|  |
| --- |
|  |
| Synopse - Pygame |
| Real-estate board game made with Pygame |

|  |
| --- |
| Oliver Mikkel Opfermann Göth  31-05-2024 |

Indholdsfortegnelse

[Introduktion 2](#_Toc167224982)

[Problemformulering 3](#_Toc167224983)

[Arbejdsspørgsmål 3](#_Toc167224984)

[Metode 3](#_Toc167224985)

[Planlægning 3](#_Toc167224986)

[Research 3](#_Toc167224987)

[Tidsplan 4](#_Toc167224988)

[Pygame 4](#_Toc167224989)

[Grundlæggende komponenter 4](#_Toc167224990)

[Grundlæggende Introduktion til Pygame 5](#_Toc167224991)

[Installation 5](#_Toc167224992)

[Grundlæggende Struktur 6](#_Toc167224993)

[Lydhåndtering 7](#_Toc167224994)

[Animation 7](#_Toc167224995)

# Introduktion

Velkommen til et spændende 4. semester projekt, hvor vi vil udforske og lære gennem udviklingen af et spil i Python ved hjælp af Pygame-frameworket. I dette projekt vil vi tage inspiration fra det klassiske brætspil Monopoly og skabe vores egen digitale version af det.

Formålet med dette projekt er ikke kun at implementere et spil, men også at dykke ned i en række koncepter og teknologier inden for softwareudvikling og spiludvikling. Vi vil udforske emner såsom objektorienteret programmering, brugergrænsefladedesign, spillogik, algoritmer til spilfunktioner og meget mere.

Gennem dette projekt vil du have mulighed for at lære og anvende nye færdigheder og teknikker, samtidig med at du styrker din forståelse for Python-programmering og softwarearkitektur. Projektet vil give dig mulighed for at udforske og eksperimentere med forskellige løsninger og tilgange, samtidig med at du arbejder på at skabe et spil, der er sjovt og engagerende for brugerne.

Du vil blive opfordret til at undersøge og implementere unikke funktioner og mekanikker i spillet, der adskiller det fra den traditionelle Monopoly-oplevelse og giver det sit eget særpræg. Dette kan omfatte tilføjelse af nye regler, specialfelter, karakterer eller endda integration af online multiplayer-funktionalitet.

Dette projekt vil være individuelt, hvilket betyder, at du har frihed til at udforske og fordybe dig i de områder af spiludvikling, der interesserer dig mest. Gennem projektets forløb vil der være løbende vejledning og støtte til rådighed for at hjælpe dig med at nå dine mål og skabe en imponerende digital udgave af Monopoly.

Vi ser frem til at se, hvad du vil opnå med dette spændende projekt, og hvordan du vil udforske og lære gennem udviklingen af dit eget spil i Pygame. Lad os gå i gang og skabe noget fantastisk sammen!

# Problemformulering

I dette projekt vil jeg undersøge og udforske processen med at udvikle et digitalt brætspil ved at skabe min egen version af Monopoly ved hjælp af Python og Pygame-frameworket.

## Arbejdsspørgsmål

1. Hvad er Pygame, og hvilke grundlæggende komponenter består det af?
2. Hvordan bruger man Pygame? (animation og lyd)
3. Hvordan håndteres brugerinput i Pygame for at styre spillerfiguren?
4. Hvordan kan man lave et brætspil/ brætspil bane i Pygame?
5. Hvordan implementeres spillogikken, herunder scoring og afslutning af spillet?
6. Hvordan kan man lave en simple AI, til dette projekt

# Metode

Måden hvorpå jeg ønsker at finde svar på ovennævnte problemformulering, vil primært foregå igennem researcharbejde. Denne research afgrænser til bl.a. læsning af relevante dokumentation at se videovejledninger samt selv at udvikle egne eksperimenter. For at sikre sig kildernes korrekthed og relevans, vil jeg – så vidt det er muligt arbejde primært med den respektive teknologiske officielle dokumentation.

# Planlægning

Planlægning- og struktureringsmæssigt har jeg bevidst valgt ikke at anvende nogen bestemte anerken

# Research

# Tidsplan

# Pygame

Pygame er en bibliotekspakke til Python, der primært anvendes til udvikling af spil og multimediaprojekter. Det giver udviklere et sæt værktøjer og funktioner til at oprette interaktive applikationer, især inden for 2D-spiludvikling. Pygame er baseret på Simple DirectMedia Layer (SDL), hvilket gør det platformuafhængigt og understøtter forskellige operativsystemer som Windows, macOS og Linux.

