

# Oddaja seminarske dokumentacije - 1. seminar pri predmetu Računalniška grafika in tehnologija iger

Aljaž Kozina, Denis Kotnik, Kristian Žarn, Klemen Červ

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

E-pošta:

kozinc@gmail.com,  
denis.kotnik@gmail.com,  
kristian.zarn@gmail.com,  
klemen.cerv@gmail.com

## Abstract

*V tretjeosebni pustolovščini vodimo ladjo, polno zaklada mimo nasprotnikov do pristanišča, medtem ko se trudiš, da bi izgubil čim manj zaklada.*

## 1 Pregled igre

Naša igra je tretjeosebna pustolovščina, navdihnjena z igro Overboard![1], v kateri igraš pirata, ki želi zbrati čimveč zaklada, brez da bi mu drugi pirati potopili ladjo. V zaprtem svetu se glavni igralec izogiba drugim ladjam ali pa jih napada s svojim orožjem medtem ko išče zaklad.



Slika 1: Naša igra v WebGL.

## 2 Seminarska dokumentacija

*Dokumentacija naj bo v formatu PDF in naj vsebuje do 5000 znakov. V dokumentaciji opišite delo v okviru seminarja. Opišite uporabljene pristope in metode, ki ste jih uporabili pri izdelavi seminarja.*

### 2.1 Pregled

Pri izdelavi našega seminarja smo uporabili tehnologije WebGL, Three.js[2], itd. Za nalaganje 3D objektov smo uporabili OBJMTLLoader, za detekcijo tipkanja na tipkovnici smo uporabili knjižnico KeyboardState. Za prikaz vode smo uporabili shader, ki je vodi priredil animirano teksturo normal z namenom prikaza realistične vode. Pri razvoju naše igre smo si pomagali z uporabo GitHuba.

V trenutni različici igre igralec lahko pobira cekine in

strelja ladje - trenutno je cilj, da pobere čimveč cekinov, brez da bi se potopil. Te ladje se gibajo po vnaprej določeni poti in streljajo, če igralec pride v bližino.

### 2.2 Potek programa

V naši igri se ob naložitvi okna inicializirajo potrebne spremenljive za igro, kot so scena, renderer, elementi HUD(življenje in rezultat) ter poslušalci tipkovnice. V sceni so ustvarjene tri kamere, dve za pogled igralca, tretja pa za prikaz mini-zemljevida z uporabo viewporta.

Nato se v sceno naložijo ladje - ladja igralca, pa tudi vsi sovražniki, ki se trenutno gibajo po prednastavljenih tirih. S pomočjo OBJMTL loaderja nato naložimo kopno, ki igralcu omejuje igralni prostor (vse ladje in kopno z uporabo metode addCollisionObject registriramo kot objekte, ki se jim preverja trčenje). Dodano je tudi pristanišče, ki pa kot igralni predmet trenutno še nima pomena. Nato je po zemljevidu naključno posejanih 50 kovancev, ki jih igralec s pomočjo metode checkCollect pobira in si tako povečuje rezultat.

V sceno so nato naloženi viri svetlobe (AmbientLight, 2 Directional Light), shaderji in pa Skybox, ki igralcu olepšajo svet.

Ko se igra naloži, se igralec nato lahko premika, strelja krogle(Krogla.js) in pobira kovance.

### 2.3 Opis pomembnejših funkcij

- *updateHUD* - tukaj se posodobijo elementi HUD, trenutno življenje in kovanci
- *checkCollision* - tukaj preverimo trčenje med objekti
- *update* - tukaj posodabljam premikanje objekta in pohandlamo trčenje (v primeru ladje, ob trčenju z kopnim ali drugo ladjo, premik in rotacijo izničimo)

## Literatura

- [1] Wikipedia, "Overboard! — Wikipedia, the free encyclopedia." <http://en.wikipedia.org/w/index.php?title=Overboard!&oldid=679064544>, 2015. [Online; accessed 10-November-2015].
- [2] Wikipedia, "Three.js — Wikipedia, the free encyclopedia." <http://en.wikipedia.org/w/index.php?title=Three.js&oldid=686226385>, 2015. [Online; accessed 17-November-2015].