UNO code-overview

By UNOrganized

In this file you can find an overview over the folders and files that may have changed compared to the example project we were given. This file contains all member variables and member functions used in the changed or newly written code.

The used colors correspond to:
black: folders
green: code files (.h or .cpp)
blue: member variables
red: member functions or name of tests

- src
  - common
    - exceptions
      - UnoException
        - std::string _msg
        - UnoException
        - char* what
    - game_state
      - cards
        - card
          - serializable_value<int>* _index
          - serializable_value<int>* _color
          - card
          - ~card
          - int get_index
          - int get_color
          - bool can_be_played_on
          - void change_color
          - void write_into_json
          - card* from_json
        - discard_pile
          - std::vector<card*> _cards
          - discard_pile
          - ~discard_pile
          - bool can_play
          - card* get_top_card
          - int check_special_card
          - void setup_game
          - bool try_play
          - bool change_color
          - void write_into_json
          - discard_pile* from_json
        - draw_pile
          - std::vector<card*> _cards
          - serializable_value<bool>* _questionmarkbool
          - draw_pile
          - ~draw_pile
          - void shuffle
          - void set_questionmarkbool
          - bool is_empty
          - int get_nof_cards
          - card* get_top_card
          - void setup_game
          - bool draw
          - boid distribute_card
          - card* remove_top
          - void write_into_json
          - draw_pile* from_json
      - player

  - hand
    - std::vector<card*> _cards
    - hand
    - card* remove_card
    - int get_nof_cards
    - int get_score
    - std::vector<card*> get_cards
    - bool try_get_card
    - void setup_round
    - bool add_card
    - bool remove_card
    - void dump_cards_except_player
    - std::vector<card*>::iterator get_card_iterator
    - void write_into_json
    - hand* from_json
  - player
    - serializable_value<std::string>* _player_name
    - serializable_value<bool>* _called_uno
    - serializable_value<int>* _score
    - hand* _hand
    - std::string _game_id
    - player
    - ~player
    - bool has_called_uno
    - int get_nof_cards
    - hand* get_hand
    - std::string get_player_name
    - int get_score
    - int get_hand_score
    - void set_score
    - bool set_called_uno
    - bool add_card
    - bool remove_card
    - void setup_round
    - void dump_cards_except_played
    - void write_into_json
    - player* from_json
- game_state
  - int _max_nof_players
  - int _min_nof_players
  - std::vector<player*> _players
  - draw_pile* _draw_pile
  - discard_pile* _discard_pile
  - serializable_value<bool>* _is_started
  - serializable_value<bool>* _is_finished
  - serializable_value<int>* _round_number
  - serializable_value<int>* _current_player_idx
  - serializable_value<int>* _previous_player_idx
  - serializable_value<int>* _play_direction

- serializable_value<int>* _starting_player_idx
- serializable_value<bool>* _unanswered_wildcard
- serializable_value<bool>* _missed_uno_call
- serializable_value<bool>* _play_with_questionmark
- serializable_value<int>* _players_in_new_round
- game_state
- ~game_state
- int get_player_index
- bool ist_full
- bool is_started
- bool is_finished
- bool is_player_in_game
- bool is_allowed_to_play_now
- std::vector<player*>& get_players
- int get_round_number
- bool get_unanswered_wildcard
- bool get_missed_uno_call
- bool get_play_with_questionmark
- int get_players_in_new_round
- draw_pile* get_draw_pile
- discard_pile* get_discard_pile
- player* get_current_player
- player* get_previous_player
- void setup_round
- bool remove_player
- bool add_player
- bool start_game
- bool new_round
- bool draw_card
- bool play_card
- void skip_player
- void reverse_direction
- bool set_color
- void execute_questionmark
- void update_current_player
- void wrap_up_round
- void write_into_json
- game_state* from_json

- network
  - requests
    - call_uno_request
      - call_uno_request
      - void write_into_json
      - call_uno_request* from_json
    - client_request
      - RequestType _type
      - std::string _req_id
      - std::string _player_id
      - std::string _game_id

