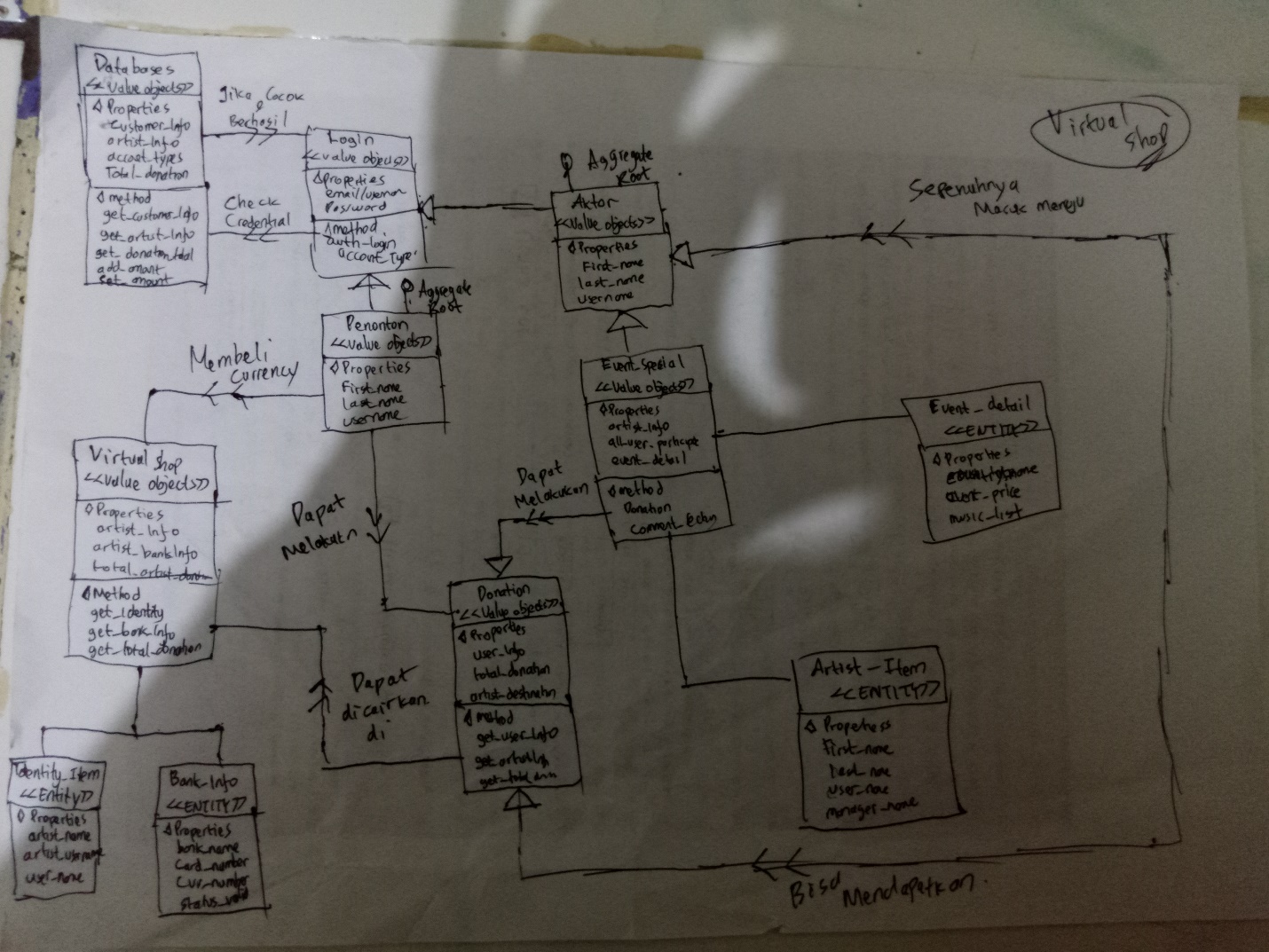
1. **Home Pages**



1. **Profile Pages**



1. **Virtual Shop**



1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 15 points]** **Domain Services**.

**Jawaban:**

d

1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 15 points]** **Factories**.

**Jawaban:**

d

1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 15 points]** **Repositories**.

**Jawaban:**

d

1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 15 points]** **Domain Events**.

**Jawaban:**

d

1. **[LO 1, LO 2, LO 3, LO 4, & LO 5, 20 points]** Anda juga diminta menggunakan **Event Sourcing untuk memastikan tidak ada data history yang hilang**. **Berikan contoh code snippet dari salah satu Entity dalam design sistem anda** yang menggambarkan cara kerja Event Sourcing!.

**Jawaban:**

**Snippet Code (Customer Info di Register)**

**BaseAggregateRoot.cs**

**public** **interface** IAggregateRoot**<out** TKey**>** **:** IEntity**<**TKey**>{**

**public** long Version **{** **get;** **}**

IReadOnlyCollection**<**IDomainEvent**<**TKey**>>** Events **{** **get;** **}**

void ClearEvents**()**

**}**

**public** **interface** IEntity**<out** TKey**>{**

TKey Id **{** **get;** **}**

**}**

**CreateCustomer.cs**

**public** class CreateAccountHandler **:** INotificationHandler**<**CreateAccount**>{**

**private** **readonly** IEventsService**<**Customer**,** Guid**>** \_customerEventsService**;**

**private** **readonly** IEventsService**<**Account**,** Guid**>** \_accountEventsService**;**

**public** **async** Task Handle**(**CreateAccount command**,** CancellationToken cancellationToken**){**

var customer **=** **await** \_customerEventsService**.**RehydrateAsync**(**command**.**CustomerId**);**

**if(null** **==** customer**)**

**throw** **new** ArgumentOutOfRangeException**(**nameof**(**CreateAccount**.**CustomerId**),** "invalid customer id"**);**

var account **=** **new** Account**(**command**.**AccountId**,** customer**,** command**.**Currency**,** command**.**firstName**,** command**.**lastName**,** command**.**DoB**,** command**.**username**,** command**.**gender**);**

**await** \_accountEventsService**.**PersistAsync**(**account**);**

**}**

**}**