

# Spesifikasi Tugas Besar

## IF1210 Dasar Pemrograman

### Deskripsi Persoalan

Pada zaman dahulu kala, terdapat sebutir telur misterius. Telur tersebut menetas menjadi seekor pokemon bernama Arceus. Ia memiliki sebuah kemampuan, yaitu menciptakan pokemon-pokemon lain. Setiap harinya, ia membuat sebuah pokemon baru, sehingga pokemon telah memenuhi permukaan planet.



Sayangnya, Arceus berada di planet yang berbeda. Anda sebagai mahasiswa STEI ITB tertarik dengan Arceus. Anda ingin mengetahui bagaimana rasanya memiliki kemampuan Arceus. Namun, anda tidak mengetahui bagaimana caranya membuat pokemon baru. Untungnya, anda mengerti cara memprogram di Pascal. Oleh karena itu, anda ingin membuat sebuah program yang dapat mensimulasikan kemampuan Arceus. Dapatkah Anda membuatnya?

# Spesifikasi Program

## F01 - Login

Trainer harus login terlebih dahulu sebelum dapat berinteraksi dengan pokebon. Jika pengguna sudah pernah login sebelumnya, maka ia akan otomatis mengambil data dari file eksternal.

```
$ login
Masukkan username: Ret

Selamat datang Ret!
```

Jika nama trainer yang diinput belum pernah login sebelumnya, maka ia akan menjalankan permainan dengan data permainan default (didefinisikan sendiri)

```
$ login
Masukkan username: Rex

Tidak ada username Rex pada database. Sistem akan memulai permainan baru.
```

Username "admin" secara khusus dibuat untuk menjalankan trainer dengan akses lebih.

```
$ login
Masukkan username: admin

User admin telah login.
```

## F02 - Pokebon Database

Command untuk melihat pokebon apa saja yang ada di pokedex. Output diberikan dengan format **Nomor pokebon | Nama pokebon | Tipe pokebon | Evolusi selanjutnya**. Jika tidak ada evolusi selanjutnya dari pokebon tersebut, cukup keluarkan "-" pada entri tersebut. Output diurutkan berdasarkan nomor dari suatu pokebon.

```
$ pokedex

Pokebon di pokedex:

1 | Charmanber | Api | Charmeleber
2 | Charmeleber | Api | -
```

## F03 - Mengecek Inventori

Command untuk melihat isi inventori beserta pokebon apa saja yang dimiliki. Isi inventori diberikan dengan format **Nomor inventori | Nama Pokebon | Level Pokebon | Kondisi Pokebon**. Banyak pokebon maksimal yang dapat ditampung oleh trainer adalah 10.

### **\$ inventori**

Isi inventory saat ini:

- 1 | Charmanber | Level 2 | Lelah
- 2 | Charmanber | Level 1 | Sehat

## F04 - Lepaskan Pokebon

Trainer dapat melepas pokebon yang dimilikinya dengan memberikan nomor inventori dari pokebon yang bersangkutan.

### **\$ lepas**

Masukkan nomor inventori yang ingin dilepas: **1**

Pelepasan berhasil!

Isi inventory saat ini:

- 1 | Charmanber | Level 1 | Sehat

Jika tidak ada pokebon pada inventori yang bersangkutan, maka ia akan mengeluarkan pesan error.

### **\$ lepas**

Masukkan nomor inventori yang ingin dilepas: **3**

Tidak ada pokebon pada nomor inventori 3.

## F05 - Encounter Chance

Setiap pokebon memiliki kemungkinan acak untuk ditemui di alam. Persentase acak dibebaskan, selama suatu pokebon pasti memiliki chance lebih besar dibandingkan chance dari hasil evolusi pokebon tersebut. Jumlah dari kemungkinan menemukan setiap pokebon harus bernilai 100% atau sangat mendekati 100%. Output dari persentase diberikan sampai 2 angka di belakang koma. **Setiap pokebon harus memiliki chance > 0%**

Hint: Buatlah sebuah array yang menyatakan porsi masing-masing dari setiap pokebon

### **\$ pokebonChance**

Kemungkinan menemukan setiap pokebon:

- Pikaboo = 30.00%
- Charmanber = 50.00%
- Charmeheber = 20.00%

## F06 - Menangkap Pokebon

Trainer dapat menangkap pokebon secara acak sesuai dengan encounter chance masing-masing pokebon. Setiap pokebon yang didapat akan memiliki kondisi lelah, dan dipastikan trainer akan selalu mendapat pokebon yang ditemuinya.

### **\$ tangkap**

Kamu mendapat Pikaboo!

