

OOP GROUP PROJECT
TEAM_20

DUNGEON_RPG EDITION

B123245011 楊鎧櫟
B123040015 陳進發
B123245007 許珈瑜

INTRO

01 Character & Game Idea



02 Project Goal & OOP Architecture

03 Character & Skill UML_character

04 DungeonBattleEnv, Round Turn & Agent / Strategy

05 Pygame Demo & Reflection



Character & Game Idea

01

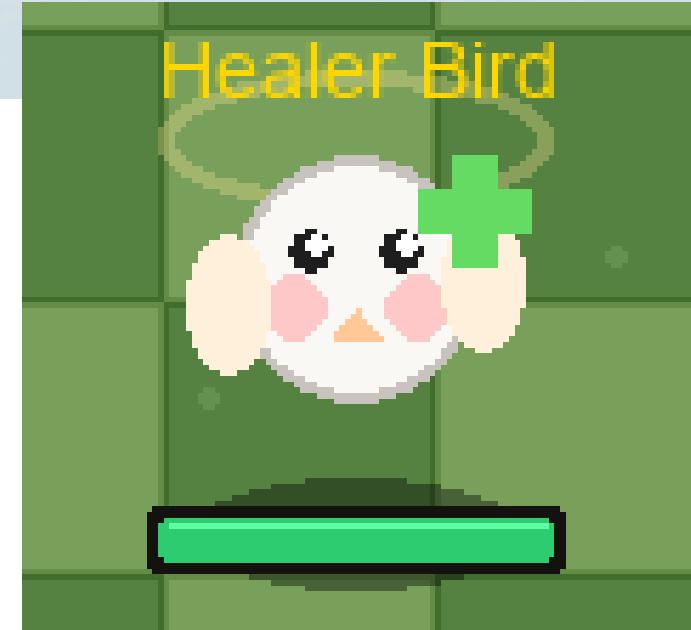


Character



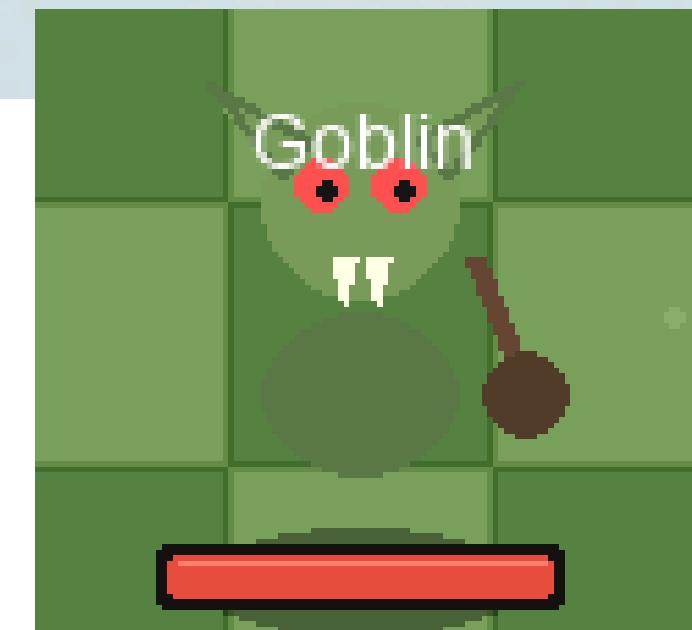
Kiwi bird

Main Character



Healer bird

Teammate



Goblin

Mobs



Green Slime

Mobs



Fire Dragon

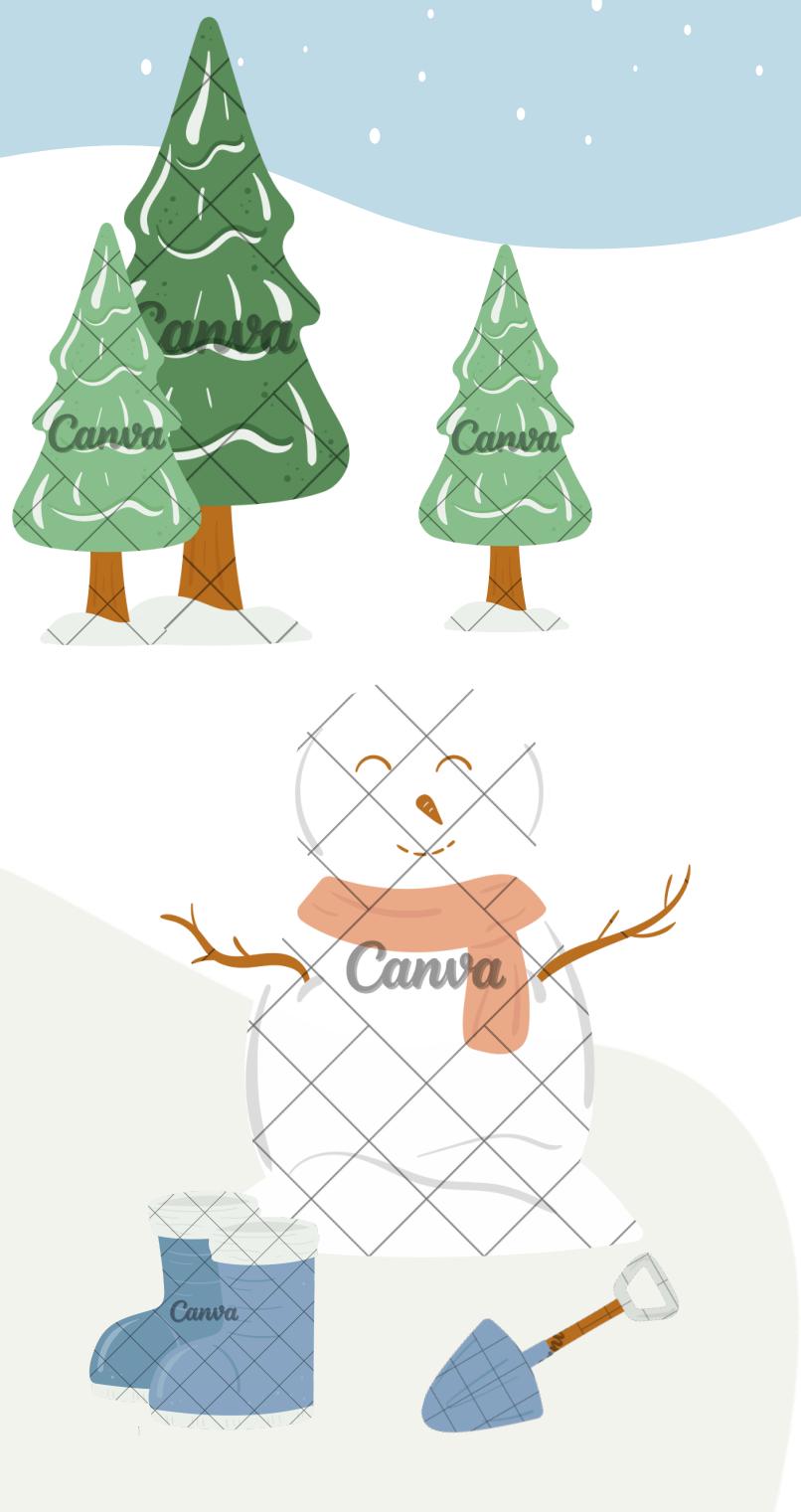
Boss

GAME IDEA

3 Level RPG Of Dungeon

