



LET'S PLAY SPAKBOR

HILLS

START



M IRHBAR A
182223050



IRDINA ILMIYAH YOSAPAT
182223060



SEBASTIAN ALBERN HUGHROHO
182223074



KELOMPOK 4



HOW SPAKBOR HILLS GAME WORKS?

USER MANUAL

CARA MEMULAI PERMAINAN

- Jalankan program dengan `jgradlew run` melalui terminal
- Pada menu utama, pilih opsi:
 - start game : memulai game
 - quit : keluar dari game
 - help : meminta bantuan mengenai game
 - credits
- Lalu ketika kamu memilih start game, kamu akan diminta mengisi nama. Kamu bisa langsung tekan enter setelahnya!



Setelah kamu mengisi nama, game akan memulai hari pertamamu di Spakbor Hills!

GAMEPLAY

Selamat datang di Spakbor Hills! sebuah game bertani berbasis Command Line Interface (CLI) yang dibuat khusus untuk membantu Dr. Asep Spakbor memulai hidup baru sebagai petani. Lewat game ini, kamu akan menjalani keseharian sebagai petani yang menanam, mereman, memasak, memancing, hingga bersosialisasi dengan warga desa. Tapi hati-hati, meski kehidupannya sederhana, banyak strategi dan manajemen energi yang harus kamu pikirkan!



TUJUAN PERMAINAN

Game ini tidak memiliki akhir tetap, tapi kamu bisa mengejar dua milestone utama:

- Memiliki total uang (gold) minimal 17.209
- Menikah dengan salah satu NPC

Kalau salah satu dari dua target itu tercapai, kamu akan disuguhkan halaman end game stats yang menampilkan berbagai pencapaianmu. Tapi kamu tetap bisa lanjut bermain setelah itu!

AKSI YANG BISA DILAKUKAN

Dalam Spakbor Hills, kamu punya banyak aksi yang bisa dilakukan setiap harinya. Namun perlu diingat, setiap aksi membutuhkan energi dan memakan waktu. Berikut ini beberapa aksi utama:

Chatting	Ngobrol dengan NPC untuk menambah hubungan (heart point). Memberikan efek +10 heart point, -10 energi dan -10 menit.
Gifting	Memberikan item ke NPC. Efek hubungan tergantung itemnya (loved, liked, hated). Memberikan efek +heart point, -5 energi, dan -10 menit
Proposing	Melamar NPC untuk jadi tunangan (fiance). Heart point harus 150! Butuh Proposal Ring dan memberikan efek -10 energi (diterima) / -20 energi (ditolak) dan akan -1 jam.
Marry	Menikahi NPC yang sudah jadi tunangan. Setelah nikah, hari langsung skip ke jam 22.00. Butuh Proposal ring dan memberikan efek -80 energi, time skip.
Watching	Nonton TV di rumah (jika punya TV) untuk lihat ramalan cuaca. Butuh TV dan memberikan efek -5 energi, -15 menit.
Selling	Menaruh barang ke shipping bin. Penjualan diproses saat kamu tidur. Butuh shipping bin dan memberikan efek -15 menit.
⚠️ Kalau energimu sampai habis (-20), kamu bakal otomatis tidur. Jadi atur energi baik-baik, ya!	
	
Tilling	Membajak lahan kosong agar bisa ditanami. Butuh hoe dan memberikan efek -5 energi dan -5 menit untuk setiap tile
Recover Land	Mengembalikan lahan bajakan menjadi tanah biasa lagi. Butuh Pickaxe dan memberikan efek -5 energi dan -5 menit untuk setiap tile
Planting	Menanam seed ke tanah yang sudah dibajak. Butuh Seed dan memberikan efek -5 energi dan -5 menit untuk setiap tile
Watering	Menyiram tanaman. Tapi kalau sedang hujan, kamu bisa hemat energi! Butuh Watering Can dan memberikan efek -5 energi dan -5 menit untuk setiap tile.
Harvesting	Mereman hasil tanaman yang sudah tumbuh. Memberikan efek -5 energi dan -5 menit untuk setiap tile
Fishing	Memancing ikan dengan sistem tebak angka. Lokasi, musim, waktu, dan cuaca memengaruhi jenis ikan dimana dilakukan di pond, sungai, danau, atau laut. Butuh Fishing Rod & memberikan efek -5 energi, +15 menit.
Cooking	Memasak makanan dari bahan-bahan yang kamu kumpulkan. Butuh Resep, bahan makanan, dan bahan bakar (coal/firewood), memberikan efek -5 energi dan -5 menit untuk setiap tile.
Sleeping	Tidur untuk mengisi ulang energi. Kalau kamu tidur dalam kondisi sangat lelah, kamu cuma akan dapat setengah energi.
Visiting	Mengunjungi rumah NPC atau tempat-tempat di luar farm. Memberikan efek -10 energi dan -15 menit.

HOW SPAKBOR HILLS GAME WORKS?