Med Pygame kan udviklere oprette vinduer, håndtere brugerinput som tastatur og mus, tegne grafik og manipulere billeder, afspille lyd og musik, håndtere kollisioner mellem spilobjekter, og meget mere. Det er et alsidigt værktøjssæt, der gør det muligt for både begyndere og erfarne udviklere at skabe alt fra enkle arkadespil til mere komplekse simuleringer.

Da Pygame er baseret på Python, gør det sprogets syntaks og struktur let tilgængelig for udviklere, hvilket gør det relativt nemt at lære og bruge. Samtidig giver det også mulighed for integration af Python's omfattende økosystem af biblioteker og værktøjer, hvilket giver udviklere stor fleksibilitet og kontrol over deres projekter.

## Grundlæggende komponenter

Pygame er et værdifuldt bibliotek til Python, der åbner døren for spiludvikling og multimediaapplikationer. Som en del af din undersøgelse skal du forstå de grundlæggende komponenter i Pygame. Lav en detaljeret beskrivelse af hver komponent nedenfor:

1. pygame.display: Dette modul styrer spilvinduet og skærmen, hvor spillet vises. Beskriv, hvilke funktioner det giver, og hvordan det håndterer interaktioner med brugeren.

Eksempel på pygame.display, som jeg bruger i min kode:

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, linje/række

Automatisk genereret beskrivelse

1. pygame.Surface: En vigtig del af Pygame er Surface, der fungerer som et lærred til at tegne grafik på. Forklar, hvordan det bruges til at håndtere grafik i spillet.
2. pygame.event: Gennemgå betydningen af dette modul i forbindelse med brugerinput. Hvordan kan udviklere bruge det til at lytte efter og reagere på forskellige begivenheder?
3. pygame.image: Diskutér vigtigheden af billedhåndtering i spiludvikling og hvordan dette modul gør det muligt at indlæse og manipulere billeder.



1. pygame.sprite: Hvad er formålet med sprites i Pygame, og hvordan hjælper dette modul udviklere med at organisere og styre spilobjekter?
2. pygame.mixer: Beskriv, hvordan lyd er integreret i spil med dette modul, og hvad det tilbyder i form af funktionalitet.
3. pygame.font: Hvordan understøtter dette modul tekstbehandling i Pygame-applikationer? Forklar, hvordan det bruges til at tegne tekst på skærmen.
4. pygame.time: Diskutér vigtigheden af tidsstyring i spiludvikling og hvordan dette modul hjælper udviklere med at kontrollere spilhastighed og tidsrelaterede begivenheder.

For hver komponent, skal din beskrivelse være detaljeret og give en klar forståelse af dens funktioner og anvendelse i Pygame. Husk at illustrere dine svar med relevante eksempler, hvor det er relevant.

# Grundlæggende Introduktion til Pygame

Pygame er et kraftfuldt bibliotek til at lave spil og multimedieapplikationer i Python. Her er en grundlæggende vejledning til at komme i gang med animation og lyd i Pygame.

## Installation

Først skal du installere Pygame. Dette kan gøres med pip:



## Grundlæggende Struktur

Pygame applikationer har typisk en hovedsløjfe, der håndterer begivenheder, opdaterer spillets tilstand og tegner alt på skærmen.

1. **Initialisering**: pygame.init() initialiserer alle Pygame-moduler.

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede, Grafik

Automatisk genereret beskrivelse

1. **Skærmindstillinger**: pygame.display.set\_mode() opsætter skærmens bredde og højde. pygame.display.set\_caption() sætter vinduets titel.

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede

Automatisk genereret beskrivelse

1. **Farver**: Farver defineres som RGB-værdier.

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

1. **Animation**: Position og hastighed for en rektangel animeres.
2. **Lyd**: pygame.mixer.init() initialiserer Pygame's lydmodul, og pygame.mixer.Sound() indlæser en lydfil.
3. **Main Loop**: Håndterer begivenheder, opdaterer animation og tegner alt på skærmen. Den bruger også pygame.time.Clock().tick(60) for at begrænse spillet til 60 FPS.

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

## Lydhåndtering

For at arbejde med lydfiler, skal du sørge for at have korrekte formater (som WAV for lyd). Her er nogle grundlæggende funktioner:

1. **Afspilning af lyd**: sound.play()
2. **Stop af lyd**: sound.stop()
3. **Indstilling af volumen**: sound.set\_volume(0.5) (0.0 til 1.0)

## Animation

For at animere flere objekter, kan du bruge klasser til at organisere spritesende: