

# Spesifikasi Tugas Besar 1: Purry Leveling

## IF2010 Pemrograman Berorientasi Objek

Version History:

- Revisi 1: 12/04/2025 - Revisi Link Github Classroom dan Format Nama Tim
- Revisi 2: 18/04/2025 - Penambahan atribut stats pada unit (implisit juga pada mob) dan jadwal asistensi
- Revisi 3: 22/04/2025 - Penambahan library GUI yang diperbolehkan dan revisi Shop restock
- Revisi 4: 24/04/2025 - Penghapusan atribut *level* pada *item*
- Revisi 5: 1/05/2025 - Penambahan waktu penggeraan dan pengubahan waktu demo (Senin tidak dianjurkan, tetapi boleh diisi)

# Daftar Isi

<b>Daftar Isi</b>	<b>2</b>
<b>Deskripsi Persoalan</b>	<b>3</b>
<b>Mekanisme</b>	<b>5</b>
Unit	5
Character	5
Skill dan Skill Path	6
Effect	8
Dungeon	9
Mobs	12
Shop	14
Items	16
Damage	17
Inventory	17
Save/Load	18
<b>Spesifikasi Sistem</b>	<b>20</b>
Ketentuan Umum	20
Ketentuan Teknis	20
<b>Bonus</b>	<b>21</b>
1. Quest System	21
2. Cheat	22
3. Antarmuka Pengguna Grafis (APG)/Graphical User Interface	23
4. Kreativitas	23
<b>Konfigurasi</b>	<b>24</b>
<b>Panduan Pengerjaan</b>	<b>26</b>
<b>Panduan Laporan</b>	<b>27</b>
<b>Asistensi</b>	<b>27</b>
<b>QnA</b>	<b>28</b>

## Deskripsi Persoalan



Sumber: Gambar dibuat menggunakan DALL-E, model AI oleh OpenAI

"Wiuuu Wiuuu 🚨," suara sirene menggema memenuhi markas OWCA. Monitor-monitor di seluruh ruangan menyala terang, menampilkan pesan darurat yang tidak pernah dibayangkan sebelumnya: "**DIMENSI DUNGEON TELAH TERBUKA!**".

Para agen OWCA segera berkumpul, kebingungan bercampur panik. Di tengah kekacauan, Mr. Monogram muncul di layar utama dengan wajah seriusnya. "Agen-agen OWCA, keadaan darurat! Sebuah dimensi misterius penuh monster telah muncul tepat di bawah markas kita. Dungeon ini bukan sembarang tempat, melainkan tempat yang berisi tantangan dan makhluk-makhluk ganas yang bisa menghancurkan dunia kita jika tidak dihentikan!"

Setelah diselidiki, ternyata **Cyborg**, musuh bebuyutan OWCA, menggunakan perangkat teknologi canggih untuk membuka gerbang menuju dimensi dungeon ini. Melihat ancaman besar tersebut, Mr. Monogram menugaskan para agen terbaiknya untuk memasuki dungeon, mengalahkan monster-monster yang bersembunyi di tiap chamber, dan menutup gerbang dimensi selamanya.

Namun, para agen OWCA membutuhkan persiapan matang. Mereka harus memilih karakter yang tepat sesuai gaya bertarung mereka: Fighter yang gagah dengan kekuatan fisik luar biasa, Mage yang pintar mengendalikan sihir, Assassin yang lincah dan gesit, Necromancer yang misterius dengan kekuatan gelapnya, atau Berserker yang mampu menerobos musuh dengan mudah.

Selain itu, agen-agen juga harus bijak dalam memilih dungeon untuk dijelajahi, karena dungeon memiliki tingkat kesulitan berbeda—mulai dari Rank E yang paling mudah hingga Rank S yang menakutkan. Di setiap chamber, tantangan makin berat hingga menghadapi Boss yang mengerikan di akhir dungeon. Belum lagi ancaman Double Dungeon yang muncul secara tiba-tiba, meningkatkan kesulitan serta hadiah secara drastis!

Untuk mendukung misi berbahaya ini, OWCA telah menyiapkan sebuah shop khusus dengan item-item penting seperti Weapon, Armor, Potion, dan Pendant, yang dapat membantu agen mengalahkan monster di dungeon. Item dengan rarity tinggi pun tersedia, namun hanya agen yang berhasil kembali dengan selamat dari dungeon yang dapat memperoleh gold dan membeli item-item tersebut.

Sayangnya, kemampuan OWCA terbatas dalam hal pemrograman sistem dungeon dan shop ini. Mr. Monogram kembali meminta bantuan Purry yang, tentu saja, tidak bisa memprogram kode. Oleh karena itu, bantu OWCA dan Purry membuat sistem Dungeon OWCA ini agar para agen dapat berjuang menyelamatkan dunia dari ancaman mengerikan yang diciptakan oleh **Cyborg**!

# Mekanisme

## Unit

Unit adalah kelas dasar yang merepresentasikan semua entitas hidup yang dapat berpartisipasi dalam pertarungan. Terdiri dari Character dan Mob. Character dikendalikan oleh pemain dan dapat memiliki stats, skill, skill path, dan inventory. Mob, yaitu musuh yang biasanya dikendalikan oleh sistem, muncul dalam dungeon, dan dapat memberikan loot saat dikalahkan. Mekanisme pertarungan yang akan dilakukan adalah *turn-based*.

Setiap unit memiliki atribut dan fungsi berikut.

Atribut	Fungsi
Name	attack(Unit)
Current Health	takeDamage(int)
Max Health	heal(int)
Health Regen	restoreMana(int)
Current Mana	useSkill(Skill)
Max Mana	addSkill(Skill)
Mana Regen	removeSkill(Skill)
Attack Damage	addEffect(Effect)
Skills	removeEffect(Effect)
Active Effects	reset()
Stats	

## Character

Pada saat memulai permainan, setiap pemain memilih satu karakter. Setiap karakter memiliki skill dan stats yang berbeda-beda. Setiap karakter memiliki atribut berikut.

