

PROPOSAL PROYEK KECIL
PEMROGRAMAN BERORIENTASI OBJEK



InForHell
Survive the semester

Nasi Padang

Syahri Banun	5054241003
Fazle Mawla Wahyuhanda	5054241020

DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN
KOMUNIKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
2025

A. Latar Belakang Permainan

Perkembangan teknologi dan dunia hiburan digital telah melahirkan berbagai jenis permainan interaktif yang tidak hanya berfungsi sebagai hiburan, tetapi juga sebagai media untuk menyalurkan kreativitas dan mengasah kemampuan berpikir logis. Dalam mata kuliah Pemrograman Berorientasi Objek, pengembangan game menjadi salah satu sarana efektif untuk memahami dan menerapkan prinsip-prinsip OOP secara konkret melalui implementasi *Inheritance*, *Encapsulation*, *Composition*, *Polymorphism*, dan lain-lain.

Mahasiswa Teknik Informatika kerap dihadapkan pada berbagai tantangan akademik maupun non-akademik, seperti tugas, proyek, praktikum, rapat organisasi, program kerja, hingga partisipasi dalam lomba. Aktivitas yang padat tersebut seringkali menimbulkan fenomena burnout mahasiswa informatika, yaitu kondisi stres akibat tekanan akademik yang dihadapi secara terus-menerus. Namun, di tengah kesibukan dan tekanan tersebut, mahasiswa Teknik Informatika juga dikenal memiliki cara unik untuk bertahan, dengan semangat pantang menyerah sebagai ciri khasnya.

Berdasarkan fenomena tersebut, lahirlah ide untuk mengembangkan sebuah permainan bertema survival, yang merepresentasikan perjuangan mahasiswa Informatika ITS dalam menghadapi berbagai bentuk “musuh” kehidupan kampus, baik akademik maupun non-akademik. Dalam permainan ini, pemain akan berjuang melawan simbol-simbol tekanan kuliah seperti deadline, bug, laporan, tugas, praktikum, hingga rapat mendadak, dalam suasana yang dikemas secara interaktif dan menghibur. Game ini tidak hanya menggambarkan kenyataan kehidupan mahasiswa, tetapi juga menjadi bentuk refleksi dan hiburan bagi para pemain yang pernah atau sedang menjalani perjuangan yang sama.

Dengan terinspirasi dari game populer seperti *Holocure*, *Vampire Survivors*, dan *Megabonk*, proyek ini mengusung konsep *action-survival* 2D dimana pemain harus bertahan hidup dari gelombang musuh yang terus meningkat seiring waktu. Namun, perbedaan utama terletak pada tema yang disesuaikan dengan kehidupan mahasiswa Teknik Informatika ITS, sehingga menciptakan rasa familiar dengan kehidupan nyata.

Game *InForHell* dikembangkan dengan pendekatan *Object-Oriented Programming*, di mana setiap entitas dalam permainan seperti player, musuh, senjata, dan item direpresentasikan sebagai objek dengan atribut dan perilaku masing-masing. Pendekatan ini tidak hanya memperkuat konsep modularitas dan *reusable code*, tetapi juga melatih kemampuan berpikir terstruktur dalam merancang sistem kompleks yang terdiri dari banyak komponen saling berinteraksi.

Melalui proyek ini, diharapkan mahasiswa tidak hanya menghasilkan sebuah game yang menarik dan menghibur, tetapi juga memperoleh pemahaman yang lebih dalam tentang penerapan prinsip OOP dalam proyek nyata, serta

bagaimana logika pemrograman dapat diterjemahkan menjadi game yang dinamis dan interaktif.

B. Deskripsi Permainan

InForHell adalah permainan *action-survival* 2D yang menggambarkan perjuangan mahasiswa Teknik Informatika ITS dalam menghadapi kerasnya kehidupan perkuliahan. Pemain berperan sebagai seorang mahasiswa yang harus bertahan hidup di tengah serangan dari berbagai permasalahan akademik dan non-akademik. Mulai dari tugas, proyek, praktikum, rapat organisasi, program kerja, hingga partisipasi dalam lomba.

