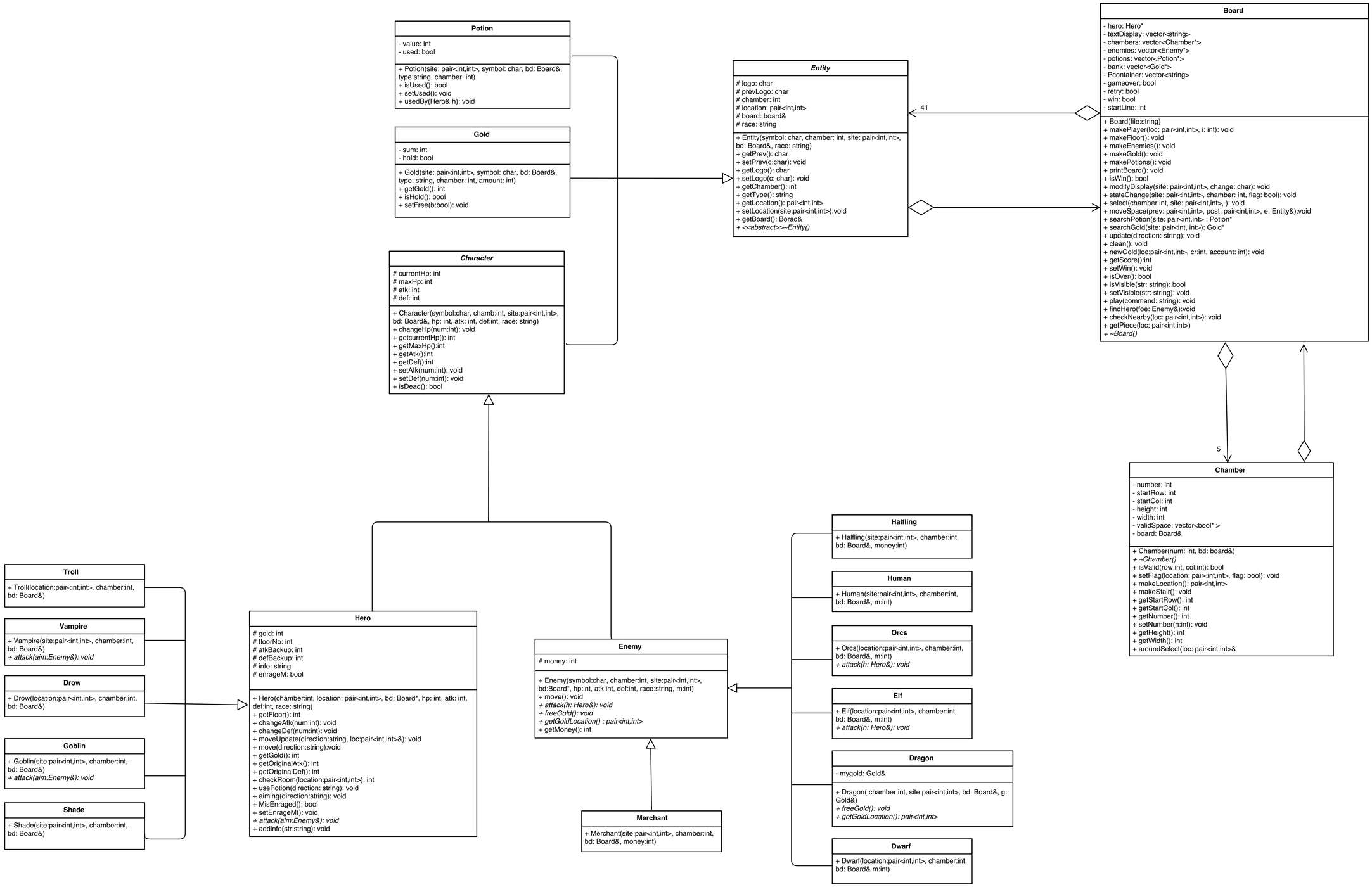
Potion
- value: int
- used: bool
+ Potion(site: pair<int,int>, symbol: char, bd: Board&, type:string, chamber: int)
+ isUsed(): bool
+ setUsed(): void
+ usedBy(Hero& h): void
Gold
- sum: int
- hold: bool
+ Gold(site: pair<int,int>, symbol: char, bd: Board&, type: string, chamber: int, amount: int)
+ getGold(): int
+ isHold(): bool
+ setFree(b:bool): void
Character
currentHp: int
maxHp: int
atk: int
def: int
+ Character(symbol:char, chamb:int, site:pair<int,int>, bd: Board&, hp: int, atk: int, def:int, race: string)
+ changeHp(num:int): void
+ getcurrentHp(): int
+ getMaxHp():int
+ getAtk():int
+ getDef():int
+ setAtk(num:int): void
+ setDef(num:int): void
+ isDead(): bool
Entity
logo: char
prevLogo: char
chamber: int
location: pair<int,int>
board: board&
race: string
+ Entity(symbol: char, chamber: int, site: pair<int,int>, bd: Board&, race: string)
+ getPrev(): char
+ setPrev(c:char): void
+ getLogo(): char
+ setLogo(c: char): void
+ getChamber(): int
+ getType(): string
+ getLocation(): pair<int,int>
+ setLocation(site:pair<int,int>):void
+ getBoard(): Borad&
+ <<abstract>>~Entity()
41
Board
- hero: Hero*
- textDisplay: vector<string>
- chambers: vector<Chamber*>
- enemies: vector<Enemy*>
- potions: vector<Potion*>
- bank: vector<Gold*>
- Pcontainer: vector<string>
- gameover: bool
- retry: bool
- win: bool
- startLine: int
+ Board(file:string)
+ makePlayer(loc: pair<int,int>, i: int): void
+ makeFloor(): void
+ makeEnemies(): void
+ makeGold(): void
+ makePotions(): void
+ printBoard(): void
+ isWin(): bool
+ modifyDisplay(site: pair<int,int>, change: char): void
+ stateChange(site: pair<int,int>, chamber: int, flag: bool): void
+ select(chamber int, site: pair<int,int>,): void
+ moveSpace(prev: pair<int,int>, post: pair<int,int>, e: Entity&):void
+ searchPotion(site: pair<int,int> : Potion*
+ searchGold(site: pair<int, int>): Gold*
+ update(direction: string): void
+ clean(): void
+ newGold(loc:pair<int,int>, cr:int, account: int): void
+ getScore():int
+ setWin(): void
+ isOver(): bool
+ isVisible(str: string): bool
+ setVisible(str: string): void
+ play(command: string): void
+ findHero(foe: Enemy&):void
+ checkNearby(loc: pair<int,int>): void
+ getPiece(loc: pair<int,int>)
+ ~Board()
5
Chamber
