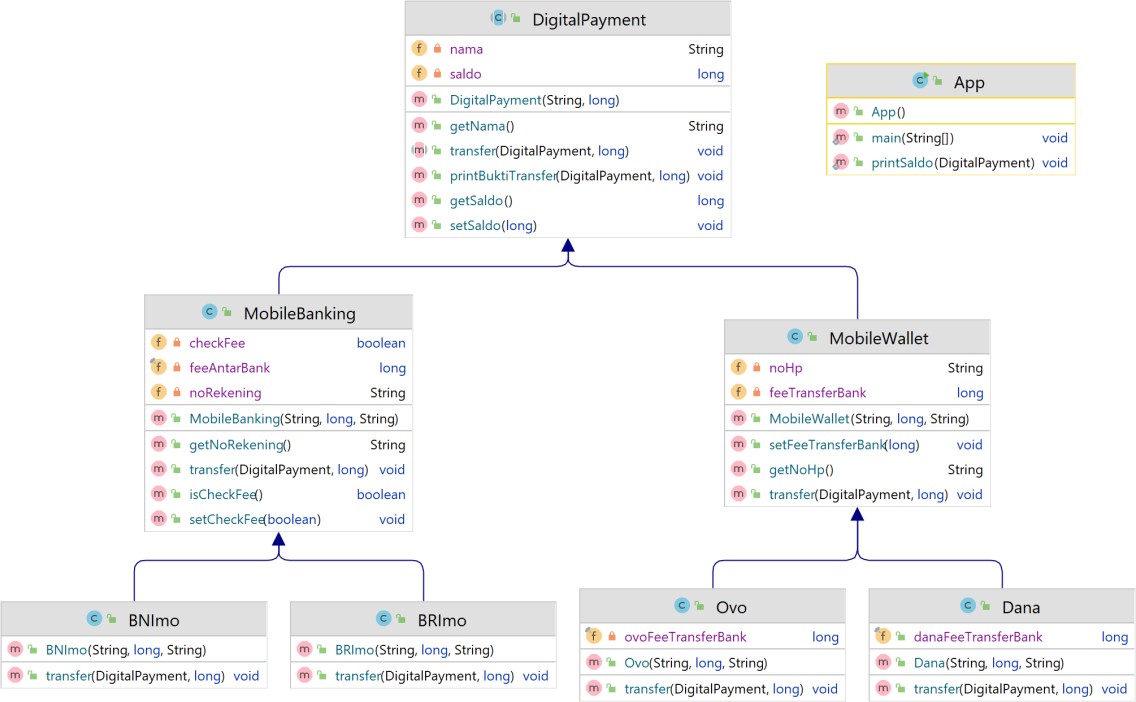
|  |  |
| --- | --- |
| **UNGUIDED PRAKTIKUM RPL-BO**  **POLIMORFISME** | **Soal Tipe A** |

# Waktu Pengerjaan: 2 jam 30 menit Petunjuk Pengumpulan Jawaban

Jika Anda berhasil membuka file ini, berarti Anda telah menerima tugas unguided dengan baiak. Silahkan mengerjakan tugas yang diberikan dengan teliti!

# Soal (100 Poin)

Digital payment yang biasa kita gunakan umumnya diklasifikasikan ke dalam dua kategori, yaitu Mobile Banking (e-Banking) dan Mobile Wallet (e-Wallet). Kamu sebagai seorang programmer yang handal diminta oleh perusahaan untuk membuat proyek bernama e- Transaction. Proyek ini memiliki prinsip kerja yang menyerupai QRIS dimana seseorang dapat melakukan transfer ke berbagai macam bentuk digital payment. Perhatikan gambar class diagram di bawah ini untuk membantu kamu dalam mengerjakan proyek tersebut.



