

Entities & Responsibilities

GameWorld: Manages game initialisation & admin tasks (e.g game config, game setup, main game loop, player turn.)

Tile: Represents the tiles which the dragons will interact with (stand on)

PlayableEntity: Represents the playable entity a player interacts with

GameBoard: Runs the interactions with the game board by the players (e.g performing movement, flipping chit card)

ChitCard: Represents the chit cards and their effects

EventBus: Handles registration of listeners, and notification of appropriate listeners on event fire

WinEventXxxx: Publisher and listeners for win event

Drawable: Indicates that the object is drawable by pygame

DrawInstruction: A data class for organising data required for drawing

Patterns Used

Observer: WinEventPublisher, WinEventListener  
• Why?: Don't have to check all starting tiles to see if win occurred. Allows for wins from other sources

Singleton: EventBus  
• Why?: Should be one central event bus managing all events

Todo

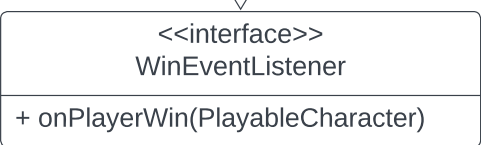
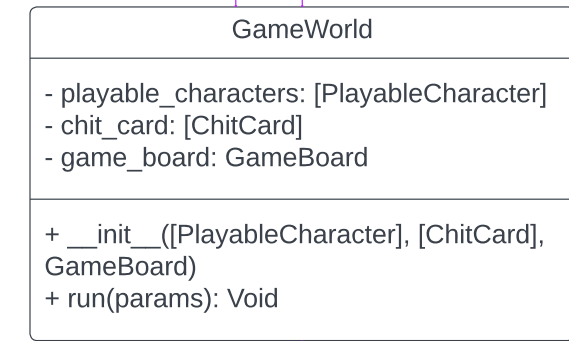
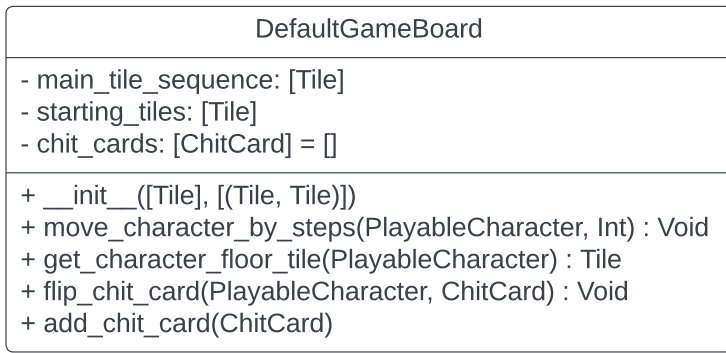
Cardinalities

Notes

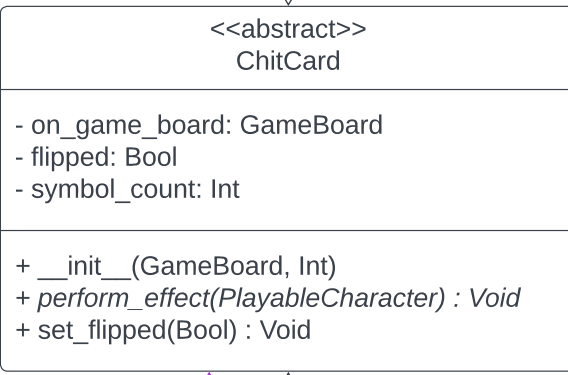
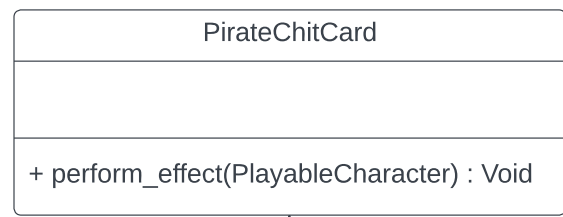
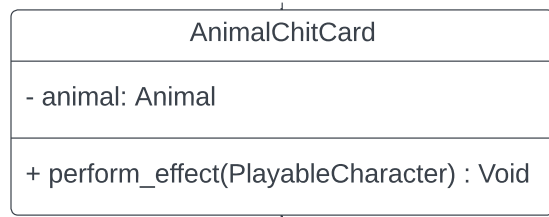
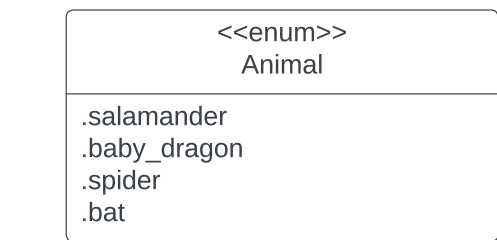
Upcasts are safe

