

KELOMPOK 8



ACTION DAN ADVENTURE



PBO RA

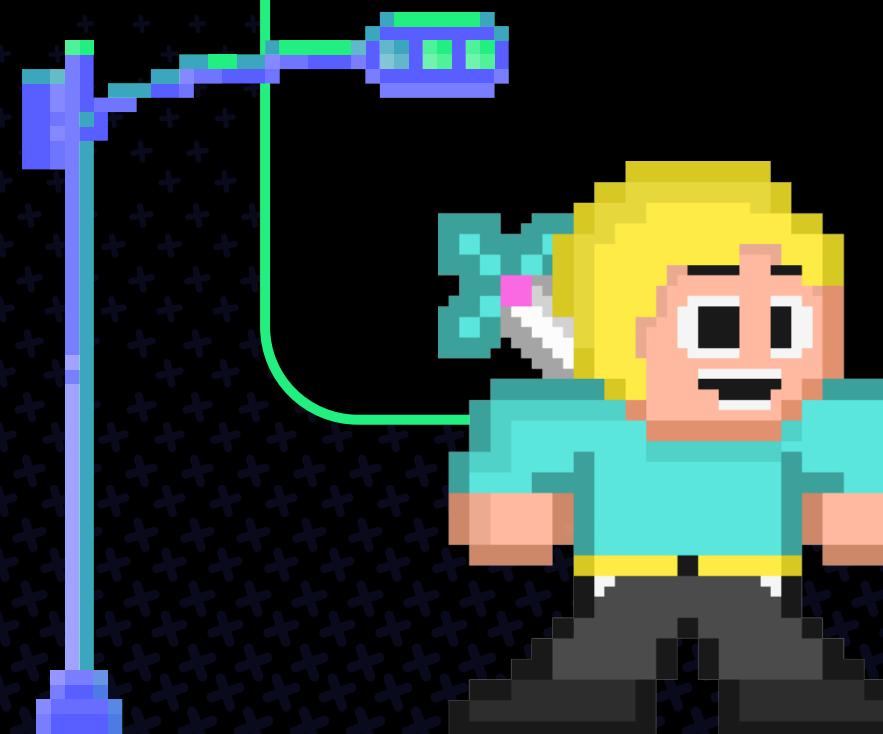
# "JUPPA JUNGLE"

START

MENU

SIGN IN

PRESENTASI TUGAS BESAR PBO



MENU

01

07

12



# KELompok

◆ ANGGOTA :



Fadzilah saputri  
123140149



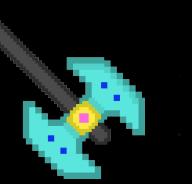
Hildyah maretasya A  
123140151



Martino kelvin  
123140165



Raisya Syifa s  
123140169



ATIKA ADELIA  
123140172



MENU

01

07

12



# NAMA GAME "JUPPA JUNGLE"



Kata 'Jungle' menggambarkan latar permainan yang penuh dengan pepohonan, rintangan alam, dan suasana hutan yang membentuk suatu dunia petualangan di mana tempat player melompat, menghindari musuh, dan mengumpulkan koin. Sedangkan, 'Juppa' merupakan seorang penjelajah pemberani yang harus menavigasi berbagai rintangan untuk mencapai gerbang level berikutnya.

SIGN IN



PLAY



# ALUP GAME

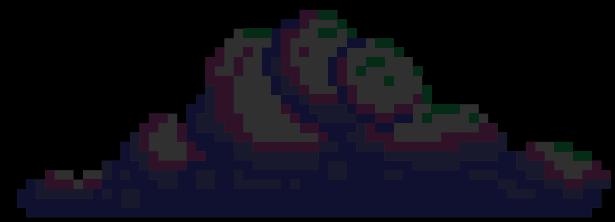
Pemain mengendalikan karakter Juppa untuk bergerak ke kanan, kiri, dan melompat menggunakan keyboard. Koin tersebar di platform dan bisa ditemukan di atas, di antara rintangan, atau di tempat tersembunyi yang memerlukan aksi seperti melompat atau membunuh musuh. Karakter Juppa harus menghindari atau menyerang musuh yang bergerak otomatis; jika Juppa terkena musuh, nyawa akan berkurang. Juppa juga harus melompat antar platform untuk menghindari jatuh ke jurang.