WAKTU, MUSIM, DAN CUACA



- 1 detik dunia nyata = 5 menit waktu di game
- 1 hari dimulai jam 06.00 dan berakhir jam 02.00 dini hari
- 1 musim berlangsung selama 10 hari
- Cuaca bisa Sunny atau Rainy. Saat hujan, kamu tidak perlu menyiram tanaman karena otomatis tersiram air hujan.

AKSES WORLD MAP



jembatan di Spakbor Hills ada 2, dimana lokasinya ada di pojok kanan jembatan!! (dari farm map)



HUBUNGAN DENGAN NPC



Di Spakbor Hills, kamu bisa berteman, menjalin hubungan, bahkan menikah dengan NPC. Interaksi bisa dilakukan lewat:

- Chatting
- Gifting (efek tergantung item yang dikasih: loved, liked, hated)

Jika hubungamu cukup dekat (heartPoints = 150), kamu bisa melamar NPC dengan Proposal Ring, dan menikah kesokan harinya.

NAVIGASI & INTERAKSI

Selama kamu lagi main seperti biasa (playState), kamu bisa gerakin karakter dan berinteraksi sama objek atau NPC di dunia game. Gampang banget kok!

Gerak Karakter:

- W atau Panah Atas → jalan ke atas
- S atau Panah Bawah → jalan ke bawah
- A atau Panah Kiri → jalan ke kiri
- D atau Panah Kanan → jalan ke kanan



Interaksi:

- E → tombol serbaguna. Bisa buat ngobrol sama NPC, buka pintu, ambil item, dll.
- Enter → mirip kayak E, kadang lebih nyaman dipake buat konfirmasi di menu atau dialog.

INTERAKSI MENU & UI

Nah, selama main, kamu bakal sering ketemu yang namanya gameState. Ini semacam mode-mode dalam game yang beda fungsi tombolnya. Ini penjelasan gampangnya:

PAUSE STATE (P)

P → nge-pause atau resume game. Kalau lagi pause, waktu di game berhenti total. Cocok buat mikir strategi 😊



INVENTORY STATE (I)

- I → buka atau tutup tas kamu
- WASD / panah → navigasi antar item
- E → pakai item yang dipilih
- Delete → buang item dari tas
- Escape → keluar dari mode gifting dan balik ke playState



STATS STATE (C)

C → buka atau tutup layar statistik karakter kamu



DIALOG STATE

E → lanjut dialog atau nutup kotak dialog. Biasanya pas ngobrol sama NPC



NPC CONTEXT MENU STATE

- E → ngobrol sama NPC
- G → masuk ke mode kasih hadiah (langsung buka inventory di mode gifting)
- P → melakukan propose ke NPC
- Escape → nutup menu NPC dan balik ke mode main biasa



COOKING STATE

- W/S atau panah atas/bawah → pilih resep
- Enter → pilih dan coba masak resep
- E atau Escape → keluar dari menu masak
- Backspace → balik dari tampilan detail resep ke list resep



SHIPPING BIN STATE

- E → buka atau tutup shipping bin
- WASD / panah → navigasi antar item
- Enter → jual item yang dipilih ke bin



CHEAT WAKTU

Kalau kamu lagi pengen eksperimen atau ngetes sesuatu...

- Coba tekan / → bakal muncul dialog cheat yang memungkinkan kamu ubah hari dan waktu dalam game. Lumayan buat ngeskip waktu atau setting kondisi tertentu.

STORE STATE

- E → buka atau tutup menu toko
- WASD / panah → navigasi item di toko
- Enter → beli item yang dipilih



ACTIONS IN SPAKBOR HILLS

COOKING



Ketika berinteraksi, `gameState` game akan berubah menjadi `cookingState`, dan antarmuka `CookingMenu` akan ditampilkan.

Setelah masuk ke `cookingState`, pemain akan melihat:

1. Daftar Resep: Di sisi kiri layar, akan ditampilkan daftar resep yang tersedia.
2. Detail Resep: Saat pemain memilih resep dari daftar, detail mencakup nama resep, bahan dan jumlah, item hasil masakan, status kepemilikan resep



Penyelesaian Memasak dan Hasil Masakan

Memasak bukanlah proses instan. Masakan akan "matang" seiring berjalaninya waktu dalam game. `FoodItem` yang sudah jadi (berdasarkan `outputItemName` dari resep) akan dibuat menggunakan `ItemFactory` dan secara otomatis ditambahkan ke inventaris pemain.



Setelah pemain memilih resep untuk dimasak:

1. Pengecekan Bahan & Bahan Bakar
2. Pengambilan Bahan
3. Memasukkan ke Antrean Memasak
4. Status Memasak



EATING



Tidak semua item dapat dikonsumsi. Hanya item yang mengimplementasi kan antarmuka (`interface`)

`IConsumable` yang dapat digunakan untuk fungsi ini.