Sebagai mahasiswa Teknik Informatika ITS, pemain memulai permainan dengan kemampuan dasar yang dimiliki karakternya, kemudian mengumpulkan inspirasi (EXP) dari musuh yang dikalahkan untuk meningkatkan level dan membuka berbagai item atau skill baru yang nantinya dapat menjadi kombinasi serangan untuk bertahan hidup selama permainan. Tujuan utama permainan adalah bertahan selama mungkin sambil meningkatkan kemampuan hingga batas yang tak hingga.

Semakin lama pemain bertahan, semakin kuat dan banyak musuh yang muncul, menuntut strategi dalam penggunaan skill dan pengambilan item yang tepat. Musuh dalam permainan dilengkapi dengan elemen kecerdasan buatan (AI) berbasis algoritma *pathfinding*, yang memungkinkan mereka mengejar pemain secara adaptif, menciptakan tantangan yang dinamis dan realistis.

C. Manfaat Permainan

Permainan *Inforhell* tidak hanya menghadirkan hiburan, tetapi juga memiliki berbagai manfaat bagi pemain, terutama dalam konteks simulasi kehidupan mahasiswa Teknik Informatika yang dikemas secara interaktif dan menyenangkan. Adapun manfaat permainan ini adalah sebagai berikut:

1. Sebagai Media Hiburan Interaktif

Inforhell memberikan pengalaman bermain yang seru dengan sentuhan realitas kehidupan kampus. Pemain bisa menikmati permainan sambil merasakan familiaritas dunia perkuliahan.

2. Melatih Strategi dan Pengambilan Keputusan

Pemain harus menentukan arah gerak dan strategi untuk menghindari musuh-musuh. Hal ini melatih kemampuan berpikir taktis serta pengambilan keputusan cepat dalam kondisi tertekan.

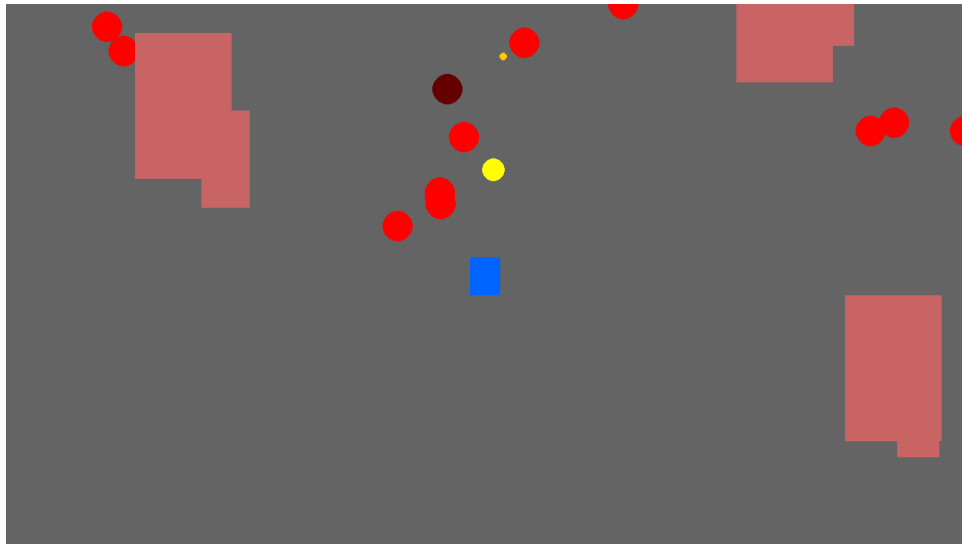
3. Meningkatkan Fokus dan Refleks

Dalam permainan, pemain perlu bereaksi cepat terhadap pergerakan musuh yang menggunakan sistem *pathfinding AI*. Kondisi ini menuntut pemain untuk tetap fokus dan meningkatkan koordinasi antara penglihatan dan gerakan tangan.

