

Revolution Prezentáció

Csapat tagjai:

- Hortobágyi Balázs:
LJZMWA
- Vereczkei Benjámín:
SGLGPX
- Rajnai Gábor:
C9VLNK



A Játékról röviden:

***RTS* - Real-time Strategy, valós idejű stratégiai játék.**

Játék célja:

A játék célja a folyamatos progresszió és talpon maradás a fejlődés közben.

Cél a túlélés és az ellenfél bot kiiktatása

Feladatkörök:

Balázs:
Fullstack



Beni:
Backend



Gábor:
Frontend



A játék **fundamentális alapjait** alkotta meg.
Megjelenítő/generáló osztályok megalkotása.

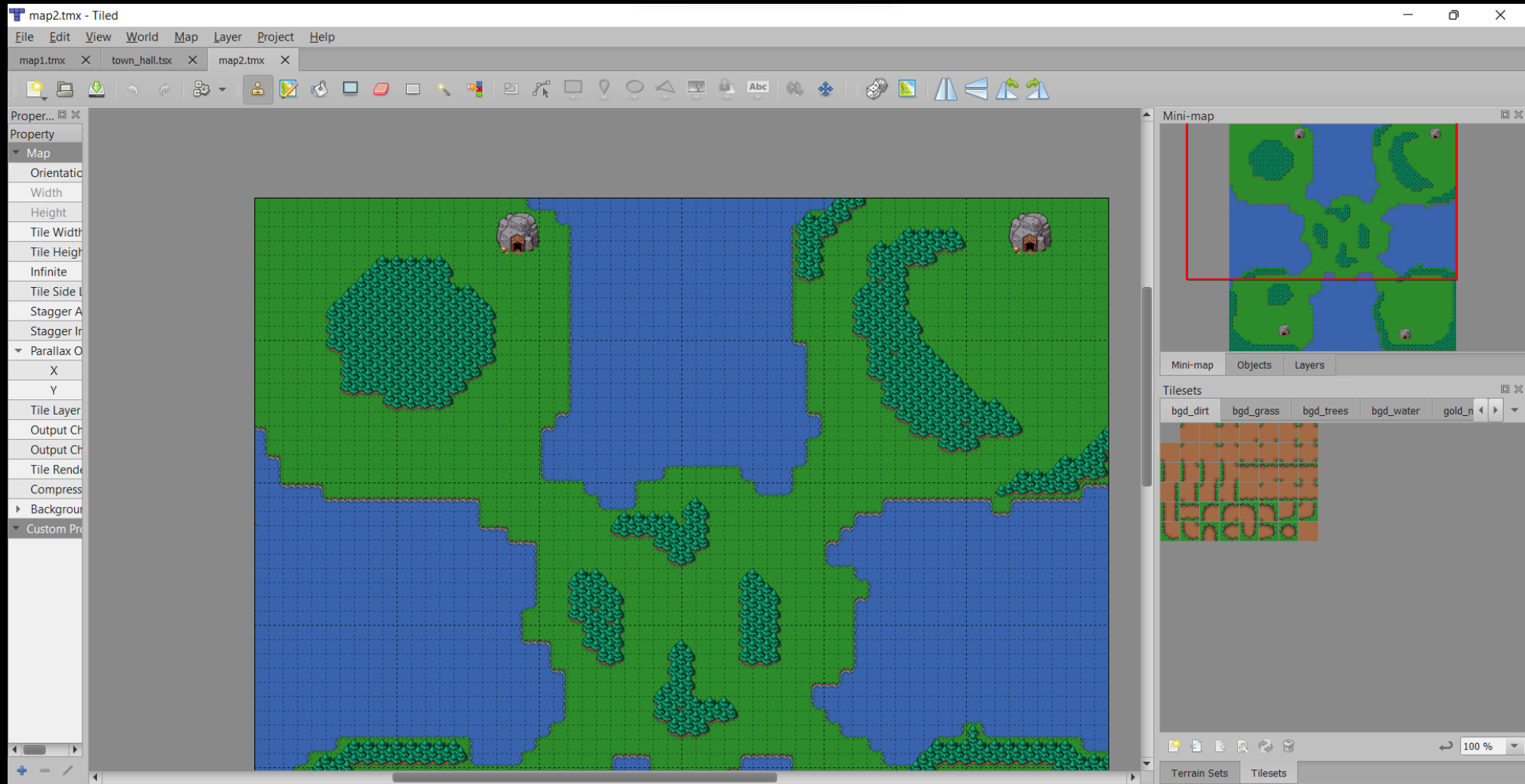
Az **AI, PathFinding** és a bonyolultabb összetett algoritmusokért felelt a csapatban.