GameApplication.\_\_init\_\_: Configure and initialise playable characters, game board & its tiles & its chit cards.  
GameApplication.run: Contains the main game loop. Handles policy of player turns, and drawing logic (based on drawable)

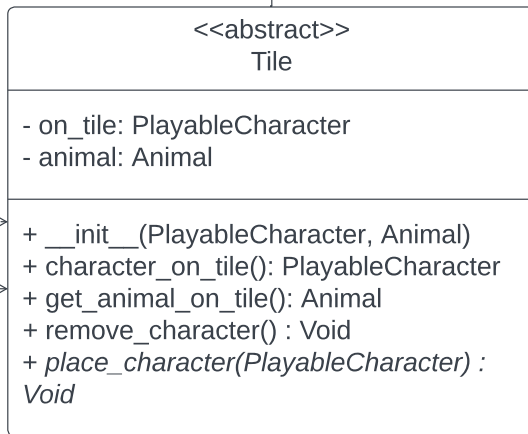
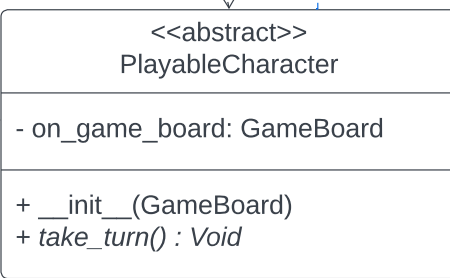
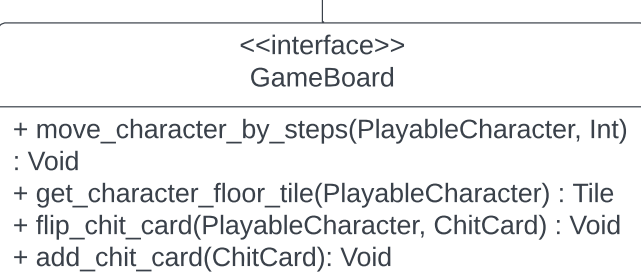
DefaultGameBoard.\_\_init\_\_: First parameter is set main tile sequence. Second parameter is [(starting tile, next tile)]. The second parameter (starting tiles) will be used to make insertions to main tile sequence property.  
DefaultGameBoard.starting\_tiles: [starting tiles]  
DefaultGameBoard.move\_character\_by\_steps: Moves a dragon from its tile by x number of steps  
DefaultGameBoard.flip\_chit\_card: Flip a chit card on this game board. Chit card object will be obtained by click listener in pygame.  
DefaultGameBoard.add\_chit\_card: Add a chit card to the game board



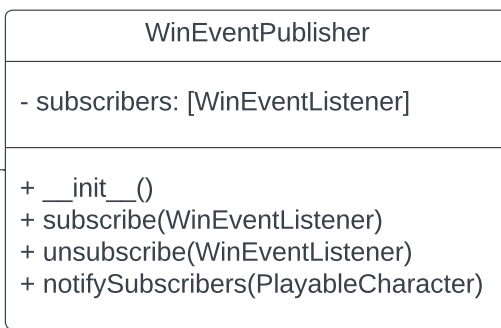
WinEventListener.onPlayerWin: On a player win, do something



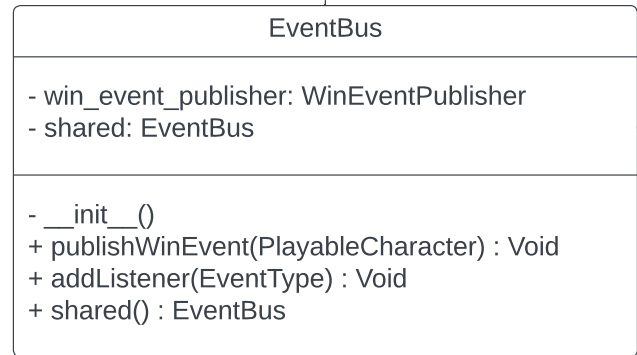
ChitCard.\_\_init\_\_: Create a chit card with the # number of symbols  
ChitCard.perform\_effect: Perform the chit card's effect on the dragon.  
Run by flipChitCard  
ChitCard.set\_flipped: Set flipped