# ENTITAS



PLAYER



ENEMY



HEALTH

COIN

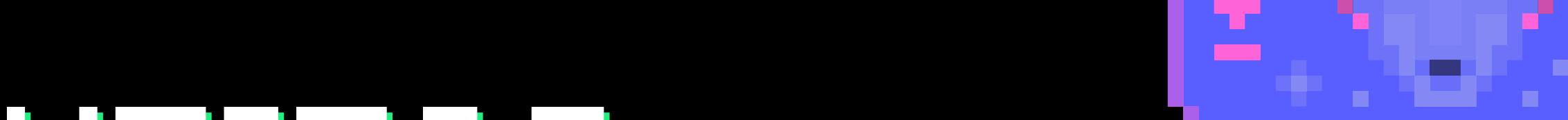


SENJATA



PLATFORMS

GERBANG  
LEVEL



KE HALAMAN AGENDA

01 07 12



# KELAS (CLASS)

GAME

COIN

CREATE

ENEMY

SKY

PALM

PLAYER

TILE

PARTICLEFFECT

LEVEL

ANIMATEDTILE

CLOUDS

UI

STATICTILE

OVERWORLD

# OBJEK



player =  
Player(position)

crate = Crate(Size,  
x, y)

Clouds = Clouds  
(horizon, level\_width,  
cloud\_number)

particle\_effect =  
ParticleEffect(pos,  
type)

tile = Tile(size, x, y)

palm =Palm(size, x,  
y, path, offset)

Node = Node(pos, status,  
icon\_speed, path)

ui = UI(surface)

enemy =  
Enemy(Size, x, y)

sky = Sky(horizon,style)

icon = Icon(pos)

level = Level(current\_level,  
surface, create\_overworld,  
change\_coins,  
change\_health)

coin = Coin(size,  
x, y, path, value)

water = Water(top,  
level\_width)

overworld =  
Overworld(start\_level,  
max\_level, surface,  
create\_level)

game = Game()



★★★★★

SKOR TERTINGGI 2500



PEMAIN 2

# ENKAPSULASI

KELAS

KELAS COIN

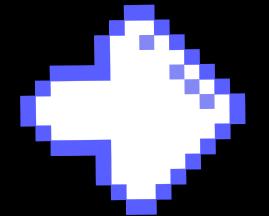
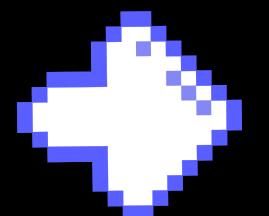
KELAS UI

ATRIBUT

\_VALUE

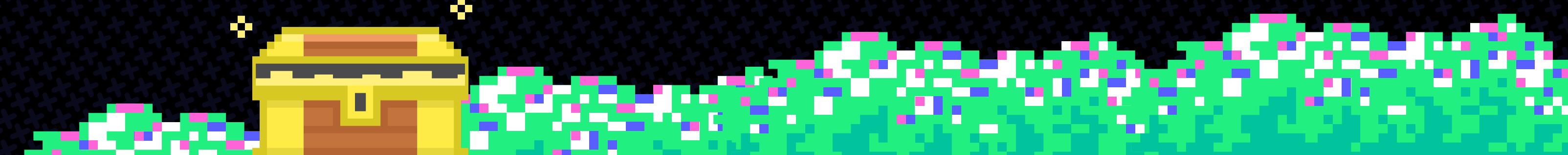
\_CURRENT\_HEALHT

\_FULL\_HEALHT



# PEWARISAN

Kelas Induk	Kelas Turunan
<code>pygame.sprite.Sprite</code>	<code>Player, Tile, Node, Icon, ParticleEffect, Player, Tile</code>
<code>AnimatedTile</code>	<code>Enemy</code>
<code>Tile</code>	<code>Static Tile, AnimatedTile</code>
<code>StaticTile</code>	<code>Crate</code>
<code>AnimatedTile</code>	<code>Coin, Palm</code>