Atribut
Level
Exp
Mastery
Gold

## Skill dan Skill Path

Setiap karakter minimal memiliki 1 skill dengan ketentuan skill sebagai berikut.

1. Nama
2. Mana Cost (skill bisa tidak membutuhkan mana)
3. Mastery Cost
4. Damage
5. Effect Chance (bisa bernilai 1 apabila pasti terjadi)
6. Effects (skill bisa tidak memiliki effect)

Setiap skill memiliki skill path-nya masing-masing. Sebagai contoh berikut.

Character A memiliki skill fireball (base skill), skill fireball ini memiliki skill path sebagai berikut.

- Name: Mastery 1
- Skills: [fireball 2, flamethrower]

Fireball 2 memiliki mastery cost 5 dan flamethrower memiliki mastery cost 3. Mastery cost awal dari karakter adalah 0 dan akan bertambah saat naik level (mirip dengan exp). Misal, saat naik ke level 2, karakter mendapatkan mastery cost sebesar 3. Karakter tersebut bisa “belajar” skill baru, yakni flamethrower yang membutuhkan mastery cost sebesar 3. Maka, character dapat unlock skill tersebut dan menambahkannya ke atribut skills. Mastery cost berkurang sebesar 3.

Jika suatu child skill path telah di-“pelajari” semua, maka skill parent-nya akan hilang. Misal, fireball 2 dan flamethrower sudah di-“pelajari” maka fireball (base skill) akan hilang.

Setiap skill tidak harus memiliki skill path. **Buatlah skill tree** yang membuat pemain tidak bisa menabung mastery cost untuk mendapatkan skill yang paling bagus. Namun, bertahap dengan membuat skema skill tree seperti contoh di bawah ini. Kalian wajib membuat setidaknya 3 node skill path dengan minimal 1 percabangan untuk setiap skill pada karakter.



Setiap karakter memiliki tipe yang berbeda-beda. Setiap tipe karakter memiliki atribut yang harus diturunkan dari kelas Character. Implementasi dari setiap fungsi dibebaskan yang penting sesuai dengan spesifikasi berikut.

Tipe	Atribut	Fungsi
Fighter	<ul style="list-style-type: none"> <li>blockChance</li> </ul>	<ul style="list-style-type: none"> <li>takeDamage -&gt; ada blockChance terhadap attack yang masuk</li> <li>useSkill -&gt; menambahkan nyawa berdasarkan strength</li> <li>levelUp -&gt; menambahkan semua stat, namun strength paling besar</li> </ul>
Mage	<ul style="list-style-type: none"> <li>extraMana</li> </ul>	<ul style="list-style-type: none"> <li>attack -&gt; setiap attack menambahkan mana sebesar extraMana</li> <li>useSkill -&gt; mengurangi mana cost dan menambahkan magic damage berdasarkan intelligence</li> <li>levelUp-&gt; menambahkan semua stat, namun intelligence paling besar</li> </ul>

Assassin	<ul style="list-style-type: none"> <li>criticalChance</li> <li>criticalMultiplier</li> </ul>	<ul style="list-style-type: none"> <li>attack -&gt; tambahan critical</li> <li>takeDamage -&gt; miss chance berdasarkan agility</li> <li>levelUp -&gt; menambahkan semua stat, namun agility paling besar</li> </ul>
Necromancer	<ul style="list-style-type: none"> <li>summon</li> <li>summonChance</li> </ul>	<ul style="list-style-type: none"> <li>attack -&gt; summon anak buah, next turn anak buah memberikan damage berdasarkan intelligence</li> <li>useSkill -&gt; lifesteal nyawa musuh berdasarkan intelligence</li> <li>levelUp -&gt; menambahkan semua stat, namun intelligence paling besar</li> <li>reset -&gt; reset summon, setelah pertarungan selesai</li> </ul>
Berserker	<ul style="list-style-type: none"> <li>rageMultiplier</li> </ul>	<ul style="list-style-type: none"> <li>attack -&gt; menambahkan multiplier</li> <li>useSkill -&gt; menambahkan multiplier</li> <li>levelUp -&gt; menambahkan semua stat, namun strength paling besar</li> <li>reset -&gt; reset multiplier, setelah pertarungan selesai</li> </ul>

## Effect

Setiap unit dapat terkena effect. Effect dapat berasal dari item atau skill. Effect terdiri dari beberapa atribut dan fungsi sebagai berikut.

Atribut	Fungsi
Name	apply(Unit)
Description	remove(Unit)
Duration	
RemainingDuration	

Silahkan berkreasi untuk membuat effect sendiri **minimal 5 effect item dan 5 effect skill untuk pemain!**

**Note:** apply & remove effect akan mengubah suatu nilai atribut pada unit. Buat mekanisme yang mudah pada apply & remove!

**Kata kunci:** virtual

## Dungeon



Sumber: [https://static.wikia.nocookie.net/solo-leveling/images/0/00/Anime\\_Episode\\_8\\_picture\\_63.png](https://static.wikia.nocookie.net/solo-leveling/images/0/00/Anime_Episode_8_picture_63.png)

Agen OWCA dapat memasuki sebuah dungeon untuk berlatih, mendapatkan item, menambah level, dan kebutuhan lainnya. Dungeon memiliki beberapa tingkat yang akan menentukan seberapa sulit musuh yang akan dilawan di dalamnya. Beberapa tipe dungeon dapat memiliki *chambers*, yakni semacam *stage* tertentu yang di dalamnya berisi tipe mob atau tantangan yang spesifik pada ruangan tersebut. Pada setiap dungeon, selain chamber terakhir akan melawan Basic Mob secara acak, sedangkan pada chamber terakhir akan melawan Boss Mob secara acak.