Cara Mengonsumsi Item :

1. Membuka Inventaris
2. Memilih Item Konsumsi
3. Menggunakan Item



Proses Konsumsi oleh `EatingController`

1. Panggilan `consume()`
2. Delegasi ke Item: Daripada `EatingController` yang langsung mengubah statistik pemain, logika konsumsi didelegasikan ke item `IConsumable` itu sendiri (`item.consume(gp)`).
3. Efek Konsumsi: Energi (dan/atau statistik lain) pemain akan pulih sejumlah nilai yang ditentukan oleh item tersebut dan item tersebut akan dihapus dari inventaris pemain

TILLING



Pemain mendekati `LandTile` yang ingin diolah dan Pemain memastikan Hoe adalah `activeItem` mereka.

Tekan tombol interaksi (E) dan `FarmingController` akan menerima permintaan interaksi dan memanggil metode `till()`:

1. Sistem memeriksa apakah `targetTile` saat ini berstatus **LAND**.
2. Pengecekan Energi: Sistem juga memeriksa apakah pemain memiliki cukup energi (`player.getEnergy() < ENERGY_COST_PER_TILE`).
3. Pengurangan Energi & Waktu
4. Perubahan Status Tile : Pemain akan melihat visual **LandTile** berubah



Mengembalikan `LandTile` dari status **SOIL** atau **WATERED** kembali menjadi **LAND**



Pemain memastikan Pickaxe adalah `activeItem` mereka dan pencet E sehingga `FarmingController` akan menerima permintaan interaksi dan memanggil metode `recover()`:

1. Sistem memeriksa apakah `targetTile` saat ini berstatus **SOIL** atau **WATERED**
2. Pengecekan Energi
3. Pengurangan Energi & Waktu
4. Perubahan Status Tile

RECOVER LAND



ACTIONS IN SPAKBOR HILLS

PLANTING



Pemain harus memilih bibit sebagai **activitem**



Pemain menekan tombol interaksi (E) pada **LandTile** yang sudah diolah.



FarmingController akan menerima permintaan interaksi dan memanggil metode `plant()`:

1. Pengecekan Kondisi dan cek Energi
2. Pengurangan Energi, Waktu dan item
3. Perubahan Status Tile : `targetTile` diubah menjadi **PLANTED**
4. tanaman akan mulai tumbuh seiring waktu game berjalan

WATERING



Menyiram **LandTile** yang sudah diolah atau ditanami. Alat yang Dibutuhkan: WateringCan (penyiram).



Rekan tombol E dan **FarmingController** akan menerima permintaan interaksi dan memanggil metode `water()`:

1. Pengecekan Kondisian Penyecekan Energi
2. Pengecekan Isi Penyiram
3. Pengurangan Energi, Waktu dan air
4. Perubahan Status Tile

HARVESTING



Pemain mendekati **LandTile** yang berstatus **HARVESTABLE** dan Pemain memastikan tidak ada alat yang aktif (yaitu, activitem adalah "Hand" atau "**No Item**").



Pemain menekan tombol interaksi (E).



FarmingController akan menerima permintaan interaksi dan memanggil metode `harvest()`:

1. Pengecekan Kondisi, Alat dan Energi
2. Pengurangan Energi & Waktu
3. Penambahan Hasil Panen
4. Perubahan Status Tile dan Pembaruan Visual

FISHING



Pemain memerlukan **FishingRod** (pancing) sebagai **activitem** mereka dan Pemain harus menemukan **FishableSpot** (spot memancing) di dunia game.



Pemain menekan tombol interaksi (E) dan **GamePanel** akan mendeketi interaksi ini dan mendlegaskannya ke **FishingController** melalui metode `handleFishingAttempt()`



Menentukan Potensi Ikan (Mencari ikan yang Cocok):

1. **FishingController** akan mendapatkan daftar semua **FishItem** yang ada dalam game
2. Untuk setiap ikan, **FishingController** akan memanggil metode `isCatchable()` pada FishItem tersebut. Penangkapan ikan berdasarkan Musim, Cuaca, Waktu dan Lokasi
3. Hanya ikan-ikan yang memenuhi semua kriteria `isCatchable()` ada di `catchableFish`



Setiap **FishableSpot** memiliki tipe **FishLocation** (POND, MOUNTAIN_LAKE, OCEAN, FOREST_RIVER) yang akan memengaruhi jenis ikan yang bisa ditangkap.



Berhasil Menangkap :

1. Energi pemain akan dikurangi dan Waktu dalam game akan bertambah.
2. **FishItem** berhasil ditangkap ditambahkan ke inventaris Jumlah ikan yang ditangkap pemain akan dihitung

Tidak Berhasil Menangkap:

1. Energi pemain akan dikurangi dan Waktu dalam game akan bertambah.



Mencoba Menangkap Ikan:

1. Jika ada **catchableFish** yang ditemukan, sistem akan mencoba menangkap ikan dari daftar tersebut secara acak



ACTIONS IN SPAKBOR HILLS

SELLING



Memungkinkan pemain untuk menjual item-item yang mereka miliki dan akan mendapatkan gold dari yang dijual.

