LAPORAN TUGAS BESAR

IF2212 Pemrograman Berorientasi Objek

Plants vs Zombies

**Kelompok 17**

OOPSquad

Dipersiapkan oleh:

Ihsan Faishal Rasyid / 18217032

Shofura Salma / 18218035

Christovito Hidajat / 18218043

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Sekolah Teknik Elektro dan Informatika ITB** | Nomor Dokumen | Halaman |
| *IF2212-TB-1-01* | *24* |
| *31-03-2020* |

Daftar Isi

[1 Ringkasan 4](#_Toc36585426)

[2 Penjelasan Tambahan Spesifikasi Tugas 4](#_Toc36585427)

[2.1 Pengaturan Posisi Plants 4](#_Toc36585428)

[2.2 Menghilangkan Plants 4](#_Toc36585429)

[2.3 Tambahan Jenis Plants 4](#_Toc36585430)

[2.4 Kondisi Menang 4](#_Toc36585431)

[3 Program Utama dan Dokumentasi Kelas 5](#_Toc36585432)

[3.1 Dokumentasi Kelas Program 5](#_Toc36585433)

[3.2 Main Program 8](#_Toc36585434)

[3.3 Fitur-fitur yang Dikerjakan 9](#_Toc36585435)

[3.4 Screenshot Program 14](#_Toc36585436)

[*3.4.1* *Menu* 14](#_Toc36585437)

[*3.4.2* *Interface Backyard* 15](#_Toc36585438)

[*3.4.3* *Command “PLANT”* 16](#_Toc36585439)

[*3.4.4* *Tampilan Entitas* 17](#_Toc36585440)

[*3.4.5* *Command “SKIP”* 18](#_Toc36585441)

[*3.4.6* *Command “SHOVEL”* 19](#_Toc36585442)

[*3.4.7* *Game Over* 21](#_Toc36585443)

[*3.4.8* *Win* 21](#_Toc36585444)

[4 Algoritma Menarik 21](#_Toc36585445)

[4.1 Method zombieMove 21](#_Toc36585446)

[5 Pembagian Kerja dalam Kelompok 22](#_Toc36585447)

[6 Lampiran 23](#_Toc36585448)

[6.1 Deskripsi Tugas Besar 23](#_Toc36585449)

[6.2 Notulen Rapat 23](#_Toc36585450)

[6.3 Log Activity Anggota Kelompok 23](#_Toc36585451)

Daftar Tabel

[Tabel 1 Dokumentasi Kelas Program 6](#_Toc36585357)

[Tabel 2Fitur-fitur yang Dikerjakan 10](#_Toc36585358)

[Tabel 3Pembagian Kerja dalam Kelompok 23](#_Toc36585359)

[Tabel 4 Log Activity Kelompok 24](#_Toc36585360)

**Daftar Gambar**

[Gambar 1Inharitance Diagram 9](#_Toc36585490)

[Gambar 2Flowchart Main Program 10](#_Toc36585491)

[Gambar 3Screenshot Program Menu 15](#_Toc36585492)

[Gambar 4 Screenshot Program Interface Backyard 16](#_Toc36585493)

[Gambar 5 Screenshot Program Command PlANT 17](#_Toc36585494)

[Gambar 6 Screenshot Program Tampilan Entitas 18](#_Toc36585495)

[Gambar 7Screenshot Program Command SKIP 19](#_Toc36585496)

[Gambar 8 Screenshot Program Command SHOVEL 20](#_Toc36585497)

[Gambar 9 Screenshot Program Game Over 22](#_Toc36585498)

[Gambar 10 Screenshot Program Win 22](#_Toc36585499)

[Gambar 11 Algoritma Method zombieMove 23](#_Toc36585500)

# Ringkasan

Program *Plants vs Zombie* yang dikembangkan dalam rangka memenuhi tugas besar pemrograman berorientasi objek yang mengimplementasi prinsip-prinsip pada pemrograman berorientasi objek melingkupi penggunaan *inheritance, polymorphism*, abstract class / *interface, generic class,* beserta *driver* untuk menjalankan program tersebut.

Implementasi program menggunakan sejumlah kelas yang terdokumentasi pada dokumen ini, sejumlah fitur yang diimplementasikan ada yang bersifat wajib dan bersifat tambahan, permainan bersifat dua dimensi pada *command prompt.*

# Penjelasan Tambahan Spesifikasi Tugas

## Pengaturan Posisi Plants

Untuk setiap kali giliran, pemain dapat menanam dan mengatur posisi tumbuhan yang diinginkan pada *backyard* dengan memasukkan *command* PLANT. Pemain diminta menginput jenis tumbuhan beserta posisinya pada *backyard*.

## Menghilangkan Plants

Untuk setiap kali giliran, pemain dapat menggunakan fitur *remove* tumbuhan pada posisi tertentu di *backyard* seperti pada *game* aslinya dengan memasukkan *command* SHOVEL. Pemain diminta menginput posisi tumbuhan pada *backyard* yang ingin dihilangkan.