Tipe Dungeon	Spesifikasi
Rank S	<ul style="list-style-type: none"><li>Minimal level karakter: 50+</li><li>Mob level range: 50-100</li></ul>

	<ul style="list-style-type: none"> <li>• Biaya masuk 1000 gold</li> <li>• Memiliki 6 chamber</li> </ul>
Rank A	<ul style="list-style-type: none"> <li>• Minimal level karakter: 35+</li> <li>• Mob level range: 30-70</li> <li>• Biaya masuk 500 gold</li> <li>• Memiliki 5 chamber</li> </ul>
Rank B	<ul style="list-style-type: none"> <li>• Minimal level karakter: 25+</li> <li>• Mob level range: 20-50</li> <li>• Biaya masuk 300 gold</li> <li>• Memiliki 4 chamber</li> </ul>
Rank C	<ul style="list-style-type: none"> <li>• Minimal level karakter: 20+</li> <li>• Mob level range: 10-35</li> <li>• Biaya masuk 150 gold</li> <li>• Memiliki 3 chamber</li> </ul>
Rank D	<ul style="list-style-type: none"> <li>• Minimal level karakter: 10+</li> <li>• Mob level range: 0-25</li> <li>• Biaya masuk 0 gold</li> <li>• Memiliki 2 chamber</li> </ul>
Rank E	<ul style="list-style-type: none"> <li>• Minimal level karakter: 0+</li> <li>• Mob level range: 0-10</li> <li>• Biaya masuk 0 gold</li> <li>• Memiliki 1 chamber</li> </ul>
Special	<ul style="list-style-type: none"> <li>• Tidak ada minimum level karakter</li> <li>• Mob level range: 0-?</li> <li>• Tidak ada biaya masuk</li> <li>• Memiliki 100 chamber bertingkat*</li> </ul>

\* (random generated mob, tetapi setiap naik chamber tingkat kesulitannya bertambah)

## Implementasi Dungeon

### 1. Chamber System

Setiap chamber memiliki sekelompok monster (mob) yang harus dikalahkan. Setelah semua monster dalam chamber dikalahkan, pemain dapat melanjutkan ke chamber berikutnya. Chamber terakhir di setiap dungeon (kecuali Rank D dan E) harus berisi Boss.

### 2. Difficulty Scaling

Monster dalam dungeon dengan rank lebih tinggi memiliki stats yang lebih tinggi.  
Scaling factor: E (1x), D (1.5x), C (2x), B (3x), A (4x), S (5x).

### 3. Reward System

Setiap chamber memberikan rewards berupa experience points (EXP) dan gold. Selesai menaklukan dungeon, pemain mendapatkan bonus item dengan rarity dapat sesuai rank dungeon atau menggunakan RNG (rarity dibebaskan).

### 4. Double Dungeon System

Secara random, terdapat kesempatan dungeon menjadi "Double Dungeon". Double Dungeon memiliki tingkat kesulitan yang lebih tinggi (2x normal) namun rewards juga lebih besar (3x normal).

### 5. Penalti Dungeon

Ketika pemain gagal menyelesaikan dungeon tersebut karena mati, maka semua item yang didapatkan selama permainan **TIDAK** masuk ke dalam inventori (**KECUALI**, Special Dungeon), dan stat pemain akan kembali semua seperti semula sebelum pemain memasuki dungeon (semua buff, effect, dsb) hilang. Juga, akan ada penalti yakni pengurangan gold dan exp.

## Mobs

Pada setiap Chamber Dungeon, pemain akan melawan Mob tertentu sesuai dengan konfigurasi rank Dungeon. Terdapat dua jenis Mob, yaitu Basic Mob dan Boss Mob. Setiap Mob dapat memiliki berbagai macam skill **dan juga memiliki atribut Stats**. Masing-masing skill memiliki probabilitas atau kondisi tertentu pada saat dipakai. Skill berikut umum dimiliki Mob atau Boss Mob:

1. Heal

Memiliki properti basic heal value yang akan dikalikan dengan INT akhir dari Mob. Sesuaikan properti basic heal value ini untuk setiap Mob dengan kreasi kalian dan jumlah penggunaan mananya. Hasil dari perkalian merupakan jumlah health yang akan ditambahkan ke Mob pada saat heal. Probability untuk skill ini terpakai adalah 0.05.

2. Rage

**Setiap Boss Mob** memiliki skill rage yang **otomatis** akan aktif apabila **health** dari Mob hanya tersisa kurang atau sama dengan 20%. Skill ini hanya akan terpakai satu kali selama battle dan akan aktif hingga battle selesai. Selama skill ini aktif, seluruh stat dari Mob akan menjadi **1.5x lipat**.

Untuk seluruh Mob, **atribut utama dapat kalian kreasikan** dan sesuaikan untuk keseimbangan permainan, termasuk atribut skillnya. Untuk setiap skill, silahkan **tentukan penggunaan mananya** juga. Apabila mana tidak cukup untuk menggunakan skillnya, maka skill tersebut tidak akan terpakai. Berikut adalah daftar basic Mob.

