Bataille.py

```
import random
import pygame
import pygame_gui

COULEUR = ['Spade', 'Heart', 'Clubs', 'Diamond']
VALEUR = { 2: 'Deux', 3: 'Trois', 4: 'Quatre', 5: 'Cinq', 6: 'Six',7: 'Sept', 8: 'Huit', 9: 'Neuf', 10: 'Dix',11: 'Valet', 12: 'Dame', 13: 'Roi', 14: 'As' }
Paths = ['2S.jpg', '3S.jpg', '4S.jpg', '5S.jpg', '6S.jpg', '7S.jpg', '8s.jpg', '9S.jpg', '10S.jpg', 'JS.jpg', 'QS.jpg', 'KS.jpg', 'AS.jpg', '2H.jpg', '3H.jpg', '4H.jpg', '5H.jpg', '6H.jpg', '7H.jpg', '8H.jpg', '9H.
full_deck = []
for i in COULEUR:
    for j in VALEUR.keys():
        full_deck.append([j,i])
deck = full_deck
c = 0
for i in deck:
    i.append(Paths[c])
    c = c + 1
def color_value(card):
    if card[1] == 'Spade':
        return 1
    elif card[1] == 'Heart':
        return 2
    elif card[1] == 'Clubs':
        return 3
    else:
        return 4
class card(object):
    def __init__(self, deck):
        i = random.choice(range(len(deck)))
        s = deck[i]
        self.number = s[0]
        self.color = s[1]
        self.path = s[2]
        deck.pop(i)
def distributing(full_deck):
    deck = full_deck
    deck1 = []
    deck2 = []
    counter = 0
    while deck != []:
        c = card(deck)
        if counter % 2 == 0:
            deck1.append([c.number,c.color,c.path])
        else:
            deck2.append([c.number,c.color,c.path])
        counter = counter + 1
    decks = [deck1 , deck2]
    deck = full_deck
    return decks
deck = full_deck
class player_human(object):
    def __init__(self, name, my_deck):
        self.name = name
        self.deck = my_deck
    def shuffle(self, my_deck):
        new_deck = []
        while my_deck != []:
            c = card(my_deck)
            new_deck.append([c.number, c.color,c.path])
        self.deck = new_deck
        return self.deck
    def draw_card(self):
        c = deck1[0]
        print(str(self.name) + ' draws ' + str(c[0]) + ' of ' + str(c[1]))
        deck1.pop(0)
        return c
    def call_to_end(self):
        print(self.name + ' calls to end the game !!')
    def quit(self):
```

```
69          print(self.name + 'quits!!')
70          return 0
71  def computer_strategy_1(deck2):
72      i = 0
73      j = 0
74      c = 0
75      new_deck = []
76      temp_deck = deck2
77      max = 2
78      max_index = 0
79      max_card = deck2[0]
80      while j < 26:
81          for i in temp_deck:
82              if i[0] > max:
83                  max = i[0]
84                  max_index = c
85                  max_card = i
86              c = c + 1
87          j = j + 1
88          new_deck.append([max_card[0], max_card[1], max_card[2]])
89          temp_deck.pop(max_index)
90          max_index = 0
91      return new_deck
92
93  decks = distributing(full_deck)
94  deck1 = decks[0]
95  deck2 = decks[1]
96  #deck2 = computer_strategy_1(deck2)
97  pygame.init()
98  images = [pygame.image.load("B.jpg")]
99  player1 = player_human('Player1', deck1)
100 clock = pygame.time.Clock()
101 time_delta = clock.tick(60) / 1000.0
102 pygame.display.set_caption('Bataille')
103 window_surface = pygame.display.set_mode((800, 600))
104 background = pygame.Surface((800, 600))
105 background.fill(pygame.Color('#ffb72e'))
106 manager = pygame_gui.UIManager((800, 600))
107 x = 800 // 2 - 50
108 y = 600 // 2 - 25
109 Start_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y - 75), (100, 50)), text='START',manager=manager)
110 Credit_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y), (100, 50)), text='Credits',manager=manager)
111 Quit_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y + 75), (100, 50)), text='Quit',manager=manager)
112 leave_button = 'l'
113 shuffle_button = 's'
114 draw_button = 'd'
115 play_button = 'p'
116 option_3_button = '3'
117 option_5_button = '5'
118 option_10_button = '10'
119 option_full_game_button = '13'
120 end_turn_button = 'e'
121 menu_button = 'm'
122 limit = 13
123 c1 = deck1[0]
124 c2 = deck2[0]
125 draw_allowed = True