- Alur Permainan:
1. Mengakses **ShippingBin**
 2. Pemain akan melihat daftar item yang saat ini ada di **ShippingBin**
 3. Di sisi lain ada inventaris



Pemain bisa memindahkan Item ke **ShippingBin** (Untuk Dijual) ataupun menghapus Item dari **ShippingBin** (Membatalkan Penjualan)



Penjualan di **ShippingBin** diselesaikan secara otomatis pada akhir hari (saat pemain tidur)

BUYING



Pemain mendekati objek Store di dunia game dan Pemain menekan tombol interaksi (E).



Pemain akan melihat daftar item yang tersedia untuk dijual di toko. Ada indikator `player.gold` saat ini.



1. `StoreController` akan memanggil metode `sellItem(index)` : melakukan Pengecekan Dana dan Inventoris
2. Melakukan Transaksi
3. Pembelian Gagal: Jika dana tidak cukup atau inventaris penuh, aksi pembelian akan gagal dan pesan notifikasi muncul (`gp.ui.setDialog`).

SLEEPING



Pemain mendekati objek Bed di dunia game dan Pemain menekan tombol interaksi (E).



1. Proses Tidur oleh `SleepController`
2. Metode `update()` di `SleepController` (Dipanggil secara berkala oleh `GamePanel`)

Terdapat `forceSleep()`: Ini adalah metode lain yang dapat digunakan untuk memulai tidur tanpa permeriksa `canSleep()`,untuk skenario game tertentu (misalnya pemain pingsan kelelahan).

WATCHING



Pemain mendekati objek TV di dunia game lalu tekan tombol interaksi (E). Saat berinteraksi, `TV.onInteract()` dipanggil, yang kemudian memanggil `gp.watchingController.watchTV()`.



- Proses Menonton TV oleh `WatchingController`:
1. `watchTV()` mengambil ramalan cuaca untuk hari saat ini
 2. Waktu game maju 15 menit dan Energi pemain berkurang 5 point.

VISITING



Konsep visiting merupakan action memasuki rumah NPC. Pemain "mengunjungi" rumah dan NPC hanya dengan mendekat dan berinteraksi di lokasi NPC berada di peta dunia.

- Lokasi NPC: NPC memiliki koordinat dunia (wX, wY).
- Interaksi Berbasis Lokasi/Waktu



Pemain dapat berinteraksi dengan NPC, dimana bisa berbicara dan memberi hadiah

ACTIONS IN SPAKBOR HILLS

GIFTING



Mekanisme gifting tetap diimplementasikan dalam method `giftItemToNPC` di `NPCController.java`

Pengecekan Item: Pemain harus memiliki item yang valid.

- Penilaian Item: Item dinalai (Loved, Liked, Hated, atau Neutral) yang menentukan jumlah heart points yang ditambahkan atau dikurangi.
 - Loved: +25 heart points
 - Liked: +20 heart points
 - Hated (jika `doesHateAllUnlistedItems` true): -25 heart points
 - Neutral: +0 heart points
- Reaksi NPC: NPC memberikan pesan reaksi berdasarkan poin yang diberikan.
- Transisi State: Game akan masuk ke `dialogState` untuk menampilkan reaksi NPC.



Interaksi dengan NPC memicu dialog yang diatur dalam method `onInteract` di kelas `NPC` dan subkelasnya.

CHATTING

- Pemicu Interaksi: Pemain berinteraksi dengan NPC (melalui `gp.currNPC.onInteract(gp, this)`).
- Pengaturan Dialog: Method `setDialogue()` di NPC mengatur array dialog berdasarkan `relationshipStatus` dan `heartPoints` pemain.
- Menampilkan Dialog: Dialog ditampilkan di UI dan game masuk ke `dialogState`.
- Mengulang Dialog: Pemain dapat mengulang dialog hingga selesai.

PROPOSING



- Kondisi Proposal: `heartPoints` NPC harus maksimal (`MAX_HEART_POINTS`).
- NPC harus berstatus "Single".
- NPC harus berjenis kelamin "Female".
- Pemain harus memiliki item "Ring" (cincin) di inventory.

Hasil Proposal:

- Berhasil: Cincin hilang, status NPC menjadi "Fiance", tanggal pertunangan dicatat, dan pemain memiliki tunangan. Pesan selamat ditampilkan.
- Gagal: Pesan penolakan ditampilkan melalui `gp.currNPC.getProposalDeclineMessage()`, yang memberikan alasan penolakan (misalnya, heart points kurang, sudah terikat, atau bukan tipe).



