**Laporan Aplikasi**

**Bon Appétit**

****

**Kelas : IF-38-02**

**Disusun Oleh :**

**Ubassy Abdillah 1301148282**

**Raja Ryan Pradana 1301140302**

**Dzakyta Afuzagani 1301140122**

**Mohammad Zakie Faiz Rahiemy 1301144422**

**PEMROGRAMAN BERBASIS OBJEK**

**PRODI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY**

**Oktober 2016**

# Daftar Menu

[Daftar Menu 1](#_Toc464590730)

[1. Deskripsi Aplikasi 2](#_Toc464590731)

[2. Target Pengguna 2](#_Toc464590732)

[3. Desain *Class* *Diagram* 3](#_Toc464590733)

[4. Tampilan Aplikasi 4](#_Toc464590734)

[5. Daftar Fungsionalitas 4](#_Toc464590735)

[6. *Source* *Code* 5](#_Toc464590736)

[7. Pengujian Aplikasi 5](#_Toc464590737)

[8. Kebutuhan Minimum Aplikasi 6](#_Toc464590738)

[9. Laporan Scrum 6](#_Toc464590739)

[10. BurnDown Chart 7](#_Toc464590740)

[11. Penutup 7](#_Toc464590741)

# Deskripsi Aplikasi

*Bonjour*!

***Bon Appétit*** adalah aplikasi permainan dimana pengguna dapat berperan sebagai bagian dari rantai makanan. Terdapat tiga spesies yang disebut appétit dengan kemampuan yang berbeda-beda, dan satu jenis makanan yang disebut bon. Semakin banyak appétit memakan bon, atau appétit lain yang levelnya dibawah, maka semakin cepat juga appétit tersebut berevolusi menjadi organisme yang lebih besar.

Semakin lama, tingkat kesulitan akan semakin tinggi karena bukan hanya appétit milik pengguna saja yang dapat berevolusi. Apakah anda akan menduduki puncak rantai makanan, atau justru mati masuk surga? Selamat Makan!

# Target Pengguna

Target pengguna dari aplikasi ini adalah *casual gamer.* Area dan usia tidak dibatasi.

# Desain *Class* *Diagram*

|  |
| --- |
| GameMap |
| - map[][] : Asset  - row : int  - column : int |
| + GameMap(row : int, column : int)  + getRow() : int  + getColumn() : int  *+* addAsset(a : Asset)  + getAsset(a : Asset) : Asset  + display()  + display2()  + nullMap()  + getMap() : Asset [] [] |

|  |
| --- |
| Game (Application Class) |
| ~ m : GameMap  ~ run, win : bool  ~ asset[] : Asset  ~dead : int |
| + Game()  + moving(a : Asset)  + moving(a : Asset, in : char)  + isWin() : bool  + setWin(win : bool)  + getDead() : int  + setDead(dead : int)  + isRun() : bool  + setRun(run : bool)  + getM() : GameMap  + setM(m : GameMap)  + getAsset() : Asset[]  + setAsset(asset : Asset[])  + removeAsset(i : int)  + setAsset2(i : int, a : Asset)  + getUp() : char  + setUp(up : char)  + getDown() : char  + setDown(down : char)  + getLeft() : char  + setLeft(left : char)  + getRight() : char  + setRight(right : char)  + getSkill() : char  + setSkill(skill : char)  + getSave() : char  + setSave(save : char)  + getQuit() : char  + setQuit(quit : char) |

|  |
| --- |
| *Asset* |
| - type : int  - level : int  - exp : int  - duration : int  - delay : int  # expMax[] : int  - posX : int  - posY : int |
| + Asset(posY, posX : int)  *+ move(dir : char, row, collumn : int, map : Asset [] [])*  *+ evolution()*  *+ skill()*  + getType() : int  + setType(type : int)  + getLevel() : int  + setLevel(level : int)  + getExp() : int  + setExp(exp : int)  + getPosX() : int  + setPosX(posX : int)  + getPostY()  + setPosY(posY : int)  + getDuration() : int  + setDuration(speciality : int) |

|  |
| --- |
| Food |
|  |
| *+* Food(posY, posX, delay : int)  + move(dir : char, row : int, map : Asset[] [])  + evolution()  + skill()  + toString() : String |

|  |
| --- |
| *Ordo* |
|  |
| + Ordo(posX, posY, level, delay : int)  *+* evolution()  + skill()  + *move(dir : char, row, collumn : int, map : Asset[] [])* |

|  |
| --- |
| SpeciesA |
|  |
| + SpeciesA(posY, posX, level, delay : int)  + move(dir : char, row, collumn int, map : Asset[] [])  + toString() : String |

|  |
| --- |
| SpeciesC |
|  |
| + SpeciesC(posY, posX, level, delay : int)  + move(dir : char, row, collumn int, map : Asset[] [])  + toString() : String |

|  |
| --- |
| SpeciesB |
|  |
| + SpeciesB(posY, posX, level, delay : int)  + move(dir : char, row, collumn int, map : Asset[] [])  + toString() : String |