126 shuffle_allowed = True
127 end_turn_allowed = False
128 Menu = pygame.display.get_surface()
129 e0 = 5.2
130 E = 0
131 CARDS_WON = 0
132 CARDS_LOST = 0
133 deck_img = pygame.image.load("JPEG\Green_back.jpg")
134 def game(manager,decks):
135     deck1 = decks[0]
136     arena = pygame.Rect((142.5, 200), (500, 200))
137     Player1_deck_surface = pygame.Rect((322.5, 420), (120, 168))
138     Player2_deck_surface = pygame.Rect((322.5, 10), (120, 168))
139     pygame.draw.rect(background, (75, 45, 5), arena)
```

```
    pygame.draw.rect(background, (255, 255, 0), Player1_deck_surface)
    pygame.draw.rect(background, (0, 255, 100), Player2_deck_surface)
    manager.update(time_delta)
    manager.update(time_delta)
    deck_img = pygame.image.load("JPEG\Green_back.jpg")
    e = 0
    for i in deck1:
        background.blit(deck_img, Player1_deck_surface)
        background.blit(deck_img, Player2_deck_surface)
        Player1_deck_surface = pygame.Rect((322.5 + e, 420 + e), (120, 168))
        Player2_deck_surface = pygame.Rect((322.5 + e, 10 + e), (120, 168))
        e = e + 0.2
    manager.update(time_delta)
    player1_score(0)
    computer_score(0)
    first_to(manager)
    Game = pygame.display.get_surface()
    return Game
def computer_draw(deck2):
    c = deck2[0]
    print('Computer' + ' draws ' + str(c[0]) + ' of ' + str(c[1]))
    adresse = "JPEG/" + c[2]
    deck2.pop(0)
    cpath = pygame.image.load(adresse)
    drawn_card_surface = pygame.Rect((420, 210), (120, 168))
    deck_surface = pygame.Rect((322.5, 10), (126, 175))
    pygame.draw.rect(background, pygame.Color('#ffb72e'), deck_surface)
    e = 0
    Player1_deck_surface = pygame.Rect((322.5, 10), (120, 168))
    deck_img = pygame.image.load("JPEG\Green_back.jpg")
    for i in deck1:
        background.blit(deck_img, Player1_deck_surface)
        Player1_deck_surface = pygame.Rect((322.5 + e, 10 + e), (120, 168))
        e = e + 0.2
    pygame.time.delay(100)
    background.blit(cpath, drawn_card_surface)
    return c
def first_to(manager):
    f = pygame.font.Font('freesansbold.ttf', 25)
    txt1 = f.render(str('First to :'), 1, (225, 225, 225), '#4b2d05')
    txtrect1 = pygame.rect.Rect((350, 225), (100, 25))
    pygame.draw.rect(background, (75, 45, 5), txtrect1)
    background.blit(txt1, txtrect1)
def end(manager):
    background = pygame.Surface((800, 600))
    background.fill(pygame.Color('#ffb72e'))
    manager.update(time_delta)
    manager.clear_and_reset()
def see_first_card(deck1):
    c1 = deck1[0]
    adresse = "JPEG/" + c1[2]
    cpath = pygame.image.load(adresse)
    first_card_surface = pygame.Rect((322.5, 420), (126, 175))
    background.blit(cpath, first_card_surface)
def battle(card1,card2):
    if card1[0] > card2[0]:
        return 1
    elif color_value(card1) > color_value(card2) and card1[0] == card2[0]:
        return 1
    else:
        return 0
def won(e,deck_img):
    card_won_surface = pygame.Rect((12.5 + e, 210 + e), (120, 168))
    background.blit(deck_img, card_won_surface)
def lost(e,deck_img):
    card_won_surface = pygame.Rect((660 + e, 210 + e), (120, 168))
    background.blit(deck_img, card_won_surface)
def player1_score(score):
    f = pygame.font.Font('freesansbold.ttf', 25)
    txt = f.render(str(score), 1, (0, 50, 0), '#ffb72e')
    txtrect = pygame.rect.Rect((50, 399), (20, 20))
```

```
    pygame.draw.rect(background, pygame.Color('#ffb72e'),txtrect)
    background.blit(txt, txtrect)
def computer_score(score):
    f = pygame.font.Font('freesansbold.ttf', 25)
    txt = f.render(str(score), 1, (0, 50, 0), '#ffb72e')
    txtrect = pygame.rect.Rect((725, 399), (20, 20))
    pygame.draw.rect(background, pygame.Color('#ffb72e'), txtrect)
    background.blit(txt, txtrect)
def clear_arena():
    arena = pygame.Rect((142.5, 200), (500, 200))
    pygame.draw.rect(background, (75, 45, 5), arena)
    pygame.time.delay(100)
def winner(manager):
    print("Player1 is the WINNER!!!")