4. Menumbuhkan Daya Juang dan Ketekunan
Tema utama *Inforhell* adalah perjuangan dan pantang menyerah. Pemain akan belajar bahwa untuk ketekunan, kesabaran, dan semangat pantang menyerah dibutuhkan dalam kehidupan mahasiswa.
5. Sebagai Media Relaksasi dan Self-Expression
Walau terinspirasi dari realita beratnya dunia perkuliahan, permainan ini justru bisa menjadi wadah ekspresi dan pelepas stres. Pemain bisa merasa *relate* dengan masalah yang mereka hadapi di dunia nyata dengan cara yang lebih ringan dan menyenangkan.

D. Fitur-fitur Permainan

1. Serangan Otomatis
Pemain tidak perlu menekan tombol untuk menyerang, karena proyektil dari karakter akan otomatis keluar secara terus menerus. Tetapi pemain diharuskan untuk mengarahkan arah dari serangan tersebut secara manual.
2. Sistem *Level Up*
Musuh yang dikalahkan akan menjatuhkan EXP yang nantinya dapat dikumpulkan pemain untuk menaikkan level karakternya dan membuka skill atau item baru.
3. Sistem *Skill* dan Pasif
Setelah pemain mendapatkan cukup EXP untuk naik level atau menemukan *chestbox* di *map*, pemain dapat memilih *skill* atau pasif yang keluar secara random. Pemain dapat memilih meningkatkan skill atau pasif yang sudah pernah didapat atau memilih yang baru.
4. Skenario Permainan
Permainan dirancang untuk satu sesi permainan yang berlangsung hingga pemain kalah. Dengan gelombang musuh yang semakin banyak dan kuat seiring kuatnya pemain. Dalam *map* musuh akan di spawn secara struktural dan tidak akan tiba tiba berada di depan pemain. Lalu peletakan *chestbox* akan ditaruh secara *random* pada *map*. Estimasi lama permainan bergantung kepada seberapa lihai pemain dalam bertahan hidup dalam game.



Gambar 1. prototype InForHell

E. Fitur Kecerdasan Artifisial

Dalam permainan *InForHell*, elemen kecerdasan buatan (AI) diterapkan untuk mengatur perilaku musuh agar dapat bereaksi secara dinamis terhadap posisi pemain. Setiap musuh memiliki sistem *pathfinding* yang memungkinkan mereka mendeteksi dan mendekati posisi pemain secara adaptif, menciptakan pengalaman bermain yang lebih menantang dan realistis.

Algoritma *pathfinding* yang digunakan didasarkan pada konsep perhitungan jarak dan arah terpendek antara posisi musuh dan pemain pada bidang dua dimensi. Dengan pendekatan ini, setiap musuh secara berkala memperbarui lintasan pergerakannya untuk mengejar pemain, menyesuaikan arah dan kecepatan berdasarkan jarak relatif.

Selain itu, tingkat agresivitas musuh dapat disesuaikan dengan tingkat kesulitan permainan atau lama waktu bertahan pemain. Semakin lama pemain bertahan, semakin cepat dan cerdas respons musuh terhadap posisi pemain, sehingga meningkatkan tingkat tantangan secara progresif.

F. Cakupan Permainan

Target Penyelesaian Satu Semester

Fokus utama dalam satu semester adalah membangun *core gameplay loop* yang stabil dan berfungsi dengan baik, yang secara jelas menerapkan prinsip-prinsip OOP. Fitur-fitur yang akan diselesaikan meliputi:

1. Sistem Karakter Dasar:
 - a. karakter dengan statistik, *weapon* dan *skill* dasar.
 - b. Sistem pergerakan karakter.
 - c. Serangan otomatis, di mana pemain harus mengarahkan serangan secara manual.