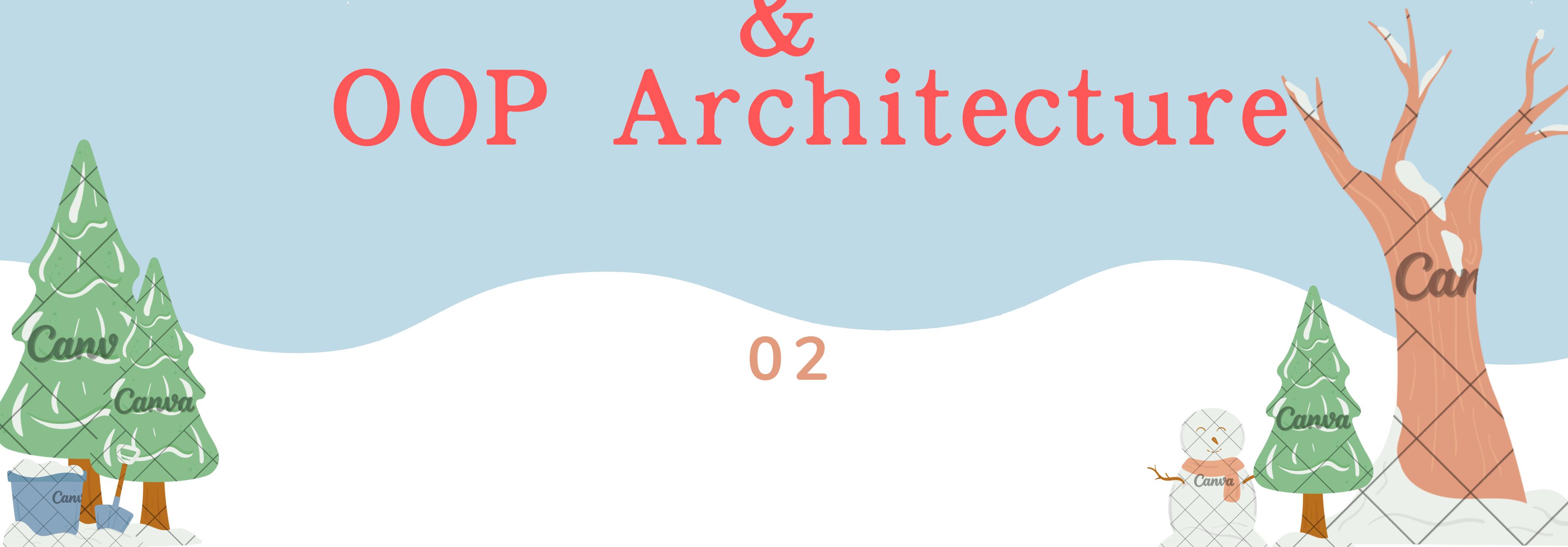
Battle Based On Rounds

Auto Or Player Control

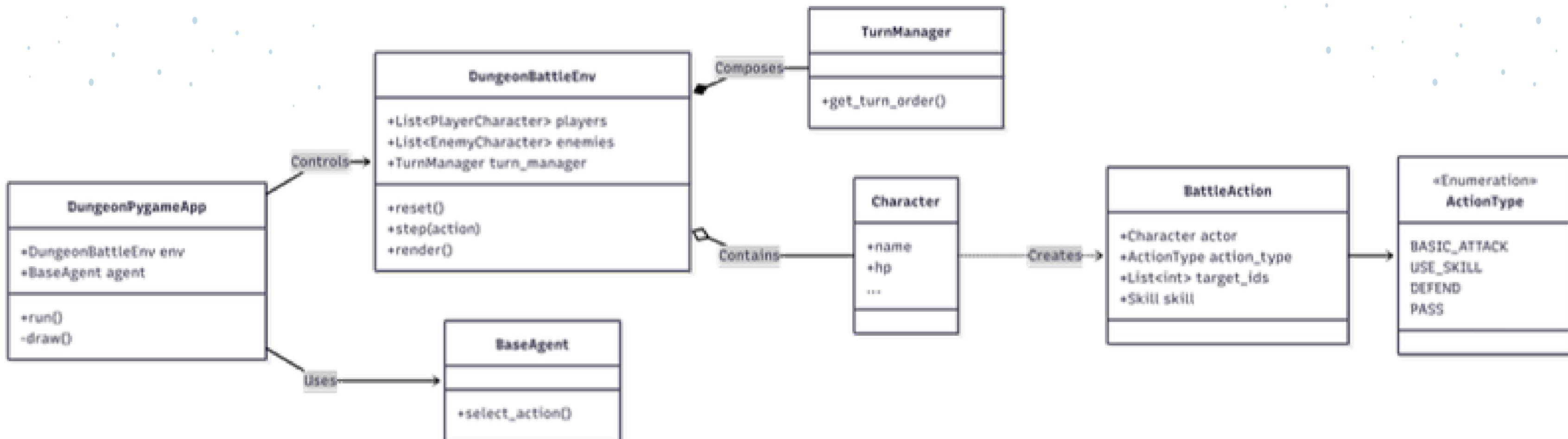


Project Goal & OOP Architecture

02



UML



PROJECT GOAL

- 專案目標
 - 用實際遊戲系統展示：**封裝、繼承、多型、策略模式**
 - 把關卡做成類似 Gym 的自訂環境：DungeonBattleEnv
 - 讓不同 Agent 可以控制主角，每回合丟 action 給 env