Mob	Skill
Slime	Acidic Corrosion Probability: 0.2  Slime menyemburkan cairan asam pekat ke target, menyebabkan damage over time (kerusakan berkelanjutan) selama 3 turn. Properti: basic damage (dikalikan int)
Goblin	Sneaky Stab Probability: 0.2

	Goblin menghilang sesaat dalam kepulan asap atau bayangan, lalu muncul tiba-tiba di belakang target untuk memberikan serangan kejutan dengan bonus critical damage, memberikan damage 2x dari attack biasa.
Skeleton	<p>Bone Shard Probability: 0.25</p> <p>Skeleton melepaskan pecahan tulang tajam dari tubuhnya ke arah pemain. Serangan ini memberikan damage fisik dan menyebabkan effect 'Brittle' (Rapuh), yang membuat pemain menerima damage fisik sedikit lebih besar dari serangan berikutnya.</p>
Orc	<p>Brutal Strike Probability: 0.15</p> <p>Orc mengayunkan senjatanya dengan sekuat tenaga ke arah pemain. Serangan ini memberikan damage fisik yang signifikan (1.5x attack biasa) dalam satu pukulan kuat dan berpeluang 0.5 untuk memberikan stun selama 1 turn sehingga Orc akan menjalankan turn kembali.</p>

Berikut adalah daftar Boss Mob

Mob	Skill
Ogre	<p>Pulverize Probability: 0.3</p> <p>Ogre mengangkat senjata besarnya dan menghantamkannya ke pemain. Serangan ini memberikan damage fisik yang sangat besar (2.5x attack biasa) dan memiliki peluang tinggi (0.9 effect chance) untuk menyebabkan efek stun selama 1 turn.</p>
Dark Knight	<p>Soul Siphon Probability: 0.35</p> <p>Dark Knight menyerang pemain dengan serangan gelap yang tidak hanya memberikan damage sihir berdasarkan INT (tentukan pengalinya sendiri), tetapi juga memiliki kemungkinan untuk menyerap sebagian kecil HP pemain pada turn berikutnya (effect, 0.8 chance) untuk memulihkan HP Dark Knight itu sendiri (0.3x damage yang diberikan).</p>
Demon Lord	<p>Infernal Curse Probability: 0.15</p>

	Demon Lord merapalkan kutukan kuat pada pemain. Kutukan ini tidak memberikan damage langsung, tetapi secara signifikan mengurangi attack power dan defense power pemain untuk durasi yang cukup lama (5 turn), membuatnya lebih rentan. Apabila musuh masih terkena effect dari skill ini, menggunakan skill yang sama hanya akan memperbarui durasi effect (kembali menjadi 5 turn)
Lich	<p>Withering Touch Probability: 0.2</p> <p>Lich mengalirkan energi nekrotik ke arah pemain. Serangan ini memberikan damage sihir gelap secara terus-menerus (menggunakan INT) selama 4 turn dan juga mengurangi mana pemain.</p>

Untuk setiap Mob, setelah Mob dikalahkan, Mob tersebut akan memberikan sejumlah EXP yang berbeda-beda (silahkan tentukan sendiri) dan kemungkinan untuk Drop Loot berupa item yang sesuai dengan konfigurasi.

## Shop



Sumber: <https://images.mmorpg.com/images/content/Images/62021/WoW-Cash-Shop.jpg>

Shop merupakan tempat dimana pemain dapat membeli dan menjual berbagai item. Implementasi Shop harus mencakup fitur-fitur berikut:

## Implementasi Shop

### 1. Shop Restocking

Setiap kali pemain kembali dari dungeon, shop akan melakukan restock. Probabilitas mendapatkan item dengan rarity tinggi lebih kecil. Probabilities: Common (60%), Uncommon (25%), Rare (10%), Epic (3%), Legendary (1.5%), Mythic (0.5%).

### 2. Item Rarity and Pricing

Pemain dapat membeli item dari shop dengan menggunakan gold. Pemain dapat menjual item ke shop untuk mendapatkan gold. Harga jual item selalu lebih rendah dari harga beli (sekitar 70% dari harga asli).