# POLIMORFISME

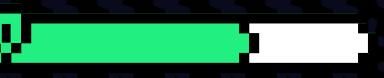
MENU

KONSEP POLIMORFISME DALAM KODE DAPAT DILIHAT DENGAN JELAS PADA METODE GET\_TILE\_TYPE(), GET\_COLLISION\_TYPE(), DAN UPDATE(). KETIGA METODE INI MENCERMINKAN BAGAIMANA SEBUAH METODE YANG MEMILIKI NAMA SAMA DAPAT BERPERILAKU BERBEDA TERGANTUNG PADA OBJEK YANG MENGIMPLEMENTASIKANNYA.



★★★★★

SKOR TERTINGGI 2500

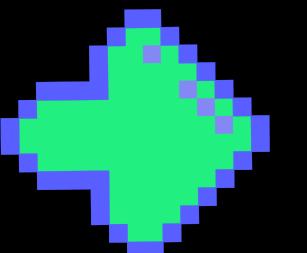


PEMAIN 2

# ABSTRAKSI

KELAS ABSTRAK

TILEINTERFACE



METODE ABSTRAK

\_GET\_TILE\_TYPE

\_GET\_COLLISION\_TYPE

# PENANGANAN KESALAHAN

Penanganan bug dalam game meningkatkan kenyamanan pemain. Salah satu masalah ditemukan pada fitur loncat, di mana karakter melompat ke bawah saat dekat objek seperti kotak atau pohon. Solusinya adalah memperbaiki logika deteksi tabrakan dan arah gaya loncatan agar karakter tetap melompat ke atas. Dengan perbaikan ini, kontrol karakter lebih intuitif dan gameplay terasa lebih responsif.

# GUI



GUI pada game Juppa Jungle dikembangkan menggunakan Pygame, pustaka Python yang mendukung pembuatan game 2D secara mudah dan fleksibel. Pygame dipilih karena antarmukanya sederhana, modulnya lengkap untuk menangani grafik, suara, dan input, serta didukung komunitas aktif dengan dokumentasi melimpah. Meski tidak secepat pustaka dari bahasa tingkat rendah, Pygame memberikan kinerja optimal untuk game 2D. Selain itu, pustaka ini mendukung lintas platform dan memungkinkan proses prototyping cepat, sehingga pengembangan dan penyesuaian desain game dapat dilakukan secara efisien.



