

Pygame játékfejlesztés MI támogatással

Nits

László

Nits László küldte be 2026. 04. 13., h - 14:52 időpontban

Terv típusa

Projektterv

Bővebb szakterület

Információs és kommunikációs technológiák

Témakör, tanulási terület

Objektumorientált programozás (OOP) és külső könyvtárak (Pygame) használata
MI segítségével

Szakmai tantárgy(ak)

Programozási alapok

Évfolyam

10. évfolyam

Tanulási, fejlesztési célok

A tanulók képesek legyenek egy komplex Pygame projektet strukturálni. Az OOP elveket alkalmazni játékobjektumokon. Az MI eszközök tudatos használata új könyvtárak megismerésére és hibakeresésre. Csoportmunka és verziókezelés gyakorlása

Fogalmak

pygame, oop, python, AI, clean coding, verziókezelés, type hinting

Szükséges eszközök

Számítógép, VS Code editor, Python környezet, Pygame könyvtár, MI asszisztensek a kódmagyarázathoz és dokumentációhoz, GitHub a projektmenedzsmenthez és kódmegosztáshoz

Előre kiadott anyagok

Játékfejlesztés Pygame modul használatával lejátszási lista 23 oktató videóval:

[Játékfejlesztés - YouTube](#)

Bevezető rész, előkészítés

1. fázis: Bevezetés, előkészítés és tervezés (1-2 óra)

- Csoportok alakítása (3 fő) és a projekt (pl. Space Invaders) céljainak kitűzése.

- Az MI eszközök használatának etikai és technikai megbeszélése (hogyan kérdezzünk, hogy megértsük a logikát).
- A GitHub mintaprojekt ([GitHub - schmidtnandor/space-invaders · GitHub](https://github.com/schmidtnandor/space-invaders)) szerkezetének elemzése: Miért van szükség külön fájlokra (pl. `alien.py`, `player.py`, `config.py`)?

Megvalósítás részletes tervezése

2. fázis: Alapok és Osztályszerkezet (2 óra)

- A PyGame ablak inicializálása és a fő játékhurok létrehozása MI segítségével.
- Az alapvető osztályok vázának megírása **típusjelölésekkel** (pl. `def __init__(self, x: int, y: int) -> None:`).
- A grafikai elemek beimportálása és megjelenítése a képernyőn.

3. fázis: Játéklógika és Interakciók (3-4 óra)

- A mozgás (billentyűzetkezelés) és a lövés mechanikájának implementálása.
- Ütközésdetektálás (Collision detection) kidolgozása a `pygame.sprite` modul segítségével, MI támogatással.
- A "tisztta kód" felülvizsgálata: beszélő változónevek, rövid metódusok, redundancia kerülése.

4. fázis: Finomhangolás és Dokumentáció (1-2 óra)

- Menürendszer, pontszámítás és "Game Over" képernyő hozzáadása.
- A kód kommentezése és a `README.md` fájl elkészítése a mintaprojekthez hasonlóan.
- Tesztelés és hibajavítás.

Tananyagkészítéshez és az órai munkához használt alkalmazás

ChatGPT

Copilot

Gemini

Google Tanterem

Google Meet

Google NotebookLM

Moodle

Értékelési terv

Az értékelés nemcsak a kész termékre, hanem a folyamatra is kiterjed, a projekt során folyamatosan értékelem a diákokat a következő szempontokon keresztül:

- **Technikai megvalósítás:** Működik-e a játék? Megfelelő-e az OOP struktúra?
- **Kódminőség ellenőrző lista:** (Isd. mellékletek)
- **MI használat:** Hogyan segítette az MI a tanulást?
- **Együttműködés:** A csoporttagok közötti munkamegosztás hatékonysága.
- **Prompt napló** értékelése (Isd. mellékletek)
- Meglévő és új ismeretek alkalmazása.
- Hibakeresés és javítás, tesztelés.

Differenciálás

- **Erősebb csoportok:** Bonyolultabb ellenség-minták (pl. Boss harc), részecske-effektek vagy többszintű pályarendszer implementálása.
- **Segítségre szoruló:** Előre elkészített sablonok (pl. egy működő játékhurok vagy konfigurációs fájl) biztosítása, szorosabb mentori támogatás az MI-vel való promptolásban.

Feladatok

Otthoni feladatok, részfeladatok megoldása a mentor segítségével és együttműködésével.

Közismereti tantárgy(ak)
informatika, IKT