

SpaceRTS

Patrik Dereh i Josip Šime Zrilić
1.5, Gimnazija Andrije Mohorovičića Rijeka
Profesor: Goran Boneta
08.05.2020

Sadržaj

1	Uvod	2
1.1	RTS igre	2
1.2	Ideja	2
1.3	O Autorima	2
1.4	Izrada projekta	2
2	Detaljan opis rada	3
2.1	Main Menu	3
2.2	Mapa	4
2.3	Kontrole	5
3	Tehničke informacije	6
3.1	Sistemska konfiguracija	6
3.1.1	Minimalna sistemska konfiguracija	6
3.1.2	Testirana sistemska konfiguracija	6
3.1.3	Preopraćena sistemska konfiguracija	6
3.2	Korištene tehnologije	7
4	Održivost	8
4.1	Planovi za budućnost	8
A	Priložene slike	9

1 Uvod

1.1 RTS igre

Strateške igrice u stvarnom vremenu(RTS) su relativno popularan žanr igrica u kojem jedan ili više igrača gradi carstvo I vojsku kako bi pobjedio protivnika bio taj protivnik drugi igrač ili umjetna inteligencija, ali kako RTS igre moraju imati relativno jednostavan dizajn kako nebi bile zamorne ,a mi smo ljubitelji igrica sa kompleksnjim igračim mehanikama mi smo odlučili kombinirati elemente RTS igrica sa elementima igrica 4X žanra koje generalno imaju kompleksnije mehanike.

1.2 Ideja

Ideju smo smislili jednog ponedjeljka za vrijeme sata povijesti kada smo krenuli malo “razbacivati” ideje. Cilj igrice je preuzeti vlast nad galaksijom ekspanzijom carstva te planetskim I svemiriskim bitkama, Igrač može igrati kao bilo koje od četiri glavna carstva od kojih svako ima svoje posebne mehanike I specijalizaciju. Zhui imaju unaprijed izgrađenu bazu te se primarno bave trgovinom, Scavangeri nemaju snažne razvijene baze niti mogu istraživati nove tehnologije, ali zato mogu sklapati tehnologije drugih carstava jedne s drugima kao I pronalaziti neke davno izgubljene tehnologije ,MSEA je carstvo robova koji svoju populaciju moraju proizvoditi te oni nemaju potrebu za hranom, Ngubu su carstvo cija posebnost lezi u manipulaciji gena, oni mogu svoju populacije genetski manipulirati kako bi se usavršili, a dostupna su im I moćna bio-oružja.

1.3 O Autorima

Mi smo učenici 4.5 razreda Gimnazije Andrije Mohorovičića I odlučili smo napraviti kompleksnu igricu jer je želimo jednog dana monetarizirati na Steam platformi.

1.4 Izrada projekta

Izrada projekta je započela 8.3.2020. Za programski jezik smo odabrali C# zbog njegove brzine I jednostavnosti objektno orijentiranog programiranja u njemu zbog njegova “multiplatmorfiteta”. Ovaj projekt predstavlja znatan izazov zbog njegove kompleksnosti stoga je još uvijek u razvoju.

2 Detaljan opis rada

2.1 Main Menu

Sadrži opcije:

