

Network Programming HW2 - 雙人俄羅斯方塊

蔡懷恩 112550020

1. 系統架構

系統由四個元件組成：

1. Lobby Server: 負責帳號註冊、登入、房間管理、和DB Server交流等功能
2. DB Server: 使用SQLite, 支援User / Room / GameLog CRUD
3. Game Server: 雙人遊戲邏輯同步、快照廣播、結束回報
4. Client (LobbyClient + GameGUI + main): 使用者操作 + Pygame 顯示

伺服器只需要架設 lobby_server.py 和 db_server.py

客戶端遊玩使用 main.py

假設 db_server和lobby_server運行在同一台上,
更改ip需更改 lobby_server.py 和 main.py

2. 協定格式(Protocol、封包格式)

Length-Prefixed Framing Protocol (in common.py):

Header: 4 bytes

Body: UTF-8 JSON 字串

發送: (struct.pack('!l', length) + data)

封包格式:

Client -> Lobby Server:

```
{
    "action": "register / login / create_room / join_room / start_game / ... ",
    "data": {"key": value}      // 根據需要放data, e.g. room_id, name
}
```

Lobby Server -> DB Server:

```
{
    "collection": "User / Room / GameLog", // DB Table
    "action": "create / read / update / query",
    "data": {"key": value}      // 根據需要放data, e.g. room_id, users
}
```

(DB Server ->) Lobby Server response-> Client: // 偶爾轉傳DB_server訊息

```
{
    "ok": True / False,
    "msg": ""
}
```

Lobby Server 廣播 -> (特定) Client: // 有時候有限定給誰

```
{
    "type": "SYSTEM / ROOM / GAME_START / INVITE",
    "msg": "{user} joined the lobby."
}
```

處理User和Game Server連線：

GameGUI -> Game Server: {"userId": self.username}

Game Server -> GameGUI:

```
{
    "type": "WELCOME",
    "role": player.role, //p1, p2
    "seed": seed,
    "bagRule": "7bag",
    "msg": f"Welcome {player.role}"
}
```

GameGUI -> Game Server:

```
{
    "type": "INPUT",
    "action": move,
    "userId": self.username
}
```

Game Server 廣播快照 -> GameGUI: // 快照間隔 = 0.5

```
{
    "type": "SNAPSHOT",
    "tick": self.tick,
    "timestamp": time.time(),
    "players": [
        {
            "name": p.name,           // 名稱
            "score": p.logic.score,   // 分數
            "alive": p.logic.alive,   // 是否存活
            "board": p.logic.get_combined_board(), // 棋盤
            "active": p.logic.active  // 正在下落的方塊的狀態
        }
        for p in self.players
    ]
}
```

3. 同步策略

遊戲同步方式: Server Authority + Snapshot

- Server Authority:
所有遊戲邏輯(方塊生成、移動、碰撞、消行、得分)由 Game Server 執行。
Client 僅負責將輸入事件(例如旋轉、左右移動)傳送給伺服器。
- 輸入事件(INPUT):
Client 每次按鍵即時傳送
{ "type": "INPUT", "action": move , "userId": username }
- 快照同步(SNAPSHOT):
伺服器每 0.5 秒 廣播一次遊戲全狀態:包含兩位玩家棋盤、方塊、分數。
Client 端以最新快照覆蓋顯示, 確保畫面一致。
- Tick 更新頻率:
每 0.5 秒 執行一次 soft_drop() → 模擬重力。

4. 玩法、規則

項目	說明
棋盤大小	10 × 20
控制操作	方向鍵 ← → 移動 / ↓ Soft Drop / ↑ Hard Drop / Z, X 旋轉
方塊生成規則	7-Bag 隨機(雙方使用相同 seed)
重力	固定(每 tick 自動下落)
得分方式	每消一行 +100 分
結束條件	兩位玩家皆死亡(頂滿)或時間超過 120 秒
勝負判定	分數高者勝, 平分則平手
無攻擊機制	雙方獨立棋盤、不互相影響