# Tampilan Aplikasi

# Daftar Fungsionalitas

Aplikasi game Bon Appetit memiliki beberapa fungsionalitas diantaranya :

1. Create Map
   * Digunakan untuk membuat map pada aplikasi
2. Add Asset
   * Digunakan untuk menambah asset ke dalam map
3. Remove Asset
   * Digunakan untuk menghapus asset dari map
4. Moving Asset
   * Digunakan untuk merubah atribut posisi dari asset
5. New Game
   * Digunakan untuk instansiasi game dan asset baru
6. Load Game
   * Digunakan untuk mengabil objek asset dari file
7. Credit
   * Digunakan untuk menampilkan orang-orang yang bertanggung jawab

# *Source* *Code*

*Source code* dari aplikasi terlampir beserta file laporan ini.

# Pengujian Aplikasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Pengujian** | ***Expected*** | **Berhasil/Tidak** | **Keterangan** |
| 1 | Console : Menu Awal Input karakter | ditolak | Berhasil | Input karakter ditolak |
| 2 | Masuk New Game | instansiasi objek game | Berhasil | Objek game siap digunakan |
| 3 | Console : Set Karakter Difficulty | diterima | Berhasil | Difficult di set pada karakter yang sudah ditentukan |
| 4 | Console : Set Karakter Difficulty yang tidak terdaftar | diterima | Berhasil | Difficulty akan di set Easy |
| 5 | Console : Set Karakter Player | diterima | Berhasil | Player akan di set sesuai karakter yang terdaftar |
| 6 | Console : Set Karakter Player yang tidak terdaftar | diterima | Berhasil | Player akan di set random |
| 7 | Input karakter Menggerakan Player | Player bergerak | Berhasil | Player akan bergerak sesuai karakter yang terdaftar |
| 8 | Input karakter Menggerakan Player yang tidak terdaftar | Player diam | Berhasil | Player terdiam |
| 9 | Input karakter skill Player | Player menggunakan skill | Berhasil | Player menggunakan skill sesuai type player |
| 10 | Input karakter save Player | Menyimpan File | Berhasil | File tersimpan |
| 11 | Masuk Load Game | Mengambil objek asset dari file | Berhasil | Mendapat Objeck |
| 12 | Masuk Credit | Menampilkan daftar orang yang bertanggung jawab | Berhasil |  |
| 13 | Console : Option | Merubah fungsi karakter | Berhasil |  |

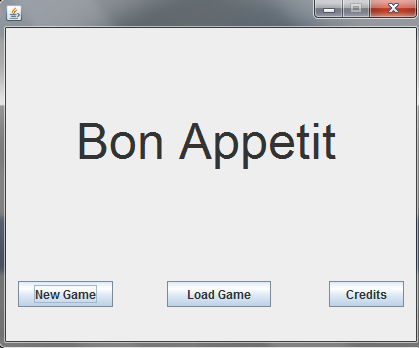
# Kebutuhan Minimum Aplikasi

Laptop / PC dengan Spesifikasi sebagai berikut :

* Processor Min Pentium 4
* RAM 512 MB
* VGA Intel HD 3000 atau sederajat

# User Manual

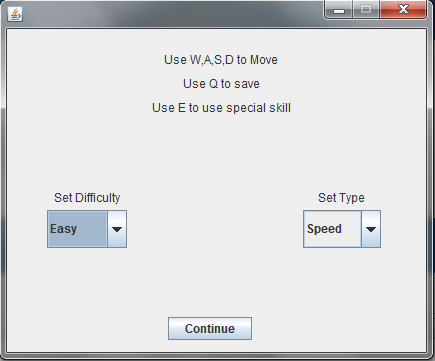
* 1. Layar Awal



Gambar 1 Gambar Layar Awal

Pada layar ini, terdapat satu judul *game* dan tiga buah tombol perintah. Tombol pertama adalah tombol New Game yang berguna untuk memulai sebuah game baru. Tombol kedua adalah tombol Load Game untuk mengakses *game* yang sudah pernah dimainkan dan disimpan di file sebelumnya. Sedangkan tombol terakhir adalah tombol Credits yaitu tombol untuk menampilkan daftar tokoh yang sudah berandil besar dalam suksesnya *game Bon Appétit* ini.