Isi inventori saat ini:

1 | Pikaboo | Level 1 | Lelah

2 | Charmanber | Level 1 | Sehat

## F07 - Bertarung dengan Pokebon

Trainer dapat bertarung dengan pokebon tak bertuan. Pokebon yang ditemui memiliki kemungkinan sesuai dengan encounter chance dari pokebon tersebut. Trainer dapat memilih pokebon apa yang ingin digunakan untuk melawan pokebon tak bertuan tersebut. Syarat dari pokebon yang dipakai adalah hanya ia tidak boleh dalam kondisi lelah. Dipastikan pokebon trainer akan selalu memenangi pertarungan. Pokebon yang dipakai juga akan mengalami kenaikan level setelah ia bertarung.

### **\$ tarung**

Kamu menemukan Pikaboo!

Pilih pokebon!

**2**

Charmanber siap bertarung!

Charmanber menang!

Charmanber mengalami kenaikan level!

Level dari Charmanber sekarang adalah level 4.

Untuk pokebon yang dipakai, ia memiliki kemungkinan sebesar 50% untuk menjadi lelah setelah bertarung.

### **\$ tarung**

Kamu menemukan Pikaboo!

Pilih pokebon!

**2**

Charmanber siap bertarung!

Charmanber menang!

Charmanber mengalami kenaikan level!

Level dari Charmanber sekarang adalah level 5.  
Karena pertarungan yang sengit ini, Charmanber menjadi lelah. Ia ingin segera kembali ke rumah.

## F08 - Statistik

Trainer dapat melihat statistik untuk permainan yang dimilikinya. Statistik berisi berapa pokemon unik yang pernah dimiliki beserta maksimum level yang pernah dicapai untuk setiap pokemon yang ada

### \$ statistik

Sekarang adalah hari ke-4  
Pokemon unik yang telah dimiliki: 2/3

Maksimum level dari setiap pokemon:  
Pikaboo: Level 5  
Charmanber: Level 2  
Charmeber: Level 0

## F09 - Membuat Pokemon

Admin dapat membuat pokemon baru. Syarat dari pokemon baru hanyalah tidak adanya pokemon pada database dengan nama yang sama

### \$ buatPokemon

Nama pokemon yang ingin dibuat: **Squirble**  
Tipe pokemon yang ingin dibuat: **Udara**

Pokemon baru berhasil dibuat!

Jika sudah ada pokemon dengan nama yang sama, maka perintah akan gagal dilaksanakan

### \$ buatPokemon

Nama pokemon yang ingin dibuat: **Charmanber**  
Tipe pokemon yang ingin dibuat: **Api**

Terdapat kesalahan pada pembuatan pokemon.

## F10 - Membuat Evolution Chart Pokemon

Admin dapat membuat keterkaitan alur evolusi antara 2 buah pokemon.

### \$ buatEvolusi

Masukkan nama pokebon sebelum evolusi: **Bulbadaun**  
Masukkan nama pokebon sesudah evolusi: **Ivydaun**

Alur evolusi berhasil disambungkan!

Jika pokebon awal sudah memiliki evolusi setelahnya **atau** pokebon akhir merupakan hasil dari evolusi, maka command akan mengeluarkan pesan error

**\$ buatEvolusi**

Masukkan nama pokebon sebelum evolusi: **Charmanber**  
Masukkan nama pokebon sesudah evolusi: **Ivydaun**

Alur evolusi tidak memenuhi syarat.

## F11 - Melihat Daftar Evolution Chart

Selama permainan berjalan, semua user dapat melihat data alur evolusi dari pokebon-pokebon yang ada.

**\$ lihatEvolusi**

List alur evolusi:

- 1 | Charmanber - Charmeleber - Charibard
- 2 | Bulbadaun - Ivydaun - Venudaun - Mega Venudaun - Omega Venudaun

## F12 - Menghapus Evolution Chart

Admin dapat menghapus evolusi yang ada. Input yang diharapkan dapat berupa angka ataupun nama pokebon. Jika input berupa angka, maka ia akan menghapus evolusi dengan nomor tersebut. Sisa dari list alur evolusi harus tetap terurut mulai dari 1

**\$ hapusEvolusi**

Masukkan evolusi yang ingin dihapus: **1**

List alur evolusi:

- 1 | Bulbadaun - Ivydaun - Venudaun - Mega Venudaun - Omega Venudaun

Jika input berupa kata, maka ia akan memutus rantai evolusi dari pokebon tersebut.

