

Nevarna voznja

Gašper Smerkolj¹, Gregor Berger²

¹Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

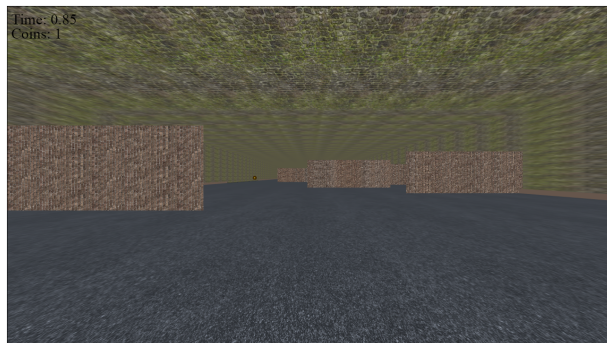
²Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

E-pošta: gs3493@student.uni-lj.si

Abstract

Pri izdelavi igre za seminarsko nalogo pri računalniški grafiki imava namen uporabiti okolje WebGL ter programski jezik Javascript. Igra bo zabavala igralca z samo težavnostjo vožnje avtomobila mimo ovir ter z zbiranjem kovancev katere bo igralec lahko uporabil. Če bo imel dovolj zbranih kovancev bo namreč lahko zapeljal čez ovire, če pa zapelje preko ovire brez zadostnega števila za določeno oviro se gre igralec na začetek igre. Vsaka različna ovira bo ob trku odvzela določena števila kovancev. Dlje časa kot traja 1 igra težje postaja izogibanje ovir.

1 Pregled igre



Slika 1: Igled igre

Odločila sva se da bova izdelala dirkaško igro2, katere je glavni namen izogibanje oviram, ter zaključiti progo z, čim več kovanci ter z daljšim časom saj čimveč kovancev kot jih pobereš dalj časa lahko igraš. Zahtevnost igre se povečuje na vsakih 20 procentov dolžine proge z povečanjem hitrosti prav tako za 20 procentov. Igra je namenjena igralcem vseh starosti. Igra se izvaja na ravni površini, katera je dolžine 900 (z: 0, z:900), ter širine 10 (x: -5,5). Objekti so v večini sestavljeni iz ravnih površin, razen kovanci kateri so krogi z robom 0.06.

1.1 Opis sveta

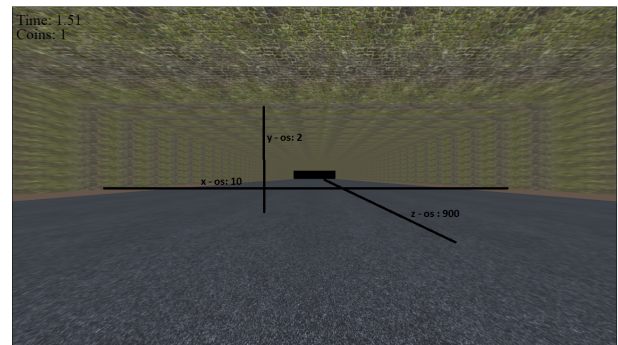
Na tem mestu podajte grob opis sveta v igri, ki ga podrobneje definirate v sledečih podpoglavjih. Prav tako definirajte v kakšnem stilu bo izdelan svet (npr. realističen, stiliziran, risankast, ipd.). Opredelite tudi ali se

bodo osebk v svetu pomikali v eni, dveh ali treh dimenzijah.

1.1.1 Pregled

V najinem svetu sva naredila objekte s katerimi lahko interagiramo. Ti objekti so na progi postavljeni naključno tako da se proga nikoli ne ponavlja. Tako je pa lahko tudi sama zahtevnost čisto naključna. Pri zidovih, če se igralec zaleti v njih, se odbije en kovanec in če je kovancev manj kot 1 je tako konec igre. Na progi imamo tudi kovance s katerimi lahko upočasnimo vozilo, ko nabereimo 5 kovancev se nam hitrost pomanjša za 10 procentov.

1.1.2 Ozadje



Slika 2: Velikost tunela

Igra se izvaja v tunelu2, v katerim ima uporabnik občutek »ujetosti«. Strop ter stene so narejene iz opek, katere spominjajo na zelo staro gradnjo tunela. Robniki so pa zgrajeni iz rjavih opek.