## Tambahan Jenis Plants

Terdapat dua jenis tumbuhan yang ditambahkan dari spesifikasi awal, yaitu tumbuhan *sunflower* dan *walnut*. Tumbuhan *sunflower* dapat menghasilkan *sunflower points* dalam periode waktu tertentu setelah ditanam. Tumbuhan *walnut* dapat menahan zombie dengan cara dimakan. Kedua tumbuhan ini tidak bisa bergerak dan menembak peluru.

## Kondisi Menang

Permainan dapat berakhir tidak hanya jika zombie sampai di ujung kiri layar, tetapi jika pemain dapat mengalahkan 10 zombie atau mendapat 100 poin, dalam artian pemain dinyatakan telah memenangkan permainan.

# Program Utama dan Dokumentasi Kelas

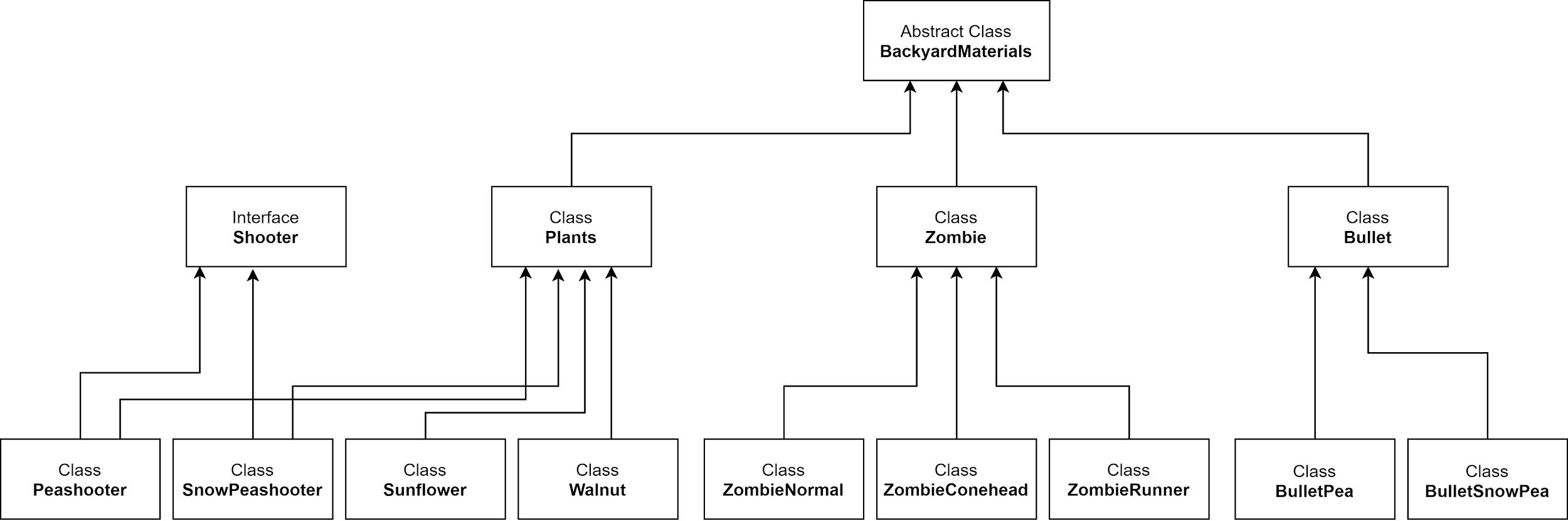
## Dokumentasi Kelas Program

Berikut adalah dokumentasi kelas yang dipakai pada program beserta deskripsinya.