**\$ hapusEvolusi**

Masukkan evolusi yang ingin dihapus: **Venudaun**

List alur evolusi:

- 1 | Charmanber - Charmeleber - Charibard

2 | Bulbadaun - Ivydaun  
3 | Mega Venudaun - Omega Venudaun

## F13 - Melakukan Evolusi pada Pokebon

Trainer dapat melakukan evolusi pada pokebon yang dimilikinya. Syarat suatu pokebon dapat di evolusi adalah telah memiliki level 5 atau lebih. **Setelah melakukan evolusi, level pokebon akan kembali ke 1 berapapun levelnya sebelum evolusi**

**\$ evolusi**

Masukkan nomor inventori dari pokebon yang ingin dievolusi: **1**

Charmanber telah dievolusi menjadi Charmeleber!

Jika tidak ada evolusi lanjutan, maka permainan akan mengembalikan pesan error

**\$ evolusi**

Masukkan nomor inventori dari pokebon yang ingin dievolusi: **1**

Charmeleber sudah merupakan evolusi terakhir.

Jika tidak ada yang memenuhi persyaratan evolusi pada nomor inventori tersebut, maka permainan akan mengembalikan pesan error

**\$ evolusi**

Masukkan nomor inventori dari pokebon yang ingin dievolusi: **1**

Tidak ada yang bisa dievolusi.

## F14 - Tidur

Trainer dapat memulihkan energi pokebonnya dengan tidur. Dengan melakukan aksi ini, trainer akan menyudahi hari ini dan melanjutkan petualangannya di esok hari.

**\$ tidur**

Pokebonmu sudah tidur dengan lelap. Mereka sudah kembali sehat!  
Sekarang adalah hari ke-5.

## F15 - Save File

Trainer dapat menjalankan fungsi save file untuk menyimpan keadaan permainannya sekarang.

**\$ save**

~~Masukkan nama file Pokebon:~~ **Pokebon.csv**

~~Masukkan nama file Evolusi:~~ **Evolusi.csv**

Masukkan nama file Inventori: **Inventori.csv**

Masukkan nama file Stats: **Stats.csv**

File berhasil disimpan!

Jika fungsi save dilakukan oleh admin, maka ia akan meminta nama file yang berbeda.

**\$ save**

Masukkan nama file Pokebon: **Pokebon.csv**

Masukkan nama file Evolusi: **Evolusi.csv**

Masukkan nama file Inventori: **Inventori.csv**

Masukkan nama file Stats: **Stats.csv**

File berhasil disimpan!

Penyimpanan file memiliki struktur sesuai definisi dibawah

## F16 - Exit

Pengguna dapat menjalankan perintah exit. Jika dijalankan saat ada permainan yang berjalan, maka ia akan menghentikan permainan. Permainan akan melakukan save secara otomatis berdasarkan file save yang telah ditentukan saat memanggil fungsi save. Jika pemain belum pernah memanggil fungsi save, maka permainan akan melakukan save pada file default (definisikan sendiri).

**\$ exit**

Selamat jalan Ret!

Jika tidak ada permainan yang berjalan, maka ia akan menghentikan interaksi

**\$ exit**

Permainan selesai.



## Spesifikasi Bonus

### B01 - Penghapusan Pokebon

Admin dapat menjalankan fungsi menghapus pokebon dari pokedex. Jika pokebon tersebut terletak pada sebuah alur evolusi, maka alur evolusi tersebut juga terhapus/terpotong sama seperti jika memanggil fungsi hapusEvolusi.

**\$ hapusPokebon**

Masukkan Pokebon yang ingin dihapus: **Pikaboo**

Pikaboo telah dihapus.

Jika terdapat user yang memiliki pokebon yang telah dihapus, maka pada saat login selanjutnya, data tersebut akan dihapus.

**\$ login**

Masukkan username: **Ret**

Selamat datang Ret!

Pokebon Pikaboo pada nomor inventori 2 telah dihapus dari inventori

Pokebon Pikaboo telah dihapus dari stats

Jika Id dari pokebon bukan merupakan id terbesar pada pokedex, maka id pokebon tersebut pada pokedex tidak akan bisa dipakai lagi

### B02 - Default Save

Jika pengguna belum menentukan save, maka permainan mempunyai default save file sendiri. Untuk setiap trainer, dipastikan default savenya berbeda antara satu sama lain. Cara mengimplementasikan bonus ini dibebaskan kepada mahasiswa, dengan tidak ada perbedaan nilai selama bonus ini dapat dijalankan dengan baik.

## Catatan Tambahan

- Pembagian fitur atau fungsionalitas di atas tidak merepresentasikan dekomposisi modul, fungsi, dan prosedur yang sesungguhnya.

