

Galaktisk Galskab i PyGame Zero – Workshop 4

I workshop 3 fik vi skabt en bølge af angribende ufoer. Når de blev ramt af rumskibets skud, blev de erstattet af animerede eksplosioner. Vi fik også tilføjet en score for hvor mange ufoer, der er ramt. I denne sidste workshop skal vi tilføje, at rumskibet mister skjold, når det bliver ramt af ufoer og "game over", hvis det ikke har flere liv tilbage.

Tilføj status-egenskaber til rumskibet

Noget af det smarte ved PyGame Zeros Actor-objekter er, at man kan tilføje egenskaber, man selv bestemmer, ved hjælp af dot-notationen (punktum). Vi skal tilføje egenskaberne `liv` og en *boolean* `ramt` samme sted som vi tilføjede `score` sidste gang. En *boolean* er en sandhedsvariabel, som har værdien `True` eller `False`. Rumskibet starter med 3 liv, med score 0 og er ikke ramt.

```
rumskib.score = 0
rumskib.liv = 3
rumskib.ramt = False
```

Når rumskibet bliver ramt

Når rumskibet bliver ramt mister det liv. Hvis der ikke er flere liv tilbage, erstatter vi rumskibet med en eksplosion. Ellers signalerer vi, at rumskibet er ramt i vores *boolean* og sætter uret til at aflyse ramt-status efter 1 sekund.

```
def aflys_ramt_rumskib():
    rumskib.ramt = False

def rumskib_ramt():
    rumskib.liv -= 1
    if rumskib.liv == 0:
        skab_eksplosion(rumskib.pos)
    else:
        rumskib.ramt = True
        clock.schedule(aflys_ramt_rumskib, 1)
```

Når en ufo støder sammen rumskibet, kalder vi den nye funktion `rumskib_ramt`. Rumskibet skal være i live og ufoen må ikke være røget ud af ufo-listen, fx. fordi det selv er ramt. Indsæt nedenstående kode til sidst i for ufo-loopet i `ufoer_update` for at tjekke om den aktuelle ufo rammer rumskibet.

```
if ufo.colliderect(rumskib) and rumskib.liv > 0 and ufo in ufo_liste:
    ufo_liste.remove(ufo)
    rumskib_ramt()
```

Vis spilleren er ramt

Rumskibet mister et liv, når det er ramt. Det tegner vi på skærmen, indtil `ramt`-status bliver aflyst ved at indsætte dette i `draw-callback`.

```
if rumskib.ramt:
    screen.draw.text(
        f"RAMT! {rumskib.liv} skjold tilbage",
        midtop=(WIDTH // 2, HEIGHT // 2),
        fontsize=70,
    )
```

Game over

Hvis der ikke er flere liv tilbage, skal vi ikke tegne rumskibet. Det er blevet erstattet af en eksplosion. I stedet skal vi vise en "game over"-skærm og stoppe ufo-sværmen. Indsæt nedenstående i draw-callback.

```
if rumskib.liv > 0:
    rumskib.draw()
else:
    screen.draw.text("GAME OVER",
        midtop=(WIDTH // 2, HEIGHT // 2), fontsize=70)
    clock.unschedule(skab_ufo)
```

Hvis der ikke er flere liv, skal rumskibet heller ikke kunne skyde længere, så `on_mouse_down` skal ændres, så funktionen kun udføres, hvis rumskibet har liv tilbage. Den nye funktion har også et kald til `sounds`, så vi kan få noget lyd på. Du kan downloade lydfilerne fra <https://github.com/stee0640/pgzero2025/>. De ligger i folderne `sounds` og `music` og skal ligge i samme foldere i forhold til dit program.

```
def on_mouse_down():
    if rumskib.liv > 0:
        sounds.laser.play()
        skud = Actor("skud1.png", pos=rumskib.midtop)
        skud_liste.append(skud)
```

På samme måde kan du give eksplosionerne lyd ved at tilføje følgende i `skab_eksplosion`.

```
sounds.boom.play()
```

Og som det allersidste, hvis du vil have baggrundsmusik, kan du indsætte følgende kald i toppen af programmet. Så vil det køre i loop.

```
music.play("battleship")
```