Tabel 1 Dokumentasi Kelas Program

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Nama | Jenis | Deskripsi |
| 1. | Backyard | Class | Merupakan super class dari sejumlah kelas-kelas yang diimplementasi pada *Plants vs Zombie,* berfungsi sebagai landasan dalam mengembangkan kelas-kelas yang ada dalam program tersebut. Memiliki atribut row dan col yang bertipe integer, serta BackyardMaterials yang berbentuk ArrayList. Memiliki konstruktor Backyard dan sejumlah method yang melingkupi fungsi untuk menambahkan / menghilangkan BackyardMaterials, serta memperbarui Backyard (updateBackyard). |
| 2. | Game | Class | Merupakan sebuah kelas yang berisi command-command untuk melakukan instansiasi objek-objek yang digunakan dalam pelaksanaan program *Plant vs Zombie,* melakukan pemerolehan sejumlah nilai yang melingkupi *zombieList*, *plantList, bulletList,* *score, totalSun,* dan *gameTurn.* |
| 3. | BackyardMaterials | Abstract Class | Merupakan sebuah abstract class yang memiliki turunan dan berisikan method setter dan getter dari health, cost, speed, damage, symbol, position yang dimiliki oleh zombie dan plants, serta method yang akan diimplementasikan oleh kelas turunannya berupa perpindahan zombie, perpindahan peluru, dan produksi matahari oleh sunflower |
| 4. | Plants | Class | Kelas yang merupakan turunan dari kelas BackyardMaterials dan menjadi superclass yang memiliki subclass Sunflower, Walnut, Peashooter, dan SnowPeashooter. Memiliki konstruktor Plants dengan parameter row dan col yang bertipe integer, dan mengeset speed = 0. Memiliki method shoot untuk menembak dan produceSun untuk menghasilkan sun yang mengubah nilai totalSun. |
| 5. | Bullet | Class | Kelas yang merupakan turunan dari kelas BackyardMaterials dan menjadi superclass yang memiliki sub class BulletPea dan BulletSnowPea yang memiliki konstruktor Bullet dengan initial value symbol = “-”, health = 0, speed = 1, cost = 0. Memiliki method untuk memindahkan posisi peluru dengan nama BulletMove yang mengeset value col += speed. |
| 6. | Zombie | Class | Kelas yang merupakan turunan dari kelas BackyardMaterials dan menjadi super class yang memiliki sub class ZombieNormal, ZombieRunner, dan ZombieConehead, yang memilki atribut inisial counter, move, dan stoppedByAnother, serta konstruktor Zombie dengan parameter row dan col yang bertipe integer dengan cost = 0. Memiliki method zombieMove yang akan memindahkan posisi Zombie berdasarkan row dan col. |
| 7. | Peashooter | Class | Kelas yang merupakan turunan dari kelas Plants. Kelas ini berisi konstruktor dari peashooter berkaitan dengan cost = 100, health=50, dan damage=10, serta memiliki simbol “P”. Selain itu juga berisi method untuk menembak pada peashooter. |
| 8. | SnowPeashooter | Class | Kelas yang merupakan turunan dari kelas Plants . Kelas ini berisi konstruktor dari snow peashooter berkaitan dengan cost = 175, health=50, dan damage=20, serta memiliki simbol “I”. Selain itu juga berisi method untuk menembak pada snow peashooter. |
| 9. | Shooter | Interface | Interface yang memiliki standar method shoot untuk diimplementasikan pada kelas-kelas Plants yang memiliki method menembak (shoot). |
| 10. | Sunflower | Class | Kelas yang merupakan turunan dari kelas Plants, memiliki static atribut sunFlowerProductionCounter dengan initial value = 0, memiliki konstruktor dengan parameter row dan col yang bertipe integer, dan mengeset atribut symbol = “S”, health = 50, cost = 50, damage = 0. Memiliki method yang menghasilkan totalSun += 25 setiap 3 hitungan sunFlowerProductionCounter |
| 11. | Walnut | Class | Kelas yang merupakan turunan dari kelas Plants, memiliki konstruktor dengan parameter row dan col yang bertipe integer, dan mengeset atribut symbol = “W”, health = 150, cost = 50, damage = 0. |
| 12. | BulletPea | Class | Kelas yang merupakan turunan dari kelas Bullet, memiliki konstruktor dengan parameter row dan col yang bertipe integer, dan mengeset atribut damage = 10. |
| 13. | BulletSnowPea | Class | Kelas yang merupakan turunan dari kelas Bullet, memiliki konstruktor dengan parameter row dan col bertipe integer, dan mengeset atribut damage = 20. |
| 14. | ZombieNormal | Class | Kelas yang merupakan turunan dari kelas Zombie. Kelas ini berisi konstruktor dari zombie normal berkaitan dengan speed, health, dan damage dari zombie tersebut, dimana zombie normal memiliki health = 60, speed = 1, damage = 5, serta memiliki simbol “z”. Selain itu juga berisi method perpindahan untuk zombie normal tersebut. |
| 15. | ZombieConehead | Class | Kelas yang merupakan turunan dari kelas Zombie. Kelas ini berisi konstruktor dari zombie conehead berkaitan dengan speed, health, dan damage dari zombie tersebut, dimana zombie conehead memiliki health = 100, speed = 1, damage = 5, serta memiliki simbol “c”. Selain itu juga berisi method perpindahan untuk zombie conehead tersebut. |
| 16. | ZombieRunner | Class | Kelas yang merupakan turunan dari kelas Zombie. Kelas ini berisi konstruktor dari zombie runner berkaitan dengan speed, health, dan damage dari zombie tersebut, dimana zombie runner memiliki health = 60, speed = 2, damage = 5, serta memiliki simbol “r”. Selain itu juga berisi method perpindahan untuk zombie runner tersebut. |
| 17. | Main | Driver | Kelas yang berisi command sebagai driver yang memanggil kelas-kelas berdasarkan perannya pada pelaksanaan (execute) program Plants vs Zombies |

Berikut adalah *inheritance diagram* yang menyatakan keterhubungan *parent class* dan *child class* serta kelas dengan interface sesuai dokumentasi kelas pada tabel di atas.