1.1.3 Ključne lokacije

Ključne lokacije so stene, robniki, zidovi, kovanci, ter vsaka petina proge. Stene so postavljene v svetu na točkah x:5.1 ter x:-5.1, robniki so postavljeni na x:5.0 ter x:-5.0, z robniki tudi vozilo iteratira, tako da nemore od te točke naprej po x-u. Zidovi ter kovanci so postavljeni v svet naključno po x osi, npr. zidovi se pojavijo vsakič ko se z os, poveča za 5. Kovanci pa ko se z os poveča za 18. Na vsaki petini proge pa se prav tako poveča hitrost. Ko pridemo do konca proge (z: 900), se igra konča z dialogom (»You won!«)

1.1.4 Velikost

Svet je predstavljen v tunelu, kateri je dimenzij 10x2x900 (x,y,z). Uporabnik pa interaktira samo po x osi.

1.1.5 Objekti

Celi svet sva izdelala sama, zidovi so izdelani iz dveh trikotnikov kateri so širine 3. Kovanci so narejeni iz polmera 0.15 ter iz 100-ih trikotnikov, na ta trikotnik je še na oglišča poslavljen pravokotnih iz sosednjih oglišč tako dobimo zaobljen rob in celota spominja na kovanec. Stene, robniki, cesta so pa prav tako sestavljeni iz trikotnikom opisanih že v prejšnjih poglavjih.

1.1.6 Čas

Najina igra se odvija v realnem času.

1.2 Igralni pogon in uporabljene tehnologije

Za seminar sva uporabila WebGL¹², HTML, CSS, JavaScript. Z HTML-jem, ter CSS-jem sva si pomagala za izpisovanje časa, ter števila kovancev na canvas, prav tako pa za uporabniški meni. Za WebGL nisva uporabila nobene dodatne knjižnice. Za trke sva sama spisala zaznavanje, naredila sva ovoj glede na širino objekta, s tem sva tudi določila mejo pri kateri se trk zazna. Pri trkih sva si tudi pomagala z časovno omejitvijo, tako da se lahko trk zgodi samo vsakih 200 milisekund. Isto zaznavanje sva uporabila tudi za to, da vozilo negre izven proge.

1.3 Pogled

Kamera je v prvi osebi in ima vidno polje 45 stopinj pred seboj. Vidi pred seboj od 0.1 pa do 950 enot.

2 Osebek

Uporabnik bo igral vozilo, v prvo osebno pogledu. Vozilo bo lahko premikal z A ter D, ter z levo in desno smerno tipko. Osebek se stalno premika naprej po progi in se nemore ustavljati. Nad drugimi objekti nima nadzora.

3 Uporabniški vmesnik

Uporabniški vmesnik predstavi pravila igranja, ter začetek igre. Med igro je viden napis za čas ter stanje kovancev. Če zmanjka kovancev se tudi pojavi napis »Game over«, ter igra se postavi na začetni meni. Uporabniški vmesnik sva implementirala s pomočjo HTML-ja ter CSS-ja. Takšen vmesnik sva izbrala, ker je preprost dizajn in ne obremenjuje samo igro.

4 Glasba in zvok

V igri je predstavljen samo zvok zagona motorja. Zvočni efekt sva pridobila z spletne strani freesoundeffect³.

¹<https://webglfundamentals.org/>

²https://web.archive.org/web/20171025022250/http://learningwebgl.com:80/blog/?page_id=1217

³<https://www.freesoundeffects.com/free-sounds/cars-10069/>

5 Gameplay

Igra se začne začne s pritiskom na »play« gumb. Začetna hitrost je nastavljena na 0.01, ta pa se tudi povečuje na vsako petino mape oziroma zniža za 10 procentov, če zberemo 5 kovančkov. Če jih toliko zberemo se tudi število kovancev zmanjša na 1, sepravi da smo relativno zmanjšali zahtevnost igre. Sam cilj igre je izogibanje ovir, ter varen prihod na cilj proge.

6 Zaključki in možne nadgradnje

Med samo izdelavo igre sva se naučila izdelovanje 3-d modelov, s pomočjo trikotnikov, delovanje filtrov na texture, pri zaznavanju trkov. sva morala biti previdna na koliko časa bo program spet zaznač trk, tako sva tudi rešila problem na preprost način. Razlika med