  - std::unordered_map<std::string, RequestType> _string_to_request_type
  - std::unordered_map<RequestType, std::string> _request_type_to_string
  - struct base_class_properties
  - client_request
  - ~client_request
  - base_class_properties create_base_class_properties
  - base_class_properties extract_base_class_properties
  - RequestType get_type
  - std::string get_req_id
  - std::string get_game_id
  - std::string get_player_id
  - std::string to_string
  - void write_into_json
  - client_request* from_json
- draw_card_request
  - int _nof_cards
  - draw_card_request
  - int get_nof_cards
  - void write_into_json
  - draw_card_request* from_json
- join_game_request
  - std::string _player_name
  - bool _first_game_of_player
  - std::string undefined_game_id
  - join_game_request
  - std::string get_player_name
  - bool get_first_game_of_player
  - void write_into_json
  - join_game_request* from_json
- new_round_request
  - new_round_request
  - void write_into_json
  - new_round_request* from_json
- pick_a_color_request
  - int _color
  - pick_a_color_request
  - int get_color
  - void write_into_json
  - pick_a_color_request* from_json
- play_card_request
  - std::string _card_id
  - play_card_request
  - std::string get_card_id
  - void write_into_json
  - play_card_request* from_json
- player_left_request
  - player_left_request

        - void write_into_json
        - new_round_request* from_json
        -
      - start_game_request
        - bool _questionmarkbool
        - start_game_request
        - bool get_questionmark
        - void write_into_json
        - start_game_request* from_json
    - responses
      - full_state_response
        - rapidjson::Value* _state_json
        - full_state_response
        - ~full_state_response
        - Rapidjson::Value* get_state_json
        - void Process
        - void write_into_json
        - full_state_response* from_json
      - request_response
        - bool _success
        - std::string _err
        - std::string _req_id
        - rapidjson::Value* _state_json
        - request_response
        - ~request_response
        - std::string get_error
        - std::string get_req_id
        - bool get_success
        - void Process
        - void write_into_json
        - request_response from_json
      - server_response
        - std::unordered_map<std::string, ResponseType> _string_to_response_type
        - std::unordered_map<ResponseType, std::string> _response_type_to_string
        - std::string _game_id
        - ResponseType _type
        - struct base_class_properties
        - server_response
        - base_class_properties create_base_class_properties
        - base_class_properties extract_base_class_properties
        - ResponseType get_type
        - std::string get_game_id
        - void Process
        - void write_into_json
        - server_response* from_json
  - serialization (nothing changed)
- server

- game_instance_manager
  - std::shared_mutex games_lut_lock
  - std::unordered_map<std::string, game_instance*> games_lut
  - game_instance* create_new_game
  - game_instance* find_joinable_game_instance
  - bool try_get_game_instance
  - bool try_get_player_and_game_instance
  - bool try_add_player_to_any_game
  - bool try_add_player
  - bool try_remove_player
- game_instance
  - game_state* _game_state
  - std::mutex modification_lock
  - game_instance
  - ~game_instance
  - std::string get_id
  - game_state* get_game_state
  - bool is_full
  - bool is_started
  - bool is_finished
  - bool start_game
  - bool new_round
  - bool try_add_player
  - bool try_remove_player
  - bool play_card
  - bool draw_card
  - bool set_color_wildcard
  - bool set_uno
- main
- player_manager
  - std::shared_mutex _rw_lock
  - std::unordered_map<std::string, player*> _players_lut
  - bool try_get_player
  - bool add_or_get_player
  - bool remove_player
- request_handler
  - request_response* handle_request
- server_network_manager
  - server_network_manager* _instance
  - std::shared_mutex _rw_lock
  - sockpp::tcp_acceptor _acc
  - std::unordered_map<std::string, std::string> _player_id_to_adress
  - std::unordered_map<std::string, sockpp::tcp_socket> _address_to_socket
  - server_network_manager
  - ~server_network_manager
  - void connect
  - void listener_loop

- void read_message
- void handle_incoming_message
- ssize_t send_message
- void broadcast_message
- void on_player_left

- client
  - app
    - Uno
      - bool OnInit
  - network
    - ClientNetworkManager
      - sockpp::tcp_connector* _connection
      - bool _connectionSuccess
      - bool _failedToConnect
      - void Init
      - void sendRequest
      - void parseResponse
      - bool connect
    - ResponseListenerThread
      - sockpp::tcp_connector* _connection
      - ResponseListenerThread
      - ~ResponseListenerThread
      - ExitCode Entry
      - void outputError
  - panels
    - ConnectionPanel
      - InputField* _serverAdressField
      - InputField* _serverPortField
      - InputField* _playerNameField
      - ConnectionPanel
      - wxString getServerAdress
      - wxString getServerPort
      - wxString getPlayerName
    - MainGamePanel
      - wxCheckBox* _playWithQuestionMarkCheckBox
      - MainGamePanel
      - void buildGameState
      - void buildWaitForNewRound
      - void buildOtherPlayerHand
      - void buildOtherPlayerLabels
      - void buildCardPiles
      - void buildTurnIndicator
      - void buildThisPlayer
      - void buildUnoButton
      - void buildScoreBoard
      - wxStaticText* buildStaticText
      - wxSize getBoundsOfRotatedSquare
      - double getEdgeLengthOfRotatedSquare