Buatlah class yang diperlukan sesuai dengan gambar class diagram di atas. Berikut adalah detail terkait masing-masing class yang perlu kamu perhatikan:

|  |  |
| --- | --- |
| **Abstract Class DigitalPayment** | |
| **Keterangan:** Class ini bersifat abstract dan **sudah tersedia di dalam arsip soal**. Tujuan dari class ini yaitu menggeneralisasi seluruh bentuk objek alat pembayaran digital. | |
| DigitalPayment(String nama, long  saldo) | Menginisialisasi atribut nama dan saldo sesuai  dengan parameter |
| transfer(DigitalPayment dp, long  nominal) | Method ini bersifat abstract dan berfungsi untuk  melakukan transfer antar alat pembayaran digital |
| printBuktiTransfer(DigitalPayment  penerima,long nominal) | Method ini berfungsi layaknya bukti transfer dan  akan dipanggil apabila proses transfer sukses. |

|  |  |
| --- | --- |
| **Class MobileBanking** | |
| **Keterangan:** Salah satu bentuk dari alat pembayaran digital yaitu Mobile Banking (e- Banking). | |
| noRekening | Atribut ini merupakan identitas untuk akun mobile  banking |
| checkFee | Atribut ini memiliki **nilai default** yaitu **false** dan bertujuan untuk memberikan keputusan terkait  penambahan fee dalam proses transfer. |
| feeAntarBank | Atribut ini memiliki **nilai default** yaitu **6000** dan berfungsi untuk tambahan biaya apabila terjadi proses transfer antar bank.  **\*Keterangan: Antar bank sama dengan beda bank** |
| MobileBanking(String nama, long  saldo, String noRekening) | Menginisialisasi atribut nama, saldo, dan  noRekening sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Method ini merupakan override dari method abstract yang ada pada class DigitalPayment. Berikut beberapa hal yang perlu kamu perhatikan:   * Apabila saldo kurang dari 0, maka transfer gagal dan tampilkan pesan input tidak valid * Apabila saldo tidak mencukupi, maka transfer gagal dan tampilkan pesan kegagalan * Apabila atribut **checkFee bernilai true**, maka **saldo pengirim akan dikurangi dengan nominal dan feeAntarBank** * **Saldo penerima** akan selalu **bertambah sesuai dengan nominal transfer**. * Apabila proses transfer berhasil, maka panggil method **printBuktiTransfer(DigitalPayment penerima,long nominal)** yang ada pada   class DigitalPayment. |

|  |  |
| --- | --- |
| **Class MobileWallet** | |
| **Keterangan:** Salah satu bentuk dari alat pembayaran digital yaitu Mobile Wallet (e-Wallet). | |
| noHP | Atribut ini merupakan identitas untuk akun mobile  wallet |
| feeTransferBank | Atribut ini akan diisi sesuai dengan fee masing- masing MobileWallet untuk melakukan transfer ke  bank. |
| MobileWallet(String nama, long  saldo, String noHp) | Menginisialisasi atribut nama, saldo, dan noHp  sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Method ini merupakan override dari method abstract yang ada pada class DigitalPayment. Berikut beberapa hal yang perlu kamu perhatikan:   * Apabila saldo kurang dari 0, maka transfer gagal dan tampilkan pesan input tidak valid * Apabila saldo tidak mencukupi, maka transfer gagal dan tampilkan pesan kegagalan * Apabila object dp merupakan instance dari BNImo atau BRImo, maka **saldo pengirim akan dikurangi dengan nominal dan feeTransferBank** * **Saldo penerima** akan selalu **bertambah sesuai dengan nominal transfer**. * Apabila proses transfer berhasil, maka panggil method **printBuktiTransfer(DigitalPayment**   **penerima,long nominal)** yang ada pada class DigitalPayment. |

|  |  |
| --- | --- |
| **Class BNImo** | |
| **Keterangan:** Salah satu platform mobile banking (e-Banking) yaitu BNI mobile. | |
| BNImo(String nama, long saldo,  String noRekening) | Menginisialisasi atribut nama, saldo, dan  noRekening sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Override dari method yang ada pada class MobileBanking. Berikut beberapa hal yang perlu kamu perhatikan:   * Apabila object dp merupakan instance dari BRImo, maka atribut checkFee pada class MobileBanking akan bernilai true. * Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileBanking. |

