

1 April 2022

TUGAS JURNAL MODUL 7

1. MEMBUAT PROJECT CONSOLE/TANPA GUI

Buka IDE misalnya dengan Visual Studio

- A. Pilih "Create a new project" kemudian pilih "Console App". Pada IDE lain pada umumnya hanya perlu membuat project baru saja.
- B. Masukkan project baru dengan nama modul7_NIM.

2. MELAKUKAN GIT COMMIT PADA PROJECT YANG DIBUAT

Task atau langkah-langkah yang perlu dikerjakan adalah sebagai berikut:

- A. Buatlah github public repository kosong (pastikan bagian "*Initialize this repository with*" tidak ada yang dicentang pada saat membuat repository baru) melalui <https://github.com/>
- B. Melakukan inisialisasi git repository di folder project yang dibuat.
- C. Pastikan untuk menambahkan file ".gitignore" baik manual atau dengan menggunakan visual studio/IDE. Untuk project dengan C# dapat melihat referensi file ".gitignore" pada link berikut ini:

<https://github.com/github/gitignore/blob/main/VisualStudio.gitignore>
- D. Membuat commit untuk versi pertama dari project yang dibuat dengan pesan commit bebas.
- E. Melakukan git push ke github repo.

3. IMPLEMENTASI RUNTIME CONFIGURATION

Dari master/main branch dan class utama, buatlah implementasi program yang memanfaatkan teknik Runtime Configuration dengan spesifikasi sebagai berikut ini:

- A. Tambahkan class baru untuk memproses file konfigurasi dengan nama "BankTransferConfig".
- B. File konfigurasi yang digunakan adalah "bank_transfer_config.json" dengan format sebagai berikut ini:

```
{
  "lang": "CONFIG1",
  "transfer": {
    "threshold": "CONFIG2",
    "low_fee": "CONFIG3",
    "high_fee": "CONFIG4"
  },
  "methods": CONFIG5,
  "confirmation": {
    "en": "CONFIG6",
    "id": "CONFIG7"
  }
}
```

C. Nilai default yang digunakan apabila file config masih belum ada yaitu:

- i. CONFIG1 => "en"
- ii. CONFIG2 => 25000000
- iii. CONFIG3 => 6500
- iv. CONFIG4 => 15000
- v. CONFIG5 => ["RTO (real-time)", "SKN", "RTGS", "BI FAST"]
- vi. CONFIG6 => "yes"
- vii. CONFIG7 => "ya"

D. Aplikasi (program utama) memiliki alur atau fungsi sebagai berikut (yang menggunakan class config "BankTransferConfig"):

- i. Pada saat program dijalankan program akan menampilkan pesan sebagai berikut tergantung nilai dari CONFIG1 atau JSON untuk key "lang":
 - *CONFIG1 == "en" => "Please insert the amount of money to transfer:"*
 - *CONFIG1 == "id" => "Masukkan jumlah uang yang akan di-transfer:"*
- ii. Setelah itu aplikasi akan menampilkan biaya transfer dan total biaya yang akan dibayarkan dengan ketentuan sebagai berikut:
 - *Apabila jumlah yang di-transfer pada bagian sebelumnya (D-i) kurang dari atau sama dengan nilai CONFIG2 atau "threshold", maka biaya transfer adalah CONFIG3 atau "low_fee". Akan tetapi jika lebih dari "threshold", maka biaya transfer diambil dari nilai CONFIG4 atau "high_fee".*
 - *Total biaya yang perlu dibayarkan adalah hasil penjumlahan dari jumlah uang yang akan ditransfer dan biaya transfer.*
 - *Pesan output apabila CONFIG1 atau "lang" bernilai "en" adalah "Transfer fee = <biaya_transfer>" dan "Total amount = <nominal_transfer + biaya_transfer>."*
 - *Pesan output apabila CONFIG1 atau "lang" bernilai "id" adalah "Biaya transfer = <biaya_transfer>" dan "Total biaya = <nominal_transfer + biaya_transfer>."*
- iii. Setelah itu aplikasi akan menampilkan pesan yang bergantung dari bahasa yang dipilih di konfigurasi:
 - *EN => "Select transfer method:"*
 - *ID => "Pilih metode transfer:"*
- iv. Kemudian juga akan dilakukan print dengan numbering untuk setiap string yang ada di JSON untuk key "methods" atau CONFIG5. Contoh output (dari nilai default):
 - *1. RTO (real-time)*
 - *2. SKN*
 - *3. RTGS*
 - *4. BI FAST*
- v. Setelah menerima input pada pertanyaan sebelumnya, aplikasi akan bertanya satu pertanyaan terakhir dengan isi yang bergantung dari nilai "lang" dan "confirmation"
 - *EN => Please type "<CONFIG6>" to confirm the transaction:*
 - *ID => Ketik "<CONFIG7>" untuk mengkonfirmasi transaksi:*
- vi. Apabila input dari user sesuai dengan CONFIG6 atau CONFIG7 (tergantung nilai "lang"), maka:

- *EN => The transfer is completed*
 - *ID => Proses transfer berhasil*
- vii. Tetapi jika input dari user tidak sesuai, maka:
- *EN => Transfer is cancelled*
 - *ID => Transfer dibatalkan*
- E. Ubah nilai default pada json file “bank_transfer_config.json” dengan nilai yang berbeda dan tunjukkan ke asisten praktikum hasil run dari aplikasi.

4. MELAKUKAN COMMIT

Pada branch master/main:

- A. Lakukan commit dengan pesan “aplikasi bank transfer dengan runtime configuration”.
- B. Lakukan push ke github ke branch yang dibuat di bagian sebelumnya.

5. PENGUMPULAN FILE/TUGAS JURNAL

Sebelum pengumpulan, praktikan wajib menunjukkan hasil run via share screen ke asprak. Kumpulkan semua file berikut dalam bentuk file zip/rar/7zip:

- A. Source code dari project yang dibuat
- B. File docx/pdf yang berisi:
 - i. Link github repository kelompok
 - ii. Screenshot hasil run (hasil console output dari proses run)
 - iii. Penjelasan singkat dari kode implementasi yang dibuat (beserta screenshot dari potongan source code yang dijelaskan).

Catatan: Tidak ada file docx/pdf (screenshot dan penjelasan) yang dikumpulkan DAN juga tidak melakukan demo/menunjukkan hasil run ke asprak maka nilai jurnal akan 0

KOMPONEN PENILAIAN

- A. Penggunaan Git dan Github **[15 pts]**
- B. Pembuatan class konfigurasi **[30 pts]**
- C. Program utama yang menggunakan class konfigurasi **[40 pts]**
- D. Laporan jurnal **[15 pts]**