2. Sistem Skill dan Pasif:

- a. Implementasi 3-5 *skill* atau pasif aktif yang bisa didapatkan pemain.
- b. Implementasi *skill* dan pasif *upgrading*.

3. Sistem Musuh dan AI:

- a. Implementasi 3-5 tipe musuh dasar dengan keunikan masing-masing.
- b. Implementasi 2-3 bos musuh yang keluar setiap waktu atau level pemain tertentu
- c. Implementasi *AI Pathfinding* agar musuh dapat mengejar pemain secara konsisten.
- d. Implementasi sistem algoritma yang menentukan jumlah atau seberapa kuat musuh tergantung pada waktu dan level pemain.

4. Sistem Progresi Pemain:

- a. Sistem pengumpulan EXP dari musuh.
- b. Sistem Level-Up.
- c. UI untuk pemilihan 3 *skill/item* acak saat naik level.

5. Manajemen Permainan:

- a. 1 *map* permainan yaitu Teknik Informatika ITS dengan tema neraka.
- b. Sistem *spawn* musuh yang terstruktur dalam gelombang yang meningkat kesulitannya seiring waktu.
- c. UI dasar yang menampilkan HP, level, bar EXP, jumlah *kill*, waktu bertahan dan lainnya.
- d. Kondisi "Game Over" ketika HP pemain mati.
- e. Sistem scoring yang didasarkan pada level yang diraih pemain, jumlah musuh yang dibunuh, dan total waktu bertahan pemain.

Target Pengembangan Lanjutan

Setelah target untuk satu semester selesai, pengembangan akan dilanjutkan untuk menambah fitur, variasi, dan tambahan agar game benar-benar matang dan siap untuk diikuti perlombaan. Fitur-fitur tambahan untuk target lomba meliputi:

1. Konten yang Lebih Kaya:

- a. Penambahan 2-3 karakter *playable* baru, masing-masing dengan *skill* awal yang unik.
- b. Penambahan 5-7 *skill* aktif dan 5-7 item pasif baru.
- c. Penambahan jenis musuh baru yang lebih variatif.

2. Sistem Permainan Lanjutan:

- a. Sistem Evolusi Skill: Kemampuan untuk menggabungkan *skill* aktif level maksimum dengan item *skill* atau pasif lain untuk menciptakan skill baru yang lebih kuat.

- b. Sistem Chestbox: Menambahkan *item drop chestbox* dari bos yang memberikan *upgrade* instan.
 - c. Peta Dinamis: Menambahkan 1-2 *map* baru dengan *layout* dan tema berbeda.
- 3. Peningkatan Teknis dan AI:
 - a. Implementasi *Dynamic Difficulty System* yang lebih canggih dan menyesuaikan agresivitas AI.
 - b. Optimasi performa untuk menangani ratusan musuh di layar secara bersamaan.
- 4. Pematangan Aset:
 - a. *Sprite* dan animasi yang dibuat dengan *style* yang lebih baik.
 - b. Integrasi *sound effects* (SFX) yang lebih variatif.
 - c. Penambahan *background music* (BGM) yang berbeda dan unik sesuai tema.
 - d. Desain UI/UX yang lebih profesional dan menarik.

G. Desain Kelas (Class Diagram)

1. GameEntity (Abstract Base Class)

Kelas abstrak yang menjadi fondasi untuk semua objek yang dapat digambar dan di-update di dalam game.

Properti:

- a. posisi: Vector2 (Menyimpan x dan y)
- b. sprite: Image
- c. rect: Rect (Untuk deteksi kolisi)
- d. is_active: bool (Penanda objek yang akan dihapus)

Metode (Abstrak):

- a. update(delta_time): Logika yang dijalankan setiap frame.
- b. render(surface): Menggambar objek ke layar.