    f = pygame.font.Font('freesansbold.ttf', 76)
    txt1 = f.render(str('CONGRATULATIONS'), 1, (250, 0, 25), '#ffffff')
    txtrect1 = pygame.rect.Rect((0, 245), (800, 80))
    pygame.draw.rect(background, (255,255,255), txtrect1)
    background.blit(txt1, txtrect1)
    g = pygame.font.Font('freesansbold.ttf', 50)
    txt2 = g.render(str('                  YOU WON'), 1, (250, 0, 25), '#ffffff')
    txtrect2 = pygame.rect.Rect((0, 400), (800, 50))
    pygame.draw.rect(background, (255, 255, 255), txtrect2)
    background.blit(txt2, txtrect2)
    pygame.display.update()
    pygame.time.wait(500)
    end(manager)
def loser(manager):
    print("Computer is the WINNER!!!")
    f = pygame.font.Font('freesansbold.ttf', 100)
    txt = f.render(str('         LOST'), 1, (255, 255, 255), '#000000')
    txtrect = pygame.rect.Rect((0, 300), (800, 80))
    pygame.draw.rect(background, (0,0,0), txtrect)
    background.blit(txt, txtrect)
    pygame.display.update()
    end(manager)
def tie(manager):
    print("Tie !!!")
    f = pygame.font.Font('freesansbold.ttf', 100)
    txt = f.render(str('        TIE'), 1, (255, 255, 255), '#4b2d05')
    txtrect = pygame.rect.Rect((0, 300), (800, 80))
    pygame.draw.rect(background, (75,45,5), txtrect)
    background.blit(txt, txtrect)
    pygame.display.update()
    end(manager)
is_running = True

while is_running:
    time_delta = clock.tick(60) / 1000.0
    pygame.time.delay(100)
    for event in pygame.event.get():

        if event.type == pygame.QUIT:
             is_running = False

        if event.type == pygame.USEREVENT:
            if event.user_type == 'ui_button_pressed':
                if event.ui_element == Start_button:
                    manager.clear_and_reset()
                    leave_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((700, 550), (85, 35)),text='Leave', manager=manager)
                    play_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((700, 275), (90, 40)),text='Play', manager=manager)
                    option_3_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((225, 300), (50, 40)),text='3', manager=manager)
                    option_5_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((275, 300), (50, 40)),text='5', manager=manager)
                    option_10_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((325, 300), (50, 40)),text='10', manager=manager)
                    option_full_game_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((375, 300), (100, 40)),text='Full Game', manager=manager)
                    manager.update(time_delta)
                    decks = distributing(full_deck)
                    game(manager,decks)
                if event.ui_element == play_button:
                    draw_allowed = True
                    shuffle_allowed = True
```

```
                    manager.clear_and_reset()
                    arena = pygame.Rect((142.5, 200), (500, 200))
                    pygame.draw.rect(background, (75, 45, 5), arena)
                    manager.update(time_delta)
                    leave_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((700, 550), (85, 35)),text='Leave', manager=manager)
                    draw_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((475, 475), (90, 40)),text='Draw', manager=manager)
                    shuffle_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((205, 475), (90, 40)),text='Shuffle', manager=manager)
                    end_turn_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((600, 475), (90, 40)),text='End Turn', manager=manager)
                    manager.update(time_delta)
                    c2 = computer_draw(deck2)
                    see_first_card(deck1)
                if event.ui_element == Credit_button:
                    manager.clear_and_reset()
                    leave_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((700, 550), (85, 35)),text='Leave', manager=manager)
                    f = pygame.font.Font('freesansbold.ttf', 25)
                    txt = f.render('Created by ELMOUHAJIR Yassir',1,(150, 0, 0),'#ffb72e')
                    txtrect = pygame.rect.Rect((225, 295), (85, 35))
                    background.blit(txt, txtrect)
                    g = pygame.font.Font('freesansbold.ttf', 20)
                    txt0 = g.render('Supported by GIID, ENSAK', 1, (15, 225, 15), '#ffb72e')
                    txtrect0 = pygame.rect.Rect((50, 500), (85, 35))
                    background.blit(txt0, txtrect0)
                if event.ui_element == option_3_button:
                    limit = 3
                    print('First to 3 wins.')
                if event.ui_element == option_5_button:
                    limit = 5
                    print('First to 5 wins.')
                if event.ui_element == option_10_button:
                    limit = 10
                    print('First to 10 wins.')
                if event.ui_element == option_full_game_button:
                    limit = 26
                    print("Player with the most cards at the end wins.")