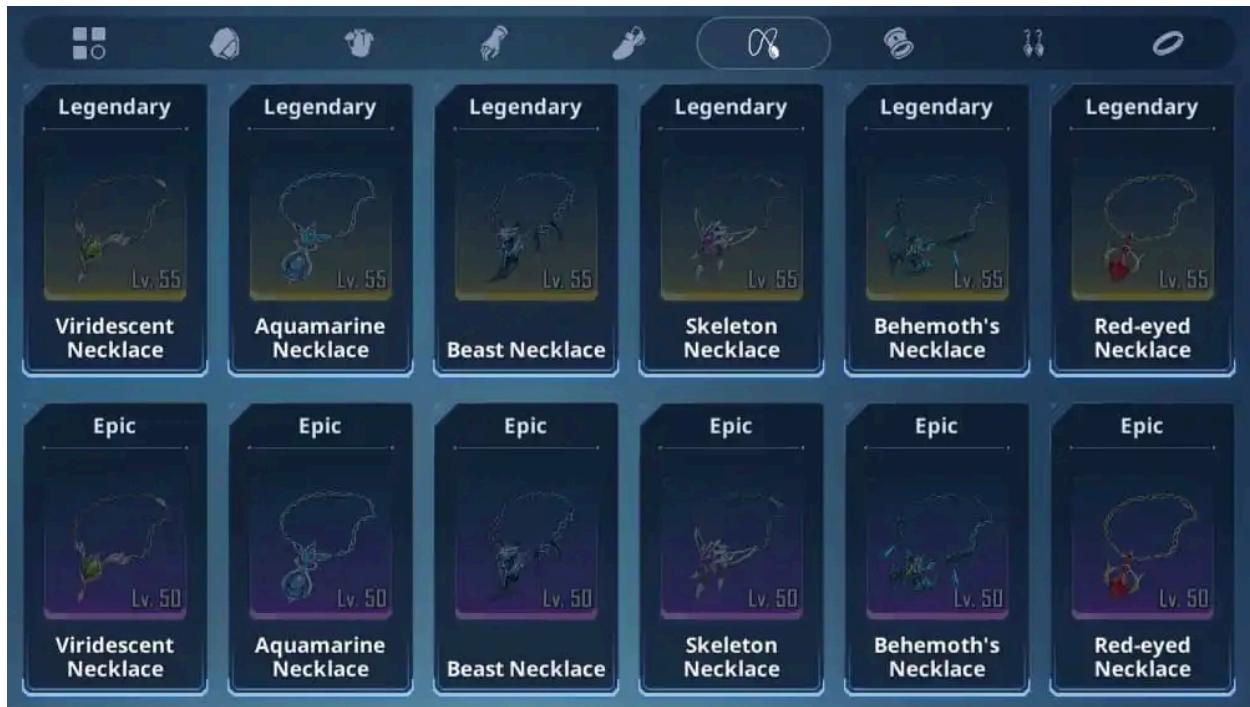
### 3. Item Buying and Selling

Item dengan rarity lebih tinggi memiliki harga lebih mahal. Pricing factor: Common (1x), Uncommon (2x), Rare (5x), Epic (10x), Legendary (20x), Mythic (50x).

### 4. Item Grouping

Shop wajib melakukan grouping berdasarkan tipe item untuk memudahkan navigasi pembelian, navigasi dan tampilan dibebaskan implementasinya. Contoh: Potion A, Potion B, dan Potion C mereka semua termasuk dalam tab 'Potion' karena tipe item tersebut adalah Potion.

## Items



Sumber: <https://i0.wp.com/gamingonphone.com/wp-content/uploads/2024/04/Solo-leveling-arise-artifact-set-accessories.jpg>

### Implementasi Items:

#### 1. Item Types (Minimal)

- Weapon: Memberikan damage base dan bonus untuk karakter saat bertarung.
- Armor: Memberikan defense bonus untuk mengurangi damage yang diterima karakter.
- Potion: Item consumable yang memberikan efek instan seperti heal atau buff sementara.
- Pendant: Memberikan buff pasif untuk karakter selama di-equip.

#### 2. Item Rarity

- Common (E): Stats dasar, tanpa efek khusus.
- Uncommon (D): Stats +20% dari base, kemungkinan kecil memiliki satu efek.
- Rare (C): Stats +50% dari base, biasanya memiliki satu efek.
- Epic (B): Stats +100% dari base, selalu memiliki setidaknya satu efek.
- Legendary (A): Stats +200% dari base, selalu memiliki dua efek.

- Mythic (S): Stats +500% dari base, selalu memiliki tiga efek.

### 3. Item Stats (Lihat bagian Efek)

- Offensive Effects: Critical Strike, Bleed, Burn, Poison, etc.
- Defensive Effects: Damage Reduction, Reflect, Immunity, etc.
- Utility Effects: Speed Boost, Mana Regeneration, Health Regeneration, etc.

### 4. Item Stats Scaling

- Item stats (damage, defense, effect) harus scale berdasarkan rarity dan level item.

## Damage

Disarankan menggunakan suatu rumus untuk menggabungkan semua efek, buff, dan base damage untuk menghitung output damage dari karakter maupun mob. Berikut adalah salah satu contoh referensi rumus perhitungannya:

$$\text{LevelFactor} = 1 + (\text{AttackerLevel} - \text{TargetLevel}) * 0.05$$

$$\text{Damage} = [(\text{BaseDamage} + \text{WeaponDamage}) * (1 + \text{BuffMultiplier} + \text{EffectMultiplier})] * \text{CriticalMultiplier} * \text{RageMultiplier} * \text{LevelFactor}\%$$

$$\text{FinalDamage} = \text{Damage} * (1 - \text{TargetDamageReduction})$$

**%Untuk Multiplier yang dikira kurang jelas, boleh dibebaskan bagaimana rumusnya. Misalnya seperti RageMultiplier yang mungkin hanya dimiliki oleh class Berserker, bisa diganti menjadi 1 nilainya jika classnya adalah selain Berserker.**

Referensi Genshin Impact Attack Formula: <https://genshin-impact.fandom.com/wiki/Damage>

## Inventory

Setiap pemain memiliki penyimpanan berupa *backpack*. *Backpack* digambarkan sebagai sebuah matriks berukuran **8 x 4**. Untuk setiap slot pada matriks, dapat menyimpan **1** jenis item dan dapat dilakukan *stacking* item hingga maksimum **64** item. Jika sebuah item sudah mencapai maksimum *stack*, maka item selanjutnya akan mengambil slot yang baru.

Selain *backpack*, pemain memiliki penyimpanan berupa item-item yang dipakai oleh pemain. Item yang dapat dipakai oleh pemain antara lain

- Weapon sebanyak 1 buah
- Armor sebanyak 1 buah untuk masing-masing kepala, badan, dan kaki

- Pendant sebanyak 1 buah

Pemain dapat mencetak *backpack* untuk melihat isi dari *backpack*. **Catatan:** Hasil cetak tidak harus mirip asalkan berbentuk seperti matriks dan mudah untuk dibaca serta memiliki informasi penting seperti **ID item** dan **jumlah item**.

#### > BACKPACK

===== [ Penyimpanan ] =====								
	A	B	C	D	E	F	G	H
01								
02	HTL (02)							
03	BAC (64)							
04	BAC (24)							

Total slot kosong: 29

Selain itu, pemain juga dapat mencetak item yang sedang dipakai seperti berikut

#### > EQUIPPED ITEM

#### ===== [ Equipped Item ] =====

Weapon : <Nama Weapon>  
 Armor (Kepala) : <Nama Armor>  
 Armor (Badan) : <Nama Armor>  
 Armor (Kaki) : <Nama Armor>  
 Pendant : <Nama Pendant>

## Save/Load

Pemain dapat menyimpan seluruh state seluruh pemain dan *shop* ke dalam sebuah berkas. Bila berkas belum ada maka berkas baru akan terbentuk, sedangkan bila folder belum ada maka tampilkan pesan *error*. Berkas yang disimpan dapat dilihat pada bagian [Konfigurasi](#).

```
> SIMPAN
```

```
Masukkan lokasi berkas state : ./config  
State berhasil disimpan
```

```
> SIMPAN
```

```
Masukkan lokasi berkas state : ./invalid  
Lokasi berkas tidak valid
```

Pemain juga dapat memuat *state* dari pemain dan *shop* dari sebuah berkas. Memuat *state* hanya dapat dilakukan pada awal permainan.

```
Apakah Anda ingin memuat state? (y/n) y  
Masukkan lokasi berkas state : ./config
```

```
// game dimulai
```

```
Apakah Anda ingin memuat state? (y/n) n
```

```
// game dimulai
```

# Spesifikasi Sistem

## Ketentuan Umum

Buatlah aplikasi berbasis **Antarmuka Baris Perintah (ABP) / Command Line Interface (CLI)** untuk permainan ini dalam bahasa **C++** dan dapat di-*compile* dengan memanfaatkan **makefile**.

## Ketentuan Teknis

Berikut adalah hal-hal yang **minimal** wajib diimplementasikan di aplikasi yang Anda buat. Diperbolehkan **menambah** dengan memerhatikan konsep-konsep OOP serta desain dari aplikasi kalian!

Perlu dicatat, yang ditulis disini hanyalah minimal dan sangat besar kemungkinan kelompok kalian akan menambah abstraksi/konsep lainnya untuk membuat aplikasi ini.

### 1. Inheritance **dan** Polymorphism

Contoh: Character, Item, serta Mob

### 2. Exception **dan** Exception Handling

- a. Validasi penggunaan Item pada Inventory.
- b. Validasi Inventory penuh.
- c. Validasi lainnya jika diperlukan

### 3. Function Overloading **dan** Operator Overloading

- a. Penggunaan + dapat digunakan untuk menambah sebuah Item kepada Inventory.
- b. Penggunaan == untuk membandingkan objek yang sejenis dan setara.

### 4. Abstract Class **dan** Virtual Function

- a. Contoh: Sebuah kelas abstrak Character yang memiliki *virtual function* yang akan *di-inherit* oleh masing-masing kelas seperti Fighter, Mage, dan Assassin.

### 5. Generic Function **atau** Generic Class

- a. Contoh: Generic class atau function untuk meng-*handle* Item pada Inventory.

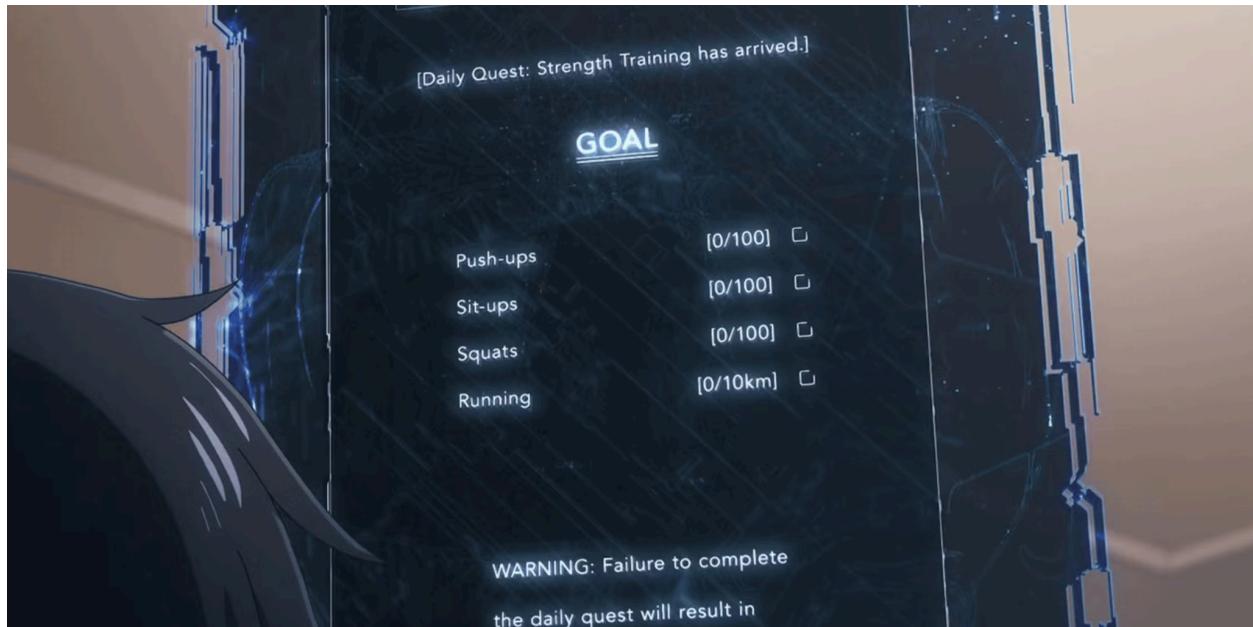
### 6. STL (*Standard Template Library*)

Bebas dipakai pada kelas apapun yang mendukung pembuatan gim. Wajib bersifat krusial ke struktur data dan mekanisme permainan, bukan hanya untuk *print* atau diletakkan pada sebuah kelas namun tidak terpakai.