2. Actor (Abstract Base Class)

Inherits dari GameEntity. Kelas abstrak untuk entitas yang hidup, memiliki HP, dan dapat bergerak.

Properti:

- a. hp_max: float
- b. hp_current: float
- c. kecepatan: float

Metode:

- a. kena_damage(amount: float): Mengurangi hp_current.
- b. heal(amount: float): Menambah hp_current.
- c. mati(): Terjadi saat hp_current <= 0.

3. Player (Mahasiswa Informatika)

Inherits dari Actor. Kelas pemain yang memiliki senjata bawaan dan inventory untuk Skill.

Properti:

- a. level: int
- b. exp: int
- c. exp_to_next_level: int
- d. senjata_default: SenjataDefault (Composition)
- e. inventory: Inventory (Composition)

Metode:

- a. update(delta_time): Menerima input pemain untuk menggerakkan posisi. Memanggil senjata_default.update() dan inventory.update().
- b. tambah_exp(amount: int): Menambah EXP dan memicu level_up().
- c. level_up(): Memberi sinyal ke GameManager untuk menampilkan upgrade.
- d. mati(): (Override) Memberi sinyal ke GameManager untuk end_game().

4. Upgrade (Abstract Base Class)

Kelas dasar non-visual untuk semua hal yang bisa di-level-up.

Properti:

- a. nama: string
- b. deskripsi: string
- c. level: int
- d. level_max: int

Metode (Abstrak):

- a. level_up(): Logika untuk meningkatkan statistik upgrade.

5. AttackMechanism (Abstract Base Class)

Inherits dari Upgrade. Kelas induk untuk semua hal yang bisa menyerang.

Properti:

- a. cooldown: float (Waktu antar serangan)
- b. timer: float (Timer internal)
- c. damage: float

Metode:

- a. update(delta_time): Mengelola timer. Jika timer \geq cooldown, panggil serang() dan reset timer.
- b. serang(): (Abstract) Logika unik tiap serangan.

6. SenjataDefault

Inherits dari AttackMechanism. Senjata bawaan yang unik untuk setiap karakter.

Properti:

- a. (Mewarisi cooldown, timer, damage, level, dll.)

Metode:

- a. serang(): (Override) Implementasi serangan unik karakter.
- b. level_up(): (Override) Meningkatkan damage atau mengurangi cooldown senjata ini.

7. Skill

Inherits dari AttackMechanism. Skill yang dikumpulkan saat naik level.

Properti:

- a. (Mewarisi cooldown, timer, damage, level, dll.)

Metode:

- a. `serang()`: (Override) Implementasi serangan skill.
- b. `level_up()`: (Override) Meningkatkan damage, jumlah proyektil, dll.

8. Item Pasif

Inherits dari Upgrade. Item yang memberikan boost stat.

Metode (Abstrak):

- a. `apply_stat(player: Player)`: Logika untuk menambah stat.

9. Inventory

Kelas non-visual yang dimiliki oleh Player untuk menampung Skill dan ItemPasif.

Properti:

- a. `owner`: Player (Referensi ke pemiliknya)
- b. `list_skill`: list[Skill]
- c. `list_item_pasif`: list[ItemPasif]

Metode:

- a. `update(delta_time)`: Melakukan loop pada `list_skill` dan memanggil `skill.update(delta_time)` untuk setiap skill.
- b. `tambah_upgrade(upgrade_data)`: Menambah Skill/ItemPasif baru ke list, atau memanggil `level_up()` pada yang sudah ada.
- c. `apply_all_pasif_stats()`: Menghitung ulang semua stat Player berdasarkan `list_item_pasif`.

10. Musuh (Abstract Base Class)

Inherits dari Actor. Kelas dasar untuk semua musuh.