                if event.ui_element == leave_button or event.ui_element == menu_button:
                    manager.clear_and_reset()
                    pygame.display.update()
                    CARDS_WON = 0
                    CARDS_LOST = 0
                    background = pygame.Surface((800, 600))
                    background.fill(pygame.Color('#ffb72e'))
                    manager.update(time_delta)
                    manager.clear_and_reset()
                    Start_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y - 75), (100, 50)),text='START', manager=manager)
                    Credit_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y), (100, 50)),text='Credits', manager=manager)
                    Quit_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((x, y + 75), (100, 50)),text='Quit', manager=manager)
                    decks = distributing(full_deck)
                if event.ui_element == shuffle_button:
                    if shuffle_allowed == True:
                        deck1 = player1.shuffle(deck1)
                        deck_surface = pygame.Rect((322.5, 420), (126, 175))
                        pygame.draw.rect(background, pygame.Color('#ffb72e'), deck_surface)
                        e = 0
                        Player1_deck_surface = pygame.Rect((322.5, 420), (120, 168))
                        deck_img = pygame.image.load("JPEG\Green_back.jpg")
                        for i in deck1:
                            background.blit(deck_img, Player1_deck_surface)
                            pygame.time.delay(10)
                            Player1_deck_surface = pygame.Rect((322.5 - e, 420 - e), (120, 168))
                            e = e + 0.2
                        see_first_card(deck1)
                        shuffle_allowed = False
                        manager.update(time_delta)
                if event.ui_element == draw_button:
                    if draw_allowed == True:
                        c1 = player1.draw_card()
                        adresse = "JPEG/" + c1[2]
                        cpath = pygame.image.load(adresse)
                        drawn_card_surface = pygame.Rect((220, 210), (120, 168))
                        deck_surface = pygame.Rect((322.5, 420), (126, 175))
                        pygame.draw.rect(background,pygame.Color('#ffb72e'),deck_surface)
```

```
                        e = 0
                        Player1_deck_surface = pygame.Rect((322.5, 420), (120, 168))
                        for i in deck1:
                            background.blit(deck_img, Player1_deck_surface)
                            Player1_deck_surface = pygame.Rect((322.5 + e, 420 + e), (120, 168))
                            e = e + 0.2
                        pygame.time.delay(50)
                        background.blit(cpath, drawn_card_surface)
                        manager.update(time_delta)
                        draw_allowed = False
                        shuffle_allowed  =False
                        end_turn_allowed = True
                    manager.update(time_delta)
                if event.ui_element == end_turn_button:
                    if end_turn_allowed == True:
                        pygame.time.wait(100)
                        if battle(c1, c2) == 1:
                            won(E, deck_img)
                            E = E + 0.2
                            won(E, deck_img)
                            E = E + 0.2
                            CARDS_WON = CARDS_WON + 1
                            player1_score(CARDS_WON)
                            print("Player Wins !!")
                        else:
                            lost(E, deck_img)
                            E = E + 0.2
                            lost(E, deck_img)
                            E = E + 0.2
                            CARDS_LOST = CARDS_LOST + 1
                            computer_score(CARDS_LOST)
                            print("Computer Wins !!")
                        see_first_card(deck1)
                        draw_allowed = True
                        shuffle_allowed = True
                        clear_arena()
                        c2 = computer_draw(deck2)
                        end_turn_allowed = False
                        manager.update(time_delta)
                if event.ui_element == Quit_button:
                    exit()
                    pygame.display.quit()
                if CARDS_WON >= limit:
                    winner(manager)
                    menu_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((675, 535), (90, 45)),text='Main Menu', manager=manager)
                    CARDS_WON = 0
                    CARDS_LOST = 0
                if CARDS_LOST >= limit:
                    loser(manager)
                    menu_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((675, 535), (90, 45)),text='Main Menu', manager=manager)
                    CARDS_LOST = 0
                    CARDS_WON = 0
                if deck1 == [] and CARDS_WON > CARDS_LOST:
                    winner(manager)
                    menu_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((675, 535), (90, 45)),text='Main Menu', manager=manager)
                    CARDS_WON = 0
                    CARDS_LOST = 0
                if (deck2 == [] or deck1 == []) and CARDS_WON < CARDS_LOST:
                    loser(manager)
                    menu_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((675, 535), (90, 45)),text='Main Menu', manager=manager)
                    CARDS_LOST = 0
                    CARDS_WON = 0
                    player1 = player_human('Player1', deck1)
                if (deck2 == [] or deck1 == []) and CARDS_WON < CARDS_LOST:
                    tie(manager)
                    menu_button = pygame_gui.elements.UIButton(relative_rect=pygame.Rect((675, 535), (90, 45)),text='Main Menu', manager=manager)
                    CARDS_LOST = 0
                    CARDS_WON = 0
                    player1 = player_human('Player1',deck1)

        manager.process_events(event)
```

```
424     manager.update(time_delta)
425     window_surface.blit(background, (0, 0))
426     manager.draw_ui(window_surface)
427     pygame.display.update()
428 pygame.quit()
429 
430 
431 
```