A **menu megjelenésért** és a játékon belüli **grafikus** elemekért felelt.
Egyedi soundtrack volt a terv.

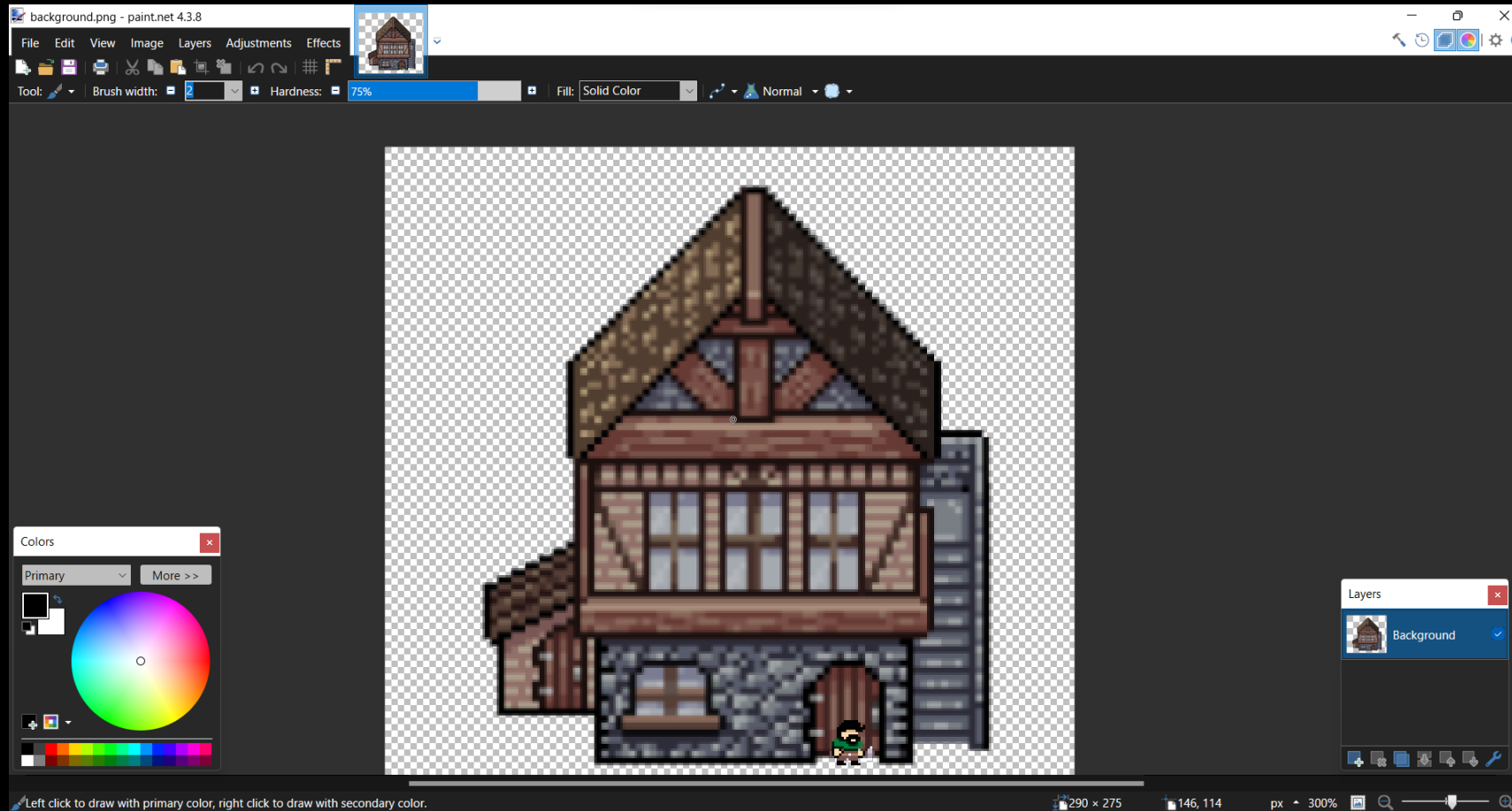
Felhasznált
módszertanok:



