I Am not Legend

Rasmus Riemke Rubæk

27/04/2023

Projekt lavet med:

Oliver Mads Kragh

Jeppe Møller Wortmann



# Resume

Vi har udviklet et spil der er baseret på “Call of Duty: Zombies”. Det er et spil hvor man som det sidste menneske, skal skyde uendelige bølger af zombier. Der er et user interface, der lader spilleren vælge sværhedsgraden af spillet. Spillet kan ikke vindes, der kan dog laves en highscore i form af mængde zombier dræbt, eller mængden af bølger overlevet.

# Projektbeskrivelse

Vi valgte netop at lave vores program fordi alle gruppemedlemmer interesserer sig for zombie-genren. Dette program inkorporerer også en stor del af de funktioner og lignende, vi har lært i de forskellige forløb.

Vi vil udarbejde et program, der minder om Call of Duty: Zombies, men med en anden målgruppe, og mindre avanceret. Målgruppen er mere fokuseret på personer der er nye til genren.

Vi vil redegøre for programmets udvikling, heri med fokus på selve koden og opsætningen.

Vi vil undersøge, hvorvidt oplevelsen med spillet kan forbedres ved hjælp af flere forskellige sværhedsgrader.

Vi vil diskutere hvorvidt produktet kan vække en interesse for zombie-genren.

Vi vil diskutere hvorvidt spillet kan udvides med flere funktioner.

Jeppe Møller Wortmann vil i sin synopsis besvare problemstillingen med fokus på Playercontrolleren, UI og våben.

Oliver Mads Kragh vil i sin synopsis besvare problemstillingen med fokus på Wavespawner og integrering af GitHub.

Rasmus Riemke Rubæk vil i sin synopsis besvare problemstillingen med fokus på troubleshooting, zombier, og 3D-objekterne.

# Programbeskrivelse

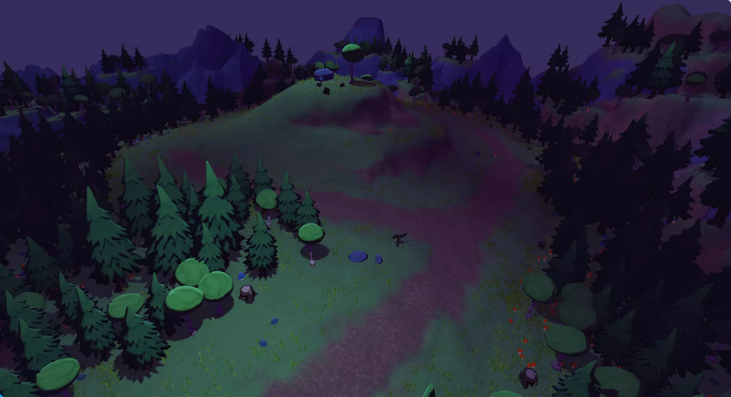
Vi har lavet et zombie spil, hvor man er den sidste person tilbage på en cirkulær bane, med en bakke i midten. Der er et user interface hvori man kan gå til indstillingerne og ændre i sværhedsgraden. Når spillet startes bestemmes ens livspoint og zombiernes skade af sværhedsgraden. UI’et viser hvor meget liv man har tilbage, hvor mange skud man har tilbage og hvor mange zombier man har dræbt. Efter spillet er startet, kan man til hver en tid pause spillet ved at trykke “escape”. Når man har Ak’en i hånden kan man holde venstreklik inde, for full-auto. Hvis man har pistolen i hånden, skal man klikke for hver gang man vil skyde et skud. Zombierne spawner i bølger, hvor størrelsen af bølgen vokser i løbet af spillet. Hvis zombierne rører spilleren mister man liv med tiden. Når man når 0 eller mindre livspoint, dør man. Så kommer der en score med hvor mange zombier man dræbte, samt muligheden for at starte et nyt spil.