    - wxPoint getPointOnEllipse
- uiElements
  - ImagePanel
    - wxImage _image
    - wxBitmap _bitmap
    - double rotation
    - int _width
    - int _height
    - ImagePanel
    - void paintEvent
    - void onSize
  - InputField
    - wxStaticText* _label
    - wxTextCtrl* _field
    - InputField
    - wxString getValue
  - Scoreboard
    - wxListCtrl* _listCtrl
    - std::vector<wxString> _playerNames
    - ScoreBoard
    - void InitializeListCtrl
    - void AddPlayer
    - void UpdateScore
    - void SelectItem
  - UnoColorDialog
    - wxString _selectedColor
    - int _selectedColorIndex
    - UnoColorDialog
    - wxString GetSelectedColor
    - int GetSelectedColorIndex
    - void OnClose
    - void OnColorSelection
    - int FindColorIndex
    - wxString GetColorNameByIndex
- windows
  - GameWindow
    - wxBoxSizer* _mainLayout
    - wxStatusBar* _statusBar
    - wxPanel* _currentPanel
    - bool _darkModeEnabled
    - void ShowPanel
    - void setStatus
    - void toggleDarkMode
    - void updateUIForDarkMode
- GameController
  - GameWindwo* _gameWindow
  - ConnectionPanel* _connectionPanel
  - MainGamePanel* _mainGamePanel
  - player* _me

- game_state* _currentGameState
- void init
- void connectToServer
- voidupdateGameState
- void WaitForNewRound
- void startGame
- void newRound
- void drawCard
- void callUno
- void playCard
- void pickColor
- wxEvtJandler getMainThreadEventHandlet
- void showError
- void showStatus
- void showPickColor
- void showNewRoundMessage
- void showGameOverMessage

- main

- unit_tests
  - card
    - PlayCardsOn1
    - SerializationEquality
    - SerializationException
  - discard_pile
    - CanPlayOnEmptyPile
    - SetupGame
    - CanPlayOnSameSympolOrColor
    - GetTopCardOfEmptyPile
    - GetTopCardOfNonEmptyPile
    - CheckSpecialCard
    - ChangeColorValid
    - ChangeColorInvalidWildcard
  - draw_pile
    - IsEmptyTest
    - GetNofCardsTest
    - SetupGame
    - DrawEmpty
    - Draw
    - DistributeCardEmpty
    - DistributeCard0
    - DistributeCard2
    - RemoveTop
    - RemoveTop_onecard
  - game_state
    - wrap_up_round1
    - wrap_up_round2
    - questionmarkcard_test
    - skip_player

  - reverse_direction
  - set_color
  - remove_player
  - add_player
  - add_player_already_started
  - draw_card
  - start_game
  - setup_round
  - update_current_player
  - get_previous_player
  - get_current_player
  - get_player_index
  - is_player_in_game
  - is_allowed_to_play_now
  - get_players
  - is_full
  - is_started
- hand
  - AddOneCard
  - AddNoCards
  - AddManyCards
  - AddManyCardsWithDuplicates
  - RemoveOneCard
  - RemoveNonexistentCards
  - RemoveAllCards
  - RemoveManyCards
  - RemoveManyDuplicateCards
  - ScoreOneCard
  - ScoreSkipcard
  - ScoreWildcard
  - ScoreNoCards
  - ScoreManyCards
  - ScoreManyCardsWithDuplicates
  - CountNoCards
  - CoundManyCardsWithDuplicates
- Player
  - ConstructorTest
  - ScoreTest
  - NumberOfCardsTest
  - HandScoreTest
  - Call_reset_UNO
  - AddCardTest
  - AddMultipleCardsTest
  - RemoveCardTest
  - SetupRound
  - GetHandTest
- request_handler
  - JoinGameTest
  - CallUnoTest

- StartGameTest
- NewRoundTest
- PlayCardTest
- DrawCardTest
- PickaColorTest
- PlayerLeftTest