START



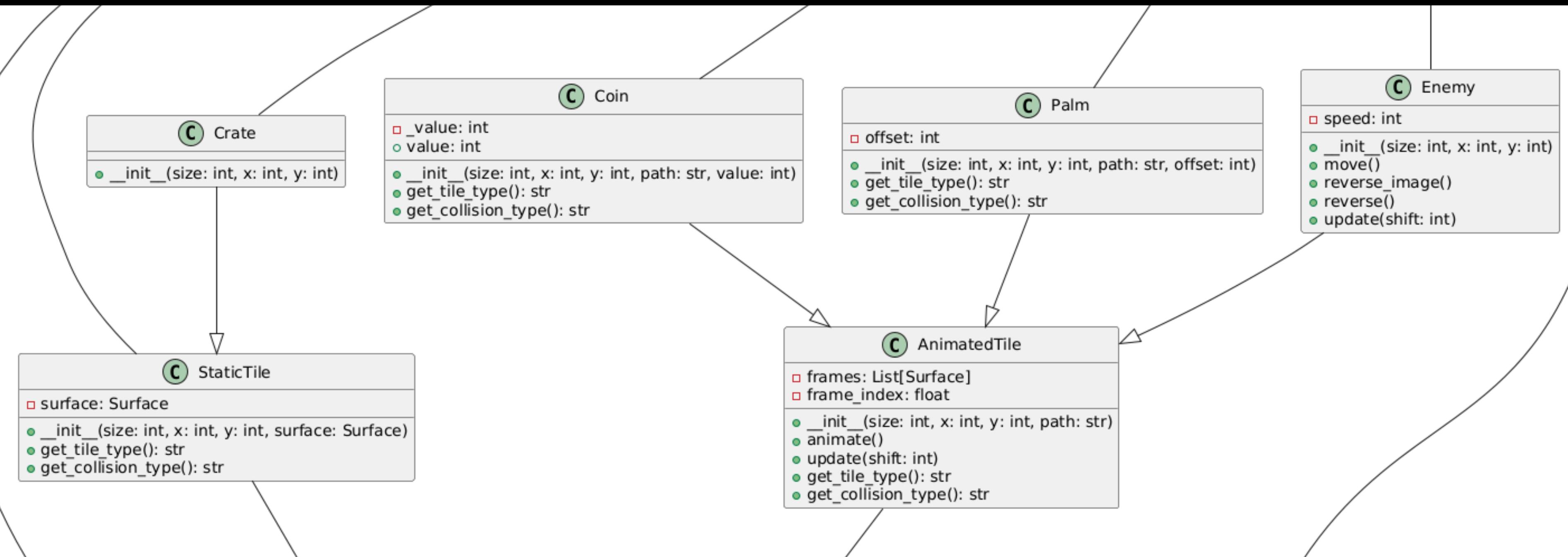
**BARCODE UML JUPPA JUNGLE**  
<https://tinyurl.com/UML-JUPPAJUNGLE-08-RA>

## C Level

```
□ display_surface: Surface
□ world_shift: int
□ current_x: int or None
□ coin_sound: Sound
□ stomp_sound: Sound
□ create_overworld: Callable
□ current_level: int
□ new_max_level: int
□ change_coins: Callable
□ player_on_ground: bool
□ sky: Sky
□ water: Water
□ clouds: Clouds
■ player: GroupSingle(Player)
■ goal: GroupSingle(StaticTile)
■ dust_sprite: GroupSingle(ParticleEffect)
■ explosion_sprites: Group(ParticleEffect)
■ terrain_sprites: Group(Tile)
■ grass_sprites: Group(StaticTile)
■ crate_sprites: Group(Crate)
■ coin_sprites: Group(Coin)
■ fg_palm_sprites: Group(Palm)
■ bg_palm_sprites: Group(Palm)
■ enemy_sprites: Group(Enemy)
■ constraint_sprites: Group(Tile)
● __init__(current_level: int, surface: Surface, create_overworld: Callable, change_coins: Callable, change_health: Callable)
● create_tile_group(layout: List, type: str): Group
● player_setup(layout: List, change_health: Callable)
● enemy_collision_reverse()
● create_jump_particles(pos: Vector2)
● horizontal_movement_collision()
● vertical_movement_collision()
● scroll_x()
● get_player_on_ground()
● create_landing_dust()
● check_death()
● check_win()
● check_coin_collisions()
● check_enemy_collisions()
● run()
```

## C UI

```
□ display_surface: Surface
□ health_bar: Surface
□ health_bar_topleft: Tuple[int,int]
□ bar_max_width: int
□ bar_height: int
□ coin: Surface
□ coin_rect: Rect
□ font: Font
□ _current_health: int
□ _full_health: int
● __init__(surface: Surface)
● get_health(): Tuple[int,int]
● set_health(current: int, full: int)
● show_health(current: int, full: int)
● show_coins(amount: int)
```