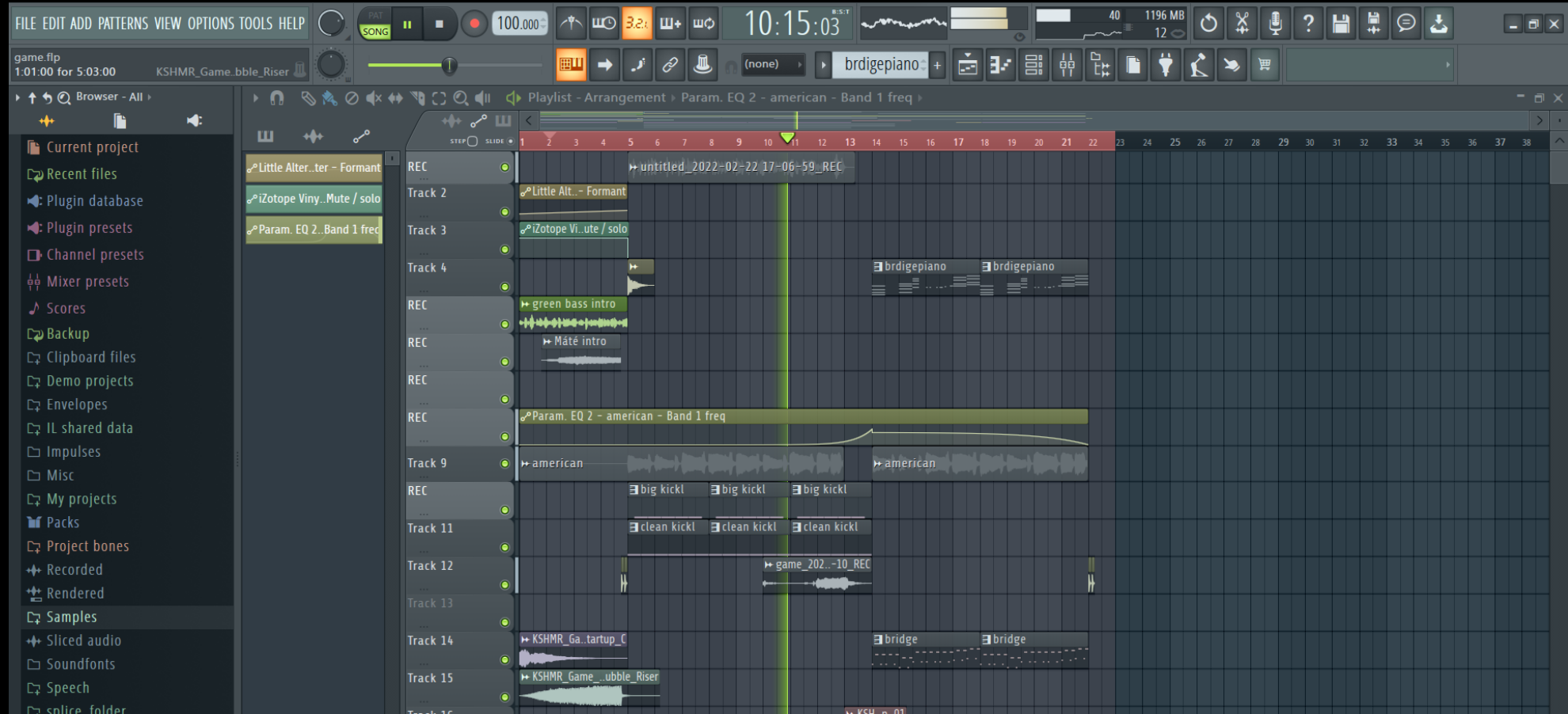
Tiled: Map készítéshez használt alkalmazás



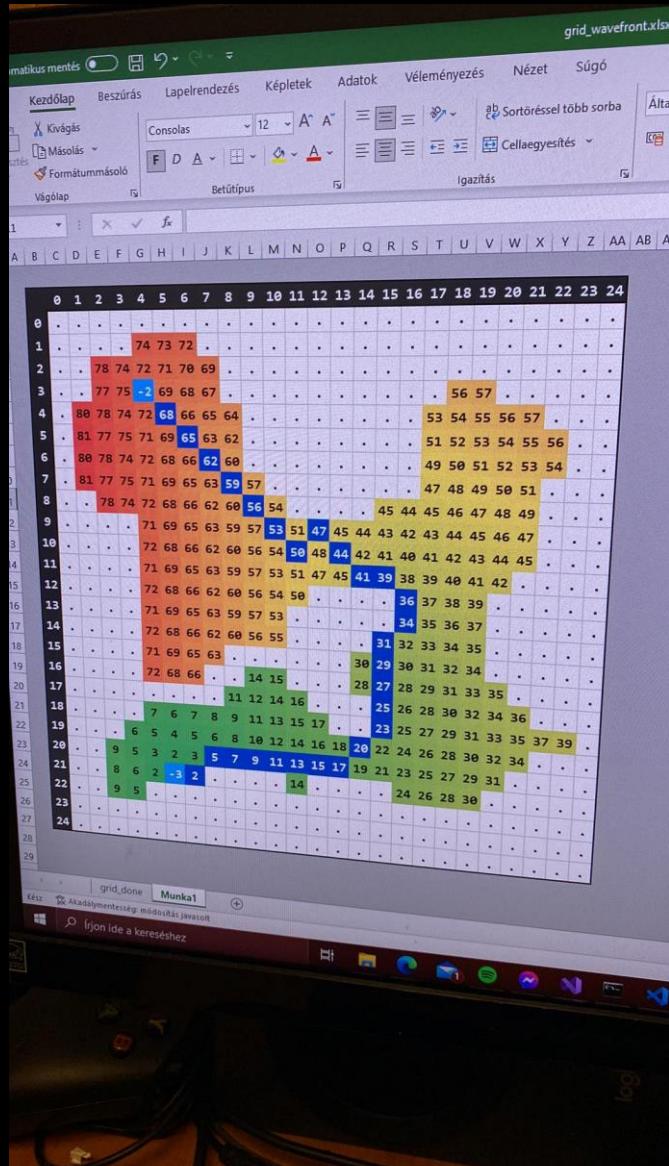
Paint.Net: Asset készítéshez használt alkalmazás