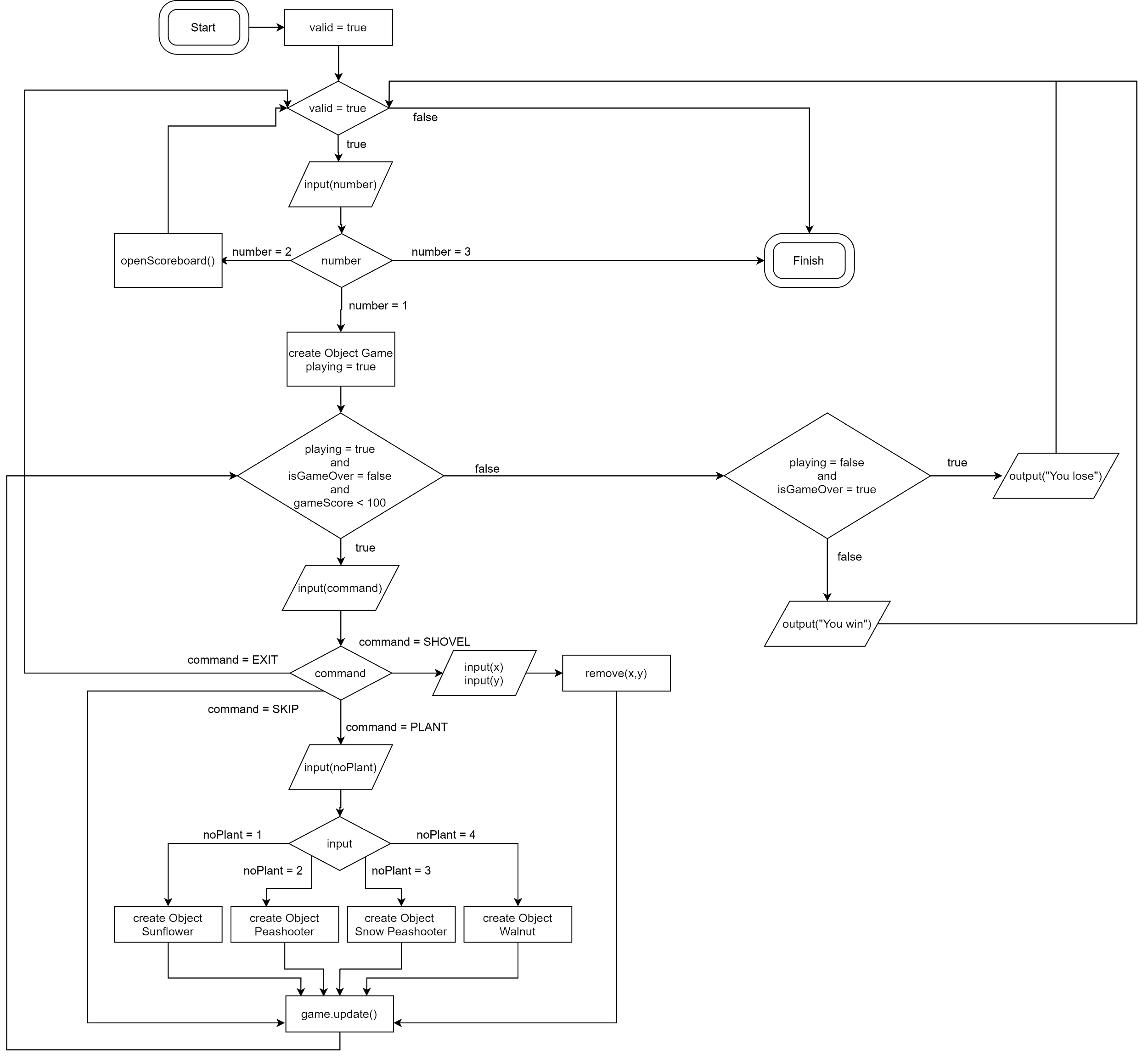


Gambar 1Inharitance Diagram

## Main Program

Awal mula permainan dimulai dengan memasukkan input berupa angka yang ingin dipilih yaitu 1, 2, 3, jika *inputan valid*, maka program akan bekerja. Jika yang diinputadalah angka 2 maka akan ditampilkan Scoreboardnya, jika yang diinput adalah angka 3, maka program akan berakhir dan keluar, jika yang di*input* adalah angka 1, maka akan berlanjut ke permainan utama, jika *inputan* tidak *valid,* maka akan dilakukan *loop* untuk mengisi *inputan.*

Jika yang di*input* adalah angka 1, maka permainan akan berlanjut ke permainan utama. Selama permainan berlangsung, tidak dalam keadaan *game over* ataupun menang, pemain akan diberikan pilihan perintah berupa SKIP, EXIT, SHOVEL, dan PLANT. Jika dijalankan perintah PLANT, maka akan keluar pilihan berupa tanaman-tanaman mana yang ingin dibeli dan ditanam untuk menghadapi serangan zombie. Jika dijalankan perintah SHOVEL, maka akan keluar pilihan berupa tanaman-tanaman mana yang ingin dihilangkan dari lahan pemain. Perintah EXIT dijalankan untuk berhenti dan keluar dari permainan.