**C** Sky

- top: Surface
- middle: Surface
- bottom: Surface
- horizon: int
- style: str
- palms: List[Tuple[Surface, Rect]]
- clouds: List[Tuple[Surface, Rect]]

- \_\_init\_\_(horizon: int, style: str = 'level')
- draw(surface: Surface)

**C** Water

- water\_sprites: pygame.sprite.Group

- \_\_init\_\_(top: int, level\_width: int)
- draw(surface: Surface, shift: int)

**C** Clouds

- cloud\_sprites: pygame.sprite.Group

- \_\_init\_\_(horizon: int, level\_width: int, cloud\_number: int)
- draw(surface: Surface, shift: int)

**C** Tile

□ image: Surface  
□ rect: Rect

● \_\_init\_\_(size: int, x: int, y: int)  
● update(shift: int)  
● get\_tile\_type(): str  
● get\_collision\_type(): str

**C** ParticleEffect

□ frames: List[Surface]  
□ frame\_index: float  
□ animation\_speed: float  
□ image: Surface  
□ rect: Rect

● \_\_init\_\_(pos: Tuple, type: str)  
● animate()  
● update(x\_shift: int)

**C** Player

□ animations: dict[str, List[Surface]]  
□ frame\_index: float  
□ animation\_speed: float  
□ image: Surface  
□ rect: Rect  
□ dust\_run\_particles: List[Surface]  
□ dust\_frame\_index: float  
□ dust\_animation\_speed: float  
□ display\_surface: Surface  
□ create\_jump\_particles: Callable  
□ direction: Vector2  
□ speed: int  
□ gravity: float  
□ jump\_speed: float  
□ collision\_rect: Rect  
□ status: str  
□ facing\_right: bool  
□ on\_ground: bool  
□ on\_ceiling: bool  
□ on\_left: bool  
□ on\_right: bool  
□ change\_health: Callable  
□ invincible: bool  
□ invincibility\_duration: int  
□ hurt\_time: int  
□ jump\_sound: Sound  
□ hit\_sound: Sound

● \_\_init\_\_(pos: Tuple, surface: Surface, create\_jump\_particles: Callable, change\_health: Callable)  
● import\_character\_assets()  
● import\_dust\_run\_particles()  
● animate()  
● run\_dust\_animation()  
● get\_input()  
● get\_status()  
● apply\_gravity()  
● jump()  
● get\_damage()  
● invincibility\_timer()  
● wave\_value()  
● update()