- number: int
- startRow: int
- startCol: int
- height: int
- width: int
- validSpace: vector<bool* >
- board: Board&
+ Chamber(num: int, bd: board&)
+ ~Chamber()
+ isValid(row:int, col:int): bool
+ setFlag(location: pair<int,int>, flag: bool): void
+ makeLocation(): pair<int,int>
+ makeStair(): void
+ getStartRow(): int
+ getStartCol(): int
+ getNumber(): int
+ setNumber(n:int): void
+ getHeight(): int
+ getWidth(): int
+ aroundSelect(loc: pair<int,int>&
Troll
+ Troll(location:pair<int,int>, chamber:int, bd: Board&)
Vampire
+ Vampire(site:pair<int,int>, chamber:int, bd: Board&)
+ attack(aim:Enemy&): void
Drow
+ Drow(location:pair<int,int>, chamber:int, bd: Board&)
Goblin
+ Goblin(site:pair<int,int>, chamber:int, bd: Board&)
+ attack(aim:Enemy&): void
Shade
+ Shade(site:pair<int,int>, chamber:int, bd: Board&)
Hero
gold: int
floorNo: int
atkBackup: int
defBackup: int
info: string
enrageM: bool
+ Hero(chamber:int, location: pair<int,int>, bd: Board*, hp: int, atk: int, def:int, race: string)
+ getFloor(): int
+ changeAtk(num:int): void
+ changeDef(num:int): void
+ moveUpdate(direction:string, loc:pair<int,int>&): void
+ move(direction:string):void
+ getGold(): int
+ getOriginalAtk(): int
+ getOriginalDef(): int
+ checkRoom(location:pair<int,int>): int
+ usePotion(direction: string): void
+ aiming(direction:string): void
+ MisEnraged(): bool
+ setEnrageM(): void
+ attack(aim:Enemy&): void
+ addinfo(str:string): void
Enemy
money: int
+ Enemy(symbol:char, chamber:int, site:pair<int,int>, bd:Board*, hp:int, atk:int, def:int, race:string, m:int)
+ move(): void
+ attack(h: Hero&): void
+ freeGold(): void
+ getGoldLocation() : pair<int,int>
+ getMoney(): int
Merchant
+ Merchant(site:pair<int,int>, chamber:int, bd: Board&, money:int)
Halfling
+ Halfling(site:pair<int,int>, chamber:int, bd: Board&, money:int)
Human
+ Human(site:pair<int,int>, chamber:int, bd: Board&, m:int)
Orcs
+ Orcs(location:pair<int,int>, chamber:int, bd: Board&, m:int)
+ attack(h: Hero&): void
Elf
+ Elf(location:pair<int,int>, chamber:int, bd: Board&, m:int)
+ attack(h: Hero&): void
Dragon
- mygold: Gold&
+ Dragon(chamber:int, site:pair<int,int>, bd: Board&, g: Gold&)
+ freeGold(): void
+ getGoldLocation(): pair<int,int>
Dwarf
+ Dwarf(location:pair<int,int>, chamber:int, bd: Board& m:int)