FL Studio: Zenei elemekhez használt DAW (digital audio workstation)



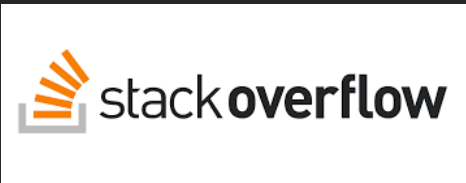
Pathfinding: Wavefront Generation



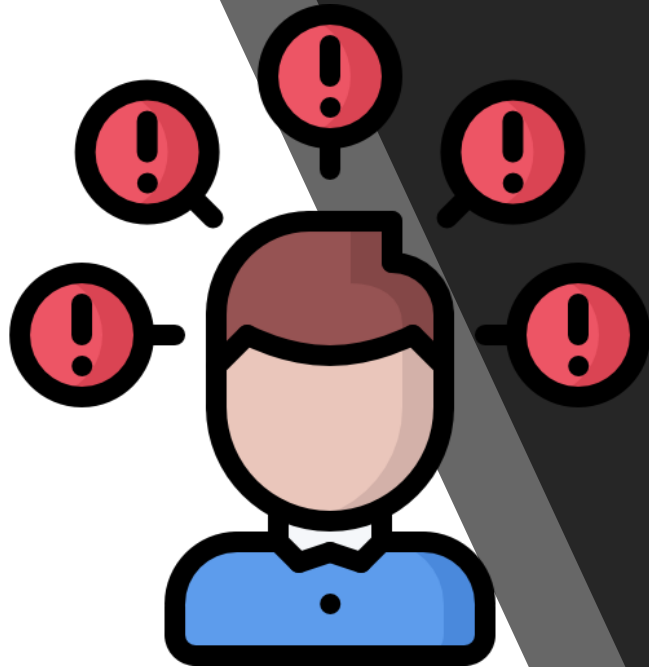
- Optimális útvonal megtalálásához használt algoritmus
- A tanulmányaink alatt **"Intelligens Rendszerek"** nevű tantárgy keretein belül találkozhattunk a technológiával.

Technikai/szakmai problémák:

- **A kód gyorsabb futása érdekében történtek optimalizációk**
- **Rengeteg apró hiba és bug (Kód csiszolása)**
- **A legtöbb problémára megoldást nyújtottak a következő oldalak:**



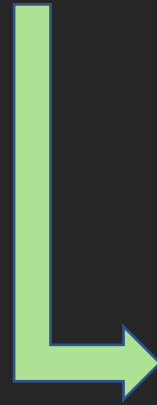
Emberi problémák:



- **Egyeztetések** – Közös munka összehozása
- A játék fejlesztése mekkora **prioritást** kapott az félév során?
- Lehet e **igazán jó** játékot **3-an fél év alatt** fejleszteni **wpf-**ben?
- **Motiváció / idő hiánya**
- Folytonos fejlesztés kimaradása (progresszív terhelés elmélete)

Értékelés / Konklúzió:

- A féléves project megvalósításának aránya: **70%**
- Egyértelműen túlvállaltuk magunkat a feladattal.
- A jövőben nagyobb projektek előtt elengedhetetlen az alapos tervezés



Szoftvertervezés egy lehetséges menete

43

1. Deployment diagrams

- A rendszer hogyan és milyen körülmények között lesz használva

2. Behavioral diagrams

- Milyen funkciókat kell a rendszernek tudnia?
- Use-case + Activity + Wireframes
- Megrendelővel közös tervezés

3. Structural diagrams

- A működést milyen modulokkal, milyen felbontással lehet megoldani
- Component + Sequence → Class diagrams
- Entity-relations diagram → DB tábla struktúrák

4. Időtervezés → Gantt diagram



Köszönjük a figyelmet!

Revolution Csapata



Prog 4 Boss

