

---

1. Responsibilities

陳映紅：Server端實作、Player（Client端）GUI 設計實作、修圖、網路溝通架構

白謹瑞：Player（Client端）網路、GUI設計實作、網路溝通結構

郭哲希：判斷遊戲規則（Map）、試寫Server/Client、Report

2. Relations Between Classes

- Card
  - ActionCard
    - BlockCard：使對手無法出pathcard
    - BombCard：將場上的一張pathcard移除
    - RepairCard：將自己或其他人的BlockCard移除
    - MapCard：看GoalCard的其中一張
  - PathCard
    - 包含各式方向（包括不通的路）共18種
    - 卡片可180度轉向
- Map：儲存場上卡牌、判斷場上卡牌動作是否合理、執行牌組動作
- Data：Server與Player間傳輸的訊息Object。
- Dealer：發牌。
- Server：從Player中取得資料、判斷合理性、更改Map，並回傳Player適當的值。
  - 使用到Map、Dealer、Data
- Player：遊戲界面、從玩家取得資料、傳給Server、更新Map與場上活動。
  - 使用到Data

3. Advantages

工作分配清楚：

將所有判斷都交給Server與Map處理，Client端的Player只負責接收訊息與更新界面。

卡片管理：

不同種類的卡片有不同的method，但是都是Card，所以可以放在同副手牌裡。

4. Disadvantages

因為Card extend太多種class，使得在call的時候會顯得冗長。

切的不夠細，有時因為要改的東西在同一份檔案（Player.java）導致分工不易。

5. How to Play

- Server: make run\_Server
- Player: make run\_Player(wait until 5 people connected)
- 抽角色：Miner (好人) / Saboteur (壞人)
  - Miner：從點連接到終點目標牌，獲得其中一張有金礦的目標牌
  - Saboteur：阻止好人達成目標。

- 抽牌：ActionCard / PathCard (手牌共6張，每次抽牌，直到所有牌皆發完且無法使用)
  - ActionCard：
    - MapCard、BombCard
      - 使用方法：先按卡片，再按地圖任一處觸發
      - MapCard會在右上角顯示此張牌是否為Gold
      - BombCard可以把場上一張PathCard移除。
    - BlockCard、RepairCard
      - 使用方法：先按卡片，再按任一玩家。
      - BlockCard使被使用方不能出PathCard，需要RepairCard解鎖
      - 有三種不同的Block/RepairCard，只有同種才能Repair成功



玩家2使用Block牌Block玩家3，只有同種的Repair卡才能夠修復、繼續遊戲。

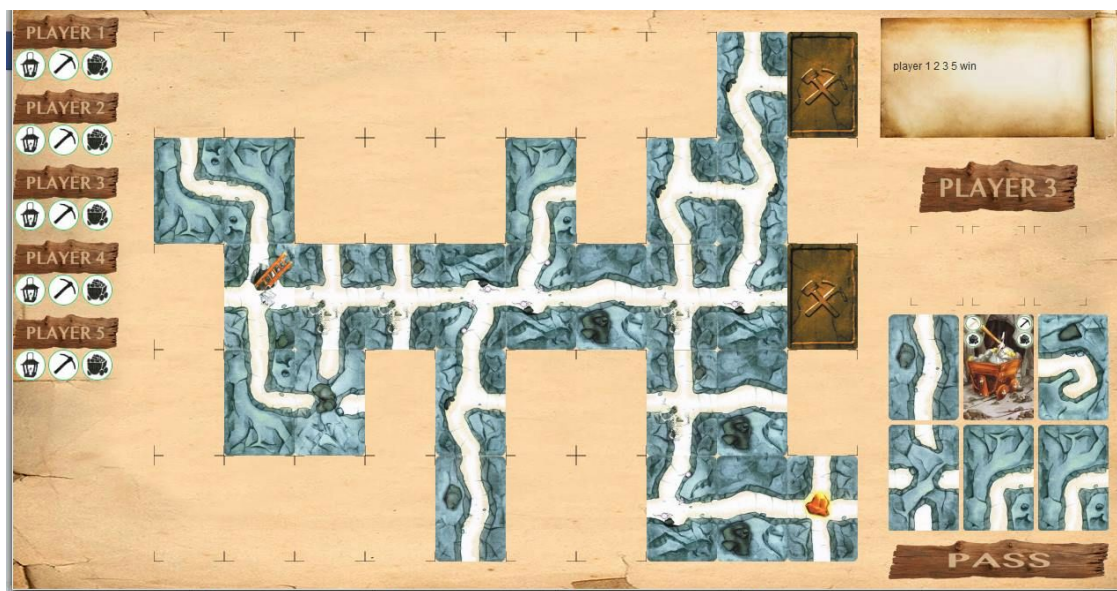


玩家3被Block，只能出ActionCard或是Pass。



玩家4使用mapcard，得知上面的goalcard不是gold.

- PathCard :
  - 使用方法 :
    - 點擊卡片、再按地圖上一處，把PathCard放到地圖上。
    - 雙擊則旋轉180度
    - 若該動作無效（無法接通），則顯示在左上角，重新選擇卡片
  - 只能放在從起點能連接到的地方（使用DFS）、且週邊與之相連的道路必須連通。
  - 三張GoalCard中只有一個是Gold，其他兩個都是普通的Stone。



連接到兩個石頭並沒有勝利，須連接到Gold好人才獲勝。

- 遊戲動作：放牌、Pass
- 結束：
  - 好人抵達金礦：好人勝利
  - 好人未抵達金礦，牌已用盡：壞人勝利