Gambar 2Flowchart Main Program

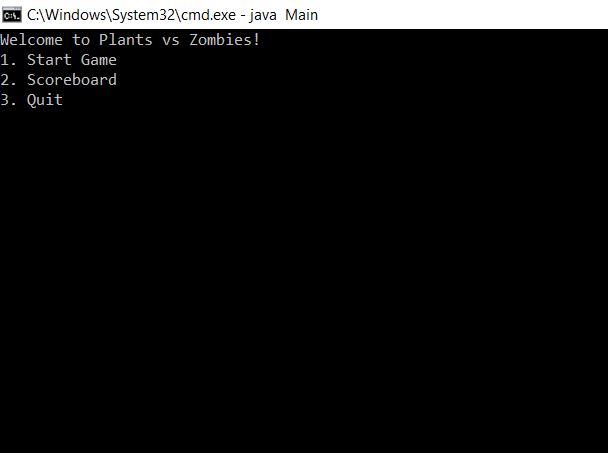
## Fitur-fitur yang Dikerjakan

Tabel 2Fitur-fitur yang Dikerjakan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Jenis Fitur | Nama Fitur | Hasil yang Diharapkan | Hasil yang Keluar |
| 1. | Wajib | Zombie muncul dari sisi kanan secara random | Zombie muncul secara random dari arah kanan dengan kemunculan random dari jenis zombienya | Sesuai |
| 2. | Wajib | Zombie bergerak ke arah kiri | Tiap jenis zombie bergerak dari arah kiri ke arah kanan | Sesuai |
| 3. | Wajib | Zombie dapat bergerak, kecepatan gerak zombie bergantung dengan tipenya | Tiap zombie dapat bergerak dan kecepatannya bergantung kepada tipenya, untuk ZombieNormal kecepatannya 1, ZombiConehead kecepatannya 1, dan ZombieRunner kecepatannya 2 (lebih cepat dari jenis lainnya). | Sesuai |
| 4. | Wajib | Zombie dapat menyerang tumbuhan yang ada di depannya, kekuatan serangan tiap zombie tergantung tipenya | Setiap serangan zombie memiliki damage yang mengurangi nilai health pada plant bergantung pada tipenya.  ZombieNormal damage 5, ZombiConehead damage 10, dan ZombieRunner damage 5. | Sesuai |
| 5. | Wajib | Zombie dapat melangkahi zombie lain sesuai kecepatannya | ZombieRunner dengan kecepatan 2 dapat membalap zombie lain dengan kecepatan 1 apabila berada di depannya | Sesuai |
| 6. | Wajib | Tumbuhan dapat dibeli pemain dengan menggunakan *sunflower points* | Tumbuhan yang ada dapat dibeli dengan poin yang sudah dikumpulkan melalui *sunflower points* yang didapatkan pemain dari *sunflower plants.* | Sesuai |
| 7. | Wajib | Tumbuhan tidak dapat bergerak | Tumbuhan tidak memiliki kecepatan (speed = 0) | Sesuai |
| 8. | Wajib | Tumbuhan menyerang dengan menembakkan sebuah peluru yang kecepatannya konstan untuk semua tipe tumbuhan | Tumbuhan Peashooter dan Snow Peashooter dapat mengeluarkan peluru dengan kecepatan konstan dan melakukan tembakan ke arah zombie. Ketika peluru mengenai zombie maupun tumbuhan lain di depannya, tampilan peluru akan hilang. | Sesuai. Namun dalam tampilan di layar, ketika peluru mengenai zombie dan berada pada petak *backyard* yang sama, simbol peluru (“-”) tersebut akan menimpa simbol zombie, dan hilang setelah zombie berjalan lagi. Hal ini juga berlaku ketika terdapat tumbuhan lain di depan peluru, maka simbol peluru akan menimpa simbol tumbuhan ketika berada pada petak yang sama |
| 9. | Wajib | Besar *damage* dari tiap peluru yang mengenai zombie bergantung dengan tipe tumbuhan | Setiap tipe tumbuhan menghasilkan bullet yang spesifik dengan tipe tumbuhan tersebut.  BulletSnowPea menghasilkan damage = 20. BulletPea menghasilkan damage = 10. | Sesuai |
| 10. | Wajib | Setiap periode giliran tertentu, pemain akan mendapatkan sejumlah *sunflower points* | Setiap pergantian giliran tertentu, pemain akan mendapatkan *sunflower points* sebesar 25 poin. | Sesuai |
| 11. | Wajib | Permainan berakhir ketika sebuah zombie sampai ke ujung kiri layar, dalam artian pemain kalah | Akhir dari permainan adalah saat Zombie sudah mencapai bagian kiri dengan selamat, lalu dikeluarkan pesan “The zombies ate your brain!You lose!” | Sesuai |
| 12. | Wajib | *Command* SKIP untuk melewatkan giliran pemain dan membiarkan entitas-entitas berjalan | Bila dilakukan perintah SKIP maka akan melewatkan giliran pemain, dengan zombie dan peluru tetap bergerak,*sunflower points*tetap bertambah | Sesuai |
| 13. | Tambahan | Saat dibeli, tumbuhan dapat diatur posisinya pada *backyard* | Pengaturan oleh *player* dimana posisi tumbuhan ingin diletakkan pada *backyard.* | Sesuai |
| 14. | Tambahan | Tumbuhan dapat di-*remove* dari *backyard* berdasarkan inputan posisinya | *Player* dapat menghapus tumbuhan dari *backyard* yang dimilikinya. | Sesuai |
| 15. | Tambahan | Setelah dibeli, tumbuhan jenis *sunflower* dapat menghasilkan *sunflower points* untuk setiap periode giliran tertentu | *Player* mendapatkan *sunflower points* pada setiap giliran permainan dijalankan. | Sesuai |
| 16. | Tambahan | Tumbuhan jenis *walnut* dapat menahan zombie untuk bergerak | Tumbuhan jenis *Walnut* memiliki kemampuan menahan zombie untuk tidak melewatinya sebelum *walnut* mati. | Sesuai. Namun, jika terdapat lebih dari satu zombie tepat di belakang tumbuhan *walnut* dan ada peluru mengenai zombie-zombie tersebut secara bersamaan, akan keluar pesan *error* yaitu IllegalStateException |
| 17. | Tambahan | Tumbuhan jenis *shooter* hanya akan menembakkan peluru jika di depannya ada zombie. Jika tidak ada zombie pada baris yang sama, tumbuhan tersebut tidak akan menembak | Tumbuhan jenis *shooter* melepaskan peluru hanya pada saat zombie terdeteksi di depannya pada baris yang sama. | Sesuai |
| 18. | Tambahan | Zombie dengan jenis *random* datang pada periode giliran tertentu | Jenis zombie yang masuk ke *backyard* dilakukan secara *random* | Sesuai |
| 19. | Tambahan | Setiap kali membunuh zombie akan menambah poin sebesar 10 | Setiap berhasil membunuh zombie *player* akan mendapat tambahan skor sebesar 10 dan dituliskan pada layar. | Sesuai |
| 20. | Tambahan | Permainan berakhir jika total poin mencapai 100, dalam artian pemain menang | Jika *player* berhasil mencapai 100 poin, maka permainan akan berakhir secara otomatis dan pemain dinyatakan menang. | Sesuai |
| 21. | Tambahan | <Load Score> |  |  |
| 22. | Tambahan | <High Score> |  |  |
| 23. | Tambahan | <Save Score> |  |  |