Pernikahan dilakukan dimana pemain harus memiliki tunangan. Dimana Pernikahan hanya bisa dilangsungkan saat hari setelah pertunangan dan pemain harus memiliki cincin



Opsi "[M] Marry" ditampilkan hanya jika pemain sudah bertunangan dengan NPC

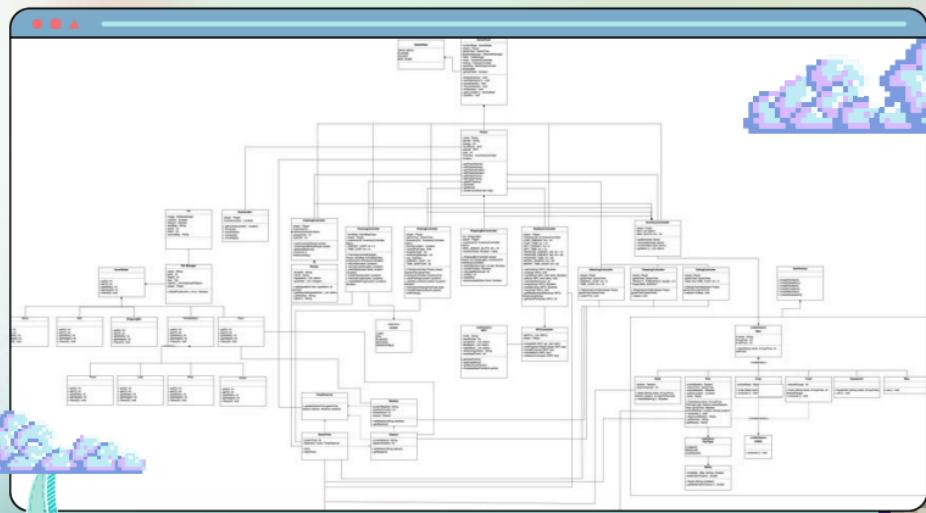


Proses Pernikahan:

- Berhasil: Status tunangan berubah menjadi "Married", tunangan diatur sebagai pasangan pemain, dan `fiancepemain` disetel null. Pesan selamat ditampilkan, dan waktu game dipercepat hingga 10:00 PM.
- Gagal: Pesan penolakan ditampilkan

MARRYING

FINAL STRUCTURE OF THE OBJECTS IN THE GAME



Dalam pengembangan game ini kami merancang sebuah Class Diagram, berfungsi seperti denah bangunan. Setiap kotak mewakili sebuah "kelas" – semacam resep atau template untuk membuat objek di game. Garis-garis yang menghubungkan kotak-kotak tersebut menunjukkan bagaimana kelas-kelas ini saling berinteraksi, bekerja sama untuk menciptakan pengalaman bermain yang utuh. Secara garis besar, kami membagi struktur game menjadi beberapa area utama:

- **Pemain (Player)**

Kelas ini memegang semua data penting karakter yang Anda kendalikan, mulai dari barang bawaan di inventaris, seberapa banyak energi yang tersisa, skillapa yang dimiliki, hingga posisinya di peta.

- **Mahluk Hidup (LivingBeing)**

Sebagai fondasi bagi siapa pun yang bernyawa di game kami, kelas ini mencakup Player dan NPC (Non-Player Character). Mereka semua berbagi ciri dasar seperti kesehatan (HP), energi, dan mungkin juga location atau position.

KARAKTER DAN ENTITAS HIDUP

- **Pengatur Waktu dan Musim (Time, Season)**

Inilah "detak jantung" dunia game kami. Kelas-kelas ini bertugas mengatur waktu yang berjalan, hari berganti, dan bagaimana musim berubah. Sangat penting, karena banyak hal di game kami bergantung pada waktu dan kondisi musim yang sedang berlangsung.

- **Peta (Map)**

Kelas ini adalah otaknya seluruh area permainan. Ia mengelola petak-petak tanah (Tile) tempat semua objek game ditempatkan dan memastikan Anda bisa bergerak dan berinteraksi di dalamnya.

- **Lokasi (Location)**

Mewakili area-area spesifik di dalam peta, mungkin setiap Tile atau grup Tile dapat memiliki Location yang unik.

DUNIA DAN LINGKUNGAN GAME

FINAL STRUCTURE OF THE OBJECTS IN THE GAME

ITEM DAN HARTA BENDA (ITEM)

- Dasar Semua Barang (Item)

Ini adalah template utama untuk setiap benda yang bisa Anda temukan atau miliki di game. Dari sini, kami membuat berbagai jenis barang lain melalui pewarisan:

- Alat (Tool): Seperti kapak atau cangkul, lengkap dengan informasi ketahanannya (durability).
- Material (Material): Bahan mentah yang bisa Anda kumpulkan untuk diolah.
- Produk (Product): Hasil dari kegiatan seperti memasak atau crafting, yang punya nilai tambah seperti berat atau energi (addedWeight, addedEnergy). Ini bisa berupa hasil panen (FarmProduct) atau barang olahan (NonFarmProduct).
- Benih (Seed): Kunci untuk menanam tanaman, tentu saja terhubung erat dengan tanaman yang akan dihasilkannya.

- Kantong Ajaib (Inventory)

Kelas ini bertugas menyimpan dan mengatur semua barang bawaan pemain dengan rapi.

- Buku Resep (Recipe)

Mengatur kombinasi Material apa saja yang dibutuhkan untuk menciptakan Product atau Toolbaru.

SISTEM AKSI & INTERAKSI PEMAIN

- Gerakan Pemain (PlayerAction)

Ini adalah fondasi untuk semua hal yang bisa Anda lakukan di game. Kelas ini adalah kelas abstrak yang memiliki banyak turunan, masing-masing mewakili sebuah aksi spesifik:

- Aksi Ekonomi : Termasuk Sell (menjual), Buy (membeli) melalui ShippingBin atau Store, serta Cook (memasak) yang mengolah material menjadi produk makanan.
- Aksi Pertanian (Farm): Mencakup semua interaksi dengan lahan dan tanaman, seperti Till (mencangkul tanah), RecoverLand (memulihkan lahan), PlantSeed (menanam benih), Water (menyiram tanaman), dan Harvest (memanen).
- Aksi Pengumpulan Sumber Daya: Seperti Mine (menambang) untuk mendapatkan mineral, Chop (menebang) untuk mengumpulkan kayu, dan Fish (memancing) untuk menangkap ikan.
- Aksi Pribadi Pemain: Meliputi Sleeping (tidur) untuk melewati waktu, Eating (makan) untuk mengonsumsi makanan, dan Watching (menonton), yang mungkin terkait dengan interaksi pasif atau event.
- Aksi Sosial (NPC Interaction): Berinteraksi dengan NPC melalui Chat (mengobrol), Gift (memberi hadiah), Propose (melamar), dan Marry (menikah), yang semuanya memengaruhi Relationship dengan NPC.

- Bangunan (Building)

Kelas ini mewakili berbagai struktur di game yang bisa Anda bangun atau berinteraksi dengannya, seperti FarmBuilding atau ToolShop.

SISTEM PERTANIAN: KEHIDUPAN DI LADANG

- Tanaman (Plant)

Ini adalah kelas pusat untuk semua tanaman yang bisa Anda tanam di ladang. Yang menarik, diagram ini menunjukkan hubungan erat antara Plant dan PlantState, yang merupakan inti dari implementasi State Pattern kami.

- Tahapan Hidup Tanaman (PlantState)

Ini adalah serangkaian "kondisi" yang bisa dimiliki sebuah tanaman. Mulai dari SeedState (saat masih benih), YoungState (tanaman muda), MatureState (siap panen), hingga WitheredState(layu). Setiap tahap ini punya perlakunya sendiri yang unik.



FINAL STRUCTURE OF THE OBJECTS IN THE GAME



MENERAPKAN DESIGN PATTERN

Dalam upaya menciptakan game yang tidak hanya berfungsi, tetapi juga fleksibel, mudah dikembangkan, dan terhindar dari kode yang berantakan, kami mengimplementasikan beberapa "Design Pattern". Ini adalah solusi-solusi desain yang sudah teruji dan sering digunakan dalam pengembangan perangkat lunak. Untuk game ini, kami fokus pada tiga pola utama: Observer, Item Factory, dan State Pattern.

STATE PATTERN

Konsep:

Untuk mengelola siklus hidup tanaman, kami menggunakan State Pattern agar logika pertumbuhan dan perubahan kondisi tanaman tetap rapi dan mudah dikelola. Tanaman akan memiliki state aktif yang mendefinisikan apa yang bisa dilakukan saat itu.

Penerapan di Game:

- Context: Plant
- State Interface: PlantState mendefinisikan metode umum seperti grow(), water(), dan harvest().
- Concrete States:
 - SeedState: Benih, belum bisa dipanen.
 - YoungState: Dalam tahap pertumbuhan.
 - MatureState: Siap dipanen.
 - WitheredState: Layu dan tidak dapat tumbuh lagi.

Contoh Alur:

- Plant memulai di SeedState.
- Saat Time (Observer) memberi tahu bahwa hari berganti, grow() dipanggil dan SeedState mengubah state menjadi YoungState, dan seterusnya.
- Saat disiram, Plant juga akan merespons berdasarkan state aktifnya.

Manfaat:

- Cleaner code: Menghindari if-else panjang di dalam Plant.
- Easy to extend: Misal, menambahkan DiseasedState tinggal buat kelas baru tanpa mengubah Plant.
- Isolated logic: Setiap state bertanggung jawab atas perlakunya sendiri, membuat debugging lebih mudah.



PABRIK SEMUA JENIS ITEM

FACTORY PATTERN

Konsep:

Item di game dibuat dengan pendekatan Factory Pattern, di mana semua jenis item seperti alat, benih, makanan, dan hasil panen dibuat lewat satu gerbang pusat bernama ItemFactory. Ini menghindari duplikasi logika pembuatan item di berbagai tempat.

Penerapan di Game:

- ItemFactory: Bertanggung jawab membuat semua item sesuai kebutuhan.
- Use Case:
 - Crafting: Saat pemain membuat item berdasarkan resep, sistem cukup panggil ItemFactory.createItem("GrilledFish").
 - Pembelian di Toko: Store cukup meminta ItemFactory untuk membuat barang tanpa tahu detailnya.
 - Loot / Drop dari Musuh: Loot juga dibuat lewat factory agar konsisten.

Manfaat:

- Centralized creation logic: Semua aturan dan atribut item disatukan di satu tempat.
- Extensible: Tambah item baru cukup tambahkan di factory, tanpa ubah toko/crafting/loot.
- Consistent & maintainable: Semua item dibuat dengan standar yang sama.

SISTEM NOTIFIKASI OTOMATIS

Konsep:

Observer Pattern kami terapkan untuk membuat sistem yang reaktif terhadap perubahan waktu dan musim tanpa perlu memeriksa kondisi secara manual. Ibaratnya seperti sistem buletin otomatis—ketika waktu atau musim berubah, semua objek terkait langsung diberi tahu dan bereaksi sesuai kebutuhannya.

Penerapan di Game:

- Publisher (Subject): Time dan Season
- Observer (Subscriber):
 - Tanaman (Plant): Berlangganan ke Season. Saat musim hujan, tanaman otomatis menjadi watered tanpa perlu penyiraman manual.
 - CookingAction / Masakan: Berlangganan ke Time. Saat waktu masak habis, makanan otomatis siap.
 - Event Harian/Musiman: Event seperti festival atau pasar dapat muncul otomatis berdasarkan waktu.
 - Aksi Pemain (misal tidur/ menambang): Ketika aksi memajukan waktu, semua observer akan diberi tahu dan memperbarui statusnya masing-masing.

Manfaat:

- Low coupling: Time dan Season tidak perlu tahu detail setiap observer-nya.
- Scalable: Menambahkan fitur baru seperti hewan atau efek musiman cukup dengan menjadikannya observer.
- Efisien dan modular, cocok untuk sistem yang bereaksi terhadap perubahan global.

PROSES PENGEMBANGAN TUGAS BESAR



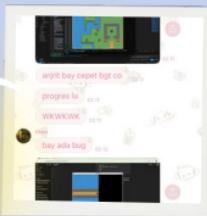
pembagian awal untuk pengerajan class diagram milestone 1



awal-awal progress pengerajan ibay kerjainnya cepet bgt guys, gacor parah ???



tapi ternyata Fathy tbt ilang jadinya ini kita ceritanya lagi bahas tubes bertiga dan untuk milestone 2 ini dikerjain oleh kita bertiga aja :((((



setelah dibuatnya class diagram, kita memutuskan untuk membagi tugas masing-masing seperti ini!!!!



RANCANGAN AWAL DAN APA SAJA PERUBAHAN YANG DILAKUKAN

Tidak terdapat perubahan pola program yang terlalu signifikan dari rancangan awal, namun selama proses pengembangan, kami melakukan banyak refaktorisasi terutama pada struktur item dan objek dalam gameplay. Kami menyadari adanya beberapa variabel dan kebutuhan yang sebelumnya tidak terpikirkan, seperti pentingnya membuat kelas atau objek khusus untuk item tertentu.