**Wajib dipakai: Vector dan Map.** Boleh menambah STL lainnya jika merasa diperlukan.

# Bonus

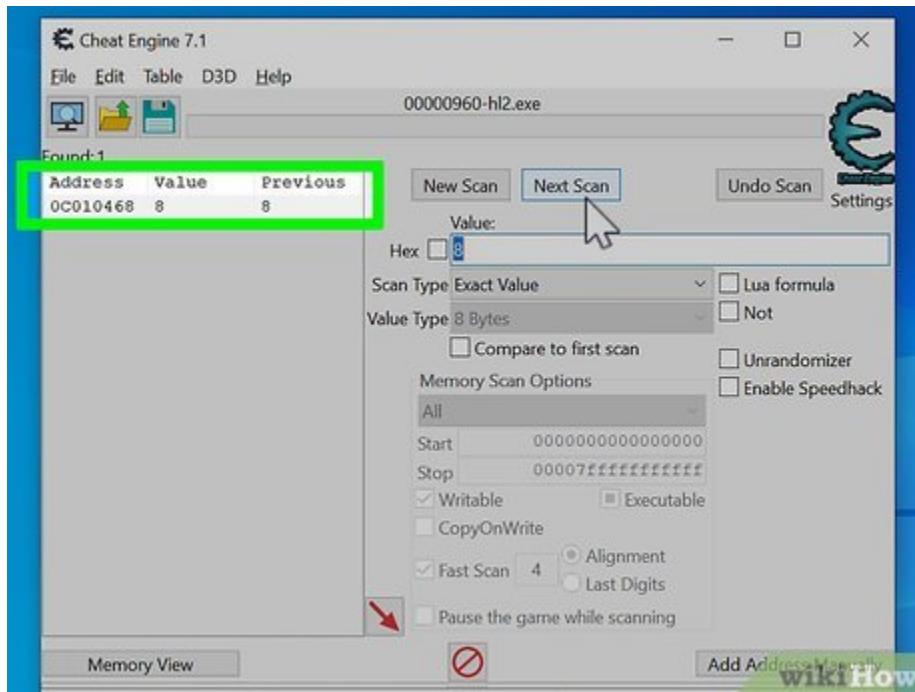
## 1. Quest System



Sumber: <https://static0.gamerantimages.com/wordpress/wp-content/uploads/2025/01/solo-leveling-quests-daily-quest-2.png>

Implementasikan sistem quest di mana setiap pengguna dapat menyelesaikan quest pada dungeon, ketika dungeon berhasil diselesaikan maka reward diberikan kepada pengguna. Implementasinya dibebaskan dan bentuk rewardnya juga dibebaskan. Quest harus bisa dikonfigurasi melalui config, harus bisa didefinisikan requirementnya apa, rewardnya apa, dsb.

## 2. Cheat



Sumber: <https://www.wikihow.com/images/thumb/6/68/363032-59.jpg/v4-460px-363032-59.jpg>

Buatlah cheat untuk mempermudah proses debugging dan demo, misalnya seperti perintah ADD\_HP 100, dan sebagainya yang bisa diakses langsung oleh pengguna buat cheat minimal 6 buah dan minimal memiliki cheat untuk ini:

1. Override dungeon finish state: FINISH\_DUNGEON menyelesaikan dungeon sekarang
2. Override quest finish state: FINISH\_QUEST menyelesaikan quest sekarang\*
3. Hesoyam: Memberikan pengguna set armor bagus, uang bertambah, darah kembali penuh
4. Tambah Hp
5. Tambah Gold
6. Reset Buff/Effect
7. God Mode

\* kalau mengimplementasikan bonus quest.

### 3. Antarmuka Pengguna Grafis (APG)/Graphical User Interface

Agen OWCA merasa tidak efektif jika hanya berlatih menggunakan tampilan terminal. Oleh karena itu, dia ingin untuk dibuatkan sebuah tampilan antarmuka sederhana untuk berlatih.

Tentunya APG harus dibuat menggunakan bahasa C++ agar tetap cocok untuk semua tim berlatih. Namun, kalian dibebaskan jika ingin menggunakan kakas bantu apapun dalam membuat APG. Tidak ada spesifikasi khusus seperti apakah tampilan antarmukanya. Kerjakan sesuai dengan kreativitas kalian.

Keseluruhan fungsionalitas dalam aplikasi harus dapat berjalan dan diterapkan dengan baik dalam APG tanpa terkecuali.

**Tidak boleh memakai game engine, pastikan GUI yang digunakan itu hanya untuk tampilan saja.  
Tidak boleh mengatur game state dan sebagainya.**

**Library yang diperbolehkan (sejauh ini): Raylib, wxWidget, imgui, sfml**

### 4. Kreativitas

Ada penilaian tambahan untuk menjadikan game ini jadi lebih menyenangkan. Misalnya menambah efek, menambah mekanisme kunci chamber, easter egg, dan lain sebagainya. Jangan lupa sebutkan di file README.md dan laporan ya!

# Konfigurasi

Diperlukan beberapa file konfigurasi yang bisa dibaca dan ditulis untuk menyimpan dan memuat state maupun konfigurasi aplikasi secara keseluruhan. Berikut adalah beberapa file dan contohnya berdasarkan hal yang disimpan:

## 1. Item

Keterangan: ItemID adalah campuran huruf atau angka yang panjangnya tiga huruf.

### item.txt

```
#<Item ID> <Item Name> <Item Type> <Rarity> <Base Atk> <Effect 1> <Effect 2>
<Effect 3>
EXC Excalibur Weapon S 500 Critical_Strike -
DRA Dragon_Armor Armor A 200 Damage_Reduction Reflect -
MNP Mana_Potion Potion C 0 Mana_Regeneration -
PHP Phoenix_Pendant Pendant B 0 Health_Regeneration -
```

## 2. Shop

### shop.txt

```
#<Item Name> <Item Rarity> <Base Price> <Stock>
Excalibur S 5000 1
Dragon_Armor A 3000 2
Mana_Potion C 500 10
Phoenix_Pendant B 2000 3
Healing_Potion D 300 15
Iron_Sword E 150 5
```