|  |  |
| --- | --- |
| **Class BRImo** | |
| **Keterangan:** Salah satu platform mobile banking (e-Banking) yaitu BRI mobile. | |
| BRImo(String nama, long saldo, String noRekening) | Menginisialisasi atribut nama, saldo, dan noRekening sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Override dari method yang ada pada class MobileBanking. Berikut beberapa hal yang perlu kamu perhatikan:   * Apabila object dp merupakan instance dari BNImo, maka atribut checkFee pada class MobileBanking akan bernilai true. * Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileBanking |

|  |  |
| --- | --- |
| **Class Dana** | |
| **Keterangan:** Salah satu platform mobile wallet (e-Wallet) yaitu DANA. | |
| danaFeeTransferBank | Atribut ini memiliki nilai default yaitu 1000 dan berfungsi sebagai biaya tambahan apabila terjadi  proses transfer dari DANA ke rekening bank. |
| Dana(String nama, long saldo, String  noHp) | Menginisialisasi atribut nama, saldo, dan noHp  sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Override dari method yang ada pada class MobileWallet. Berikut beberapa hal yang perlu kamu perhatikan:   * Isi atribut feeTransferBank pada class MobileWallet sesuai dengan danaFeeTransferBank. * Apabila object dp merupakan instance dari Ovo, maka transfer gagal dan tampilkan pesan yaitu akun DANA tidak valid. * Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileWallet. |

|  |  |
| --- | --- |
| **Class Ovo** | |
| **Keterangan:** Salah satu platform mobile wallet (e-Wallet) yaitu OVO. | |
| ovoFeeTransferBank | Atribut ini memiliki nilai default yaitu 2000 dan  berfungsi sebagai biaya tambahan apabila terjadi proses transfer dari OVO ke rekening bank. |
| Dana(String nama, long saldo, String  noHp) | Menginisialisasi atribut nama, saldo, dan noHp  sesuai dengan parameter. |
| transfer(DigitalPayment dp, long nominal) | Override dari method yang ada pada class MobileWallet. Berikut beberapa hal yang perlu kamu perhatikan:   * Isi atribut feeTransferBank pada class MobileWallet sesuai dengan ovoFeeTransferBank. |

|  |  |
| --- | --- |
|  | * Apabila object dp merupakan instance dari Dana, maka transfer gagal dan tampilkan pesan yaitu akun OVO tidak valid. * Proses transfer akan terjadi sesuai dengan aturan yang ada pada class MobileWallet |

Selanjutnya, gunakan **class App yang sudah tersedia pada arsip soal** untuk menguji program kamu. Apabila algoritma yang kamu sudah tepat, maka masing-masing skenario akan menghasilkan output yang sesuai dengan gambar di bawah ini:

|  |  |
| --- | --- |
| **Skenario 1 – 9** | **Skenario 10** |
|  | Table  Description automatically generated with low confidence |

# Komponen Penilaian:

* Program dibuat dengan menggunakan Maven dan struktur project sesuai. (**Poin 5**)
* Class yang dibuat memiliki method, constructor, serta atribut yang lengkap sesuai class diagram dan ketentuan pada soal. (**Poin 25**)
* Override method transfer pada class MobileWallet memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. (**Poin 15**)
* Override method transfer pada class MobileRekening memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. (**Poin 15**)
* Override method transfer pada class Dana, Ovo, BRImo, dan BNImo memiliki algoritma yang tepat dan sesuai dengan ketentuan pada soal. (**Poin 20** = 4 x 5 Poin)
* Output skenario 1-9 tepat dan sesuai (**Poin 10**)
* Output skenario 10 tepat dan sesuai (**Poin 10**)