# Udviklingsproces

Jeg har ikke lavet noget indtil videre, derfor er dette en total løgn, alt hvad jeg skriver her passer ikke og det er skrevet af jeppe og oliver Vi startede med at diskutere hvilket slags spil vi ville udvikle. Vi snakkede både om en version af “Subway Surfers”, og et bilspil. Vi var dog alle interesseret i at lave et FPS spil. Et spil hvor man som spiller mens man ser ud fra en persons synsvinkel. Man skal så skyde efter vilkårlige ting. Vi skulle bare bestemme os for hvad man skulle skyde efter. En skydebane hvor man skød efter stationære skydeskiver var for kedeligt og der var ikke rigtigt nogen følelse af præstation. Hvis man skulle skyde efter andre spillere, kræver det at kunne spille multiplayer, hvilket er meget svært og ressourcekrævende. Derfor endte vi med zombier. De flytter sig og giver spillet liv. Det er også meget nemt at lære, da konceptet med at dræbe zombier for at overleve ikke er nyt eller ukendt.

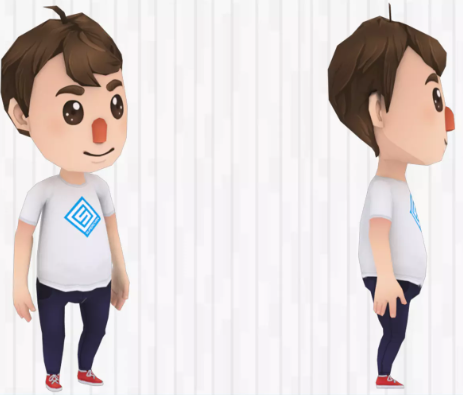
## 3D-Objekter

Nu hvor vi vidste hvad spillet skulle handle om, skulle vi for det første have en bane, og derefter nogle objekter. Vi kiggede rundt i Unity Asset Storen og endte med at vælge banen som kan ses herunder:



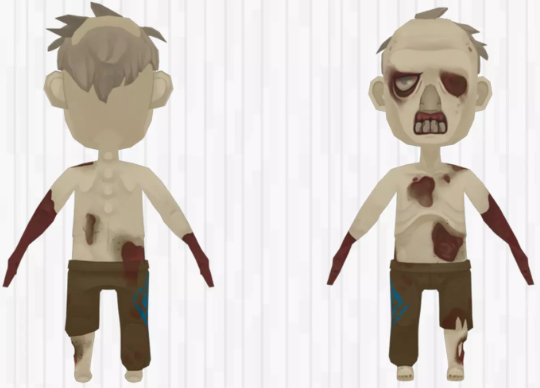
Banen er lavet af “Supercyan” og hedder “Environment Pack: Free Forest Sample”[[1]](#footnote-0). Det er et mørkt map som giver en lidt skræmmende stemning. Det passer nemlig bedst til zombie-spil. En anden vigtig egenskab denne bane har, er at der ikke er nogen vægge eller hjørner som zombier kan sidde fast i. Dette gør, at vi ikke skal lære zombierne at finde vej rundt om objekter, eller ud af hjørner.

Banen er lavet med et low-poly design. Det betyder at grafikken ikke er så realistisk, men mere tegnefilmsagtig. Denne stil passer også godt til vores målgruppe. Folk der ikke har prøvet et zombie-spil før, bliver måske for skræmte til at kunne spille, hvis grafikken er for realistisk. Med det i tankerne fandt vi modellen til vores spiller:



Den passer lige i stil med den bane, vi har valgt. Spillerens model er også lavet af “Supercyan” og hedder “Character Pack: Free Sample”[[2]](#footnote-1). Den kommer med nogle basale animationer.

På samme måde fandt vi modellen til zombierne:



modellen er igen lavet af “Supercyan” og hedder “Character Pack: Zombie Sample”[[3]](#footnote-2). Sammen med modellen, kommer et par animationer, så vi kan få zombierne til at gå, i stedet for at glide hen over jorden.