OOP ARCHITECTURE

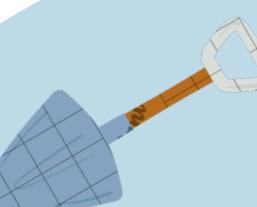
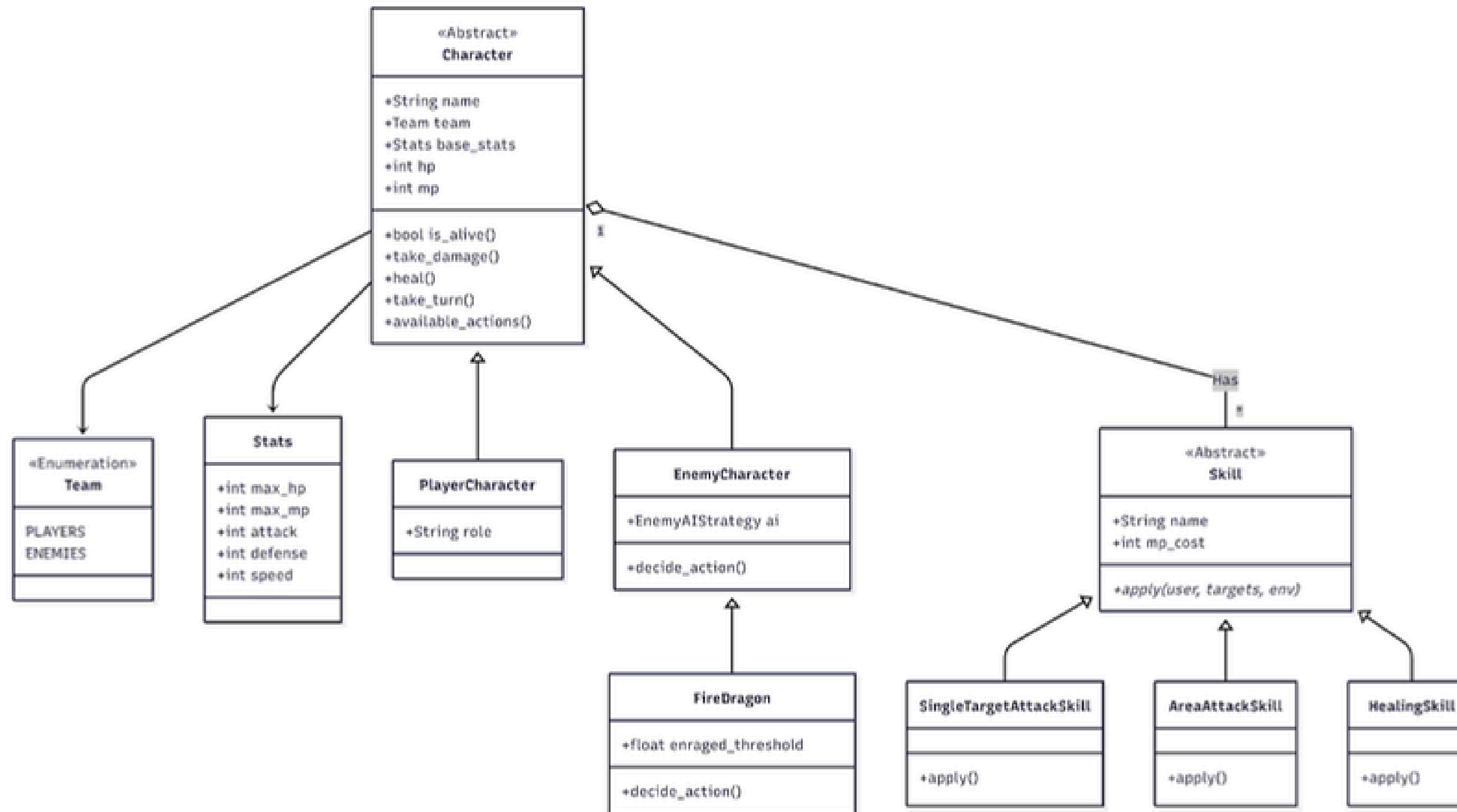
- 分層設計
 - DungeonPygameApp：只負責 UI (pygame 畫面 & 輸入)
 - DungeonBattleEnv：規則、樓層、回合流程、reward
 - TurnManager / Character / BattleAction：管理出手順序、角色狀態與動作
- BaseAgent：select_action(env, player) 統一 Agent 介面