Properti:

- a. `damage_on_touch`: float
- b. `exp_drop_amount`: int
- c. `target`: Player (Referensi ke pemain untuk dikejar)

Metode:

- a. `update(delta_time)`: Memanggil `ai_behavior()` untuk bergerak.
- b. `ai_behavior()`: Logika pathfinding untuk mengejar target.
- c. `mati()`: (Override) Memberi sinyal ke GameManager untuk `spawn_exp_orb()`.

11. Projectile (Abstract Base Class)

Inherits dari GameEntity. Kelas dasar untuk semua proyektil yang ditembakkan oleh Skill atau senjata.

Properti:

- a. `damage`: float
- b. `direction`: Vector2
- c. `kecepatan_proyektil`: float
- d. `pierce`: int (Berapa musuh yang bisa ditembus)

Metode:

- a. `update(delta_time)`: Bergerak lurus. Memeriksa kolisi dengan `list_musuh`.

12. ItemDrop (Abstract Base Class)

Inherits dari `GameEntity`. Kelas dasar untuk item yang jatuh di map dan bisa diambil.

Properti:

- a. `attraction_radius`: float (Radius tarikan ke pemain)

Metode (Abstrak):

- a. `on_pickup(player: Player)`: Logika yang terjadi saat pemain menyentuhnya.

13. GameManager (Conductor Class)

Kelas utama yang mengatur semua sistem lain. Sering disebut sebagai Conductor.

Properti:

- a. `player`: `Player`
- b. `list_musuh`: `list[Musuh]`
- c. `list_item_drop`: `list[ItemDrop]`
- d. `list_projectile`: `list[Projectile]`
- e. `game_state`: Enum (`PLAYING`, `PAUSED`, `GAME_OVER`)
- f. `waktu_bertahan`: float
- g. `upgrade_manager`: `UpgradeManager` (Kelas helper untuk mengelola logika upgrade)

Metode:

- a. `update(delta_time)`: Memanggil `update()` pada player dan semua objek di dalam semua list (musuh, item, proyektil).
- b. `render(surface)`: Memanggil `render()` pada semua objek.
- c. `spawn_musuh()`
- d. `spawn_exp_orb(posisi, amount)`
- e. `spawn_projectile(posisi, damage, direction, ...)`
- f. `spawn_aoe(posisi, damage, radius, ...)`
- g. `show_level_up_screen()`: Meng-Pause game dan meminta 3 upgrade dari `upgrade_manager`.
- h. `end_game()`: Menampilkan layar "Game Over".

H. Pembagian Kerja Tim

Nama	Tugas	Deskripsi Tugas
Syahri Banun	Writer	- Menentukan tema, latar, dan konsep utama permainan yang merepresentasikan

		<p>kehidupan mahasiswa Teknik Informatika ITS.</p> <ul style="list-style-type: none"> - Menyusun struktur cerita dan progres permainan agar sesuai dengan alur peningkatan kesulitan. - Mengatur keterkaitan antara sistem, karakter, dan mekanik permainan agar tetap konsisten dengan konsep yang diusung. - Membuat dokumentasi desain konsep dan struktur alur bermain.
	AI Programmer	<ul style="list-style-type: none"> - Mengembangkan sistem kecerdasan buatan (<i>Artificial Intelligence</i>) untuk perilaku musuh dalam permainan. - Mengimplementasikan <i>pathfinding</i> menggunakan algoritma seperti A* atau BFS agar musuh dapat mengejar pemain secara dinamis. - Mendesain <i>dynamic difficulty system</i> untuk menyesuaikan tingkat kesulitan secara otomatis berdasarkan performa pemain. - Mengatur variasi perilaku musuh agar memberikan tantangan yang beragam.
	QA Tester	<ul style="list-style-type: none"> - Melakukan pengujian internal untuk memastikan seluruh fitur berjalan sesuai rancangan. - Mengidentifikasi bug dan masalah teknis pada sistem AI maupun mekanik permainan. - Melakukan <i>game balancing</i> terhadap tingkat kesulitan, kecepatan musuh, dan durasi permainan.
Fazle Mawla Wahyuhanda	Game Designer	<ul style="list-style-type: none"> - Merancang sistem gameplay bertema <i>action-survival 2D</i> yang menggambarkan perjuangan mahasiswa menghadapi tekanan akademik dan non-akademik. - Mendesain mekanik permainan, sistem skill, leveling, dan peningkatan karakter.