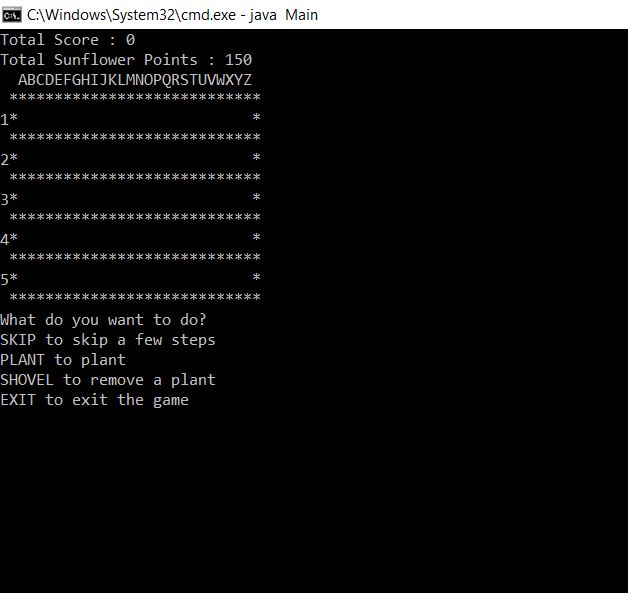
## Screenshot Program

### *Menu*

******

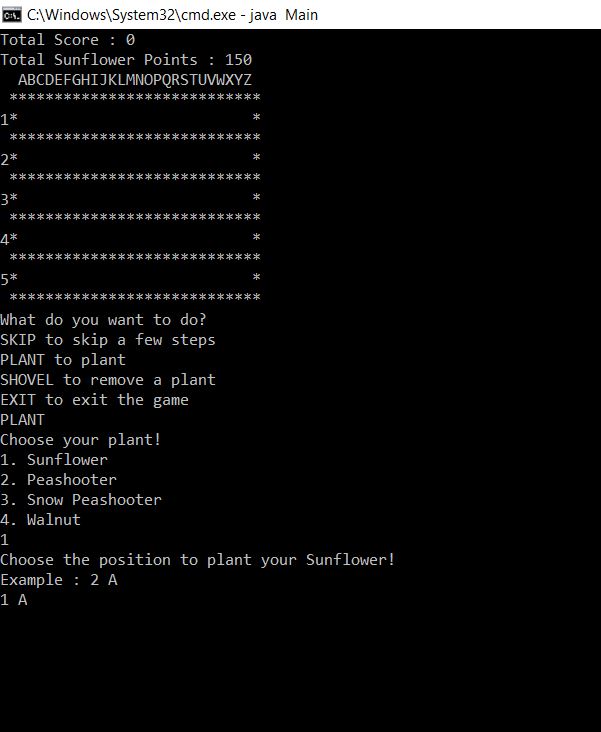
Gambar 3Screenshot Program Menu

### *Interface Backyard*

******

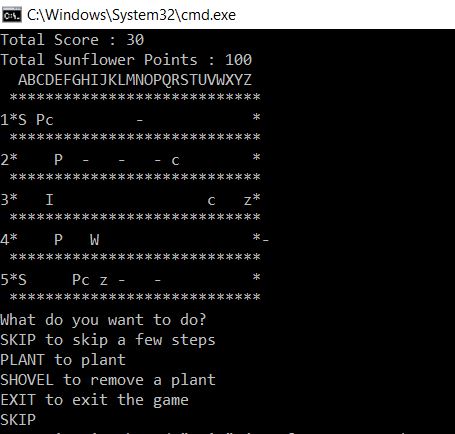
Gambar 4 Screenshot Program Interface Backyard