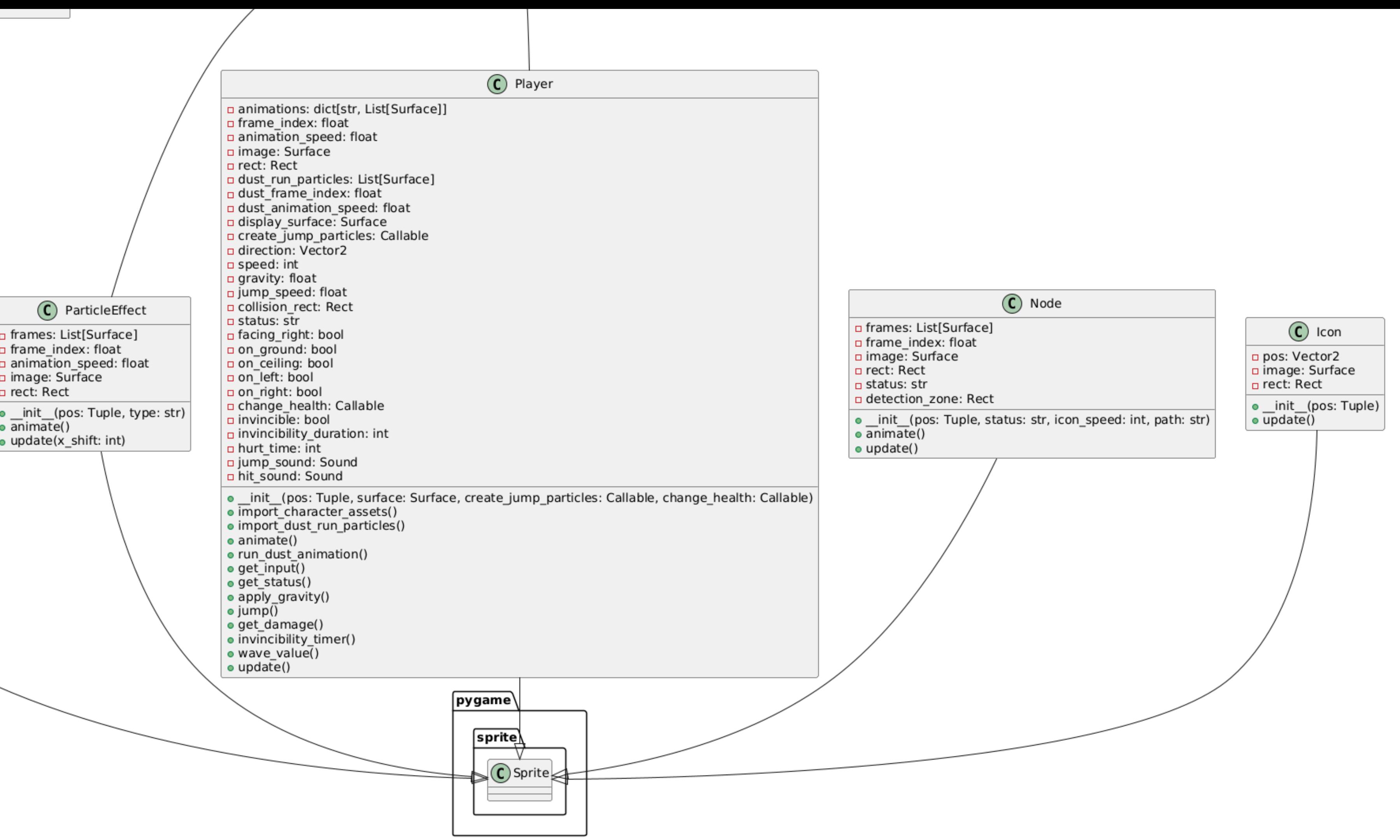
**I** TileInterface

● get\_tile\_type(): str  
● get\_collision\_type(): str

pygame

sprite

**C** Sprite

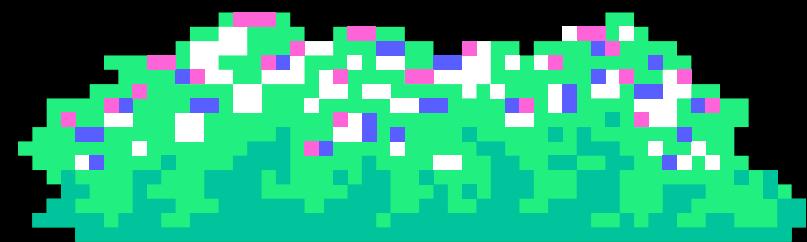
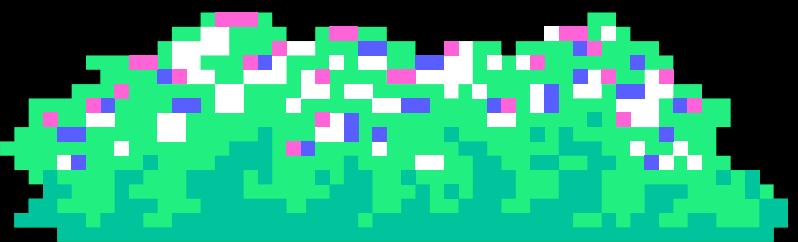


# LAMPIRAN CODE



BARCODE GITHUB JUPPA JUNGLE

<https://tinyurl.com/GitHub-TubesPBO-08-RA>



MAIN

MENU



TERIMA KASIH!