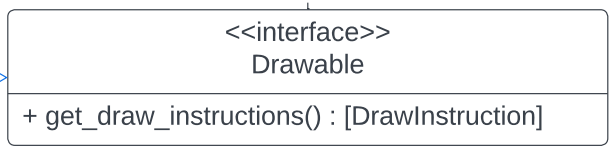
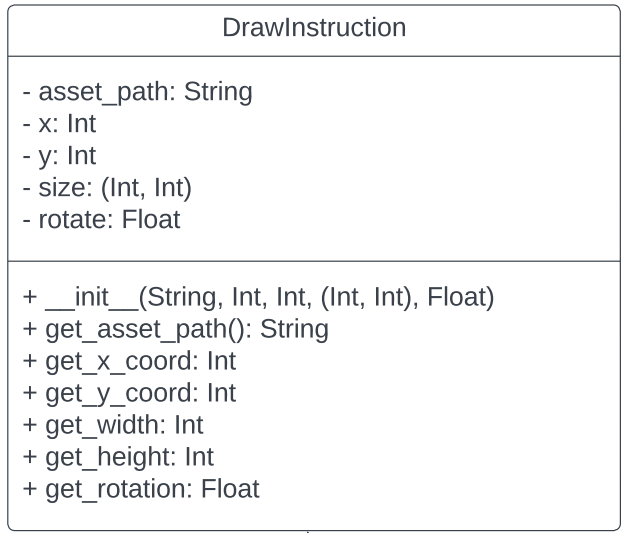
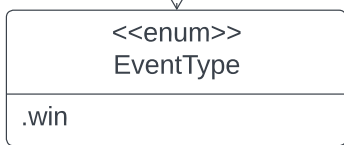
Tile.\_\_init\_\_: Initialise tile with playable character (dragon) and animal  
Tile.place\_character: Place dragon onto tile and perform related functionalities  
Tile.remove\_character: Remove the dragon that's on the tile (if any)  
Tile.character\_on\_tile: Returns the dragon on the tile (if any)



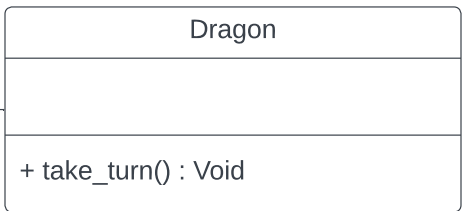
WinEventPublisher.notifySubscribers: Notify subscribers about the player who won



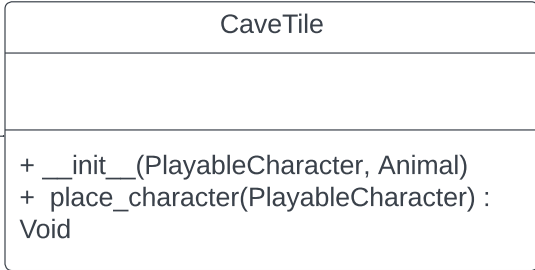
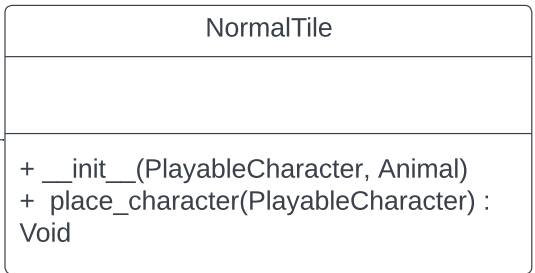
EventBus.shared: Get the eventbus singleton  
EventBus.publishWinEvent: Publish win event onto bus  
EventBus.addListener: Add a listener for a certain event type



Drawable.get\_draw\_instructions: Gets the drawing instructions for drawing an object onto the pygame screen



Dragon.take\_turn: Allows a dragon to take its turn & perform actions (e.g chit card flipping)



CaveTile.place\_character: Place dragon onto tile and perform check whether it was return back to own cave, trigger win