		<ul style="list-style-type: none"> - Mengatur tampilan visual, antarmuka (UI), dan desain lingkungan agar sesuai dengan suasana permainan. - Menentukan gaya seni (<i>art style</i>) dan atmosfer permainan yang konsisten dengan tema <i>InForHell</i>.
	Lead Programmer	<ul style="list-style-type: none"> - Mengimplementasikan seluruh logika permainan menggunakan Python dengan library Pygame. - Mengembangkan sistem kontrol karakter, mekanik skill dan pasif serta sistem spawn musuh. - Mengatur integrasi antara gameplay dan sistem AI agar permainan berjalan secara mulus dan responsif. - Melakukan <i>debugging</i>, optimasi performa, dan pengelolaan repositori proyek menggunakan Git.
	Visual & Audio Integrator	<ul style="list-style-type: none"> - Mengintegrasikan seluruh aset visual dan audio ke dalam permainan berbasis Pygame. - Menyesuaikan efek suara, musik latar, dan animasi agar mendukung suasana permainan. - Menjamin sinkronisasi antara animasi karakter, efek skill, dan elemen antarmuka agar pengalaman bermain tetap halus dan menyenangkan.

I. Kompetisi yang akan Diikuti

1. Nama Kompetisi: Indie Game Ignite (IGI) 2026
2. Tanggal Kompetisi:
 - a. Pendaftaran & Pengumpulan Game: Juni – Agustus 2026 (tentatif, menunggu pengumuman resmi dari panitia COMPFEST 18)
 - b. Seleksi dan Presentasi Final: September 2026

3. Penyelenggara: COMPFEST (Acara TIK tahunan oleh Fakultas Ilmu Komputer, Universitas Indonesia)
4. Tema Kompetisi: "Kebebasan Kreatifitas" (Creative Freedom) Mendorong developer indie dan mahasiswa untuk menciptakan permainan dengan ide paling orisinal, inovatif, dan eksekusi terbaik, tanpa batasan tema yang mengikat.
5. Batasan Teknologi:
 - a. Game dapat dikembangkan lebih lanjut menggunakan engine atau library bebas dan khusus untuk game (Godot, Unity, Unreal, Pygame, dsb.).
 - b. Platform minimal: PC (Windows/Linux), platform lain bersifat opsional.
 - c. Orisinalitas (ide, aset, dan eksekusi) adalah poin penilaian utama.
 - d. Game wajib diserahkan dalam bentuk playable build yang dapat dijalankan langsung.
6. Kesesuaian Aplikasi dengan Kompetisi: Game InForHell - Survive the semester sangat cocok untuk kompetisi IGI karena:
 - a. Mengusung genre action-survival yang sangat populer di kancah indie. Tema yang diusung sangat orisinal dan terbilang unik yaitu kehidupan mahasiswa Teknik Informatika ITS, yang sangat sesuai dengan tema "Kebebasan Kreatifitas".
 - b. Cerita dan konsep "kampus neraka" adalah bentuk ekspresi yang unik, menyajikan narasi yang sangat relevan bagi audiens mahasiswa dan juri.
 - c. Dikembangkan menggunakan Python dengan library Pygame, menunjukkan pemahaman teknis yang mendalam tentang logika game dari dasar dan penerapan OOP.
 - d. Proyek ini menggabungkan pemrograman AI (pathfinding musuh), desain sistem gameplay yang adiktif, dan penulisan konsep yang kuat menjadi satu produk yang kohesif.