

Voor mijn Pygame-project heb ik samengewerkt met twee partners: een online peer op Discord genaamd RedRiver en mijn broer, die een masterdiploma heeft in Industriële Wetenschappen ELO ICT Embedded Systems. Beide partners hebben waardevolle feedback gegeven en hebben geholpen mijn project te verbeteren. RedRiver gaf gedetailleerde feedback via Discord, terwijl mijn broer praktisch advies gaf over redundantie en structuur.

Uit de feedback van mijn partners bleek duidelijk dat mijn code te veel overbodige elementen bevatte. Dit kwam doordat ik vaak nieuwe dingen probeerde die niet altijd werkten, maar delen van die pogingen liet ik in de code staan. Hierdoor werd mijn code rommelig en ongeorganiseerd. Deze problemen vielen me in eerste instantie niet op, maar dankzij de feedback kon ik ze identificeren en aanpakken.

De communicatie vond plaats via Discord en face-to-face meetings. De Discord-server zat vol met professionele programmeurs, en mijn eerste gesprekken waren met een andere programmeur die ook nuttige tips gaf. Het ontvangen van feedback was een geweldige ervaring omdat het me hielp begrijpen welke delen van mijn code verbeterd konden worden. De suggesties van RedRiver leerden me hoe ik mijn code beter kon organiseren door onnodige variabelen te verwijderen en redundantie te verminderen. Bovendien raadde mijn broer aan om opmerkingen aan mijn code toe te voegen, zodat deze begrijpelijker zou zijn voor anderen die het zouden lezen.

Mijn rol was het programmeren en testen van de code en het implementeren van de feedback. RedRiver hielp met het vereenvoudigen van variabelen en het verbeteren van de efficiëntie, terwijl mijn broer zich richtte op het verbeteren van de structuur en het aanmoedigen van goede documentatiepraktijken. Ondanks verschillen in tijdsinzet was de samenwerking zeer waardevol en effectief.

Ik heb veel geleerd over het belang van duidelijke communicatie, hoe ik documentatie kan gebruiken om mijn code begrijpelijker te maken, en hoe ik de balans kan vinden tussen leesbaarheid en efficiëntie. Deze lessen zal ik toepassen in toekomstige projecten om professioneler te werken en mijn vaardigheden verder te ontwikkelen.

Screenshots van de Discord-gesprekken die de samenwerking en feedback tonen die hebben geleid tot verbeteringen in mijn project.

**RV** vandaag om 0t:11

Hi everyone,

I hope you're doing well! I'm currently working on a Pygame project and would really appreciate some feedback on my code. If you have some time, could you take a look and share your thoughts?

Additionally, if you're open to it, I'd love to hop on a quick voice chat to discuss the code and hear your suggestions. It would be a huge help for me to improve my skills and understanding.

Let me know if you're available! It would mean a lot.

Project: <https://github.com/UCLL-introproject-project/pygame-development-project-ao-2425-v2-RuneVerheijen> - BlackjackProject

Best regards,

RV (bewerkt)

**RedRiver** vandaag om 0t:21

First thing I would look into doing would be documenting a lot of this program.

`results = ['', '', '', '', 'TIE GAME...']` things like this are very confusing as to what they do since you can have more than 5 game

Well you'll have to do that part ofc. Ill just provide help and how you can improve this

first thing I would do is to reduce the complexity and how many variables you have

a lot of these can be removed or simplified (bewerkt)

**RV** vandaag om 0t:26

Could you give an example, please ?

**RedRiver** vandaag om 0t:26

since you arent keeping track of the suite of cards, you can keep the hidden card already in the the dealers deck

yes 1 moment

```
active = False
records = [0, 0, 0]
player_score = 0
dealer_score = 0
initial_deal = False
my_hand = []
dealer_hand = []
dealer_nand = []
outcome = 0
reveal_dealer = False
hand_active = False
outcome = 0
add_score = False
results = ['', '', '', '', 'TIE GAME...']
sound_played = False
```

the `active` variable is never changed in the entire program. There really isnt a point in having it as its always True

same with the `records` variable. You are already keeping track of the player and dealer score, so pick one of these variables to keep the actual score

maybe change the `initial_deal` variable to something like `player_turn` so that you can just set this to False and then you'll know the dealer can flip the cards over etc

you define `outcome` 2 times

and `add_score` has no purpose

**RV** vandaag om 0t:30

Thankyou [@RedRiver](#) I really appreciate your feedback I will change those things and make it better. Is it okay if I share this conversation with my tutor?

**RedRiver** vandaag om 0t:31

well theres more

**RV** vandaag om 0t:32

Please go ahead. (bewerkt)

**RedRiver** vandaag om 0t:35

`game_deck` has no purpose in the program. You define it but you never do anything with it.

a lot of redundant code on the button checking.

You often have these large resetting blocks of code that can be put inside of a function

the `reveal_dealer` variable is good, so im not sure why you have the `initial_deal` variable. You can just store all of the cards in their hands and skip the first card if its hidden

you also use deepcopy with the `game_deck` for some reason but dont use it

and this is just a nit pick

but instead of randomly picking a card and then appending / popping, you can shuffle the deck first with random, then pop from the end

```
card = random.randint(0, len(current_deck))
current_hand.append(current_deck[card - 1])
current_deck.pop(card - 1)
```

this works fine though

I would say cleaning up then documentation would be best tho

**RV** vandaag om 0t:40

Thank you, now that you told me I see it as well it was because I was trying different thing and left the code resting without a purpose. Thankyou [@RedRiver](#) really means alot.