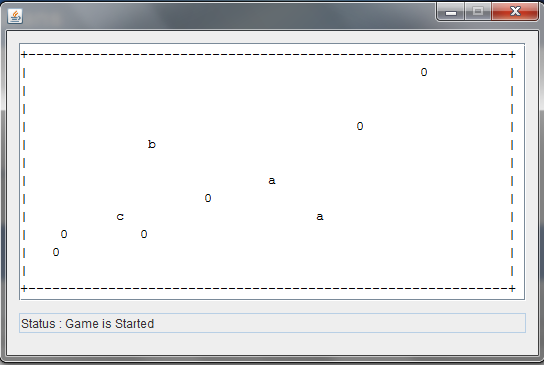
* 1. Layar New Game



Gambar 2 Layar New Game

Pada layar ini, pengguna dapat memilih tingkat kesulitan yang terdiri dari Easy, Medium, dan Hard. Tingkat kesulitan ini mempengaruhi jumlah serta level dari lawan yang muncul di layar bermain. Selain itu, pada layar ini juga pengguna dapat memilih tipe spesies ordo yang digunakan, spesies tersebut terdapat perbedaan skill. Pada layar ini juga terdapat tombol keyboard apa saja yang dapat digunakan pada *game* ini yaitu W, A, S, D yang berfungsi untuk bergerak ke atas, kiri, bawah, dan kanan. Terakhir adalah tombol *continue* untuk menuju layar *game*.

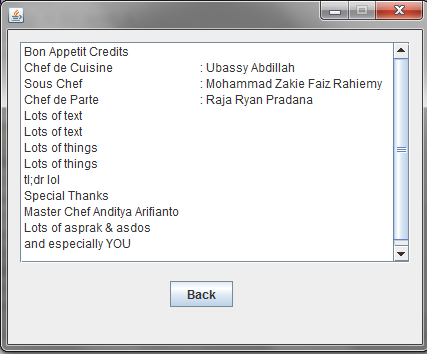
* 1. Layar Bermain



Gambar 3 Layar bermain

Pada layar ini, pengguna dapat bergerak untuk memakan *food* yang direpresentasikan dengan tanda 0. Karakter pengguna adalah karakter yang muncul di tengah layar saat layar bermain baru terbuka, sedangkan karakter lainnya adalah lawan yang harus dimakan jika levelnya dibawah level pemain. Tombol E berfungsi untuk mengaktifkan skill karakter. Pada layar ini pengguna dapat menekan tombol Q untuk menyimpan data agar dapat dibuka dikemudian hari.

* 1. Layar Credits



Gambar 4 Layar Credits

# Laporan Scrum

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NIM | Nama | Pembagian Tugas | | |
| Sprint 1 | Sprint 2 | Sprint 3 |
| 1301140302 | Raja Ryan Pradana | * Kelas Asset * Kelas Ordo | * Mockup Gui | * Membuat Dokumentasi * Read Me |
| 1301140122 | Dzakyta Afuzagani | * Membuat *artwork* | * Membuat *artwork* | * Membuat *artwork* |
| 1301144422 | Mohammad Zakie Faiz Rahiemy | * GameMap * Kelas Food * Kelas SpeciesA * Kelas SpeciesB * Kelas SpeciesC | * StrukturData | * User Manual |
| 1301148282 | Ubassy Abdillah | * Game | * Driver Console | * View Class * Controller Class * Driver Class |

# BurnDown Chart







# Penutup

*Game* ***Bon Appétit***adalah sebuah tugas besar yang kami kerjakan dengan asas *teamwork* dan penuh rasa kekeluargaan. Walaupun begitu, tidak menutup kemungkinan kami tetap melakukan kesalahan yang tidak kami sadari. Oleh karena itu kami harap seluruh pemangku kebijakan dari mata kuliah Pemrograman Berbasis Objek di semester genap tahun ajaran 2015/2016 ini dapat memberikan penilaian yang terbaik bagi kami.