Våbnene er lavet af “Simon Mercuzot” og hedder “Low Poly Weapons VOL.1”[[4]](#footnote-3)



Vi bliver ved den low-poly stil, også ved våbnene.

Skuddene består bare af en simpel kugle model.

## Zombier

Zombie prefabbet fik tagget “Enemy”, så PlayerController scriptet kunne se om det er en zombie der rører spilleren.

Zombie scriptet skal kun køre når spillet er aktivt, derfor har vi variablen “isGameActive”

private bool isGameActive;

For at zombierne skal kunne kollidere med ting, og påvirkes af tyngdekraften, gives de en rigidbody. Der defineres navnet på zombiens rigidbody, og den findes i void Start:

private Rigidbody zombieRB;

void Start()

{

zombieRB = GetComponent<Rigidbody>();

}

For at kunne følge efter spilleren, skal spilleren findes. Spilleren gives navnet “player”, og findes i void Start:

private GameObject player;

void Start()

{

player = GameObject.Find("MaleFree1");

}

Vi tjekker først om spillet er aktivt. Hvis det er, så findes retningsvektoren fra zombien til spilleren ved at trække spillerens koordinater fra zombiens koordinater. Denne vektor bruges til den retning, zombierne skal bevæge sig i. Vi får zombierne til at bevæge sig med transform.Translate kommandoen. Hastigheden zombierne bevæger sig med, er defineret som “zombieSpeed”. For at zombierne kigger på spilleren, og ikke kigger på alt muligt andet, bruges kommandoen transform.LookAt.

private GameObject player;

private float zombieSpeed = 1;

private bool isGameActive;

void Update()

{

isGameActive = GameObject.Find("UItomt").GetComponent<UI>().isGameActive;

if (isGameActive == true)

{

Vector3 targetDirection = (player.transform.position - transform.position).normalized;

transform.LookAt(player.transform);

transform.Translate(Vector3.forward \* Time.deltaTime \* zombieSpeed );

}

}

Når zombien dør, skal objektet fjernes. Når det er gjort skal der tilføjes 1 til mængden af zombier man har dræbt, som vises i UI’et. Til sidst skal vi fortælle Wavespawneren, at der er en zombie mindre på mappet. Så kan den nemlig finde ud af, hvornår der ikke er flere zombier tilbage på banen.

void Die()

{

Destroy(gameObject);

GameObject.Find("UItomt").GetComponent<UI>().killedZombies+= 1;

WaveSpawner spawner = FindObjectOfType<WaveSpawner>();

spawner.spawnedEnemies.Remove(gameObject);

}

Vi finder ud af hvornår zombien bliver ramt af et skud, med OnTriggerEnter kommandoen. Hvis det objekt der rammer zombien har tagget “Bullet”, som skuddene har, trækkes 50 livspoint fra zombiens liv. Zombien har 50 liv til at starte med, dette kan dog ændres ved hjælp af sværhedsgraden. Hvis zombiens livspoint kommer under eller lig med 0, kalder vi Die voidet.

private double startingHealth = 50;

public double currentHealth;

private double difficulty;

void Start()

{

difficulty = GameObject.Find("UItomt").GetComponent<UI>().difficulty;

currentHealth = startingHealth \* difficulty;

}

void OnTriggerEnter(Collider other)

{

if (other.CompareTag("Bullet"))

{

currentHealth = currentHealth - 50;

if (currentHealth <= 0)

{

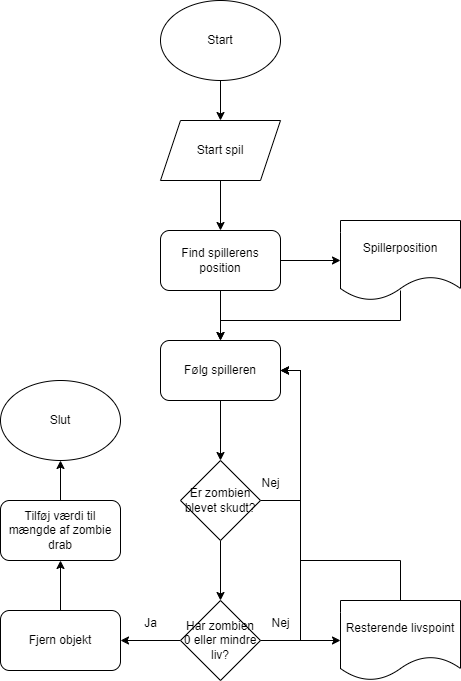
Die();

}

}

}

Herunder kan ses et forsimplet rutediagram over zombie scriptet:



## Troubleshooting

Vi havde løbende en del problemer. For det første er det muligt at fase igennem jorden, og falde uden for banen. Det sker ikke særligt tit, og kun når man løber hurtigt ind i en skråning. For at det ikke skal ødelægge oplevelsen med spillet, har vi sørget for at hvis spilleren kommer under en bestemt y-værdi, bliver man teleporteret op på træet på bakken. Dette kan ses som en slags “easter egg”. Da man ellers ikke kan komme op på træet.

void Update()

{

if (transform.position.y < -6)

{

transform.position = new Vector3(35.811f, 12.6f, 38.9f);

}

Denne kode findes i PlayerController scriptet.

Et andet problem vi stødte ind i var, at skuddene ikke pegede parallelt med spillerens synsvinkel. Vi brugte nemlig cylinderformet skud til at starte med. Vi havde svært ved at løse dette problem ved hjælp af kode. Så vi ændrede skuddene til at være kugleformet i stedet. Skuddenes rotation bliver dermed ligegyldig, så vi slipper også for at forholde os til skuddenes rotation i koden.

Vi fandt også problemer i forbindelse med skaden taget fra zombierne, og regenerationen af livspoint. Hvis vi brugte void OnCollisionEnter til at bestemme hvornår spilleren skal tage skade, tager man kun skade idet zombien rammer en, og ikke mere. Derfor brugte vi void OnCollisionStay, da den bliver ved med at tjekke om zombien rører spilleren. Dog gør den dette rigtigt mange gange i sekundet. Derfor dør man på få millisekunder efter man rører en zombie. Det fiksede vi med en lille timer.

float currentTime = 0;

float nextTime = 1;

void OnCollisionStay(Collision other)

{

if (currentTime <= 0 && other.gameObject.CompareTag("Enemy"))

{

currentHealth -= 5 \* difficulty;

currentTime = nextTime;

}

else

{

currentTime -= Time.deltaTime;

}

}

Her siger vi at idet zombien og spilleren rører hinanden, skal spilleren tage 5 livspoint skade, som så kan ændres af sværhedsgraden. Når det er sket, starter en timer som tæller ned fra 1 sekund. Når den timer når 0 sekunder, skal spilleren igen tage skade, og timeren startes igen.

På en lidt anden måde regenererer man liv. Når man har mindre end det maksimale liv sværhedsgraden tillader, skal der tilføjes 5 livspoint pr. sekund, efter 5 sekunder. Vi fik så det problem at den startede rigtigt mange Coroutines i de 5 sekunder man manglede liv. Det resulterede så i at man endte med over 1000 livspoint, og det går jo ikke. Dette fiksede vi ved at tilføje en bool der hedder “healing”. Denne tilstand blev slået til inden der regenereres, og der kan dermed kun startes en enkelt Coroutine, indtil man har fuldt liv igen. Når man så har fuldt liv igen, blev “healing” tilstanden slået fra igen, og der kan nu regenereres næste gang man mister livspoint.

public double healthPoints = 100;

public double currentHealth;

private double difficulty;

public bool healing;

void Update()

{

if (currentHealth < (healthPoints / difficulty) && !healing)

{

StartCoroutine(RegenHealth());

}

}

IEnumerator RegenHealth()

{

healing = true;

yield return new WaitForSeconds(5);

while (currentHealth < (healthPoints / difficulty))

{

currentHealth = currentHealth + 5;

yield return new WaitForSeconds(1);

}

healing = false;

}

# Brugervenlighed

Vores spil er designet til målgruppen af mennesker der ikke har spillet så mange zombie spil før. Ideen med spillet er at vække interesse for spil som ikke kan vindes, men hvor det gælder om at overleve så lang tid som muligt. Fordi alle personer starter ved forskellige færdighedsniveauer, har vi tilføjet muligheden for at ændre på sværhedsgraden. Derved kan både folk som har spillet FPS spil før, og folk der er helt nye, alle spille uden at synes at det er for nemt eller svært. Da det er et relativt simpelt spil, har vi tilføjet musik for at man ikke keder sig så hurtigt.

# Udvidelse

Der er mange måder hvorpå vi kan videreudvikle vores spil. Til at starte med kunne vi sørge for at sværhedsgraden påvirker flere ting, som løbehastigheder eller hvor hurtigt man regenererer. Dette ville gøre at man kan mærke større forskel på de forskellige sværhedsgrader. Man ville derfor skulle overveje sine eventuelle strategier når man rykker en sværhedsgrad op.

Hvis man ville bygge endnu mere på spillet, kunne man gøre så at der er en lille chance for at hver zombie udløser en power-up. Dette ville gøre spillet mere varieret og spændende.

En anden mulighed ville være at gøre, så zombierne havde en døds-animation inden de forsvandt.

En lidt mere krævende addition ville være at tilføje rekyl på våbnene, så de bevæger sig når der skydes. Det er muligt at få individuelle dele på våbnene til at bevæge sig, så bolten i løbet kan flyttes. Udover dette kan man også få magasinerne til at tages ud og puttes ind igen. Denne animation kunne tilføjes, når der reloades.

Hvis man ville gå lidt længere væk fra nybegynder-målgruppen, kunne man tilføje et point system, så når man skyder en zombie, optjener man point. Disse point kunne bruges til at opnå bedre våben, som skyder hurtigere eller skader mere. Man kunne også bruge pointene til at købe permanente powerups. Men denne slags store udvidelse ville muligvis være for meget for en person som er helt ny til zombie-genren. Dog ville det gøre spillet meget mere dynamisk og interaktivt.

# Konklusion

Ud fra alt dette kan vi konkludere at vi har lavet et spil der fokuserer på målgruppen der enten er nye til computerspil, eller bare FPS zombie-genren. Ved hjælp af kode har vi fået zombierne til at bevæge sig mod spilleren. Vi har også fået zombien til at registrere hvornår den bliver skudt, og tager skade derefter. I løbet af udviklingsprocessen har vi stødt på nogle problemer som for eksempel rotationen af skuddene og skaden taget fra zombierne. De blev dog løst med henholdsvis et andet 3D-objekt, og en timer. For at flest muligt mennesker kommer til at have en god oplevelse med spillet, har vi tilføjet forskellige sværhedsgrader. Da vores projekt sigter efter at vække interesse indenfor zombie genren, ville det jo ikke være ideelt hvis spilleren synes at det enten er alt for nemt eller svært. Da det ville blive kedeligt meget hurtigt. Vi har overvejet flere forskellige videreudviklinger, som kunne øge spillets dynamik og spændingsniveau. Nogle kan dog ødelægge ideen med at have nybegyndere som målgruppe.

1. <https://assetstore.unity.com/packages/3d/vegetation/environment-pack-free-forest-sample-168396> [↑](#footnote-ref-0)
2. <https://assetstore.unity.com/packages/3d/characters/humanoids/character-pack-free-sample-79870> [↑](#footnote-ref-1)
3. <https://assetstore.unity.com/packages/3d/characters/humanoids/fantasy/character-pack-zombie-sample-131604> [↑](#footnote-ref-2)
4. <https://assetstore.unity.com/packages/3d/props/guns/low-poly-weapons-vol-1-151980> [↑](#footnote-ref-3)