José Luis Madrigal A01745419

César Emiliano Palome A01746493

Jorge Isidro Blanco A01745907

Alejandra Cancio A01745339

Final Project Advanced Programming

First of all, this project is a custom game on PyGame, which is based on a tutorial by Julian Meyer. The theme of our game is soccer, and it is based mainly on some strikers attacking the goals and you must stop them, hitting them with a ball. But what is really interesting about the gameplay is that you will have an initial value of points that is zero, and every time a villain touches one of your three goals, a point will be subtracted, in addition to the fact that when you hit someone with a ball, they will add a point. Therefore, there will be a maximum value of goals (5) that will make you win and a minimum value (-5) that will make you lose. Also, you will have a time in the game, which is three and a half minutes, and when it is over, then you will also lose. In addition to the gameplay, you have the characters, the background and the sounds, which give the game a unique touch. The goals, the player and the villains are icons taken from the Flaticon page. As for the sounds, every time the player throws a ball, Cristiano Ronaldo's iconic "SIU" is heard, in addition to the word "Golazo" written by Perro Bermúdez being heard every time they reach your goal. Finally, the background song is "Wavin Flag", which is for many a clear representation of soccer.

It is worth mentioning that we chose this theme because we really like soccer, since we find it to be an entertaining and challenging sport. On the other hand, the characters and sounds were chosen because we felt it was important for the video game to be able to convey what can be found in a football match, so our main consumers would be sports fans, and to be more specific, those from Mexico, since in this country soccer is well known and popular. We also decided to use the gameplay of adding and subtracting points so that the game would not be tedious and would become really challenging, since we want people to develop skills of speed,

concentration and reflexes.

One of the problems we had while coding was finding the right images for the characters and objects, as they were required to have no background and also had to have a specific number of pixels. Also, making the villains take points and the balls add up was not that complicated, but what was a little difficult to understand was the way to show that counter on the screen, so that the player could see how close he is to win or lose. But in the end, we were able to fix it because we dedicated ourselves to understanding how time was put on the screen, which was shown in the tutorial.

Finally, we really liked this project as we learned new functions and commands in PyGame. Besides that, we were able to exploit our creativity a little more to create something that could entertain others. Also, it was quite interesting to know the way in which a video game is made, since we had played several, but we never thought that the code behind involves a lot of logic and work.

Code implementation and execution

juegofutbol.py

c: > Users > jlms3 > Documents > GitHub > Projects_Codes > High School > PyGame > juegofutbol.py > ...

```
                keys[1]=False
            elif event.key==pygame.K_s:
                keys[2]=False
            elif event.key==pygame.K_d:
                keys[3]=False
        if event.type==pygame.MOUSEBUTTONDOWN:
            shoot.play()
            position=pygame.mouse.get_pos()
            acc[1]+=1
            balls.append([math.atan2(position[1]-(playerpos1[1]+32),position[0]-(playerpos1[0]+26)),playerpos1[0]+32,playerpos1[1]+32])
    #9 Move player
    if keys[0]:
        playerpos[1]-=5
    elif keys[2]:
        playerpos[1]+=5
    if keys[1]:
        playerpos[0]-=5
    elif keys[3]:
        playerpos[0]+=5
    #10 - Win/Lose check
    if goals>=5:
        running=0
        exitcode=1
    if goals<=-5 or pygame.time.get_ticks()>=210000:
        running=0
        exitcode=0
    if acc[1]!=0:
        accuracy=acc[0]*1.0/acc[1]*100
    else:
        accuracy=0
# 11 - Win/lose display
if exitcode==0:
    pygame.font.init()
    font = pygame.font.Font(None, 24)
    text = font.render("Accuracy: "+str(accuracy)+"%", True, (255,0,0))
    textRect = text.get_rect()
    textRect.centerx = screen.get_rect().centerx
    textRect.centery = screen.get_rect().centery+24
```