## 3. Inventory (backpack)

### backpack.txt

```
#<ROW INDEX> <COL INDEX> <ITEM ID> <TOTAL STACK>
00 PHP 24
01 MNP 30
```

4. Mob loot

**mobloot.txt**

```
#<Mob> <ITEM ID> <PROBABILITY>
Slime MNP 0.1
Dark_Knight EXC 0.25
```

5. Inventory (equipment)

**Catatan:** jika tidak memakai Weapon, Armor, atau Pendant, maka <ITEM ID> kosong

**equipment.txt**

```
WEAPON <ITEM ID>
ARMOR_HEAD <ITEM ID>
ARMOR_BODY <ITEM ID>
ARMOR_FOOT <ITEM ID>
PENDANT <ITEM ID>
```

# Panduan Pengerjaan

Tugas Besar ini dikerjakan secara sendiri/berkelompok (*senyaman kalian mengerjakan*) yang berisi **4 sampai 5 anggota (belum termasuk anggota tambahan)**. Silakan mendata kelompok di  **IF2010 OOP - Kelompok Tugas Besar 1**, paling lambat Minggu, 13 April 2025 23:59 WIB. Setelah menentukan kelompok, buatlah tim pada GH Classroom pada link ini:

<https://classroom.github.com/a/X1idueNu>

Format nama tim <kode kelompok>, contoh: **WOW** oleh salah satu anggota kelompok. Anggota lainnya bisa join ke tim yang sudah dibuat.

Semua link kebutuhan tugas besar ada di drive ini:  Public

Adapun ketentuan lain seperti berikut:

1. Wajib bisa dijalankan di sistem operasi **LINUX** dengan G++ versi 11, 14 atau 17.
  - a. Akan ada **pengurangan nilai** jika terjadi anomali seperti **keluaran** yang tidak muncul karena sejauh pengerjaan, kalian mengerjakan di **Windows**.
  - b. Berhak tidak dinilai jika program **tidak jalan total** atau gagal mensimulasikan **majoritas kondisi** untuk *test case* yang dibuat asisten.
  - c. Windows Subsystem Linux atau **WSL** diperbolehkan untuk substitusi sistem operasi **LINUX**.
  - d. Dihimbau untuk setiap kelompok mengerjakan di **WSL** atau **Linux** daripada diakhir saat demo tidak jalan sama sekali/ada kasus unik.
2. Hanya diperbolehkan menggunakan pustaka umum dan STL. Tuliskan justifikasi pustaka yang dipakai pada laporan anda.
3. Tidak boleh plagiat dari internet maupun kelompok lain.
4. Usahakan penggunaan statik, kelas dan *method* lebih banyak (~90%) **dibandingkan** dengan penggunaan *global variable*, fungsi dan prosedur.

Pengecualian **penggunaan struct**, tidak diperkenankan sama sekali, silakan gunakan kelas dan objek.

5. Dekomposisi yang baik dan implementasi yang tidak terlalu kompleks (sebuah method tidak terlalu panjang). Pecah-pecah dan buat method baru agar tidak terlalu kompleks.
6. [DEADLINE] Tugas harus diselesaikan maksimal pada **Senin, 5 Mei 2025 23:59**.

Selain penilaian dari tes fungsional, akan dilakukan penilaian dari segi desain program. Untuk mendapatkan nilai yang maksimal dari segi desain, terapkanlah prinsip:

- **Hindari menggunakan 6 konsep OOP diatas hanya karena diwajibkan saja! Gunakanlah konsep OOP yang sesuai dengan kasusnya!**
- DRY (Don't Repeat Yourself), tidak memiliki kode duplikat, pindahkan ke fungsi/kelas.

- Memiliki struktur kelas yang mudah dipahami

## Panduan Laporan

Sebagai programmer yang baik, Anda dilatih tidak hanya untuk membuat kode, tetapi juga merancang sehingga program kalian *Maintainable*.

Dokumentasikan desain program Anda melalui laporan singkat dengan format **pdf** yang **wajib** berisi:

1. Kelas diagram program anda (lengkap dengan *header method* dan atribut kelas secara singkat).
2. Penggunaan 6 konsep OOP, tuliskan semua.
3. Pada spek diatas, ada beberapa poin yang detail implementasinya dibebaskan ke anda. Tuliskanlah teknik implementasi yang ada pilih beserta alasannya!
4. Justifikasi pustaka umum dan STL (selain wajib) yang dipakai
5. Bonus yang dikerjakan

Pembuatan laporan dapat mengikuti format berikut  **IF2010\_TB1\_Laporan\_XXX.docx**

Laporan dan kode program dikumpulkan bersamaan dengan **rilis** di Github Classroom sesuai deadline.

## Asistensi

1. Asistensi sinkron wajib dilakukan minimal **satu kali** dengan ketentuan harus dilaksanakan sebelum deadline pengumpulan, maksimal **menghubungi asisten 24 jam sebelum asistensi**.
2. Asistensi digunakan untuk bertanya mengenai spesifikasi maupun pelaksanaan yang sekiranya belum terbayang dan belum jelas.
3. Alokasi asisten untuk tiap kelompok akan diumumkan setelah deadline pendaftaran kelompok.
4. Diwajibkan untuk mengisi form asistensi yang ditandatangani pula oleh asisten  
 **IF2010\_TB1\_Asistensi\_XXX.docx**

**Disediakan asistensi secara asinkron (opsional) dengan bertanya kepada asisten secara langsung melalui LINE.**

Jadwal Asistensi akan diumumkan pada tanggal 20/21 April 2025, isi jadwal kelompok kalian berdasarkan slot waktu yang ditentukan pada sheets.

## QnA

Pertanyaan terkait dengan spesifikasi yang belum jelas atau hal-hal yang berkaitan dengan masalah teknis pada tugas besar dapat ditanyakan pada borang QnA

 IF2010 OOP - QnA Tugas Besar 1