

TODO

Builder pattern for configuring game board?

(e.g performing movement, flipping chit card)

ChitCard: Represents the chit cards and their effects

Entities & Responsibilities

Cardinalities

Have to check this one with pygame (can I do this? or do I have to use integer type?). Prefer ChitCard due to strong typing over weak integer

GameApplication: Manages game initialisation & admin tasks (e.g game

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

config, game setup, main game loop, player turn.)

Dragon Dragon.take_turn: Allows a dragon to take its turn & perform actions (e.g chit card flipping) + take_turn() : Void + __init__(GameBoard) + get_character_floor_tile(PlayableCharacter) : Tile + take_turn() : Void + flip_chit_card(PlayableCharacter, ChitCard) : Void + ___init___(Int) + run(params): Void NormalTile <<abstract>> + __init__(PlayableCharacter, Animal) GameApplication.__init__: Configure and intialise playable + place_character(PlayableCharacter): characters - on_tile: PlayableCharacter GameApplication.run: Contains the main game loop, and initialises - animal: Animal the GameBoard (tile sequence, starting tiles). Handles policy of player turns + __init__(PlayableCharacter, Animal) + character_on_tile(): PlayableCharacter CaveTile + get_animal_on_tile(): Animal + remove_character() : Void + place_character(PlayableCharacter) : Void + __init__(PlayableCharacter, Animal) + place_character(PlayableCharacter) : 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)

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