### *Command “PLANT”*

******

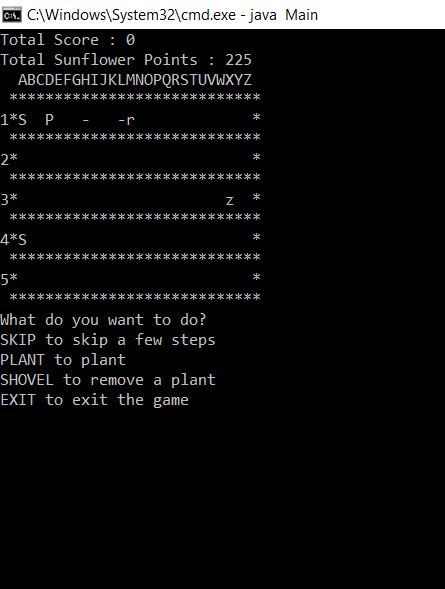
Gambar 5 Screenshot Program Command PlANT

### *Tampilan Entitas*

******

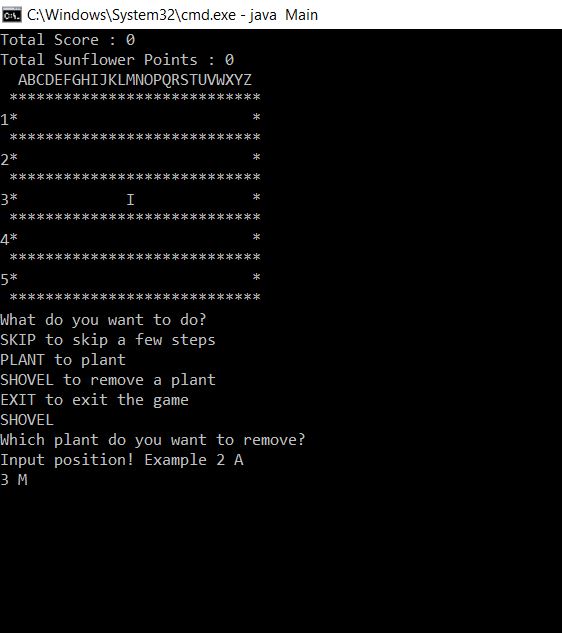
Gambar 6 Screenshot Program Tampilan Entitas

### *Command “SKIP”*

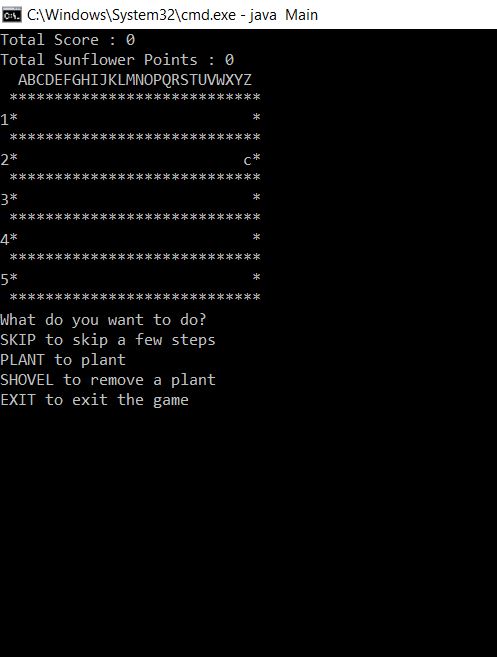
******

Gambar 7Screenshot Program Command SKIP

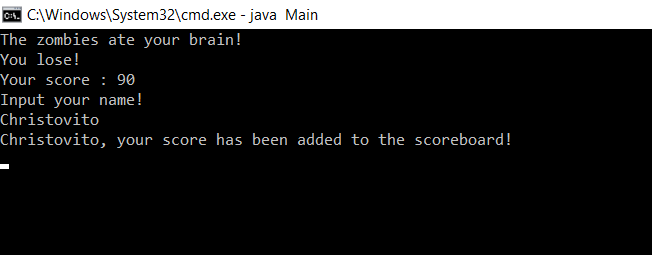
### *Command “SHOVEL”*

**

Gambar 8 Screenshot Program Command SHOVEL

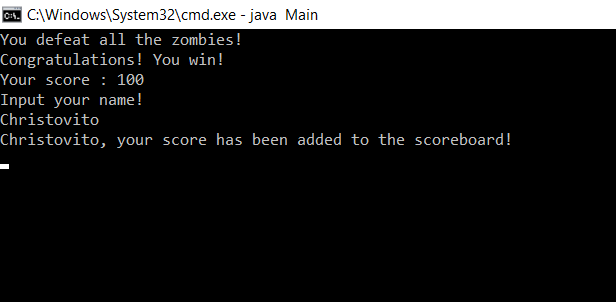
**

### *Game Over*

******

Gambar 9 Screenshot Program Game Over

### *Win*

******

Gambar 10 Screenshot Program Win

# Algoritma Menarik

## Method zombieMove

Berikut adalah algoritma *method* zombieMove, khususnya yang berada pada kelas zombieRunner.