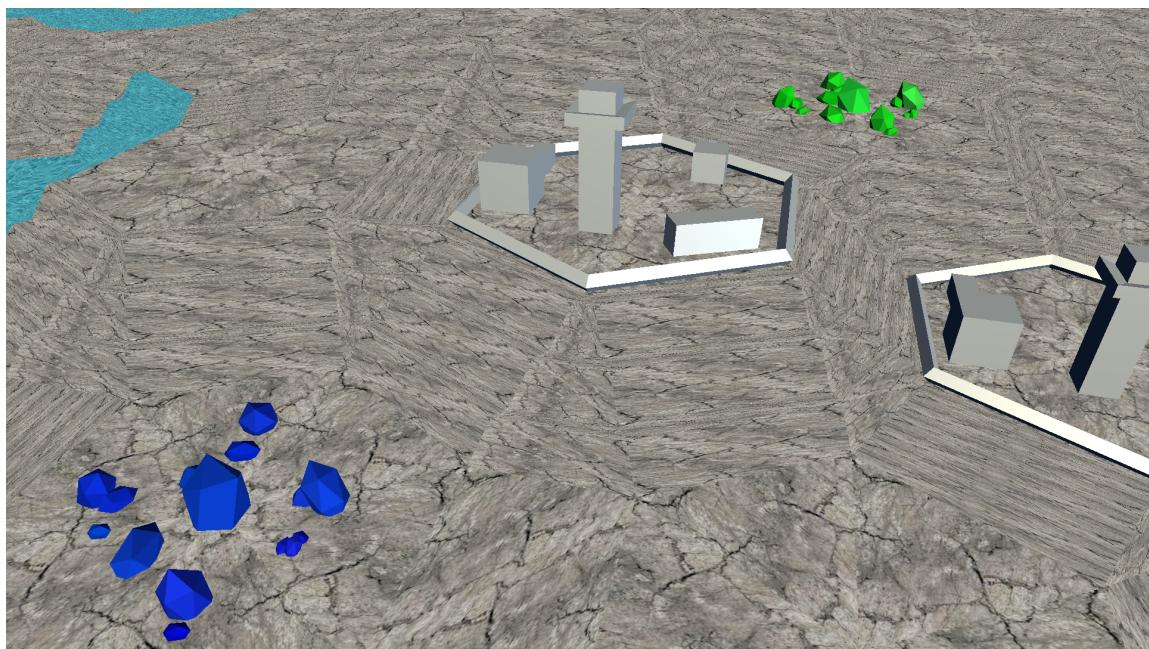
- “play” - pokreće testnu verziju igrice
- “exit”- gasi program
- “settings”- ništa jer postavke nisu još dostupne



Slika 1: Main Menu

2.2 Mapa

Teren se generira uz pomoć Perlinovog šuma, algoritma koji generira nasumične vrijednosti koje zajedno imaju prirodnog smisla tj. vrijednosti nisu u potpunosti nasumične nego svaka vrijednost djelomično ovisi o susjednim vrijednostima. Teren je izgrađen od mreže šesterokutnih čelija. Različiti planeti imaju različitu učestalost i amplitudu visinskih promjena kao i različitu učestalost planina, veličine planina i biome. Nasumično se generiraju izvori rude koji se razlikuju po rijetkosti- zeleni(najčešći),plavi(srednje čest),crveni(naj rijedji). Klikom na teren igrač može postaviti testnu građevinu.



Slika 2: Mapa, zelena i plava ora, te postavljene testne strukture

2.3 Kontrole

- WASD tipke kontroliraju kameru
- Esc terminira program
- Lijevi klik postavlja građevinu u čeliju

3 Tehničke informacije

3.1 Sistemska konfiguracija

Igra je iznimno optimizirana, te ne zahtjeva jako računalo za pokrenuti

3.1.1 Minimalna sistemska konfiguracija

- Ryzen 3 1200
- 4GB RAM-a
- Nvidia GTX 820M
- 256MB slobodnog prostora
- os: bilo koji 32 ili 64 bitni

3.1.2 Testirana sistemska konfiguracija

- Intel Core i5-9300H
- 16GB RAM-a
- Nvidia GTX 1660 Ti
- 256MB slobodnog prostora
- os: Arch Linux

3.1.3 Preopručena sistemska konfiguracija

- Intel Core i5-9300H
- 16GB RAM-a
- Nvidia GTX 1660 Ti
- 256MB slobodnog prostora
- os: Arch Linux

3.2 Korištene tehnologije

Korištene tehnologije su brze, sigurne i pouzdane, te je većina njih otvorena koda.

- Visual studio code
- Monodevelop
- C#
- Mono
- HLSL, DirectX
- Monogame
- Blender



Slika 3: Korištene tehnologije

4 Održivost

4.1 Planovi za budućnost

U narednim mjesecima igra će dalje napredovati, te bi do kraja ljeta trebala biti u potpunosti gotova.

4.2 Dokumentacija koda

Kod je dobro dokumentiran - zbog korištenja tehnologije doxygen, te je iznimno samo-objašnjiv, pružajući nam mogućnosti lagnih i brzih unapređenja i popravaka.

A Priložene slike

1	Main Menu	3
2	Mapa, zelena i plava ora, te postavljene testne strukture	4
3	Korištene tehnologije	7