## Struktur Data File Eksternal

Program perlu membaca beberapa data dari file eksternal untuk mengoperasikan sistem ini. Format file eksternal yang diminta adalah file dengan ekstensi **.csv**

### File Pokebon

ID\_Pokebon, Nama, Tipe, Evolusi\_Selanjutnya

Keterangan:

- ID\_Pokebon: Id dari Pokebon
- Nama: Nama Pokebon
- Tipe: Tipe Pokebon
- Evolusi\_Selanjutnya: Evolusi selanjutnya dari pokebon tersebut. Evolusi diasumsikan ada. Jika tidak ada evolusi setelahnya, maka isi dari kolom ini adalah '-'.

### File Evolusi Pokebon

ID\_Evolusi, Alur\_Evolusi

Keterangan:

- ID\_Evolusi: ID dari evolusi
- Alur\_Evolusi: Alur dari evolusi tersebut, setiap pokebon yang berbeda dipisahkan dengan '-'

### File Trainer

Nama, Day\_Passed, File\_Inventori, File\_Stats

Keterangan:

- Nama: Nama Trainer
- Day\_Passed: Berapa hari yang telah dilalui
- File\_Inventori: File inventori terakhir yang dipakai untuk melakukan save
- File\_Stats: File Stats terakhir yang dipakai untuk melakukan save

### File Inventori

Nomor\_Inventori, Nama\_Pokebon, Level, Kondisi

Keterangan:

- Nomor\_Inventori: Nomor inventori, urut dari 1
- Nama\_Pokebon: Nama Pokebon
- Level: Level Pokebon
- Kondisi: Kondisi Pokebon

### File Stats

Nama\_Pokebon, Max\_Level

Keterangan:

- Nama\_Pokebon: Nama Pokebon
- Max\_Level: Maximum level yang pernah didapat untuk pokebon tersebut

## Deliverables

1. *Source code* program yang dibuat sesuai standar yang diajarkan di kuliah, bersih (hanya mengandung bagian-bagian yang diperlukan), dan *well-commented*.
2. **Laporan Tugas Besar** dibuat dengan template sebagai berikut:
  - a. Halaman Cover, berisi minimum Kode dan Nama Kuliah, Nama Tugas, Nomor Kelas, Nomor Kelompok, NIM dan Nama Anggota Kelompok, Nama Sekolah (Sekolah Teknik Elektro dan Informatika), dan Perguruan Tinggi (Institut Teknologi Bandung), dan Tahun (2019).
  - b. Daftar Isi.
  - c. Daftar Tabel (jika ada).
  - d. Daftar Gambar (jika ada).
  - e. Deskripsi persoalan: berisi penjelasan kembali (dengan “bahasa sendiri”) tentang persoalan yang akan diselesaikan.
  - f. Daftar pembagian kerja anggota kelompok, berdasarkan fitur yang dibuat, meliputi desain, implementasi/koding, dan testing. Contoh:

Fitur	Implementasi *)	NIM Desainer **)	NIM Coder **)	NIM Tester **)
F01 - Login	procedure auth  function validate	16518501 16518511	16518511	16518524
F02 - Pokebon Database	procedure database	16518511	16518524	16518511
F03 - Mengecek Inventori	procedure inventory	16518524	16518501	16518501
...	...	...	...	...

\*) bisa menjadi procedure/fungsi/bagian dari program utama dan bisa lebih dari 1

\*\*) bisa dikerjakan oleh lebih dari 1 orang

- g. Checklist hasil rancangan, implementasi dan testing setiap primitif. Contoh Checklist:

Fitur	Desain	Implementasi	Testing
F01 - Login	V	X	X
F02 - Pokebon Database	V	V	V
F03 - Mengecek Inventori	X	-	-
...	...	...	...

Keterangan: V: sudah selesai dikerjakan, X: dikerjakan, tapi belum selesai, -: tidak dikerjakan sama sekali.

- h. Desain command untuk setiap primitif (berisi: nama command, masukan, dan keluaran).
- i. Desain kamus data (dalam notasi algoritmik).
- j. Desain dekomposisi algoritmik dan fungsional program.
- k. Spesifikasi untuk tiap modul/prosedur/fungsi yang dibuat (dalam notasi algoritmik).
- l. Berikan *screenshot* hasil pengujian program berdasarkan fitur-fitur pada spesifikasi. Pada setiap fitur, *screenshot* minimal berisi: (1) data masukan, (2) data keluaran.
- m. Lampiran: Hasil scan form asistensi.