|  |
| --- |
| public void zombieMove(Backyard backyard, Game game){  this.speed = 2;  for (Zombie zombie : game.zombieList) {  if (zombie == this){  for (Plants plant : game.plantList) {  if (zombie == this) {  if (this.getRow() == plant.getRow()   && this.getCol() == plant.getCol() + 1) {  this.speed = 0;  plant.health -= this.getDamage();  this.health -= plant.getDamage();  }  else if (this.getRow() == plant.getRow()   && this.getCol() == plant.getCol() + 2){  this.speed = 1;  }  }  }  if (this.getCol() - 2 < 0){  this.speed = 1;  }  }  }  backyard.removeBackyardMaterials(backyard,this.getRow(),this.getCol());  this.position.col -= this.speed;  } |

Gambar 11 Algoritma Method zombieMove

Algoritma ini menarik karena ketika zombie berada persis di depan tumbuhan, maka selama *health* zombie belum habis, zombie akan “memakan” tumbuhan tersebut dan jika tumbuhan mati, maka zombie dapat bergerak kembali. Pada algoritma di atas, terlihat bahwa atribut *speed* untuk zombie berubah-ubah tergantung dari kondisi yang dipenuhi.

Namun khusus untuk zombieRunner, harus di-*set* *speed* menyesuaikan setiap kondisi yang ada, sehingga kondisi jika ada tumbuhan di depannya yang berbeda 1 petak dan berbeda 2 petak harus dibedakan. Selain itu, ketika zombie hampir mencapai ujung kiri layar, harus diperhatikan bahwa posisi zombie tidak boleh kurang dari 0, karena akan keluar *exception* ArrayOutOfIndexBoundsException. Maka jika zombie berada minimal 2 petak dari ujung kiri layar, kecepatannya di-*set* menjadi 1.

# Pembagian Kerja dalam Kelompok

Pengerjaan tugas besar pada kelompok kami dilakukan dengan pembagian kerja sebagai berikut.

Tabel 3Pembagian Kerja dalam Kelompok

|  |  |  |
| --- | --- | --- |
| NIM | Nama | Bagian yang Dikerjakan |
| 18217032 | Ihsan Faishal Rasyid | 3 jenis kelas zombie (ZombieNormal, ZombieConehead, ZombieRunner) + 2 jenis kelas bullet (BulletPea & BulletSnowPea) |
| 18218035 | Shofura Salma | 4 jenis kelas plants (Peashooter, SnowPeashooter, Sunflower, Walnut) + 1 interface shooter |
| 18218043 | Christovito Hidajat | kelas Backyard + kelas Game + kelas abstrak Backyard Materials + kelas parent Plants + kelas parent Bullets + kelas parent Zombies + kelas Main |

# Lampiran

## Deskripsi Tugas Besar

Pengerjaan tugas besar *Plants vs Zombie* merupakan tugas untuk mengimplementasikan konsep-konsep pemrograman berorientasi objek yang melingkupi penggunaan *inheritance, polymorphism*, *abstract class* / *interface, generic class,* beserta *driver* untuk menjalankan program tersebut.

Secara umum dibagi menjadi beberapa kelas dasar berupa kelas untuk game, kelas backyard, kelas zombie, kelas plants, kelas bullet, untuk melakukan instansiasi yang menghasilkan objek yang direalisasikan pada driver yaitu kelas Main. Prinsip-prinsip *inheritance, generic class, polymorphism*, *abstract class / interface* diterapkan untuk menghasilkan variasi dari objek-objek berdasarkan perbedaan fungsi dan peranannya.

Pengerjaan tugas besar dibagi dua tahap *milestone,* pada tahap *milestone* pertama interaksi dilakukan tanpa implementasi GUI, sedangkan pada tahap *milestone* kedua interaksi dilakukan dengan implementasi GUI menggunakan penerapan *package java swing* pada kelas yang dikembangkan.

## Notulen Rapat

Kegiatan diskusi atau rapat dilakukan secara *online* dan diperoleh catatan/notulensi diskusi sebagai berikut.

* **Rapat 1**

Pada rapat pertama dipahami spesifikasi tugas besar dan dilakukan pembagian tugas terkait penyusunan kelas-kelas pada tugas besar kelompok.

* **Rapat 2**

Pada rapat kedua didiskusikan pemecahan masalah generic class, cara implementasi shoot pada program, pengembangan fitur opsional (score), serta penyusunan laporan dokumentasi tugas besar.

## Log Activity Anggota Kelompok

Berikut adalah log dari segala aktivitas yang dilakukan oleh tiap anggota kelompok selama pengerjaan tugas besar.

Tabel 4 Log Activity Kelompok

|  |  |  |
| --- | --- | --- |
| No | Nama | Kegiatan |
| 1 | Ihsan Faishal Rasyid | Membuat git repository kelompok, Mengerjakan bagian 3 jenis kelas zombie (ZombieNormal, ZombieConehead, ZombieRunner) + 2 jenis kelas bullet (BulletPea & BulletSnowPea) pada program, Menyusun laporan tugas besar, Melakukan formatting laporan tugas besar |
| 2 | Shofura Salma | Mengerjakan bagian 4 jenis kelas plants (Peashooter, SnowPeashooter, Sunflower, Walnut) + 1 interface shooter, Menyusun laporan tugas besar |
| 3 | Christovito Hidajat | Mengerjakan bagian kelas Backyard + kelas Game + kelas abstrak Backyard Materials + kelas parent Plants + kelas parent Bullets + kelas parent Zombies + kelas Main, Menyusun template laporan tugas besar, Menyusun laporan tugas besar |