Character & Skill

03



UML



CHARACTER

- 角色階層
 - Character (抽象)：共用欄位 & 方法
 - name, team, Stats、hp/mp
 - take_damage(), heal(), is_alive(), available_actions()
 - 子類別：PlayerCharacter、EnemyCharacter、FireDragon (Boss 有額外行為)
- 組合 (Composition)

Character **has** Stats & Team enum，而不是繼承數值類別

SKILL

- 技能階層
 - Skill (抽象) : apply(user, targets, env)
 - 子類 : SingleTargetAttackSkill、AreaAttackSkill、HealingSkill

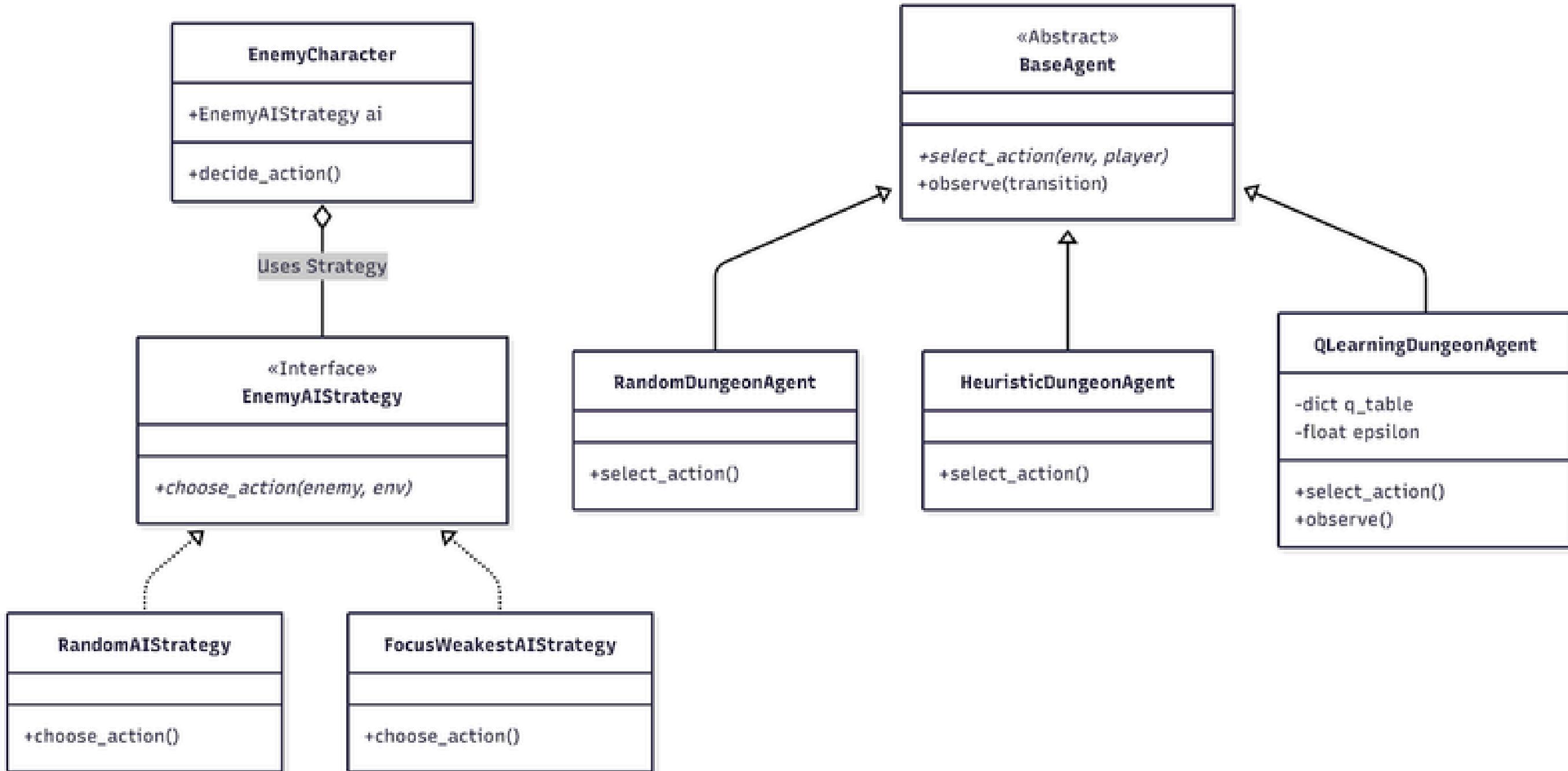
同一個 apply() 介面，依子類呈現不同效果（攻擊／範圍／補血）

DungeonBattleEnv Round, Turn & Agent / Strategy

04



UML



DungeonBattleEnv

- DungeonBattleEnv
 - reset(floor_idx): 切換樓層，生成 Kiwi 隊友 & 敵人
 - step(player_action): 完成一整個回合
- render(): 提供狀態給文字版／pygame 介面顯示

回合流程

- 回合流程
 - TurnManager 依 speed 計算出手順序
 - 玩家：
 - 手動模式 → UI 產生 BattleAction
 - 自動模式 → BaseAgent.select_action(env, player)
 - 敵人：EnemyAIStrategy.choose_action(enemy, env)
- _apply_action() 更新 HP／死亡、檢查通關或滅團、計算 reward

Agent & Strategy Pattern

- Agent & Strategy Pattern
 - 玩家 Agent : Random / Heuristic / Q-learning (簡化 Q-table, 參考 Part 2 FrozenLake)
 - 敵人 AI : RandomAIStrategy (亂打)、FocusWeakestAIStrategy (追殺殘血)

Pygame Demo & Reflection

05



DEMO TIME

python pygame_app.py : 2D dungeon

TOP : Enemy & Boss

BELLOW : Kiwi & Teammate, action button

Switch : Player control / Auto、 Random / Heuristic / Q-learning mode

Show : Effects of Attack, Heal, Fire(antigravity particle effect)

REFLECTION

- **What we learn :**
 - how to Transfer game rules to Env / Agent / Strategy
 - inheritance, OOP concepts
- **Improvement :**
 - Improve Q-learning, let it know how to fight Boss
 - More effects or characters/class

MERRY CHRISTMAS ~~~

THE END

THANKS FOR LISTENING