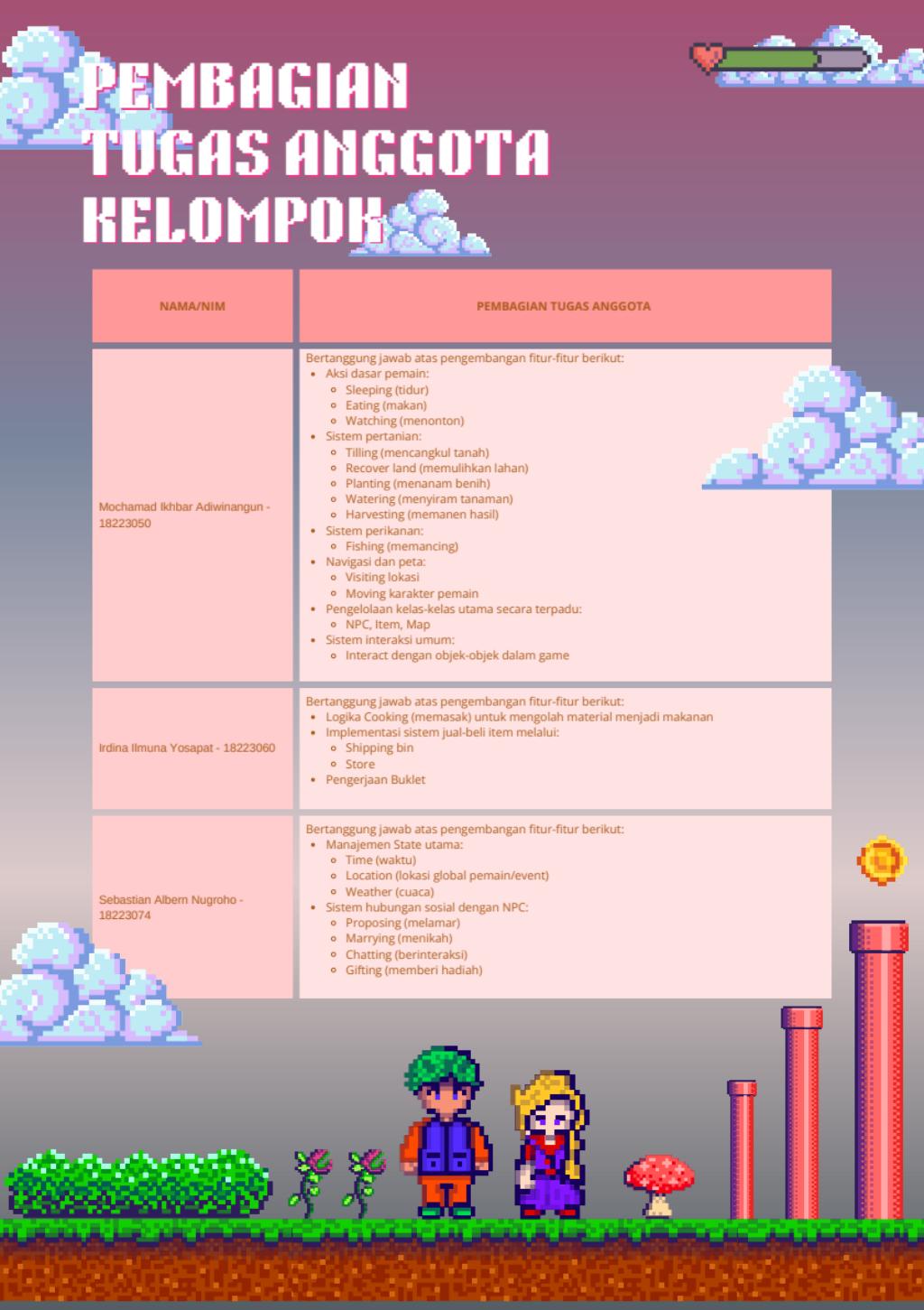
Contohnya, pada rancangan awal, item Misc tidak direncanakan untuk diekstensi lebih lanjut. Namun dalam implementasinya, ternyata terdapat perbedaan karakteristik penting antara Firewood dan Coal sebagai bahan bakar (fuel), sehingga keduanya perlu dipisahkan menjadi kelas tersendiri.

Perubahan ini muncul karena pemahaman kami terhadap kebutuhan program berkembang seiring dengan proses iterasi dan implementasi langsung. Meskipun demikian, secara umum, struktur pemisahan fungsi, atribut, dan metode untuk tiap aksi dalam game sudah cukup konsisten dan sesuai dengan prinsip OOP yang kami rancang sejak awal.



PEMBAGIAN TUGAS ANGGOTA KELOMPOK

NAMA/NIM	PEMBAGIAN TUGAS ANGGOTA
Mochamad Ikhbar Adiwinangun - 18223050	<p>Bertanggung jawab atas pengembangan fitur-fitur berikut:</p> <ul style="list-style-type: none">• Aksi dasar pemain:<ul style="list-style-type: none">◦ Sleeping (tidur)◦ Eating (makan)◦ Watching (menonton)• Sistem pertanian:<ul style="list-style-type: none">◦ Tilling (mencangkul tanah)◦ Recover land (memulihkan lahan)◦ Planting (menanam benih)◦ Watering (menyiram tanaman)◦ Harvesting (memanen hasil)• Sistem perikanan:<ul style="list-style-type: none">◦ Fishing (memancing)• Navigasi dan peta:<ul style="list-style-type: none">◦ Visiting lokasi◦ Moving karakter pemain• Pengelolaan kelas-kelas utama secara terpadu:<ul style="list-style-type: none">◦ NPC, Item, Map• Sistem interaksi umum:<ul style="list-style-type: none">◦ Interact dengan objek-objek dalam game
Irdina Ilmuna Yosapat - 18223060	<p>Bertanggung jawab atas pengembangan fitur-fitur berikut:</p> <ul style="list-style-type: none">• Logika Cooking (memasak) untuk mengolah material menjadi makanan• Implementasi sistem jual-beli item melalui:<ul style="list-style-type: none">◦ Shipping bin◦ Store• Pengerjaan Buklet
Sebastian Albern Nugroho - 18223074	<p>Bertanggung jawab atas pengembangan fitur-fitur berikut:</p> <ul style="list-style-type: none">• Manajemen State utama:<ul style="list-style-type: none">◦ Time (waktu)◦ Location (lokasi global pemain/event)◦ Weather (cuaca)• Sistem hubungan sosial dengan NPC:<ul style="list-style-type: none">◦ Proposing (melamar)◦ Marrying (menikah)◦ Chatting (berinteraksi)◦ Gifting (memberi hadiah)





THANK
YOU

БЕЛОМРОК Ч

